Methods in Biostatistics with R

Ciprian Crainiceanu, Brian Caffo, John Muschelli

Contents

1 Introduction **7**
- 1.1 Biostatistics . . . 9
- 1.2 Mathematical prerequisites . . . 13
- 1.3 R . . . 13
- 1.4 Data . . . 14

2 Introduction to R **15**
- 2.1 R and RStudio . . . 15
- 2.2 Reading R code . . . 16
- 2.3 R syntax and jargon . . . 17
- 2.4 Objects . . . 17
- 2.5 Assignment . . . 17
- 2.6 Data types . . . 19
- 2.7 Data containers . . . 19
- 2.8 Logical operations . . . 28
- 2.9 Subsetting . . . 29
- 2.10 Reassignment . . . 33
- 2.11 Libraries and packages . . . 33
- 2.12 `dplyr`, `ggplot2`, and the `tidyverse` . . . 35
- 2.13 Problems . . . 38

3 Probability, random variables, distributions **41**
- 3.1 Experiments . . . 41
- 3.2 An intuitive introduction to the bootstrap . . . 47
- 3.3 Probability . . . 53
- 3.4 Probability calculus . . . 54
- 3.5 Sampling in R . . . 65
- 3.6 Random variables . . . 70
- 3.7 Probability mass function . . . 72
- 3.8 Probability density function . . . 78
- 3.9 Cumulative distribution function . . . 87
- 3.10 Quantiles . . . 88
- 3.11 Problems . . . 93
- 3.12 Supplementary R training . . . 98

4 Mean and variance **105**
4.1 Mean or expected value . . . 105
4.2 Sample mean and bias . . . 117
4.3 Variance, standard deviation, coefficient of variation . . . 125
4.4 Variance interpretation: Chebyshev's inequality . . . 129
4.5 Supplementary R training . . . 132
4.6 Problems . . . 138

5 Random vectors, independence, covariance, and sample mean **141**
5.1 Random vectors . . . 141
5.2 Independent events and variables . . . 148
5.3 iid random variables . . . 150
5.4 Covariance and correlation . . . 155
5.5 Variance of sums of variables . . . 162
5.6 Sample variance . . . 165
5.7 Mixture of distributions . . . 169
5.8 Problems . . . 172

6 Conditional distribution, Bayes rule, ROC **179**
6.1 Conditional probabilities . . . 179
6.2 Bayes rule . . . 192
6.3 ROC and AUC . . . 203
6.4 Problems . . . 229

7 Likelihood **235**
7.1 Likelihood definition and interpretation . . . 235
7.2 Maximum likelihood . . . 245
7.3 Interpreting likelihood ratios . . . 249
7.4 Likelihood for multiple parameters . . . 252
7.5 Profile likelihood . . . 262
7.6 Problems . . . 264

8 Data visualization **269**
8.1 Histograms . . . 270
8.2 Kernel density estimates (KDEs) . . . 273
8.3 Scatterplots . . . 277
8.4 Dotplots . . . 280
8.5 Boxplots . . . 282
8.6 Bar plots and stacked bar plots . . . 289
8.7 QQ-plots . . . 295
8.8 Heat maps . . . 301
8.9 Problems . . . 310

9 Approximation results and confidence intervals **311**
9.1 Limits . . . 311
9.2 Law of Large Numbers (LLN) . . . 315

9.3 Central Limit Theorem (CLT) . . . 319
9.4 Confidence intervals . . . 325
9.5 Problems . . . 332

10 The χ^2 and t distributions 337
10.1 The χ^2 distribution . . . 338
10.2 Confidence intervals for the variance of a Normal . . . 342
10.3 Student's t distribution . . . 348
10.4 Confidence intervals for Normal means . . . 350
10.5 Problems . . . 357

11 t and F tests 361
11.1 Independent group t confidence intervals . . . 361
11.2 t intervals for unequal variances . . . 373
11.3 t-tests and confidence intervals in `R` . . . 374
11.4 The F distribution . . . 380
11.5 Confidence intervals for variance ratios of Normal distributions . . . 383
11.6 Problems . . . 386

12 Data resampling techniques 391
12.1 Jackknife and cross validation . . . 391
12.2 Bootstrap . . . 402
12.3 Problems . . . 408

13 Taking logs of data 413
13.1 Brief review . . . 413
13.2 Taking logs of data . . . 414
13.3 Interpreting logged data . . . 415
13.4 Interpretation of inferences for logged data . . . 417
13.5 Problems . . . 420

14 Interval estimation for binomial probabilities 421
14.1 Confidence intervals for a binomial proportion . . . 421
14.2 The Wald and Agresti-Coull intervals . . . 422
14.3 Bayesian intervals . . . 430
14.4 The exact, Clopper-Pearson interval . . . 439
14.5 Confidence intervals in `R` . . . 440
14.6 Problems . . . 441

15 Building a figure in `ggplot2` 445
15.1 The `qplot` function . . . 446
15.2 The `ggplot` function . . . 452
15.3 Strategies for improving plots . . . 461
15.4 Saving figures: devices . . . 477
15.5 Interactive graphics with one function . . . 478
15.6 Conclusions . . . 479
15.7 Problems . . . 480

16 Hypothesis testing **487**
16.1 Introduction . . . 487
16.2 General hypothesis tests . . . 494
16.3 Connection with confidence intervals . . . 495
16.4 Data example . . . 496
16.5 P-values . . . 499
16.6 Discussion . . . 501
16.7 Problems . . . 503

17 R Programming in the Tidyverse **507**
17.1 Data objects in the `tidyverse`: `tibbles` . . . 507
17.2 `dplyr`: pliers for manipulating data . . . 509
17.3 Grouping data . . . 515
17.4 Merging datasets . . . 516
17.5 Reshaping datasets . . . 523
17.6 Recoding variables . . . 526
17.7 Cleaning strings: the `stringr` package . . . 527
17.8 Problems . . . 541

18 Power **543**
18.1 Introduction . . . 543
18.2 Power calculation for Normal tests . . . 547
18.3 Power for the t test . . . 550
18.4 Discussion . . . 552
18.5 Problems . . . 553

19 Sample size calculations **557**
19.1 Introduction . . . 557
19.2 Sample size calculation for continuous data . . . 558
19.3 Sample size calculation for binary data . . . 568
19.4 Sample size calculations using exact tests . . . 571
19.5 Sample size calculation with preliminary data . . . 574
19.6 Problems . . . 582

References **587**

Chapter 1

Introduction

We provide a modern look at introductory biostatistical concepts and associated computational tools, reflecting the latest developments in computation and visualization using the R language environment (R Core Team 2016). The idea is to offer a complete, online, live book that evolves with the newest developments and is continuously enriched by additional concepts, better examples, and updated R tools. A version of the book will be offered as a hard copy, but, at the core, this is an online book that is reproducible and continuously updated. We provide a one-stop platform that combines theoretical, methodological, and sampling concepts, while simultaneously teaching R and data analysis in realistic scenarios. There are many books that accomplish one or more of these objectives, but we want to cover all of them simultaneously. It is the way we would have liked to learn biostatistics and data science.

Biostatistics is easy to teach poorly. Too often, books focus on methodology with no emphasis on programming and practical implementation. In contrast, books focused on R programming and visualization rarely discuss foundational topics that provide the infrastructure needed by data analysts to make decisions, evaluate analytic tools, and get ready for new and unforeseen challenges. Thus, we are bridging this divide that had no reason to exist in the first place. The book is unapologetic about its focus on biostatistics, that is statistics with biological, public health, and medical applications, though we think that it could be used successfully for large statistical and data science courses.

The book introduces the biostatistical methods necessary for a master's level biostatistician or data analyst. However, it should be accessible to undergraduate students who have a background in calculus and linear algebra and are passionate about data analysis. It covers a wide range of topics combined with R programming and uses a reproducible format that interweaves methodological concepts, data, software, and results. Modern teaching of biostatistical concepts is much easier using the associated R tools. Indeed, without R it would be

difficult to truly understand crucial concepts such as Monte Carlo simulations, bootstrap, permutation testing, coverage probability, central limit theorem, and probability density functions. The basics of `R` programming and syntax can be learned by reading and understanding this book, but additional effort will be necessary to master `R` programming.

We have used data from multiple public sources, but we are especially thankful for the Sleep Heart Health Study (SHHS) (Dean et al. 2016; Quan et al. 1997; Redline et al. 1998) and the Kirby21 (Landman et al. 2011) datasets, which are used extensively throughout the book. In particular, we would like to thank the leaders and researchers of SHHS for conducting this "multi-center cohort study implemented by the National Heart Lung & Blood Institute NHLBI) to determine the cardiovascular and other consequences of sleep-disordered breathing." We would also like to thank the National Sleep Research Resource (NSSR) and the many researchers who have worked on organizing and publishing exceptional, publicly available resources for sleep research. The data used in this book are publicly available and are used under a data user agreement specific for this book. And we thank Naresh Punjabi, friend and mentor, who introduced us to the wonderful world of sleep research, and Bennett Landman, who collected, described, and made publicly available the Kirby21 dataset. According to the Neuroimaging Informatics Tools and Resources Clearinghouse (NITRC), the Kirby21 dataset contains "scan-rescan imaging sessions on 21 healthy volunteers (no history of neurological disease). Imaging modalities include MPRAGE, FLAIR, DTI, resting state fMRI, B0 and B1 field maps, ASL, VASO, quantitative T1 mapping, quantitative T2 mapping, and magnetization transfer imaging. All data have been converted to NIFTI format." For the purposes of this book we use a small subsample of these data. We would like to thank the many participants in these studies who donated their time and data for the advancement of knowledge and science.

The book is the result of a long term collaboration between the three authors and reflects their combined research and teaching philosophies.

Ciprian Crainiceanu, PhD, received his doctorate in statistics from Cornell University in 2003 and is a professor of biostatistics at Johns Hopkins University. He has taught the master's level Methods in Biostatistics course, using and expanding on materials borrowed from Dr. Caffo, who, in turn, distilled materials developed over many years by other Johns Hopkins University Biostatistics faculty. Dr. Crainiceanu is a generalist, who works in many different scientific areas. He has specialized in wearable and implantable technology (WIT) with application to health studies and neuroimaging, especially in structural magnetic resonance imaging (MRI) and computed tomography (CT) with application to clinical studies. Drs. Crainiceanu and Caffo are the co-founders and co-directors of the Statistical Methods and Applications for Research in Technology (SMART) research group.

Brian Caffo, PhD, received his doctorate in statistics from the University of Florida in 2001 before joining the faculty at the Johns Hopkins Department

of Biostatistics, where he became a full professor in 2013. He has pursued research in statistical computing, generalized linear mixed models, neuroimaging, functional magnetic resonance imaging, image processing, and the analysis of big data. He created and led a team that won the ADHD-200 prediction competition and placed twelfth in the large Heritage Health prediction competition. He was the recipient of the Presidential Early Career Award for Scientists and Engineers (PECASE), the highest award given by the US government for early career researchers in Science, Technology, Engineering, and Mathematics (STEM) fields. He also co-created and co-directs the Data Science Specialization, a popular massive open online course (MOOC) degree on data analysis and computing, with over three million enrollments.

John Muschelli, PhD, received his doctorate in biostatistics from the Johns Hopkins Bloomberg School of Public Health in 2016 before joining the faculty there in the same year. He received his master's degree under Dr. Caffo and his PhD under Dr. Crainiceanu. He has pursued research in statistical computing, neuroimaging, computed tomography in patients with hemorrhagic stroke, image processing, and interactive graphics. He is a founder of the Neuroconductor project (https://neuroconductor.org/), which aims at centralizing the biomedical imaging tools for `R`. He has created a number of `R` packages on topics ranging from biomedical imaging to interfacing with APIs for bibliometric analysis. He has also created short courses in neuroimaging in `R` and is passionate about introducing users to `R`.

1.1 Biostatistics

We begin with the difficult task of defining the subject that we are attempting to teach. We start with the definition that our department (Biostatistics at Johns Hopkins) agreed upon in its 2007 self-study. **"Biostatistics is the theory and methodology for the acquisition and use of quantitative evidence in biomedical research. Biostatisticians develop innovative designs and analytic methods targeted at increasing available information, improving the relevance and validity of statistical analyses, making best use of available information and communicating relevant uncertainties."**

This definition will serve our needs well enough, though additional insights are necessary, since much has happened in the world of data since 2007. In short order, a data revolution is happening and biostatistics is a key part of it. If one wants to take part in this revolution in biological sciences, public health and medicine, some degree of mastery of biostatistics is key. However, biostatistics is a challenging subject to master if one is interested in understanding both the underlying concepts and their appropriate implementation. In this book we try to achieve these complementary goals.

As the pace of the data revolution accelerates, it is worth asking whether devoting the amount of space herein to the fundamentals of biostatistics is worthwhile. It is our opinion that while learning *how to* push the biostatistical buttons to conduct a routine analysis is perhaps straightforward, understanding *why to* apply a specific technique or *what and when to* change is often difficult and requires more insight. Moreover, jobs that require only a cursory understanding of statistics and routine statistical analyses are the most likely to disappear and be automated out of existence. Critical thinking, creativity, and understanding the specifics of scientific problems demand more effort, a higher level of thinking, and a broader understanding of science. Thus, we believe that jobs requiring these skills will not be automated anytime soon. The principles of data analysis are crucial to solving existent and future analytic problems. Indeed, preparing the student for what may happen in data analysis 10 years from now is daunting, but learning the basic principles is a step in the right direction, as principles are changing slower than emerging data challenges. Biostatistics can be viewed as a powerful and practical philosophy of science, in which the scientific hypothesis, the experiment, the data, the model, and the associated inference form the basis of scientific progress. In the end, biostatistics is hard because it is focused on solving difficult problems using simple approaches and *simplicity is difficult to achieve in complex scenarios.*

The class that inspired this book is often taken by students with very different backgrounds and skill levels. Calculus, linear algebra, and a moderate level of mathematical literacy are prerequisites for it. However, even if a student has the prerequisites does not mean they need to take it or that they will perform well. Indeed, by far the hardest problem for students taking this class has been to adapt to the different way of thinking required by working with data. Students with strong engineering or mathematical backgrounds are often puzzled by the fact that mathematical mastery is not enough to become an expert in the easier, softer, "dirtier" concepts of biostatistics. Students with strong scientific backgrounds can also become frustrated by the slower pace of introduction of useful concepts, mathematical intricacies, and modeling abstractions. Biostatistics is not hard in the usual sense that it requires either deep mathematical concepts or committing large books to memory. In fact, the required mathematical level is reasonable and students are seldom required to memorize concepts. Thus, the associated biostatistics exams at Johns Hopkins are open everything, including book, laptop, and internet. The difficulty of biostatistics rests in the different philosophy, the different way of thinking, and the different set of skills required to process human communication of scientific problems and their translation into well-defined problems that can be solved with data. Once a scientific problem is defined, the biostatistics philosophy is to try to solve it using the simplest possible approaches that are not too simplistic. Thus, parsimony and inductive reasoning are fundamental concepts in biostatistics.

In the logical discourse of knowledge biostatistics has a very well-defined place. Indeed, biostatistics starts with the current accepted state of knowledge (the collection of null hypotheses) and uses data to inductively refute or reinforce

parts of the knowledge or generate new potential knowledge (new null hypotheses). In the biostatistical philosophy there is no truth, just the state of current knowledge. In contrast, mathematics starts with a set of axioms (the truth) and develops knowledge deductively based on logic. These differences lead to vastly different philosophies, especially when one is interested in solving specific scientific problems. With the large increase in data availability and complexity, the importance of biostatistical philosophy and practice will continue to increase and deeply influence our society and knowledge.

A key goal of this text is to teach the different way of thinking present in biostatistical applications. This methodology has many similarities to that of other empirical sciences, such as social sciences and economics. However, a distinguishing feature of biostatistics is that the elements of biological sciences, public health, and medical science are key components of the problem. Given this wide range of scientific applications, of particular importance in our study will be translating scientific ideas into formal inductive inferential problems cast in the language of statistics. We emphasize the value of simple methods that do not cross the line into being simplistic. This process of balancing performance, parsimony, and interpretability, has elements of art and science.

Probably the most important role of a biostatistician is to bring scientists together, help bridge the scientific language barriers, and create productive and truly interdisciplinary research teams. Inter- or trans-disciplinary research is a hot topic that is extremely easy to talk about and hard to implement. Many universities have or are launching interdisciplinary programs that often do not work or under-perform without apparent explanation. The actual reason is that interdisciplinary research requires a lot of effort and risk taking that remains under-rewarded in Academia. Moreover, it requires the honest intent of participants to be part of a team of equals, the will to listen and learn a different scientific language and approach, and a continuous application of collaborative principles dedicated to defining and solving scientific problems.

Biostatistics is used extensively in studies that impact the lives of billions of people. Below we present a few examples indicating the importance of careful biostatistical thinking.

1.1.1 Example 1: Cancer screening

The Canadian National Breast Cancer Screening studies (Miller et al. 2002, 2000, 2014) were large landmark clinical trials studying the impact of mammography. The first publication based on this large randomized screening trial found no benefit of early tumor detection via digital mammography for women aged 40-49, contradicting standard radiological practice at the time. Many discussion articles have criticized the study on statistical grounds. For example, Burhenne and Burhenne (1993) focused on statistical power and the ability to generalize from the sample. One can see from this study the important role that

statistics and empiricism plays; standard health practice for a large proportion of women depends on the findings.

1.1.2 Example 2: Harvard first-borns

From 75 to 80 percent of students at Harvard are first-borns. Do first-born children work harder academically, and so end up over-represented at top universities? Yes, claims noted philosopher Michael Sandel (Sandel 2010). But Millner and Calel (2012) find a simple fault in the statistical reasoning and give a more plausible explanation: wealthy and well-educated parents tend to have fewer children.

1.1.3 Example 3: Oscar winners

Redelmeier and Singh (2001) identified all actors and actresses ever nominated for an Academy Award in a leading or a supporting role up to the time of the study ($n = 762$). Among them were 235 Oscar winners. For each nominee, another cast member of the same sex who was in the same film and was born in the same era was identified ($n = 887$) and these were used as controls. The overall difference in life expectancy was 3.9 years (79.7 years for award winners vs. 75.8 years for controls; p-value = .003). To avoid the possible selection bias, an analysis using time-dependent covariates (winners counted as controls until they won the Oscar) did not find significant differences (Sylvestre, Huszti, and Hanley 2006). This is called *selection or immortal bias*.

1.1.4 Example 4: Hormone Replacement Therapy (HRT)

A large clinical trial (the Women's Health Initiative) published results in 2002 (Rossouw et al. 2002) that contradicted prior evidence on the efficacy of hormone replacement therapy (HRT) for post-menopausal women and suggested a negative impact of HRT for several key health outcomes. *Based on a statistically based protocol, the study was stopped early due to an excess number of negative events.*

1.1.5 Example 5: ExtraCorporeal Membrane Oxygenation (ECMO) treatment

Bartlett et al. (1985) published results of a clinical trial conducted at a major neonatal intensive care center. The trial compared standard treatment with a promising new extracorporeal membrane oxygenation treatment (ECMO) for newborn infants with severe respiratory failure. Ethical considerations led to a statistical randomization scheme whereby only one infant received the control

therapy, thereby opening the study to sample-size based criticisms; see Rosenberger and Lachin (2015).

1.1.6 Summary

Biostatistics plays a central role in biological sciences, public health, and medical applications and provides a platform for correct design, analysis, and interpretation of data. Biostatistics requires: (1) a tight coupling of the biostatistical methods with ethical and scientific goals of research; (2) an emphasis on the scientific interpretation of statistical evidence to impact policy; and (3) a detailed acknowledgment of assumptions and a comprehensive evaluation of the robustness of conclusions to these assumptions.

1.2 Mathematical prerequisites

Calculus, linear algebra, and a moderate level of mathematical literacy are needed to understand this book. In short, you should be able to solve the following integral

$$\int_0^\infty x^2 \exp(-ax)dx$$

in 3 minutes and be able to explain how you did it. You should not have any questions about "what's the wiggly sign?" and "what's the funny 8?" and it should be 100% clear why the integral is finite only when $a > 0$. Also, you should be able to explain why

$$\log(3/10^a) = \log(3) - a\log(10)$$

1.3 R

For this book we will use the statistical software `R` (@R, https://cran.r-project.org/) because it is free, flexible, and up-to-date due to the many packages contributed by the community. `R` contains a variety of packages including parametric and nonparametric tests, survival analysis, regression, and machine learning (random forests, support vector machines, neuronal networks, clustering). Understanding biostatistics at a deeper level is hard because it requires understanding a combination of difficult new concepts. For example, it is hard to understand what a random variable, a probability density function, or weak convergence is without hands-on experience with a dedicated analytic software, such as `R`. Here we will emphasize the close connection between theoretical concepts and practical questions, using dedicated `R` software. In a world moving quickly towards more computerization and data intensive decision making it is

unimaginable to teach a biostatistics course without introducing the basics of scientific computing.

Moreover, though statistical concepts can be explained using simulations and other `R` functions, `R` is a powerful analytic language. We wish to provide you the tools to understand biostatistical concepts as well as the ability to analyze data using the methods we discuss. Of course, `R` is not the only possible platform for this purpose. Indeed, SAS and STATA are also widely used by statisticians, while `Python` is another powerful platform used extensively in data analysis. Programs such as `SPSS`, `Minitab` and `Excel`, which can be great for some analyses, are insufficient for the needs of this book.

There are many ways to learn R. There are many introductory tutorials online, such as Code School and DataCamp. This general R reference card is useful to see many commonly-used `R` functions. The next chapter introduces some basic concepts in `R` that will be used throughout the book, though every chapter will contain commented `R` code interweaved with biostatistical concepts. Several additional dedicated `R` chapters are available throughout the book and introduce increasingly more advanced topics in `R` to complement the introduction of biostatistical concepts.

1.4 Data

All data located for this course can be downloaded on the leanpub book page or can be downloaded directly from GitHub.

Chapter 2

Introduction to R

This chapter covers the following topics

- R and RStudio
- Reading R code
- R syntax
- Data Classes
- Libraries and packages

2.1 R and RStudio

In this book R (R Core Team 2016) refers to "a language and environment for statistical computing and graphics" (http://www.r-project.org/). R was built on the S language, which was developed by Bell laboratories, and S+, which was an improvement of S. Knowing the history or these languages is unimportant, but the fact that R was built on other languages can explain some of its quirks.

When you download R from the R-project website, this contains the R language and a graphical user interface (GUI). This GUI is a set of windows that allow you to interact with the R console, plotting windows, and script editor, which is essentially a text editor. If you used R from a command line or a terminal, this is still R, but only the R console – the thing you interact with and where you pass R commands.

RStudio is a company that makes software for R, namely the RStudio IDE (integrated development environment). This IDE is another GUI that allows users to interact with R with some added benefits. We will refer to this IDE/set of windows as RStudio; we will specify if we are referring to the company by stating RStudio Inc. The RStudio software is not exactly written by the R "team,"

but has become the standard IDE used when working in R and we recommend it.

In the following sections, we will use double quotes (") when introducing any jargon that may be new to the user.

2.2 Reading R code

We will discuss how to read the R code in these chapters. Code is color labeled to help readers determine what operations are going on in the code. Let us start with an easy addition operation:

```
1 + 5
```

```
[1] 6
```

We see the code is surrounded by a slightly dark box, which shows the user that it is either code or output. The output of the command is printed directly after the code, if there is any. When the code is run we say that the statement `1 + 5` was "evaluated" or "executed." Comments in code are marked by the pound/hashtag `#`. Anything after the pound is treated as a comment and is not run/evaluated by R. If this symbol is at the beginning of a line, then that whole line is treated as a comment:

```
# 1 + 5
1 + 5 # the code to the left is run
```

```
[1] 6
```

```
# I'm just a comment and can use any character I want!
```

Comments are in italics and have a different text color. There is no operation/symbol for multi-line comments in R code; a `#` must be placed in the front of each line to comment them out. In RStudio, there are shortcuts for this if you look under the "Code" menu for "Comment/Uncomment Lines".

In addition to this, when things are placed in quotes, they are treated differently in R so the code shows them in a different color from numbers.

```
"hey"
```

```
[1] "hey"
```

We will cover the difference between character strings and numbers later.

2.3 R syntax and jargon

Below we will briefly cover some topics of R code syntax. Overall, we will discuss objects, data classes, and subsetting.

2.4 Objects

In R, almost everything is an "object" or "variable." This setup is different than other statistical software, such as STATA or SAS. In STATA or SAS, most things are relevant only with respect to a data set. The fact that R can hold things that are not relevant to a data set (e.g., the mean age of a group or a linear model) makes it confusing at first to those coming from those other languages. After learning how to manipulate these objects, it will become clear that this is useful to create tables and figures that require these models or numbers.

2.5 Assignment

To make an object you "assign" it to an object name. Assignment basically means you set something equal to something else. There are two ways to do this; we call them assignment "operators", which is to say they operate on things. One way is to use a single equals sign:

```
x = 5
x
```

```
[1] 5
```

After running this command, the object `x` has the value of `5`. In most cases, when you simply execute a command that is just an object, the contents of that object are printed. We could explicitly do this by running `print(x)`. We would say that "`x` was assigned to the value of `5`" or "`x` was set to `5`". The other way to perform assigment is using the "assignment operator," which is `<-`. This is the less than symbol (`<`) immediately followed by a hyphen (`-`) without spaces.

```
x <- 4
x
```

```
[1] 4
```

We see that `x` is now reset to the value of `4`. Some prefer the equals sign as this is more common in other programming languages and requires one character to assign objects. Others prefer the assignment operator because that was the main assignment operator for R for many years and shows the "direction" of the assignment. When using `=`, the object getting assigned is always on the left-hand side. In R, we will use these assignment operators interchangeably. There

is also a "forward assignment" operator that exists, but this is **rarely** used and we do not recommend using it:

```
4 -> new_var
new_var
```

```
[1] 4
```

We can assign `y` to a value and then use that variable in operations:

```
y = 6
x + y
```

```
[1] 10
```

When creating objects, you can use alpha-numeric characters (upper and lower case), but a number cannot be the first character. You can also use the underscore (`_`) and period (`.`). Many prefer using underscores over periods, as the period has a different meaning in other languages. For example, these are acceptable object names:

```
my_fun_variable
myFunVariable
my.fun.variable
my.fun_variable
```

but some are not a preferred syntax for all, especially those that include periods and underscores. You cannot use many special symbols such as the dollar sign (`$`), quotes (`'` or `"`), pound (`#`), hyphen (`-`), or **spaces** in object names. For example, these are not acceptable variable names:

```
it's_my_variable
patient age
money_in_$_us
patient-name$
```

2.5.1 Operations

You can perform the standard operations on vectors, such as addition/subtraction (`+`, `-`), division/multiplication (`/`, `*`), and exponents (`^` or `**`). You can also perform most mathematical functions such as absolute value (`abs`), square root (`sqrt`), natural logarithm/exponential (`log`/`exp`), and so on. These operations are performed entrywise. Type `?base::Math` (this notation will be explained in section 2.11.2) to see a list of the standard operations.

2.6 Data types

In R, there are different data types, but we will focus here on the three most fundamental: numeric (numbers), character (strings or "words"), and logicals (TRUE or FALSE). Above, when we assigned x to 4, we would say x is a numeric vector. Had we assigned "hey" to an object, let's say z, so that z = "hey", we would say z is a character vector. We could create a logical vector w with the code w = TRUE. Nota Bene: in logical vectors TRUE and FALSE are all capital letters and are **not** in quotes. All examples in this section are "vectors of length 1."

We will also discuss one additional data type: factors. Factors are categorical data types. The categories of a factor are called the "levels" of that factor. The levels of factors usually contain human-readable levels (e.g., "Male" and "Female"). Factors become increasingly useful when we fit models, as one can define the baseline category by the levels of the factor, which affects the interpretation of coefficients. Some refer to vectors of length 1 as a "scalar". In other languages, there is a distinction between scalars and vectors, but not in R. So you can think of a vector as a 1-dimensional object that can have anywhere from 0 to a **large** number of elements.

2.6.1 Advanced topic: integer vs. double

In R, numbers are generally represented in 2 ways: as integers and doubles. Integers are simply round numbers, with no decimals. All other numbers are doubles. Doubles can be round numbers, but they are stored differently than a number that is explicitly an integer. Typically, data use doubles, while indexing, referencing, and subsetting use integers. Overall, though, R will know how to handle numbers without much work on the part of the user and give warnings if the user tries to do something unexpected with numbers.

2.7 Data containers

In every example shown above, the value or object is simply one element. R can contain much more complex pieces of data in different containers. We will build from a single number, to multiple numbers, to an object with rows and columns with all numbers, then an object with multiple rows and some columns with numbers and others that have non-numbers.

2.7.1 More than one number: vectors

In R, anything having one dimension is generally referred to as a "vector." A vector has a specified length. Technically, a vector may have length 0, which

means nothing is in it. But usually vectors have one or multiple objects in them. Those objects all must have the same **data type** (e.g., all numbers or all characters or all logicals).

2.7.2 Creating vectors

There are multiple ways of creating vectors; we will first discuss the most general way: using the `c` command/function. Whenever we reference functions in the text, we will either try to be clear that it is a function (as opposed to an object/variable) or write the command with parentheses (e.g., `c()`). The `c()` function allows you to **c**ombine values into a vector. Here we will assign `x` to a new value:

```
x = c(1, 4)
print(x)
```

```
[1] 1 4
```

We see that `x` is a numeric vector of length 2 and has the values 1 and 4. In functions, the syntax is function name, followed by an open parenthesis `(`, then the arguments, then a close parenthesis `)`. Anything within those parentheses are "passed" into that function, which we refer to as "arguments" of that function. In the code above, `R` will evaluate/execute/run the `c()` function, then assign that output to the value `x`. There are functions that can access information about the vector. One commonly used function is `length()`, which reports how long a vector is:

```
length(x)
```

```
[1] 2
```

We can also assign that output to a new variable! Here we will assign the length of `x` to the variable `length_of_x`:

```
length_of_x = length(x)
```

Note again, as we assigned the output to a variable, it was not printed.

2.7.3 Sequences

In many cases, you want to create a sequence of numbers. Some examples would be patient identifiers or an index for a patient visit. We will show you how to create flexible sequences of numbers with the `seq()` function below. Many times, however, we want just numbers incremented by 1. In `R`, there is a shortcut for this, called the colon operator (`:`). You simply write `start:end` for a sequence of numbers from `start` to `end`, like so:

```
1:5
```

```
[1] 1 2 3 4 5
```

```
2:4
```

```
[1] 2 3 4
```

Note that if you would like a sequence of negative numbers, it needs to be treated a bit differently:

```
-5:1
```

```
[1] -5 -4 -3 -2 -1  0  1
```

This is interpreted as a sequence from `-5` to `1`, not `-5` to `-1`. If you want the latter sequence, you would write:

```
-(5:1)
```

```
[1] -5 -4 -3 -2 -1
```

Here we note that parentheses are not only used in functions, they are used to encapsulate or group a statement or "expression". The `seq()` function allows you to create more flexible sequences, which take arguments `from`, `to`, and `by`. By default, `by = 1`, so it creates sequences that increment by `1` unit:

```
seq(1, 5)
```

```
[1] 1 2 3 4 5
```

```
seq(1, 7, by = 0.4)
```

```
 [1] 1.0 1.4 1.8 2.2 2.6 3.0 3.4 3.8 4.2 4.6 5.0 5.4 5.8 6.2 6.6 7.0
```

Note, the code:

```
seq(1, 7, by = 0.4)
```

```
 [1] 1.0 1.4 1.8 2.2 2.6 3.0 3.4 3.8 4.2 4.6 5.0 5.4 5.8 6.2 6.6 7.0
```

When we put (`by = 0.4`) it was the first time we directly assigned an argument in a function to a value by name. By default, a function will assume the order of the input into the function in the order of the arguments of the function. This behavior is common and works well when you are writing code with well-established functions that will almost never change the arguments or the order of those arguments. Regardless, you should specify arguments **by name** in a function when you can, as this is a safer way of writing code. Also, you do not need to know the order of the arguments if you know what the argument names are. For example, we can perform the same operation as above by doing:

```
seq(by = 0.4, to = 7, from = 1)
```

```
 [1] 1.0 1.4 1.8 2.2 2.6 3.0 3.4 3.8 4.2 4.6 5.0 5.4 5.8 6.2 6.6 7.0
```

even though `to` is the first argument defined in the function. Moreover, the RStudio IDE allows you to write the function name and type the left parenthesis (`(`). Then hit the Tab key to show you the arguments. We will try to assign arguments by name throughout the book when possible.

In many cases where you can input a highly flexible number of things into an argument, it may be specified in the help file with "3 dots," or an ellipsis (`...`). This is the case when pasting together strings (`paste()`), which we will cover below, or when combining elements in the `c()` function. Also, if the function takes one argument, it is commonly named `x` and usually you do not specify `x =`, you simply pass the one argument.

2.7.4 Operations on numeric vectors

You can do all the same operations on numeric vectors as you did when there was only one element.

```
x_length_2 = c(1, 2)
x_length_2 * 3
```

```
[1] 3 6
```

```
sqrt(x_length_2)
```

```
[1] 1.000000 1.414214
```

One major difference is when you are performing operations with two vectors:

```
x_length_3 = c(10, 20, 30)
x_length_2 + x_length_3
```

```
Warning in x_length_2 + x_length_3: longer object length is not a multiple
of shorter object length
```

```
[1] 11 22 31
```

First and foremost, note that this command executed. This may be surprising to some. You do get a warning here, but this is an important case. We see that the last number is `31`. When trying to add the two vectors, `R` saw that `x_length_2` had one less element than `x_length_3`. It then **padded** the `x_length_2` object to the length of `x_length_3`, by repeating `x_length_2` again. This is called a "wraparound effect", and can be useful, but very dangerous. Therefore, `R` made `x_length_2 = c(1, 2, 1)` to make the lengths match and added the `1` to the `30`.

Note, the warning was not that they were not the same length, but rather the lengths were not **multiples** of each other:

```
x_length_4 = c(10, 20, 30, 40)
x_length_2 + x_length_4
```

```
[1] 11 22 31 42
```

Here we have a wraparound effect, but no warning! Many times this will not be what you want to happen. This effect is important to learn when dealing with vectors. Though this effect is still possible when working with data sets, it is less common when using columns of the data set to add or manipulate together because they are constrained to have the same number of elements, which is the number of rows in the data set.

2.7.5 Some standard operations on vectors

We can determine the number of elements in a vector using the `length` function:

```
length(x_length_3)
```

```
[1] 3
```

If we would like to know the unique elements of a vector, we use the `unique` function:

```
x = c(2, 1, 1, 4)
unique(x)
```

```
[1] 2 1 4
```

Coupled with `length`, this can provide the number of unique elements: `length(unique(x))`. This function can be useful for counting the number of unique patients in a data set, for example. Note, `unique` does not sort the unique elements. To sort the elements, you would use `sort`:

```
sort(x)
```

```
[1] 1 1 2 4
```

```
sort(unique(x))
```

```
[1] 1 2 4
```

Although not as commonly used in all cases, we will discuss the `sample` function as it will be highly relevant in later chapters on the bootstrap. Without any other arguments, `sample` will perform a permutation on a vector `x`, where all the elements of `x` are simply reshuffled randomly:

```
sample(x)
```

```
[1] 1 2 1 4
```

The help file for sample (accessed by running the `?sample` command) provides two additional useful arguments: `size` and `replace`. The `size` argument indicates how many samples are to be drawn. By default, it will resample as many

as the length of `x`. For example, if `x` is of length 4, then the result from `sample` will be of length 4. You can sub-sample `x` with a size smaller than its length as:

```
sample(x, size = 2)
```

```
[1] 2 1
```

If you try a size greater than the length of `x`, by default, you will get an error because of the `replace` argument. By default, `replace` is `FALSE`, so sampling is done **without replacement**. This error occurs because you cannot draw more samples out of the data than exist in the data without replacement. If the `replace` option is `TRUE`, then sampling will be done **with replacement**, indicating that the same element can be repeated multiple times and others may not be sampled at all.

```
# sample(x, size = 10) will error
sample(x, size = 10, replace = TRUE)
```

```
 [1] 1 1 2 4 2 1 4 4 4 1
```

This will allow us to draw samples of any size with replacement from a vector. Many times, we are sampling the same length as the vector with replacement, as in the bootstrap procedure discussed later, so the `size` argument can be omitted:

```
sample(x, replace = TRUE)
```

```
[1] 1 1 1 2
```

2.7.6 Creating logical and character vectors

We can also create character and logical vectors in the same way:

```
trues_and_falses = c(TRUE, TRUE, FALSE)
print(trues_and_falses)
```

```
[1]  TRUE  TRUE FALSE
```

```
our_names = c("Ciprian", "Brian", "John", "You")
print(our_names)
```

```
[1] "Ciprian" "Brian"   "John"    "You"
```

Note that character vectors, when printed, have quotes around the "elements".

2.7.7 Data type coercion

If we combine two values of different data types into a vector, `R` has to change one of the data types. This, again, is because vectors can only have one data

type in them. This process is called "coercion", as R "coerces" one of the types to another type so that it can be combined with the other values.

Logicals can be coerced to numerics (0 and 1 values) and coerced to characters ("TRUE" and "FALSE", note the quotes). Numerics can be coerced into characters (5.6 changes to "5.6"). We can see this process as follows:

```
c(5.2, TRUE, FALSE)
```

```
[1] 5.2 1.0 0.0
```

```
c("hey", TRUE)
```

```
[1] "hey"  "TRUE"
```

```
c(5.2, "hey")
```

```
[1] "5.2" "hey"
```

```
c(5.2, TRUE, "hey")
```

```
[1] "5.2"  "TRUE" "hey"
```

This is important when doing data manipulation and data input. If you try to read in data where one of the rows has non-numeric data (such as the letters n/a), then R may treat that whole column as characters, which can cause problems in analysis. You can determine the data type of an object with the typeof() function:

```
typeof(c(5.2, TRUE, FALSE))
```

```
[1] "double"
```

This function is not used as often as some other functions we will discuss on how to coerce vectors into different data types. Above, the data are still a vector (the data **class**) but the elements of that vector are now changed to numeric (the data **type**).

2.7.8 More than one dimension: matrices

Vectors are very useful, but may not be sufficient for all scenarios. A matrix is like a vector, but it has two dimensions, rows and columns. Like vectors, matrices can contain only **one data type**. You can construct a matrix with the matrix() command:

```
matrix(1)
```

```
     [,1]
[1,]    1
```

We see that a matrix is printed out differently than a vector. There is a label above the columns (`[,1]`) and to the left of the rows (`[,2]`). These reference the column and row notation for matrices. Let us construct another matrix with three columns and two rows, from a vector of values.

```
matrix(1:6, nrow = 2, ncol = 3)
```

```
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

Here we see the matrix organized in rows and columns and we specified the number of rows and columns. If we only specified one, `R` would try to divide the length of the vector by the number of rows/columns specified and assign the other dimension. For example:

```
matrix(1:6, nrow = 2) # no ncol specified, same result
```

```
     [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

By default, **`R` does many things column-wise**, and `matrix()` fills the matrix column-wise. We can fill in the data by rows, using the `byrow = TRUE` option:

```
matrix(1:6, nrow = 2, byrow = TRUE) # filled by row
```

```
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
```

2.7.9 More than one dimension: `data.frames`

A `data.frame` is like a matrix in that it has rows and columns. A `data.frame` can have columns of different data types. For example, the first column can be age, a numeric variable, and the second column can be a three-level categorical variable of handedness with values of left, right, and ambidextrous. A `data.frame` is similar in nature to a spreadsheet of data. Indeed, the most common data class for data sets is a `data.frame`.

You can construct a `data.frame` with the `data.frame()` command. Note, a default on the `data.frame()` function is `stringsAsFactors = TRUE`, which indicates that all strings should be converted to factors. Many times, you want to clean these columns and then convert them to factors directly. So we will set that option to `FALSE`:

```
df = data.frame(age = c(25, 30, 32, 42),
                handed = c("left", "right", "ambidextrous", "left"),
```

```
                stringsAsFactors = FALSE)
df
```

```
  age       handed
1  25         left
2  30        right
3  32 ambidextrous
4  42         left
```

The arguments of `data.frame` do not include `age` or `handed`; these are column names for these variables. Note, although `data.frame`s are like matrices, they do not have the same printing with the `[,1]` and `[1,]` notation.

2.7.10 Dimensions of an object

There are functions that get attributes about a matrix/`data.frame` that are useful for analysis. The `dim` function returns the dimensions (rows and columns, respectively) of a data set:

```
dim(df)
```

```
[1] 4 2
```

Each of these can be accessed for the **n**umber of **row**s or **col**umns using the `nrow` and `ncol` functions, respectively:

```
nrow(df)
```

```
[1] 4
```

```
ncol(df)
```

```
[1] 2
```

If you pass in a simple integer/number to `sample`, e.g., `nrow(df)`, it will sample from `1:nrow(df)`:

```
sample(nrow(df))
```

```
[1] 2 4 1 3
```

This may be useful for sampling/subsetting from the rows of a data set, which will be explained later in section 2.9.3.

2.7.11 Viewing the data

Getting a quick peek at the first few rows of a `data.frame` is useful for getting an idea of the data structure, using the `head` function. This is especially useful when the dimensions are very large:

```
head(df)
```

```
  age       handed
1  25         left
2  30        right
3  32 ambidextrous
4  42         left
```

In RStudio, the `View()` function will bring up a spreadsheet-like viewer of your data sets. This viewer is very useful for exploring your data. Be careful to use `View` interactively and not to set `View` commands inside your `R` scripts, as this may cause errors when running them or compiling R Markdown documents.

2.8 Logical operations

We will discuss some operations in `R` that perform some logical tests that return logical data. There are some operations in `R`: greater/less than (`>`, `<`), greater/less than or equal to (`>=`, `<=`), equal to (`==` "double equals"), and not equal to (`!=`). These are called "relational operators" or "comparison operators," but they are always just referenced by their name (e.g., greater than).

```
x = c(1, 3, 4, 5)
x < 4
```

```
[1]  TRUE  TRUE FALSE FALSE
```

```
x > 3
```

```
[1] FALSE FALSE  TRUE  TRUE
```

```
x >= 3
```

```
[1] FALSE  TRUE  TRUE  TRUE
```

```
x == 3
```

```
[1] FALSE  TRUE FALSE FALSE
```

```
x != 3
```

```
[1]  TRUE FALSE  TRUE  TRUE
```

There are also the standard "logical operators": not/negate (`!`), AND (`&`), and OR (`|`). The OR operator is called the "pipe" and it is a straight vertical line. For example, we can combine some of the previous statements:

```
x = c(1, 3, 4, 5)
x > 3 | x == 1 # greater than 3 or equal to 1
```

```
[1]  TRUE FALSE  TRUE  TRUE
```

```
!(x > 3 | x == 1) # negates the statement - turns TRUE to FALSE
```

```
[1] FALSE  TRUE FALSE FALSE
```

```
x != 3 & x != 5
```

```
[1]  TRUE FALSE  TRUE FALSE
```

```
x == 3 | x == 4
```

```
[1] FALSE  TRUE  TRUE FALSE
```

2.8.1 The %in% Operator

The last two statements select on two specific elements from `x`. Many times, we want to keep or exclude based on a number of values. The `%in%` operator is very useful for that:

```
x %in% c(3,4)
```

```
[1] FALSE  TRUE  TRUE FALSE
```

You pass the object on the left hand side, then apply the `%in%` operator, and then specify the values to match in the object. The result is a logical indicator if those values are in the object. This operation is very different than testing for equality:

```
y = c(4,3)
y == c(3,4)
```

```
[1] FALSE FALSE
```

`R` is trying to test if `y` is equal to the right hand side **element-wise** (at each element, lining them up). Therefore, this will return `TRUE` for the first element if the first elements in both vectors are the same, `FALSE` otherwise, for each element across the vector.

2.9 Subsetting

2.9.1 Subsetting vectors

In vectors, you subset elements of the vector using "square brackets" (`[` and `]`). You can subset using the index of the element you want, where the first element has the index `1`.

```
x = c(1, 3, 4, 5)
x[4] # 4th element = 5
```

```
[1] 5
```

You can also subset using a logical vector of the same length as the vector:

```
x > 2 # returns a logical vector
```

```
[1] FALSE  TRUE  TRUE  TRUE
```

```
x[ x > 2 ] # we can use that to subset
```

```
[1] 3 4 5
```

2.9.2 Subsetting matrices

In matrices, you subset elements of the vector using "square brackets" (`[` and `]`), but the syntax is `[row subset, column subset]`. Again, you can subset using the index of the element (row/column) you want, but can also subset using a logical vector. The logical vectors must be the same length as the dimension to subset (row or column):

```
mat = matrix(1:6, nrow = 2) # no ncol specified, same result
mat[1:2, 1]
```

```
[1] 1 2
```

```
mat[1, 1:2]
```

```
[1] 1 3
```

```
mat[1, c(FALSE, TRUE, FALSE)]
```

```
[1] 3
```

If the column names or the row names of the `matrix` are set, then these can be used to subset the `matrix`. You do not mix-and-match though: if you are subsetting with indices, logicals, or names, you do only one type of subsetting at that time. For example, you would **not** do:

```
mat[, c(FALSE, 3, TRUE)]
```

```
     [,1] [,2]
[1,]    5    1
[2,]    6    2
```

as the subsetting vector would be **coerced** to a numeric and subset the third and first columns (in that order). If either subsetting is missing, then **all** elements of that dimension are presented.

```
mat[1:2, ]
```

```
     [,1] [,2] [,3]
[1,]    1    3    5
```

```
[2,]    2    4    6
```

```
mat[, 2:3]
```

```
     [,1] [,2]
[1,]    3    5
[2,]    4    6
```

```
mat[, c(FALSE, TRUE, FALSE)]
```

```
[1] 3 4
```

By default, if you subset a `matrix` and it returns only one column or one row, the output is a `vector`:

```
class(mat[,1])
```

```
[1] "integer"
```

```
class(mat[1])
```

```
[1] "integer"
```

```
class(mat[1,1:2])
```

```
[1] "integer"
```

But if the subset is two-dimensional, then the result is still a `matrix`:

```
class(mat[,1:2])
```

```
[1] "matrix"
```

```
class(mat[1:2,])
```

```
[1] "matrix"
```

You can ensure that the result is a `matrix`, even if the result is one-dimensional using the `drop = FALSE` argument:

```
mat[,1, drop = FALSE]
```

```
     [,1]
[1,]    1
[2,]    2
```

```
class(mat[,1, drop = FALSE])
```

```
[1] "matrix"
```

2.9.3 Subsetting `data.frames`

The bracket syntax for subsetting a `matrix` can be used for a `data.frame` as well. Because `data.frame` can contain columns of multiple data types, when a `data.frame` is subset with only one row, it returns a `data.frame`, not a vector.

```
df[1,]
```

```
  age handed
1  25   left
```

```
class(df[1,])
```

```
[1] "data.frame"
```

```
df[1, c("age", "handed")]
```

```
  age handed
1  25   left
```

```
class(df[1, c("age", "handed")])
```

```
[1] "data.frame"
```

But as each column of a `data.frame` is of the same data type, when one column is subset in a `data.frame`, then a `vector` is returned:

```
df[,1]
```

```
[1] 25 30 32 42
```

```
class(df[,1])
```

```
[1] "numeric"
```

```
df[1, c("age")] # still only 1 column - age
```

```
[1] 25
```

```
class(df[, c("age", "handed")]) # 2 columns still
```

```
[1] "data.frame"
```

Again, if you set `drop = FALSE`, then a `data.frame` is returned:

```
class(df[, c("age"), drop = FALSE]) # returns a data.frame
```

```
[1] "data.frame"
```

2.9.3.1 Dollar sign subsetting

Almost always, the columns of a `data.frame` are named. When this is the case, you can subset the `data.frame` as above using brackets, but you can also use

the dollar sign `$`:

```
df[, "age"]
```

```
[1] 25 30 32 42
```

```
df$age
```

```
[1] 25 30 32 42
```

Note though that the dollar sign only subsets **one** column at a time, so you cannot do:

```
df$c("age", "handed")
```

Column names should have only letters and numbers and start with a letter. Other symbols, such as "special characters" slashes (`/` or `\`), dollar signs (`$`), pound symbol (`#`), or dashes (`-`) need to be changed to a period. Column names should also not have spaces (but technically can).

2.10 Reassignment

Similar to how we can assign the output of an assignment to a new object:

```
x4 = x[4]
```

we can also change or reassign the values subsetted in the object:

```
x2 = x
x2[4] = 8
x2
```

```
[1] 1 3 4 8
```

The number of elements in the subset on the left-hand side of the assignment should equal the number of elements on the right hand side. For example, we can assign two values in `x2`:

```
x2[c(1,5)] = c(6, 4)
x2
```

```
[1] 6 3 4 8 4
```

2.11 Libraries and packages

When you download `R`, it consists of a set of "base" `R` functions. A collection of functions in `R` is called an `R` package or `R` library. The strength of `R` is the additional packages that are available for download that are written by other

R users and developers. A large number of these packages are hosted on the Comprehensive R Archive Network (CRAN, https://cran.r-project.org).

2.11.1 Installing packages

The `install.packages` function is used to install packages. For example, to install the `dplyr` package, the syntax is:

```
install.packages("dplyr")
```

This may require you to choose a CRAN mirror from which to download. These mirrors have copies of packages throughout the world so that downloading can be quick. You can choose based on which country you are in, or the `0-Cloud` mirror, hosted from RStudio, which is generally fast. This will download a package to your hard drive in your `R` "library".

2.11.2 Help

The `help` function will pull up the associated help file with a function. For example, if you would like to see the help file for the `filter` function, you can write:

```
help("filter")
```

The question mark (`?`) is also a shorthand function for pulling up the help:

```
?filter
```

The double question mark (`??`) is shorthand for the `help.search` function, which will search the help files for a pattern/word in multiple places of all the help files in all the packages installed.

```
??"linear model"
help.search("linear model")
```

Note: if you want to search something with a space using the `??` syntax, you need to encapsulate it in quotes.

2.11.3 Loading packages

Though `install.packages` downloads a package to your hard drive, this does not load the package into the `R` session. The package must be loaded with the `library` command:

```
library(dplyr)
```

Now that the `dplyr` package is loaded, you can use functions from that package, such as the `select` function. At any given point you may have tens or hundreds of packages downloaded/installed on your computer, but only load a handful when doing an analysis or calling a script.

2.11.4 Function masking

Most packages are loaded without any messages. But in the case above, we have a message about "masking". To explain masking, let us say that a package (package A) is loaded into memory/the `R` session with a function called `fcn` and another package (package B) is loaded with a function called `fcn`. Now, since package B was loaded second, if you type `fcn`, then `R` will assume you want `fcn` from package B. Hence, the **order** of loading packages may matter, and the function `fcn` from package A is "masked". It is still available to be used, but you have to explicitly state that you want the package A version. This can be done using the double colon operator (`::`). If you write `A::fcn`, then `R` understands that you want to use the function `fcn` from package A, regardless if package B is loaded.

Most of the time, this is not an issue due to the fact that many packages do not have functions with the same name. Also, in many `R` scripts, only the packages necessary for the analysis are loaded. When `R` is started, the `base` and `stats` packages are loaded by default. When loading the `dplyr` package above, the `filter` function from the `stats` package is masked by the `dplyr filter` function. If users want to use the `stats filter` function, then they can by running `stats::filter`. Note, these functions are very different: `stats::filter` applies linear filters of time series, whereas the `dplyr::filter` function is for subsetting rows of a `data.frame`. If you use the help function `?filter` after loading in `dplyr`, it will ask which help file you want to see.

Function masking is different from extending functions from other packages. For example, if package B depends on package A and extends the `fcn` function to do things to a **different set of classes**, as opposed to being a different function altogether, then this is a different situation.

2.12 dplyr, ggplot2, and the tidyverse

Most of the basics of `R` and subsetting have been discussed. For most long-term `R` users, the syntax of these processing steps is how they were taught and continue to use `R`. In the past few years, a series of additional tools for data manipulation, analysis, and visualization have been proposed. For data manipulation, the `dplyr` function has added a set of "verbs" for data analysis. Wickham (2014) proposed a framework for data structures of "tidy data", where:

> each variable is a column, each observation is a row, and each type of observational unit is a table

The `ggplot2` package (Wickham 2016) added a rigorous visualization framework in `R`, based on the "Grammar of graphics" (Wilkinson 2006). These packages alongside an additional set of packages developed with these new principles make up the "tidyverse" (https://www.tidyverse.org/). The `tidyverse` package is an umbrella package that contains these packages so that one can be installed and loaded if abiding by this philosophy. This package supplements the `base` package of `R`, commonly referred to as "base `R`". Though we will use both base `R` and the tidyverse, we believe most serious analysts agree that the recommendation is to use the tools that allow you to quickly explore your data in a way that is readable to another programmer.

2.12.1 Tibbles

A `tbl` ("tibble") is an additional data class/container to the ones mentioned above, implemented in the `tibble` package. It is like `data.frame` above, but with some very important differences (adapted by the help file for `tibble`) :

- Never coerces inputs (i.e. strings stay as strings!).
- Never adds row.names.
- Never changes column names.
- Only recycles length 1 inputs. (no wraparound)
- Automatically adds column names.

If you subset a `tbl` using bracket notation, but only the variable is given, then that always returns a `tbl`:

```
tbl = tibble::as_tibble(df)
class(tbl[, "age"])
```

```
[1] "tbl_df"     "tbl"        "data.frame"
```

This behavior is different from that of a `data.frame`, which would have returned a vector. As of version 1.4.1, tibbles can also use the `drop` argument :

```
tbl[, "age", drop = TRUE]
```

```
[1] 25 30 32 42
```

Also, if you accidentally print a large `data.frame`, it will try to print as many rows as the default allows and all the columns. This printing can lock your `R` console for a while, and is usually not helpful. A `tbl` will only print out the first few rows and columns that fill your screen by default. You can construct a `tbl` similar to a `data.frame` above, but using the `data_frame` function (note the difference between the period and underscore). If you want to convert a `data.frame`, you can use `dplyr::as.tbl` or `tibble::as_tibble`. Some data input functions will be introduced later to show how to read in data as `tbl`s.

2.12.2 The pipe (%>%)

One additional tool, which has been incorporated into the tidyverse, is the "pipe," which is the `%>%` operator. The pipe was introduced in the `magrittr`, in reference of the painting "The Treachery of Images" from Ren'e Magritte. Here we have a simple vector `x`:

```
x = c(5, 2, 3, 42, 6)
```

Let us look at a few ways to calculate the square root of the mean of `x` and assign it to `y`. Let us try one line:

```
y = sqrt(mean(x))
```

This operation is clear and not too cluttered. Note, `R` evaluates using the "middle-out" process, where you can think of a function as parentheses, in the order of mathematical operations.

Let us try two lines:

```
y = mean(x)
y = sqrt(y)
```

Here we assign `y` to the mean of `x`, then reassign `y` to the square-root of the mean. This process is clear and follows linearly, but is a bit unnecessary for such a simple action. Below we can use the pipe:

```
y = x %>% mean %>% sqrt
```

You can think of the pipe as reading "then." So we take `x`, then perform the `mean`, then the square-root. The default in using the pipe is that the object on the left hand side of the pipe is the first argument of the function you are piping into on the right hand side. There are ways around this, but the `tidyverse` package generally abides by this when designing functions. The pipe, at first, may not seem as compelling, but code can become more complicated. Many analysis steps involves taking a data set, subsetting the columns, transforming variables, sorting the rows, and summarizing the data set into summary statistics. These operations are usually too complicated to put in one line, but work fine with reassignment and the pipe.

2.12.3 White space

In `R`, you can write an expression on multiple lines or on the same line as long as that function is not a completed function by the end of the line. For example, the previous piping code could also have been written as:

```
sqrt(
  mean(x)
)
```

```
[1] 3.405877
```

or

```
x %>%
  mean %>%
  sqrt
```

```
[1] 3.405877
```

As the pipe assumes there is something on the right hand side of it, `R` sees the pipe at the end of the first line and knows to "keep going." Similarly, `R` knows that the `sqrt` function must have both an open parenthesis and a closed parenthesis and it knows to keep looking for more information in the following lines. If you are working interactively in `R` and forget to close a parenthesis, quote, or curly brace or have an unfinished line, `R` will have a `+` symbol instead of the standard prompt symbol `>`. If you would like to stop the command and start another, you can break usually using the `Esc` key (in RStudio) or `Cmd+C` or `Ctrl+C` depending on your operating system.

2.13 Problems

Problem 1. Consider the vector:

```
num <- c("1.5", "2", "8.4")
```

a. Convert `num` into a numeric vector using `as.numeric`
b. Convert `num` into a factor using `factor`, calling it `num_fac`.
c. Convert `num_fac` into a numeric using `as.numeric`.
d. Convert `num_fac` into a numeric using `as.numeric(as.character())`

Problem 2. Consider the vector:

```
num <- c(0, 1, 1, 0)
```

a. Append `TRUE` to the `num` vector using `c`.
b. Check the class of `num` using `class`.
c. Convert `num` into a logical vector using `as.logical`.
d. Convert `num` into a logical vector where the result is `TRUE` if `num` is `0`, using the `==` operator.

Problem 3. Consider the data set `data_bmi` from above. Perform the following operations:

a. Extract the column nams of `data_bmi` using the `colnames()` function.
b. Subset the `AGE` column using `data_bmi[, "AGE"]`.
c. Subset the `AGE` column using `data_bmi$AGE`.
d. Subset the `AGE` and `BMI` columns using brackets and the `c()` function.

Problem 4. Here we will work with sequences and lengths:

a. Create a sequence from `1` to `4.5` by `0.24`. Call this `run_num`.
b. What is the length of `run_num`.
c. Extract the fifth element of `run_num` using brackets.

Problem 5. Let's create a `tibble` called `df`:

```
df = dplyr::data_frame(x = rnorm(10), y = rnorm(10), z = rnorm(10))
```

a. Extract the column `x` using the `$`.
b. Extract the column `x` using the `[,]` notation.
c. Extract columns `x` and `z`.
d. Extract the third and fifth rows of `df` and columns `z` and `y`.

Problem 6. Let's create a `tibble` called `df`:

```
df = dplyr::data_frame(x = rnorm(10), y = rnorm(10), z = rnorm(10))
```

a. Get the mean of the column `x` using the `$` operator.
b. Pipe (`%>%`) the column `x` into the mean function.
c. Look at the `summarize` function to look ahead to future chapters and try to do this in the `dplyr` framework.

Problem 7. Consider the data set `data_bmi` using the BMI data, but read in using `readr`:

```
file_name = "bmi_age.txt"
data_bmi = readr::read_table2(file = file_name)
```

a. What is the class of `data_bmi`?
b. What is the class of `data_bmi[, "AGE"]`?
c. What is the class of `data_bmi$AGE`?
d. What is the class of `data_bmi[, "AGE", drop = TRUE]`?

Problem 8. Consider the data set `data_bmi` using the BMI data, but read in using `readr`:

```
file_name = "bmi_age.txt"
data_bmi = readr::read_table2(file = file_name)
```

```
Parsed with column specification:
cols(
  PID = col_double(),
  BMI = col_double(),
  SEX = col_double(),
  AGE = col_double()
)
```

a. What is the mean of the `AGE` column?
b. Set the 3rd element of `data_bmi$AGE` to be 42?

c. What is the mean of the `AGE` column now?

Problem 9. Use `data_bmi` from the above problem:

a. Remove the `X5` column using `data_bmi$X5 = NULL`
b. Create `mat`, which is `data_bmi` as a matrix, using `as.matrix`.
c. Try to extract `AGE` from `mat` using the `$`. What happened?
d. Extract `AGE` from `mat` using the `[,]` notation.

Chapter 3

Probability, random variables, distributions

This chapter covers the following topics

- Experiments
- An intuitive introduction to the bootstrap
- Probability and probability calculus
- Sampling in R
- Random variables
- Probability mass function
- Probability density function
- Cumulative distribution function
- Quantiles
- Supplementary `R` training

3.1 Experiments

An experiment is the process of data collection for a target population according to a specific sampling protocol that includes rules for what, when, where, and how to collect data on experimental units (e.g. individuals) from the target population. For example, in a clinical trial for a diabetes prevention treatment the target population could be all individuals who are pre-diabetics before the trial starts and satisfy additional, well-defined, inclusion criteria. Any subject from the target population who agrees to participate in the trial can be selected up to a pre-defined maximum sample size. Subjects are then randomized to treatment with some probability. In many cases, treatment is assigned with a probability of 0.5.

The primary outcome could be the level of glycated hemoglobin (HbA1C), which can provide a measure of how much glucose is in the bloodstream, a means of diagnosing diabetes. One could also collect demographic variables (e.g., age, gender, race) and additional health related outcomes (e.g,. adverse effects, blood pressure, blood glucose variability). Data are sampled before the trial starts, at baseline (when the trial starts), at various interim visits (e.g., every three months), and at the end of the trial. Collecting the data is done during scheduled in-hospital visits, but could also entail continuous monitoring (e.g., using implantable glucose monitors) or ecological momentary assessment (EMA) using phone or computer apps. Data can be collected by obtaining blood and urine samples, downloading the data from the implantable monitor, and via the internet or cellular network. Designing such a trial requires intense pre-planning and a rigorous schedule to obtain high quality data. Of course, not all experiments are as complex. For example, it would be interesting to take any experiment that one could think about and parse it into its components. The simplest such experiment is flipping a coin or waiting for nature to flip the coin of who will get lung cancer in the next 10 years.

Consider the outcome of an experiment such as:

- a collection of measurements from a sampled population
- measurements from a laboratory experiment
- the result of a clinical trial
- the results of a simulated (computer) experiment
- values from hospital records sampled retrospectively

3.1.1 Notation

To develop these concepts rigorously, we will introduce some notation for the experiment setup.

- The sample space Ω is the collection of possible outcomes of an experiment
  Example: a six-sided die roll = {1; 2; 3; 4; 5; 6}, a coin flip = {heads; tails}
- An event, say E, is a subset of Ω
  Example: die roll is even E = {2; 4; 6}
- An elementary or simple event is a particular result of an experiment
  Example: die roll is a four, $\omega = 4$
- $\emptyset$ is called the null event or the empty set

The sample space can be extremely complex or very simple. Let us consider the die roll example. If the die is rolled once, any number between 1 and 6 could be the outcome for a total of 6 possible outcomes. However, if the experiment is to roll the die twice, the outcome is the pair of all possible combinations $(1,1),(1,2),\ldots,(6,6)$ for a total of 36 possible outcomes; here we count $(1,2)$ and $(2,1)$ as two different outcomes. If the experiment is to roll the die n times, the outcomes are all the possible n-tuples with the numbers 1 through 6 in every

position for a total number of outcomes equal to 6^n. To get an idea of just how big this sample can be, note that the number of possible ordered n-tuples for $n = 130$ is larger than 10^{82}, the estimated number of atoms in the known universe (http://www.universetoday.com/36302/atoms-in-the-universe/).

Thus, it would make little sense to play this game until all possible outcomes have been obtained, but it would make perfect sense to predict the mean of the n outcomes or predict how many outcomes will be larger than 5. Biostatistics is concerned with extracting useful, actionable information from complex outcomes that can result from even the simplest experiments.

While the fair die example may be a little dry, consider the example of a study of cognitive impairment in older adults. In this case, we could conceptualize the assignment of health status as a die where three out of the six faces have a one on it, two faces have a two on it, and one has a three on it. This assigns to individuals one of the following possible levels of cognitive impairment (none, moderate, severe). The die is considered unfair because the probabilities of assigning levels of impairment are likely not to be equal; this is an example when unfairness is probably good, as most people will be assigned to "non-impaired". Nonetheless, *alea iacta est* ("the die is cast").

3.1.2 Interpretation of set operations

Data analysis, manipulation, and tabulation are intrinsically related to data sets and logic operators. Indeed, we often try to understand the structure of the data and that requires understanding how to subset the data and how to quantify the various relationships between subsets of the data. To better understand this, we need to introduce the theoretical set operations and their interpretations.

- $\omega \in E$ implies that E occurs when ω occurs
- $\omega \notin E$ implies that E does not occur when ω occurs
- $E \subset F$ implies that the occurrence of E implies the occurrence of F (E is a subset of F)
- $E \cap F$ implies the event that both E and F occur (intersection)
- $E \cup F$ implies the event that at least one of E or F occur (union)
- $E \cap F = \emptyset$ means that E and F are mutually exclusive/disjoint, or cannot both occur
- E^c or $\bar{E}$ is the event that E does not occur, also referred to as the complement
- $E \setminus F = E \cap F^c$ is the event that E occurs and F does not occur

3.1.3 Set theory facts

DeMorgan's laws.

- $(A \cup B)^c = A^c \cap B^c$

Example: If an alligator or a turtle you are not $(A \cup B)^c$ then you are not an alligator and you are also not a turtle $(A^c \cap B^c)$

- $(A \cap B)^c = A^c \cup B^c$

  Example: If your car is not both hybrid and diesel $(A \cap B)^c$ then your car is either not hybrid or not diesel $(A^c \cup B^c)$
- $(A^c)^c = A$
- $(A \cup B) \setminus C = (A \setminus C) \cup (B \setminus C)$ where $\setminus$ means "set minus"

Proving the equality of two sets, $A = B$, is done by showing that every element of A is in B (i.e., $A \subset B$) and every element in B is in A (i.e., $B \subset A$). For example, let us show that $(A \cup B)^c = A^c \cap B^c$. We first show that $(A \cup B)^c \subset A^c \cap B^c$. Let $\omega \in (A \cup B)^c$, which, by definition, implies that $\omega \notin (A \cup B)$. Thus, $\omega \notin A$ and $\omega \notin B$, which implies $\omega \in A^c \cap B^c$. Indeed, by *reductio ad absurdum*, if $\omega \in A$ or $\omega \in B$, then, by definition, $\omega \in A \cup B$, which would be contradictory. The fact that $(A \cup B)^c \supset (A^c \cap B^c)$ can be shown using a similar approach. Several problems in this chapter can be solved using this general strategy. The approach is not difficult, but it needs to follow the strict rule of logic.

In general, set operations are based on logical operators: AND (in `R &`), OR (in `R |`), NOT (in `R !`). These logical operators can be combined and can produce complex combinations that are extremely useful in practice. For example, having a data set one may want to focus on a subset that contains only African American men from age 65 to 75 who are non-smokers. This phrase can immediately be translated into logical operators and the data can be extracted using `R` programming. This is a routine application of set theory that will become indispensable in practice (below we provide an example of exactly how to proceed).

We start by reading a small data set. First, let us tell `R` where the file is:

```
file_name = file.path("data", "bmi_age.txt")
```

Here, the variable/object `file_name` contains the path of the dataset, stored in a text file `bmi_age.txt`. In this example, the file is located in the `data` sub-folder and the path is **relative** to the current working directory (e.g., is a sub-folder). The `file.path` function is used to create this path, inserting forward slashes (`/`) where necessary. If you download and use the data then change the file path to the correct folder on your computer.

The data are loaded using the `read.table` function below. We specify arguments/options to `read.table` that the file contains a header row (`header = TRUE`) and we do not want to convert strings/words to a categorical variable/factor when the data are read in (`stringsAsFactors = FALSE`).

```
data_bmi = read.table(file = file_name, header = TRUE,
                      stringsAsFactors = FALSE)
```

We can simply write the object name out to show the data (here the entire data set is displayed because it is small).

```
data_bmi
```

```
   PID BMI SEX AGE
1    1  22   1  45
2    2  27   0  57
3    3  31   1  66
4    4  24   1  49
5    5  23   0  33
6    6  18   0  40
7    7  21   0  65
8    8  26   1  59
9    9  34   1  65
10  10  20   0  42
```

We also display the dimension of the data, which indicates that there are 10 rows and 4 columns:

```
dim(data_bmi)
```

```
[1] 10  4
```

using the `dim` function. In general, datasets are much larger and displaying all data would not be feasible. Displaying only a subset of the dataset would provide the general structure. This can be done using `head(data_bmi)` instead of `data_bmi`. If data are large with many variables, then one could just look at the name of the variables (column names):

```
colnames(data_bmi)
```

```
[1] "PID" "BMI" "SEX" "AGE"
```

We can also do simple data calculations, such as estimate the mean and standard deviation of the BMI variable:

```
mean(data_bmi$BMI)
```

```
[1] 24.6
```

```
sd(data_bmi$BMI)
```

```
[1] 4.993329
```

Again, we use the dollar sign (`$`) to indicate that we are extracting a single variable from the `data_bmi` dataset. For simplicity, we will assign all the columns of the dataset to their own separate variables using the `attach` function. We do not recommend this approach in general, especially when multiple datasets

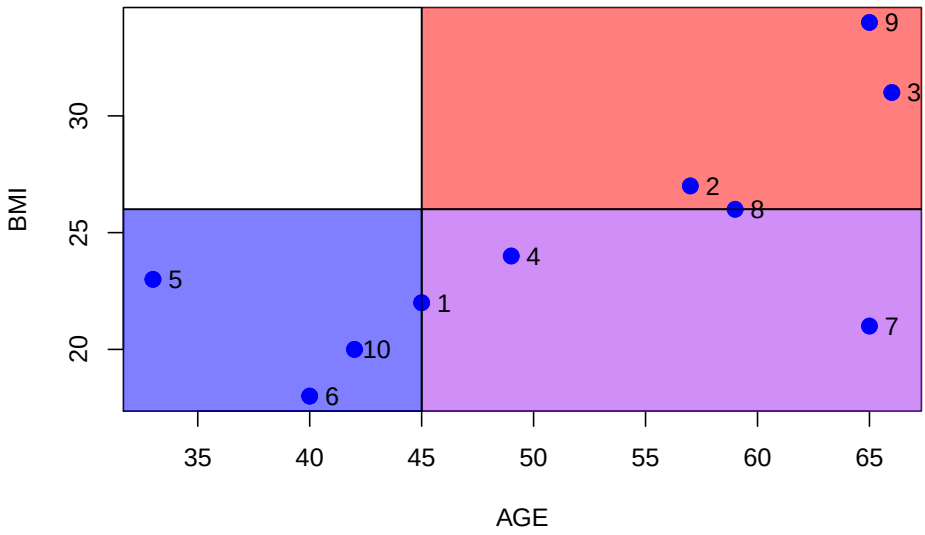


Figure 3.1: BMI and Age Relationship.

are being used, since variables across records can become de-linked and independently sorted. We will do this only for this data set to simplify the code example.

```
attach(data_bmi)
```

We can also show a basic plot (fancier plotting options will be shown later) of age vs. BMI:

```
plot(AGE, BMI,type="p",pch=20,cex=2,col="blue")
rect(xleft = 45, xright = 100, ybottom = 0, ytop = 26,
     col = scales::alpha("purple", 0.5))
rect(xleft = 45, xright = 100, ybottom = 26, ytop = 100,
     col = scales::alpha("red", 0.5))
rect(xleft = 0, xright = 45, ybottom = 0, ytop = 26,
     col = scales::alpha("blue", 0.5))
points(AGE, BMI,type="p",pch=20,cex=2,col="blue")
text(x = AGE + 1, y = BMI, labels = PID, col = "black")
abline(h = 26, v = 45, col = "black")
```

Here the call to `abline` is for drawing vertical and horizontal lines so that we can count the numbers in each respective box. We now make the connection between set theory and operations clearer in the context of the data. In particular, we emphasize how set operations translate into logic operators that can then be used for data extraction and operations. Consider the following subsets of subjects: A subjects with BMI$<$ 26 and B subjects older than 45 (AGE$>$ 45). Construct a logical indicator for which records fall into A

```
index_BMI_less_26 = BMI < 26
```

which is represented by the points in the purple/blue regions in Figure 3.1, and B

```
index_AGE_greater_45 = AGE > 45
```

which is represented by the points in the purple/red regions in Figure 3.1. Display the IDs for A and B. Here PID is the unique patient ID, though in many applications the PID can be more complex than just integer numbers.

```
PID[index_BMI_less_26]
```

```
[1]  1  4  5  6  7 10
```

```
PID[index_AGE_greater_45]
```

```
[1] 2 3 4 7 8 9
```

Let us calculate $(A \cap B)^c$, the complement of the intersection between A and B, which is shown by the non-purple regions in Figure 3.1. These are subjects who do not/are (note the c that stands for complement) (have a BMI less than 26) and (older than 45) and

```
index_A_int_B_compl = !(index_BMI_less_26 & index_AGE_greater_45)
PID[index_A_int_B_compl]
```

```
[1]  1  2  3  5  6  8  9 10
```

The translation of $(A \cap B)^c$ into `R` is `!(index_BMI_less_26 & index_AGE_greater_45)`. Note that `!` indicates is not, or complement, and `&` indicates and, or intersection. So, the resulting IDs are everybody, except the subject with IDs 4 and 7. It would be instructive to conduct the same type of analysis for $A^c \cup B^c$, $(A \cup B)^c$, and A^c. While this dataset is relatively small for educational purposes, similar subsetting approaches are both essential and impossible to do by hand in large datasets.

3.2 An intuitive introduction to the bootstrap

A major problem in practice is that, even if we run two identical experiments, data are never the same. For example, if two large epidemiological studies collect data on the effects of smoking on developing lung cancer, and have the exact same sampling protocol and length, then the studies will select different people, with different demographic information and different times of transition from being healthy to having lung cancer. However, even though the two samples will be different, they will have some things in common. Those things are the target of estimation, the probability and time for conversion from being healthy to developing lung cancer, and the original target population (e.g., the

US population). In practice we often have one study and we are interested in understanding what would happen if multiple studies were conducted. The reason for that is fundamental and it has to do with the generalizability of the experiment. Indeed, a study would collect data for a subsample of the population to make predictions about the rate and expected time of transition to lung cancer in the overall population.

Bootstrap is a widely used statistical technique based on resampling the data. The bootstrap is designed to create many potential studies that share the same characteristics with the study that collected the data and may represent the variability of running these experiments *without actually running the experiments.*

The nonparametric bootstrap is the procedure of resampling with replacement from a dataset, where the number of observations in each resampled dataset is equal to the number of observations in the original dataset.

Given a dataset and a function of the data (e.g., mean, standard deviation, predictor), the bootstrap characterizes the distribution of the function of the data (a.k.a. estimator) by producing a set of possible values for that function of the data. This is done by resampling the data with replacement and applying the function of the data for every sample. The general form of the bootstrap is:

The nonparametric bootstrap algorithm

INPUT: data, D, number of bootstraps, B, function $f : S \to \mathbb{R}^p$, where S is the data space
Repeat for $b = 1, \ldots, B$

- Step 1. Resample the data with replacement and denote it D^b
- Step 2. Calculate the statistic $X_b = f(D^b)$

OUTPUT: $X_1, \ldots, X_B$

The simplicity of the algorithm is exceptional, which is why it has become one of the standard tools for quantifying the sampling distribution of an estimator. The bootstrap was introduced by Efron (1979) and is typically taught as an advanced method after many other concepts are covered. However, the basic mechanics of the bootstrap are extremely easy to use in practice. The justification, the interpretation, and the theoretical underpinnings of the bootstrap are more difficult, but for now we will not concern ourselves with these details.

The easiest way to understand the bootstrap is to apply it to the data and analyze the format of the data at intermediary steps and results. Below we introduce some more advanced code to illustrate the bootstrap. An in-depth introduction of the bootstrap will be done later. The bootstrap is a resampling technique that is extremely powerful and relatively easy to understand. Let us conduct $B = 3$ bootstrap samples of the data and print them. This procedure is referred to as nonparametric bootstrap of subjects.

```
set.seed(4132697)
nboot=3                    #number of boostrap samples
nsubj=nrow(data_bmi)       #number of subjects
for (i in 1:nboot)         #start bootstrapping
  {#begin the bootstrap for loop
  resampling = sample(nsubj, replace=TRUE)     #sample with replacement
  bootstrapped_data = data_bmi[resampling,]    #resample the data set
  print(bootstrapped_data)                    #print bootstrapped datasets
  cat(" \n \n")                               #add lines between datasets
  }#end the bootstrap for loop
```

```
     PID BMI SEX AGE
6      6  18   0  40
8      8  26   1  59
2      2  27   0  57
10    10  20   0  42
3      3  31   1  66
10.1  10  20   0  42
9      9  34   1  65
3.1    3  31   1  66
1      1  22   1  45
4      4  24   1  49
```

```
    PID BMI SEX AGE
6     6  18   0  40
6.1   6  18   0  40
10   10  20   0  42
8     8  26   1  59
7     7  21   0  65
8.1   8  26   1  59
1     1  22   1  45
1.1   1  22   1  45
6.2   6  18   0  40
5     5  23   0  33
```

```
    PID BMI SEX AGE
7     7  21   0  65
9     9  34   1  65
3     3  31   1  66
7.1   7  21   0  65
3.1   3  31   1  66
9.1   9  34   1  65
8     8  26   1  59
```

```
9.2   9  34   1  65
2     2  27   0  57
8.1   8  26   1  59
```

A close look at these bootstrap samples will indicate what is meant by creating many potential studies that share the same characteristics. The subjects in every bootstrap sample are the same as those in the data, though some may not appear and others may appear multiple times. All information about the subject is preserved (information across columns is not scrambled), and the number of subjects in each bootstrap sample is the same as the number in the original dataset. This was done by using *sampling with replacement* of the subjects' IDs using the same probability. Sampling with replacement can be conceptualized as follows:

1) consider a lottery machine that contains identical balls that have the name, or IDs, of n subjects;
2) sample one ball, record the ID and place the ball back into the lottery machine;
3) repeat the experiment n times.

The reason why this type of sampling is called *with replacement* is that the ball is placed back into the lottery machine. If the ball is not placed back into the lottery machine, the sampling scheme is quite different and the sampling is called without replacement. The results of sampling with replacement can contain the same number, or ID, multiple times because the ball is placed back into the lottery machine and is allowed to be re-sampled. The results of sampling *without replacement* cannot contain the same number, or ID, more than once.

In general, $B = 100$ bootstraps can be sufficient for many applications, but $B = 10000$ or more may be needed in others. The processing time for the bootstrap is linear in B, but the total computation time can be dramatically reduced by parallel computing (running on multiple computers simultaneously). This can be especially useful when obtaining $f(D^b)$ is time-intensive. Here we set the seed and the same exact boostrap samples are obtained every time this book is compiled. However, in practice if you run the code multiple times without the `set.seed()` option do not be surprised that every time the 3 bootstrap data sets change. This happens because of the random nature of the sampling mechanism. The total number of different nonparametric bootstrap datasets is 10^{10}, or 10 billion. If you want to always obtain the same dataset you can set the seed (this would need to be done before the sampling code) of the pseudo-random number (PRN) generator in `R`

```
set.seed(4132697)
```

The `set.seed()` function resets the PRN for the next simulation and does not have an effect on the previous simulation. There is nothing special about the

seed we chose above. You do not have to set a seed, then randomly generate another seed; one seed is sufficient. One strategy is to use the date of when the code was created: (e.g. `20180211`).

Resampling techniques, like the bootstrap, will be used extensively in practice. One of the most important applications is to quantify the variability of estimators and produce confidence intervals for specific parameters of interest. Consider, for example, the case when we would like to construct a 95% bootstrap confidence interval for the difference in the means of BMI of men and women, given this data set. The concept of a confidence interval has not been formally introduced. Instead, we simply focus on finding the values of the difference in the mean that could be obtained if one collected data on five other men and five other women from the same target population. Because we do not have the target population, we simply resample with replacement from the five men and five women from the data we have. First, we calculate the difference in average BMI between men and women in the original sample

```
men_bmi<-BMI[SEX==1]                    #vector of BMI values for men
women_bmi<-BMI[SEX==0]                  #vector of BMI values for women
n_women<-length(women_bmi)              #number of women in the dataset
n_men<-length(men_bmi)                  #number of men in the dataset
```

Display the difference in mean BMI between women and men

```
m_diff <- mean(men_bmi) - mean(women_bmi)
m_diff
```

```
[1] 5.6
```

This is the observed difference in the mean BMI between men and women. In biostatistics we call this the estimator, which is a function of the data. This number, 5.6, in itself cannot predict exactly what the mean difference in BMI will be in an identical study that collected data on five other men and five other women. It is natural to expect that the numbers observed in other studies should be close to 5.6, but we would like to know just how close. It is useful to observe that in this context, the function $f(\cdot)$ introduced in the bootstrap definition is the mean of the BMI for men minus the mean of the BMI for women. In general, this function can be much more complicated, as long as it is computable.

Here we conduct a bootstrap with 10000 resamples of the data. The philosophy is quite simple: create bootstrap samples of the data and for each sample calculate the mean difference between the BMI of men and women.

```
B_boot<-10000                           #number of bootstrap samples
mean_diff=rep(NA,B_boot)                #define the vector storing the differences in BMI
for (i in 1:B_boot)
  {#Begin bootstrap
    mm<-mean(men_bmi[sample(1:n_men,replace=TRUE)])       #calculate the mean for men
  mw<-mean(women_bmi[sample(1:n_women,replace=TRUE)])   #calculate the mean for women
```

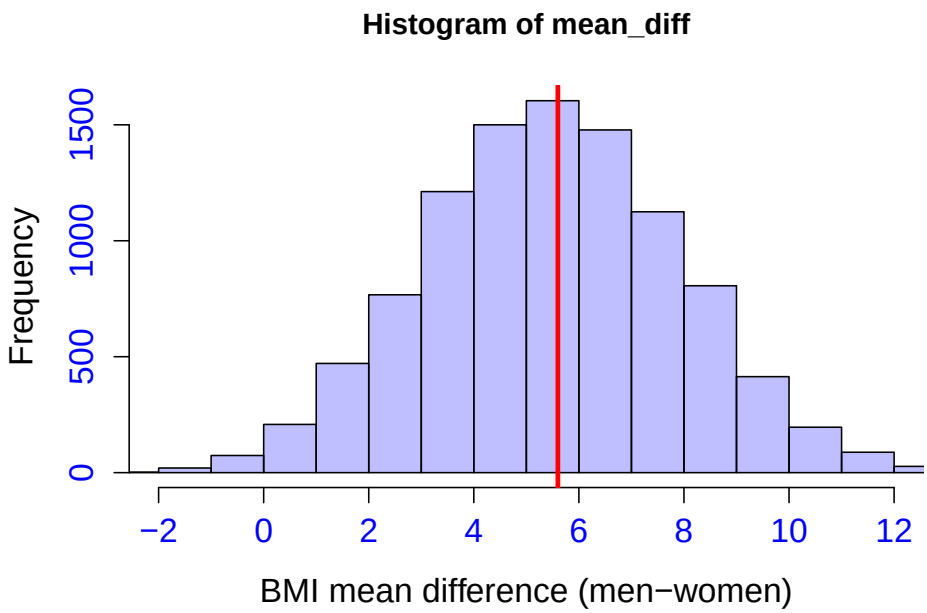


Figure 3.2: Histogram of 10000 bootstrap samples of the difference in mean between the BMI of women and men.

```
    mean_diff[i]<-mm-mw   #calculate and store the difference
    }#End bootstrap
```

The result of the bootstrap sample is a vector of length 10000 that contains plausible results for differences in the mean of the BMI of men and women. These results will tend to be close to 5.6 with some results a little farther away. A good way to visualize just how close these plausible values are to 5.6 is to plot the histogram of the 10000 bootstrap samples of the mean difference:

```
hist(mean_diff,breaks=20,col=rgb(0,0,1,1/4), xlim=c(-2,12),
     xlab="BMI mean difference (men-women)", cex.lab=1.3,
     cex.axis=1.3, col.axis="blue")
abline(v = m_diff,col="red",lwd=3)
```

Figure 3.2 displays this histogram and indicates that there are more than 1500 (15%) plausible values for the difference in means between 5 and 6 with very few values below 0 or above 12. While these results cannot pinpoint what will be the difference in a different study, they can provide information about how close to 5.6 these results are likely to be. Of course, if the sample size were larger, the plausible values would be closer to 5.6. This is sometimes referred to as the sampling variability. All concepts will be rigorously defined in later chapters.

With this distribution, we can calculate the mean, standard deviation, and 95%

confidence intervals (CIs) based on the Normal approximation or the bootstrap distribution (more details later once we learn about quantiles of the standard normal and empirical quantiles of a distribution).

```
mBoot<-mean(mean_diff)                          #calculate the mean of the differences
sdBoot<-sd(mean_diff)                           #calculate the sd of the differences
CI1<-c(mBoot-1.96*sdBoot,mBoot+1.96*sdBoot)     #normal approximation to the 95% CI
CI2<-quantile(mean_diff,probs=c(0.025,0.975))   #bootstrap 95% CI
```

Display and compare the two 95% CIs (normal and bootstrap approximations)

```
round(CI1, digits = 2) # normal approximation
```

```
[1]  0.82 10.35
```

```
CI2  # bootstrap approximation
```

```
 2.5% 97.5%
  0.8  10.4
```

The difference in length of the bootstrap and Normal approximation intervals as percent of the length of the Normal interval is:

```
[1] 0.69
```

which is less than 1% difference. Figure 3.3 displays the bootstrap results together with the 95% Normal approximation confidence intervals (red) and the bootstrap quantile approximation (blue):

```
hist(mean_diff, probability=TRUE, col=rgb(0,0,1,1/4), breaks=20,
     xlim=c(-2,12), xlab="BMI mean difference (women-men)", cex.lab=1.3,
     cex.axis=1.3, col.axis="blue")
abline(v = CI1,col="red",lwd=3)
abline(v = CI2,col="blue",lwd=3)
```

3.3 Probability

We have already introduced some concepts (bootstrap, confidence intervals) that have not been defined, though are very useful and intuitive. To better understand and anchor these concepts in a conceptual framework we need to introduce the notion of probability. Much of biostatistics uses this concept. For example, one is often interested in answering questions of the type:

(1) what is the probability that a 56 year-old female who was just diagnosed with breast cancer will die in 10 years?
(2) how much will the 1-year survival probability increase for a patient who receives a kidney transplant?
(3) what are the odds of becoming infected with the human papillomavirus (HPV) from an unprotected sexual contact?

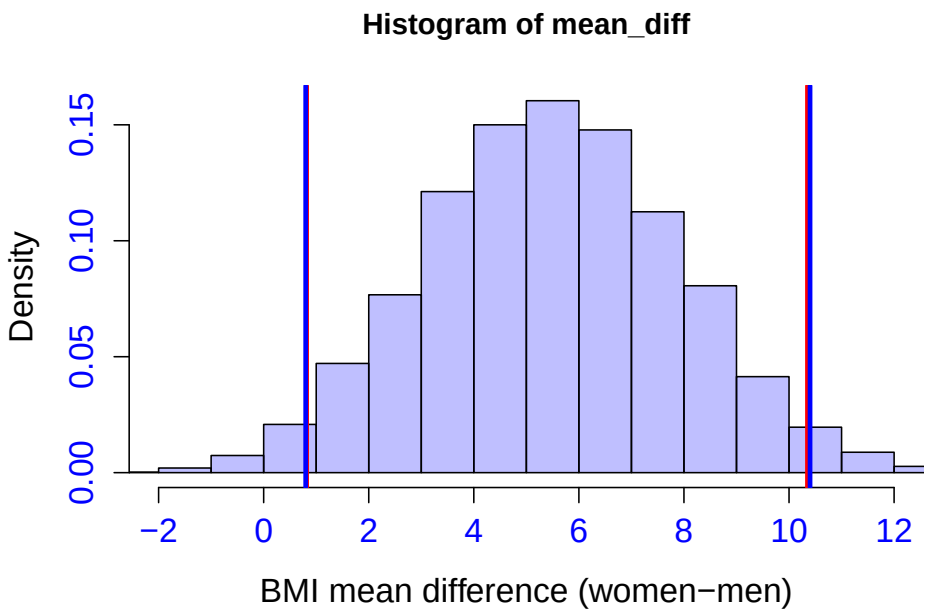


Figure 3.3: Histogram of 10000 bootstrap samples of the difference in mean between the BMI of women and men with 95% cofidence intervals.

All of these questions, and many others, rely on the idea that such a thing as probability exists. Probability has been found to be extraordinarily useful, even if true randomness is an elusive, undefined, quantity. While there is complete agreement on the mathematical rules probability must follow, disagreements about the *interpretation* of probability persist. For example, the frequentist interpretation of probability is the *proportion of times an event occurs in an infinite number of identical repetitions of an experiment.* The Bayesian interpretation of probability is *a subjective degree of belief* with the understanding that for the same event, two separate people could have different probabilities. Bayesian statistics was named after the 18^{th} century Presbyterian Minister and mathematician Thomas Bayes (Bayes, Price, and Canton 1763). While philosophical differences can be quite large, the practical differences to data analysis and inference are negligible.

3.4 Probability calculus

A probability measure, P, is a real valued function defined on the collection of possible events so that the following hold:

1. For every event $E \subset \Omega$, $0 \leq P(E) \leq 1$
2. $P(\Omega) = 1$

3. If E_i, for $i = 1, \ldots, \infty$ are mutually exclusive events ($E_i \cap E_j = \emptyset$ for every $i \neq j$) then
$$P\left(\bigcup_{i=1}^{\infty} E_i\right) = \sum_{i=1}^{\infty} P(E_i) \; .$$

This is called *countable additivity*. Countable additivity implies finite additivity as can be seen by setting $E_i = \emptyset$ for $i \geq N$ in the previous equation, where N is an arbitrary finite number. It is easy to show that $P(\emptyset) = 0$, which implies:

$$\begin{aligned} P\left(\bigcup_{i=1}^{N} E_i\right) &= P\left(\bigcup_{i=1}^{N} E_i \cup \emptyset\right) \quad \text{as any} \quad E = E \cup \emptyset \\ &= P\left(\bigcup_{i=1}^{N} E_i \cup \bigcup_{i=N+1}^{\infty} \emptyset\right) \\ &= \sum_{i=1}^{N} P(E_i) + 0 \quad \text{since} \quad E_i \cap \emptyset = \emptyset \\ &= \sum_{i=1}^{N} P(E_i) \; . \end{aligned}$$

Finite additivity does not imply countable additivity and it is not sufficient for defining probability. There exist probability theories that require only finite additivity, but they have strange properties. Note that for $N = 2$ we have

$$P(A \cup B) = P(A) + P(B)$$

if $A \cap B = \emptyset$. The probability function is defined on $\mathcal{F}$, which is a collection of subsets of the original set Ω. The subsets of Ω that belong in $\mathcal{F}$ are called measurable because they can be assigned a probability. When Ω is finite, things are clear, as $\mathcal{F}$ is the collection of all subsets of Ω and can be obtained by direct enumeration. For example, when $\Omega = \{1, 2, 3\}$ we have

$$\mathcal{F} = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$$

The number of possible events is $|\mathcal{F}| = 2^3 = 8$ when $|\Omega| = 3$. If $|\Omega| = n$ then $|\mathcal{F}| = 2^n$. Here $|A|$ denotes the cardinality of the set A and is equal to the number of elements in that set. While the size of $|\mathcal{F}|$ increases very quickly with n, to completely characterize the probability on $\mathcal{F}$ we only need the probability of the elementary events $\{1\}, \{2\}, \{3\}$. Indeed, if $p_1 = P(\{1\})$, $p_2 = P(\{2\})$ and $p_3 = P(\{3\})$ then we can easily calculate the probability of any other event. For example, $P(\{1, 2\}) = P(\{1\}) + P(\{2\}) = p_1 + p_2$. Note that p_3 is actually redundant as $p_3 = 1 - p_1 - p_2$, so the probability function on $\mathcal{F}$ is completely described by two probabilities, p_1 and p_2, of elementary events. If $|\mathcal{F}| = n$, the

probability function is completely defined by $n-1$ probabilities of elementary events.

When Ω is a continuous set, the definition gets much trickier. In this case we assume that $\mathcal{F}$ is sufficiently rich so that any set that we are interested in will be in it. For example, if $\Omega = \mathbb{R}$ is the set of real numbers then all intervals on the real line are Borel-measurable. This sounds fancy, but it basically means that the probability function assigns a probability to every interval of the form $[a, b]$, where a and b could be finite or infinite. As we will see in the case of probability distribution functions, the definition depends on defining probabilities of the type $P\{(-\infty, a]\}$ for all $a \in \mathbb{R}$. Without going into details, if the probability is defined on all intervals $[a, b]$ then it is also defined over all measurable sets. For practical purposes we will only deal with probabilities of intervals.

Based simply on the rules of probability and set theory it is relatively easy to prove

- $P(\emptyset) = 0$
- $P(E) = 1 - P(E^c)$
- $P(A \cup B) = P(A) + P(B) - P(A \cap B)$
- If $A \subset B$ then $P(A) \leq P(B)$
- $P(A \cup B) = 1 - P(A^c \cap B^c)$
- $P(A \cap B^c) = P(A) - P(A \cap B)$
- $P(\cup_{i=1}^n E_i) \leq \sum_{i=1}^n P(E_i)$
- $P(\cup_{i=1}^n E_i) \geq \max_i P(E_i)$

Let us prove a couple of these results. Note that, by definition, $\Omega = E \cup E^c$ and $E \cap E^c = \emptyset$. Thus,

$$1 = P(\Omega) = P(E \cup E^c) = P(E) + P(E^c) \ ,$$

which implies that $P(E) = 1 - P(E^c)$. The equality $1 = P(\Omega)$ follows from the axioms of probability, whereas the equality $P(E \cup E^c) = P(E) + P(E^c)$ follows from finite additivity and the fact that $E \cap E^c = \emptyset$. Some of the set theory results are left unproven and are given as problems at the end of this chapter. We will prove the following result $P(\cup_{i=1}^n E_i) \leq \sum_{i=1}^n P(E_i)$ to illustrate the concept of proof by induction. A proof by induction shows that the property (in this case the inequality) holds for $n = 2$ and proceeds by assuming that it holds for $n-1$ and showing that it holds for n.

Proof for $n = 2$:

$$\begin{aligned} P(\bigcup_{i=1}^{2} E_i) &= P(E_1 \cup E_2) \\ &= P(E_1) + P(E_2) - P(E_1 \cap E_2) \end{aligned}$$

and since $P(E_1 \cap E_2) \geq 0$ it follows that $P(E_1 \cup E_2) \leq P(E_1) + P(E_2)$ for every E_1 and E_2.

Assumption: Now let us assume that $P(\cup_{i=1}^{n-1} E_i) \leq \sum_{i=1}^{n-1} P(E_i)$ and take $A = \cup_{i=1}^{n-1} E_i$ and $B = E_n$.

Induction: Since $P(A \cup B) \leq P(A) + P(B)$ and $A \cup B = \cup_{i=1}^{n-1} E_i$ it follows that:

$$\begin{aligned} P(\bigcup_{i=1}^{n} E_i) &= P(\bigcup_{i=1}^{n-1} E_i \cup E_n) \\ &\leq P(\bigcup_{i=1}^{n-1} E_i) + P(E_n) \text{ since } P(E_n) \geq 0 \\ &\leq \sum_{i=1}^{n-1} P(E_i) + P(E_n) \text{ by assumption} \\ &= \sum_{i=1}^{n} P(E_i) \end{aligned}$$

where the second inequality is true by the assumption that the inequality holds for $n-1$. Getting a feel for proving elementary probability results can be done by trying to prove these results and using Venn diagrams to understand the symbols from set theory. Indeed, what may look daunting in terms of set theory symbols may actually be a simple concept to understand visually.

3.4.1 Example: Sleep disorders

The National Sleep Foundation (www.sleepfoundation.org) reports that around 3% of the American population has sleep apnea. They also report that around 10% of the North American and European population has restless leg syndrome (RLS). Similarly, they report that 58% of adults in the US experience insomnia. Does this imply that 71% of people will have at least one of these three sleep problems?

The answer is no because the events are not mutually exclusive. That is, if a person has sleep apnea it does not mean that they cannot experience insomnia or suffer from RLS. It is important to translate the words describing the problem and conceptualize them into statistical formulas that can be used to assist reasoning. To do this define

- $A_1 = \{\text{Person has sleep apnea}\}$
- $A_2 = \{\text{Person has RLS}\}$
- $A_3 = \{\text{Person has insomnia}\}$

The event {Person has *at least one* of the three sleep problems} can be written as $A_1 \cup A_2 \cup A_3$ and we are interested in quantifying $P(A_1 \cup A_2 \cup A_3)$, that

is, the probability that a random person chosen from the population will have at least one of the three sleep problems. We already know that $P(A \cup B) = P(A) + P(B) - P(A \cap B)$. Setting $A = A_1 \cup A_2$ and $B = A_3$ it follows that

$$P(A_1 \cup A_2 \cup A_3) = P(A_1 \cup A_2) + P(A_3) - P(\{A_1 \cup A_2\} \cap A_3)$$

Because $P(A_1 \cup A_2) = P(A_1) + P(A_2) - P(A_1 \cap A_2)$ and $P(\{A_1 \cup A_2\} \cap A_3) = P(\{A_1 \cap A_3\} \cup \{A_1 \cap A_3\})$ it follows that

$$P(A_1 \cup A_2 \cup A_3) = P(A_1) + P(A_2) + P(A_3) - P(A_1 \cap A_2) - P(\{A_1 \cap A_3\} \cup \{A_2 \cap A_3\})$$

By setting $A = A_1 \cap A_3$ and $B = A_2 \cap A_3$ we have $P(\{A_1 \cap A_3\} \cup \{A_2 \cap A_3\}) = P(A_1 \cap A_3) + P(A_2 \cap A_3) - P(A_1 \cap A_2 \cap A_3)$. Putting everything together we have

$$\begin{aligned} P(A_1 \cup A_2 \cup A_3) = & P(A_1) + P(A_2) + P(A_3) \\ & - P(A_1 \cap A_2) - P(A_1 \cap A_3) - P(A_2 \cap A_3) \\ & + P(A_1 \cap A_2 \cap A_3) \end{aligned}$$

So, $P(A_1 \cup A_2 \cup A_3) = 0.71 - \text{Other stuff}$, where Other stuff ≥ 0. Of course, we already knew that, but the formula above explains exactly *how* we have been overestimating $P(A_1 \cup A_2 \cup A_3)$ by $P(A_1) + P(A_2) + P(A_3) = 0.71$. More precisely, what would be left out is the counting of the subjects who have at least two of the sleep problems and those who have all three sleep problems. The formula suggests what additional information one would need to calculate the probability of having at least one of the three sleep problems.

3.4.2 Example: Birthday problem

In a given room what is the probability that at least two people have the same birthday (not necessarily the same year)? Assume that birthdays occur randomly and with the same probability throughout the year. Let us start by calculating this probability for a room with $n = 2$ people and observe that

$$P(\text{at least two}) = 1 - P(\text{none})$$

We will enumerate the number of possibilities for birthdays to occur such that no two birthdays fall on the same day. For a pair of two individuals there are a total of 365×365 possibilities for pairs of birthdays. The number of pairs of birthdays that are not the same is 365×364, because the first person can have any day of the year as a birthday (365) whereas the second person can only have any day of the year as birthday, except the birthday of the first person (364). Thus, when $n = 2$

$$P(\text{at least two}) = 1 - \frac{365}{365} \times \frac{364}{365} \, .$$

Using a similar reasoning for $n = 3$ we have

$$P(\text{at least two}) = 1 - \frac{365}{365} \times \frac{364}{365} \times \frac{363}{365} ,$$

and for a general n we have

$$P(\text{at least two}) = 1 - \frac{365}{365} \times \frac{364}{365} \times \ldots \times \frac{365 - n + 1}{365} .$$

There is an assumption that we have glossed over, which was the *randomness* assumption. This assumption allowed us to do the calculations by allowing the second individual to have any of the other days as a birthday irrespective of the birthday of the first individual. This, of course, would not be true if subjects were not chosen *independently from one another*. For example, if the two subjects were chosen to have the same birthday month, the assumption of independence would be violated and the probabilities would be very different (higher or lower?). The other assumption was that each day has an equal probability of being a birthday and that no particular days are favored to be birthdays in this population. Calculations would be computationally harder under the no equal probability assumption, but they are doable. We will discuss these concepts in detail, but it is important to observe how probability calculations and explicit assumptions about the sampling mechanisms go together.

We show how to calculate the probability for having two people with the same birthday in a room. If there are more than 365 people the probability is 1 (here we conveniently ignore leap years). Thus, we calculate the probability only for rooms with fewer than $n < 365$ people.

```
n=1:364                          #Vector of number of people in the room
pn=n                             #Vector that will contain the probabilities
for (i in 1:364) {
  pn[i]<- 1-prod(365:(365-i+1))/365^i
}  #Prob. of >= 2 people with same B-day
```

The function `prod()` calculates the product of the elements of the argument vector. In this case we are working with very large numbers, so it may be indicated to use `prod(365:(365-i+1)/365)` instead of `prod(365:(365-i+1))/365^i`, whereas the first statement takes the product of fractions and the second takes the products of integers, which can be **very large**, then divides it to make it a fraction. Another useful technique when calculating the product of a sequence of numbers, say $l_1, \ldots, l_n$ is to use $\prod_i l_i = \exp\{\sum_i \log(l_i)\}$. Of course, this is a trivial equality, but it is much easier to numerically calculate $\sum_i \log(l_i)$ and then take the exponential of this sum when l_i's are very large or very small. This will be the case especially when we will deal with *likelihoods*.

Also, instead of using the `for` loop above, we can use the `cumprod` function which does a `cum`ulative `prod`uct:

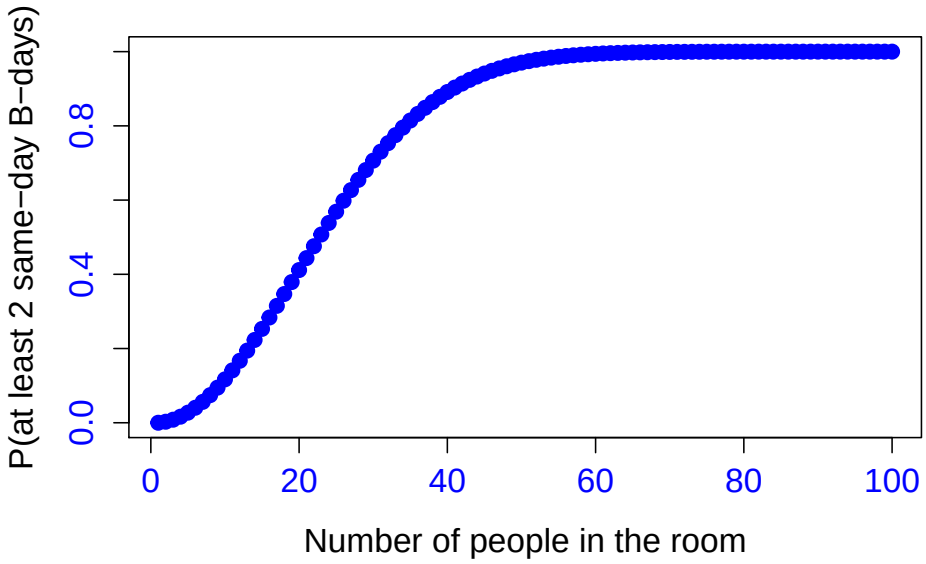


Figure 3.4: Probability that at least 2 people will have the same birtday (y-axis) as a function of the number of people in the room (x-axis.)

```
pn2 = 1 - cumprod(1:n/365)
```

```
Warning in 1:n: numerical expression has 364 elements: only the first used
```

Display results for the size of the group equal to 23 and 57.

```
round(pn[c(23,57)], digits = 3)
```

```
[1] 0.507 0.990
```

Note that for a group of 23 people there is a 50% chance that at least two people will have the same birthday, whereas for a group of 57 people there is a 99% chance. These results may be a bit surprising, but the probability calculation is quite straightforward. We now plot the probability of having at least two people with the same birthday as a function of the number of people in the group. We will also learn some details about plotting. Note the various options that are used to make the plot look better. You may want to play with these options to understand their effects on the final plot.

```
plot(n[1:100],pn[1:100],type="p",pch=19,col="blue",lwd=3,
     xlab="Number of people in the room",ylab="P(at least 2 same-day B-days)",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

Figure 3.4 displays these probabilities up to the group size $n = 100$, because probabilities for $n > 100$ are extremely close to 1. We show how the probability can be estimated using simulations (also called the Monte Carlo method). Sup-

pose that we are interested in *estimating* the probability that two individuals have the same birthday in a group of n individuals. Here it is the corresponding R code:

```
set.seed(7654098)
n=23                    #number of people in the room
n_sim=10000             #number of Monte Carlo simulated rooms with n people
are_2_birthdays<-rep(NA,n_sim)  #vector of indicators for >=2 same B-day
for (i in 1:n_sim)
  {#begin simulations
  #calculate the number of unique birthdays in a group of n people
  n_birthdays<-length(unique(sample(1:365,n,replace = TRUE)))
  are_2_birthdays[i]<-(n_birthdays<n)
  }#end simulations
mean(are_2_birthdays)
```

```
[1] 0.4994
```

The Monte Carlo simulation result does not provide a perfect calculation of the probability, but it is extremely close. This could be further improved by increasing the number of simulations to $100,000$ or more. This provides an example of how powerful simulations can be; indeed, one could easily adapt the code to take into account unequal probabilities of birthdays for the different days of the year or more complicated decision rules. For example, suppose that we are interested in obtaining the probability that in a room with n people exactly two girls are born on the same day and one boy is born on the same day with the two girls but at least three boys are born on the same day, which could be different from the birthday of the two girls. Of course, this is a highly artificial example, but it provides an example of a question that could easily be answered by simulations and with great difficulty using explicit calculations.

We will unpack the simulation process a little bit to better understand how it works. We start by a little more closely investigating what the following R code does

```
x<-sample(1:365,n,replace = TRUE)
x
```

```
 [1]  85 157 360 155 167  85 274  46 291  32 339  13 263  85 113 319 221
[18] 105 100   1 105  24 325
```

So, the code produces numbers between 1 and 365 that assign birthdays to individuals. For example, the first person in this sample was assigned 85 and the second was assigned 157 indicating that they were born on `March 26` and `June 6`, respectively (here we deal with non-leap years only). These birthdays are assigned randomly and the same birthday can be assigned again. Indeed, in this case there are three individuals who were born on `March 26` (day 85) and two individuals who were born on `April 15` (day 105). This happens even though the probability of somebody being born on `March 26` and `April 15` is

the same for every other day of the year. This is ensured by the definition of the `sample` function. The reason that the same date can occur is that the sampling is with replacement, as indicated by `replace=TRUE`. So, this simple `R` line does the following:

(1) assigns a date of birth at random among all possible days of the year to the first person in the room where every birthday has probability 1/365;
(2) assigns a date of birth at random among all possible days of the year to the second individual in the room independently of the assignment of the first person's birthday;
(3) keeps on applying the same procedure for every person in the room. The following code:

```
unique(x)
```

```
 [1]  85 157 360 155 167 274  46 291  32 339  13 263 113 319 221 105 100
[18]   1  24 325
```

```
length(unique(x))
```

```
[1] 20
```

explains how the number of unique birthdays is calculated among the 23 birthdays in the sample. The function `unique(x)` is simply a vector transformation of `x` that keeps only records that are unique. Indeed, note that `unique(x)` contains 85 and 105 only once and the length of the vector is only 20 because there were 3 subjects with the same `March 26` and 2 subjects with the same `April 15` birthday. Therefore, among the 23 birthdays for this sample there are exactly 20 different samples. The `are_2_birthdays[i]` vector stores the True/False flag of whether there is more than one identical birthday in the sample. This is obtained by comparing `length(unique(x))=20` with the number of people in the group `n=23`. In this case `length(unique(x)) < n` indicating that "there are at least two individuals in the group with the same birthday." Thus, the flag will be true indicating that in this random group of 23 subjects there are at least two individuals with the same birthday. This is repeated `n_sim <- 10000` times indicating that the final vector `are_2_birthdays` has 10000 entries, where every entry indicates whether or not in one of the $10,000$ groups of 23 individuals there are individuals who have the same birthday. Thus, the `mean(are_2_birthdays)` will provide the proportion of times when groups of 23 individuals contain individuals with the same birthday. In this simulation the proportion was 0.4994 indicating that in 4994 groups there were individuals with the same birthday and in $5,006$ groups there were no individuals with the same birthday. This is close to the true 0.5073 probability of 23 birthdays in the sample.

Similarly, we will introduce two new functions that may have been used above. Instead of using `length` and `unique` on the sample of birthdays, we could have done:

```
any(duplicated(x))
```

```
[1] TRUE
```

where the `duplicated` function returns `TRUE` when it finds a duplicate in the data. Note, if you have a vector `c(1, 1, 3)`, only the second `1` will return `TRUE`. The `any` function takes in a logical statement and returns back a single logical if **any** elements are true. Thus, as with most things in programming in `R`, there are many ways to approach the problem.

In this and most examples, translating even human communication of scientific concepts into explicit and actionable biostatistical concepts needs to be done carefully and thoughtfully. Indeed, in our experience more than 90% of the problem is to formulate and understand it. Typically, this part of biostatistics is not taught or emphasized and the biostatistician is expected to "pick up" the skills. While we acknowledge that teaching communication and translation is difficult, we will emphasize throughout the book the need to understand problems at a deeper level. Indeed, we often encounter cases when researchers say they understand something without fully understanding it and they are interested in the tools that will get the job done. This works as long as the data analyst performs the same protocol over and over again on exactly the same data structure with the same problems. However, when something changes the biostatistician who can adapt and understand the problem at the deeper level will tend to be more appreciated and acknowledged.

3.4.3 Example: The Monty Hall problem

There are three doors with a prize behind one and worthless joke prizes (goats) behind the other two. The contestant selects a door. Then Monty Hall shows the contents of one of the remaining two (Monty Hall never opens the door to the actual prize.) After showing the joke prize behind the open door, the contestant is asked if he or she would like to switch. Should they? The problem was originally posed in a letter by Steve Selvin to the American Statistician in 1975 (Selvin 1975) and has puzzled many scientists. For example, according to Wikipedia, Paul Erdős was only convinced by the results of simulation.

Suppose that you choose door number 1; similar reasoning holds for any other door choice. Consider all three scenarios in terms of distribution of the prize and joke prizes (Goat).

	Scenario 1	Scenario 2	Scenario 3
Door 1	Prize	Goat	Goat
Door 2	Goat	Prize	Goat
Door 3	Goat	Goat	Prize
Switch	Lose	Win	Win

The answer is that it is actually better to switch the door because the probability of success is 1/3 if one does not switch the door and 2/3 if one switches the door. In Scenario 1 the prize is behind door 1. In this case Monty Hall would open either of the doors with a joke prize. Switching doors in this case is a losing strategy. Consider now the second scenario when the prize is behind door 2. Because you chose door 1 Monty Hall will have to open door 3, which has the joke prize. In this case switching doors would be a winning strategy. The exact same scenario happens if the prize is behind door 3. Thus, if the strategy is to choose 1 and not switch then the probability of winning is 1/3. If the strategy is to switch then the probability is 2/3, or double the probability of winning under the no-switch strategy.

We will now simulate the Monty Hall problem and we will provide Monte Carlo simulations of the game. This should illustrate two things: (1) that probabilities can be different depending on the strategy for opening the door; and (2) that Monte Carlo simulations are a powerful strategy to quickly answer complex questions in games of chance.

Estimate the winning probability with switch and no-switch strategies. Play the Monty Hall game 10,000 times with each strategy

```
n_games=10000                                #set the number of games
doors=1:3                                    #label doors
no_door_switching=rep(NA,n_games)            #use storage vectors for successes
door_switching=no_door_switching
for (i in 1:n_games)
    {#begin playing
    draw_prize=sample(doors,size=1)      #randomly assign 1 prize to a door
    draw_door=sample(doors,size=1)       #randomly choose 1 door
    no_door_switching[i]=(draw_prize==draw_door)       #win/loose for no switching
    if (draw_door==draw_prize)  {                     #if chosen door has prize
      show_door=sample(doors[-draw_prize],size=1)     #MH opens a no prize door
    }else {                                           #if chosen door has no prize
      show_door=doors[c(-draw_door,-draw_prize)]      #MH opens a no prize door
    }
    door_switch=doors[c(-draw_door,-show_door)]       #identify door to switch
    door_switching[i]=(draw_prize==door_switch)       #win/loose if switching
}#end playing
```

Compare the frequency of success of the two strategies

```
round(c(mean(door_switching),mean(no_door_switching)),digits=3)
```

```
[1] 0.657 0.343
```

This is the probability of winning when one plays the game many times. If the authors of this book had the choice, they would always switch. However, there is no *guarantee* of winning under either strategy, just a large increase in the probability of winning by using the switching strategy. This is a simple

game and we provided a way of playing it many times, very fast on a computer. The lesson here is that much more complicated games and strategies can be understood and analyzed using similar strategies. As long as the rules of the game are clear, simulations can be used to understand and quantify the role of probability. On a more somber note, studying the probability of dying or becoming unhealthy in a given period of time is not a game. However, there is a direct connection between the rules of probability and potential strategies to improve the odds of survival even when not everybody can be saved (win). Reasoning under uncertainty, especially in cases of life and death, is extremely hard, but crucial to advancing science. This is the reason why we take a close look at probability, understanding its rules, and eventually link it to data and experiments.

3.5 Sampling in R

Sampling and simulations are very important in data analysis. Understanding a model for data and simulating its real life characteristics are important for model building and refinement. Moreover, simulation is a powerful tool for computing quantities that are difficult to calculate mathematically and for characterizing the uncertainty of estimation procedures. Sampling, probability, and statistics go hand-in-hand and understanding sampling is one of the fundamental ways of understanding variability. Here we introduce some basic sampling ideas in `R` to facilitate this process.

3.5.1 Sampling without replacement

In this case after each draw the number that is drawn cannot be drawn again (e.g. lottery)

```
x1 <- sample (1:6)                    #a random permutation
x2 <- sample (1:6)                    #and another
x3 <- sample (1:6)                    #and another
x1
```

```
[1] 4 1 3 6 5 2
```

```
x2
```

```
[1] 6 4 3 2 5 1
```

```
x3
```

```
[1] 3 5 1 6 2 4
```

The sample with replacement without specifying how many samples to draw will produce a random permutation in `R`. This happens because the default

parameters are set to sample with replacement a number of times equal to the length of the vector and `replace=FALSE`. In general there are $n!$ permutations for a vector of length n, so for the vector `1:6` there are $6! = 6\times5\times4\times3\times2\times1 = 720$ distinct permutations. Random permutations are a type of resampling without replacement. They are extensively used in *random permutation testing*, for example, when one is interested in comparing a treatment to a control group. In this case a test statistic is constructed (e.g. the difference in mean of the two groups). However, to assign statistical significance to the observed discrepancy between groups, one would need to know what types of values are consistent with the observed data. A permutation of treatment labels approach could provide the null distribution of the test statistic (more about these concepts later). To see how this works let us consider the simple BMI/SEX example introduced earlier:

```
file.name = file.path("data", "bmi_age.txt")
data_bmi = read.table(file = file.name, header = TRUE,
                      stringsAsFactors = FALSE)
attach(data_bmi)
test_statistic<-mean(BMI[SEX==1])-mean(BMI[SEX==0])
test_statistic
```

```
[1] 5.6
```

The value of the test statistic in this dataset represents the difference in the average BMI of men and women. We conclude that there is a positive difference equal to 5.6 between the average BMI of men and women in this sample. However, if there is no association between BMI and SEX then we expect this difference to be larger than 0 in 50% of the samples. So, the fact that the difference is positive is not that surprising. The question is whether a value as large as 5.6 is a reasonable value if we believe there is no association between BMI and SEX. Let us have a second look at the BMI/SEX data

```
data_bmi[,2:3]
```

```
   BMI SEX
1   22   1
2   27   0
3   31   1
4   24   1
5   23   0
6   18   0
7   21   0
8   26   1
9   34   1
10  20   0
```

This representation of the data means that the pair of BMI/SEX information is assigned to a specific individual. Under the assumption that BMI is not

associated with SEX we could randomly permute the SEX-assignment labels among the individuals.

```
data_BMI_perm<-data_bmi
random_perm<-sample(1:length(SEX))
data_BMI_perm[,"SEX"]<-sample(data_BMI_perm[random_perm,"SEX"])
random_perm
```

```
 [1]  3  6  1  2  7  4  5  9 10  8
```

```
data_BMI_perm[random_perm,c("BMI", "SEX")]
```

```
   BMI SEX
3   31   1
6   18   1
1   22   0
2   27   1
7   21   0
4   24   0
5   23   0
9   34   0
10  20   1
8   26   1
```

Let us have a closer look at what the `random_perm` vector contains and what is its action on the data. The first entry of the vector is 3, which indicates that the SEX of the third person in the data will be assigned the SEX of the first person in the data. In this case, both are male (label=1) and the data line for subject 1 does not change. The second entry of the vector is 6, indicating that the SEX of the sixth person in the data will be assigned to the second person in the data. As both are female (label=0) the data line for subject 2 does not change. The fourth entry of the `random_perm` vector is 2, indicating that the SEX of the fourth person in the data (who is a male in the original dataset) will be assigned the SEX of the second person in the data (who is a female in the original dataset). This is a case when the gender assignment is switched. The number of men and women is the same, but the label assignment has changed for some of the people via the random permutation. Subjects 1 through 3 have not had their SEX assignment changed, but subject 4 was switched from man to woman and subject 6 was switched from woman to man. The SEX reassignment permutation is breaking any associations between SEX and BMI that may have existed in the original dataset. Thus, an acceptable value of the test statistic when we know that there is no association between SEX and BMI is:

```
ind_m_perm<-(data_BMI_perm[,"SEX"]==1)
mean(data_BMI_perm[ind_m_perm,"BMI"])-mean(data_BMI_perm[!ind_m_perm,"BMI"])
```

```
[1] -0.4
```

Of course, there are many such acceptable values when there is no association

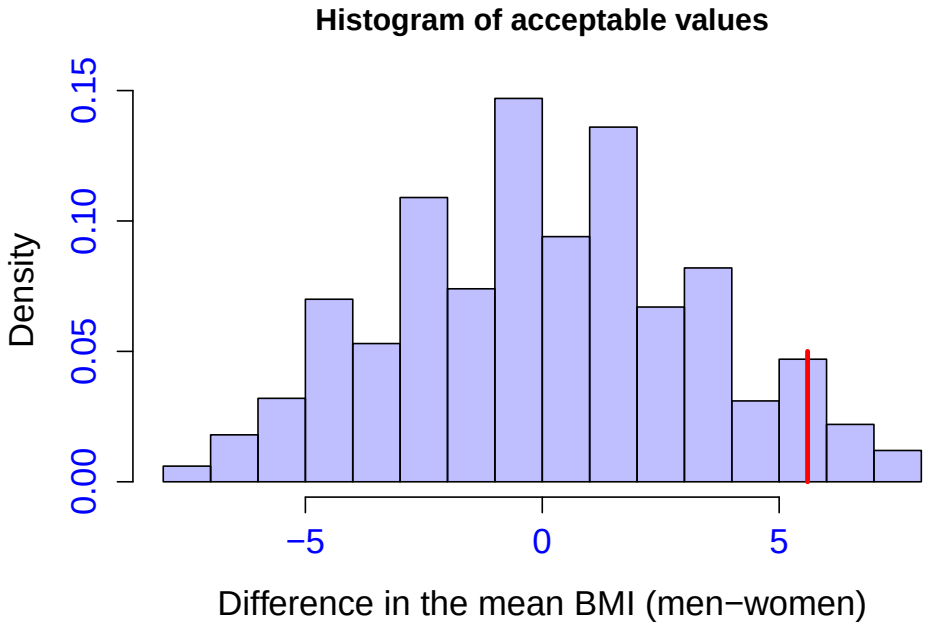


Figure 3.5: Distribution of differences in the mean between the mean BMI of men and women using 10000 permutations of the SEX variable. This breaks the association between BMI and SEX if any existed.

between SEX and BMI. In general, we cannot do an exhaustive search over all possible permutations, so a finite number of permutations is used instead. This is done below using the following R code:

```
n_perm=1000
accept_values=rep(NA,n_perm)
for (i in 1:n_perm)
{
data_BMI_perm<-data_bmi
random_perm<-sample(1:length(SEX))
data_BMI_perm[,"SEX"]<-data_BMI_perm[random_perm,"SEX"]
ind_m_perm<-(data_BMI_perm[,"SEX"]==1)
accept_values[i]<- mean(data_BMI_perm[ind_m_perm,"BMI"]) -
  mean(data_BMI_perm[!ind_m_perm,"BMI"])
}
hist(accept_values,probability=TRUE,col=rgb(0,0,1,1/4),breaks=20,
     xlab="Difference in the mean BMI (men-women)",
     main="Histogram of acceptable values",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(c(test_statistic,test_statistic),c(0,0.05),col="red",lwd=3)
```

The histogram in Figure 3.5 provides the distribution of acceptable values for the test statistic, which, in our case, is the difference in BMI between men and women. The red line indicates the observed value of this difference in the original dataset. One question could be just how extreme is the observed value of the BMI difference for this dataset. One way to calculate that is by obtaining the frequency of how many times the acceptable values (under the assumption of no association between BMI and SEX) are more extreme than the observed value of the test statistic. This is obtained simply as:

```
p_value=mean(accept_values>test_statistic)
p_value
```

```
[1] 0.045
```

This frequency is 4.5%, which indicates that the observed value of difference in BMI is quite unlikely under the assumption of no association between BMI and SEX in our dataset. This probability will be referred to as the p-value for the observed difference for the permutation test. The p-value will play a very important role in hypothesis testing. For now, we just use it to illustrate a principle.

3.5.2 Sampling with replacement.

In this case after each draw the number is placed back in the box and can be drawn again (e.g. lottery)

```
x1 <- sample (1:6,10,replace=TRUE)          #sampling with replacement
x2 <- sample (1:6,10,replace=TRUE)          #again
x3 <- sample (1:6,10,replace=TRUE)          #and again
x1
```

```
 [1] 6 1 3 2 2 1 6 3 4 4
```

```
x2
```

```
 [1] 1 5 2 1 1 3 2 5 2 6
```

```
x3
```

```
 [1] 2 3 3 2 1 2 3 4 1 3
```

Once the vector is sampled, various operations can be applied to it

```
sum (x1 == 3)                               #how many are equal to 3
```

```
[1] 2
```

```
max (x1)                                    #maximum of x1
```

```
[1] 6
```

```
median(x1)                                  #median of x1
```

```
[1] 3
```

The *nonparametric boostrap* is sampling with replacement of the vector with a number of draws equal to the length of the vector. Do not underestimate the importance of simple ideas. Statistics is the science of simplicity.

```
x <- sample ( 1:10, 10, replace=TRUE )      #nonparametric bootstrap
n=30                                        #for an arbitrary n
x <- sample ( 1:n, n, replace=TRUE )        #nonparametric bootstrap
```

For a vector of length n there are n^n nonparametric boostrap samples. If $n = 10$, there are 10^{10}, or 10 billion, possible bootstrap samples.

3.6 Random variables

A random variable is a numerical outcome of an experiment. The random variables that we study will come in two varieties, discrete and continuous. Discrete random variables are random variables that take on only a countable number of possibilities, whereas continuous random variables can take any value on the real line or some subset of the real line. Examples of random variables include:

- The 0/1 outcome of the flip of a coin
- The Dead/Alive outcome for a patient with lung cancer after five years
- The hypertension status of a subject randomly drawn from a population
- The outcome from the roll of a die
- The no/mild/severe outcome for a cognition impairment test
- The BMI of a subject four years after a baseline measurement
- The time to death of a person suffering from diabetes

The concept of a random variable can be defined more rigorously, but we believe that such a definition would not help either with the intuition or with the practice of biostatistics. Though basic and practical, the concept of a random variable remains hard to explain and understand. Consider the following experiment: sample, at random, from the US population $n = 300$ individuals and ask each one of them whether they support the death penalty. Each person can respond yes/no and the proportion of people who support the death penalty is the outcome of the experiment. Of course, if we wait until *after* the experiment is finished there is nothing random left; we simply have the data and a proportion of individuals among the 300 sampled who support the death penalty. However, *before* the experiment is started there is much uncertainty about what the result of the experiment will be. Because we do not know the result of the experiment before the experiment is run, we call the result of the experiment a *random* variable.

Now let us investigate what we really know after the data were collected on 300 individuals and let us assume that exactly 180 people have said they approve of the death penalty. This says that in this sample of 300 people the death penalty approval rate was 60%, which would signal that the approval rate in the population is quite high. But is it exactly 60% or could it be 58% or 61% or maybe 70%? One solution could be to talk to every person in the US, but this is practically impossible and a huge waste of resources. Indeed, by the time such a census would be finished it is likely that the overall proportion of death penalty supporters has already changed. Another solution could be to replicate the experiment many times, say 1000, calculate the proportion of supporters in every sample, and construct a histogram of acceptable values for the proportion of supporters of the death penalty. The problem is that the study will now cost 1000 times more and will require a completely different level of logistics. Indeed, one reason that most polling agencies use 500 to 2000 subjects per poll is that polling more people becomes extremely complicated and costly. What we would like is to have the cake and eat it, too. That is, we would like to conduct one poll, not spend the money on more data, and obtain a distribution of acceptable values for the proportion of people supporting the death penalty. The bootstrap is offering this solution. Indeed, a nonparametric bootstrap sample can do exactly that: build a large number of experiments with 300 subjects by sampling the subjects with replacement and calculating the proportion every time. An alternative solution, based on the central limit theorem (CLT) can also be used to calculate the *confidence interval*. This indirect way of thinking is very powerful in data analysis, but it is difficult to master because the concepts of uncertainty are not intuitive and are beyond simple mathematical logic. This is the reason why many students who have a very good mathematical background can have a hard time transitioning into data analysis. Indeed, mathematical concepts are often perfect and deductive, and proofs thereof are final. Data and real world are much, much more complicated and require a different way of thinking and a completely different platform for inference.

A random variable is the outcome of the experiment before the experiment is run, and the realization of the random variable is the observed outcome of the experiment. The difference between the two is the time needed to conduct the experiment. For example, in a prospective study of lung cancer survival the random variable is whether an individual will develop cancer within 10 *years of when monitoring of that individual had begun. To see the result of the experiment we would need to wait* 10 *years and record whether the person developed lung cancer or not. The reason why random variables are useful is that planning the experiment, developing the sampling scheme, and implementing the experiment are conducted* 10 *years before data are collected. Moreover, the study needs to be planned such that the results are generalizable to other identical experiments and to the target population (the population we sample from).*

3.7 Probability mass function

While before running the experiment we cannot know exactly what the outcome of the experiment will be, we typically know all possible outcomes. For example, for a particular person with lung cancer we do not know whether they will be dead or alive in five years. However, we know that they will be either dead (0) or alive (1). The distribution of such a variable is completely characterized if we know the probability of being dead, $p(0)$, or alive, $p(1) = 1 - p(0)$, after five years. For discrete variables there are only a finite or countable number of possible outcomes and the distribution is completely specified if the probability of every possible outcome of the experiment is known.

The probability mass function (pmf) of a random variable X is the function that assigns a probability that X takes a particular value. If $\mathcal{K}$ is the set of all possible outcomes for X, the pmf of X is $p(k) = P(X = k)$ for every $k \in \mathcal{K}$. To be a valid pmf, a function $p(\cdot)$ must satisfy

(1) $p(k) \geq 0$ for every $k \in \mathcal{K}$
(2) $\sum_{k \in \mathcal{K}} p(k) = 1$

The sum above is taken over all possible values of the experiment outcome and can be taken over a finite or infinite number of experimental outcomes. For example, for a Bernoulli random variable $\mathcal{K} = \{0, 1\}$ and the sum has two elements, $p(0) + p(1) = 1$, while for a Poisson random variable $\mathcal{K} = \{0, 1, \ldots, \}$ and the sum is infinite, $p(0) + p(1) + \ldots = 1$.

3.7.1 Example: Bernoulli random variables and sampling

A Bernoulli random variable is the outcome of an experiment with outcomes success, labeled 1, or failure, labeled 0. There are many such experiments, including flipping a coin and recording heads or tails, enrolling individuals with lung cancer in a study and recording who died and did not die after five years, or using a health intervention (e.g., intensive nutrition counseling) and recording whether or not it was successful (e.g., the child did not progress to diabetes).

Let the random variable X be the result of a coin flip where $X = 0$ represents tails and $X = 1$ represents heads. If the coin is fair, the pmf for X is given by $p(0) = P(X = 0) = 0.5$ and $p(1) = P(X = 1) = 0.5$. A more compact way of writing the pmf is

$$p(x) = 0.5^x 0.5^{1-x} \text{ for } x = 0, 1 \, .$$

We are using X to denote the random variable, which can be thought of as what is known about the realization of the experiment before the experiment is run, and x to denote the realization of the random variable, which can be thought of as what was obtained after the experiment was run. Suppose that the coin being flipped is the coin that nature uses to decide whether an individual with lung cancer will die in five years. This is not a fair coin in either a pure statistical

or human definition, but it is a conceptual coin that nature flips constantly. If θ is the probability of dying from lung cancer in five years then the pmf of this experiment is

$$p(x) = \theta^x(1-\theta)^{1-x} \text{ for } x = 0, 1 \ .$$

This formula is just a compact way of writing $p(0) = \theta^0(1-\theta)^{1-0} = 1-\theta$ and $p(1) = \theta^1(1-\theta)^{1-1} = \theta$, where we use some of the least fancy math possible. Of course, in this case surviving would be labeled as a "success" and "dying" as a failure, but these labels can be used interchangeably as long as one clearly states what is 1 and what is 0. While the coin flipping experiment is basic and quite scientifically uninteresting, one is extremely interested in understanding the probability of survival of lung cancer patients. From the point of view of a patient it is important to know whether he or she will survive or not for at least five years after lung cancer diagnosis. The doctor cannot predict whether he or she will survive, but can provide the probability of survival. For example, according to the American Cancer Society the five-year survival rate for people with stage 2A non-small cell lung cancer (NSCLC) is about 30%. These rates were obtained by observing survival time of individuals with this type of cancer. This duality between what is known before the experiment (survival probability for an individual who was just identified with stage 2A NSCLC) and what is known after the experiment (whether this particular individual died five years later) is the cornerstone of statistical thinking. Indeed, biostatistics is focused on reasoning under uncertainty and providing information before the experiment is run. In this case the patient does not need to wait five years to know the probability of survival, but will need to wait, and hope, that they will be alive five years later. Biostatistical thinking and inference does not have the elegance of strict mathematical thinking, because nature and life provide problems that are much harder to solve. Recall that *a random variable, X, can be thought of as what is known about the experiment before the experiment is run, and the outcome, x, of the experiment is the realization of the experiment after the experiment is run.*

We will now discuss simulation of Bernoulli random variables, which will facilitate getting used to the randomness associated with Bernoulli experiments using simulations in `R`. Basically, every random variable can be derived from simple Bernoulli draws (more about that later). Below we show how to: (1) draw 21 samples from independent Bernoulli random variables with probability of success of 0.5 (21 fair coin flips); and (2) draw 21 samples from independent Bernoulli random variables with probabilities of success: $0.00, 0.05, 0.10, 0.15, \ldots, 0.95, 1.00$, respectively.

```
x1<-rbinom(21,1,0.5)
x2<-rbinom(21,1,seq(0,1,length=21))
x1
```

```
 [1] 0 0 1 1 0 1 1 0 0 0 1 1 1 0 1 0 0 0 1 0 0
```

```
x2
```

```
 [1] 0 0 0 0 0 0 0 1 0 0 1 0 1 1 1 1 1 0 0 1 1
```

The two vectors are different and they reflect the different coin flipping mechanisms. The first vector is more chaotic in terms of changes between 0 and 1, whereas the second vector tends to have more zeros at the beginning and fewer zeros towards the end. This happens because the success probabilities for `x2` are much higher towards the end of the vector than towards the beginning. Of course, in practice we would not know anything about the probability of success or failure, we would only see the outcome of the experiment. The question is what could data teach us about what we should expect to see in the future. For this we need to run both experiments again, as follows:

```
x1<-rbinom(21,1,0.5)
x2<-rbinom(21,1,seq(0,1,length=21))
x1
```

```
 [1] 1 1 0 0 0 1 0 1 1 1 1 1 1 1 0 1 1 0 0 0 1
```

```
x2
```

```
 [1] 0 0 0 0 1 0 0 0 0 1 1 0 0 1 1 1 1 1 1 1 1
```

Let us compare the results of the `x1` and `x2` experiments. We can see that there is a lot more uncertainty in the first experiment than in the second. Indeed, based on the first experiment, `x1`, it would be harder to predict what the second experiment `x1` will be. Indeed, there seems to be more structure in the outcome of the experiment `x2`, as there are more zeros towards the beginning of the vector and fewer towards the end. We recommend conducting more simulations of this type and trying to understand what the outcomes mean.

3.7.2 Example: Poisson random variables and sampling

A Poisson random variable is the outcome of an experiment with measurements that are counts between 0 and ∞ and a specific pmf. There are many experiments that have counts as outcomes, including counting the number of patients arriving at a clinic on a given day, counting the number of waterborne pathogens in a water sample, and counting the number of reads in an RNA-seq analysis. In contrast with the 0/1 outcome case, there are many different distributions of counts. The Poisson distribution is the most popular distribution of counts and has the pmf:

$$P(X = x) = \frac{\lambda^x e^{-\lambda}}{x!} \text{ for } x = 0, 1, \ldots$$

Here $\lambda \in (0, \infty)$ is the average number of counts for the experiment. We will now learn how to simulate Poisson random variables. We start by simulating independently two periods of 15 days with an average number of patients per day $\lambda = 20$

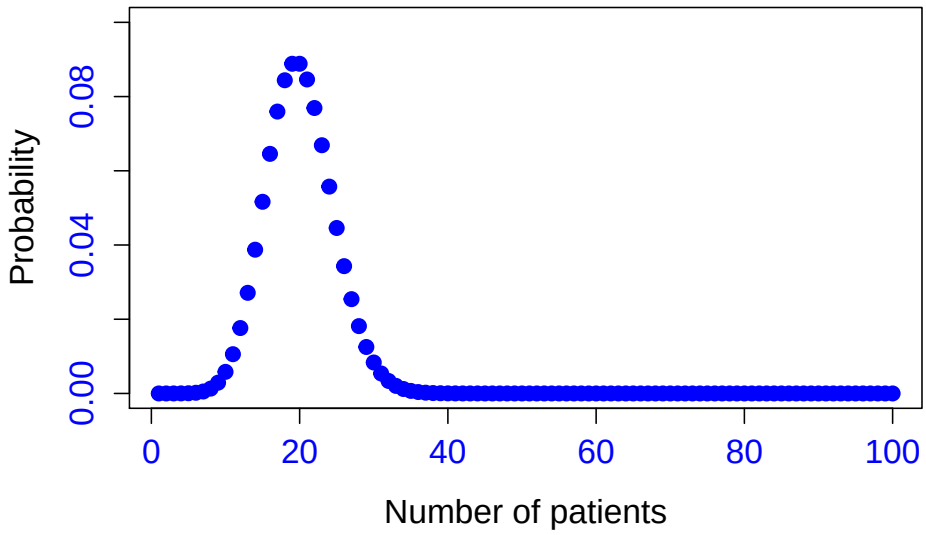


Figure 3.6: Probability mass function for a Poisson with mean 20.

```
rpois(15,20)
```

```
 [1] 18 24 17 15 14 20 11 17 27 17 14 19 17 21 26
```

```
rpois(15,20)
```

```
 [1] 20 17 22 17 18 16 17 19 19 21 20 15 26 19 24
```

These two realizations correspond to a reasonable scenario, when the number of patients varies by day, can be either larger or smaller than the average $\lambda = 20$, and can be any integer number between 0 and ∞. Even though the two 15-day intervals have a different pattern of number of patients, they do share something in common. Indeed, the number of patients hovers around the mean of the distribution. The mean of the Poisson distribution does not need to be an integer (e.g. $\lambda = 20.6$ is perfectly acceptable) but the realizations of the Poisson variable are always integers.

Consider a random variable, X, which has a Poisson(λ) distribution; this is often represented in biostatistics as $X \sim$ Poisson(λ). Figure 3.6 displays the pmf of a Poisson(λ) random variable for $\lambda = 20$

```
x=1:100
lambda=20
plot(x,dpois(x,lambda),type="p",pch=19,col="blue",lwd=3,
     xlab="Number of patients",ylab="Probability",cex.lab=1.3,
     cex.axis=1.3,col.axis="blue",ylim=c(0,0.1))
```

For every value taken by the random variable there is an assigned probability.

For example, if the mean number of patients per day is 20, the largest probability corresponds to 19 and 20 subjects, each with a probability of 0.089. The fact that the largest probability is assigned to 19 and 20 subjects is not surprising, but the pmf implies that only 17.8% of the days will have either 19 or 20 patients. This means there will be many other days with a different number of subjects. For example, the probability of seeing 18 or 21 subjects on a given day is 0.084, smaller than the probability of seeing either 19 or 20, but close. On the graph these probabilities are visualized as the two blue dots immediately beneath the two blue dots indicating the maximum of the pmf. Using the rules of probability, if a random variable, X, has a Poisson(20) distribution then

$$P(X \in \{18, 19, 20, 21\}) = P(X = 18) + P(X = 19) + P(X = 20) + P(X = 21) = 0.346$$

Suppose we would like to predict how many patients will show up tomorrow. We cannot predict exactly how many patients will show up, but we know that the average number of patients is 20. We know that 19 and 20 are the most likely numbers of patients, but the probability of either 19 or 20 patients is just 0.178. Moreover, a number of 18 or 21 subjects is less likely than 19 or 20, but not by much. The four possible outcomes 18, 19, 20, and 21 provide the most likely combination of four numbers of subjects that are likely to come to the clinic on a given day, but the probability of having any of 18, 19, 20, or 21 subjects is only 34.6%. It makes sense to continue this process and incorporate less likely outcomes. It is easy to obtain (use `R` to convince yourself) that

$$P(X \in \{12, 13, \ldots, 28\}) = 0.944$$

indicating that $\approx 95\%$ of the days will have a number of patients between 12 and 28. This is called a *prediction interval* and is based on what we know about the experiment before the experiment is run, that is, what we know today about the number of patients that will show up tomorrow. Today we know that the outcome is the number of patients, and the number of patients, X, is distributed $X \sim \text{Poisson}(20)$. Of course at the end of the day tomorrow we will have the outcome of the experiment, x, which is how many subjects did show up. Once the experiment is run and the number of subjects is recorded there is no uncertainty left about how many subjects have shown up. But uncertainty remains about the day after tomorrow and so on.

Philosophical discussions about the difference between the random variable, X, and the outcome of the random variable, x, can obscure the simple ideas we discuss here. We find such discussions relatively unproductive and impractical. For example, students often get stuck on the idea that, in fact, the outcome of a coin flip is the result of well-known laws of physics including the force applied to the coin, the location, spin angle, etc. This is true, and thoroughly unuseful as we do not know these values, there are millions of other things that could affect the outcome, and it is ultimately impractical to measure all possible effects. For these reasons, randomness is an exceptionally useful concept that allows us to make inference about the future or design other similar experiments. The

deterministic view of the world, especially in the biological, social and medical sciences, is impractical and needs to be replaced with the stochastic view. The sooner the student accepts this the faster he or she will understand the deeper concepts of biostatistics.

Of course, in practice we do not know that the number of patients on a given day follows a Poisson(20) distribution, or any other distribution. We only see the data, the number of patients over many days. Thus, it makes sense to use the data and try to infer the underlying mechanism (distribution) that generates the number of patients per day. To see how this works, we will simulate 1000 days (approximately 3 years of data) from a Poisson(20) and then simply plot the frequency of seeing a particular number of patients. For example, for 20 patients we will calculate how many days had exactly 20 patients and we will divide the number by 1000. This will reconstruct the pmf from observed data

```
y<-rpois(1000,20)              #simulate 1000 independent Poisson(20)
py=rep(0,100)                  #storage vector of Poisson probabilities
for (i in 1:length(py))        #for every possible outcome between 1 and 100
    {py[i]<-mean(y==i)}        #calculate the frequency of observing i subjects
```

The `table` function works quite well here also to compute a frequency table of the elements of `y`:

```
tab = table(y)
tab
```

```
y
  8   9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25
  2   1   3  11  20  37  45  50  55 101  96  81 102  83  66  66  64  34
 26  27  28  29  30  31  32  33  34  36  37
 19  21  19   8   4   4   2   2   2   1   1
```

We see the frequency counts for the observed values of `y`. Now, we can get the probability by dividing by `1000` or simply using `prop.table`:

```
prop.table(tab)
```

```
y
    8     9    10    11    12    13    14    15    16    17    18    19
0.002 0.001 0.003 0.011 0.020 0.037 0.045 0.050 0.055 0.101 0.096 0.081
   20    21    22    23    24    25    26    27    28    29    30    31
0.102 0.083 0.066 0.066 0.064 0.034 0.019 0.021 0.019 0.008 0.004 0.004
   32    33    34    36    37
0.002 0.002 0.002 0.001 0.001
```

Figure 3.7 displays the theoretical and empirical (reconstructed from observations) Poisson distributions

```
x=1:100                        #Set the x-axis, where the Poisson pdf is evaluated
lambda=20                      #Set the Poisson success rate
```

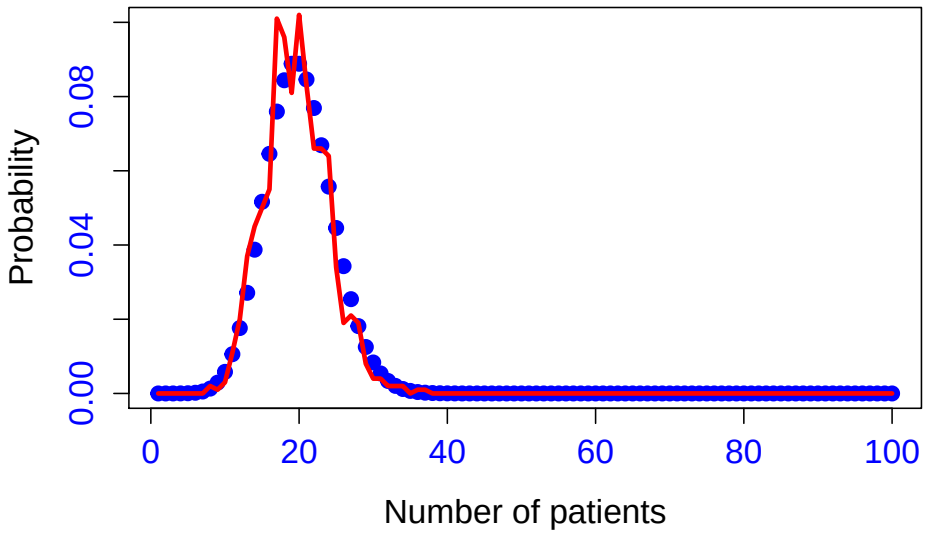


Figure 3.7: Probability mass function for a Poisson with mean 20 together with the frequency of observing a number of subjects per day out of 1000 days.

```
plot(x,dpois(x,lambda),type="p",pch=19,col="blue",lwd=3,
     xlab="Number of patients",ylab="Probability",cex.lab=1.3,
     cex.axis=1.3,col.axis="blue",ylim=c(0,0.1))
lines(1:100,py,col="red",lwd=3)  #here, lwd controls the thickness of the line
```

There is a good agreement between the theoretical (blue dots) and empirical (red line) distributions, though there are some discrepancies. Indeed, the theoretical frequencies of 18, 19, and 20 are 8.5%, 8.8% and 8.8%, respectively, whereas the observed frequencies were 9.4%, 8.1% and 10.1%, respectively. Such differences are due to sampling variability and they occur even when we know precisely what the underlying theoretical mechanism is. In practice, what we would see is just the red line describing the empirical distribution. One of the targets of inference is to try to guess (or estimate, or guesstimate) what could have been the true underlying theoretical mechanism that generated the data. This is an example of an inverse problem when we know the data and try to infer the data generating mechanisms. This is one of the most prevalent practical problems in science.

3.8 Probability density function

So far, we have discussed discrete variables (Bernoulli, Poisson) that have a finite or countable number of possible outcomes. However, there are many experiments that have a continuous (uncountable number) of outcomes. For ex-

ample, for a particular person with lung cancer, who is alive, the survival time is a continuous variable. While survival time could be discretized into a countable number of outcomes by, for example, counting the number of minutes between today and the person's death, this discretization is not practical. Instead, we use continuous variables and introduce the probability density function (pdf).

A probability density function (pdf) is a function, say $f(\cdot)$, associated with a continuous random variable, X, such that *areas under pdfs correspond to probabilities for that random variable*

$$P(X \in I) = \int_I f(x)dx \quad \text{for every interval } I$$

To be a valid pdf, the function $f(\cdot)$ must satisfy

(1) $f(x) \geq 0$ for all x
(2) $\int_{-\infty}^{\infty} f(x)dx = 1$

There are many similarities between pmfs and pdfs, but an important difference is that continuous variables assign zero probability to any particular outcome, $P(X = x) = 0$ for any x, and can only assign positive probability to intervals. For example, the probability that a person will survive a lung cancer diagnosis for *exactly* five years is zero, but the probability that he or she will survive five years plus or minus one minute is small, but positive. This makes sense, as pinpointing the exact time of death with below millisecond accuracy is both impractical and unuseful. A continuous variable can be discretized in many different ways. For example, if X is the survival time we can easily dichotomize it using the transformation $Y = I(X > 5)$, which is a Bernoulli random variable indicating whether or not the person will survive for more than five years. The probability of surviving for more than five years can then be written as

$$P(Y = 1) = P(X > 5) = \int_5^{\infty} f(x)dx \; .$$

Other finer discretizations can be obtained by separating the survival time into smaller intervals, say $I_0 = [0, 1)$, $I_1 = [1, 2)$, $I_2 = [2, 3)$, $I_3 = [3, 4)$, $I_4 = [4, 5)$, $I_5 = [5, \infty)$. Then we can define the random variable $Y = \sum_{k=0}^{5} kI(X \in I_k)$, which assigns the outcome $Y = k$ if and only if $X \in I_k$. Here $I(X \in I_k)$ denotes the indicator that the survival time will be in the interval I_k. The discrete random variable Y takes values $0, 1, \ldots, 5$ with the pmf

$$p(k) = P(Y = k) = P(X \in I_k) = \int_{I_k} f(x)dx \; .$$

Of course, one can make these approximations finer and finer until there is no practical difference between $f(x)$ evaluated in I_k and $p(k)/|I_k|$. Here $|I_k|$ denotes the length of the interval I_k. Thus, there is a strong connection between the pmf and pdf, where the pdf can be interpreted as the probability of being in a small interval divided by the length of the interval.

3.8.1 Example: Exponential random variables and sampling

Assume that the time in years from diagnosis until death of persons with lung cancer follows a density like

$$f(x) = \begin{cases} \frac{1}{5}e^{-x/5} & \text{for} \quad x > 0\ ; \\ 0 & \text{for} \quad x \leq 0\ . \end{cases}$$

First, we want to check that this is a valid pdf. Obviously, $f(x) \geq 0$ for all x because e raised to any power is strictly positive. Moreover,

$$\int_{-\infty}^{\infty} f(x)dx = \int_{-\infty}^{0} 0dx + \int_{0}^{\infty} \frac{1}{5}e^{-x/5}dx = -e^{-x/5}\Big|_{0}^{\infty} = 1\ .$$

If X denotes the survival time with the pdf described above then we say that X has an exponential distribution with mean survival time $\theta = 5$ years and denote $X \sim \exp(5)$. The pdf allows us to compute various quantities of interest. Suppose that we would like to know what is the probability that a randomly selected person with a exp(5) distribution survives for more than six years? This can be obtained as

$$P(X \geq 6) = \int_{6}^{\infty} \frac{1}{5}e^{-x/5}dx = -e^{-x/5}\Big|_{6}^{\infty} = e^{-6/5} \approx 0.301\ .$$

This can also be obtained directly in R without conducting the integral calculations.

```
surv_prob=pexp(6, 1/5, lower.tail = FALSE)
round(surv_prob,digits=3)
```

```
[1] 0.301
```

We would like to visualize this probability as the area under the pdf and also learn how to add arrows and text to a plot. The R code below and Figure 3.8 are designed to do that.

```
exp_pdf_eval<-seq(0,20,by=0.1)
exp_pdf<-dexp(exp_pdf_eval,1/5)

#plot the exponential pdf as a blue thicker line (lwd=3)
plot(exp_pdf_eval,exp_pdf,type="l",col="blue",lwd=3,
     xlab="Survival time (years)",ylab="Density",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")

#build the shaded area under the curve
#the shaded area is a polygon (note how vertexes are constructed)
surv_pdf_eval<-c(seq(6,20,by=0.1),seq(20,6,by=-0.1))
```

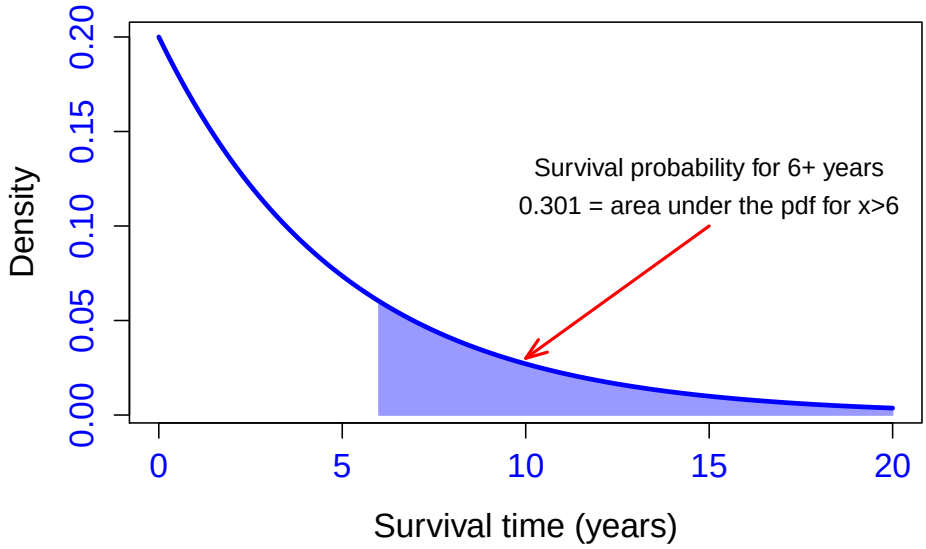


Figure 3.8: Probability density function (pdf) for an exponential random variable with mean 5 years and the interpretation of probability as area under the pdf.

```
poly_surv<-c(rep(0,length(seq(6,20,by=0.1))),dexp(seq(20,6,by=-0.1),1/5))

#plot the shaded area, note that it is transparent to give it a better look
polygon(surv_pdf_eval, poly_surv, col=rgb(0, 0, 1, 0.4),border=rgb(0, 0, 1, 0.4))

#add arrows and text
arrows(15,0.1,10,0.03,col="red",lwd=2,length = 0.15, angle = 20)
text(15,0.13, "Survival probability for 6+ years")
text(15,0.11,"0.301 = area under the pdf for x>6")
```

Figure 3.8 illustrates the general idea about the interpretation of probability given a pdf. The shaded area is equal to the area under the curve and is equal to $P(X > 6) = P(X \geq 6) = \int_6^\infty \frac{1}{5} e^{-x/5} dx$ (x-axis values over 20 are omitted for presentation purposes). The pdf completely characterizes the distribution of the random variable X. However, just as in the case of discrete random variables, it cannot be used to predict exactly what the outcome of a particular experiment will be.

The same exact principles can be applied to answer other questions before the experiment is run. For example, we would like to calculate the probability that a randomly selected person with an exp(5) distribution survives strictly more than five and strictly less than six years. This can be obtained using direct

integration

$$P(5 < X < 6) = \int_5^6 \frac{1}{5} e^{-x/5} dx = -e^{-x/5}\Big|_5^6 = e^{-1} - e^{-6/5} \approx 0.067$$

indicating that approximately 6.7% of individuals from this population will have a survival time between 5 and 6 years. The same calculation can be done directly in R avoiding manual integration as follows

```
surv_prob=pexp(6, 1/5)-pexp(5, 1/5)
round(surv_prob,digits=3)
```

```
[1] 0.067
```

As an exercise, plot the area under the curve corresponding to this probability using the tools shown earlier. As survival probability is a continuous variable, it assigns no probability to individual points and $P(5 < X < 6) = P(5 \leq X \leq 6) = \int_5^6 0.2e^{-x/5} dx$.

3.8.2 Example: Double exponential random variables

Consider the case when one repeatedly measures the systolic blood pressure (SBP) of a person and for every measurement we calculate the *difference* (or residual) between the measurement and the long-term average blood pressure of the individual. Of course, some of these differences are positive and some are negative and it may be reasonable to assume that errors are symmetric around zero. A potential distribution that could characterize these differences is the double exponential distribution with the pdf

$$f(x) = 0.5e^{-|x|} \quad \text{for } x \in \mathbb{R}\,,$$

where $|x|$ denotes the absolute value of x, that is $|x| = x$ if $x \geq 0$ and $|x| = -x$ if $x < 0$. Let us show that this is a valid density. Obviously, $f(x) > 0$ for every $x \in \mathbb{R}$ because e raised to any power is strictly positive. Moreover

$$\int_{-\infty}^{\infty} 0.5e^{-|x|} dx = \int_{-\infty}^{0} 0.5e^{x} dx + \int_{0}^{\infty} 0.5e^{-x} dx = 2\int_{0}^{\infty} 0.5e^{-x} dx = -e^{-x}\Big|_0^{\infty} = 1\,.$$

The equality $\int_{-\infty}^{0} 0.5e^{x} dx = \int_{0}^{\infty} 0.5e^{-x} dx$ is due to the symmetry of the pdf around zero and can be shown doing the change of variable $y = -x$. The following plot provides the visual representation of the standard double exponential distribution

```
x=seq(-5,5,length=101)                    #set the grid where the evaluate the pdf
fx=exp(-abs(x))/2                         #calculate the pdf on the grid
plot(x,fx,type="l",col="blue",lwd=3,xlab="error",ylab="PDF",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

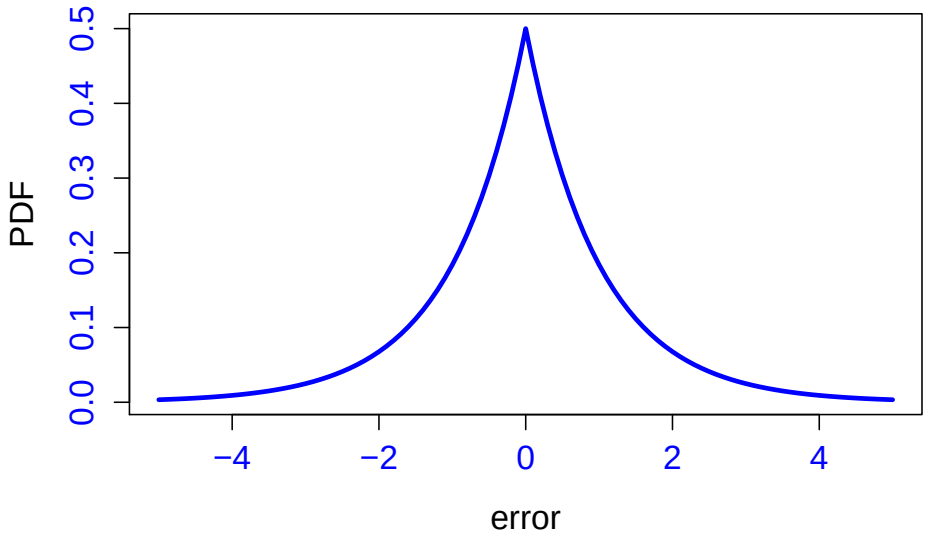


Figure 3.9: Probability density function of the standard double exponential distribution.

Figure 3.9 displyas the pdf of the standard double exponential distribution indicating that it is symmetric around zero and has a singularity at zero (the pdf is not differentiable at 0). This distribution is sometimes used in practice as an alternative to the Normal distribution to represent the distribution of errors and is referred to as the Laplace distribution. It is also the implicit choice of distribution of residuals in least absolute deviations regression (Portnoy and Koenker 1997) and as the LASSO prior (Tibshirani 1996). A more general version of the double exponential distribution depends on two parameters, μ and σ, and has the form

$$f(x) = \frac{1}{2\sigma} \exp\left\{-\frac{|x-\mu|}{\sigma}\right\} \text{ for } x \in \mathbb{R}.$$

It is easy to show that this is a valid density. To better understand the effect of the parameters on the distribution we show the changes in the pdf as a function of μ and σ. We first analyze changes in μ by keeping the scale parameter σ fixed. Figure 3.10 the pdfs corresponding to $\mu = 1, 2, 5$ and $\sigma = 1$.

```
library(VGAM)
mu=c(0,1,5)
sigma=1
x=seq(-7,12,length=1001)
y=matrix(rep(NA,length(mu)*length(x)),ncol=length(x))
for (i in 1:3)
  y[i,]<-dlaplace(x, location = mu[i], scale = sigma)

plot(x,y[1,], type="l",col="blue", lwd=3,
```

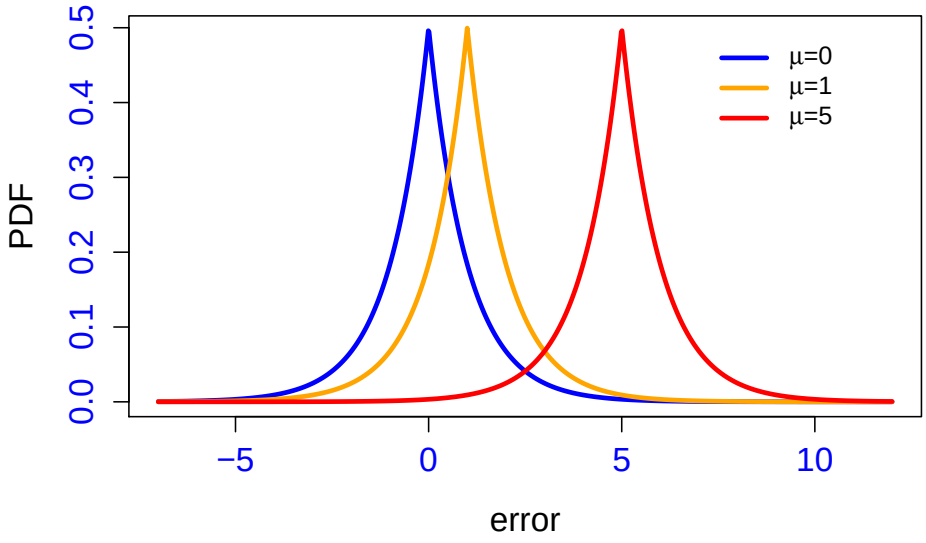


Figure 3.10: Probability density function of the double exponential distribution with different means (location parameters.)

```
     xlab="error", ylab="PDF",
     cex.lab=1.3, cex.axis=1.3 ,col.axis="blue")
lines(x,y[2,],type="l",col="orange",lwd=3)
lines(x,y[3,],type="l",col="red",lwd=3)
legend(7,0.5,
       c(expression(paste(mu,"=0")),expression(paste(mu,"=1")),
         expression(paste(mu,"=5"))),lwd=c(3,3,3),
       col=c("blue","orange","red"),bty = "n")
```

Figure 3.10 indicates that the double exponential pdfs are symmetric around their mean, μ, and that only the location of the distribution changes when μ changes. For this reason the parameter μ is often referred to as the location parameter. We now analyze changes in σ by keeping the mean parameter μ fixed. Figure 3.11 displays the pdfs corresponding to $\sigma = 1, 2, 5$ and $\mu = 0$.

```
mu=0
sigma=c(1,2,4)
x=seq(-7,7,length=1001)
y=matrix(rep(NA,length(sigma)*length(x)),ncol=length(x))
for (i in 1:3)
{y[i,]<-dlaplace(x, location = mu, scale = sigma[i])}
plot(x,y[1,],type="l",col="blue",lwd=3,xlab="error",ylab="PDF",cex.lab=1.3,cex.axis=1.3
lines(x,y[2,],type="l",col="orange",lwd=3)
lines(x,y[3,],type="l",col="red",lwd=3)
legend(3,0.5,c(expression(paste(sigma,"=1")),expression(paste(sigma,"=2")),
```

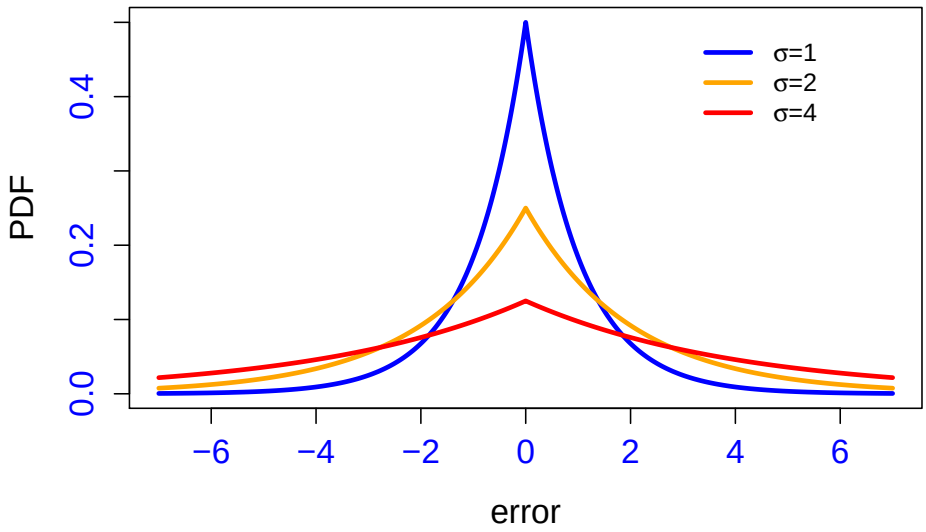


Figure 3.11: Probability density function of the double exponential distribution with different sigmas (scale parameters.)

```
            expression(paste(sigma,"=4"))),lwd=c(3,3,3),
    col=c("blue","orange","red"),bty = "n")
```

This time the pdfs are all centered at $\mu = 0$, but as σ increases there is less probability around zero and more probability away from zero (in the tails of the distribution). The implication is that a larger σ corresponds to larger errors of measurement. For this reason the parameter σ is called the scale parameter. What we have seen for the double exponential distribution is a common characteristic of many other distributions. Starting with a standard distribution (in this case the standard double exponential) and introducing location and scale parameters is a general technique for building flexible families of distributions. Such families are called, quite boringly, *location-scale families of distributions*. The Normal, or Gaussian, distribution is the best known location-scale family of distributions and it will play an essential role in this book.

We now investigate the agreement between the theoretical distribution and the empirical distribution of a double exponential distribution. Below we construct the empirical pdf of 100 simulated observations from a double exponential distribution with mean 2 and scale parameter 2.

```
#mean parameter
mu=2
#scale parameter
sigma=2
#generate 100 independent samples
```

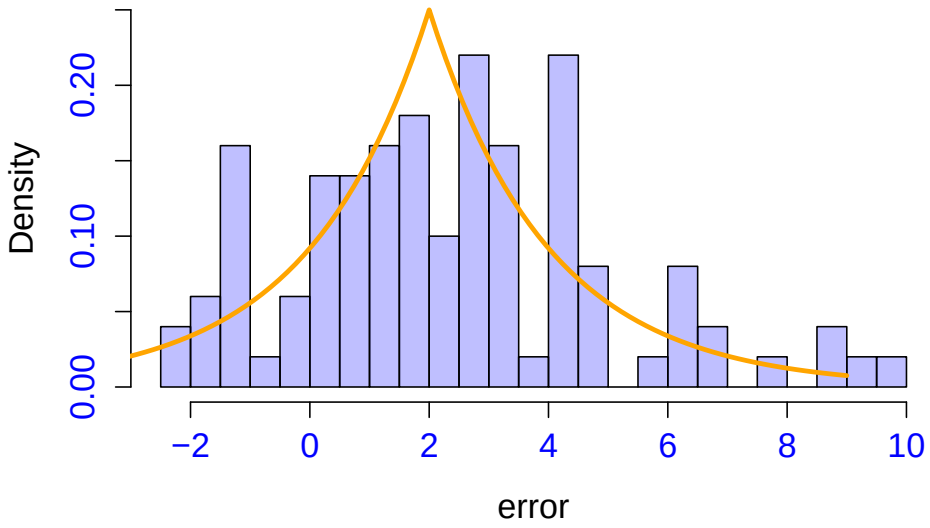


Figure 3.12: Theoretical standard double exponential distribution versus the histogram of a sample of size 100 from the same distribution.

```
y=rlaplace(100,mu,sigma)

#construct the empirical pdf of y
#plot the histogram of observations (empirical pdf)
hist(y,breaks=20,probability=TRUE,col=rgb(0,0,1,1/4),
     xlab="error",main="",cex.lab=1.3,cex.axis=1.3,
     col.axis="blue",ylim=c(0,.25))

#construct the theoretical pdf
#grid around the true mean
x=seq(mu-7,mu+7,length=101)
#true pdf of the double exponential
fx=dlaplace(x, location = mu, scale = sigma)
#plot true pdf
lines(x,fx,col="orange",lwd=3)
```

Figure 3.12 indicates that the histogram of the 100 random samples resembles the true pdf, but there are also discrepancies due to random sampling variation. This is not surprising, but it provides a good lesson: even when we know precisely the sampling mechanisms, there are differences between the observed and theoretical distributions. The size of these differences depends on the number of samples collected (amount of information). Try, for example, to redo the plot with 1000 or 10000 samples.

3.9 Cumulative distribution function

The cumulative distribution function (cdf) of a random variable X is defined as the function

$$F(x) = P(X \leq x) \ .$$

The survival function of a random variable X is

$$S(x) = P(X > x) = 1 - F(x) \ ,$$

indicating that the survival function is 1 minus the cdf of the random variable. Note the inequality in the cdf is less than or equal to, which is relevant in discrete data because $P(X = x) \geq 0$, but not in continuous variables as $P(X = x) = 0$. Thus, be careful with these definitions when calculating a cdf or survival function. For continuous random variables the pdf, $f(x)$, is the derivative of the cdf, $F(x)$, that is $f(x) = F'(x)$ for all x. This provides an excellent tool to work with the pdf and cdf interchangeably, as $F(x) = \int_{-\infty}^{x} f(u)du$.

3.9.1 Example: The exponential cdf

Consider again the example of survival time for a patient with lung cancer, X, and assume that $X \sim \exp(5)$. We would like to calculate the cdf and survival function for X. The survival function is

$$S(x) = \int_x^\infty \frac{1}{5} e^{-t/5} dt = -e^{-t/5} \Big|_x^\infty = e^{-x/5} \ ,$$

and the cdf is $F(x) = 1 - e^{-x/5}$. Using the chain rule for differentiation we obtain

$$\frac{\partial}{\partial x} F(x) = -\frac{\partial}{\partial x} e^{-x/5} = -\left\{ \frac{\partial}{\partial x} \left(-\frac{x}{5} \right) \right\} e^{-x/5} = \frac{1}{5} e^{-x/5} = f(x) \ ,$$

which indicates that the pdf, $f(x)$, can be recovered from the derivative of the cdf, $F(x)$.

3.9.2 Example: The double exponential cdf

While there is a one-to-one relationship between the pdf and cdf of a distribution, the plots corresponding to the two functions have different appearance and interpretation. Consider the same example of a double exponential distribution and plot the cdfs (instead of the pdfs) corresponding to $\mu = 1, 2, 5$ and $\sigma = 1$.

The cdfs in Figure 3.13 increase over the entire range and are bounded between 0 and 1, with both 0 and 1 being acceptable values. The steepness of the curve does not change with μ, while the location of probability accumulation is shifted

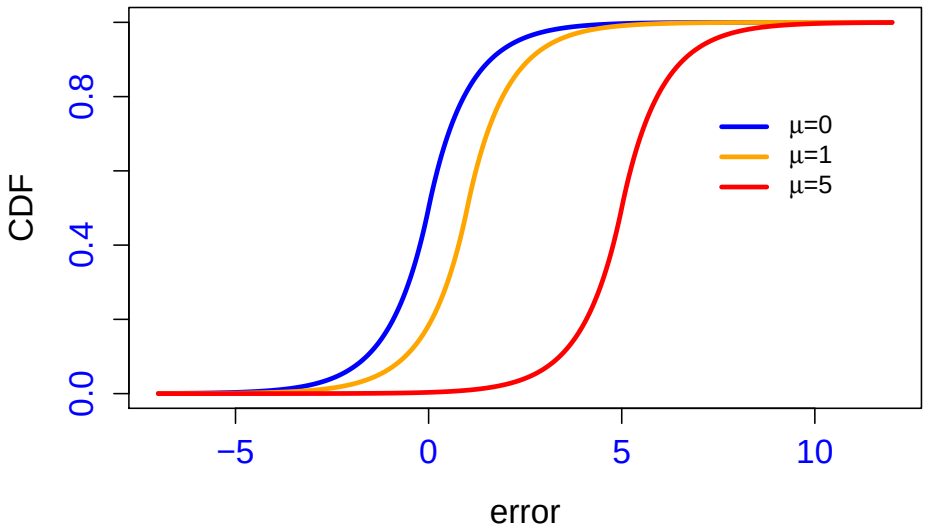


Figure 3.13: Cumulative distribution function of the double exponential distribution with different means (location parameters.)

(to the right for increased μ). These cdfs are characteristics of a location change in a distribution, with the cdfs translated to the right for increased μ. Let us investigate what happens with the cdfs when $\sigma = 1, 2, 5$ and $\mu = 0$.

Figure 3.14 displays the cdfs for the double exponential distribution with mean 0 and $\sigma = 1, 2, 5$. In this case the cdf curves are not simply translated and, indeed, they cross-over at $\mu = 0$ (why?). Distributions that are more dispersed (higher peak, thicker tails of the pdf) tend to have shallower slopes of the cdf. This indicates slower accumulation of probability (compare, for example, the cdfs corresponding to $\sigma = 1$ and $\sigma = 4$).

3.10 Quantiles

For a probability value $\alpha \in [0, 1]$ the α^{th} quantile of the distribution with cdf $F(\cdot)$ is the point x_α such that

$$F(x_\alpha) = \alpha \, .$$

The α^{th} percentile is simply the α^{th} quantile with α expressed as a percent instead of probability. The median is the 0.5 quantile (or 50^{th} percentile). You may also hear about tertiles (quantiles corresponding to $\alpha = 1/3$ and $\alpha = 2/3$), quartiles (quantiles corresponding to $\alpha = 1/4, 2/4, 3/4$), quintiles (quantiles corresponding to $\alpha = 1/5, 2/5, 3/5, 4/5$), and deciles (quantiles corresponding to $\alpha = 1/10, 2/10, \ldots, 9/10$).

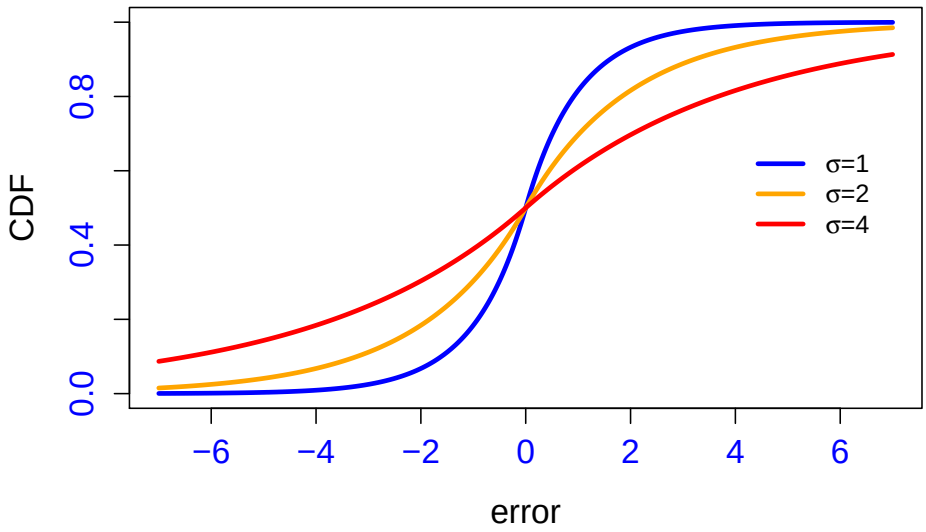


Figure 3.14: Cumulative distribution function of the double exponential distribution with different sigmas (scale parameters.)

Let us calculate the 0.25 quantile (a.k.a. first quartile) of an exponential distribution with mean survival time of five years. This can be obtained by solving for x

$$.25 = F(x_{.25}) = 1 - e^{-x_{.25}/5} \implies x_{.25} = -\log(.75) \approx 1.44 \, .$$

Here we have just used the direct definition of the quantile and solved the equation $\alpha = F(x_\alpha)$. In general, $x_\alpha = F^{-1}(\alpha)$, where $F^{-1}(\cdot)$ represents the functional inverse of the cdf. As an interesting aside, the inverse of cdf functions are often used as link functions for probability distributions. For example, the inverse cdf of a Normal distribution provides the probit link and the inverse of the logistic function provides the logit link. Quantiles of the distribution can also be obtained directly from R (no need for books with tables of distributions).

```
surv_time=qexp(.25, 1/5)
round(surv_time,digits=3)
```

```
[1] 1.438
```

This can be interpreted as 25% of a population with an exponential survival distribution of five years will die during the first 1.44 years. We can calculate a much wider range of quantiles as follows.

```
tq<-qexp(seq(0.01,0.99,length=100), 1/5)
round(tq,digits=2)
```

```
  [1]  0.05  0.10  0.15  0.20  0.25  0.31  0.36  0.41  0.47  0.52  0.58
 [12]  0.63  0.69  0.75  0.80  0.86  0.92  0.98  1.04  1.10  1.17  1.23
```

```
 [23]  1.29  1.36  1.42  1.49  1.56  1.62  1.69  1.76  1.83  1.91  1.98
 [34]  2.05  2.13  2.20  2.28  2.36  2.44  2.52  2.60  2.69  2.77  2.86
 [45]  2.95  3.04  3.13  3.22  3.32  3.42  3.52  3.62  3.72  3.82  3.93
 [56]  4.04  4.15  4.27  4.39  4.51  4.63  4.76  4.89  5.02  5.16  5.30
 [67]  5.44  5.59  5.75  5.91  6.07  6.24  6.41  6.60  6.78  6.98  7.18
 [78]  7.40  7.62  7.85  8.10  8.35  8.62  8.91  9.21  9.53  9.88 10.25
 [89] 10.65 11.08 11.56 12.08 12.67 13.34 14.11 15.02 16.13 17.57 19.59
[100] 23.03
```

3.10.1 Quantile-Quantile plots or QQ-plots

QQ-plots provide a fast way of comparing two distributions. Indeed, two equal distributions should have identical quantiles. One thing that can be done is to compare the theoretical and empirical quantiles from three different simulations. We simply generate three independent exponential random samples of size 30 and then calculate their empirical quantiles

```
x1<-rexp(30, 1/5)
x2<-rexp(30, 1/5)
x3<-rexp(30, 1/5)
eq1<-quantile(x1,seq(0.01,0.99,length=100))
eq2<-quantile(x2,seq(0.01,0.99,length=100))
eq3<-quantile(x3,seq(0.01,0.99,length=100))
```

The theoretical quantiles for the exp(5) distribution are stored in the vector `tq` and the empirical quantiles are stored in the vectors `eq1`, `eq2`, `eq3`. Figure 3.15 display these QQ-plots

Even though data were generated from the theoretical distribution, we can see sizeable differences between the theoretical and empirical distributions, especially in the upper quantiles (larger quantiles corresponding to higher probabilities). Moreover, the empirical distributions do not agree with each other. For example, in the first sample (blue dots) the first 20 subjects died in 6.3 years, whereas in sample 2 (orange dots) the first 20 subjects died in 4.14 years. In contrast, the theoretical distribution predicts that the first 2/3 of the population would die in 5.49 years. Such discrepancies should not be surprising as they are a manifestation of sampling variability and characterize what we would see in practice if three independent studies collected data on 30 subjects each. The data will not be the same, though they will share certain characteristics. Biostatistics provides a quantitative way of characterizing what differences are expected and what differences may be due to other causes. Variability is especially pronounced for the higher quantiles, because there is less certainty when estimating smaller probabilities. In this case there is no uncertainty for small quantiles because all data are bounded at zero. However, large variability should be expected in the lower tail of the distribution when the support of the distribution is unbounded.

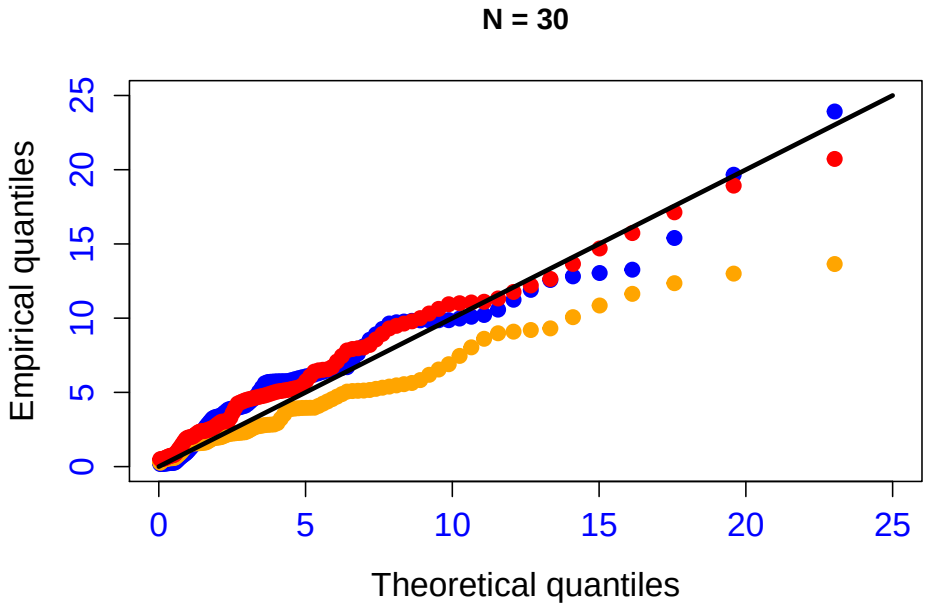


Figure 3.15: Quantile-Quantile plots for three samples from an exponential with mean 5 versus the quantiles of three samples of size 30 from the same distribution.

Now, let us investigate what happens when $n = 100$

and $n = 500$

Figures 3.16 and 3.17 indicate that the amount of uncertainty in the empirical quantiles decreases and they are closer and closer to the theoretical ones. It is important to remember that: (1) QQ-plots exhibit increased variability in the tails; (2) variability decreases with increased sample size; and (3) inspecting a QQ-plot provides a qualitative assessment of agreement between two distributions.

As a take-home message, we introduced four concepts that will be important throughout the book: pdf (pmf), cdf (integrated pdf function), quantiles (inverse cdf function), and random sampling from a distribution. In `R` for every one of these concepts there is a corresponding function. For example, for the exponential distribution the pdf function is `dexp`, the cdf function is `pexp`, the quantile function is `qexp`, and the random sampling function is `rexp`. The same convention applies to other distributions. For example, for the Normal (Gaussian) distribution the corresponding functions are `dnorm`, `pnorm`, `qnorm`, and `rnorm`. Thus, having a computer and `R` removes the need for books of statistical tables, but does not remove the need to understand what it is that we are computing.

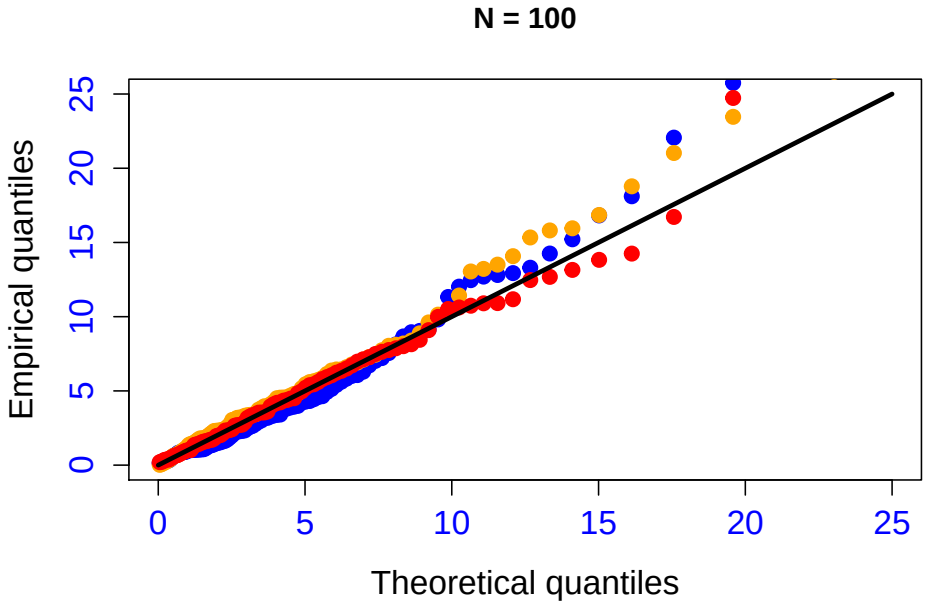


Figure 3.16: Quantile-Quantile plots for three samples from an exponential with mean 5 versus the quantiles of three samples of size 100 from the same distribution.

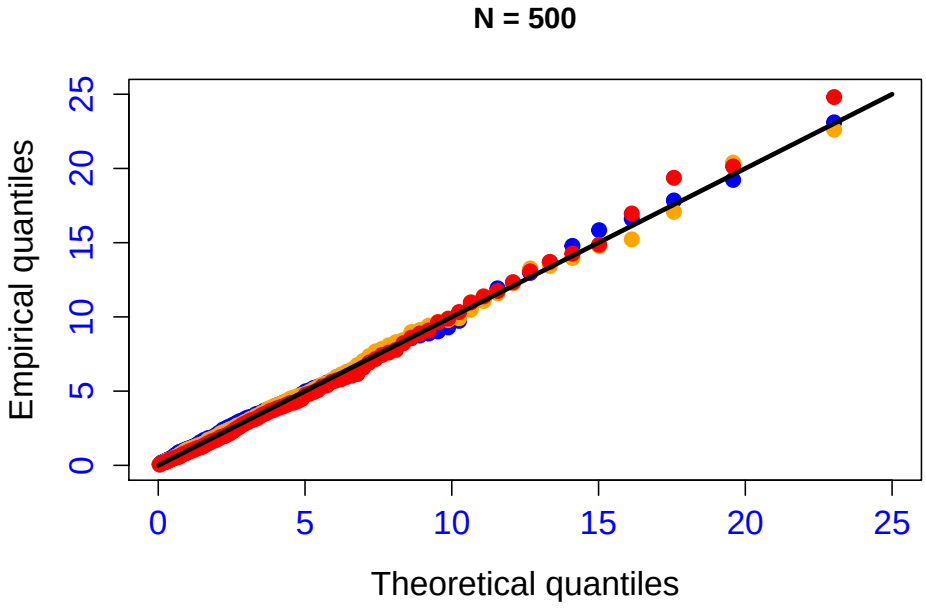


Figure 3.17: Quantile-Quantile plots for three samples from an exponential with mean 5 versus the quantiles of three samples of size 500 from the same distribution.

3.11 Problems

Problem 1. Assume that A, B, C are subsets of Ω. Show that

a. $(A \cup B) \cup C = A \cup (B \cup C)$ and $(A \cap B) \cap C = A \cap (B \cap C)$.
b. $A \cup B = B \cup A$ and $A \cap B = B \cap A$.
c. $(A \cup B) \cap C = (A \cap B) \cup (A \cap C)$ and $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$.
d. $A \cup A^c = \Omega$ and $A \cap A^c = \emptyset$.
e. $(A \cup B)^c = A^c \cap B^c$ and $(A \cap B)^c = A^c \cup B^c$.
f. $A \setminus B = A \cap B^c$.
g. $(A^c)^c = A$.
h. $\Omega^c = \emptyset$ and $\emptyset^c = \Omega$.
i. $A \subset B \implies B^c \subset A^c$.
j. $(A \cup B \cup C)^c = A^c \cap B^c \cap C^c$ and $(A \cap B \cap C)^c = A^c \cup B^c \cup C^c$.

Problem 2. Show the following

a. $P(\emptyset) = 0$.
b. $P(A) = 1 - P(A^c)$.
c. If $A \subset B$ then $P(A) \leq P(B)$.
d. For any A and B, $P(A \cup B) = P(A) + P(B) - P(A \cap B)$.
e. $P(A \cup B) = 1 - P(A^c \cap B^c)$.
f. $P(A \cap B^c) = P(A) - P(A \cap B)$.
g. $P(\cup_{i=1}^n A_i) \leq \sum_{i=1}^n P(A_i)$.
h. $P(\cup_{i=1}^n A_i) \geq \max_i P(A_i)$.

Problem 3. Cryptosporidium is a pathogen that can cause gastrointestinal illness with diarrhea; infections can lead to death in individuals with a weakened immune system. During a recent outbreak of cryptosporidiosis in 21% of two-parent families at least one of the parents contracted the disease. In 9% of the families the father contracted cryptosporidiosis, while in 5% of the families both the mother and father contracted cryptosporidiosis.

a. What event does the probability "one minus the probability that both parents have contracted cryptosporidiosis" represent?
b. What's the probability that either the mother or the father has contracted cryptosporidiosis?
c. What's the probability that the mother has contracted cryptosporidiosis but the father has not?
d. What's the probability that the mother has contracted cryptosporidiosis?
e. What's the probability that neither the mother nor the father has contracted cryptosporidiosis?
f. What's the probability that the father has contracted cryptosporidiosis but the mother has not?

Problem 4. Consider the data set `data_bmi` using the BMI data:

```
file_name = "bmi_age.txt"
data_bmi = read.table(file = file_name, header = TRUE,
```

```
                                    stringsAsFactors = FALSE)
```

and the sets: (1) A subjects with BMI$<$ 22 or male (labeled `SEX=1`); (2) B subjects with AGE$<$ 65 and female (labeled `SEX=0`); and (3) C subjects with BMI$>$ 20 or (male strictly older 50).

a. Obtain A, B, and C directly from the data.
b. Check that the results from Problem 1 hold for these set.
c. Use `R` to extract the sets from the data and conduct the various operations necessary to solve b.

Problem 5. Consider the vector

```
our_names <- c("John","Ciprian","Brian")
```

a. List all possible boostrap samples of `our_names`.
b. How many boostrap samples of `our_names` are there? Explain.
c. Conduct 12 bootstrap samples of `our_names`, print and describe your results.

Problem 6. Consider the vector

```
letter <- c("J","o","h","n","C","i","p","r","i","a","n","B","r","i","a","n")
```

a. Conduct several bootstrap samples of the vector `letter`.
b. Sample 3 times with replacement 7 letters from the vector `letter`.
c. Describe all differences and similarities between the bootstrap of `our_names` and `letters`.
d. Explain, intuitively, what happens in a bootstrap of the two vectors with the letters `i` and `a`.

Problem 7. Suppose $h(x)$ is such that $h(x) > 0$ for $x = 1, 2, \ldots, I$. Argue that $p(x) = h(x)/\sum_{i=1}^{I} h(i)$ is a valid pmf.

Problem 8. Suppose a function $h(\cdot)$ is such that $h(x) > 0$ for every $x \in \mathbb{R}$ and $c = \int_{-\infty}^{\infty} h(x)dx < \infty$. Show that $f(x) = h(x)/c$ is a valid density.

Problem 9. Suppose that, for a randomly drawn subject from a particular population, the proportion of his or her skin covered in freckles follows a density that is constant on $[0, 1]$. This is called the *uniform density on* $[0, 1]$. That is, $f(x) = k$ for $0 \leq x \leq 1$.

a. Draw this density. What must k be?
b. Suppose a random variable, X, follows a uniform distribution between 0 and 1. What is the probability that X is between 0.1 and 0.7? Interpret this probability in the context of the problem.
c. Verify the previous calculation in `R`. What's the probability that $a < X < b$ for generic values $0 < a < b < 1$?
d. What is the cumulative distribution function associated with this probability density function? Plot it in `R`.

e. What is the median of this density? Interpret the median in the context of the problem.
f. What is the 95^{th} percentile? Interpret this percentile in the context of the problem.
g. Do you believe that the proportion of freckles on subjects in a given population could feasibly follow this distribution? Why or why not?

Problem 10. Let U be a continuous random variable with a uniform density on $[0, 1]$ and $F(\cdot)$ be any strictly increasing cdf

a. Show that $F^{-1}(U)$ is a random variable with cdf equal to F.
b. Describe a simulation procedure in `R` that can simulate a sample from a distribution with a given cdf $F(\cdot)$ using just samples from a uniform distribution.
c. Simulate 100 Normal random variables using only simulations from a uniform distribution and the normal inverse cdf function in `R`.

Problem 11. Let $0 \leq \pi \leq 1$ and $f_1(\cdot)$ and $f_2(\cdot)$ be two continuous probability density functions (pdf) with associated cumulative distribution functions (cdf) $F_1(\cdot)$ and $F_2(\cdot)$ and survival functions $S_1(\cdot)$ and $S_2(\cdot)$, respectively. Let $g(x) = \pi f_1(x) + (1 - \pi) f_2(x)$

a. Show that $g(\cdot)$ is a valid density.
b. Write the cdf associated with g in terms of $F_1(\cdot)$ and $F_2(\cdot)$.
c. Write the survival function associated with g in the terms of $S_1(\cdot)$ and $S_2(\cdot)$.

Problem 12. Radiologists have created a cancer risk summary that, for a given population of subjects, follows the *logistic* probability density function

$$f(x) = \frac{e^{-x}}{(1 + e^{-x})^2} \quad \text{for } -\infty < x < \infty$$

a. Show that this is a valid density.
b. Calculate the cdf associated with this pdf.
c. What value do you get when you plug 0 into the cdf? Interpret this result in the context of the problem.
d. Define the *odds* of an event with probability p as $p/(1 - p)$. Prove that the p^{th} quantile of this distribution is $\log\{p/(1-p)\}$, which is the natural log of the odds of an event with probability p.

Problem 13. Epidemiologists consider that the time (in years) until a specific organ of the body stops functioning normally follows the *Pareto* cdf

$$F(x) = \begin{cases} 1 - \left(\frac{x_0}{x}\right)^{\alpha} & \text{for} \quad x \geq x_0 \\ 0 & \text{for} \quad x < x_0 \end{cases}$$

The parameter x_0 is called the scale parameter, while α is the shape or tail index parameter. The distribution is often denoted by $\text{Pa}(x_0, \alpha)$.

a. Derive the density of the Pareto distribution.

b. Plot the density and the cdf for $x_0 = 1, 2, 5$ and $\alpha = 0.1, 1, 10$; interpret these plots and explain the effects of α and x_0.
c. Generate Pareto random variables using simulated uniform random variables in `R`.
d. What is the survival function associated with this density?
e. Interpret in the context of the problem the survival function value for $x = 10$ years when the parameters are $\alpha = 1$ and $x_0 = 2$.
f. Find the p^{th} quantile for this density. For $p = .8$ interpret this value in the context of the problem.

Problem 14. Suppose that a probability density function is of the form cx^k for some constant $k > 1$ and $0 < x < 1$.

a. Find c.
b. Find the cdf.
c. Derive a formula for the p^{th} quantile of the distribution.
d. Let $0 \leq a < b \leq 1$. Derive a formula for $P(a < X < b)$.

Problem 15. Suppose that the time in days until hospital discharge for a certain patient population follows a probability density function $f(x) = c\exp(-x/2.5)$ for $x > 0$.

a. What value of c makes this a valid density?
b. Find the cumulative distribution function for this density.
c. Find the survival function.
d. Calculate the probability that a person takes longer than 11 days to be discharged.
e. What is the median number of days until discharge?

Problem 16. The (lower) incomplete gamma function is defined as $\Gamma(k, c) = \int_0^c x^{k-1}\exp(-x)dx$. By convention $\Gamma(k, \infty)$, the complete gamma function, is written $\Gamma(k)$. Consider a density

$$\frac{1}{\Gamma(\alpha)}x^{\alpha-1}\exp(-x) \quad \text{for } x > 0,$$

where $\alpha > 0$ is a known number.

a. Argue that this is a valid density.
b. Write out the survival function associated with this density using gamma functions.
c. Let β be a known number; argue that

$$\frac{1}{\beta^\alpha\Gamma(\alpha)}x^{\alpha-1}\exp(-x/\beta) \quad \text{for } x > 0$$

is a valid density. This is known as the *gamma density* and is denoted $\Gamma(\alpha, \beta)$.
d. Plot the Gamma density for different values of α and β.

Problem 17. The *Weibull density* is useful in survival analysis. Its form is given by

$$\frac{\gamma}{\beta}x^{\gamma-1}\exp\left(-x^{\gamma}/\beta\right) \quad \text{for } x > 0,$$

where $\gamma > 0$ and $\beta > 0$ are fixed known numbers.

a. Demonstrate that the Weibull density is a valid density.
b. Calculate the survival function associated with the Weibull density.
c. Calculate the median and third quintile of the Weibull density.
d. Plot the Weibull density for different values of γ and β.

Problem 18. The Beta function is given by $B(\alpha,\beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1}$ for $\alpha > 0$ and $\beta > 0$. It can be shown that

$$B(\alpha,\beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta).$$

The *Beta density* is given by $\frac{1}{B(\alpha,\beta)}x^{\alpha-1}(1-x)^{\beta-1}$ for fixed $\alpha > 0$ and $\beta > 0$. This density is useful for modeling random variables with values in $(0,1)$ (e.g. proportions or probabilities).

a. Argue that the Beta density is a valid density.
b. Argue that the uniform density is a special case of the Beta density.
c. Plot the Beta density for different values of α and β.

Problem 19. A famous formula is $e^{\lambda} = \sum_{x=0}^{\infty}\frac{\lambda^x}{x!}$ for any value of λ. Assume that the count of the number of people infected with a particular virus per year follows a probability mass function (pmf) given by

$$P(X = x) = \frac{e^{-\lambda}\lambda^x}{x!} \quad \text{for } x = 0, 1, 2, \ldots$$

where $\lambda > 0$ is a fixed known number. This is known as the *Poisson mass function* and is typically denoted by $P(\lambda)$.

a. Argue that $\sum_{x=0}^{\infty} P(X = x) = 1$.

Problem 20. Let X be the number of sexual contacts a person has until contracting HIV. Let p be the probability p of contracting HIV from one sexual contact. The mass function for this process is given by $P(X = x) = p(1-p)^{x-1}$ for $x = 1, 2, \ldots$. This is called the *geometric probability mass function*.

a. Argue mathematically that this is a valid probability mass function. Hint: the geometric series is given by $\frac{1}{1-r} = \sum_{k=0}^{\infty} r^k$ for $|r| < 1$.
b. Calculate and interpret in the context of the problem the survival distribution $P(X > x)$ for the geometric distribution.

Problem 21. We say that a random variable X is stochastically less than a random variable Y if $P(X > x) \leq P(Y > x)$ for every $x \in \mathbb{R}$.

a. Show that if X is stochastically less than Y, $F_X(x) \geq F_Y(x)$ for every $x \in \mathbb{R}$, where $F_X(\cdot)$ and $F_Y(\cdot)$ are the cdfs of the X and Y random variables, respectively.

b. Show that X is stochastically less than $a + X$ for every $a > 0$.
c. Show that X is not stochastically less than aX for every $a > 0$.
d. Show that if $X > 0$ is a positive random variable, aX is stochastically less than X for every $a < 0$.

Problem 22. Consider the `bmi_age.txt` dataset. Construct a QQ plot for the variables AGE and BMI versus the theoretical Normal distribution. Bootstrap both vectors three times and display the QQ plots for every sample.

Problem 23. We would like to know whether there is an association between AGE and BMI in the `bmi_age.txt`. Bootstrap the data 10000 times and for every bootstrap sample calculate the correlation between the BMI and AGE. This can be obtained in `R` as

```
cor_BMI_AGE<-cor(BMI,AGE)
```

a. Plot a histogram of these correlations.
b. What percent of these correlations is below 0?

Problem 24. For problems 1a, 1b, 1c, identify nontrivial subsets A, B, and C of subjects from the `bmi_age.txt` dataset and illustrate the operations using the rules of logic. Identify and describe the logical indexes in `R` that extract these subsets and illustrate the connection between the theoretical set operators and logical operators in `R`.

3.12 Supplementary R training

This part of the book can be used as laboratory material or as additional material for learning `R`. Let us start with data collected during a class survey asking which of 16 homework problems students would like to get help with during lab. There were 13 students who answered the questionnaire and each indicated one or more problems.

We will plot the results of the survey and learn a few new plotting skills. We start by creating the data vectors

```
questions <- 1:16
n_students<-13
votes <- c(5,0,3,5,2,9,2,3,6,2,2,4,4,3,2,3)
votes.pct <- round((votes/n_students)*100,1)
```

Reorder questions and number of votes by descending number of votes

```
o <- order(votes, decreasing=TRUE)
o
```

```
 [1]  6  9  1  4 12 13  3  8 14 16  5  7 10 11 15  2
```

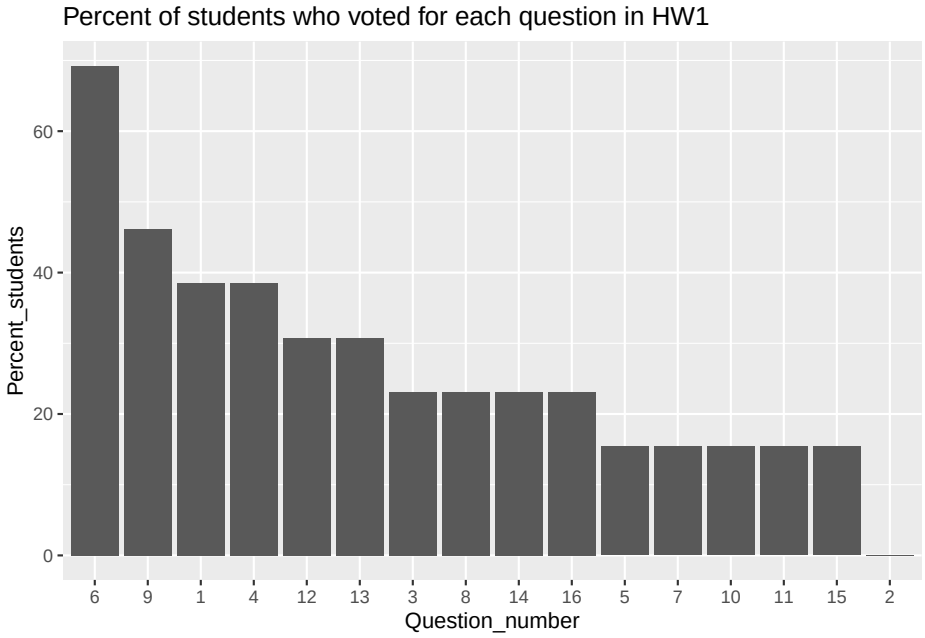


Figure 3.18: Percent students by question number.

Once votes are re-ordered in decreasing order, we re-arrange all data vectors according to the same index. This will help make a nicer plot.

```
questions.sort <- questions[o]
votes.sort <- votes[o]
votes.pct.sort <- votes.pct[o]
```

We will use the `ggplot2` package by (Wickham 2016). Many biostatisticians prefer to use `ggplot2` over `plot`, but there is no agreement with respect to what to use. The authors of this book are split in their preferences and we prefer to work with both functions.

```
library(ggplot2)
```

First make a bar plot of question numbers versus votes using the function `ggplot` function in the `ggplot2` package. We build a data frame first

```
data_questions <- data.frame(
    Question_number = factor(questions.sort, levels=questions.sort),
    Percent_students = votes.pct.sort)
```

Create matrices in `R`

```
my.matrix.rows <- matrix(c(1:9), nrow = 3, ncol = 3)
my.matrix.cols <- matrix(c(1:9), nrow = 3, ncol = 3, byrow = TRUE)
```

```
my.matrix.rows
```

```
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
```

```
my.matrix.cols
```

```
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
[3,]    7    8    9
```

and show how R handles matrix calculations. This is how to do entry-wise calculations

```
my.matrix.rows*my.matrix.cols
```

```
     [,1] [,2] [,3]
[1,]    1    8   21
[2,]    8   25   48
[3,]   21   48   81
```

```
my.matrix.rows+my.matrix.cols
```

```
     [,1] [,2] [,3]
[1,]    2    6   10
[2,]    6   10   14
[3,]   10   14   18
```

and true matrix multiplication

```
my.matrix<-my.matrix.rows %*% my.matrix.cols
my.matrix
```

```
     [,1] [,2] [,3]
[1,]   66   78   90
[2,]   78   93  108
[3,]   90  108  126
```

You may want to do these operations by hand and check them against the results in R. Here is how to access the first row and the entry $(2, 3)$ of `my.matrix`

```
my.matrix[1,]
```

```
[1] 66 78 90
```

```
my.matrix[2,3]
```

```
[1] 108
```

Simulate data from a Normal distribution and plot a histogram of the simulations. To understand the structure of the function used for random sampling you can simply use the `help` command in `R`

```
#the help command (most important r command!)
?rnorm
#equivalent to previous line
help(rnorm)
```

Here `eval=FALSE` is used to illustrate the `R` code in `Rmarkdown` without evaluating it.

```
#simulate two independent random samples of length 100 from N(0,1)
random.sample.1 <- rnorm(100, mean = 0, sd = 1)
random.sample.2 <- rnorm(100, mean = 2, sd = 1)

#plot a histogram of the 100 observations
hist(random.sample.1, col = rgb(1,0,0,0.5),breaks=20,xlim=c(-3,5),
     cex.lab=1.3,cex.axis=1.3,col.axis="blue",
     main="Two transparent histograms",
     xlab = "Random samples")
#add the histogram of the second 100 samples
hist(random.sample.2,col = rgb(0,0,1,0.5),add =T,breaks=20)
```

Alternative ways to plot the simulated data using `boxplot`

```
#boxplot
boxplot(random.sample.1, random.sample.2, col = c('blue', 'red'))
```

and `qqplot`

```
#qqplot
qqplot(random.sample.1, random.sample.2,
       main = 'Random Sample 1 versus Random Sample 2')
```

Evaluate statistics of the sample including the mean, standard deviation, min and max

```
#calculate the mean of each of the random samples
mean(random.sample.1)
```

```
[1] 0.08342127
```

```
mean(random.sample.2)
```

```
[1] 1.899799
```

```
#calculate the standard deviation
sd(random.sample.1)
```

```
[1] 0.8879839
```

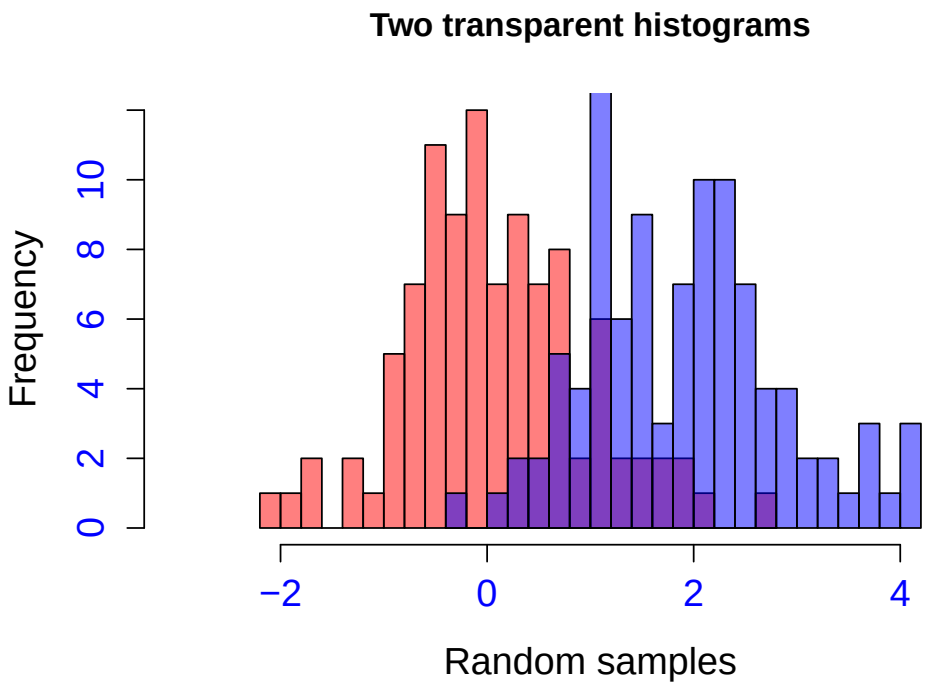


Figure 3.19: Plotting two histograms of two different samples in the same figure.

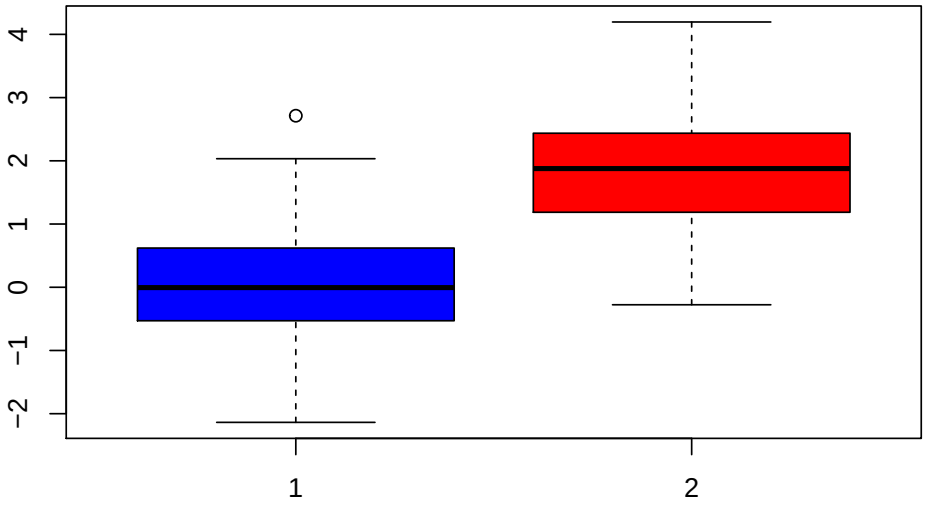


Figure 3.20: Plotting the boxplots of two different samples in the same figure.

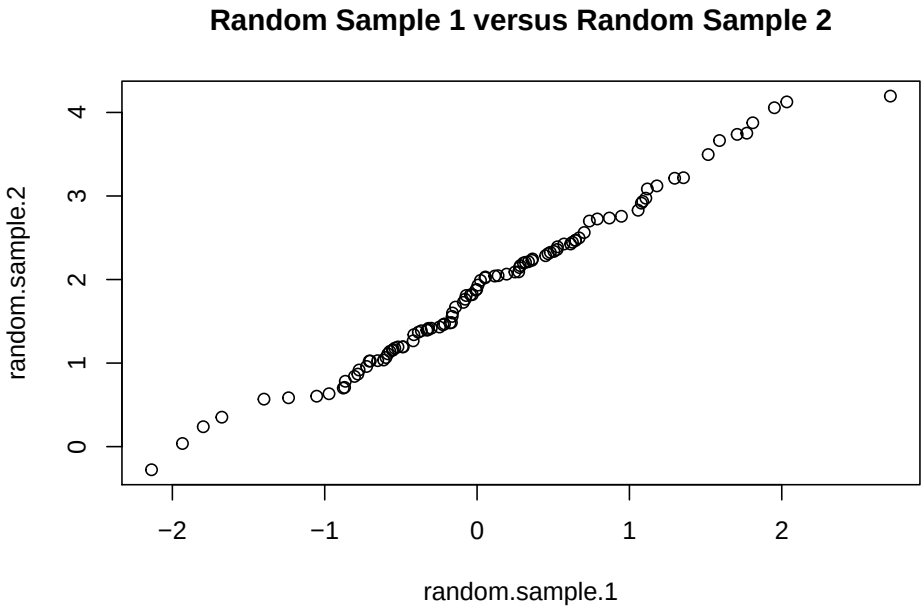


Figure 3.21: Plotting the QQ plots of two different samples in the same figure.

```
sd(random.sample.2)
```

```
[1] 0.9459916
```

```
##calculate the min and max
min(random.sample.1)
```

```
[1] -2.136947
```

```
max(random.sample.2)
```

```
[1] 4.195687
```

Simulate data from other distributions

```
#simulate 100 samples from the exponential distribution
random.exponential <- rexp(100, rate = 1)

#simulate 1000 samples from the exponential distribution
random.uniform <- runif(1000, min = 0, max = 1)
```

Chapter 4

Mean and variance

This chapter covers the following topics

- Mean or expected value
- Sample mean and bias
- Variance, standard deviation, coefficient of variation
- Variance interpretation: Chebyshev's inequality
- Supplementary `R` training

4.1 Mean or expected value

The expected value or mean of a random variable is the center of mass of its distribution. Below we will discuss the mean of discrete and continuous random variables separately.

4.1.1 Mean of a discrete random variable

For a discrete random variable X with pmf $p(x)$, the mean is defined as follows

$$E[X] = \sum_x xp(x) = \sum_{\text{possible values of } X} xp(x) \ ,$$

where the sum is taken over all possible values x of X. The expected value is the center of mass of the locations x with weights (probability) $p(x)$. This definition applies to $x \in \mathbb{R}$, but can be used for $x \in \mathbb{R}^k$ with $k \geq 1$. The interpretation remains the same irrespective of the dimension of x. The mean is the simplest and most important thing to understand in biostatistics. Indeed, most of biostatistics deals with how and why to calculate means.

Consider first the case of a random variable, X, that can take four values, $x \in \{-4, -3, 3, 4\}$ with the same probability, $p(x) = 0.25$. The mechanics of calculating the mean of this random variable is simple and require just plugging in values into the mean formula

$$E(X) = (-4) \times 0.25 + (-3) \times 0.25 + 3 \times 0.25 + 4 \times 0.25 = 0 \; .$$

The left-top panel in Figure 4.1 provides the intuition behind this calculation. The orange triangle points to the mean, or center of mass, of the distribution. One could imagine a playground seesaw, where four children of exactly the same weight sat at the locations indicated by $\{-4, -3, 3, 4\}$. If the seesaw rested on the mean $E(X) = 0$ then the four children would be perfectly balanced and nothing would happen. Of course, as every parent knows, this is a gross misrepresentation of reality as there has been no reported case of four kids on a playground in perfect balance. The idea, though, should be the same: the mean has the property that in the absence of external forces the system would be in perfect balance if suspended on or by the mean.

Consider now the case when the random variable X takes four other values $x \in \{-4, 1, 3, 4\}$ with the same probability, $p(x) = 0.25$. The mean of X in this case is:

$$E(X) = (-4) \times 0.25 + 1 \times 0.25 + 3 \times 0.25 + 4 \times 0.25 = 1 \; .$$

The right-top panel in Figure 4.1 below provides a representation of this random variable distribution and its mean, $E(X) = 1$. In this case we can imagine that three of the children are now on one side of the seesaw and the child who was at -3 moved to 1. The mean, or center of mass, has shifted towards the right side of the seesaw and is, in fact, under the child at 1. The system would be in balance if the seesaw rested on the mean $E(X) = 1$. If the seesaw rested to the left of the mean, it would tilt to the right and if it rested to the right of the mean, it would tilt to the left.

As children never weigh the same or adults may be forced into the pleasure of seesaw play, let us consider the case when we have a random variable X that takes the same four values $x \in \{-4, 1, 2, 3\}$, but with probabilities $p(X = -4) = 0.6$, $P(X = 1) = 0.2$, $P(X = 3) = 0.15$, $P(X = 4) = 0.05$. The mean of X in this case is

$$E(X) = (-4) \times 0.60 + 1 \times 0.20 + 3 \times 0.15 + 4 \times 0.05 = -1.55 \; .$$

A representation of the distribution of this random variable and its mean is shown in the left-bottom panel of Figure 4.1. Note that the mean has shifted to the left in the direction of the person who weighs the most. Similarly, the right-bottom panel displays the mean for the case of a random variable X that takes the same four values $x \in \{-4, 1, 3, 4\}$, but with probabilities $p(X = -4) = 0.05$,

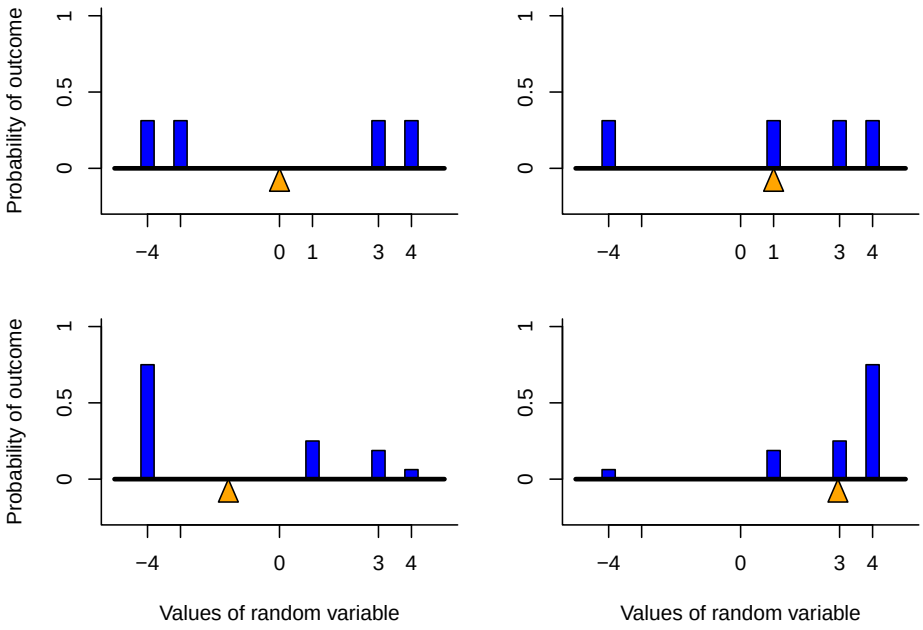


Figure 4.1: Examples of discrete random variables and their means.

$P(X = 1) = 0.15$, $P(X = 3) = 0.20$, $P(X = 4) = 0.60$. The mean of X in this case is

$$E(X) = (-4) \times 0.05 + 1 \times 0.15 + 3 \times 0.20 + 4 \times 0.60 = 2.95 \, .$$

The mean has a few important properties. For example, $\min(X) \leq E(X) \leq \max(X)$. This is indicated in the panels in Figure 4.1 by the tip of the orange triangle always being between the largest and smallest value possible for the random variable X. Moreover, the mean $E(X)$ is not necessarily one of the values that the random variable takes. Indeed, in three of the examples discussed above the mean was 0, -1.55, and 2.95, respectively, and none of these values was among the possible values for X. The mean is often a good descriptor of the central tendency of the distribution, though it can be heavily influenced by exceptionally large or small values.

4.1.1.1 Example: Bernoulli random variables

Consider the example in which the outcome of an experiment is whether a person survives for more than five years after a lung cancer diagnosis, and let us assume that the probability of survival is $P(X = 1) = 0.5$. We would like to

know the expected value of survival and we simply need to apply the formula

$$E(X) = 0 \times P(X = 0) + 1 \times P(X = 1) = 0 \times 0.5 + 1 \times 0.5 = 0.5 \, .$$

If we use the same geometric intuition from the seesaw example this answer is obvious: if two equal weights are placed at 0 and 1, the center of mass is at 0.5. In practice the probability of survival is not 0.5, and could even be unknown. Suppose that $P(X = 1) = p$, which implies that $P(X = 0) = 1 - p$, as the person will either survive or die in five years. In this case the expected value of the survival variable is

$$E(X) = 0 \times P(X = 0) + 1 \times P(X = 1) = 0 \times (1 - p) + 1 \times p = p \, ,$$

indicating that for a Bernoulli random variable, X, with probability of success p the mean (or expected value) is $E(X) = p$. The interpretation of this result is that if one selects at random a group of people from a population of individuals with cancer with five-year survival probability of p, the expected proportion of individuals who survive after five years is $100p\%$.

4.1.1.2 Example: Die roll random variables

Let us consider the experiment of rolling a fair die and denote by X the associated random variable. The variable X can take only the values $\{1, 2, \ldots, 6\}$ and, because the die is fair, $P(X = x) = 1/6$ for all $x \in \{1, 2, \ldots, 6\}$. Here, and throughout the book, we will use upper case X to denote the random variable (what we know about the experiment before the experiment is run) and the lower case x to denote the outcome of the random variable (what we observe as a result of the experiment).

We would like to calculate the mean, or expected value, of X

$$E(X) = \sum_{x=1}^{6} xP(X = x) = 1 \times \frac{1}{6} + \ldots + 6 \times \frac{1}{6} = \frac{21}{6} = 3.5 \, .$$

Again, the geometric argument makes this answer obvious without calculation. Just as in many other examples, the mean is not the most likely value the random variable takes. In fact, $P\{X = E(X)\} = P(X = 3.5) = 0$. Now, we would like to investigate how to obtain the mean of X^2. First, observe that X^2 takes only the values $\{1^2, 2^2, \ldots, 6^2\}$ and takes each of these values with the same probability, 1/6. Thus,

$$E(X^2) = \sum_{x=1}^{6} x^2 P(X^2 = x^2) = 1^2 \times \frac{1}{6} + \ldots + 6^2 \times \frac{1}{6} = \frac{91}{6} \approx 15.167 \, .$$

Observe that $\text{Var}(X) = E(X^2) - \{E(X)\}^2 = 15.167 - 12.25 = 2.917 > 0$ indicating that, in this case, $E(X^2) \geq \{E(X)\}^2$. This is a particular example of

Jensen's inequality (more about this later) and holds for every random variable. The quantity $\text{Var}(X) = E(X^2) - \{E(X)\}^2$ will be referred to as the variance of X and will play an important role in this book. We could also use R to conduct these calculations as follows

```
x=1:6                                   #Vector of possible values for a 1-die roll
ex2=sum(x^2*rep(1/6,6))                 #Expected value of X^2
ex=sum(x*rep(1/6,6))                    #Expected value of X
varx=ex2-ex^2                           #Variance of X
round(varx,digits=3)
```

```
[1] 2.917
```

Similarly, we can obtain the expected value of the random variable $\sqrt{X}$ as

$$E(\sqrt{X}) = \sum_{x=1}^{6} \sqrt{x} P(\sqrt{X} = \sqrt{x}) = \sqrt{1}\frac{1}{6} + \ldots + \sqrt{6}\frac{1}{6} \approx 1.805 \, .$$

In contrast to the case when we took the square transformation, $E(\sqrt{X}) - \sqrt{E(X)} = 1.805 - \sqrt{3.5} \approx -0.066 < 0$. For this case $E(\sqrt{X}) \leq \sqrt{E(X)}$, which also happens to be a particular case of the Jensen's inequality. The reason that $E(X^2) \geq \{E(X)\}^2$ is that the function $h(x) = x^2$ is convex (its graph "holds water") while the reason for having the inequality $E(\sqrt{X}) \leq \sqrt{E(X)}$ is that the function $h(x) = \sqrt{x}$ is concave (its graph "does not hold water").

4.1.1.3 Example: categorical distribution

The Bernoulli random variable is very useful to represent Yes/No, disease/healthy outcomes, but we may want to do more than that in practice. For example, consider the outcome of an experiment where the mental health of individuals is assessed in a prospective cohort study. For each individual the degree of impairment is assigned to "healthy", "mild", or "serious". A natural reaction for this would be to say that this is a random variable, X, with three possible outcomes, but it would make no sense to define the mean of this random variable as

$$E(X) = \text{healthy} \times p(X = \text{healthy}) + \text{mild} \times P(X = \text{mild}) + \text{serious} \times P(X = \text{serious}) \, ,$$

as we cannot assign numbers to labels. Of course, we can assign, for example, the number 1 to healthy, 2 to mild, and 3 to serious, which would provide a number for the formula above. However, such assignments are arbitrary and lead to different values for different label assignments. Thus, an alternative strategy needs to be found. The idea is to consider that the random variable is a vector with three 0/1 entries, where all entries are zero with the exception of the one corresponding to the particular mental health status. More precisely, $X = (X_1, X_2, X_3)$, where $X_i \in \{0, 1\}$ for $i = 1, 2, 3$ and $\sum_{i=1}^{3} X_i = 1$. For example, if the health of the person is mildly impaired, the realization of the

random variable is $(0, 1, 0)$, while if the health of the person is seriously impaired, the realization is $(0, 0, 1)$. The mean for multivariate vectors is defined entrywise, that is

$$E(X) = \{E(X_1), E(X_2), E(X_3)\} ,$$

where $E(X_i)$ is the proportion of persons who belong to category i. For example, $E(X_1)$ is the proportion of persons whose mental health is not impaired, while $E(X_2)$ is the proportion of persons with mild mental impairment. In this case we need to have $E(X_1) + E(X_2) + E(X_3) = 1$ because these are mutually exclusive categories and they cover the entire population. This definition is a generalization of the Bernoulli distribution, which would only have two categories and can easily be extended to an arbitrary number of categories. In this book we do not emphasize multivariate distributions, but such data are quite common in health studies.

4.1.2 Mean of a continuous random variable

For a continuous random variable, X, with pdf $f(x)$ the mean or expected value is defined as follows

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx .$$

Geometrically, if we have a continuous body with the density of mass distributed according to $f(x)$, $E(X)$ is the center of mass of that body. We now consider a few examples of continuous distributions that will be used extensively in this book.

4.1.2.1 Example: continuous distributions

Consider that the random variable X with a uniform distribution between 0 and 1, and we denote $X \sim \mathrm{U}[0, 1]$. The pdf for this distribution is $f(x) = 1$ if $x \in [0, 1]$ and 0 otherwise. The plot for this density is shown in Figure 4.2.

This is a valid density because $f(x) \geq 0$ for every x and

$$\int_{-\infty}^{\infty} f(x) dx = \int_{-\infty}^{0} 0 dx + \int_{0}^{1} 1 dx + \int_{1}^{\infty} 0 dx = x \Big|_0^1 = 1 .$$

We would like to calculate the expected value of a variable X with a Uniform distribution between zero and one. We apply the rules of integration and obtain

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx = \int_{-\infty}^{0} x \times 0 dx + \int_{0}^{1} x \times 1 dx + \int_{1}^{\infty} x \times 0 dx = \frac{x^2}{2} \Big|_0^1 = \frac{1}{2} .$$

This is the mean of the standard Uniform distribution. In general, if $a < b$ we say that the variable X has a uniform distribution on $[a, b]$ and we denote $X \sim$

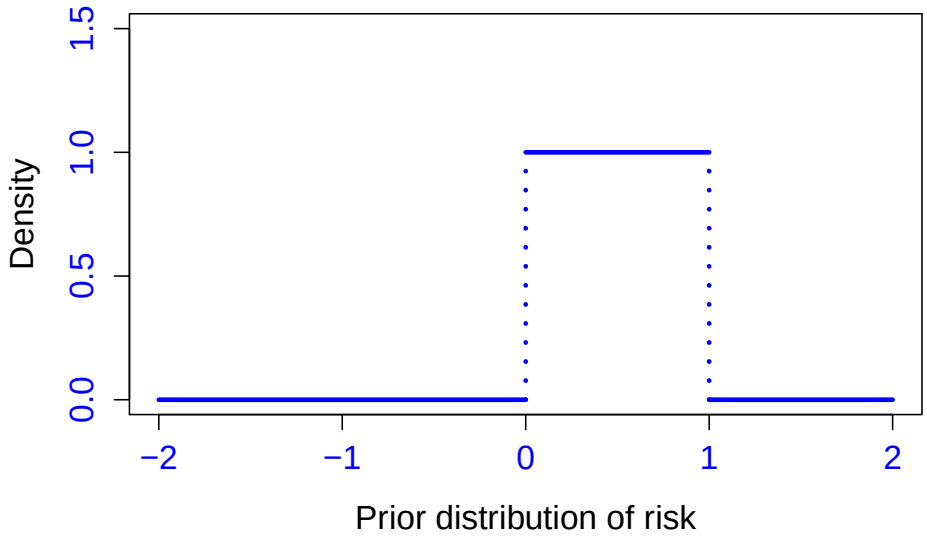


Figure 4.2: The pdf of the Uniform density on [0,1].

U$[a,b]$ if the pdf of X is $f(x) = 1/(b-a)$ for $x \in [a,b]$ and 0 everywhere else. The Uniform distribution is often used to characterize the lack of information about a probability. For example, if we are just starting a study on the probability of death from a new type of infection, we could reasonably assume that we do not know anything about it and that its prior distribution is U$[0,1]$. The distribution is also useful to generate Monte Carlo samples from other distributions. Drawing random samples from the Uniform distribution is easy in R

```
y1<-runif(10000)            #random samples from a U[0,1]
y2<-runif(10000,-2,15)      #random samples from a U[-2,15]
```

We now introduce the following four additional continuous distributions: Normal (or Gaussian), Exponential, Gamma, and Beta. A random variable is said to follow a standard Normal distribution and is denoted as $X \sim N(0,1)$ if its pdf is

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \quad \text{for} \quad x \in \mathbb{R}\,.$$

This is a pdf because $f(x) > 0$ for every $x \in \mathbb{R}$ and $\int_{-\infty}^{\infty} e^{-x^2/2} dx = \sqrt{2\pi}$. We do not prove this result, as it is just above the limit of what is necessary to know from calculus. However, it is easy to show that

$$E(X) = \int_{\infty}^{\infty} x e^{-\frac{x^2}{2}} dx = \int_{-\infty}^{0} x e^{-x^2/2} dx + \int_{0}^{\infty} x e^{-x^2/2} dx\,.$$

In the first integral we apply the change of variable $y = -x$ and we obtain

$$\int_{-\infty}^{0} x e^{-x^2/2} dx = \int_{\infty}^{0} (-y) e^{-y^2/2} d(-y) = \int_{\infty}^{0} y e^{-y^2/2} dy = -\int_{0}^{\infty} y e^{-y^2/2} dy\,,$$

which shows that the mean of a random variable $X \sim N(0,1)$ is $E(X) = 0$. The top-left panel in Figure 4.3 displays the pdf of a standard normal distribution (blue) together with the location of its mean (red vertical line). The pdf is symmetric around zero and has the famous "bell shape" with the tails of the distribution decreasing rapidly to zero as one is at least 2 or 3 away for the mean. It is easy to show, using exactly the same trick, that if a distribution is symmetric around μ then the mean of the distribution is μ. The Normal distribution is often used to characterize the distribution of errors and will play a crucial role in the Central Limit Theorem (CLT) and the construction of confidence intervals.

A random variable is said to follow an exponential distribution with mean $\theta > 0$ and is denoted as $X \sim \exp(\theta)$ if its pdf is $f(x;\theta) = \theta e^{-\theta x}$ for $x \geq 0$ and 0 otherwise. This distribution is often used in survival analysis to model the time until a certain event happens (e.g. death, cure, release from the hospital). Because time is positive, the support of the exponential distribution is the set of positive real numbers, $\mathbb{R}_+$. It is relatively easy to show that $f(x;\theta)$ is a pdf and that

$$E(X) = \int_0^\infty \theta x e^{-\theta x} dx = \int_0^\infty -x \frac{\partial}{\partial x}\{e^{-\theta x}\} dx = -x e^{-\theta x}\Big|_0^\infty + \int_0^\infty e^{-\theta x} dx = \frac{1}{\theta}\,.$$

The third equality follows from the integration by parts formula (please refresh your calculus memory, if necessary), while the last inequality holds because $\lim_{x\to\infty} x e^{-\theta x} = 0$ and $\int_0^\infty e^{-\theta x} dx = -\frac{1}{\theta}\int_0^\infty \frac{\partial}{\partial x}\{e^{-\theta x}\} dx$. The top-right panel in Figure 4.3 displays the pdf of the exponential distribution (blue line) with $\theta = 1/5$ and the mean of the distribution $E(X) = 1/\theta = 5$ (red vertical line).

A random variable is said to follow a Gamma distribution with parameters $\alpha > 0$ and $\beta > 0$ and is denoted as $X \sim \text{Gamma}(\alpha,\beta)$ or $X \sim \Gamma(\alpha,\beta)$ if its pdf is

$$f(x;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-x\beta} \quad \text{for} \quad x > 0$$

and 0 otherwise. Here $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x} dx$ is the Gamma function. The exponential distribution $\exp(\theta)$ is a particular case of the Gamma family of distributions because it corresponds to $\alpha = 1$ and $\beta = 1/\theta$. The Gamma distribution is also used in survival analysis, but has applications in other areas, including modeling annual healthcare expenditures, pathogen or toxicant concentrations in water, daily calorie intake, knee reaction time in a medical experiment, and the kidney glomerular filtration rate. We now show that this is indeed a legitimate pdf. Obviously, $f(x;\alpha,\beta) \geq 0$ for every $x \in \mathbb{R}$ and, by making the change of variable $y = x\beta$

$$\int_0^\infty \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-x\beta} dx = \frac{\beta^\alpha}{\Gamma(\alpha)} \int_0^\infty \left\{\frac{y}{\beta}\right\}^{\alpha-1} e^{-y} d\left(\frac{y}{\beta}\right) = \frac{\beta^\alpha}{\beta^\alpha \Gamma(\alpha)} \int_0^\infty y^{\alpha-1} e^{-y} dy = 1\,.$$

The last equality holds by the definition of the $\Gamma(\alpha)$ function and the result implies that $\int_0^\infty x^{\alpha-1} e^{-x\beta} dx = \Gamma(\alpha)/\beta^\alpha$ for any $\alpha > 0$ and $\beta > 0$. Let us

calculate the mean of the Gamma distribution

$$E(X) = \frac{\beta^\alpha}{\Gamma(\alpha)} \int_0^\infty x x^{\alpha-1} e^{-\beta x} dx = \frac{\beta^\alpha}{\Gamma(\alpha)} \int_0^\infty x^{(\alpha+1)-1} e^{-\beta x} dx = \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\Gamma(\alpha+1)}{\beta^{\alpha+1}} = \frac{\Gamma(\alpha+1)}{\Gamma(\alpha)\beta} .$$

We show that $\Gamma(\alpha+1) = \alpha\Gamma(\alpha)$, which proves that $E(X) = \alpha/\beta$. Note that

$$\Gamma(\alpha+1) = \int_0^\infty x^\alpha e^{-x} dx = \int_0^\infty x^\alpha \frac{\partial}{\partial x}\{-e^{-x}\} dx = -x^\alpha e^{-x}\Big|_0^\infty + \int_0^\infty \frac{\partial}{\partial x}\{x^\alpha\} e^{-x} dx = \alpha\Gamma(\alpha) .$$

For the third equality we applied again the integration by parts formula and the last equality follows from the definition of the $\Gamma(\alpha)$ function and the fact that $\frac{\partial}{\partial x}\{x^\alpha\} = \alpha x^{\alpha-1}$. The bottom-left panel in Figure 4.3 displays the Gamma$(2, 1/5)$ density (blue) and its mean $E(X) = \alpha/\beta = 2/(1/5) = 10$ (red vertical line). To avoid confusion, in `R` the Gamma distribution is parameterized in terms of the shape parameter α and the scale parameter $1/\beta$. Whenever working with the Gamma distribution it is important to know which exact parameterization is used. An easy way to remember is that in the parameterization in this book $E(X) = \alpha/\beta$, whereas in `R` and other sources $E(X) = \alpha\gamma$, where $\gamma = 1/\beta$.

A random variable is said to follow a Beta distribution with parameters $\alpha > 0$ and $\beta > 0$ and is denoted as $X \sim \text{Beta}(\alpha, \beta)$ if its pdf is

$$f(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1}(1-x)^{\beta-1} \quad \text{for} \quad x \in (0, 1)$$

and 0 otherwise, where $B(\alpha, \beta) = \int_0^1 x^{\alpha-1}(1-x)^{\beta-1} dx$. It is easy to show that this is a pdf and the mean is

$$E(X) = \frac{1}{B(\alpha, \beta)} \int_0^1 x x^{\alpha-1}(1-x)^{\beta-1} dx = \frac{B(\alpha+1, \beta)}{B(\alpha, \beta)} .$$

It can be shown that $B(\alpha, \beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ (proof omitted here), which implies that

$$B(\alpha+1, \beta) = \frac{\Gamma(\alpha+1)\Gamma(\beta)}{\Gamma(\alpha+\beta+1)} = \frac{\alpha\Gamma(\alpha)\Gamma(\beta)}{(\alpha+\beta)\Gamma(\alpha+\beta)} = \frac{\alpha}{\alpha+\beta} B(\alpha, \beta) .$$

Therefore, the mean of the Beta distribution with parameters α and β is $\alpha/(\alpha+\beta)$. The domain of the Beta distribution is $(0, 1)$, which makes it particularly well suited for modeling probability and risk. The bottom-right panel in Figure 4.3 displays the Beta distribution (blue) with parameters $\alpha = 10$ and $\beta = 2$ and its mean $E(X) = \alpha/(\alpha+\beta) = 10/12 \approx 0.833$ (red vertical line).

Truthfully, this is the point where most students are yawning the hardest and we are fully aware of that. But, we did not promise that this would be a math-free book and in the first chapter we provided fair warnings about the level of calculus we expect. More importantly, this is about the highest level of calculus

required and we will use it only when it is truly necessary. Probably one of the most important skills of a biostatistician is to know what is not necessary to solve a particular problem. Difficult math is seldom the answer in data analysis, though intuition and experience can go a long way.

4.1.2.2 Example: distribution plots

We now show how to make the panels in Figure 4.3 for the four different distributions: $N(0,1)$ (top-left panel), exp(1/5) (top-right panel), Gamma$(2,1/5)$ (bottom-left panel), and Beta$(10,2)$ (bottom-right panel). The corresponding pdfs are shown as blue lines and the means are indicated as vertical red lines.

```
par(mfrow = c(2,2),
          oma = c(1,1,1,1) + 0.1,
          mar = c(4,4,0,0) + 0.1)
x1=seq(-3.5,3.5,length=101)
y1<-dnorm(x1)
plot(x1,y1,type="l",col="blue",lwd=3,xlab="N(0,1)",ylab="Density",
     main="",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(c(0,0),c(0,0.4),col="red",lwd=3)

x2=seq(0,15,length=101)
y2<-dexp(x2,1/5)
plot(x2,y2,type="l",col="blue",lwd=3,xlab="exp(1/5)",ylab="",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(c(5,5),c(0,0.2),col="red",lwd=3)

x3=seq(0,40,length=101)
y3<-dgamma(x3,shape=2,scale=5)
plot(x3,y3,type="l",col="blue",lwd=3,
     xlab="Gamma(2,1/5)",ylab="Density",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(c(10,10),c(0,0.073),col="red",lwd=3)

x4=seq(0,1,length=101)
y4<-dbeta(x4,shape1=10,shape2=2)
plot(x4,y4,type="l",col="blue",lwd=3,xlab="Beta(10,2)",ylab="",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(c(0.83,0.83),c(0,4.2),col="red",lwd=3)
```

The mean is the center of mass both for discrete and continuous distributions, though the mean is not required to be a possible outcome of the experiment. These plots indicate that the mean does not need to be the middle of the variable domain or the point of maximum probability (likelihood). However, the mean is the point of maximum likelihood for symmetric unimodal distributions, such as the Gaussian (normal) or Laplace (double-exponential) distributions. The

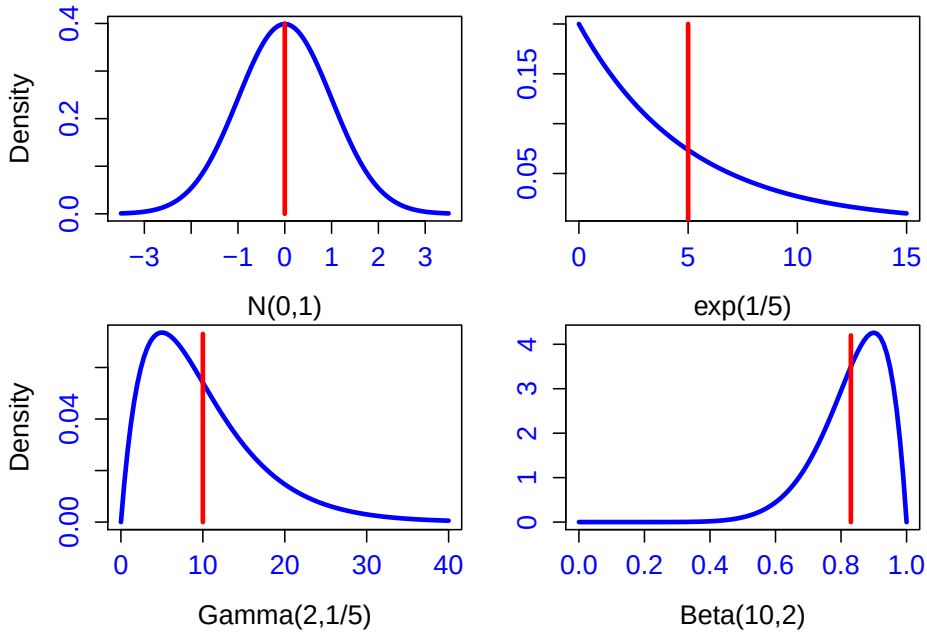


Figure 4.3: The pdfs of four continuous distributions.

mean is not typically the median. However, for symmetric distributions, often used as error distributions in statistical models, the mean and the median are equal. The mean can be heavily affected by the skewness of the distribution and may have different interpretations depending on the shape of the distribution.

4.1.3 Rules about expected values

The expected value is a linear operator. This means that if a and b are not random and X and Y are two random variables then

- $E(aX + b) = aE(X) + b$
- $E(X + Y) = E(X) + E(Y)$.

We will prove the first result, while the second result will be proved when we discuss the pdf of a vector of random variables. Let $f_X(\cdot)$ and $f_Y(\cdot)$ be the pdf of the distribution of the X and Y variables, respectively. By definition

$$E(aX + b) = \int (ax + b) f_X(x) dx = a \int x f_X(x) + b \int f_X(x) dx = aE(X) + b \,,$$

because $E(X) = \int x f_X(x) dx$ and $\int f_X(x) dx = 1$, as $f_X(\cdot)$ is a pdf. This result shows that the expected value is a linear operator. That is, if $h(\cdot)$ is a linear function, then $E\{h(X)\} = h\{E(X)\}$. If $h(\cdot)$ is not linear, then $E\{h(X)\} \neq h\{E(X)\}$. For example, $E(X^2) \neq E^2(X)$ and $E\{\log(X)\} \neq \log\{E(X)\}$.

We now show that $E(X^2) \geq \{E(X)\}^2$ for any random variable with the exception of the trivial case when $X = E(X)$. As $\{X - E(X)\}^2 \geq 0$ it follows that

$$\begin{aligned} 0 &\leq E\{X - E(X)\}^2 \\ &= E\{X^2 - 2XE(X) + E^2(X)\} \\ &= E(X^2) - 2E(X)E(X) + E^2(X) \\ &= E(X^2) - E^2(X) \ , \end{aligned}$$

where the second equality follows from the linearity of the mean operator and the fact that $E(X)$ and $E^2(X)$ are constants.

4.1.4 Example: two stage survival

Consider the following hypothetical example of a lung cancer study in which all patients start in phase 1, transition into phase 2, and die at the end of phase 2. Unfortunately, but inevitably, all people die. Biostatistics is often concerned with studying approaches that could prolong or improve life. We assume the length of phase 1 is random and is well modeled by an exponential distribution with mean of five years. Similarly, the length of phase 2 is random and can be modeled by a Gamma distribution with parameters $\alpha = 5$ and $\beta = 4$. Suppose that a new drug that can be administered at the beginning of phase 1 increases 3 times the length of phase 1 and 1.5 times the length of phase 2. Consider a person who today is healthy, is diagnosed with phase 1 lung cancer in 2 years, and is immediately administered the new treatment. We would like to calculate the expected value of the survival time for this person. Denote by X the time from entering in phase 1 to entering phase 2 and by Y the time from entering phase 2 to death without taking treatment. Thus, the total survival time is $2 + 3X + 1.5Y$ and the expected total survival time, in years, is

$$E(2 + 3X + 1.5Y) = 2 + 3E(X) + 1.5E(Y) = 2 + 3 \times 3 + 1.5 \times 5/4 = 12.875 \ .$$

We emphasize here, and throughout the book, that the hardest thing about Biostatistics is to understand the specific scientific language associated with the problem and to translate it into precise statements that can be coded, modeled, and addressed. Human language is often not precise, which requires the biostatistician to think carefully about what is said, what is not said, what pieces of information are useful, what can be discarded, and what additional information is needed. All of this requires thinking and understanding, which is the hardest component of biostatistical work. Probably the best way of learning these skills is to work on multiple projects that raise different types of challenges, contain different data types, and require answering different scientific problems.

4.2 Sample mean and bias

We have discussed the theoretical mean of a theoretical distribution and how to do the necessary calculations. However, in practice we do not know either the true mean or the true distribution from which observations are drawn. Indeed, consider the example of a study in which 10 individuals with myocardial infarction (MI) are followed until death and the time from MI diagnosis to death is recorded for each one of them in years. Suppose that data look like this

$$5,\ 4.5,\ 3.3,\ 7,\ 2,\ 1.7,\ 4.1,\ 6.2,\ 2.4,\ 8.3\ .$$

The average (or sample mean) is 4.45 years, which seems to provide a good description of the observed data, or at least its central tendency. This is called the sample mean, which is easy to calculate for the observed data and provides a simple one-number summary of the collected data. So, why do biostatisticians insist on talking about the true mean, an arbitrary unknown theoretical quantity? There are many reasons, but here we only discuss three: generalizability, sampling assumptions, and exchangeability. Because these are fundamental and deeply complementary concepts in biostatistics, we will take the time to talk about them in some detail. These concepts should underpin the foundation of any self-respecting data analysis discipline and will show just how inevitable the biostatistical approach is.

After collecting the data, calculating the empirical mean, 4.45, was almost instinctive. But, the biostatistician needs to think about how to interpret the number and especially why the mean was calculated. Indeed, if another study is run and data are collected for another 10 subjects then, almost surely, the sample mean will be different. Actually, any biostatistician would take a 100 to 1 bet that the sample mean will not be identical. Make that 1000 to 1 while you are at it. Since the sample mean in this new sample will be different, is there any value in collecting the above sample and calculating 4.45? We instinctively think so and expect that the average survival time in the new sample will be close to 4.45. Of course, we do not know how close and might be tempted to say that the new observed mean will be between, say 4.25 and 4.65. Any self-respecting researcher should, however, be able to put his or her money down and bet on his or her prediction about future results. Generalizability is the property of the scientific approach to produce generalizable knowledge (e.g., the empirical mean in a new study will be close to 4.45). We have already seen that the empirical mean is not generalizable because its inherent randomness makes it irreproducible in the exact sense of the word. However, we will see that the information contained in the sample mean is reproducible in the context of biostatistical modeling when we assume that there exists a theoretical mean and a theoretical distribution.

Sampling assumptions are inherently connected to generalizability of results. Assume, for example, that we are now told that the 10 individuals who had MI are all female from 30 to 35 years old. Thus, we would expect that the

sample mean 4.45 would contain a lot more information about young female MI survivors than old male MI survivors. Thus, the population one samples from and the way sampling is conducted (e.g., at random, clustered, convenience) can have a fundamental effect on the generalizability of the findings. Defining a theoretical distribution in simple but precise terms provides the framework for conducting and conceptualizing experiments.

Exchangeability is one of the least recognized and most pervasive assumptions in data analysis. Intuitively, exchangeability is the property that, up to certain characteristics, experimental units are equivalent. For example, in the MI survivors example, exchangeability requires that future MI survival times behave like the ones already observed. Exchangeability is, thus, strongly related to generalizability and sampling assumptions. In the next Chapter we will introduce the concept of independent and identically distributed (iid) random variables, which is the simplest theoretical framework for generalizability, sampling assumptions, and exchangeability.

Let X_i for $i = 1, \ldots, n$ be a collection of random variables, each from a distribution with mean μ. The sample mean is defined as

$$\overline{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i.$$

Using the rules of expectation, it follows immediately that

$$E(\overline{X}_n) = \frac{1}{n}E\left\{\sum_{i=1}^{n} X_i\right\} = \frac{1}{n}\sum_{i=1}^{n} E(X_i) = \frac{n\mu}{n} = \mu.$$

Therefore, the expected value of the sample mean is the population mean that it is trying to estimate for any sequence of random variables with the same mean. Any function of the data is called an estimator and any unknown quantity μ is called a parameter or estimand. For example, both X_3 and $\overline{X}_n$ are estimators and μ is a parameter. We say that a function of the data $U(X_1, \ldots, X_n)$ is an unbiased estimator of the parameter μ if

$$E\{U(X_1, \ldots, X_n)\} = \mu.$$

For any estimator $U(X_1, \ldots, X_n)$ and parameter μ the bias is defined as

$$b(U, \mu) = E\{U(X_1, \ldots, X_n)\} - \mu\ .$$

By definition, the estimator $U(X_1, \ldots, X_n)$ is called unbiased for μ if $b(U, \mu) = 0$. Given the result above we conclude that the sample mean, $\overline{X}_n$, is an unbiased estimator of the true mean μ. Recall that after running the experiment we can calculate the sample mean, which will contain information about the true mean. However, the true mean is not available to us, just the observations from the experiment. Being unbiased is a good thing for an estimator. Consider the case when one is interested in measuring somebody's weight. If the scale consistently shows two additional kilograms to every person who steps on the scale, the scale is biased for the true weight of the person, μ.

4.2.1 Using R to understand the empirical mean

Let us play some games and see whether we can uncover some order from randomness. We start by simulating the mean of n die rolls, where n can be 5, 10, 20, or 100. Each experiment is repeated 1000 times; for example, for $n = 5$ we roll the die 5 times, record the outcome of each die roll, take the average of the 5 die rolls, store the average, and then repeat the same experiment 1000 times. The result of this experiment will be a vector of length 1000 with each entry representing the average of 5 die rolls. Similar experiments are run for the average of 10 die rolls, and so on. Note that the experiments will require a different number of die rolls, but the results of all experiments will be a vector of length 1000. For example, for the experiment with the average of $n = 5$ die rolls a total of 5000 rolls will be required, whereas for $n = 20$ a total of 20000 rolls will be required. Such experiments would be impractical, but are basically instantaneous using a computer, as we describe below

```
set.seed(7654098)
mx5 <- rep(0, 1000)             #means of n=  5 die rolls
mx10=mx5                        #means of n= 10 die rolls
mx20=mx5                        #means of n= 20 die rolls
mx100=mx5                       #means of n=100 die rolls
for ( i in 1:1000 )
  {#begin rolling the die
  #roll the die n=5 times, take the mean
  mx5[i]   <- mean(sample(1:6,5,replace=T))
  mx10[i]  <- mean(sample(1:6,10,replace=T))
  mx20[i]  <- mean(sample(1:6,20,replace=T))
  #roll the die n=100 times, take the mean
  mx100[i] <- mean(sample(1:6,100,replace=T))
  }#end rolling the die
```

Figure 4.4 displays the histograms of the sample means of n die rolls. Each panel contains the histogram of 1000 realizations of the mean of n die rolls, where n is 5, 10, 20, 100. Each histogram can be viewed as the empirical distribution of the mean over a fixed number of samples, n.

This is a case when we know that the mean is 3.5, which is roughly the center of each of the four histograms. The spread of the empirical means around the true theoretical mean for all numbers of rolls indicates that the empirical means are seldom equal to the theoretical means. As the number of rolls that are averaged increases (from 5 to 100), the distribution of the means becomes tighter and tighter around the true mean. In this case the histograms of the empirical means are symmetric and relatively bell-shaped. The four panels in Figure 4.4 illustrate the manifestation of two very powerful concepts in biostatistics: the law of large numbers and the central limit theorem. We will discuss these results in detail later, but the ever tighter distribution around the true mean is the law of large numbers, while the bell-shape of the distribution of the

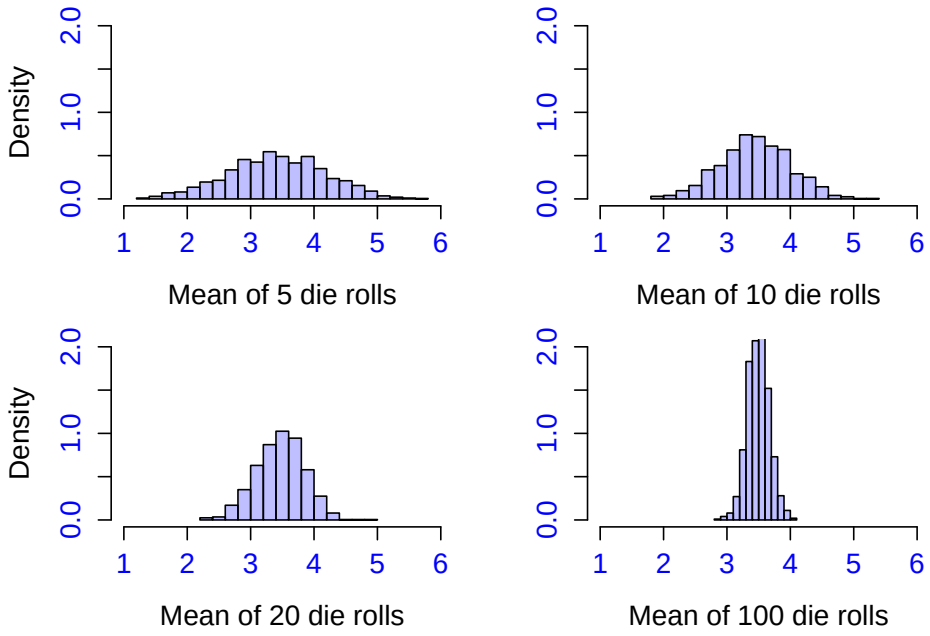


Figure 4.4: Histograms of the sample means for different number of experiments.

average for larger numbers of rolls is the central limit theorem. Teaching these concepts is extremely difficult, though a thorough understanding of the sampling mechanisms described in this section should provide the much needed intuition.

4.2.2 Example: BMI and RDI

We consider now an example from the Sleep Heart Health Study (SHHS). The SHHS is a multicenter study on sleep-disordered breathing, hypertension, and cardiovascular disease (Quan et al. 1997). The data in this book were downloaded from the National Sleep Research Resource https://sleepdata.org/datasets and were approved for the use in this book. The SHHS drew on the resources of existing, well-characterized, epidemiologic cohorts, and conducted further data collection, including measurements of sleep and breathing. In this book we will analyze a subset of the SHHS visit 1 data, which contain 5804 participants and included 52.4% women and 47.6% men. We start by downloading the data

```
file.name = file.path("data", "shhs1.txt")
data.cv<-read.table(file=file.name,header = TRUE,na.strings="NA")
attach(data.cv)
dim(data.cv)
```

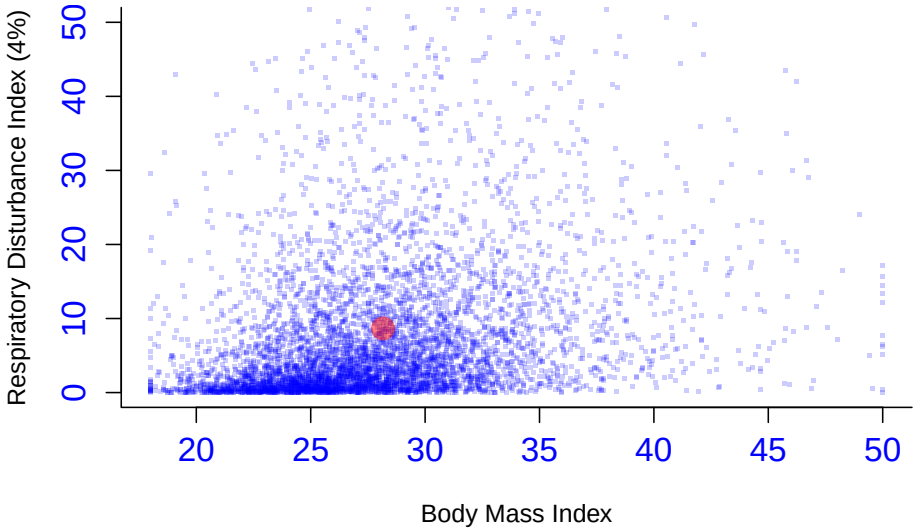


Figure 4.5: Scatter plot of BMI versus RDI in the SHHS.

```
[1] 5804   30
```

The dataset contains 5804 rows, where each row corresponds to a person enrolled in the SHHS, and 30 columns, where each column, except `pptid`, which is used as an identifier, corresponds to a person-specific variable. For now we will focus on the Body Mass Index (BMI) and Respiratory Disturbance Index (RDI). More precisely, we will work with the variables `bmi_s1` and `rdi4p`, the overall RDI at 4% oxygen desaturation. This is the ratio of the count of all apneas and hypopneas associated with at least a 4% oxygen desaturation to the total sleep time expressed in hours (https://sleepdata.org/). The 4% oxygen desaturation refers to blood's oxygen level drop by 4% from baseline. This variable is often used to characterize the severity of sleep apnea, with larger values corresponding to worse outcomes.

Figure 4.5 provides the scatter plot of BMI versus RDI, though, for presentation purposes, we only show RDI less than 50. Probably one of the best things to do when looking at a new dataset is to plot the data and identify some of their characteristics. This plot indicates that RDI$\geq$ 0 (by definition), that there is a higher density of observations for small RDI (note the deeper shades of blue close to the x-axis) and that most people in the SHHS sample have a BMI$\geq$ 20. The casual observer should note the large amount of variability in this plot, as for any BMI level there is a large range of observed RDI values.

We also calculate the mean BMI and RDI

```
round(mean(bmi_s1,na.rm=TRUE),digits=2)
```

```
[1] 28.16
```

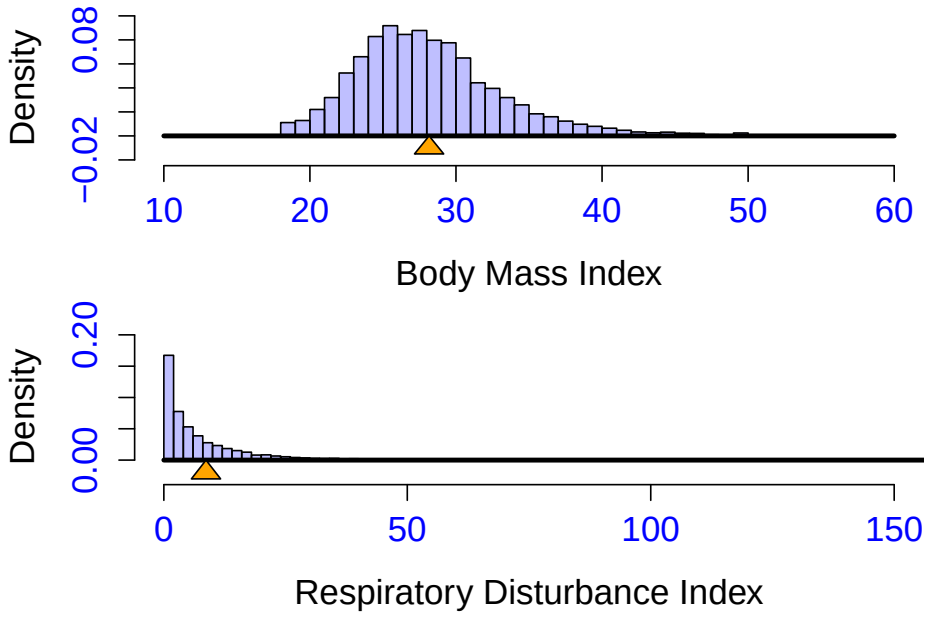


Figure 4.6: Marginal distributions of BMI and RDI in the SHHS.

```
round(mean(rdi4p,na.rm=TRUE),digits=2)
```

```
[1] 8.66
```

The center of the red circle has the x-axis coordinate equal to the empirical mean BMI (28.16) and y-axis coordinate equal to the empirical mean RDI (8.66). This is the mean of the two dimensional vector of observations $(\text{BMI}_i, \text{RDI}_i)$ for $i = 1, \ldots, 5804$. We can look more closely at each variable separately.

The two panels in Figure 4.6 provide the histograms of the two observed variables and the means indicated by orange triangles. The BMI histogram is slightly skewed with a heavier right tail, which pulls the mean closer to the upper range of the BMI range relative to the mode of the distribution (point of highest density of BMI values). The RDI plot is much more skewed and is, in fact, very hard to represent well. The reason is that it contains some very large RDI values. Indeed, there are 119 individuals with RDI$>$ 50 (2.1%) and 4 individuals with RDI$>$ 100 (0.07%). Moreover, there are 1264 individuals with an RDI$<$ 1 (21.8%), which accounts for the mode of the distribution being close to 0. It is a good exercise to calculate these numbers from the data directly using `R`. The distribution shown in Figure 4.5 is called the joint empirical distribution of BMI and RDI, whereas the two distributions in Figure 4.6 are the marginal empirical distributions of of BMI and RDI, respectively.

The parallel between these examples and the seesaw example (Figure 4.1) for

discrete distributions should not be lost. Indeed, one can imagine that we assign a weight to every bin of the histogram proportional to the number of subjects in that particular bin. Then the seesaw would be in perfect balance if it rested on the tip of the orange triangle.

This example indicates just how easy and intuitive it is to work with the data, calculate the mean, and produce histograms or other summaries. However, as before, we are interested in understanding what information from these figures is generalizable. For example, is the average BMI of 28.16 in the sample equal to the average BMI of the US population? As before, we are 100% sure that this is not the case. Moreover, we know that the SHHS sample is not representative of the U.S. population. For example, participants less than 65 years of age were over-sampled on self-reported snoring to augment the prevalence of Sleep Disrupted Breathing (SDB). Implicitly, this sampling procedure over-sampled individuals with higher BMI, which might have induce bias. Indeed, according to the NHANES study study, which took place in the US for a period comparable to the SHHS enrollment, the average adult man has a BMI of 26.6 and the average adult woman has a BMI of 26.5. However, even if we obtained two independent SHHS samples from the US population using identical protocols, the mean BMI would not be identical. They would be close, but not identical. Biostatistical philosophy postulates that the observed mean is an approximation of the unknown mean and that different samples can have different observed (or empirical) means. An important art associated with this philosophy is how to make inference about this unknown mean without collecting one or more additional samples. More precisely, can we make predictions about what the mean could be in a different sample collected with the same protocol without the cost and time associated with the process? This is one of the most practical problems in data analysis that humans have been able to address after several millennia of thinking about data, experiments, and science.

4.2.3 Example: brain

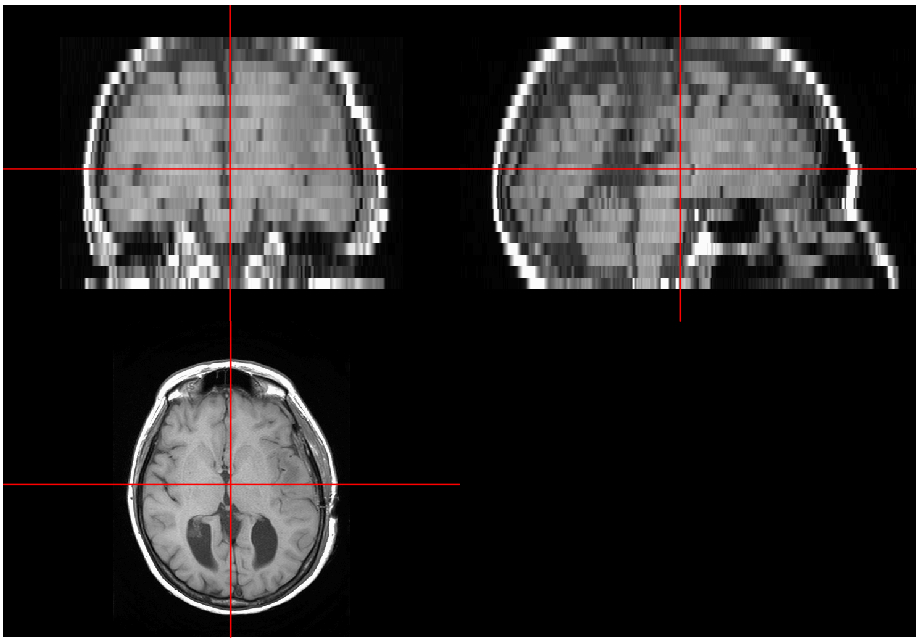
Let us consider another example, this time in three dimensions. The data are a T1-weighted magnetic resonance image (MRI) of a healthy brain from the Kirby21 dataset (Landman et al. 2011). We would like to plot the data and find the center of gravity (cog), or mean, of the image. First we need to understand a little bit about the structure of the data. We use the `R` package `oro.nifti` (Whitcher, Schmid, and Thornton 2011) from Neuroconductor (Muschelli et al. 2019).

```
library(oro.nifti)
file.name = file.path("data/NIfTI", "T1.nii.gz")
nii_T1=nii_T2=readNIfTI(file.name,reorient=FALSE)
dim(nii_T1)
```

```
[1] 512 512  22
```

This indicates that the data are stored in a three dimensional array and the total number of entries is $512 \times 512 \times 22 \approx 5.8$ million. For every entry in the matrix an intensity value is assigned. Many of the entries (3D pixels, or voxels) do not correspond to human tissue and simply contain the T1-weighted value of air, which, in this case is assigned to 0. There are about 2.5 million voxels with an intensity equal to 0 and we would like to identify and mask out these voxels from the data. Indeed, if we do not do that, the mean of the image will be the mean of the bounding box around the brain image, which could be very different from the brain cog. Below we define the image mask, and take the column means of the voxel locations for those voxels that do not have a T1-weighted intensity equal to 0.

```
mask = nii_T1 > 0
inds = which(mask > 0, arr.ind = TRUE)
cog = colMeans(inds[, 1:3])
cog = floor(cog)
nii_T1[ nii_T1 > 500] = 500 # remove
nii_T1 = calibrateImage(nii_T1)
orthographic(nii_T1, xyz = c(cog))
```



This approach does not consider the intensities of the voxels; instead it takes the average of the x, y, z coordinates of those voxels that do not correspond to air. The cross-hair in the orthographic image indicates this mean in the coronal (top-left panel), saggittal (top-right panel), and axial (bottom-left panel) planes, respectively. As the dimension of the data increases from one to three dimensions, we are still able to show the empirical mean. However, as the

number of dimensions increases further, and increase it will, visualizing the data in all dimensions simultaneously becomes problematic. This is why in practice a good first step is to use all possible two-dimensional plots to explore the data. One could ask, of course, why is the cross-hair an empirical mean and what is the theoretical mean? Ideally, we would like the theoretical mean to be a well defined quantity, for example, the center of mass of the brain. However, we do not have the brain, just a pretty blurry image of the head that contains the brain, the skull, and some additional tissues (have you noticed the nose?). Moreover, if we replicate the experiment and we image the brain again, many things would change, including the alignment of voxels, the definition of air, the orientation of the image, the strength of the signal, and the amount of non-brain tissues. All these, and many other changes, lead to a different estimated mean of the brain. Simply acknowledging that the process of identifying (or in biostatistics speak, estimating) the mean is subject to error can help the researcher identify the sources of errors, incorporate them into the inference about the true underlying mean, and help build better processing and analytic pipelines that could address some of the problems noted above.

4.3 Variance, standard deviation, coefficient of variation

The variance of a random variable is a measure of spread of the distribution of that random variable. If X is a random variable with mean μ, the variance of X is defined as

$$\text{Var}(X) = E(X - \mu)^2 \ ,$$

which is the expected squared distance from the mean. If X is a discrete random variable with pmf $p(x) = P(X = x)$, then

$$\text{Var}(X) = \sum_x (x - \mu)^2 p(x) \ .$$

If X is a continuous random variable with pdf $f(x)$, then

$$\text{Var}(X) = \int_{-\infty}^{\infty} (x - \mu)^2 f(x) dx \ .$$

A convenient computational form for the variance is $\text{Var}(X) = E(X^2) - \{E(X)\}^2$, which was proved earlier in this chapter. If a is a constant, then $\text{Var}(aX) = a^2\text{Var}(X)$. Indeed, we already know that $E(aX) = aE(X)$ and

$$\text{Var}(aX) = E\{aX - E(aX)\}^2 = E[a^2\{X - E(X)\}^2] = a^2\text{Var}(X) \ .$$

The standard deviation of a random variable X is defined as the square root of the variance. More precisely,

$$\text{sd}(X) = \{E(X - \mu)^2\}^{1/2} \ .$$

Standard deviation is sometimes preferred in practice because it is expressed in units on the original scale of the data, just like the mean, whereas the variance is expressed in squared units. Moreover, if a is a constant, then $\text{sd}(aX) = a\text{sd}(X)$.

The coefficient of variation of a random variable with mean $E(X) \neq 0$ is defined as

$$\text{cv}(X) = \frac{\text{sd}(X)}{|E(X)|} = \frac{\{E(X-\mu)^2\}^{1/2}}{|\mu|} ,$$

where $|a|$ is the absolute value of a. The coefficient of variation is a unitless measure of the amount of variability (as described by the standard deviation) relative to the mean. If a is a constant, then $\text{cv}(aX) = \text{cv}(X)$, indicating that the coefficient of variation is invariant to the scaling of the random variable. The coefficient of variation is the inverse of the signal-to-noise ratio used in engineering. It has a strong link to Cohen's d and the effect size used in sample size calculations. The coefficient of variation is not defined for $\mu = 0$.

4.3.1 Example: effect of the mean and variance on distributions

Figure 4.7 displays the effect of mean and variance on the shape of the Normal and Gamma distributions. The top-left panel displays the pdfs of the Normal distributions centered at 0 with variance 1 (blue), 4 (orange), and 8 (red), respectively. When the mean is fixed for the Normal distribution and the variance increases, the maximum of the pdf decreases and more probability is allowed to shift into the tails of the distribution. In practice this happens when there is more noise or uncertainty in measurements. The top-right panel displays the pdfs of the Normal distributions with mean 0 and variance 8 (red), with mean 2 and variance 4 (orange), and with mean 4 and variance 1 (blue). The pdfs indicate that the mean affects the location of the distribution and the variance affects the spread. The bottom-left plot displays the Gamma distributions with mean 20 and variance 40 (blue), 80 (orange), and 200 (red), respectively. Compared to the Normal distributions the Gamma distributions are not symmetric and do not remain centered around the mode (location of largest value of the pdf). The variance increase is manifested in a lower peak of the pdf, a shift of the peak to the left (smaller values), and a much heavier tail to the right. This indicates that, indeed, for a fixed mean, the variance continues to characterize the spread of the distribution, though in a less intuitive way than in the case of the Normal distribution. The bottom-right plot displays three Gamma distributions with variance 100 and means 31.6 (blue), 22.4 (orange), and 14 (blue); this has the effect of keeping the variance fixed and changing the mean of the Gamma distribution. Just as in the case of the Normal distribution, as the mean decreases, the mode of the distribution shifts to the left. However, in contrast to the Normal distribution, there are also pronounced changes in the shape of the pdf. This happens because the mean and variance parameters in

the Gamma distribution are linked, whereas they are not for the Normal distribution. One should remember that the intuition about the mean and variance as location and spread of the distribution works especially well for symmetric distributions. The intuition holds for other distributions, but the link is more nuanced.

```
x=seq(-10,10,length=201)
par(mfrow = c(2,2),
          oma = c(1,1,1,1) + 0.1,
          mar = c(5,5,0,0) + 0.1)
y1<-dnorm(x)
y2<-dnorm(x,0,2)
y3<-dnorm(x,0,4)
plot(x,y1,type="l",col="blue",lwd=3,
     xlab="Same mean Normal",ylab="Density",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(x,y2,col="orange",lwd=3)
lines(x,y3,col="red",lwd=3)

y1<-dnorm(x,4,1)
y2<-dnorm(x,2,2)
y3<-dnorm(x,0,4)
plot(x,y1,type="l",col="blue",lwd=3,
     xlab="Different mean/Var Normal",ylab="",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue",main="")
lines(x,y2,col="orange",lwd=3)
lines(x,y3,col="red",lwd=3)

x=seq(0,80,length=101)
y1<-dgamma(x,shape=10,scale=2)
y2<-dgamma(x,shape=5,scale=4)
y3<-dgamma(x,shape=2,scale=10)
plot(x,y1,type="l",col="blue",lwd=3,
     xlab="Same mean Gamma",ylab="Density",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(x,y2,col="orange",lwd=3)
lines(x,y3,col="red",lwd=3)

x=seq(0,80,length=101)
y1<-dgamma(x,shape=10,scale=sqrt(10))
y2<-dgamma(x,shape=5,scale=sqrt(20))
y3<-dgamma(x ,shape=2,scale=sqrt(50))
plot(x,y1,type="l",col="blue",lwd=3,
     ylim=c(0,0.06),xlab="Same Var Gamma",ylab="",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(x,y2,col="orange",lwd=3)
```

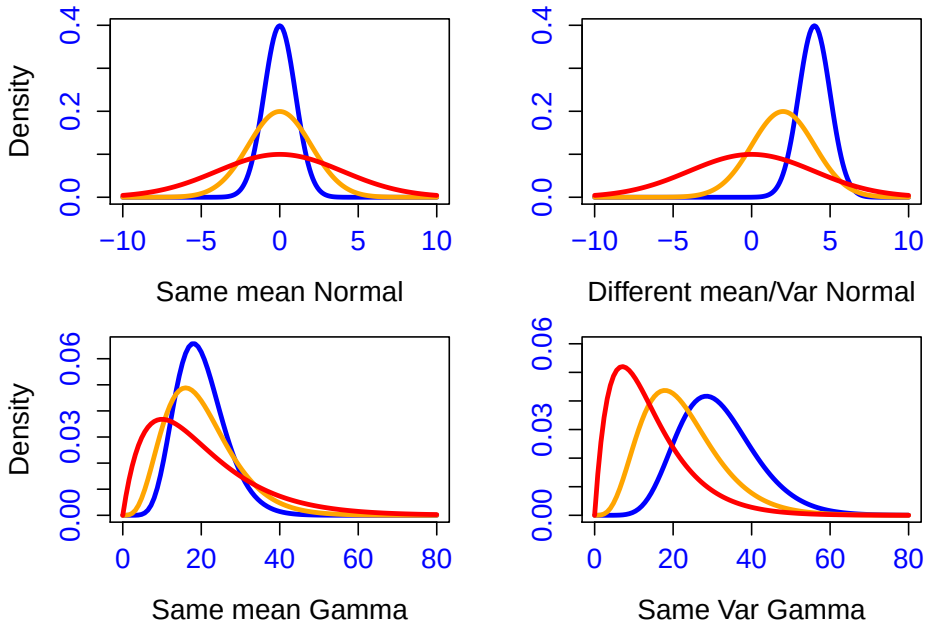


Figure 4.7: Effect of parameters on the shape and location of the Normal and Gamma distributions.

```
lines(x,y3,col="red",lwd=3)
```

4.3.2 Example: to be

Consider the case of the Bernoulli random variable, the most Shakespearean of random variables. Indeed, no other Biostatistical concept represents better the "To be, or not to be" phrase of Prince Hamlet. In this example, assume that the probability of being alive (to be) five years after stage IA lung cancer diagnosis is p. We denote by X the "to be/not to be" outcome of the experiment of waiting for five years to see whether the person survives. In this case $X \sim \text{Bernoulli}(p)$. We know that $E(X) = p$ and we obtain the variance as

$$\text{Var}(X) = E(X^2) - \{E(X)\}^2 = E(X) - p^2 = p - p^2 = p(1-p) \ .$$

The second equality holds because the random variable X takes only the values 0 and 1, which implies that $X^2 = X$. The coefficient of variation for a Bernoulli random variable is $\text{cv}(X) = \sqrt{(1-p)/p}$.

Consider the case of a random variable Y that takes values $0 \leq Y \leq 1$ and has the mean $E(X) = p$. Because $Y^2 \leq Y$ it follows that $E(Y^2) \leq E(Y) = p$, which

implies that

$$\text{Var}(Y) = E(X^2) - \{E(X)\}^2 \leq p - p^2 = p(1-p) \ .$$

Therefore, for any variable with mean p the variance of the Bernoulli random variable is the largest among random variables bounded by 0 and 1. For example, for every Beta random variable there exists a Bernoulli random variable with the same mean and larger variance Moreover, for $p = 1/2$ the variance of the Bernoulli random variable is $1/2 - 1/4 = 1/4$. Since $(p - 1/2)^2 \geq 0$ it follows that $p^2 - p + 1/4 \geq 0$, which is equivalent to $p(1-p) \leq \frac{1}{4}$. Thus, the largest variance for a random variable Y that takes values in $[0, 1]$ is $1/4$ and is achieved for a Bernoulli$(1/2)$ random variable.

4.4 Variance interpretation: Chebyshev's inequality

The mean and the variance of the distribution are extremely important in practice as they provide a simple two-number characterization of very complex objects, such as distributions. Here we will start to lay the foundations for using these quantities to conduct inference about the unknown mean. For this, we will introduce and prove Chebyshev's inequality. This will help us prove the law of large numbers (which, essentially, states that the empirical mean does not change much after a practical number of observations) and it will provide a way of constructing confidence intervals for the true, unknown, mean based on one sample. Chebyshev's inequality states that for any random variable X with mean μ and variance σ^2

$$P(|X - \mu| \geq k\sigma) \leq \frac{1}{k^2} \ .$$

We first prove this result and then discuss the intuition and its uses. If $f(x)$ is the pdf of X, by definition,

$$P(|X - \mu| \geq k\sigma) = \int_{\{x:|x-\mu|>k\sigma\}} 1 \times f(x)dx \ .$$

Since for every $\{x : |x - \mu| > k\sigma\}$ we have $(x-\mu)^2/k^2\sigma^2 \geq 1$ it follows that

$$\int_{\{x:|x-\mu|>k\sigma\}} 1 \times f(x)dx \leq \int_{\{x:|x-\mu|>k\sigma\}} \frac{(x-\mu)^2}{k^2\sigma^2} \times f(x)dx \ .$$

Because the function being integrated is positive it follows that integrating over all x will further increase the integral, that is

$$\int_{\{x:|x-\mu|>k\sigma\}} \frac{(x-\mu)^2}{k^2\sigma^2} \times f(x)dx \leq \int \frac{(x-\mu)^2}{k^2\sigma^2} \times f(x)dx \ ,$$

where the lack of indication over what domain the integration is done means that it is done over the entire domain. Because k and σ are constants it follows that

$$\int \frac{(x-\mu)^2}{k^2\sigma^2} \times f(x)dx = \frac{1}{k^2\sigma^2}\int (x-\mu)^2 f(x)dx = \frac{\text{Var}(X)}{k^2\sigma^2} = \frac{\sigma^2}{k^2\sigma^2} = \frac{1}{k^2}\ .$$

This type of inequality is very useful in biostatistics as it provides the link between the observation X, the unknown mean μ, and the standard deviation σ. When read carefully, Chebyshev's inequality states that the probability that the observed value is more than $k\sigma$ away from its mean is small and rapidly decreasing with k. The inequality can be re-written as

$$P(X - k\sigma < \mu < X + k\sigma) > 1 - \frac{1}{k^2}\ ,$$

which indicates that the *the true mean of the variable X is at most $k\sigma$ away from the observation X with probability at least $1 - 1/k^2$*. This is in line with the philosophy we have outlined. While the observation does not tell us exactly what the true mean is or what the empirical mean will be in a future experiment, it does provide a lot of useful information. More precisely, it allows us to build confidence intervals for the mean of a random variable. For example, for $k = 4.47$ there is at least a 95% chance that the interval $(X - 4.47\sigma, X + 4.47\sigma)$ covers the true mean μ. Assuming for a moment that σ is known (it typically is not and will be estimated in practice) this interval can be calculated from the data.

4.4.1 Chebyshev versus parametric assumptions

The confidence interval derived from Chebyshev's inequality is typically conservative (too long, it covers the true value with more probability than the stated probability). This happens because its derivation makes no assumption about the distribution of X. If some knowledge is available about the distribution of X, then the length of the confidence interval can be substantially reduced. For example, if we know that the distribution of X is Normal, then there is a 95% chance that the interval $(X - 1.96\sigma, X + 1.96\sigma)$ covers the true mean μ. This interval has a length that is 43% of the length of the Chebyshev confidence interval for the same level of confidence (95%). Thus, having information about the shape of the distribution can dramatically reduce the length of the confidence intervals (and p-values; more about this later.)

We are interested in calculating the $P(|X - \mu| \geq k\sigma)$ for different values of k (number of sigmas away from the mean). We consider several values for $k\sigma \in \{2, 3, 4, 5\}$. If we make no assumption about the distribution then the Chebyshev's inequality leads to the following upper limits

```
k=2:5                                   #Multiples of SD
round(1/k^2,digits=3)                   #Chebyshev upper limit
```

```
[1] 0.250 0.111 0.062 0.040
```

For a Normal random variable $N(\mu, \sigma)$ it is easy to show that $P(|X - \mu|/\sigma > k) = 2\{1 - \Phi(k)\}$, where $\Phi(k)$ is the cdf of a $N(0, 1)$ random variable. Thus, the probability of exceeding $k\sigma$ is

```
2*(1-pnorm(k))                                  #Probability for a Normal distribution
```

```
[1] 4.550026e-02 2.699796e-03 6.334248e-05 5.733031e-07
```

Consider now a t distribution with 3 degrees of freedom, denoted as $t(3)$. This distribution has not been introduced, but we will work with it nonetheless. The standard deviation of the $t(3)$ distribution is $\sqrt{3}$ and the probability of exceeding $k\sqrt{3}$ by a variable distributed as $t(3)$ is $2\{1 - F_{t(3)}(k\sqrt{3})\}$, where $F_{t(3)}(\cdot)$ is the cdf of the $t(3)$ distribution. This can be calculated as

```
sdt3=sqrt(3)                                    #SD of a t(3) distribution
round(2*(1-pt(k*sdt3,df=3)),digits=3)           #Probability for a t(3) distribution
```

```
[1] 0.041 0.014 0.006 0.003
```

For the Gamma$(2, 1/2)$ distribution the mean is $\mu = 2 * 2 = 4$ and the standard deviation is $\sigma = \sqrt{2 * 2^2} \approx 2.83$. We can write

$$P(|X-\mu| \geq k\sigma) = P(X \leq \mu-k\sigma)+P(X \geq \mu+k\sigma) = F_{\Gamma(2,2)}(\mu-k\sigma)+1-F_{\Gamma(2,2)}(\mu+k\sigma),$$

where $F_{\Gamma(2,2)}(\cdot)$ is the cdf of the Gamma distribution with parameters $(2, 1/2)$.

```
sh=2                                            #Shape parameter of Gamma(sh,sc)
sc=2                                            #Scale parameter of Gamma(sh,sc)
m=sh*sc                                         #Mean of Gamma(sh,sc)
sdg = sqrt(sh*sc^2)                             #SD of Gamma(sh,sc)
pgam=pgamma(m-k*sdg,shape=sh,scale=sc)+1-pgamma(m+k*sdg,shape=sh,scale=sc)
round(pgam,digits=3)                            #Probability for a Gamma(2,2) distribution
```

```
[1] 0.047 0.014 0.004 0.001
```

For convenience we collect these results in the table below. This table provides the probability of being at least $k\sigma$ away from the mean in a particular experiment as a function of the type of assumption about the distribution (by row) and by the size of the deviation (by column).

$k\sigma$	2	3	4	5
Chebyshev	0.250	0.111	0.063	0.040
$t(3)$	0.041	0.014	0.006	0.003
Gamma(2,1/2)	0.046	0.014	0.004	0.001
Normal	0.046	0.003	6.3e-5	5.7e-7

The probability of exceeding a certain threshold $k\sigma$ is much larger using the

Chebyshev upper bound, as the inequality holds for any random variable, including random variables with very heavy tails and heavy skew. While the $t(3)$ distribution is heavy-tailed, there are other distributions that have even heavier tails. The Normal distribution tends to have the smaller probabilities of exceeding a particular multiple of the standard deviation, though for 2 standard deviations it does not seem to very much matter whether one uses the Normal, Gamma(2,2) or $t(3)$ distribution. Differences become much more pronounced for higher values of the multiplier, k. This is due to the much thinner tail of the Normal (the tail is dominated by $\exp(-x^2/2)$), which converges very quickly to zero. Distributions with slower tail decays than the Normal, such as t(3) and Gamma(2,2), are called heavy-tail distributions.

4.4.2 Example: IQ

IQs are often said to be distributed with a mean of 100 and a standard deviation of 15. We are interested in quantifying the probability that a random person has an IQ higher than 160 or below 40. Thus, we want to know the probability of a person being more than 4 standard deviations from the mean. Chebyshev's inequality suggests that this will be no larger than 6% (see table above). However, IQ distributions are often cited as being bell shaped, in which case this bound is very conservative. Indeed, the probability of a random draw from a Normal distribution being 4 standard deviations from the mean is 6.3×10^{-5} (6 one thousandth of one percent). These differences in probability are extreme and show just how much is being gained by having some idea about the underlying distribution of the random variable.

4.5 Supplementary R training

This part of the book can be used as laboratory material or as additional material for learning R. We start by looking in more detail at the Normal distribution. We will also learn how to write simple functions. We start by generating a random sample of size $n = 1000$ and plot it as a function of observation number.

```
## set the seed
set.seed(33)
## Generate 1000 observations from a Normal with mean 2 and variance 1
x <- rnorm(1000, mean = 2, sd = 1)
## plot observations as a function of observation number
plot(x,col="blue",pch=20,lwd=3,xlab="Observation number",
     ylab="Random sample from N(2,1)",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

Figure 4.8 is what we expect from a random sample from a Normal distribution and it looks like symmetric white noise around the theoretical mean $\mu = 2$ with a range roughly between -1 and 5, about 3 standard deviations away from the

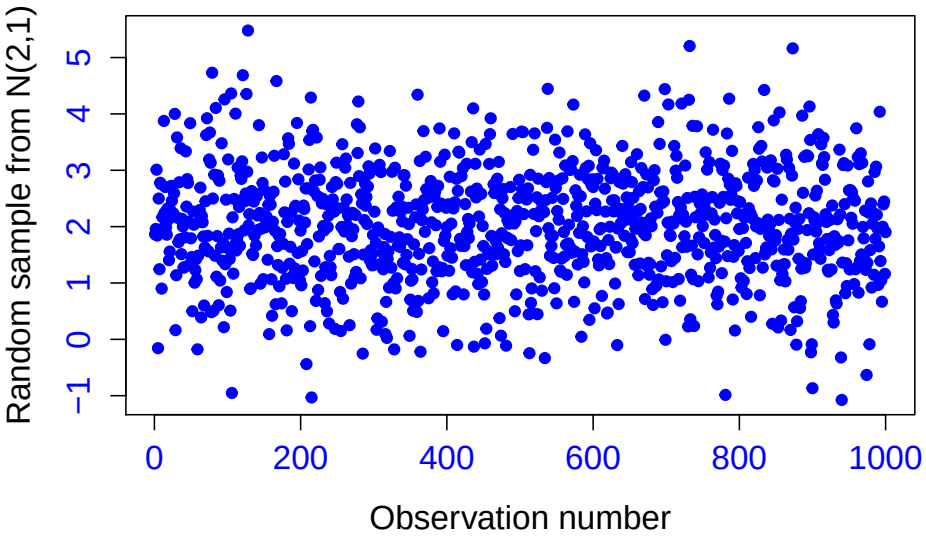


Figure 4.8: Illustration of the scatter plot for a N(2,1) by the index number of the sample.

mean in either direction. In general, an inspection of the data as a function of observation number could provide quick insights into the distribution of the data, possible structure, or unusual observations. There are many ways to visually explore the data. For example, histograms and QQ-plots provide visual displays of the distribution and their comparison with hypothetical theoretical distributions. Figure 4.9 displays the same data as Figure 4.8, but in the form of a histogram. The histogram is reasonably symmetric and could be thought of as a having a quasi bell-shape, though even for $n = 1000$ the match is not perfect. This is due to sampling variability, which is biostatistics speak for the differences between theoretical distribution and observed data and between observed data in two different samples.

```
## make a histogram of the observations
hist(x,probability=T,col=rgb(0,0,1,1/4),breaks=30,xlab="Sample N(2,1)",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue",main="")
```

Figure 4.10 displays the quantile-quantile (QQ) plot for the same sample used in Figures 4.8 and 4.9. The QQ plot displays the empirical quantiles of the observed data relative to the theoretical quantiles of a $N(0,1)$ random variable. Because the quantiles of any Normal random variable are linear transformations of those of a standard Normal, one would expect the plot of empirical versus theoretical quantiles to be a line. The line is not the line of identity, except when the samples are drawn from the $N(0,1)$ distribution. In general QQ-plots can be obtained between any two distributions. However, when only the first argument of the function is given, `qqnorm(x)`, then quantiles are shown versus

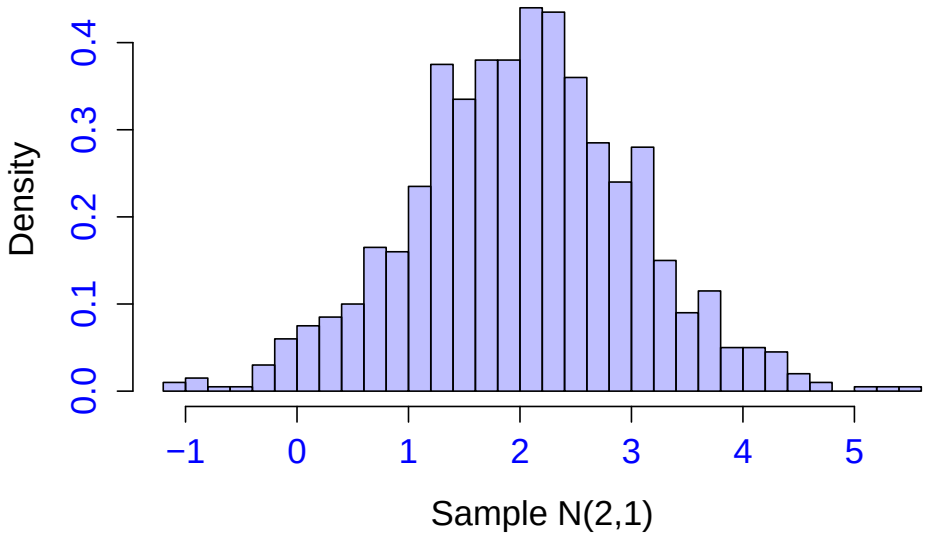


Figure 4.9: Histogram of 1000 samples from a N(2,1) distribution.

the $N(0,1)$ quantiles. The plot shows that there is very good agreement between the theoretical and empirical quantiles, with some exception for the lower quantiles of the distribution. These quantiles are harder to estimate in finite samples, which means that they are more variable. While empirical quantiles have not been yet introduced, we think that it is important to start using the concepts early on. Fear not for we will introduce these concepts in due time.

```
## compare the quantiles of our sample to that of a Normal distribution
qqnorm(x,cex.lab=1.3,cex.axis=1.3,col.axis="blue",pch=20,lwd=3,col="blue")
```

We focus now on the theoretical Normal distribution. Figures 4.11 and 4.12 display the pdf and cdf of a $N(0,1)$ distribution, respectively.

```
# create a sequence from -5 to -5
y <- seq(-5, 5, by = .01)
## plot the pdf of a standard Normal with dnorm
plot(y, dnorm(y, mean = 0, sd = 1), type = 'l',lwd=3,col="blue",ylab="",
     xlab="pdf of N(0,1)",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

```
##plot the cdf of a standard normal with pnorm
plot(y, pnorm(y, mean = 0, sd = 1), type = 'l',lwd=3,col="blue",ylab="",
     xlab="cdf of N(0,1)",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

The quantiles of the Normal distribution can be obtained in R as

```
##the qnorm function is the inverse of the pnorm function
round(qnorm(c(0.25,.5,0.75)),digits=4)
```

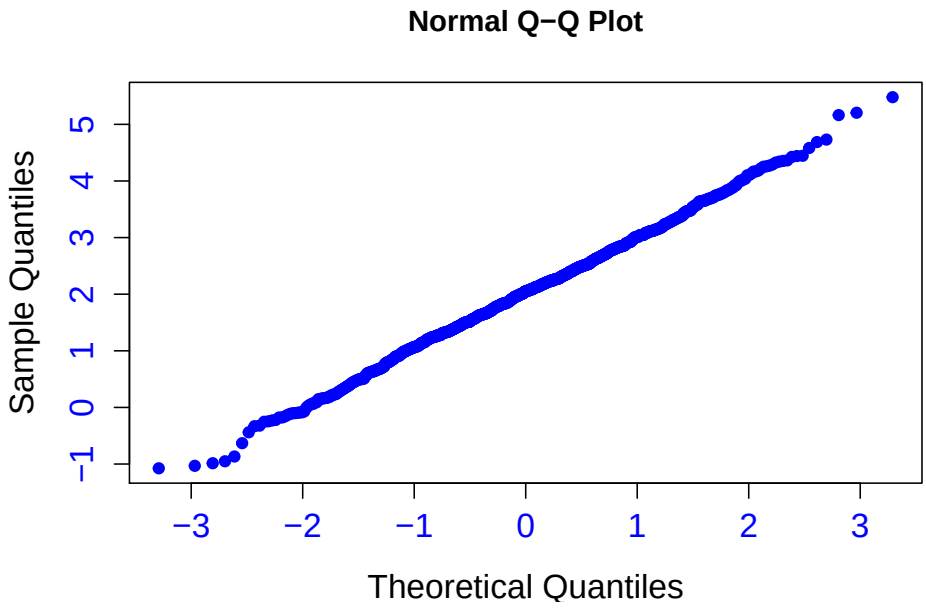


Figure 4.10: Quantile-quantile plot for a sample of size 1000 from a N(2,1) distribution.

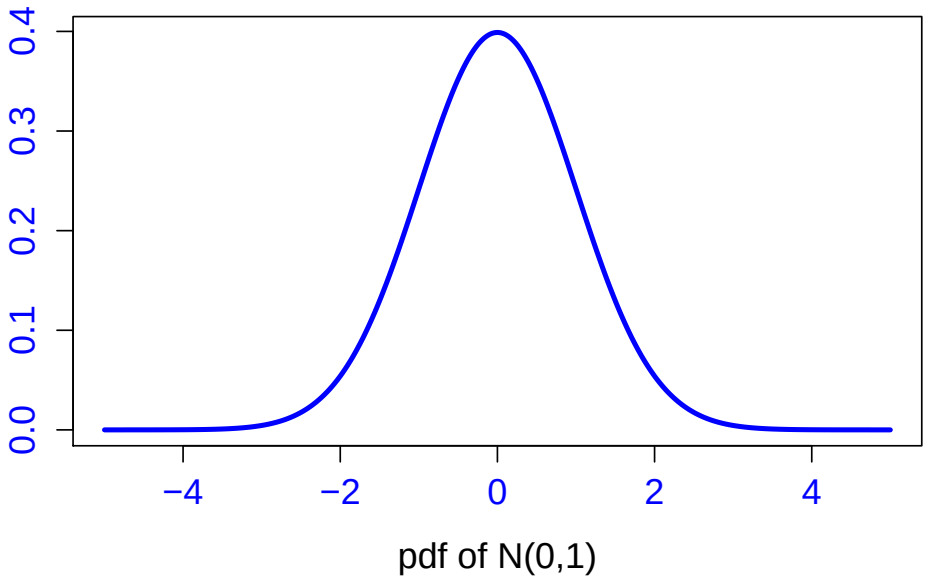


Figure 4.11: Theoretical pdf of a N(0,1) distribution.

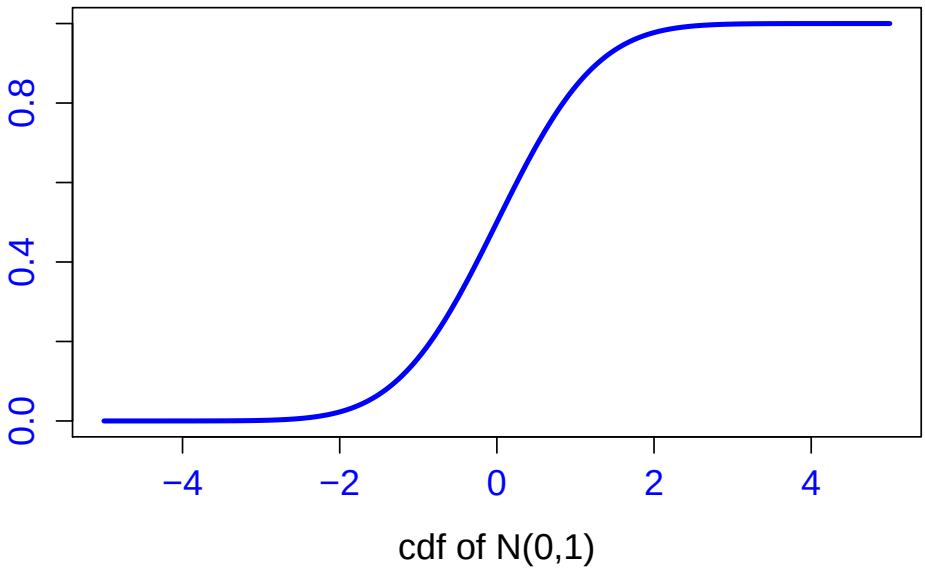


Figure 4.12: Theoretical cdf of a N(0,1) distribution.

```
[1] -0.6745  0.0000  0.6745
```

We now show how to calculate and plot the likelihood for the Normal distribution. To do this we will start discussing functions in R, which are created for specific tasks. Below we show how to create a function that takes as input a vector of data, in this case x, the 1000 dimensional vector of simulated independent $N(1, 2)$ random variables, and mu, a scalar where the log-likelihood is evaluated. The output is the value of the log likelihood and is returned using return(results). A function can return vectors, matrices, data frames and lists when the output is more complex.

```
#create the log likelihood function
logL <- function(data, mu){
  vec <- -1*(data-mu)^2/(2*1)
  result <- sum(vec)
  return(result)}

#compute log likelihood of values of mu ranging from -10 to 10
mu.vals <- seq(-10,10,0.1)
logL.vals <- vector(length=length(mu.vals))
for(i in 1:length(mu.vals))
{logL.vals[i] <- logL(data=x, mu=mu.vals[i])}
```

The likelihood was calculated on an equally spaced grid of potential values of the parameter and is stored in the vector logL.vals where each entry corresponds

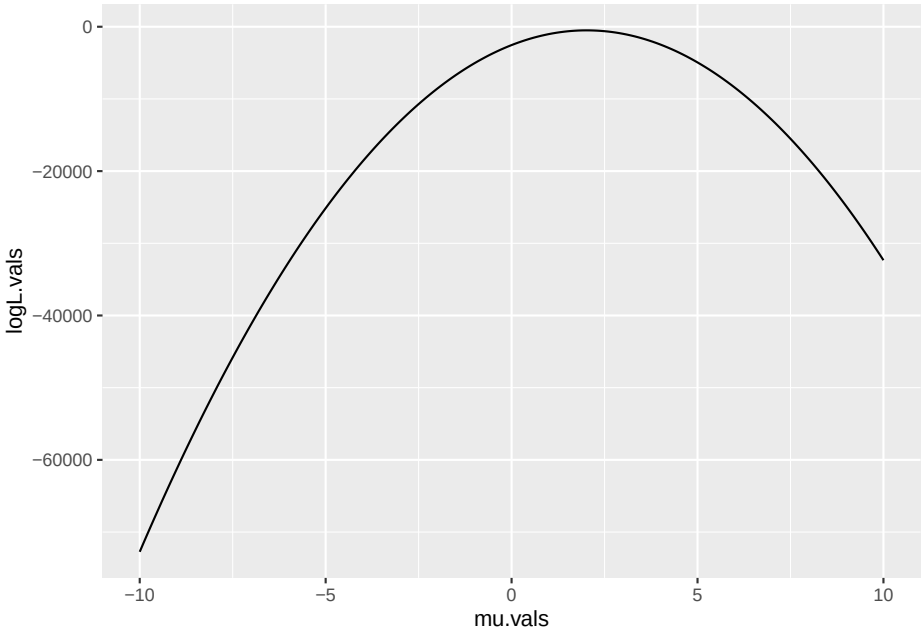


Figure 4.13: Log-likelihood for a sample of 1000 observations from a N(2,1) distribution.

to a grid point between −10 and 10. The next step is to plot the log-likelihood

```
#plot log likelihood
library(ggplot2)
qplot(x=mu.vals, y=logL.vals, geom="line")
```

This log-likelihood function displayed in 4.13 is an inverted parabola, is concave (does not hold water), has a maximum, and is symmetric around the maximum. We would like to know exactly where the maximum of the likelihood is attained and compare it with the `mean(x)`.

```
#which value maximizes the likelihood?
MLE.mu <- mu.vals[which.max(logL.vals)]
#the analytic MLE
MLE.mu2 <- mean(x)
MLE.mu
```

```
[1] 2
```

```
round(MLE.mu2,digits=3)
```

```
[1] 2.02
```

The two values obtained here numerically are very close, but do not appear

exactly identical. We will show later that, in fact, they are exactly identical. The reason for the discrepancy is that here we are using a grid of points to evaluate the likelihood that is just not fine enough.

We now show that log-likelihoods are numerically much better behaved than likelihoods. The reason is that the value of the likelihood is usually too small to compute. To see this we simply conduct the same exact calculations using the likelihood instead of the log-likelihood functions. We first define the likelihood function `L` as follows

```
L <- function(data, mu){
  vec <- dnorm(x=data, mean = mu, sd = 1)
  result <- prod(vec)
  return(result)}
```

We evaluate the likelihood function at its maximum for 3 values and for 1000 values, respectively. The likelihood function is positive, but very small. Indeed, even when using the first 3 observations the maximum likelihood (evaluated at the true mean function) is 0.038. When $n = 1000$ `R` cannot distinguish between the maximum likelihood and 0. Of course, these are numerical, not theoretical, problems and they can be addressed.

```
round(L(data=x[1:3], mu=2),digits=4)
```

```
[1] 0.0377
```

```
L(data=x, mu=2)
```

```
[1] 0
```

The take home message here is that in biostatistics one often has to carefully understand the numerical implications of the various operations.

4.6 Problems

Problem 1. Using the rules of expectations prove that $\text{Var}(X) = E[X^2] - E[X]^2$ where $\text{Var}(X) = E[(X - \mu)^2]$ for discrete random variables.

Problem 2. Let $g(x) = \pi_1 f_1(x) + \pi_2 f_2(x) + \pi_3 f_3(x)$ where f_1 and f_2 are densities with associated means and variances μ_1,σ_1^2, μ_2, σ_2^2, μ_3, σ_3^2, respectively. Here $\pi_1, \pi_2, \pi_3 \geq 0$ and $\sum_{i=1}^3 \pi_i = 1$. Show that g is a valid density. What is its associated mean and variance?

Problem 3. Suppose that a density is of the form $(k+1)x^k$ for some constant $k > 1$ and $0 < x < 1$.

a. What is the mean of this distribution?
b. What is the variance?

Problem 4. Suppose that the time in days until hospital discharge for a certain patient population follows a density $f(x) = \frac{1}{3.3}\exp(-x/3.3)$ for $x > 0$.

a. Find the mean and variance of this distribution.
b. The general form of this density (the exponential density) is $f(x) = \frac{1}{\beta}\exp(-x/\beta)$ for $x > 0$ for a fixed value of β. Calculate the mean and variance of this density.
c. Plot the exponential pdf for $\beta = 0.1, 1, 10$.

Problem 5. The Gamma density is given by

$$\frac{1}{\beta^\alpha \Gamma(\alpha)} x^{\alpha-1}\exp(-x/\beta) \quad \text{for} \quad x > 0$$

for fixed values of α and β.

a. Derive the variance of the gamma density. You can assume the fact that the density integrates to 1 for any $\alpha > 0$ and $\beta > 0$.
b. The Chi-squared density is the special case of the Gamma density where $\beta = 2$ and $\alpha = p/2$ for some fixed value of p (called the "degrees of freedom"). Calculate the mean and variance of the Chi-squared density.

Problem 6. The Beta density is given by

$$\frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1} \quad \text{for} \quad 0 < x < 1$$

and

$$B(\alpha,\beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha+\beta).$$

a. Derive the mean of the beta density. Note: the following is useful for simplifying results: $\Gamma(c+1) = c\Gamma(c)$ for $c > 0$.
b. Derive the variance of the beta density.

Problem 7. The Poisson probability mass function is given by

$$P(X = x) = \frac{e^{-\lambda}\lambda^x}{x!} \quad \text{for} \quad x = 0, 1, 2, 3, \ldots$$

a. Derive the mean of this mass function.
b. Derive the variance of this mass function. Hint, consider $E[X(X-1)]$

Problem 8. Suppose that, for a randomly drawn subject from a particular population, the proportion of their skin that is covered in freckles follows a uniform density (constant between 0 and 1).

a. What is the expected percentage of a (randomly selected) person's body that is covered in freckles? (Show your work.)
b. What is the variance?

Problem 9. You have an MP3 player with a total of 1000 songs stored on it. Suppose that songs are played randomly *with replacement*. Let X be the number of songs played until you hear a repeated song.

a. What values can X take, and with what probabilities?
b. What is the expected value for X?
c. What is the variance for X?

Problem 10. Consider the SHHS data. Provide the bivariate scatter plots and means for all pairs of variables: BMI (`bmi_s1`), Age (`age_s1`), time in bed (`time_bed`), RDI (`rdi4p`), and waist circumference (`waist`).

Problem 11. For the variables in the previous problem display the histograms and produce QQ-plots. Describe each plot and explain what information they provide. Provide and interpret the empirical deciles of the distributions. Which density do you think would fit each of the observed data best?

Problem 12. Produce three bootstrap samples of the SHHS data and repeat the QQ-plots in the previous problem for each of the bootstrap samples. Also, produce the QQ-plots for the bootstrap samples versus each other. Interpret your results.

Problem 13. Produce the 95% confidence interval for the mean and median of each of the variables using the bootstrap. Are the intervals based on the percentiles of the bootstrap samples symmetric? Why or why not?

Problem 14. Produce a report written in `Rmarkdown` that would provide an initial data exploration analysis that you would be proud to share with your scientific collaborators. The first page should be the executive summary of your findings. Each plot or table should describe an important characteristic of the data and the text should provide a good flow of the story that you have learned from exploring the data.

Problem 15. Consider the brain image data and find the center of mass of the left side of the brain in the T1-weighted and T2-weighted brain images.

Chapter 5

Random vectors, independence, covariance, and sample mean

This chapter covers the following topics

- Random vectors
- Independent events and variables
- iid random variables
- Covariance and correlation
- Variance of sums of variables
- Sample variance
- Mixture of distributions

5.1 Random vectors

Random vectors are random variables collected into a vector.

- If X and Y are random variables, then (X, Y) is a random vector
- If $X_1, \ldots, X_n$ are random variables, then $(X_1, \ldots, X_n)$ is a random vector
- The columns of most common data structures are realizations of a random vector. Each column is a realization of one random variable

Consider the Sleep Heart Health Study (SHHS) data example

```
file.name = file.path("data", "shhs1.txt")
data.cv<-read.table(file=file.name,header = TRUE,na.strings="NA")
attach(data.cv)
dim(data.cv)
```

```
[1] 5804   30
```

Recall that there are 5804 subjects in the dataset (the number of rows) and 29 variables (the number of columns minus 1). The reason is that the first column contains the patient IDs in the SHHS. We extract a set of 6 realizations of random variables together with the corresponding subject ID and show the header of the table.

```
variable.index<- c("pptid", "waist", "rdi4p", "gender",
                   "age_s1", "smokstat_s1", "bmi_s1")

head(data.cv[variable.index])
```

```
  pptid waist      rdi4p gender age_s1 smokstat_s1   bmi_s1
1     1    86  1.4380826      1     55           2 21.77755
2     2   107 17.8021978      1     78           0 32.95068
3     3    82  4.8535565      0     77           0 24.11415
4     4    85  0.7973422      1     48           0 20.18519
5     5    76  2.7567568      0     66           2 23.30905
6     6    95  3.7209302      1     63           0 27.15271
```

The first column contains the SHHS-specific subject IDs, while each of the other columns contains the realization of the random variable for that particular subject. For example, the second subject in the SHHS has the ID 2 (`pptid=2`), has a waist circumference of 107 centimeters (`waist=107`), has a respiratory disruption index at 4% oxygen desaturation of 17.8 (`rdi4p=17.8`), is a male (`gender=1`), is 78 year old (`age_s1=78`), was never a smoker (`smokstat_s1=0`), and has a body mass index (BMI) equal to 32.95 (`bmi=32.95`). The vector (`rdi4p`, `age_s1`, `bmi_s1`) is a random vector of length 3 and its realizations for 5804 subjects are stored in `data.cv`.

Just as with random variables, the best way to think about random vectors is that they are the outcome of an experiment measuring multiple characteristics *before the experiment is conducted*. What is shown here are the realizations of these random vectors *after the experiment was run*. Of course, once the experiment is run (e.g. measurements are obtained) the realizations of the random vectors are vectors of numbers. We will use upper case bold fonts to denote random vectors and lower case bold fonts to denote realizations of random vectors. For example $\mathbf{X} = (X_1, X_2, X_3)$ could denote the random vector of `rdi4p`, `age`, and `bmi` before the information is collected, and $\mathbf{x} = (x_1, x_2, x_3)$ after the information is collected. Here for the second subject $x_1 = 17.8$, $x_2 = 78$ and $x_3 = 32.95$, indicating that there is no uncertainty left once measurement is conducted. To make things a little complicated, this assumes that the measurement mechanism is perfect. While this could reasonably be assumed for `age`, things are less clear for `rdi4p` and `bmi`; however, for now we will assume that measurement is perfect and we later will revisit measurement error.

Just as in the case of scalar random variables, random vectors with continuous

random variable entries are completely characterized by their probability density function (pdf). Of course, the pdf will have the same dimension as the dimension of the vector. For a vector of two continuous random variables $\mathbf{X} = (X_1, X_2)$ the joint probability density function is $f_{\mathbf{X}}(x_1, x_2)$, which satisfies $f(x_1, x_2) \geq 0$ for every x_1 and x_2 and

$$\int\int f_{\mathbf{X}}(x_1, x_2)dx_1dx_2 = 1 \ .$$

Once the joint pdf is available, the pdf of the individual random variables (a.k.a. the marginal pdf) can be obtained as

$$f_{X_1}(x_1) = \int f_{\mathbf{X}}(x_1, x_2)dx_2 \quad \text{and} \quad f_{X_2}(x_2) = \int f_{\mathbf{X}}(x_1, x_2)dx_1 \ .$$

It is easy to show that both $f_{X_1}(\cdot)$ and $f_{X_2}(x_2)$ are proper pdfs. A natural question to ask is *Why should we care about multivariate random variables and why should we deal with double or multidimensional integrals?* The answer is very simple: because data are multivariate (we often collect more than one measurement for each subject) and we are interested both in each individual measurement (marginal distributions) and in the mutual relationships between these variables (joint patterns of covariability). We did not say that it was going to be easy; but all these structures are natural representations of practical datasets. The faster one opens one's mind to these concepts the easier it will be to understand biostatistics at a deeper level.

Consider just two variables, `rdi4p` and `age_s1`, from the SHHS dataset. Figure 5.1 displays the bivariate plot, or scatter plot, of these realizations, with the y-axis cut at 50 for improved visuals. In the context of the observed SHHS data, the joint density $f_{\mathbf{X}}(x_1, x_2)$ is a theoretical concept that we will never know precisely. Instead, we have just samples (empirical realizations) of the random variable (X_1, X_2)=(`rdi4p`,`age_s1`). The reason we make such plots is to understand the joint covariability of the variables and to uncover potential relationships between them. One could, for example, wonder whether the average `rdi4p` changes with age. One possible way to investigate such an association is to calculate the average `rdi4p` in 10-year intervals and plot it. Figure 5.2 provides the same information as Figure 5.1, but it also adds the means of `rdi4p` in 10-year increments in age.

The plot in Figure 5.2 looks slightly different from the one in Figure 5.1 because we changed the y-axis to only show values of the `rdi4p` up to 20. This was done to ensure that the changes in `rdi4p` with age can be better visualized. We see that there seems to be an increase in average `rdi4p` as a function of age from 40 to 80 years with, maybe, a small decrease after 80 years. Also, in this population, the largest increases in the average `rdi4p` are between the groups with age$<$ 50 and age$>$ 50 and between the groups age$<$ 60 and age$>$ 60. This type of relationship can only be investigated if we consider the bivariate (or multivariate) nature of the data.

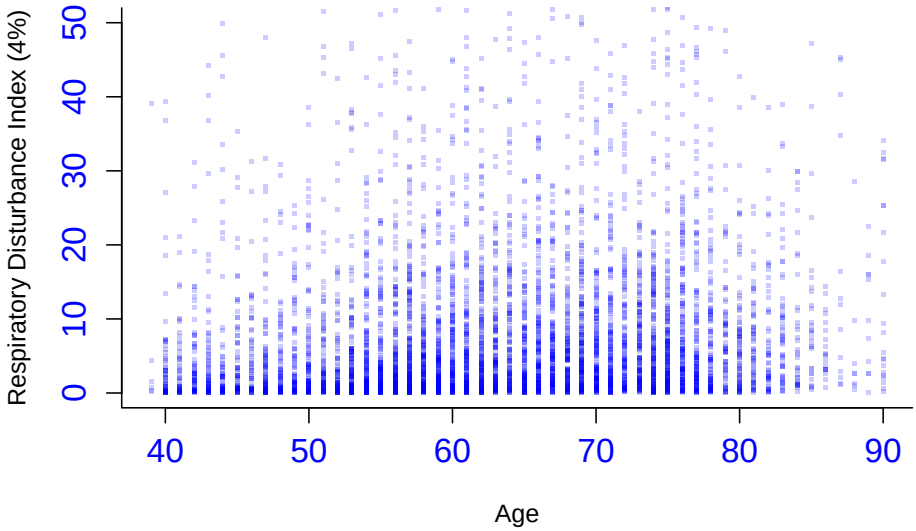


Figure 5.1: Scatter plot of age versus respiratory disruptance index at 4% saturation expressed in counts per hour in the SHHS.

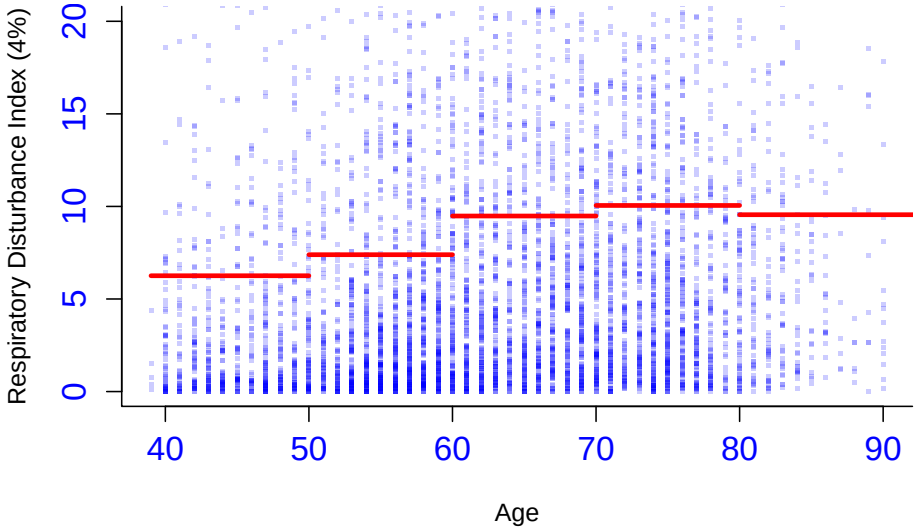


Figure 5.2: Scatter plot of age versus respiratory disruptance index at 4% saturation expressed in counts per hour together with the locals means of `rdi4p` in the SHHS.

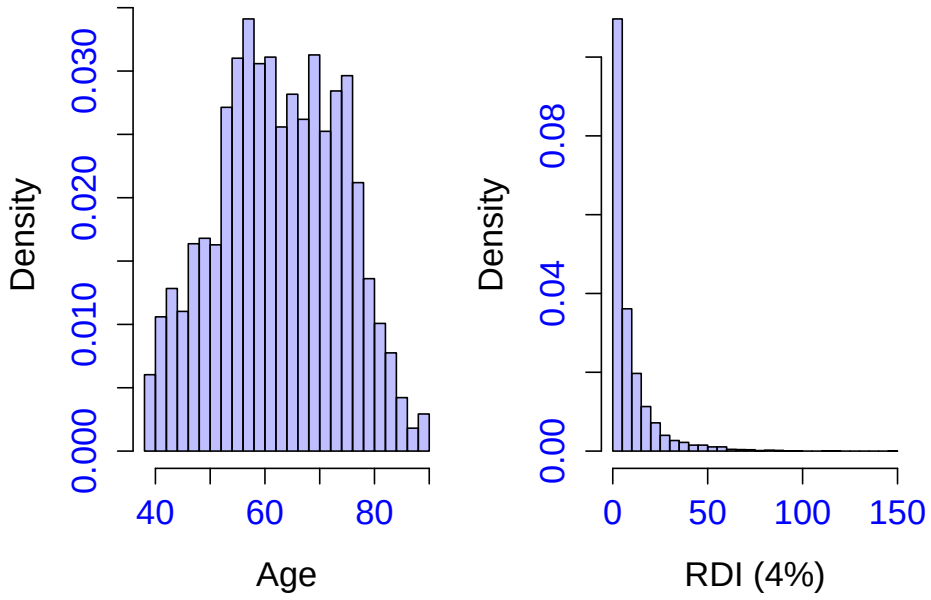


Figure 5.3: Marginal distribution of age and respiratory disruptance index at 4% saturation expressed in counts per hour in the SHHS.

The distribution displayed in Figures 5.1 and 5.2 are referred to as the bivariate, or joint, empirical distribution of `age_s1` and `rdi4p`. In contrast, Figure 5.3 provides the individual, or marginal, empirical distributions of `age_s1` and `rdi4p`. The marginal empirical distributions displayed in Figure 5.3 provide a different perspective on the information available in Figures 5.1 and 5.2. For example, the distribution of `age_s1` is relatively uniform between 55 and 75, where most of the sampled population's age is. This is probably due to the SHHS sampling design. In contrast, the distribution of `rdi4p` is highly skewed with a large probability mass close to zero. Indeed, 22% of the SHHS sample population has an `rdi4p` less than 1. The proportion of individuals with higher `rdi4p` (more affected by sleep disturbance) decreases fast, but there are a few individuals with very large `rdi4p`. For example, there are 2.1% individuals who have an `rdi4p` larger than 50. These characteristics of the observed data provide the highly skewed aspect of the `rdi4p` histogram.

However, as informative as these two histograms are, they do not provide information about the mutual associations shown in the bivariate plot. The reason is that once we separate the two columns we lose the mutual information provided by the joint pairs of `age_s1` and `rdi4p` at the individual level. In biostatistics the marginal distributions can be viewed as the joint distributions averaged (or integrated) over the other variable. This is the reason why data are actually forcing us to think about multidimensional data and pdfs.

For discrete random variables $\mathbf{X} = (X_1, X_2)$ the joint probability mass function is $f_{\mathbf{X}}(x_1, x_2)$, which satisfies $f(x_1, x_2) \geq 0$ for every x_1 and x_2 and

$$\sum_{x_1} \sum_{x_2} f_{\mathbf{X}}(x_1, x_2) = 1 \ ,$$

where the sum is taken over all possible values of the random variables. To illustrate, consider the random variables gender (`gender`) and hypertension (`HTNDerv_s1`), which are both binary. To obtain the marginal empirical distributions of these variables we simply need to calculate what proportion of men and hypertensive individuals are in the sample. This can be obtained as

```
round(mean(gender==1,na.rm=TRUE),digits=3)
```

```
[1] 0.476
```

```
round(mean(HTNDerv_s1==1,na.rm=TRUE),digits=3)
```

```
[1] 0.427
```

Thus, there are 47.6% men in the sample and 42.7% individuals who are hypertensive. Just as in the continuous case this is simple to obtain and highly informative, but does not contain the joint information encapsulated in the vector (`gender`,`HTNDerv_s1`). There are 5804 pairs of observations because no individuals have missing gender or HTN information. To obtain the joint distribution we simply count how many subjects are male and have hypertension, how many subjects are female and have hypertension, how many subjects are male and do not have hypertension, and how many subjects are female and do not have hypertension. Results are shown below in a two by two table.

```
        male female
HTN     1213   1265
non-HTN 1552   1774
```

These results illustrate the joint empirical distribution of `gender` and `HTNDerv_s1` in the SHHS visit 1. One can easily obtain the marginal distributions from the joint distribution, but not the other way around. Indeed, the proportion of men in the sample is

$$1213 + 1552/(1213 + 1552 + 1265 + 1774) = 0.476$$

and the proportion of individuals who are hypertensive is

$$(1213 + 1265)/(1213 + 1552 + 1265 + 1774) = 0.427 \ .$$

The empirical joint distribution of `gender` and `HTNDerv_s1` in the SHHS is obtained by dividing each cell in the previous two by two table by 5804, the number of subjects in the sample. We obtain

```
         male female
HTN     0.209  0.218
non-HTN 0.267  0.306
```

The estimated probability of being a female and having hypertension in SHHS is 0.218, which is close to the probability of being a man and having hypertension, 0.209. Thus, it makes sense to ask whether these observations are consistent with a true difference in the population between men and women or whether the difference is due to sampling variability. We will learn more about how to test for such differences, but this is a good example of how the testing problem appears naturally from the data.

We say that two random variables X_1 and X_2 are independent if and only if

$$f_{\mathbf{X}}(x_1, x_2) = f_1(x_1) f_2(x_2) \ ,$$

where $f_1(x_1)$ and $f_2(x_2)$ are the marginal pdfs of the random variables X_1 and X_2, respectively. To be more explicit, we could have used the notation $f_{X_1}(x_1)$ and $f_{X_2}(x_2)$ instead of $f_1(x_1)$ and $f_2(x_2)$, respectively, but this notation becomes more cumbersome. We will use the following fact extensively in this book: If the random variables $\mathbf{X} = (X_1, \ldots X_n)$ are independent, then their joint density or mass function is the product of their individual densities or mass functions. That is, if $f_i(\cdot)$ is the pdf of the random variable X_i then

$$f_{\mathbf{X}}(x_1, \ldots, x_n) = \prod_{i=1}^{n} f_i(x_i) \ .$$

In this definition we do not require the random variables to have the same distribution. Let us try to build the intuition about independence by focusing on the `gender` and `HTNDerv_s1` example. In this context, we would like to know what it would mean that the two variables are independent. If they are independent the interpretation could be that hypertension is "assigned" to individuals independent of their gender. Denote by $\pi_s = P(\text{sex} = 1)$ and $p_h = P(\text{HTN} = 1)$ and recall that these variables are called `gender` and `HTNDerv_s1` in the SHHS dataset. Independence would require

$$P(\text{sex} = 1, \text{HTN} = 1) = P(\text{sex} = 1) P(\text{HTN} = 1) = \pi_1 p_1 \ .$$

We do not know exactly what the values of π_1 and p_1 are but we have the estimators $\widehat{\pi}_1 = 0.476$ and $\widehat{p}_1 = 0.427$. Thus, a good estimator of $P(\text{sex} = 1, \text{HTN} = 1)$ is $\widehat{\pi}_1 \times \widehat{p}_1 = 0.203$. Similarly, $P(\text{sex} = 1, \text{HTN} = 0) = \pi_1(1 - p_1)$, $P(\text{sex} = 0, \text{HTN} = 1) = (1 - \pi_1)p_1$, and $P(\text{sex} = 0, \text{HTN} = 0) = (1 - \pi_1)(1 - p_1)$ and each of them can be estimated by simply plugging the $\widehat{\pi}_1$ and $\widehat{p}_1$ values instead of π_1 and p_1, respectively. Thus, if independence holds, we would expect the two by two matrix of probabilities to be

```
         male female
HTN     0.203  0.224
non-HTN 0.273  0.300
```

This matrix was obtained by estimating the marginal probabilities (π_1 and p_1) and using the independence assumption. The entries of this matrix are very

close to the corresponding entries of the previous matrix, which was obtained by estimating the joint distribution of the data without using the independence assumption. This may indicate that the independence assumption may be reasonable in this example, that is, that HTN assignment was done by nature independently of sex in this population. However, in future chapters we will provide rigorous ways of testing such a hypothesis.

5.2 Independent events and variables

Two events E and F are independent if

$$P(E \cap F) = P(E)P(F) \ .$$

It is easy to show that if A is independent of B, then

- A^c is independent of B
- A is independent of B^c
- A^c is independent of B^c

Two random variables, X and Y are independent if for any two sets A and B

$$P\{(X \in A) \cap (Y \in B)\} = P(X \in A)P(X \in B) \ .$$

5.2.1 Example: five-year lung cancer mortality

Consider two individuals who have just been diagnosed with stage IB NSCLC and we know that the five-year survival rate is 45%. Assuming that the survival of one individual does not depend on the survival of the other individual, we can calculate the probability that both of them will be alive in five years. Consider the events

- A={First individual dies in five years}, $P(A) = 0.45$
- B={Second individual dies in five years}, $P(B) = 0.45$
- $A \cap B$={Both the first and second individual die in five years}

Then, by assuming independence

$$P(A \cap B) = P(A)P(B = 0.45 \times 0.45 = 0.2025 \ .$$

Independence is a very strong assumption in biostatistics and is often used to simplify calculations. While independence may not hold exactly, conditional independence (of which we will learn more later) is an extremely useful and practical assumption.

5.2.2 Example: Sudden Infant Death Syndrome

Unfortunately, independence is not easy to understand and incorrect assumptions can have real life consequences. Consider, for example, the case of Dr. Roy Meadow, a pediatrician who testified in the murder trial of Sally Clark, whose two sons died at the age of 11 weeks in December 1996, and 8 weeks in January 1998, respectively. Based on an estimated prevalence of sudden infant death syndrome (SIDS) of 1 out of 8543, Dr. Meadow testified that the probability of a mother having two children with SIDS was $1/8543^2$, or less than 1 in 72 million. The mother on trial was convicted of murder in 1999, as reported by the British Medical Journal. Her conviction was overturned after her second appeal in 2003 after serving more than 3 years of her sentence. Clark's experience caused her to develop psychiatric problems and she died at age 42 from alcohol poisoning Wikipedia. More information about SIDS can be found, for example, on the US CDC website.

Let us investigate and see what assumptions were made by Dr. Meadow and what were his likely mistakes. For the purposes of this book, the principal mistake was to assume that the events of having SIDS within a family are independent. That is, $P(A_1 \cap A_2)$ is not necessarily equal to $P(A_1)P(A_2)$, where A_1 and A_2 are the events that the first and second sibling die of SIDS, respectively. The reason is that biological processes that have a believed genetic or familial environmental component tend to be dependent within families. Moreover, the estimated prevalence was obtained from reports on single cases, which provide no information about recurrence of SIDS within families.

5.2.3 Example: Monte Carlo simulations

Monte Carlo simulations can be used to calculate probabilities that are harder to calculate directly or provide an empirical alternative for confirming results obtained theoretically. Consider, for example, the case when X_1, X_2 are independent random variables with a $N(0,1)$ distribution. We would like to calculate $P(X_1 > 1.5, X_2 > 1) = P(X_1 > 1.5)P(X_2 > 1)$. Of course, this can be calculated numerically, as shown below

```
probt=(1-pnorm(1.5))*(1-pnorm(1))
```

We can also use Monte Carlo simulations to estimate the same probability and compare it with the theoretical one

```
#Set the seed for fully reproducible simulation results
set.seed(234901)

#Set the number of independent random samples to 100000
nsim=100000
```

```
#Simulate independently two random variables from a N(0,1)
x1=rnorm(nsim)
x2=rnorm(nsim)

#Calculate the frequency of jointly exceeding 1.5 and 1, respectively
probs=mean((x1>1.5) & (x2>1))

#Theoretical value
round(probt,digits=4)
```

```
[1] 0.0106
```

```
#MC-based simulation value
round(probs,digits=4)
```

```
[1] 0.0108
```

```
#Percent difference between theoretical and MC values
round(100*abs(probs-probt)/probt,digits=3)
```

```
[1] 2.176
```

In this case because we estimate a small probability we used more MC simulations and the difference even for such a small probability is less than 3%.

5.3 iid random variables

Consider a set of n independent random variables $X_1, \ldots, X_n$ and let $f_i(\cdot)$ for $i \in \{1, \ldots, n\}$ be the pdf (or pmf) of the ith random variable. We say that these variables are independent and identically distributed (iid) if they are independent and all $f_i(\cdot)$ are equal. More precisely, $f_1(\cdot) = \ldots = f_n(\cdot) = f(\cdot)$ and the joint distribution of the vector $\mathbf{X} = (X_1, \ldots, X_n)$ can be written as

$$f_{\mathbf{X}}(x_1, \ldots, x_n) = \prod_{i=1}^{n} f(x_i) \, .$$

The main difference from the definition of independence is that there is no index on the pdfs in the right side of the equation. This is due to the assumption of *identical distribution*. The iid assumption on random variables is the default assumption for random samples and many of the important theories of biostatistics are founded on assuming that variables are iid.

5.3.1 Example: five-year lung cancer mortality

Suppose that we know that the probability of survival for five years for a particular type of cancer is p. Four individuals with this type of cancer are followed for

five years and the second individual died, while the other three survived. Thus, the outcome of the experiment can be coded as $(1, 0, 1, 1)$ and we would like to calculate the joint probability mass function of this collection of outcomes. We denote by X_i the random variable of survival for the ith individual and we can calculate the joint probability of obtaining this collection of outcomes (before the experiment is run) as

$$\begin{aligned} P(X_1 = 1, X_2 = 0, X_3 = 1, X_4 = 1) &= P(X_1 = 1)P(X_2 = 0)P(X_3 = 1)P(X_4 = 1) \\ &= p(1-p)pp = p^3(1-p) \ . \end{aligned}$$

This calculation was possible because we assumed independence, while the result depends only on the information that *one of the four people died*, not on the information that *the second person died*. In other words, to calculate the probability it is sufficient to know that one individual died and not who that individual is. Mathematically, this is expressed by the commutativity of the product and indicates that exactly three other events have the same probability. These events are $(0, 1, 1, 1)$, $(1, 1, 0, 1)$ and $(1, 1, 1, 0)$.

Consider now the case when we have n individuals who have been diagnosed with this type of cancer and let $\mathbf{x} = (x_1, \ldots, x_n)$ be the collection of outcomes after five years for these individuals. Here $x_i \in \{0, 1\}$ indicates whether the ith person is dead (0) or alive (1) five years later. Just as in the previous scenario, the vector $\mathbf{x}$ contains a sequence of zero and ones of length n, the number of people in the sample. It is convenient to observe that $P(X_i = x_i) = p^{x_i}(1 - p_{x_i})^{1-x_i}$. This, of course, is trivial, as for $x_i = 0$ we have $P(X_i = 0) = p^0(1-p)^{1-0} = 1-p$ and for $x_i = 1$ we have $P(X_i = 1) = p^1(1-p)^{1-1} = p$. In this case we obtain

$$P(\mathbf{X} = \mathbf{x}) = \prod_{i=1}^{n} P(X_i = x_i) = \prod_{i=1}^{n} p^{x_i}(1 - p_{x_i})^{1-x_i} = p^{\sum x_i}(1-p)^{n-\sum x_i} \ .$$

For $n = 4$ and $x_1 = 1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 1$ we obtain that the probability is $p^3(1-p)$, as before. The probability of the event depends only on $\sum_{i=1}^{n} x_i$, the number of individuals who survived after 5 years, and not on who survived. From the point of view of calculating the probability it is enough to know the outcome of the sufficient statistic $\sum_{i=1}^{n} X_i$. The sufficient statistic plays an important role in biostatistics, as, conditional on the model, it contains all the necessary information expressed in the experiment. The reason is that it is much simpler to work with several simpler summaries of highly complex data than with the original data. Of course, the idea here is to find the simplest such sufficient statistic, as there are many sufficient statistics. Indeed, the data themselves are sufficient, but are not particularly useful without assumptions, explicit or implicit.

In an apocryphal story, one of the authors was approached by a collaborator who wanted to make no assumptions about the data. The author asked for the data on a memory stick, inserted it into the computer, made a copy, and handed the data back to his collaborator *without any results*. The only way to do what

the collaborator had asked was to simply return the data without touching or analyzing them.

5.3.2 Example: standard Normal

Consider the case when we have n iid samples $X_1, \ldots, X_n$ from the $N(0,1)$ distribution, which has mean 0 and variance 1. The standard Normal density is

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} .$$

We would like to derive the joint pdf for the vector $\mathbf{X} = (X_1, \ldots, X_n)$. By independence we have

$$f_{\mathbf{X}}(\mathbf{x}) = \prod_{i=1}^{n} f(x_i) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}} e^{-\frac{x_i^2}{2}} = \frac{1}{(2\pi)^{n/2}} e^{-\frac{\sum x_i^2}{2}} .$$

Here we used the fact that $e^a e^b = e^{a+b}$. Just as in the case of the Bernoulli distribution for cancer survival the joint distribution of the standard Normal depends only on $\sum x_i^2$, a much simpler summary of the data. It is easy to show that $(\sum x_i, \sum x_i^2)$ is a sufficient statistic for the $N(\mu, \sigma^2)$ distribution.

5.3.3 Simulating independent discrete variables

Independence is a fundamental concept and assumption in biostatistics. The standard simulation tools in R are set up to produce pseudo-independent random variables. Here we provide some examples of how to simulate random variables and how to interpret the results.

5.3.3.1 Independent Bernoulli draws

We start by simulating 10 independent Bernoulli random variables with the probability of success equal to $p = 0.3$.

```
set.seed(7840256)
#Draw 10 Bernoulli RV realizations with probability 0.3
x<-rbinom(10,1,prob=0.3)
x
```

```
 [1] 0 1 0 0 1 0 1 1 1 0
```

Interestingly, we would expect to obtain $0.3 \times 10 = 3$ successes in the simulation. Instead, we obtained 5. This does not happen because the random number generator is broken, but because it is possible to obtain any number of successes between 0 and 10, though the probability of obtaining each different number of

successes is different. Suppose we now want to obtain 3 independent draws from a Bernoulli(p) distribution for each $p \in \{0, 0.1, \ldots, 1\}$. This can be done in R because it allows drawing independent samples from Bernoulli distributions with different probabilities. Below we illustrate this.

```
#construct a vector of equally spaced probabilities
bernm<-seq(0,1,by=0.1)

#Simulate a vector of size 3 times the length of the prob. vector
#Because the number of draws exceeds the length of the prob. vector
#the probability vector is recycled
#these are independent, not iid variables
x<-rbinom(3*length(bernm),1,prob=bernm)

#Store into a matrix with 3 rows, each row corresponds to one repetition
#do not forget to check that the matrix format contains what you want
#on the row and columns, respectively
mx=matrix(x,ncol=length(bernm),byrow=TRUE)

#Each column corresponds to a success probability (0, 0.1, ..., 1)
mx
```

```
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
[1,]    0    0    1    0    1    0    0    1    1     1     1
[2,]    0    0    0    0    0    1    1    1    1     1     1
[3,]    0    0    0    0    1    1    1    0    1     1     1
```

5.3.3.2 Independent Poisson draws

We now simulate 10000 independent draws from a Poisson(20) distribution

```
#Simulate 10000 iid Poisson realizations
x<-rpois(10000,20)
```

We also simulate 24 independent draws from the Poisson(λ) distribution, where $\lambda \in \{1, 2.5, 5, 7.5, 10, 1000\}$. Because the number of means is smaller than the number of samples we will use the same recycling trick used for the Bernoulli random variable

```
#Set the Poisson means
poism<-c(1,2.5,5,7.5,10,1000)

#Simulate 24 independent Poisson; the Poisson mean is recycled 4 times
#These are independent, not iid variables
x<-rpois(24,poism)
x
```

```
 [1]    0    2    7   11   10  951    3    6    7    6   11 1000    2    6
```

```
[15]    6   10    9  967    0    2    4    9    8  957
```

5.3.3.3 Independent Normal draws

We also simulate 1000 realizations from a $N(2, 9^2)$ random variable and calculate their mean

```
#Simulate 1000 independent N(2,9)
x<-rnorm(1000,mean=2,sd=9)
round(mean(x),digits=2)
```

```
[1] 1.85
```

In some instances one might be interested in simulating more complex scenarios. Below we show such an example, not because it is practical, but to emphasize the flexibility of the simulation approaches. In particular, we would like to simulate 150 independent observations from $\mathrm{N}(\mu, \mu^2/9)$ for every $\mu \in \{1, 2, \ldots, 100\}$. All these random variables have the same coefficient of variation of $1/3$.

```
#Define the mean vector of length 100
normm<-1:100

#Set a standard deviation vector
sdm<-normm/3

#Simulate independently 150 observations from N(normm,sdm)
#Note the use of recycling
x<-rnorm(150*length(normm),mean=normm,sd=sdm)

#Store simulations in a 150 by 100 dimensional matrix
#Each row corresponds to one of the 150 independent samples
#Each column corresponds to one of the means
#Samples are iid by column and independent (non iid) by row
mx=matrix(x,ncol=length(normm),byrow=TRUE)
```

After conducting simulations it is always a good idea to inspect the results and make sure they were correctly conducted and stored. Below we show such a quality control approach. Since column j should correspond to mean j, the empirical mean on column j should be close to j. Again, none of the empirical means is exactly equal to the theoretical mean, but they should be close.

```
dim(mx)
```

```
[1] 150 100
```

```
round(colMeans(mx),digits=3)
```

```
  [1]  0.996  2.072  3.034  3.914  5.212  6.040  6.817  7.693  8.966  9.822
```

```
 [11] 11.174 11.803 13.004 14.221 14.815 16.652 17.238 17.627 19.447 19.606
 [21] 21.364 22.772 22.232 23.244 23.894 24.364 27.297 28.810 28.730 29.451
 [31] 30.523 31.534 32.850 34.340 33.857 37.367 36.568 37.529 40.604 41.563
 [41] 42.043 40.322 42.696 44.869 44.282 45.778 47.674 46.073 48.923 49.678
 [51] 51.072 50.290 52.386 53.652 54.845 55.787 57.536 58.090 56.108 61.341
 [61] 61.211 64.421 62.147 65.877 65.108 63.107 66.345 67.722 70.611 70.195
 [71] 74.193 72.407 72.003 72.219 75.286 80.614 78.597 79.007 77.080 78.088
 [81] 79.846 79.849 83.296 83.168 87.170 88.475 84.565 87.530 86.772 92.242
 [91] 90.892 91.435 91.446 93.640 98.246 92.785 95.826 95.182 98.649 97.429
```

5.3.4 Product of independent variables

We show that if X_1 and X_2 are independent random variables, then $E(X_1X_2) = E(X_1)E(X_2)$. We prove this for the case when the variables are continuous and have pdfs equal to $f_1(x_1)$ and $f_2(x_2)$, respectively. By independence $f_{\mathbf{X}}(x_1, x_2) = f_1(x_1)f_2(x_2)$ and

$$E(X_1X_2) = \int\int x_1x_2f_{\mathbf{X}}(x_1, x_2)dx_1dx_2 = \int\int \{x_1f_1(x_1)\}\{x_2f_2(x_2)\}dx_1dx_2 \ .$$

If we first integrate with respect to x_2 then $x_1f_1(x_1)$ is just a constant and can be placed in front of the integral. More precisely

$$\int\int \{x_1f_1(x_1)\}\{x_2f_2(x_2)\}dx_1dx_2 = \int [\{x_1f_1(x_1)\} \int x_2f_2(x_2)dx_2]dx_1 \ .$$

The integral $\int x_2f_2(x_2)dx_2$ is, by definition, equal to $E(X_2)$, which is a number that does not depend on x_1 and we obtain

$$E(X_1X_2) = \int [\{x_1f_1(x_1)\}dx_1 \times E(X_2) = E(X_1)E(X_2) \ .$$

5.4 Covariance and correlation

The covariance between two random variables X_1 and X_2 is defined as

$$\text{Cov}(X_1, X_2) = E\{[X_1 - E(X_1)][X_2 - E(X_2)]\} \ .$$

Since

$$[X_1 - E(X_1)][X_2 - E(X_2)] = X_1X_2 - X_1E(X_2) - E(X_1)X_2 + E(X_1)E(X_2)$$

and $E[X_1E(X_2)] = E(X_1)E(X_2)$, $E[E(X_1)X_2] = E(X_1)E(X_2)$ and $E[E(X_1)E(X_2)] = E(X_1)E(X_2)$ it follows that

$$\text{Cov}(X_1, X_2) = E(X_1X_2) - E(X_1)E(X_2) \ ,$$

which is a convenient computational formula for the covariance. It is important to understand that the covariance between two random variables is a measure of association between them. The first formula provides the interpretation. Indeed, if the variables X_1 and X_2 tend to be above (or below) their mean at the same time, then the product $[X_1 - E(X_1)][X_2 - E(X_2)] > 0$ and the covariance will be positive. The tendency variables to be above or below their respective means at the same time indicates that the two variables *covary* positively. Conversely, if one variable is above the mean of the variable when the other is below the mean, then $[X_1 - E(X_1)][X_2 - E(X_2)] < 0$ and the two variables tend to *covary* negatively. This explanation is still quite opaque as we are dealing with theoretical quantities. But, let us consider the case when we have n pairs of random vectors (X_{1i}, X_{2i}), $i = 1, \ldots, n$, from the joint distribution of (X_1, X_2) and have a closer look at the sample covariance (or estimator of the true covariance):

$$\widehat{\text{Cov}}(X_1, X_2) = \frac{1}{n}\sum_{i=1}^{n}(X_{1i} - \overline{X}_{1\cdot})(X_{2i} - \overline{X}_{2\cdot}) \ ,$$

where $\overline{X}_{1\cdot} = \sum_{i=1}^{n} X_{1i}/n$ and $\overline{X}_{2\cdot} = \sum_{i=1}^{n} X_{2i}/n$ are the sample averages of the two random variables, respectively. To better understand the formula, we consider the association between `age_s1` and `rdi4p` in the SHHS. In this case the variable indexed 1 is `age_s1`, the variable indexed 2 is `rdi4p`, $n = 5804$, the mean `age_s1` is $\overline{x}_{1\cdot} = 63.1$ years, and the mean `rdi4p` is 8.66 events per hour. We have switched to the lower case notation to indicate that these are realizations of random variables. Below we calculate the necessary quantities for obtaining the empirical covariance

```
cov_data<-cbind(age_s1,rdi4p)
cov_data<-cbind(cov_data,age_s1-mean(age_s1,na.rm=TRUE))
cov_data<-cbind(cov_data,rdi4p-mean(rdi4p,na.rm=TRUE))
cov_data<-cbind(cov_data,cov_data[,3]*cov_data[,4])
colnames(cov_data)<-c("age","rdi4p","centered_age",
                      "centered_rdip4","residual_prod")
round(head(cov_data),digits=2)
```

```
     age rdi4p centered_age centered_rdip4 residual_prod
[1,]  55  1.44        -8.13          -7.22         58.71
[2,]  78 17.80        14.87           9.15        135.97
[3,]  77  4.85        13.87          -3.80        -52.72
[4,]  48  0.80       -15.13          -7.86        118.93
[5,]  66  2.76         2.87          -5.90        -16.91
[6,]  63  3.72        -0.13          -4.93          0.66
```

Note, for example, that the first individual is 55 years old, which is below the average age in the population, 63.1, and has an `rdi4p` equal to 1.44, which is also below the mean `rdi4p`, 8.66. The third and fourth columns contain the centered `age` and `rdi4p` variables, or $r_{1i} = x_{1i} - \overline{x}_{1\cdot}$ and $r_{2i} = x_{2i} - \overline{x}_{2\cdot}$, respectively. We call these centered variables residuals and denote the

corresponding variables as `centered_age` and `centered_rdi4p`. For the first subject $r_{11} = 55 - 62.9 = -8.13$ (the slight inconsistency is due to rounding error) years and $r_{21} = 1.44 - 8.66 = -7.22$ events per hour. The product $r_{11} \times r_{21} = -8.13 \times (-7.22) = 58.7$ is positive (because both `age_s1` and `rdi4p` are below their corresponding means). This product is expressed in years times events per hour, probably not the most intuitive unit of measurement. The fifth column labeled `res_prod` contains this value and refers to the product of residuals r_{1i} and r_{2i}. The second person is 78 years old and has an `rdi4p` of 17.80 events per hour, which are both above their corresponding means, resulting in a positive product of residuals $r_{12} \times r_{22} = 135.97$. The third person is older than the average, but has an `rdi4p` below the average, resulting in a negative product of residuals $r_{13} \times r_{23} = -52.72$. This should explain that, in general, not all products of residuals will be either positive or negative, but that the covariance will measure the general tendency of variables to covary. The sample covariance is obtained by taking the average of the `residual_prod` column, which is the last entry of the vector below

```
round(colMeans(cov_data,na.rm=TRUE),digits=2)
```

```
          age          rdi4p   centered_age centered_rdip4  residual_prod
        63.13           8.66           0.00           0.00          14.45
```

Thus, the sample covariance is equal to 14.45, which is positive and expressed in years times number of events per hour. In `R` this can be obtained directly using

```
round(cov(age_s1,rdi4p,use="pairwise.complete.obs"),digits=3)
```

```
[1] 14.451
```

It is worth revisiting the residual products $r_{1i}r_{2i}$ and think of each such subject-specific product as an estimator of the covariance between the random variables based solely on the information from subject 1. Thus, the population level estimator is obtained by averaging the subject-specific estimators. This is a general strategy that is used over and over again in biostatistics to obtain more precise estimators of the target parameters. Let us evaluate the bias of $r_{1i}r_{2i}$. Note that

$$(X_{i1} - \overline{X}_{1\cdot})(X_{i2} - \overline{X}_{2\cdot}) = \{(X_{i1} - \mu_1) + (\mu_1 - \overline{X}_{1\cdot})\}\{(X_{i2} - \mu_2) + (\mu_2 - \overline{X}_{2\cdot})\}\,,$$

where $\mu_1 = E(X_{i1})$ and $\mu_2 = E(X_{i2})$. The right side of the equality can be further expanded to

$$(X_{i1}-\mu_1)(X_{i2}-\mu_2)+(X_{i1}-\mu_1)(\mu_2-\overline{X}_{2\cdot})+(\mu_1-\overline{X}_{1\cdot})(X_{i2}-\mu_2)+(\mu_1-\overline{X}_{1\cdot})(\mu_2-\overline{X}_{2\cdot})$$

Thus, to calculate the expected value of $(X_{i1} - \overline{X}_{1\cdot})(X_{i2} - \overline{X}_{2\cdot})$ we need to calculate the expected value of each one of these four terms separately. For

the first term, by definition, $E\{(X_{i1}-\mu_1)(X_{i2}-\mu_2)\} = \mathrm{Cov}(X_1, X_2)$. For the second term, we observe first that

$$(\mu_2 - \overline{X}_{2\cdot}) = \mu_2 - \frac{1}{n}\sum_{k=1}^{n} X_{k2} = -\frac{1}{n}\sum_{k=1}^{n}(X_{k2}-\mu_2) \; .$$

Plugging this back into the expectation formula for the second term we obtain

$$E\{(X_{i1}-\mu_1)(\mu_2 - \overline{X}_{2\cdot})\} = -\frac{1}{n}\sum_{k=1}^{n} E\{(X_{i1}-\mu_1)(X_{k2}-\mu_2)\} \; .$$

Because for every $k \neq i$ X_{i1} and X_{k2} are independent we have

$$E\{(X_{i1}-\mu_1)(X_{k2}-\mu_2)\} = E(X_{i1}-\mu_1)E(X_{k2}-\mu_2) = 0 \; ,$$

which implies that

$$E\{(X_{i1}-\mu_1)(\mu_2 - \overline{X}_{2\cdot})\} = -\frac{1}{n}E\{(X_{i1}-\mu_1)E(X_{i2}-\mu_2)\} = -\frac{\mathrm{Cov}(X_1, X_2)}{n} \; .$$

Similarly we obtain that

$$E\{(\mu_1 - \overline{X}_{1\cdot})(X_{i2}-\mu_2)\} = -\frac{\mathrm{Cov}(X_1, X_2)}{n} \; .$$

For the last term we need to calculate

$$E\{(\mu_1 - \overline{X}_{1\cdot})((\mu_2 - \overline{X}_{2\cdot}))\} = \frac{1}{n^2}E\{\sum_{k=1}^{n}\sum_{l=1}^{n}(X_{k1}-\mu_1)(X_{l2}-\mu_2)\} \; .$$

As before, for every $k \neq l$ we have $E\{(X_{k1}-\mu_1)(X_{l2}-\mu_2)\} = 0$ indicating that

$$\begin{aligned} E\left\{(\mu_1 - \overline{X}_{1\cdot})((\mu_2 - \overline{X}_{2\cdot}))\right\} &= \frac{1}{n^2}\sum_{k=1}^{n} E\left\{(X_{k1}-\mu_1)(X_{k2}-\mu_2)\right\} \\ &= \frac{n\mathrm{Cov}(X_1, X_2)}{n^2} \\ &= \frac{\mathrm{Cov}(X_1, X_2)}{n} \; . \end{aligned}$$

Thus, putting all these results together it follows that

$$E\{(X_{1i} - \overline{X}_{1\cdot})(X_{2i} - \overline{X}_{2\cdot})\} = \frac{n-1}{n}\mathrm{Cov}(X_1, X_2) \; ,$$

which indicates that the subject-specific covariance estimator is biased with the bias equal to

$$\frac{n-1}{n}\mathrm{Cov}(X_1, X_2) - \mathrm{Cov}(X_1, X_2) = -\frac{\mathrm{Cov}(X_1, X_2)}{n}.$$

Thus, an unbiased estimator of the covariance based only on the subject-level data is

$$\frac{n}{n-1}(X_{1i} - \overline{X}_{1\cdot})(X_{2i} - \overline{X}_{2\cdot}) .$$

From this, it is easy to show that the sample covariance is a biased estimator of the covariance and that

$$E\{\widehat{\text{Cov}}(X_1, X_2)\} = \text{Cov}(X_1, X_2) - \frac{\text{Cov}(X_1, X_2)}{n} .$$

An easy way to obtain an unbiased estimator of the covariance is to use

$$\frac{1}{n-1}\sum_{i=1}^{n}(X_{i1} - \overline{X}_{1\cdot})(X_{i2} - \overline{X}_{2\cdot}) .$$

When n is moderate or large the distinction between dividing by n or $n-1$ is irrelevant, though sometimes, when n is small, some differences exist.

If X_1 and X_2 are two independent random variables, then $\text{Cov}(X_1, X_2) = 0$; this means that independent random variables have a zero covariance. This result is easy to prove, as $\text{Cov}(X_1, X_2) = E(X_1X_2) - E(X_1)E(X_2)$ and we have already shown that if X_1 and X_2 are independent, then $E(X_1X_2) = E(X_1)E(X_2)$. It is easy to show that $\text{Cov}(X_1, X_2) = \text{Cov}(X_2, X_1)$ and that $\text{Cov}(X_1, X_2)$ can be positive or negative. An important result (that will not be proved here) is the Cauchy-Schwarz inequality

$$|\text{Cov}(X_1, X_2)| \leq \sqrt{\text{Var}(X_1)\text{Var}(X_2)} ,$$

where $|a|$ is the absolute value of a. One of the problems with covariances is that they are expressed in units that combine the units of both variables and cannot be compared across pairs of variables that have different units and are hard to transport from one study to another. To circumvent this problem the correlation of two random variables is defined as

$$\text{Cor}(X_1, X_2) = \frac{\text{Cov}(X_1, X_2)}{\sqrt{\text{Var}(X_1)\text{Var}(X_2)}} ,$$

which is a unitless value between -1 and 1. A value of 0 of the correlation corresponds to no linear association between the two random variables, while a value of 1 or -1 can be obtained if and only if X_2 can be expressed as $X_2 = \alpha + \beta X_1$, that is, there is a perfect linear association between the two variables. Just as in the case of the covariance, the correlation is a purely theoretical concept, but it can be estimated from the data by

$$\widehat{\text{Cor}}(X_1, X_2) = \frac{\sum_{i=1}^{n}(X_{1i} - \overline{X}_{1\cdot})(X_{2i} - \overline{X}_{2\cdot})}{\sqrt{\sum_{i=1}^{n}(X_{1i} - \overline{X}_{1\cdot})^2 \sum_{i=1}^{n}(X_{2i} - \overline{X}_{2\cdot})^2}} ,$$

where there is no division by n because it cancels out between the numerator and denominator. This can be calculated directly from the matrix we have built earlier.

```
numerator=sum(cov_data[,5],na.rm=TRUE)
denominator1<-sum(cov_data[,3]^2,na.rm=TRUE)
denominator2<-sum(cov_data[,4]^2,na.rm=TRUE)
cor_age_rdi4p<-numerator/sqrt(denominator1*denominator2)
round(cor_age_rdi4p,digits=3)
```

```
[1] 0.104
```

It can also be calculated directly from the data without going through the data transformations

```
round(cor(age_s1,rdi4p,use="pairwise.complete.obs"),digits=3)
```

```
[1] 0.104
```

This small value of the correlation should not be surprising given the scatter plot of `age_s1` versus `rdi4p`, which indicates that there is a weak positive association between the two variables. This association is exhibited by the increase in the mean `rdi4p` as a function of age. Of course, we would like to know whether this small positive association is due to the particular sample we collected, or whether the evidence that it is a true association is statistically strong. One possibility is to obtain a confidence interval based on nonparametric bootstrap for the correlation coefficient. This can be calculated below

```
#Set the number of bootstrap samples
n.boot=1000

#Set the vector that records observed correlations
corr_vec=rep(NA,n.boot)

#Conduct the boostrap sampling for the correlation
for (i in 1:n.boot) {
  sample.vector<-sample(1:length(age_s1),replace=TRUE)
  corr_vec[i]=cor(age_s1[sample.vector],rdi4p[sample.vector],
                  use="pairwise.complete.obs")
}

#Obtain the mean, standard deviation and
# 95% confidence intervals for correlation
mean_corr<-mean(corr_vec)
sd_corr<-sd(corr_vec)
CI<-c(mean_corr-2*sd_corr,mean_corr+2*sd_corr)
CI_r<-round(CI,digits=3)
```

The 95% is $(0.077, 0.13)$, which does not cover zero, indicating that there is enough statistical evidence against the hypothesis that there is no association between `age_s1` and `rdi4p`. We can obtain the covariance and correlation matrix estimators of a collection of variables in `R` as follows

```
subset.data.cv<-data.cv[,c("age_s1","rdi4p","bmi_s1")]
round(cov(subset.data.cv,use="pairwise.complete.obs"),digits=3)
```

```
        age_s1   rdi4p bmi_s1
age_s1 125.962  14.451 -4.679
rdi4p   14.451 154.575 20.142
bmi_s1  -4.679  20.142 25.887
```

```
round(cor(subset.data.cv,use="pairwise.complete.obs"),digits=3)
```

```
       age_s1 rdi4p bmi_s1
age_s1  1.000 0.104 -0.082
rdi4p   0.104 1.000  0.318
bmi_s1 -0.082 0.318  1.000
```

The covariance matrix contains the variance estimators on the main diagonal. For example, the estimated variances of `age_s1` and `rdi4p` are 125.962 and 154.575, respectively. The off-diagonal terms provide the covariances between pairs of variables. For example, the estimated covariance between `rdi4p` and `age` is 14.451 and the covariance between `rdi4p` and `bmi` is 20.142. The covariance matrix is symmetric because of the symmetry of the covariance formula. The correlation matrix contains 1 on the main diagonal because the correlation between any variable and itself is 1. The off-diagonal entries of the matrix are the estimated correlations between pairs of variables. For example, the estimated correlation between `age_s1` and `rdi4p` is 0.104 and between `rdi4p` and `bmi` is 0.318.

5.4.1 No correlation does not imply independence

Consider a random variable $X \sim N(0,1)$ and note that X and X^2 are not independent. Indeed,

$$P(X>1, X^2>1) = P(X>1) \neq P(X>1)P(X^2>1) = P(X>1)P(X<-1 \quad \text{or} \quad X>1)$$

because $P(X>1) = 0.159$ and $P(X<-1 \quad \text{or} \quad X>1) = 0.317$. In fact, one could say that there is a perfect association between X and X^2, as one is the square of the other. Let us show that the correlation between these variables is 0. We know that $E(X) = 0$ and $E(X^2) = 1$. Therefore

$$\text{Cov}(X, X^2) = E(X^3) - E(X)E(X^2) = E(X^3)\ .$$

However,

$$E(X^3) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x^3 e^{-\frac{x^2}{2}} dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{0} x^3 e^{-\frac{x^2}{2}} dx + \frac{1}{\sqrt{2\pi}} \int_{0}^{\infty} x^3 e^{-\frac{x^2}{2}} dx\ .$$

If in the first integral we make the change of variable $y = -x$, then we obtain

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{0} x^3 e^{-\frac{x^2}{2}} dx = \frac{1}{\sqrt{2\pi}} \int_{\infty}^{0} (-y)^3 e^{-\frac{(-y)^2}{2}} d(-y) = -\frac{1}{\sqrt{2\pi}} \int_{0}^{\infty} y^3 e^{-\frac{y^2}{2}} dy\ ,$$

which shows that $E(X^3) = 0$ and $\text{Cov}(X, X^2) = 0$. Note that this is true for any symmetric variable, not just the Normal. However, if a random vector has a multivariate Normal distribution and the correlation is 0 between two of the random variables, then they are independent. *Thus, it is practical to think about the correlation as a measure of linear association between variables.*

5.5 Variance of sums of variables

If X_i for $i = 1, \ldots, n$ is a collection of random variables, then

$$\text{Var}(\sum_{i=1}^{n} a_i X_i + b) = \sum_{i=1}^{n} a_i^2 \text{Var}(X_i) + 2 \sum_{i<j} a_i a_j \text{Cov}(X_i, X_j) \; .$$

Note that $E(\sum_{i=1}^{n} a_i X_i + b) = \sum_{i=1}^{n} a_i \mu_i + b$, where $\mu_i = E(X_i)$. Thus,

$$\text{Var}(\sum_{i=1}^{n} a_i X_i + b) = E\{\sum_{i=1}^{n} a_i (X_i - \mu_i)\}^2 = E\{\sum_{i=1}^{n} a_i^2 (X_i - \mu_i)^2 + \sum_{i \neq j} a_i a_j (X_i - \mu_i)(X_j - \mu_j)\} \; .$$

Since $E\{a_i^2 (X_i - \mu_i)^2\} = a_i^2 \text{var}(X_i)$, $E\{a_i a_j (X_i - \mu_i)(X_j - \mu_j)\} = a_i a_j \text{Cov}(X_i, X_j)$, and $\text{Cov}(X_i, X_j) = \text{Cov}(X_j, X_i)$ the result follows.

5.5.1 Variance properties

If the variables are not correlated then $\text{Cov}(X_i, X_j) = 0$ for every $i \neq j$ and

$$\text{Var}(\sum_{i=1}^{n} a_i X_i + b) = \sum_{i=1}^{n} a_i^2 \text{Var}(X_i) \; .$$

A commonly used implication of these properties is that if a collection of random variables X_i are uncorrelated then the variance of the sum is the sum of the variances. More precisely, if X_i are uncorrelated then

$$\text{Var}(\sum_{i=1}^{n} X_i) = \sum_{i=1}^{n} \text{Var}(X_i) \; .$$

This follows directly by setting $b = 0$ and $a_1 = \ldots = a_n = 1$. Therefore, it is sums of variances that tend to be useful, not the sums of standard deviations. That is, the standard deviation of the sum of a bunch of independent random variables is the square root of the sum of the variances, not the sum of the standard deviations. The calculation of the variances provides us with the formula for the standard deviation of a sum of independent random variables. More precisely,

$$\text{SD}(\sum_{i=1}^{n} X_i) = \sqrt{\sum_{i=1}^{n} \text{Var}(X_i)} \; .$$

This formula looks similar to the sum of standard deviations, though

$$\mathrm{SD}(\sum_{i=1}^{n} X_i) = \sqrt{\sum_{i=1}^{n} \mathrm{Var}(X_i)} \neq \sum_{i=1}^{n} \sqrt{\mathrm{Var}(X_i)} = \sum_{i=1}^{n} \mathrm{SD}(X_i) \ .$$

In general $\sum_{i=1}^{n} \mathrm{SD}(X_i)$ is much larger than $\mathrm{SD}(\sum_{i=1}^{n} X_i)$. Consider, for example, the case when X_i are independent with variance 1. In this case $\mathrm{SD}(\sum_{i=1}^{n} X_i) = \sqrt{n}$ and $\sum_{i=1}^{n} \mathrm{SD}(X_i) = n$. When the number of observations is 100 the difference between the sum of standard deviations and the standard deviation of the sum is of an order of magnitude. These differences have very important practical consequences in terms of calculating the length of confidence intervals and when investigating the statistical significance of the observed data.

5.5.2 Standard error of the empirical mean

Using the results in the previous section it follows that X_i for $i = 1, \ldots, n$ are independent identically distributed random variables with variance $\sigma^2 = \mathrm{Var}(X_i)$, then the variance of the sample mean $\overline{X}_n = \sum_{i=1}^{n} X_i/n$ is

$$\mathrm{Var}(\overline{X}_n) = \sum_{i=1}^{n} \frac{1}{n^2} \mathrm{Var}(X_i) = \frac{n}{n^2} \mathrm{Var}(X_1) = \frac{\sigma^2}{n} \ .$$

This result was simply obtained by replacing $a_i = \frac{1}{n}$ in the previous equation and summing over n equal variances. This is an important result that indicates how information is being accumulated. Remember that according to Chebyshev's inequality applied to $\overline{X}_n$ we have

$$P\{|\overline{X}_n - \mu| \leq k\sqrt{\mathrm{Var}(\overline{X}_n)}\} \geq 1 - \frac{1}{k^2} \ .$$

As $\mathrm{Var}(\overline{X}_n) = \sigma^2/n$ we obtain that, irrespective of the distribution of the random variables X_i

$$P(\overline{X}_n - k\frac{\sigma}{\sqrt{n}} \leq \mu \leq \overline{X}_n + k\frac{\sigma}{\sqrt{n}}) \geq 1 - \frac{1}{k^2} \ .$$

Let us unpack this result and provide a little bit of intuition. Recall that μ is a theoretical parameter of interest that will never be known exactly. Instead, we use data to calculate the empirical mean $\overline{X}_n$ that we hope is close to the theoretical mean that we are trying to estimate. This formula explains exactly what is meant by "close." Indeed, assume for a moment that the standard deviation, σ, is known and we choose a particular small probability $1/k^2$. In this case, the interval between $\overline{X}_n - k\sigma/\sqrt{n}$ and $\overline{X}_n + k\sigma/\sqrt{n}$ will cover the true value of the parameter μ with high probability. The length of this (confidence)

interval is $2k\sigma/\sqrt{n}$, which decreases at the rate of the square root of the number of observations, n. That is, if we need an interval that is twice as short, then we need four times as many independent observations. There is no guarantee that the true mean will be in the interval after the experiment is run, just that the mean will be in the interval with a certain probability before the experiment is run. In other words, if the experiment is run many times, then the true mean will be covered by the realized confidence interval (the one calculated based on data) at least $100(1-1/k^2)\%$ of the time.

Students and scientists working on data analysis complain that this interpretation of closeness and confidence intervals is too complicated, hard to understand, and non-intuitive. We agree that the interpretation requires quite a bit of ruminating and lacks the satisfying quality of final statements about exactly where the true mean is. However, given the complexity of new technology, the enormous costs in time and resources dedicated to science and its applications, and the vast computational tools dedicated to algorithmic implementation, it is a small price to pay to take a minute to think about what it all means. So, our answer is simple: get over it, ruminate, and try to understand.

To better understand the difference between the variance of the observation and the variance of the mean of observations consider the case when 9 independent observations $X_1, \ldots, X_9$ are obtained from a $N(2, 9)$ distribution and let $\overline{X}_9$ be the empirical mean of these observations. We know that $E(\overline{X}_9) = 2$ and $\text{Var}(\overline{X}_9) = 9/9$. It can be shown that $\overline{X}_9 \sim N(2, 1)$. Figure 5.4 displays and compares the theoretical and empirical (sampling) distributions of X_i and $\overline{X}_9$, respectively. For the empirical distributions we used samples of size 100 from the theoretical distributions.

The left panel in Figure 5.4 represents the distribution of individual variables, which have a $N(2, 9)$ distribution. The spread of this distribution can be interpreted as our knowledge (or lack thereof) about where a future observation from the same distribution is likely to fall. The larger spread corresponds to less knowledge about the position of a future observation. The right panel displays the theoretical and empirical distributions for the mean of 9 observations $N(2, 9)$. Note that the mean of the 9 observations has a tighter (narrower) distribution. In biostatistics we say that this is the effect of accumulating information about the mean of the variables. These differences imply that observing additional data does not provide any additional information about future realizations, but it provides a lot of information about the future mean of those observations. This information is represented as reduction of variability and can be visualized as the narrowing of the distribution.

When X_i, $i = 1, \ldots, n$ are iid random variables $\text{Var}(\overline{X}_n) = \frac{\sigma^2}{n}$, where $\sigma^2 = \text{Var}(X_i)$ is the variance of a single observation. In biostatistics the standard deviation of the mean $\sigma/\sqrt{n}$ is called the *standard error of the sample mean*. An easy way to remember is that the sample mean has to be less variable (or more precise) than a single observation, therefore its standard deviation is

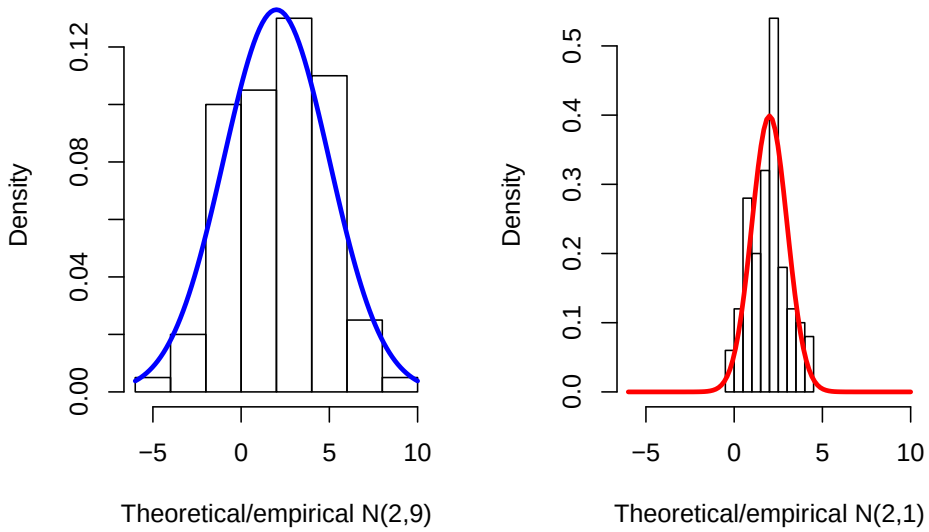


Figure 5.4: Left panel: theoretical and empirical distribution obtained from a sample of size 100 for the N(2,9). Right panel: theoretical and empirical distribution obtained from a sample of size 100 for the N(2,1).

divided by $\sqrt{n}$, where n is the number of independent variables contributing to the mean.

5.6 Sample variance

If X_i $i = 1, \ldots, n$ are random variables with variance σ^2 the *sample variance* is defined as

$$S^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X}_n)^2}{n-1} .$$

This formula deserves a closer look, as well. Each variable $X_i - \overline{X}_n$ is the deviation between the observed variable from one experiment and the mean of all observations. Thus, taken together, all variables $X_i - \overline{X}_n$ provide information about the spread of observations around their empirical mean. One could also think of $r_i = X_i - \overline{X}_n$ as the residual information contained in the ith observation, X_i, after incorporating its information into the empirical mean $\overline{X}_n$. Thus, the square of this residual $r_i^2 = (X_i - \overline{X}_n)^2$ is an estimator of the variability of the data. Because we have n such estimators it makes sense to take the average of all of them. Note that we are summing the estimators of the variance, just as in the theoretical case where we have shown that the variance of the sum is the sum of the variances. The only small detail is why we divide by $n-1$ instead of n. This, of course makes absolutely no difference when n is moderate or large, but can have quite an effect when $n \leq 20$. For now, we should remember that

sample (or empirical) variance is (nearly) the mean of the squared deviations from the mean.

5.6.1 Example: sample variance in the SHHS

We are using the `age_s1` and `rdi4p` data to illustrate the step-by-step calculations of the sample (or empirical) variance. Below we show the realizations of the variables `age` and `rdi4p` in the first two columns. Recall that the mean age is 63.13 and the mean `rdi4p` is 8.66.

```
var_data<-cbind(age_s1,rdi4p)
var_data<-cbind(var_data,(age_s1-mean(age_s1,na.rm=TRUE))^2)
var_data<-cbind(var_data,(rdi4p-mean(rdi4p,na.rm=TRUE))^2)
colnames(var_data)<-c("age","rdi4p","squared_res_age","squared_res_rdip4")
round(head(var_data),digits=2)
```

```
     age rdi4p squared_res_age squared_res_rdip4
[1,]  55  1.44           66.16             52.09
[2,]  78 17.80          221.00             83.66
[3,]  77  4.85          192.26             14.45
[4,]  48  0.80          229.04             61.75
[5,]  66  2.76            8.21             34.80
[6,]  63  3.72            0.02             24.35
```

The third and fourth columns contain the squared centered `age_s1` and `rdi4p` variables, or $r_{1i}^2 = (x_{1i} - \overline{x}_{1\cdot})^2$ and $r_{2i}^2 = (x_{2i} - \overline{x}_{2\cdot})^2$, respectively. Note that we have switched from capital letters for random variables to lower case letters for the realization of these random variables. We call these squared residuals `squared_res_age` and `squared_res_rdip4`. For the first subject $r_{11}^2 = (55 - 63.134)^2 = 66.16$ years squared and $r_{21}^2 = (1.438 - 8.6555)^2 = 52.09$ events per hour squared. To better follow the steps, we have used more decimals for the calculations, though we only provided results up to the second decimal. The sample variances are obtained by summing up the values of the vectors `squared_res_age` and `squared_res_rdip4` and dividing by the number of terms contributing to the sum. When we have missing data, as happens in most studies, one needs to be careful about the denominator. First, let us calculate and display the column sums

```
#Calculate the column sums for all variables
#Remove NA data from calculating the sums
column_sums<-colSums(var_data,na.rm=TRUE)
names(column_sums)<-c("Sum of age","Sum of rdi4p",
                      "SS of age res.","SS of rdi4p res.")
round(column_sums,digits=3)
```

```
      Sum of age     Sum of rdi4p   SS of age res. SS of rdi4p res.
       366430.00         50236.62        730957.71        897000.34
```

```
#Find out how many subjects contribute data for the age and rdi4p variance
non_na_age<-sum(!is.na(age_s1))
non_na_rdi4p<-sum(!is.na(rdi4p))
non_na_age
```

```
[1] 5804
```

```
non_na_rdi4p
```

```
[1] 5804
```

This indicates that both the `age_s1` and `rdi4p` information is complete (there are no missing data entries). Thus, we obtain the sample variance estimators as follows

```
var_age<-column_sums[3]/(non_na_age-1)
names(var_age)<-"Age variance"
var_rdi4p<-column_sums[4]/(non_na_rdi4p-1)
names(var_rdi4p)<-"rdi4p variance"
```

Display the variance for `age_s1` and `rdi4p`, respectively

```
round(var_age,digits=2)
```

```
Age variance
      125.96
```

```
round(var_rdi4p,digits=2)
```

```
rdi4p variance
        154.58
```

Of course, these variances can be calculated directly in `R` by avoiding all the steps we have unpacked here by simply doing.

```
round(var(age_s1,na.rm=TRUE),digits=2)
```

```
[1] 125.96
```

```
round(var(rdi4p,na.rm=TRUE),digits=2)
```

```
[1] 154.58
```

So, what have we learned from the fact that the sample mean `rdi4p` is 8.66 and the sample variance is 154.58 based on 5804 observations? Well, we know that the true mean `rdi4p` in the population is probably pretty close to 8.66. Using Chebychev's inequality we know that with at least $1 - 1/k^2$ probability the interval

$$(\overline{X}_n - k\frac{\sigma}{\sqrt{n}}, \overline{X}_n + k\frac{\sigma}{\sqrt{n}})$$

covers the true mean. Choosing, for example, $k = 5$, we obtain that the interval

$$(\overline{X}_n - 5\frac{\sigma}{\sqrt{5804}}, \overline{X}_n + 5\frac{\sigma}{\sqrt{5804}})$$

covers the true mean with probability at least equal to 0.96. To conduct the explicit calculations we pretend for now that we can replace the true σ with the sample one $\sigma \approx \sqrt{154.58} = 12.43$; more about the estimated σ in Chapter 11 where we discuss t-based confidence intervals. After the experiment is run (data are collected) the realized confidence interval is $(8.66 - 5 \times 12.43/\sqrt{5804}, 8.66 + 5 \times 12.43/\sqrt{5804})$, which is

$$(7.84, 9.48) \ .$$

Disappointingly, we have no idea whether the true mean is in this interval. It either is or it is not. What we do know is that before the experiment was run there was at least a 96% chance that what we will obtain will contain the true mean. We would just want to remind the reader that, while this may be perceived as disappointing, it does represent the reality of data collection and of the observed variability among results of experiments even when experiments are designed to be identical. The philosophy and interpretation behind the numbers is a little complicated, but makes perfect sense.

5.6.2 The sample variance is unbiased

We are now calculating the bias of the sample variance and we show that $\frac{n-1}{n}(X_i - \overline{X}_n)^2$ is an unbiased estimator of the variance. Consider independent (or uncorrelated) random variables X_i, $i = 1, \ldots, n$ with $E(X_i) = \mu$ and $\mathrm{Var}(X_i) = \sigma^2$. It follows that

$$E(X_i - \overline{X}_n)^2 = E\{(X_i - \mu) - (\overline{X}_n - \mu)\}^2 \ ,$$

which becomes

$$E(X_i - \mu)^2 - 2E\{(X_i - \mu)(\overline{X}_n - \mu)\} + E(\overline{X}_n - \mu)^2 \ .$$

The first term is by definition $\sigma^2 = E(X_i - \mu)^2$ and the third term is equal to $\mathrm{Var}(\overline{X}_n) = \sigma^2/n$ because $E(\overline{X}_n) = \mu$. To evaluate the second term observe that

$$\overline{X}_n - \mu = \frac{1}{n}\sum_{j=1}^{n}(X_j - \mu) \ ,$$

which implies that

$$E\{(X_i - \mu)(\overline{X}_n - \mu)\} = \frac{1}{n}\sum_{j=1}^{n} E\{(X_i - \mu)(X_j - \mu)\} \ .$$

For every $i \neq j$ we have $E\{(X_i - \mu)(X_j - \mu)\} = \text{Cov}(X_i, X_j) = 0$ because the variables are assumed to be independent. For $i = j$ we have $E\{(X_i - \mu)(X_j - \mu)\} = E(X_i - \mu)^2 = \text{Var}(X_i) = \sigma^2$. Thus,

$$E\{(X_i - \mu)(\overline{X}_n - \mu)\} = \frac{\sigma^2}{n} .$$

Putting everything together we obtain that

$$E(X_i - \overline{X}_n)^2 = \sigma^2 - 2\frac{\sigma^2}{n} + \frac{\sigma^2}{n} = \sigma^2\frac{n-1}{n} .$$

This shows that $n(X_i - \overline{X}_n)^2/(n-1)$ is an unbiased estimator of σ^2. Therefore, the average of n such unbiased estimators is an unbiased estimator of σ^2. But the average of $n(X_i - \overline{X}_n)^2/(n-1)$ is the sample variance. Note that we have shown that every properly re-scaled squared residual is an unbiased estimator of the true variance. This result was used to show that the empirical variance is unbiased, but it is important in and of itself.

5.7 Mixture of distributions

So far, we have discussed individual distributions, though in practice we often deal with distributions that are combinations (mixtures) of multiple distributions. For example, the observed `rdi4p` distribution in the SHHS is likely a combination of `rdi4p` distributions for groups of individuals who are healthy, mildly affected, and strongly affected by sleep disrupted breathing (SDB). Another example can be found in genomic association studies where individual single-nucleotide polymorphism (SNP) are tested for association with a particular disease. The distribution of p-values (a measure of the strength of association between SNPs and disease) across SNPs is likely to have a mixture of at least two distributions, one corresponding to truly associated SNPs and one corresponding to truly non-associated SNPs. Thus, it makes sense to provide the theoretical definition of a mixture of distributions and the intuition behind sampling from such a mixture.

5.7.1 Definition and examples

If $f_1(\cdot)$ and $f_2(\cdot)$ are two pdfs and $\pi \in [0, 1]$ is a probability we say that a variable X follows a mixture distribution with mixture probability π if the pdf of X is

$$f(x) = \pi f_1(x) + (1 - \pi)f_2(x) .$$

It is relatively easy to show that $f(\cdot)$ is, indeed, a pdf. Let us calculate its mean and variance.

$$E(X) = \int xf(x)dx = \int x\{\pi f_1(x) + (1 - \pi)f_2(x)\}dx .$$

Therefore,

$$\begin{aligned} E(X) &= \int xf(x)dx = \int x\{\pi f_1(x) + (1-\pi)f_2(x)\}dx \\ &= \pi \int xf_1(x)dx + (1-\pi)\int xf_2(x)dx \\ &= \pi\mu_1 + (1-\pi)\mu_2 \ , \end{aligned}$$

where $\mu_1 = \int xf_1(x)dx$ and $\mu_2 = \int xf_2(x)dx$ are the means of the first and second distributions, respectively. This indicates that the mean of a mixture of distributions is the mixture of the means with the same mixing weights. Using similar calculations we obtain that

$$E(X^2) = \pi(\sigma_1^2 + \mu_1^2) + (1-\pi)(\sigma_2^2 + \mu_2^2) \ ,$$

where σ_1^2 and σ_2^2 are the variances of the distributions with pdfs $f_1(\cdot)$ and $f_2(\cdot)$, respectively. Therefore,

$$\text{Var}(X) = \pi(\sigma_1^2 + \mu_1^2) + (1-\pi)(\sigma_2^2 + \mu_2^2) - \{\pi\mu_1 + (1-\pi)\mu_2\}^2 \ .$$

This last formula is neither particularly informative nor worth remembering, but we thought that it would be a nice exercise to flex a few brain muscles. Also, these formulas can be useful if we want to estimate the mixture distribution parameters using a method of moments approach (admittedly, not yet discussed). Here we have discussed the mixture of two distributions, but this can be easily generalized to more than two distributions.

5.7.2 Simulating a mixture distribution

To better understand how a mixture can be obtained, below we discuss a mixture of two Normal distributions and plot both the theoretical pdf and the histogram of simulated data in Figure 5.5.

```
#Set a reasonable grid of values for the distribution
x=seq(-3,10,length=201)

#Obtain the pdf of the mixture of two Normals
dx=.7*dnorm(x)+.3*dnorm(x,m=5,sd=2)

#Simulate data from a mixture
#This starts by simulating from each distribution individually
X1<-rnorm(1000)
X2<-rnorm(1000,m=5,sd=2)

#And then mix them
#Here the U variable is choosing which distribution data will be simulated from
```

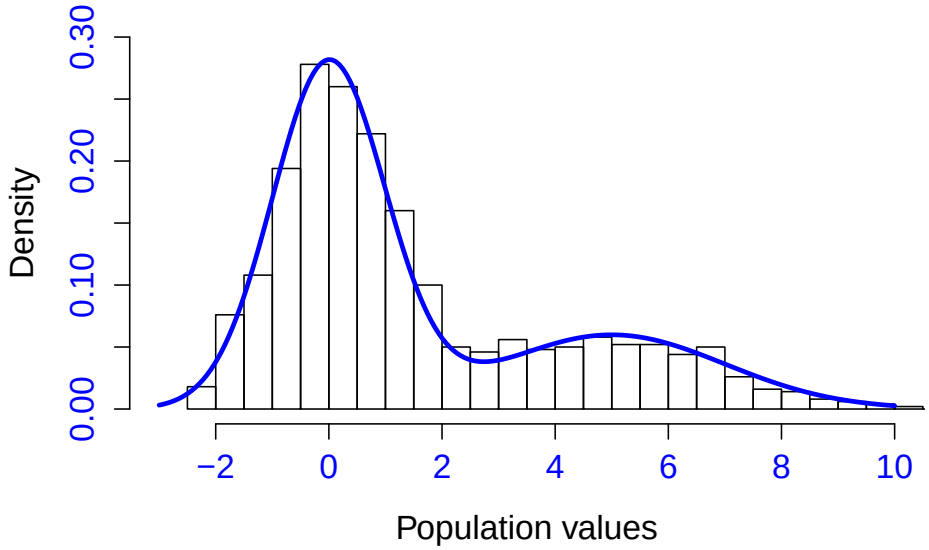


Figure 5.5: Theoretical and empirical distribution obtained from a sample of size 1000 for a mixture distribution between a $N(0,1)$ with probability 0.7 and a $N(5,4)$ with probability 0.3.

```
U<-rbinom(1000,1,p=.7)

#Here U can be only 0 or 1.
#If U=1 then simulate from the first distribution
#If U=0 then simulate from the second distribution
#This formula basically shows the mixing of the ingredients
#0.7 (the Bernoulli probability) controls the proportion from each ingredient
X=U*X1+(1-U)*X2

#Plot the histogram of realizations and the theoretical distribution
hist(X,breaks=30,probability=TRUE,xlim=c(-3,10),ylim=c(0,0.3),
     xlab="Population values",cex.lab=1.3,cex.axis=1.3,
     col.axis="blue",main="")
lines(x,dx,type="l",col="blue",lwd=3)
```

The way we built the simulation goes to the very core of how mixtures are being thought of. Indeed, the Bernoulli variable U is the variable that chooses from which population to sample at a particular time. The probability that governs the distribution of U (in this case 0.7) controls the proportion of samples from each population. In the case when we have a mixture of more than two populations the multinomial distribution can be used to select which population is being selected. For example, if we have three distributions and we want to mix them with probabilities 0.2, 0.5, and 0.3 then we would sample

```
U<-rmultinom(1, 1, c(0.2,0.5,0.3))
U
```

```
     [,1]
[1,]    1
[2,]    0
[3,]    0
```

In this particular instance this indicates that the sample from population 1 should be chosen.

Figure 5.5 indicates clearly the two distributions (note the two local maxima) and the result of the mixing. The first peak of the pdf is taller because the first distribution has a higher probability and has a smaller variance. Here we are giving a canonical example of mixture distribution plot and not all mixtures of distributions need to look like this one. Indeed, the second peak does not need to be apparent and the two distributions do not need to be symmetric. Take, for example, the distribution of `rdi4p` displayed in the right panel of Figure 5.3, which does not exhibit additional peaks, it is highly skewed and bounded by zero (because nobody can have a negative number of breathing-related events). However, it makes perfect sense to expect that there are sub-populations with distinct `rdi4p` distributions. Whether this should have any consequence in practice is debatable, as we are interested in simple parametric distributions that describe observed phenomena. Having the option to explore mixtures of distributions adds another tool to the biostatistical toolbox.

5.8 Problems

Problem 1. When at the free-throw line for two shots, a basketball player makes at least one free throw 90% of the time. 80% of the time, the player makes the first shot and 70% of the time she makes both shots.

a. Does it appear that the player's second shot success is independent of the first?
b. What is the conditional probability that the player makes the second shot given that she made the first?
c. What is the conditional probability that the player makes the second shot given that she missed the first?

Problem 2. Assume that an act of intercourse between an HIV infected person and a non-infected person results in a 1/500 probability of spreading the infection. How many acts of intercourse would an non-infected person have to have with an infected person to have a 10% probability of becoming infected? State the assumptions of your calculations.

Problem 3. You meet a person at the bus stop and strike up a conversation.

In the conversation, it is revealed that the person is a parent of two children and that one of the two children is a girl. However, you do not know the gender of the other child, nor whether the daughter she mentioned is the older or younger sibling.

a. What is the probability that the other sibling is a girl? What assumptions are you making to perform this calculation?
b. Later in the conversation, it becomes apparent that she was discussing the older sibling. Does this change your probability that the other sibling is a girl?

Problem 4. A particularly sadistic warden has three prisoners, A, B, and C. He tells prisoner C that the sentences are such that two prisoners will be executed and one let free, though he will not say who has what sentence. Prisoner C convinces the warden to tell him the identity of one of the prisoners to be executed. The warden has the following strategy, which prisoner C is aware of. If C is sentenced to be let free, the warden flips a coin to pick between A and B and tells prisoner C that person's sentence. If C is sentenced to be executed he gives the identity of whichever of A or B is also sentenced to be executed.

a. Does this new information about one of the other prisoners give prisoner C any more information about his sentence?
b. The warden offers to let prisoner C switch sentences with the other prisoner whose sentence he has not identified. Should he switch?

Problem 5. Given below are the sexes of the children of 7745 families of four children recorded in the archives of the Genealogical Society of the Church of Jesus Christ of Latter Day Saints in Salt Lake City, Utah. M indicates a male child and F indicates a female child.

Sequence	Number	Sequence	Number
MMMM	537	MFFM	526
MMMF	549	FMFM	498
MMFM	514	FFMM	490
MFMM	523	MFFF	429
FMMM	467	FMFF	451
MMFF	497	FFMF	456
MFMF	486	FFFM	441
FMMF	473	FFFF	408

a. Estimate the probability distribution of the number of male children, say X, in these families using these data by calculating proportions.
b. Find the expected value of X.
c. Find the variance of X.
d. Find the probability distribution of $\hat{p}$, where $\hat{p}$ is the proportion of children in each family who are male. Find the expected value of $\hat{p}$ and the variance of $\hat{p}$

Problem 6. Quality control experts estimate that the time (in years) until a specific electronic part from an assembly line fails follows (a specific instance of) the **Pareto** density

$$f(x) = \begin{cases} \frac{3}{x^4} & \text{for} \quad 1 < x < \infty \\ \\ 0 & \text{for} \quad x \leq 1 \end{cases}$$

a. What is the average failure time for components with this pdf?
b. What is the variance?
c. The general form of the Pareto density is given by

$$\frac{\beta \alpha^{\beta}}{x^{\beta+1}}$$

for $0 < \alpha < x$ and $\beta > 0$ (for fixed α and β). Calculate the mean and variance of the general Pareto density.

Problem 7. You are playing a game with a friend where you flip a coin and if it comes up heads you give her a dollar and if it comes up tails she gives you a dollar. You play the game ten times.

a. What is the expected total earnings for you? (Show your work; state your assumptions.)
b. What is the variance of your total earnings? (Show your work; state your assumptions.)
c. Suppose that the coin is biased and you have a .4 chance of winning for each flip. Repeat the calculations in parts a and b.

Problem 8. Note that the R code

```
temp <- matrix(sample(1 : 6, 1000 * 10, replace = TRUE), 1000)
xBar <- apply(temp, 1, mean)
```

produces 1,000 averages of 10 die rolls. That is, it is like taking ten dice, rolling them, averaging the results and repeating this 1,000 times.

a. Do this in R. Plot histograms of the averages.
b. Take the mean of `xBar`. What should this value be close to? (Explain your reasoning.)
c. Take the standard deviation of `xBar`. What should this value be close to? (Explain your reasoning.)

Problem 9. Note that the R code

```
xBar <- apply(matrix(runif(1000 * 10), 1000), 1, mean)
```

produces 1,000 averages of 10 uniforms.

a. Do this in R. Plot histograms of the averages.

b. Take the mean of `xBar`. What should this value be close to? (Explain your reasoning.)
c. Take the standard deviation of `xBar`. What should this value be close to? (Explain your reasoning.)

Problem 10. Consider two binary random variables X_1 and X_2 and let n_{11} be the number of times in a sample when both X_1 and X_2 are equal to zero, n_{12} the number of times when X_1 is equal to 0 and X_2 is equal to 1, n_{21} the number of times X_1 is equal to 1 and X_2 is equal to zero, and n_{22} the number of times when both X_1 and X_2 are equal to 1.

a. Derive the estimated covariance and correlation formulas for X_1 and X_2.
b. Calculate these quantities using the obtained formulas for the CHD, CVD, and sex variables in the SHHS.
c. Compare these results with those obtained directly from using `R` commands
d. Obtain bootstrap-based confidence intervals for the covariance and correlation.

Problem 11. Make scatterplots of the variables `Waist`, `rdi4p`, `StOnsetP`, `age`, `bmi` using the `pairs` function in `R` using the code

```
pairs(~waist+rdi4p+StOnsetP+age_s1+bmi_s1,pch=".",col=rgb(0,0,1,.2),
      cex=3,main="Two by two scatterplots")
```

Problem 12. For the variables `waist` and `bmi_s1` calculate the correlation between the transformed variables

```
waist_T<-waist*(waist>Tw)
bmi_T<-bmi_s1*(bmi_s1>Tb)
```

where `Tw` and `Tb` are the empirical quantiles of the distributions of `waist` and `bmi_s1` corresponding to the same probability. Plot this correlation as a function of the probability for a grid of probabilities between 0 and 1. Provide a similar plot for the variables

```
waist_S<-waist[(waist>Tw) & (bmi_s1>Tb)]
bmi_S<-bmi_s1[(waist>Tw) & (bmi_s1>Tb)]
```

and compare it with the plot for the previous variables. Explain the differences between the first two variables, labeled `_T`, and the second two variables, labeled `_S`. Compare and explain the differences between the two correlation plots.

Problem 13. Using the variables `waist`, `rdi4p`, `StOnsetP`, `age_s1`, `bmi_s1` in SHHS:

a. For every pair of variables calculate separately the mean of the product and the product of the means.
b. For every variable calculate the mean of the square of the variable and the square of the mean.

c. Produce estimators of the standard deviation of the mean and the mean of the standard deviations.
d. Produce confidence intervals for all these quantities using the nonparametric bootstrap.

Problem 14. Let X_i be iid random variables with variance σ^2 and let $V_i < -a(X_i - \overline{X}_n)^2$ be a class of estimators of σ^2 that depends on the scalar a. The Mean Square Error (MSE) of the estimator V_i is defined as

$$\text{MSE}(V_i) = E(V_i - \sigma^2)^2.$$

a. Show that $\text{MSE}(V_i) = \text{Bias}^2(V_i) + \text{Var}(V_i)$, where $\text{Bias}(V_i) = E(V_i) - \sigma^2$
b. Find the value of a that minimizes the $\text{MSE}(V_i)$
c. What is the ratio between the optimal MSE and the MSE of the unbiased estimator?

In general, we are interested in estimators that have low MSE, which is a combination of bias and variance. It is sometimes acceptable to replace an unbiased estimator with an estimator that has lower MSE. An important area of Biostatistics is concerned with finding the lowest variance estimators among the class of unbiased estimators. These estimators are called Uniform Minimum Variance Unbiased Estimators (UMVUE). Given the lack of imagination Biostatisticians have at naming extremely important quantities, it should not be surprising to find the same concept under many other, sexier, names.

Problem 15. Consider the mixture of n distributions

$$f(x) = \sum_{i=1}^{n} \pi_i f_i(x) \ ,$$

where $f_i(\cdot)$ are pdfs. Let X be a random variable with the pdf $f(\cdot)$, $\mu = E(X)$ and $\mu_i = \int x f_i(x) dx$.

a. Show that for any function of $M(\cdot)$ of X we have $E\{M(X)\} = \sum_{i=1}^{n} \pi_i \int M(x) f_i(x) dx$. If $M(X) = X^k$ then we call $E\{M(X)\} = E(X^k)$ the kth moment of the X random variable.
b. If $\mu_i = \int x f_i(x) dx$ is the mean of the ith component show that

$$E(X - \mu)^k = \sum_{i=1}^{n} \sum_{j=0}^{k} \binom{k}{j} \pi_j (\mu_i - \mu)^{k-j} \int (x - \mu_i)^j f_i(x) dx.$$

c. Use the previous result to re-derive the variance formula for a mixture of two variables.

Problem 16. We would like to fit a mixture of two Gamma distributions to the `rdi4p` data from SHHS. To do that we will use a method called moments-matching, or methods of moments (MoM). More precisely, we will obtain the theoretical moments and we will set them equal to the observed moments. Denote by X the random variable that led to the realizations of `rdi4p`.

a. Show that a mixture of two Gamma distributions depends on five parameters and write down explicitly those parameters. Denote the vector of these parameters θ.
b. Calculate the first five empirical moments of the distribution

$$\widehat{M_k} = \frac{1}{n}\sum_{i=1}^{n} X_i^k$$

and the first five expected values of X^k as functions of θ.
c. Set the empirical moments equal to the theoretical moments and solve for θ.
d. Plot the estimated mixture using MoM versus the histogram of the `rdi4p` data.

Chapter 6

Conditional distribution, Bayes rule, ROC

This chapter covers the following topics

- Conditional probabilities
- Bayes rule
- ROC and AUC

6.1 Conditional probabilities

Conditional probabilities are extremely important to understand, as they represent many phenomena that are observed in practice. They are relatively difficult to understand and operate with, but they occur naturally in applications. Below we provide a few motivating examples.

6.1.1 Motivation of conditional probabilities

A simple example of conditional probability can be given using the die rolling experiment. Note that the probability of getting a one when rolling a (standard, unbiased) die, without additional information, can be assumed to be one-sixth. However, if the additional information that the die roll was an odd number (hence 1, 3 or 5) is available, the probability of a one is now one-third, while the probability of getting a two is zero. Even though rolling the die is not an explicit health research experiment, per se, there are many research experiments that can be modeled by a biased K-faced die. Indeed, consider the case when one collects mental health data on individuals in a population and each individual's mental health status is characterized as either healthy (1) or affected. For those

with a mental health disorder the severity is characterized as mild (2), moderate (3), severe (4), or very severe (5).

Consider now the case when we are interested in the probability that a 30-year-old woman will develop breast cancer in the next 10 years. This can be rephrased as the probability of developing breast cancer in the next 10 years for a person, *conditional* on the fact that *she is a woman* and *her current age is* 30. Intuitively, we know that conditioning on the age of the person is important and it should be obvious that conditioning on sex is probably even more important. Indeed, according to www.cancer.org the probability for a 30-year-old woman to develop breast cancer in the next 10 years is 0.44% (or 1 in 227) while for a 50-year-old woman it is 2.38% (or 1 in 28). The probability at age 50 is 5.4 times larger than at age 30, indicating that conditioning on age of the woman really matters. The effect of conditioning on sex is far more dramatic. Indeed, according to www.cancer.org, breast cancer in men is about 100 times less common than in women. While data are not available for men in as much detail as for women, it is reasonable to assume that the difference between sexes in the probability of contracting breast cancer holds, at least approximately, across age groups.

To start modeling these words statistically, let us denote by X the random variable "a person develops breast cancer in the next 10 years." Then the probability we are interested in can be written as

$$P(X = 1|\ \text{sex} = 1,\ \text{age} = 30)\ .$$

Here $X = 1$ encodes that the result of the experiment will be "develop breast cancer in 10 years," `sex=1` encodes sex female, and `age=30` is self-explanatory. The vertical bar between `X=1` and `sex=1` is the conditioning statement. The event to the left of the bar is the event whose probability we are interested in, whereas what is to the right of the bar is what we condition on. What we condition on is restricting the event of interest to a particular subset of the population and helps define exactly what type of event we are trying to define. The take home message here is that language can be quite imperfect, but communication can be improved by insisting on using well defined formulas.

Another example is when we try to evaluate the probability of surviving more than one year for a man who is 50 years old and has an estimated glomerular filtration rate (eGFR) equal to 20. eGFR is a measure of kidney function based on the result of blood creatinine test, age, body size and sex www.kidney.org. If X is the random variable survival, then we are interested in calculating probabilities of the type

$$P(X > 1|\ \text{sex} = 0,\ \text{age} = 50,\ \text{eGFR} = 20)\ .$$

6.1.2 Definition of conditional probabilities

To understand and quantify such probabilities we need a formal definition of conditional probability as well as a deeper understanding of the conditioning

operation on data. Consider B to be an event of non-zero probability, that is $P(B) > 0$. Then the conditional probability of an event A given that B has occurred is

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \ .$$

If A and B are independent then

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)P(B)}{P(B)} = P(A) \ ,$$

indicating that if events are independent, then conditioning on one of them does not change (add information to) the probability of the other event. To be specific, consider the die roll example. In this case the event of interest is $A = \{1\}$ and the event we are conditioning on is $B = \{1, 3, 5\}$. Therefore the probability of obtaining a one given that the roll is odd is

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)}{P(B)} = \frac{1/6}{3/6} = \frac{1}{3} \ .$$

Here we used that $A \cap B = \{1\} \cap \{1, 3, 5\} = \{1\} = A$.

While such simple examples are useful, let us consider one more directly connected to health research. Consider the `rdi4p` and `age` variables in the SHHS dataset. Suppose we are interested in the probability that a random individual from SHHS has an `rdi4p` between 5 and 10 events per hour conditional on his or her age being from 70 and 80. We denote by A the event that a random individual from SHHS has an `rdi4p` between 5 and 10 and by B the event that a random individual from SHHS has an age between 70 and 80. Statistically, we are interested in the $P(A|B) = P(A \cap B)/P(B)$.

Just as in the case of the die experiment, the calculation is straightforward, but we want to build up the intuition behind the conditioning. We first plot the data in Figure 6.1, shown only up to an `rdi4p` value of 20, for plotting purposes.

The blue shaded area in Figure 6.1, from `age_s1` 70 to 80 is the event we condition on, B, which is that the individual is from 70 to 80 years old. In this dataset there are subjects who are in this age range, more precisely there are exactly 1543 individuals in this age range, out of a total of 5804 individuals who had their age recorded. Thus, the frequency of subjects from 70 to 80 years of age in the SHHS is $P(B) = 1543/5804 = .266$. Here we use the estimated probability and the theoretical probability interchangeably to build up intuition, though one should remember that these are estimates of the true probabilities and not theoretical probabilities. Now, let us visualize the conditioning action on the data. Recall that we are interested in those subjects who have an `rdi4p` from 5 to 10, conditional (or given that) individuals are from 70 to 80 years old. Figure 6.2 provides a visualization of this set of individuals

The red band in Figure 6.2 indicates all subjects whose `rdi4p` is from 5 to 10 events per hour, while the intersection between the red and the blue bands

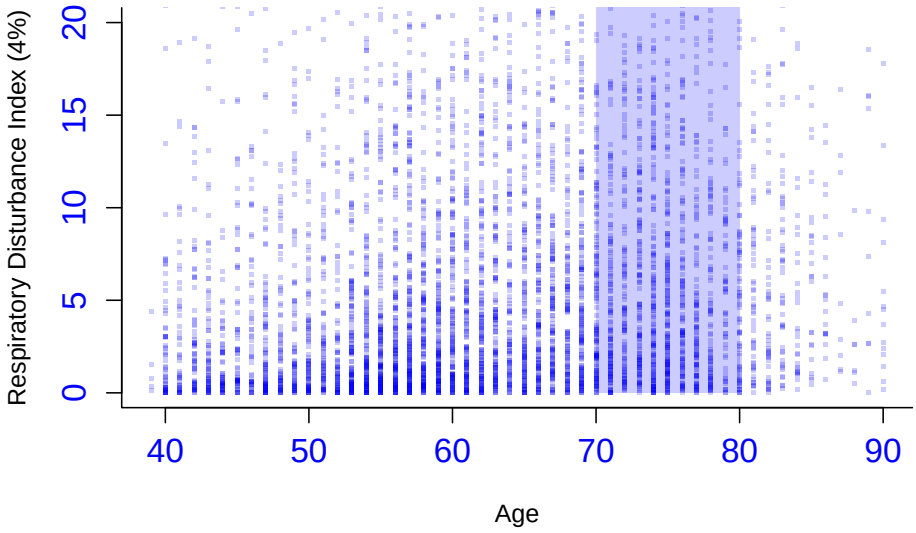


Figure 6.1: Respiratory Disturbance Index at 4% oxygen desaturation versus age in the SHHS. The y-axis is cut at 20 events per hour for presentation purposes. The blue shaded area indicates individuals from 70 to 80 years old.

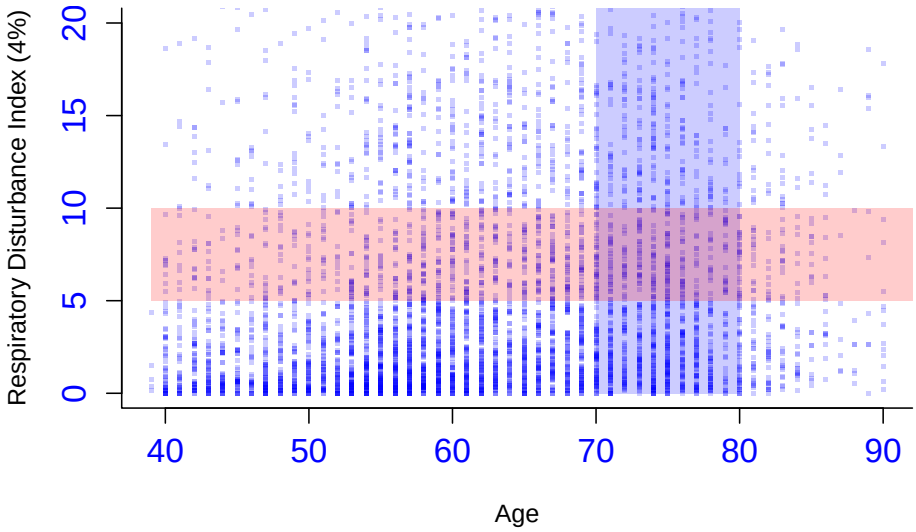


Figure 6.2: Respiratory Disturbance Index at 4% oxygen desaturation versus age in the SHHS. The blue shaded area indicates individuals from age 70 to 80. The red shaded area indicates individuals with an `rdi4p` from 5 to 10 events per hour.

represents those subjects who are from 70 to 80 years of age *and* have an `rdi4p` from 5 to 10 events per hour. Note that if we were interested in the probability that a random subject is both from 70 to 80 years of age and has an `rdi4p` from 5 to 10 then we need to count all subjects in the intersection between the red and the blue bands and divide by the number of subjects in the entire dataset. To do that, we build indices, as we have described before

```
index_age7080_rdip4p510<-((age_s1>=70) & (age_s1<=80) & (rdi4p>=5) & (rdi4p<=10))
n_int=sum(index_age7080_rdip4p510)
```

and we obtain that there are 316 individuals in this set. Thus,

$$P(A \cap B) = \frac{316}{5804} = 0.054 \, .$$

The conditioning statement says to not divide by the total number of subjects in the data set, but restrict the calculation only to individuals who are from 70 to 80 years old. More precisely, count how many subjects are in the intersection and divide by the number of subjects in the blue shaded area. Of course, the number of subjects in the blue shaded area, 1543, is much smaller than the number of subjects in the SHHS data set. Thus, we get

$$P(A|B) = \frac{316}{1543} = 0.205 >> 0.054 \, .$$

Necesarily, $P(A|B) > P(A \cap B)$ because $P(B)$, the probability of the event we condition on is smaller than 1. It is guaranteed to be much larger if $P(B)$ is small. Note, however, that $P(A)$ can be either higher or lower than $P(A|B)$.

Let us see how the same probability could have been obtained by literally following the definition of conditional probability

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{316/5804}{1543/5804} = \frac{316}{1543} = 0.205 \, ,$$

where the number of subjects in the SHHS population is getting canceled out.

In biostatistics we often refer to $P(A)$ as the marginal (or unconditional) probability of the event A, to $P(A \cap B)$ as the joint probability of the A and B events, and to $P(A|B)$ as the conditional probability of the event A given B. These are fundamental concepts that will re-occur frequently. Understanding the conditional probability will be essential to understanding complex models that can be represented by simple conditional distributions. But, more about this later.

This example makes a few intuitive points:

- when estimating conditional probabilities, the `and` operation (the intersection) reduces the sample size we are focusing on;
- the conditional probability is always higher than the probability of the intersection of two events;

- the difference between conditioning and taking the intersection of the two events is that we restrict the space to the event we condition on (e.g., the subjects from 70 to 80 years old);
- more complex conditional probabilities are harder to represent visually, but the principles remain the same;
- plotting the data and thinking through concepts can help make verbal statements precise.

6.1.3 Conditional probabilities and mass functions

Let us continue with the empirical intuition behind conditional events built in the previous section. We would like to imagine the case when the event we are conditioning on shrinks (think a one-year instead of the 10-year age interval). In this case, the number of individuals in that interval (blue shaded area) can be reduced dramatically, while the intersection between the red and blue shaded area is also reduced. In the example below, both vanish relatively quickly. Thus, for realistic cases, such as the SHHS, there is a real problem estimating conditional probabilities when the events we condition on become very small. However, we would still like to define, interpret, and compute the distribution of `rdi4p` for a specific age, say 70 years. This is the limit of empirical thinking and the reason to introduce theoretical conditional probabilities.

Consider the case when we have two random variables, X and Y, and want to introduce the theoretical definition of conditional probabilities. Let $f_{X,Y}(x,y)$ be a bivariate density or mass function for random variables X and Y. Let $f_X(x)$ and $f_Y(y)$ be the associated marginal mass function or densities disregarding the other variables

$$f_Y(y) = \int f_{X,Y}(x,y)dx \quad \text{or} \quad f_Y(y) = \sum_x f_{X,Y}(x,y)dx \ ,$$

where the integral is used if X is continuous and the sum is used if X is discrete. Then the **conditional** density or mass function of X *given that* $Y = y$ is

$$f_{X|Y}(x|y) = f_{X,Y}(x,y)/f_Y(y)$$

Here we have used indices to indicate exactly what distribution the density refers to. In cases when there is no danger of misinterpretation of densities, indices could be dropped, while remembering that the function $f(\cdot)$ may refer to completely different distributions. Bayesians, have a particularly simple notation, where $f_{X|Y}(x|y) = [x|y]$, $f_{X,Y}(x,y) = [x,y]$ and $f_Y(y) = [y]$. Thus, in this decluttered notation the same expression can be written as

$$[x|y] = [x,y]/[y]$$

or, equivalently,

$$[x,y] = [x|y][y] = [y|x][x] \ .$$

This notation is especially clean and useful when the number and complexity of conditional distributions increases. For pedagogical reasons, we will be using the indexing notation, whenever necessary.

When both random variables are discrete, the definition of conditional probability is exactly the definition for conditional events where A is the event that $X = x_0$ and B is the event that $Y = y_0$. However, the continuous definition is harder to motivate, since the events $X = x_0$ and $Y = y_0$ have probability 0 (recall the SHHS example). However, a useful motivation can be provided by taking the appropriate limits by defining $A = \{X \leq x_0\}$ and $B = \{Y \in [y_0, y_0 + \epsilon]\}$. In the SHHS, A could be the event that `rdi4p` is less than x_0, where x_0 could be, for example, 5, 10, or 15 events per hour. The event B could be the event that `age` is between y_0 and a small increment in age $y_0 + \epsilon$. Here ϵ is the width of the blue shaded area in Figures 6.1 and 6.2.

Because the event $Y \in [y_0, y_0 + \epsilon]$ has a non-zero probability we can write the probability of the conditional event $\{X \leq x_0 | Y \in [y_0, y_0 + \epsilon]\}$ as

$$
\begin{aligned}
P(X \leq x_0 | Y \in [y_0, y_0 + \epsilon]) = P(A|B) &= \frac{P(A \cap B)}{P(B)} \\
&= \frac{P(X \leq x_0, Y \in [y_0, y_0 + \epsilon])}{P(Y \in [y_0, y_0 + \epsilon])} \\
&= \frac{\int_{y_0}^{y_0+\epsilon} \int_{-\infty}^{x_0} f_{X,Y}(x, y) dx dy}{\int_{y_0}^{y_0+\epsilon} f_Y(y) dy} \\
&= \frac{\int_{y_0}^{y_0+\epsilon} \int_{-\infty}^{x_0} f(x, y) dx dy / \epsilon}{\int_{y_0}^{y_0+\epsilon} f(y) dy / \epsilon} .
\end{aligned}
$$

The first line of equalities is simply the definition of conditional probabilities. The equality on the second line is just the explicit form of the conditional probability for these two particular events, A and B. The third line equality is the hardest one to explain, but recall that the probability of joint events for X and Y is governed by the joint pdf $f_{X,Y}(x, y)$. To obtain the $P(X \leq x_0, Y \in [y_0, y_0 + \epsilon])$ we need to integrate $f_{X,Y}(x, y)$ over the domain $D = (-\infty, x_0] \times [y_0, y_0 + \epsilon]$. Therefore,

$$
P(X \leq x_0, Y \in [y_0, y_0 + \epsilon]) = \int \int_D f_{X,Y}(x, y) dx dy = \int_{y_0}^{y_0+\epsilon} \int_{-\infty}^{x_0} f_{X,Y}(x, y) dx dy.
$$

A similar argument holds for the equality $P(Y \in [y_0, y_0 + \epsilon]) = \int_{y_0}^{y_0+\epsilon} f_Y(y) dy$. The last equality was obtained by dividing both the numerator and the denomi-

nator by ϵ. We will investigate what happens to the numerator and denominator when we let $\epsilon \to 0$. The numerator can be written as

$$\frac{\int_{-\infty}^{y_0+\epsilon}\int_{\infty}^{x_0} f_{X,Y}(x,y)dxdy - \int_{-\infty}^{y_0}\int_{-\infty}^{x_0} f_{X,Y}(x,y)dxdy}{\epsilon} = \frac{g_1(y_0+\epsilon) - g_1(y_0)}{\epsilon} ,$$

where $g_1(y_0) = \int_{-\infty}^{y_0}\int_{-\infty}^{x_0} f_{X,Y}(x,y)dxdy$.

Thus, when we let $\epsilon \to 0$ we obtain the limit $g_1'(y_0) = \int_{-\infty}^{x_0} f_{X,Y}(x,y_0)dx$. This last equality follows from the so called fundamental theorem of calculus; for any integrable function $h(\cdot)$

$$\frac{\partial}{\partial y}\int_{-\infty}^{y} h(t)dt = h(y) .$$

Similarly, for the denominator we have

$$\frac{\int_{-\infty}^{y_0+\epsilon} f_Y(y)dxdy - \int_{-\infty}^{y_0} f_Y(y)dxdy}{\epsilon} = \frac{g_2(y_0+\epsilon) - g_2(y_0)}{\epsilon} ,$$

where $g_2(y_0) = \int_{-\infty}^{y_0} f_Y(y)dy$. When we let let $\epsilon \to 0$ we obtain the limit $g_2'(y_0) = f_Y(y_0)$, or the marginal pdf of Y evaluated at y_0. Putting all these results together we obtain that

$$\lim_{\epsilon\to\infty} P(X \le x_0 | Y \in [y_0, y_0+\epsilon]) = \int_{-\infty}^{x_0} \frac{f_{X,Y}(x,y_0)}{f_Y(y_0)}dx = \int_{-\infty}^{x_0} f_{X|Y=y_0}(x)dx ,$$

where $f_{X|Y=y_0} = f_{X,Y}(x,y_0)/f_Y(y_0)$ is, by definition the conditional pdf of X given $Y = y_0$. These results indicate that the cumulative distribution function of the variable X conditional on the ever shrinking event $Y \in [y_0, y_0+\epsilon]$ can be obtained by integrating the conditional pdf $f_{X|Y=y_0}(x)$. Another way of thinking about this equality is that

$$P(X \le x_0 | Y = y_0) = \int_{-\infty}^{x_0} \frac{f_{X,Y}(x,y_0)}{f_Y(y_0)}dx$$

and taking the derivative with respect to x_0 we obtain

$$f_{X|Y}(x_0,y_0) = \frac{f_{X,Y}(x_0,y_0)}{f_Y(y_0)} .$$

This indicates that the definition of the conditional pdf makes sense and that it has a practical interpretation. As we mentioned, this is a case when the intuition built on empirical evidence can break down. However, the intuition built up from the theory above gives us a way to estimate continuous conditional probabilities from observed data: simply take a small ϵ (and hopefully there are enough data to actually estimate the probabilities).

In practice, the recipe for obtaining conditional pdfs is extremely simple if the joint pdf is known. More precisely

$$f_{X|Y=y}(x) = \frac{f_{X,Y}(x,y)}{f_Y(y)} = \frac{f_{X,Y}(x,y)}{\int f_{X,Y}(x,y)dx} .$$

Geometrically, the conditional density is obtained from the joint density by taking the relevant slice of the joint density and appropriately renormalizing it. The reason renormalization is necessary is to ensure that the conditional density is a pdf that integrates to 1. Indeed, it is easy to check that

$$\int f_{X|Y=y}(x)dx = \int \frac{f_{X,Y}(x,y)}{f_Y(y)}dx = \frac{1}{f_Y(y)} \int f_{X,Y}(x,y)dx = \frac{f_Y(y)}{f_Y(y)} = 1 ,$$

and this equality holds for every value y of Y we condition on. This is an almost trivial set of equalities, but they explain the normalizing constant idea, where the normalizing constant is the marginal pdf of the variable we condition on.

In this case, the normalizing constant is the marginal pdf of Y, $f_Y(y)$. Sometimes, for simplicity we can write $f_{X|Y=y}(x) \propto f_{X,Y}(x,y)$ to indicate that the conditional pdf is proportional to the joint pdf up to the normalizing constant $f_Y(y)$.

6.1.3.1 Example: Gamma conditionals

Consider the bivariate pdf

$$f_{X,Y}(x,y) = ye^{-xy-y},$$

where $x \geq 0, y \geq 0$. We would first like to plot this pdf using 3D plotting tools available in `R`

```
#Set an equally spaced grid between 0 and 4
#Do the same for y
x=seq(0,4,length=40)
y=x
#Store the pdf values in a matrix
matxy=matrix(rep(0,1600),ncol=40)
#Calculate and store the pdf at every location in the bivariate grid
for (i in 1:40)
  {for (j in 1:40){matxy[i,j]=y[j]*exp(-x[i]*y[j]-y[j])}}
```

Recall that the bivariate pdf represents the concentration of probability for the bivariate random vector (X, Y). In this example, the two random variables are not independent because $f(x,y) \neq g(x)h(y)$ for any $g(\cdot)$ or $h(\cdot)$. That is, the bivariate pdf cannot be *separated* as a product between a function that depends only on x and a function that depends only on y. Plotting 3D objects in `R` can be done in different ways. Here we first explore the `persp` function, though we

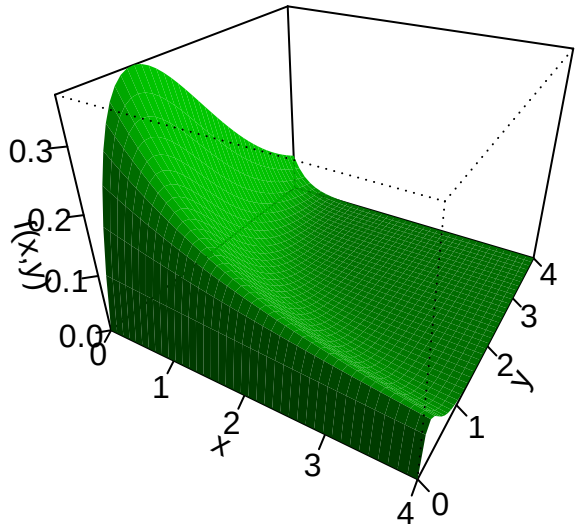


Figure 6.3: Three dimensional plot representing the joint pdf $f_{X,Y}(x,y) = ye^{-xy-y}$.

will also explore heat and contour maps. Recall that we are really plotting are points $\{x, y, f_{X,Y}(x,y)\}$ in three dimensional space, where the third dimension $f_{X,Y}(x,y)$ is the value of the pdf at location (x,y). The plot is done on a mesh of points in the interesting part of the function, that is, where it is not zero or very close to zero.

```
persp(x, y, matxy, theta = 30, phi = 30, col = c("green3"),
      ltheta = 120, shade = 0.9, expand="0.75",zlab="f(x,y)",
      ticktype = "detailed",border=NA)
```

We can also create an interactive 3D plot using the function `plot_ly` in the `R` package `plotly`. (This only shows up in the HTML version but not the PDF. If you are viewing the pdf, try running the code yourself!)

```
library(plotly)
plot_ly(x = x, y = y, z = matxy) %>% add_surface()
```

The pdf displayed in Figures 6.3 and 6.4 (if you are viewing this in the HTML version) indicates that areas of high probability are close to $x = 0$ and $y = 1$ with a shallower slope from $(0,1)$ to $(4,1)$ (x-direction from the maximum) and a steeper slope from $(0,1)$ to $(0,4)$. The distribution is not symmetric and is bounded by 0 both in the x and y direction. This indicates that both the X and Y random variables do not take negative values.

Figure 6.5 displays the same plot using a heat map, as implemented in the `R` package `fields`. The package allows for the color bar representing the values

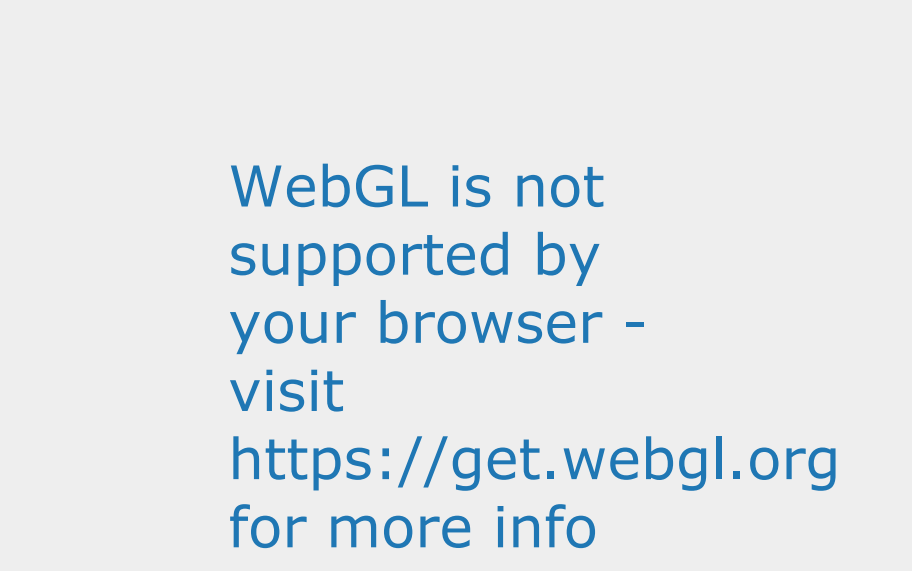


Figure 6.4: Three dimensional interactive plot representing the joint pdf $f_{X,Y}(x,y) = ye^{-xy-y}$. Not available in PDF version.

of $f_{X,Y}(x,y)$ to be attached to the plot. Other functions in `R` that render 3D objects include `plot3D`, `surface3d`, `contour` and `rgl`.

```
library(fields)
image.plot(x,y,matxy)
```

So far, we have rendered the joint pdf of two variables. Let us consider the case when we are interested in the conditional pdf of the X variable given that $Y = 2$. To do that we simply write down the formula

$$f_{X|Y=2}(x) = \frac{f_{X,Y}(x,2)}{f_Y(2)} = \frac{2e^{-2x-2}}{f_Y(2)} \ .$$

Here

$$f_Y(2) = \int_0^\infty f_{X,Y}(x,2)dx = \int_0^\infty (2e^{-2x-2})dx = 2e^{-2} \times \{-0.5e^{-2x}\}\Big|_0^\infty = e^{-2} \ .$$

This indicates that

$$f_{X|Y=2}(x) = 2e^{-2x} \ ,$$

for $x \geq 0$ and 0 otherwise. This can be recognized as the exponential distribution with mean $1/2$.

We have mentioned that the conditional density is obtained from the joint density by taking the relevant slice of the joint density and appropriately renormalizing it. Let us visualize this process for the calculations that we have just

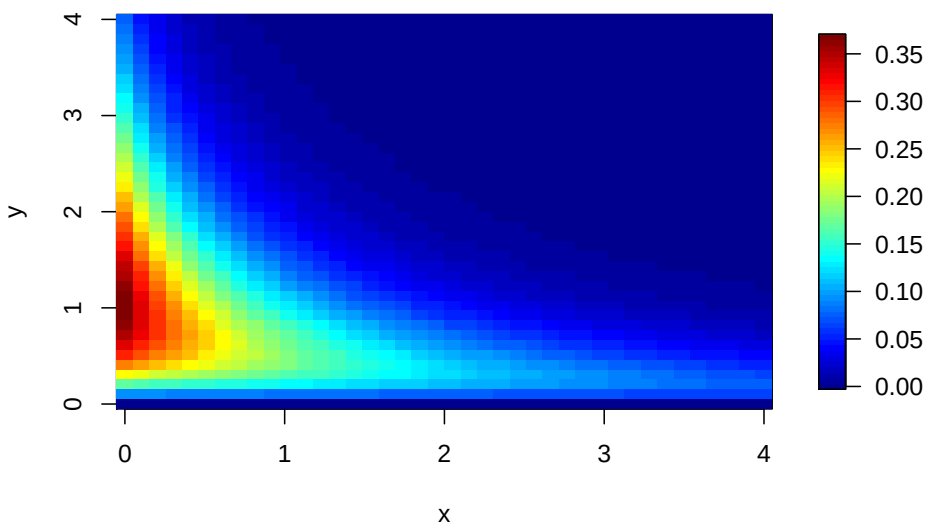


Figure 6.5: Heat map plot representing the joint pdf $f_{X,Y}(x,y) = ye^{-xy-y}$.

conducted. Recall that we conditioned on $Y = 2$, which is indicated on the bivariate plot as a black dashed line. Figure 6.6 displays the same information as Figure 6.5, but it adds the line (or slice) at the value $Y = 2$ on which we are conditioning.

Once $Y = 2$, we can imagine taking a slice of the bivariate heat map along the dashed line and making a plot of the values along the line. These values are shown as the dashed black line in Figure 6.7 and represent the unnormalized conditional density. Indeed, the density does not integrate to 1; and, in fact, we know that it integrates to $e^{-2} \approx 0.135$. To obtain a proper conditional pdf this function needs to be divided by e^{-2} and the resulting function is shown in Figure 6.7 as a solid blue line. This is the conditional density of X given $Y = 2$.

```
plot(x,2*exp(-2*x-2),lwd=3,type="l",lty=2,ylim=c(0,2),ylab="",
     bty="n",cex.lab=1.3,cex.axis=1.3, col.axis="blue")
lines(x,2*exp(-2*x),lwd=3,col="blue")
```

In general, for $Y = y_0$, we have

$$f_Y(y_0) = \int_0^\infty f_{X,Y}(x,y_0)dx = e^{-y_0}\int_0^\infty y_0 e^{-xy_0}dx = -e^{-y_0}\int_0^\infty \frac{\partial}{\partial x}(e^{-xy_0})dx\,,$$

which implies that

$$f_Y(y_0) = -e^{-y_0}(e^{-xy_0})\Big|_0^\infty = e^{-y_0}\,.$$

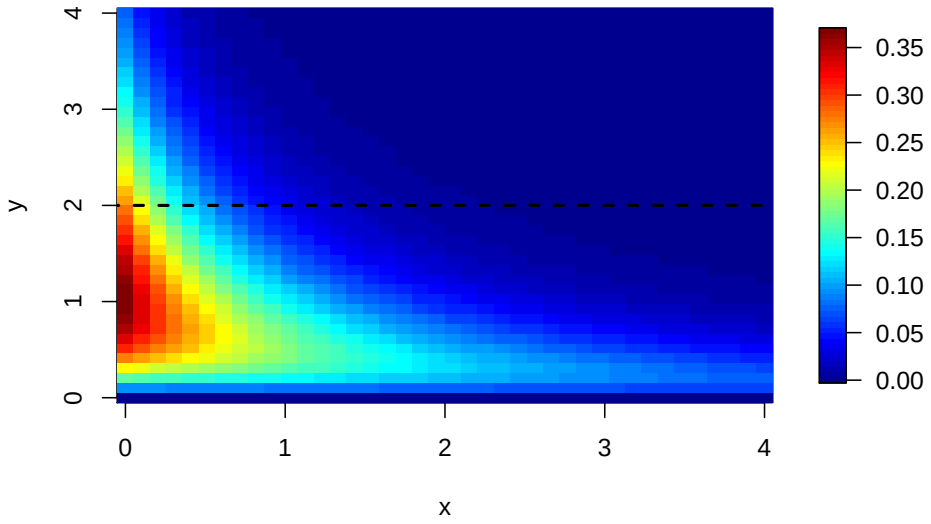


Figure 6.6: Heat map plot of the joint pdf $f_{X,Y}(x,y) = ye^{-xy-y}$ and slice at $Y = 2$.

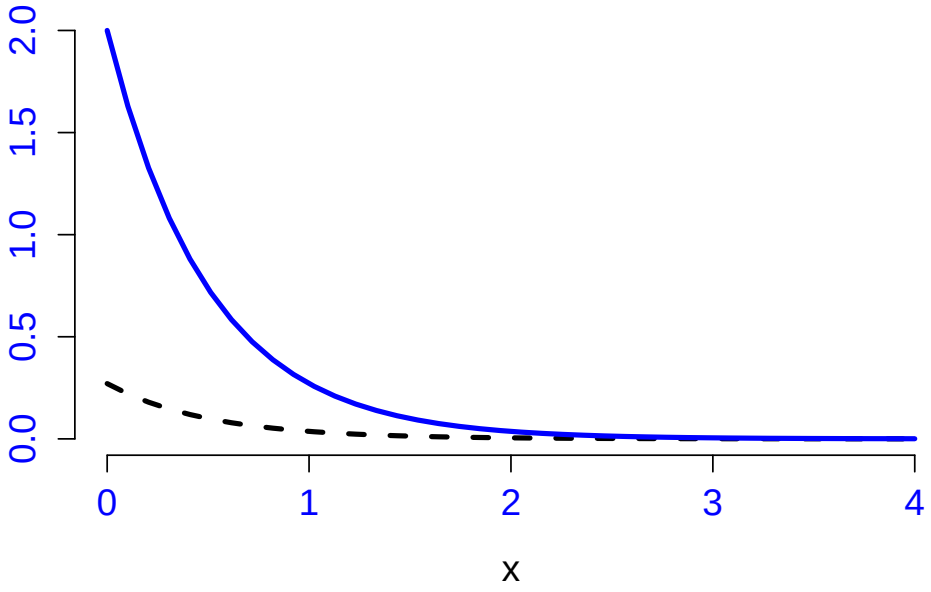


Figure 6.7: Unnormalized (dashed black line) and normalized (solid blue line) conditional pdf for $X|Y = 2$ when the joint pdf is $f_{X,Y}(x,y) = ye^{-xy-y}$.

Therefore, for any $y_0 > 0$ we have

$$f_{X|Y=y_0}(x) = \frac{f_{X,Y}(x, y_0)}{f_Y(y_0)} = \frac{y_0 e^{-xy_0 - y_0}}{e^{-y_0}} = y_0 e^{-xy_0} \ ,$$

indicating that the conditional distribution of $X|Y = y_0$ is an exponential distribution with mean $1/y_0$.

6.1.3.2 Example: uniform disk

Consider now the case when

$$f_{X,Y}(x, y) = \frac{1}{\pi r^2} \quad \text{for} \quad \{(x, y) : x^2 + y^2 \leq r\} \ ,$$

and 0 otherwise. This joint pdf is uniform (has the same value) at every point in the disk centered at 0 with radius r. We would like to know what is the conditional density for X given that $Y = 0$. When $Y = 0$ X can take any value between $-r$ and r. Because the density is uniform on the disk, geometrically it makes sense for the conditional density to also be uniform on its domain, $[-r, r]$. We write this as $f_{X|Y=0} \propto 1$, meaning that $f_{X|Y=0}(x) = c$ for $x \in [-r, r]$, where c is a constant. Because the conditional pdf has to integrate to 1 it follows that

$$f_{X|Y=0}(x) = \frac{1}{2r} \quad \text{for} \quad \{x : -r \leq x \leq r\} \ .$$

The same result can be obtained following the definition of the conditional pdf.

6.2 Bayes rule

Bayes rule is a fundamental rule of probability that allows the calculation of the conditional probability $P(A|B)$ when it is easy to obtain information about $P(B|A)$. (As an aside, there is a fair amount of deeply entrenched warfare over whether it should be Bayes, Bayes' or Bayes's rule. We guess that Bayes's is probably correct as there was only one Bayes in question and it is his rule. However, many in the field have taken the universally-known-to-be-wrong step of dropping the apostrophe altogether, a convention we adopt; good thing we are statisticians and not grammarians.)

In some sense, Bayes rule reverses the order of conditioning and provides the formulas for conducting the calculations. In a more general context, Bayes rule relates the conditional density $f_{Y|X=x}(y)$ to the conditional density $f_{X|Y=y}(x)$ and the marginal density $f_Y(y)$. For continuous random variables the Bayes rule is

$$f_{Y|X=x}(y) = \frac{f_{X|Y=y}(x) f_Y(y)}{\int f_{X|Y=t}(x) f_Y(t) dt} \ .$$

This result is an immediate consequence of the definition of the conditional pdf, which indicates

$$f_{Y|X=x}(y)f_X(x) = f_{X,Y}(x,y) = f_{X|Y=y}(x)f_Y(y) \ .$$

By recalling that the marginal density of X is $f_X(x) = \int f_{X,Y}(x,t)dt = \int f_{X|Y=t}(x)f(t)dt$, one could ask why the integral is with respect to t and not with respect to y. The reason is that we need to integrate relative to a general variable, as x and y are fixed for the purpose of this calculation. If the variable Y is discrete then the Bayes formula can be written as

$$f_{Y|X=x}(y) = \frac{f_{X|Y=y}(x)f_Y(y)}{\sum_t f_{X|Y=t}(x)f_Y(t)} \ .$$

A particular case of the Bayes rule or formula is when we have two sets A and B. In this case the formula becomes

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)} = \frac{P(A|B)P(B)}{P(A|B)P(B) + P(A|B^c)P(B^c)} \ .$$

The last equality can be easily obtained from the formulas

$$P(A) = P(A \cap B) + P(A \cap B^c) = P(A|B)P(B) + P(A|B^c)P(B^c) \ .$$

The first equality is true because $A = A \cap (B \cup B^c)$ and $B \cap B^c = \emptyset$, while the second equality follows directly from the definition of the conditional probabilities. The proof of this result could also be provided using the discrete version of the Bayes rule on the variables X and Y, which are the indicators that the events A and B occur, respectively. This formula indicates that in order to calculate the conditional probability $P(B|A)$ we need $P(A|B)$, and the marginal (prior) probabilities, $P(B)$ and $P(A)$. Or, if $P(A)$ is not readily available we would need $P(A|B^c)$ instead.

6.2.1 Example: diagnostic tests

Diagnostic tests are widely used in medical practice to identify individuals, who are at higher risk of having a disease, and are often used for determining the appropriate treatment. For example, according to the National Institute of Diabetes and Digestive and Kidney Diseases "the hemoglobin A1C (HbA1c) test is a blood test that provides information about a person's average levels of blood glucose over the past 3 months." A person is diagnosed with diabetes if the result of the HbA1c test is above 6.5%. Another example is the routine Systolic Blood Pressure (SBP) test, which indicates hypertension if SBP is greater than 140 millimeters of mercury (mm Hg).

Diagnostic tests have common features: (1) they are applied to individuals in a population, (2) they are designed to partition the population in individuals

who test positive and individuals who test negative, and (3) they can lead to further medical actions for individuals who test positive (e.g., exceed a particular threshold).

Consider the case when the test is applied to n individuals sampled from a population. The diagram in Figure 6.8 provides a depiction of both the disease state and the four possible combinations between test outcomes and the true state of disease. If the person has the disease and the test is positive (indicating that the person has the disease), then this is called a true positive. We denote by a the number of true positive tests in the population. If the person does not have the disease and the test is negative, then the result is called a true negative and we denote by d the number of true negatives. Both true positives and true negatives are desired results of a diagnostic test. Unfortunately, diagnostic tests can also lead to two types of errors. First, individuals who do not have the disease may have a positive test result indicating that they actually have the disease. This is called a false positive (a false alarm) and may lead to further testing and unnecessary treatment for an individual who is healthy. We denote by b the number of individuals who have been falsely identified as having the disease when they actually do not have it. Second, individuals who have the disease may have a negative test result indicating that they do not have the disease. This is called a false negative and may lead to lack of necessary treatment and unnecessary complications later on. We denote by c the number of individuals who have been incorrectly classified as not having the disease when they actually have it. Of course, better diagnostic tests would have a smaller rate of false negative and false positive results in the population. In general, we would like both the number of false positive and false negative findings to be 0. The number of false positives could easily be reduced to 0 by declaring that nobody has the disease. Of course, this would have a negative effect on the number of false negatives, which would be equal to the number of individuals in the population that have the disease. Similarly, and just as unhelpful, the number of false negatives could be reduced to zero by declaring that everybody has the disease. Of course, this would inflate the number of false positives, which would become equal to the number of people in the population who do not have the disease.

For convenience, denote by $T+$ and $T-$ the events that the result of a diagnostic test is positive or negative, respectively. Let $D+$ and $D-$ be the event that the subject of the test has or does not have the disease, respectively. The *sensitivity of the test* is defined as $P(T+|D+)$, the probability that the test is positive conditional on the fact that the subject actually has the disease. The *specificity of the test* is defined as $P(T-|D-)$, the probability that the test is negative conditional on the fact that the subject does not have the disease. In the example shown in the diagram above a reasonable estimator of the sensitivity of the test is

$$\widehat{P}(T+|D+) = \frac{a}{a+c} \ ,$$

the proportion of individuals who tested positive among those who have the

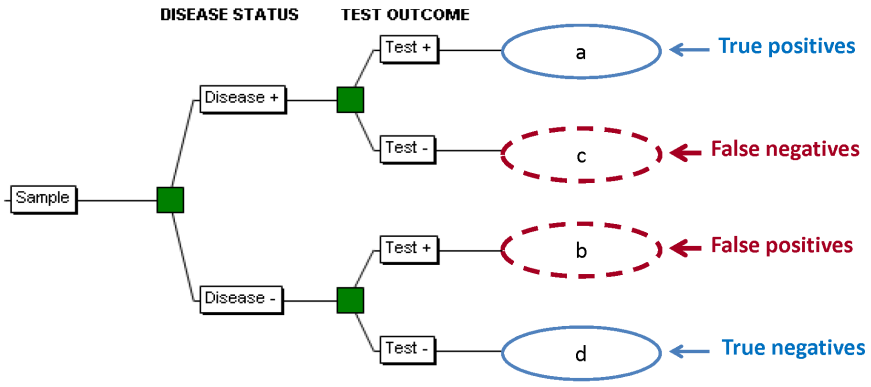


Figure 6.8: Diagram of a diagnostic test showing the possible outcomes: true positives, false negatives, false positives, and true negatives.

disease. There are a individuals who have tested positive among those who have the disease and there are $a+c$ individuals who have the disease. Similarly, a reasonable estimator of the specificity of the test is

$$\widehat{P}(T-|D-) = \frac{d}{b+d} ,$$

the proportion of individuals who test negative among those who do not have the disease. There are b individuals who have tested negative among the individuals who do not have the disease and there are $b+d$ individuals who do not have the disease.

The *positive predictive value (PPV)* of the test is $P(D+|T+)$, the probability that the subject has the disease conditional on testing positive. The *negative predictive value (NPV)* of the test is $P(D-|T-)$, the probability that the subject does not have the disease conditional on testing negative. The *prevalence of the disease* is $P(D+)$, the marginal probability of the disease. An estimator of PPV is

$$\widehat{P}(D+|T+) = \frac{a}{a+b} ,$$

where a is the number of subjects who have the disease among those who tested positive and there are $a+b$ individuals who tested positive. Similarly, an estimator of NPV is

$$\widehat{P}(D-|T-) = \frac{d}{c+d} ,$$

where d is the number of subjects who do not have the disease among those who tested negative and $c+d$ is the number of subjects who tested negative. An estimator of the prevalence of the disease is

$$\widehat{P}(D+) = \frac{a+c}{a+b+c+d} ,$$

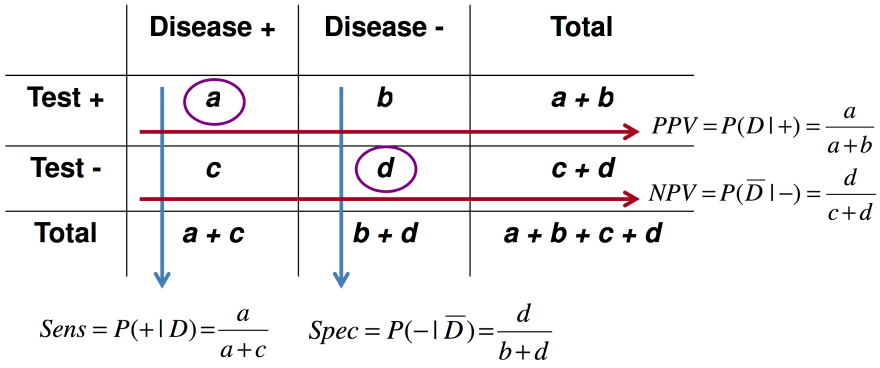


Figure 6.9: Two by two table showing the summary of results from a diagnostic test procedure combined with its characteristics: sensitivity, specificity, PPV, and NPV.

where $a+c$ is the number of individuals who have the disease and $n = a+b+c+d$ is the number of individuals in the study.

All these definitions can be collected in one concise two by two table, as shown in Figure 6.9. To make things more interpretable we provide now two examples.

6.2.1.1 Example: testing for HIV

A study comparing the efficacy of HIV tests concluded that HIV antibody tests have a sensitivity of 99.7% and a specificity of 98.5%. Suppose that a subject, from a population with a 0.1% prevalence of HIV, receives a positive test result. We are interested in calculating the probability that this subject has HIV. Mathematically, we would like to calculate $P(D+|T+)$, the probability of being HIV positive $(D+)$ given that one test was positive $T+$. What is known is the sensitivity of the test, $P(T+|D+) = 0.997$, the specificity of the test, $P(T-|D-) = 0.985$ and the prevalence of the disease in the target population, $P(D+) = 0.001$. Using Bayes formula we can calculate the quantity that we are interested in

$$P(D+|T+) = \frac{P(T+|D+)P(D+)}{P(T+|D+)P(D+) + P(T+|D-)P(D-)}$$

$$= \frac{P(T+|D+)P(D+)}{P(T+|D+)P(D+) + \{1 - P(T-|D-)\}\{1 - P(D+)\}}$$

$$= \frac{.997 \times .001}{.997 \times .001 + .015 \times .999}$$

$$= .062$$

Thus, in this population, a positive test result corresponds to only a 6% probability that the subject is actually HIV positive. This is the positive predictive value of the test (PPV). The low positive predictive value is due to low prevalence of the disease and the somewhat modest specificity relative to the prevalence of the disease. Suppose now that it was known that the subject was an intravenous drug user and routinely had intercourse with an HIV infected partner. In this situation the sensitivity and specificity of the test remain unchanged, though the disease prevalence would change quite dramatically. Indeed, let us assume that the prevalence of the disease in the population of individuals who are intravenous drug users and routinely have intercourse with an HIV infected partner is $P(D+) = 0.04$, that is, 40 times the prevalence in the overall population. For this person the probability of being HIV positive is

$$P(D+|T+) = \frac{P(T+|D+)P(D+)}{P(T+|D+)P(D+) + P(T+|D-)P(D-)}$$

$$= \frac{P(T+|D+)P(D+)}{P(T+|D+)P(D+) + \{1 - P(T-|D-)\}\{1 - P(D+)\}}$$

$$= \frac{.997 \times .04}{.997 \times .04 + .015 \times .96}$$

$$= .73$$

This is a much higher probability of being infected than for a person at random from the general population. This happens because of the prior knowledge that the disease prevalence in this population is much higher than in the general population. It is important to note that the evidence implied by a positive test result does not change with the prevalence of disease in the subject's population.

Indeed, the data are the same: an individual tested positive for an HIV test. However, what does change is our interpretation of that evidence due to the prior knowledge about the disease prevalence.

6.2.1.2 Example: effect of prevalence on the performance of diagnostic tests

Consider the example in Figure 6.10, which indicates that 1000 subjects were tested, 50 who had the disease and 950 who did not, with a disease prevalence $\widehat{P}(D+) = 50/1000 = 0.05$. Out of 50 individuals who had the disease 48 tested positive and 2 tested negative, while out of the 950 individuals who did not have the disease 47 tested positive and 903 tested negative. The (estimated) sensitivity of this test is $\widehat{P}(T+|D+) = 48/50 = 0.96$ and the specificity is $\widehat{P}(T-|D-) = 903/950 = 0.95$, which are both relatively good for a test. However, the PPV is $\widehat{P}(D+|T+) = 48/95 = 0.51$, which can be considered relatively low. This is due to the relatively low prevalence, $\widehat{P}(D+) = 0.05$. Indeed, PPV is estimated by $48/(48+47)$, where 48 is the number of subjects who test positive and have the disease and 47 is the number of subjects who test positive and do not have the disease. Because the prevalence is small the number of subjects who both test positive and have the disease cannot be too large; indeed, even if the sensitivity were perfect that number would be 50 and the PPV would barely change to 0.52. This should provide us with the intuition about the limitation of an excellent, or even perfect, sensitivity, in low prevalence diseases. We conclude that PPV is substantially adversely affected by the number of subjects who test positive and do not have the disease (in our case 47). First, the number of individuals who do not have the disease is much larger than the number of subjects who have the disease when the disease is rare (in our case $950 >> 50$). The test has a good specificity (0.95), but even a small percent of errors for those without the disease can lead to 47 false positive tests. This number of false positives becomes comparable with the number of true positives, which reduces the PPV of the test.

Consider now a test with almost identical sensitivity and specificity, but in a population where the disease is more prevalent. These data are shown in Figure 6.11 and indicates that the size of the total population is still 1000, though this time 200 have the disease and 800 do not; the disease prevalence in this population is $\widehat{P}(D+) = 200/1000 = 0.20$. Out of 200 individuals who have the disease 190 tested positive and 10 tested negative, while out of the 800 individuals who did not have the disease 40 tested positive and 760 tested negative. The (estimated) sensitivity of this test is $\widehat{P}(T+|D+) = 190/200 = 0.95$ and the specificity is $\widehat{P}(T-|D-) = 760/800 = 0.95$. The PPV for this test is $\widehat{P}(D+|T+) = 190/230 = 0.83$, which is much higher than in the previous example. This is primarily due to the higher prevalence of the disease, $\widehat{P}(D+) = 0.20$. Indeed, PPV is estimated by $190/(190+40)$, where 190 is the number of subjects who test positive and have the disease and 40 is the number of

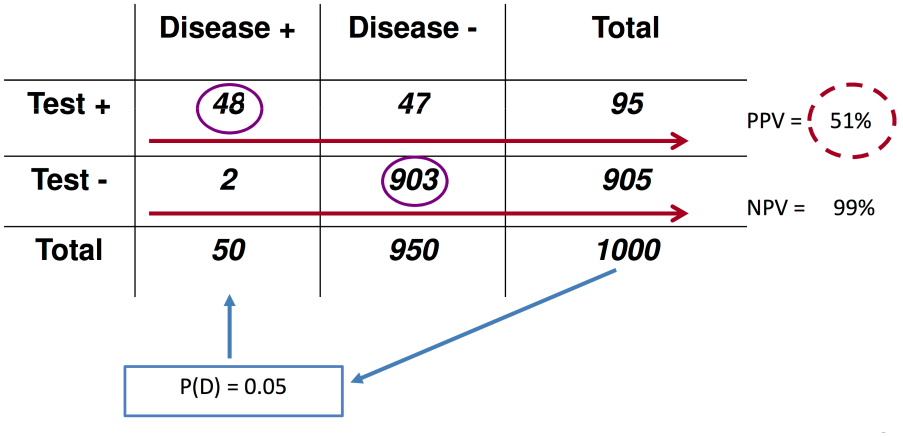


Figure 6.10: Two by two table showing the summary of results from a diagnostic test procedure combined with its characteristics: sensitivity, specificity, PPV, and NPV. Disease prevalence is P(D)=0.05.

subjects who test positive and do not have the disease. Because the prevalence is larger the number of subjects who both test positive and have the disease is larger. PPV is less affected by the number of subjects who test positive and do not have the disease (in our case 40) because this is compensated by the larger number of true positive tests (190). First, the number of individuals who do not have the disease is much larger than the number of subjects who have the disease when the disease is rare (in our case $950 >> 50$). The test has a good specificity (0.95), but even a small percent of errors for those without the disease can lead to 47 false positive tests. This number of false positives becomes comparable with the number of true positives, which reduces the PPV of the test.

The PPV and NPV are extremely important quantities in prediction. Indeed, an individual patient is interested in his or her probability of having the disease given that he or she tested positive or negative. However, it is important to remember that these quantities are a combination of the properties of the test (sensitivity and specificity) and the prior prevalence in the population. A criticism of PPV and NPV is that for an individual it is often hard to know from what population he or she is drawn. Indeed, if the individual is taken at random and tested we could use the known prevalence of the disease in the population. However, this is rarely the case and NPV and PPV depend on the unknown (latent) status of the population the patient belongs too. We have added this discussion to emphasize that PPV and NPV should not be used without reference to the prevalence of the disease population the individual was drawn from and that this information may not be available in realistic scenarios.

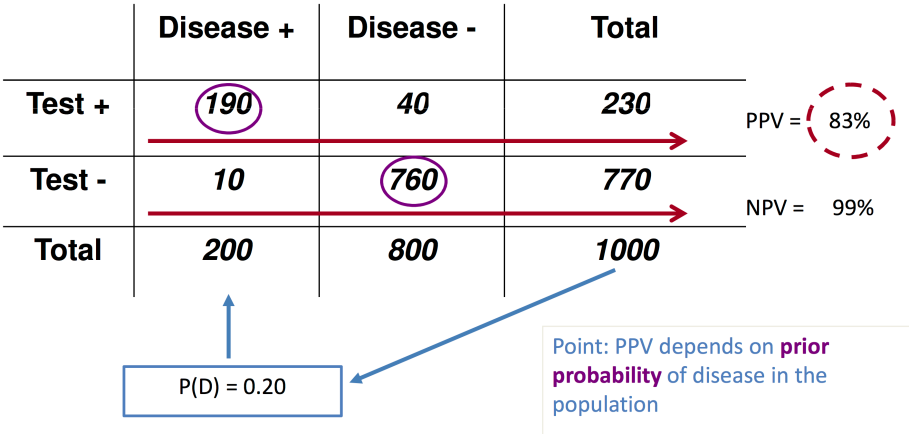


Figure 6.11: Two by two table showing the summary of results from a diagnostic test procedure combined with its characteristics: sensitivity, specificity, PPV, and NPV. Disease prevalence is P(D)=0.20.

6.2.1.3 Likelihood ratios of diagnostic tests

The *diagnostic likelihood ratio of a positive test* is

$$\text{DLR}_+ = \frac{P(T+|D+)}{P(T+|D-)} = \frac{\text{sensitivity}}{1-\text{specificity}} .$$

For a test to have any value we expect that $\text{DLR}_+ > 1$, indicating that the probability of having a positive test is higher for a person who has the disease relative to one who does not. In general, we expect $\text{DLR}_+ >> 1$. An estimator of DLR_+ is

$$\widehat{\text{DLR}}_+ = \frac{a/a+c}{1-d/(b+d)} = \frac{a}{b}\frac{b+d}{a+c} .$$

Note that $\widehat{\text{DLR}}_+ > 1$ if and only if $ad > bc$, where a and d are the number of subjects correctly classified and b and c are the numbers of subjects incorrectly classified, respectively.

The *diagnostic likelihood ratio of a negative test* is

$$\text{DLR}_- = \frac{P(T-|D+)}{P(T-|D-)} = \frac{1-\text{ sensitivity}}{\text{specificity}} .$$

For a test to have any value we expect that $\text{DLR}_- < 1$, indicating that the probability of having a negative test result for a person who has the disease is smaller than for one person who does not. In general, we expect $\text{DLR}_- << 1$.

An estimator of DLR+ is

$$\widehat{\text{DLR}}_- = \frac{1-(a/a+c)}{d/(b+d)} = \frac{c}{d}\frac{b+d}{a+c} .$$

Just as in the case of $\widehat{\text{DLR}}_+$ we have $\widehat{\text{DLR}}_+ < 1$ if and only if $ad > bc$, the exact same inequality.

Using the Bayes rule we have

$$P(D+|T+) = \frac{P(T+|D+)P(D+)}{P(T+|D+)P(D+) + P(T+|D-)P(D-)}$$

and

$$P(D-|T+) = \frac{P(T+|D-)P(D-)}{P(T+|D-)P(D-) + P(T+|D+)P(D+)} ,$$

where the denominators in both formulas are the same and are equal to $P(T+)$, the marginal probability of a positive test. By taking the ratio we obtain that

$$\frac{P(D+|T+)}{1-P(D+|T+)} = \frac{P(T+|D+)}{P(T+|D-)} \times \frac{P(D+)}{1-P(D+)}$$

For any probability p of an event we define the odds of that event to be

$$\text{Odds}(p) = \frac{p}{1-p} .$$

There clearly is no information gained or lost from going from the absolute probability p to the Odds(p) or back, though the interpretation can change quite a bit. For example, an odds of 1 corresponds to equal chances of an event to happen or not happen $p = 1-p = 0.5$, while an odds of 3 corresponds to a probability twice as large for an event to happen than to not ($p = 0.75$ and $1-p = 0.25$). Odds are always positive and they simply change the scale on which we discuss probability. With this definition, the previous results imply that

$$\text{post test odds}(D+) = \text{DLR}_+ \times \text{ prior test odds}(D+) ,$$

where $\text{DLR}_+ = P(T+|D+)/P(T+|D-)$ is the diagnostic likelihood ratio of a positive test. Indeed, before (or prior to) conducting the test the odds of having the disease, for a person taken at random from the population, are $P(D+)/\{1-P(D+)\}$. However, for a person drawn at random from the population, who tests positive, the odds are $P(D+|T+)/\{1-P(D+|T+)\}$. The formula is the product between the prior odds of having the disease and the diagnostic likelihood ratio of a positive test. Here, DLR_+ contains the information about the experiment, which came up positive, and the prior odds contain the prior information about the prevalence of the disease in the population. In the HIV example, the prior test odds of the disease are $0.001/0.999 \approx 0.001$ and are very small. The $\text{DLR}_+ = P(T+|D+)/\{1-P(T-|D-)\} = 0.997/(1-0.985) = 66.5$, indicating the positive test made the odds of disease 66 times the pretest odds.

Equivalently, the hypothesis of disease is 66 times more supported by the data than the hypothesis of no disease after running the test. The odds remain very small despite the strong multiplying effect of the DLR_+ simply because the prior odds were just so small to start with.

Similarly, if a subject has a negative test result, it can be shown that

$$\text{post test odds}(D-) = \text{DLR}_- \times \text{prior test odds}(D-) \ .$$

For the HIV test example $DLR_- = (1-0.997)/0.985 = 0.003$ indicating that the post-test odds of disease are now 0.3% of the pretest odds given the negative test. Or, the hypothesis of disease is supported 0.003 times that of the hypothesis of absence of disease given the negative test result.

In practice we may actually have two different independent diagnostic tests that are applied or the same test could be repeated. We would like to see what happens with the accumulation of evidence and compare the evidence from multiple tests. Let us assume that we have a both a positive first and second test. After applying the first test the post-test odds of having the disease are

$$\frac{P(D+|T_1+)}{P(D-|T_1+)} = \text{DLR}_+(T_1)\frac{P(D+)}{P(D-)} \ ,$$

where for precision of notation we denoted by $\text{DLR}_+(T_1)$ the diagnostic likelihood ratio of a positive test when using test T_1. Once the second test is applied and is positive we have

$$\begin{aligned}\frac{P(D+|T_1+,T_2+)}{P(D-|T_1+,T_2+)} &= \text{DLR}_+(T_2)\frac{P(D+|T_1+)}{P(D-|T_1+)} \\ &= \ \text{DLR}_+(T_2) \times \text{DLR}_+(T_1) \times \frac{P(D+)}{P(D-)} \ .\end{aligned}$$

Thus, if the two tests are independent we have the diagnostic likelihood ratios of the two tests simply multiply the prior odds of the disease. This is particularly useful when we would like to repeat the test to "obtain confirmation of the original findings." Note, in particular, how before the first test $P(D+)/P(D-)$ is the prior odds of having the disease and $P(D+|T_1+)/P(D-|T_1+)$ is the posterior odds of having the disease. However, after the experiment is run this posterior odds of having the disease becomes the prior odds of having the disease before the second test is run.

Let us see what happens if we apply the HIV test twice and it comes back positive both times. In this situation $P(D+)/P(D-) = 0.001$, but $\text{DLR}_+(T) \times \text{DLR}_+(T) = 66.5^2 = 4422.3$ indicating that the posterior odds of being HIV positive is

$$\frac{P(D+|T_1+,T_2+)}{P(D-|T_1+,T_2+)} = 4422.3 * 0.001 = 4.42 \ ,$$

making it more than four times more likely to have the disease than to not have the disease. This would correspond to a conditional probability of having the

disease of

$$P(D+|T_1+,T_2+) = \frac{4.42}{1+4.42} = 0.815 \ .$$

One more independent positive test would result in a posterior odds of the disease of

$$\frac{P(D+|T_1+,T_2+,T_3+)}{P(D-|T_1+,T_2+,T_3+)} = 66.5^3 * 0.001 = 294.08 \ ,$$

which would correspond to a probability of having the disease of

$$P(D+|T_1+,T_2+,T_3+) = \frac{294.08}{1+294.08} = 0.997 \ .$$

This shows that repeated positive tests dramatically increase the probability that the person has, indeed, the disease.

6.2.2 A bit of house cleaning

We emphasize that in the examples above, we are estimating hypothesized true probabilities with data. To be good statisticians, we should quantify the uncertainty in those estimates. Do not worry, we will give the tools for doing that later in the text.

A second, somewhat philosophical issue always arises when discussing diagnostic testing. Under a frequency definition of probability, either a subject has the disease or not. So his or her person-specific PPV and NPV are technically either one or zero. This line of thinking, while theoretically true, is not very useful, other than perhaps suggesting the need to be careful with language. Better language for PPV, for example, is that we are considering the probability that a random person has the disease when drawn from a relevant population of people with positive tests.

6.3 ROC and AUC

Sensitivity and specificity are widely used in practice to characterize the performance of classifiers. More precisely, in practice one often has a binary outcome (e.g., dead/alive, healthy/disease, failure/success) using covariates or predictors.

6.3.1 Prediction of binary outcomes

Consider the case when the outcome is binary and we have one continuous predictor, or risk score. This can happen in a large number of applications including measuring a biomarker used to predict disease status (biomarker-based diagnosis test), constructing a univariate risk score from multivariate covariates,

and using clinical scores based on a combination of objective and subjective patient assessments. From a notational perspective, let D be the binary outcome variable, which can be either 0 or 1 and let R be a scalar random variable, which will be used as predictor for D. Here we make the convention that larger values of R correspond to higher predicted risk. Let $\mathcal{R}$ be the range of possible values for the variable R; for convenience we can always make this range $[0, 1]$ using a suitable transformation. For every threshold $r \in [[0, 1]$ we can predict $\widehat{D} = 1$ if $R > r$. This is now a diagnostic test and its theoretical sensitivity is

$$\text{Sensitivity}(r) = P(R > r|D = 1) \ ,$$

while its theoretical specificity is

$$\text{Specificity}(r) = P(R \leq r|D = 0) \ .$$

The theoretical receiver operating characteristic (ROC) curve is defined as

$$\{\text{Sensitivity}(r), 1 - \text{Specificity}(r) : r \in [0, 1]\} \ .$$

The area under the ROC is denoted by AUC and is one of the main criteria for assessing discrimination accuracy. We will prove that the *AUC is the probability that the model will assign a higher probability of an event to the subject who will experience the event than to the one who will not experience the event.* Note first that

$$\text{Sensitivity}(t) = S(t) = P(X > r|D = 1) = \int_r^1 f(x|D = 1)dx$$

and that

$$1 - \text{Specificity}(r) = P(r) = P(X > r|D = 0) \ .$$

Therefore

$$\text{AUC} = \int_1^0 S(r)\frac{d}{dr}P(r) = \int_0^1 S(r)f(r|D = 0)dr = \int_0^1 \int_r^1 f(s|D = 1)f(r|D = 0)dsdr \ .$$

The first equality holds because we integrate with respect to $P(r)$ from the smallest values of $P(r)$ to the largest value of $P(r)$. The smallest value of the specificity is 0 and is attained at $r = 1$, while the largest value of $P(r)$ is 1 and is attained at $r = 0$. The second equality holds because the derivative with respect to r of $P(r)$ is equal to $-f(r|D = 0)$, which changes the limits of integration. The last equality comes simply from the definition of the double integral and by placing $f(r|D = 0)$ as a term under the integral with respect to s (because it does not depend on s). An equivalent way of writing the double integral is

$$\text{AUC} = \int_{\{(r,s):0\leq r<s\leq 1\}} f(s|D = 1)f(r|D = 0)dsdr \ .$$

Upon close inspection one can observe that if R_i and R_j are two scores for the i and j individuals then

$$\text{AUC} = P(R_i > R_j|D_i = 1, D_j = 0) \ .$$

Indeed, the joint pdf of $(R_i, R_j|D_i = 1, D_j = 0)$ is $f(r_i|D = 1)f(r_j|D = 0)$ because R_i and R_j are assumed to be independent given their disease status. Therefore

$$P(R_i > R_j|D_i = 1, D_j = 0) = \int_{\{(r_i,r_j):0\leq r_j<r_i\leq 1\}} f(r_i|D = 1)f(r_j|D = 0)dr_i dr_j \ ,$$

which is the same formula as the one for AUC with the exception that the arguments of the conditional distributions were changed as follows $s \to r_i$ and $r \to r_j$. Of course, the AUC has no value in terms of helping the prediction after the disease status is observed. Instead, the AUC can be used to evaluate the probability that a person who will develop the disease was assigned a larger risk score than one who will not develop the disease *before the experiment is run*. In practice, an estimate of AUC can be obtained after data are collected and one assumes that the discriminating characteristics of the associated risk score are generalizable to the population. We now provide examples of how the AUC and ROC are used in practice.

6.3.2 Example: building up the intuition using the SHHS

Consider the SHHS and let us try to predict who has moderate to severe sleep apnea from other covariates. Moderate to severe sleep apnea is defined as an `rdi4p` at or above 15 events per hour. There are

```
MtS_SA=rdi4p>=15
n_positive =sum(MtS_SA)
n_positive
```

```
[1] 1004
```

1004 individuals in the SHHS who have moderate to severe sleep apnea among the 5804 individuals in SHHS for a prevalence of $\widehat{P}(D+) = 0.178$, or 17.8% of the SHHS population. Individuals in SHHS were oversampled for increased likelihood of having sleep apnea and this prevalence is not representative for the overall US population. Indeed, the prevalence of obstructive sleep apnea associated with accompanying daytime sleepiness is approximately 3 to 7% for adult men and 2 to 5% for adult women in the general population (Punjabi 2008). We would like to predict the outcome using a few covariates including `gender`, `age_s1`, `bmi_s1`, and `HTNDerv_s1`. The last variable is hypertension status. To do this, we fit a generalized linear model (GLM) regression of the variable `MtS_SA`, moderate to severe sleep apnea, on these covariates. We have not yet covered regression and we will not cover it for some time, but this is not a serious reason not to use it, especially in its push-button form. Thus, the regression is done in `R` as

```
fit<-glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1,family="binomial")
summary(fit)
```

```
Call:
glm(formula = MtS_SA ~ gender + age_s1 + bmi_s1 + HTNDerv_s1,
    family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0350  -0.6558  -0.4447  -0.2672   2.8334

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -8.478417   0.374489 -22.640   <2e-16 ***
gender       1.160011   0.079171  14.652   <2e-16 ***
age_s1       0.032785   0.003695   8.873   <2e-16 ***
bmi_s1       0.139142   0.007386  18.839   <2e-16 ***
HTNDerv_s1   0.186753   0.077047   2.424   0.0154 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5314.5  on 5760  degrees of freedom
Residual deviance: 4660.6  on 5756  degrees of freedom
  (43 observations deleted due to missingness)
AIC: 4670.6

Number of Fisher Scoring iterations: 5
```

The results of the regression indicate that all covariates are signifcant if a p-value < 0.05 is used as a criterion (again we are using an undefined concept, but we encourage you to follow along). Male `gender` (here `gender`$= 1$ indicates male), older `age_s1`, higher `bmi_s1`, and being `HTNDerv_s1` positive all increase the probability of having moderate to severe sleep apnea. The summary also indicates that there were 43 observations due to missingness in `bmi_s1`. Beyond the information about what covariates are significantly associated with the outcome, we also obtain a prediction equation. More precisely, using the point estimators of the effects we can build the following predictor, or risk score, for moderate to severe sleep apnea for any subject i in SHHS

$$R_i = 1.16\text{gender}_i + 0.033\text{age}_i + 0.14\text{BMI}_i + 0.19\text{HTN}_i \, .$$

Here we ignore the intercept, whose role is primarily to scale the predictor R_i to agree with the observed prevalence of the disease. For now, we are just building the intuition of how these predictors are built from the data. Below we present a relevant subset of the information necessary to understand the process

```
predictor_rdi4p<-1.16*gender+0.033*age_s1+0.14*bmi_s1+0.19*HTNDerv_s1
intuition<-cbind(gender,age_s1,bmi_s1,HTNDerv_s1,rdi4p,MtS_SA,predictor_rdi4p)
round(head(intuition),digits=2)
```

```
     gender age_s1 bmi_s1 HTNDerv_s1 rdi4p MtS_SA predictor_rdi4p
[1,]      1     55  21.78          1  1.44      0            6.21
[2,]      1     78  32.95          1 17.80      1            8.54
[3,]      0     77  24.11          0  4.85      0            5.92
[4,]      1     48  20.19          1  0.80      0            5.76
[5,]      0     66  23.31          1  2.76      0            5.63
[6,]      1     63  27.15          1  3.72      0            7.23
```

Among the first six subjects in the dataset, only the second one has moderate to severe sleep apnea (`rdi4p` is 17.8 events per hour). Interestingly, this individual had all the indicators that increase the probability of having sleep apnea: he is a man, is older (78), has a high BMI (32.95 indicates that the person is obese), and is hypertensive. Thus, his risk score was

$$R_2 = 1.16 \times 1 + 0.033 \times 78 + 0.14 \times 32.95 + 0.19 \times 1 = 8.54 \, ,$$

which happens to be the largest value of the risk score among the first 6 subjects. In contrast, the third individual is a female, of about the same age (77), whose BMI is 24.11 (considered normal), and who does not have hypertension. The risk score for this woman was

$$R_3 = 1.16 \times 0 + 0.033 \times 77 + 0.14 \times 24.11 + 0.19 \times 0 = 5.92 \, ,$$

a smaller value than for the second individual. Of course, in general, we will encounter individuals with high risk scores who do not have moderate to severe sleep apnea and we will encounter individuals with low risk scores who do have it. To better understand this we will plot the risk score versus `rdi4p` and identify exactly how good the prediction equation is. But first, we would like to understand the marginal distribution of the linear predictor.

```
hist(predictor_rdi4p,probability=T,col=rgb(0,0,1,1/4),breaks=30,
     xlab="Risk score",
     main="",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(c(8.54,8.54),c(0,0.4),col="red",lwd=3)
lines(c(5.92,5.92),c(0,0.4),col="red",lwd=3)
```

Figure 6.12 displays the histogram of the risk scores, R_i, for every subject in the SHHS. This histogram indicates that most individuals in the SHHS will have a risk score between 4 and 9, with few exceptions, and that the distribution of scores has a reasonably symmetric distribution that could be unsuccessfully argued to be Normal. We have also indicated the two individuals discussed above in terms of risk score (the left vertical red line is for the woman and right vertical red line is for the man). This should provide additional intuition about how complex information about individuals is incorporated into the risk score.

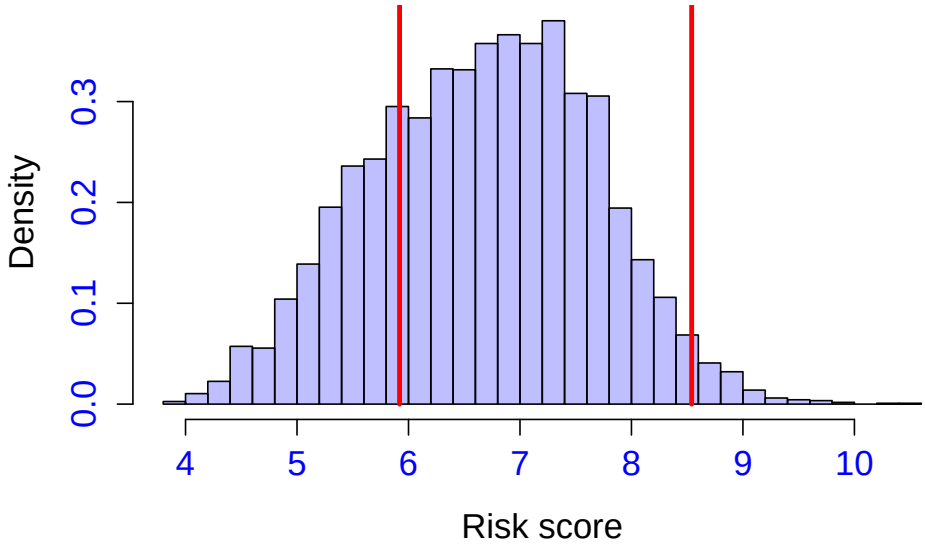


Figure 6.12: Risk score for moderate to severe sleep apnea estimated from a logistic regression with gender, age, BMI, and hypertension as predictors.

Individuals in the left tail will tend to be younger, have a lower BMI, and more likely to be women and not have hypertension. The individuals in the upper tail will tend to be older, have a higher BMI, and more likely to be men and have hypertension. This type of graph provides a lot of insight into what could happen and it is always interesting to explore the reasons why individuals end up in the extreme tails of the distribution.

Figure 6.13 displays the risk score versus `rdi4p` indicating that, on average, higher risk scores correspond to higher observed `rdi4p`. Here we have cut the plot at `rdi4p` less than 40 events per hour for visualization purposes, but one should remember that many subjects have an `rdi4p` higher than 40. To better visualize the connection between the risk score and `rdi4p`, we calculate the average `rdi4p` stratified by the risk score. A good stratification of the risk score could be: < 5, $[5, 6)$, $[6, 7)$, $[7, 8)$, $[8, 9)$, and ≥ 9.

Once we stratify the risk score, Figure 6.14 displays the means of `rdi4p` variable in every risk score stratum. The plot indicates that there is a clear increase in average `rdi4p` as a function of strata and the increase actually accelerates for risk scores above 6. This should not be surprising as the risk score was designed to help predict individuals who will have an `rdi4p` above 15 and not designed specifically to predict `rdi4p`. However, it is reassuring that the risk score seems to be associated with `rdi4p` even below 15.

A threshold on the scores can be used to predict who has the disease and who does not have it. What we are trying to do is predict which individuals will have an `rdi4p` above 15 using the risk scores. A simple solution is to consider

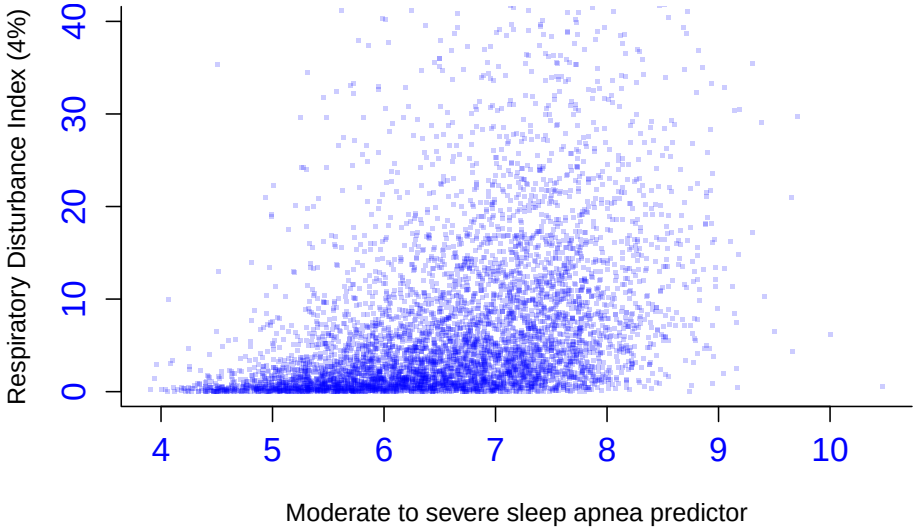


Figure 6.13: Scatter plot of the estimated subject-specific risk score and the observed respiratory disturbance index at 4% oxygen desaturation.

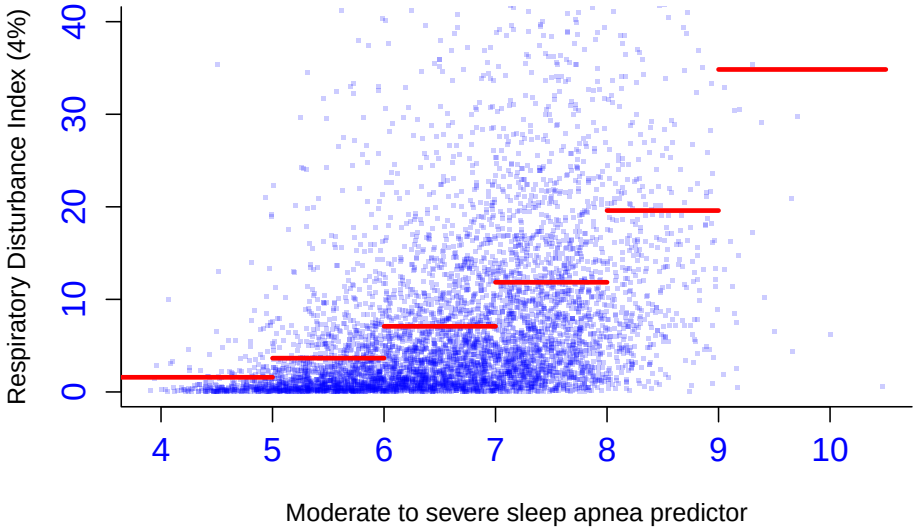


Figure 6.14: Scatter plot of the estimated subject-specific risk score and the observed respiratory disturbance index at 4% oxygen desaturation together with its means in strata defined by the risk scors.

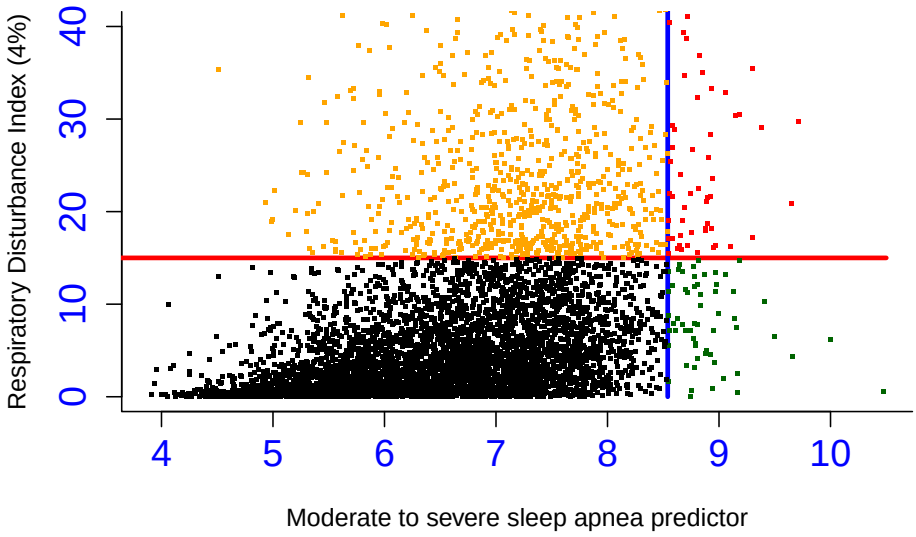


Figure 6.15: Same scatter plot as in the previous two Figures. Red horizontal line at 15 events per hour is the demarcation line between having and not having the disease. Blue horizontal line at 8.54 is the demarcation line between being predicted to have or not have the disease.

a threshold on the risk scores and predict that everybody above that particular threshold will have an `rdi4p` above 15 and everybody below that threshold will have an `rdi4p` below 15. However, there are many thresholds that can be used. Indeed, consider, for example, the threshold $R = 8.54$, which corresponds to the risk score of the second subject in our dataset. The procedure is to predict that every person with a risk score $R_i \geq 8.54$ has the disease and every person with $R_i \leq 8.54$ does not. Figure 6.15 provides the color coding for understanding the procedure. The horizontal red line in Figure 6.15 displays the value of observed `rdi4p` of 15, the limit between having and not having moderate to severe sleep apnea. All subjects on or above the red line have the disease. The vertical blue line in Figure 6.15 displays the value of the risk score of 8.54. All subjects to the right of the vertical blue line are predicted to have the disease, while all subjects on or to the left of the line are predicted not to have it.

Figure 6.15 also provides color coding for the type of correct and incorrect predictions that can be made. The red dots indicate the true positive tests, that is, those predictions that are correct for individuals who have the disease. In this dataset there are 81 true positive tests (not all shown in the plot due to cutting of the y-axis). The green dots indicate the false positive tests, that is, those predictions that indicated that the subjects will have the disease when they do not. There are 59 false positive tests. The black dots indicate the number of true negative tests, that is, those predictions that indicated that the subjects do not have the disease when they do not. There are 4703 true

negative tests. The orange dots represent the false negative tests, that is, those predictions that indicated that the subjects do not have the disease when they do have it. There are 918 false negative tests.

The plot is slightly misleading due to the amount of overplotting of black dots, especially in the low `rdi4p` range, corresponding to healthy individuals. Using the same definition for diagnostic tests we can define the estimated sensitivity, specificity, PPV, and NPV for the risk score and a corresponding threshold of risk, say $T = 8.54$. Indeed, the estimated sensitivity of the test is $81/(81+918) \approx 0.08$, which is the number of true positive tests (red dots) divided by the number of subjects who have the disease (red and orange dots). The specificity of the test is $4703/(4703 + 59) = 0.988$, which is the number of true negative tests (black dots) divided by the number of individuals who do not have the disease (black and green dots). Results indicate that, at this threshold, the test is not very sensitive but is highly specific. The PPV is $81/(81+59) = 0.58$, the number of true positive tests (red dots) divided by the number of positive predictions (red dots and green dots). This indicates that more than half of the positive predictions are accurate. The NPV is $4703/(4703 + 918) = 0.84$, the number of true negative tests (black dots) divided by the number of negative predictions (black and orange dots). This indicates that $\approx 85\%$ of the negative predictions are accurate.

Of course, the threshold that we chose was arbitrary. Indeed, any other threshold could be used. For example, we could use the threshold $R = 5.92$, the risk score of the third individual in the dataset. Figure 6.16 is similar to Figure 6.15, except that the prediction threshold for having the disease (blue vertical line) is moved substantially to the left. This results in a much larger number of true positives (red dots) and a substantial reduction in the number of false negatives (orange dots), though it comes at the price of a much smaller number of true negatives (black dots) and a substantial increase in the number of false positives (green dots). The sensitivity, specificity, PPV, and NPV for this threshold are 0.94, 0.29, 0.22, and 0.96. Thus, lowering the threshold leads to much improved sensitivity and NPV, but much reduced specificity and PPV.

6.3.3 Visualization of risk distributions

So far we have seen the distribution of risk in the population, but it is interesting to compare the distributions of risk by disease status. Figure 6.17 displays the risk distribution for individuals without moderate to severe sleep apnea (red histogram) and for individuals with moderate to severe sleep apnea (blue histogram). The distribution of risk scores of individuals with the disease is clearly shifted to the right, but there is no complete separation between those with disease and without disease. This is rather common in health applications, where clustering is not necessarily obvious and disease populations risk predictors can be masked among the risk predictors of healthy individuals. The red histogram is much higher than the blue histogram because here we used the

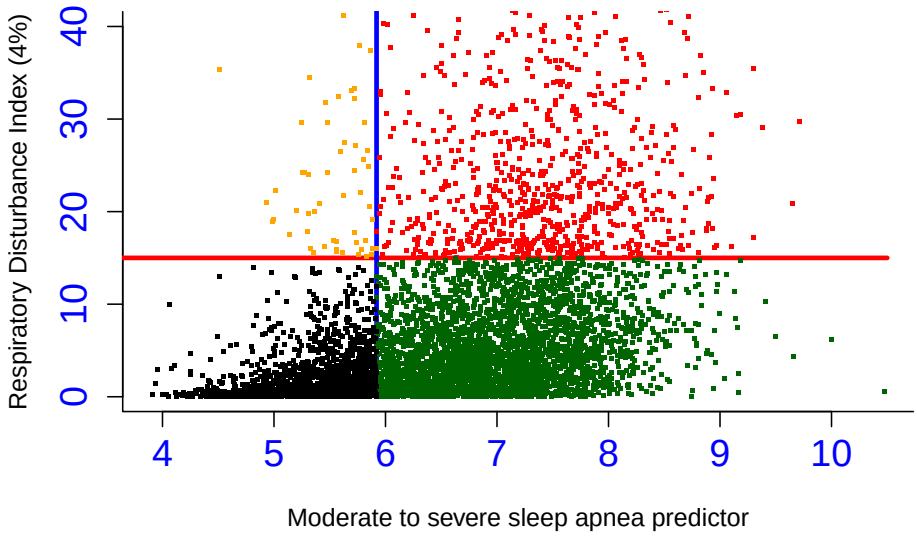


Figure 6.16: Same scatter plot as in the previous two Figures. Red horizontal line at 15 events per hour is the demarcation line between having and not having the disease. Blue horizontal line at 5.92 is the demarcation line between being predicted to have or not have the disease.

frequency on the y-axis. Since the number of individuals without the disease is much higher than the number of individuals with the disease, it is natural for the red histogram to visually dominate the blue histogram.

```
ll<-min(predictor_rdi4p,na.rm=TRUE)
ul<-max(predictor_rdi4p,na.rm=TRUE)
#plot the histogram of risk scores for individuals
#without moderate to severe sleep apnea
hist(predictor_rdi4p[MtS_SA==0], col = rgb(1,0,0,0.25),
     breaks=40,xlim=c(ll,ul),
     xlab="Moderate to severe sleep apnea predictor",
     cex.axis=1.5,col.axis="blue",main=NULL,cex.lab=1.3)
#add the histogram of risks for individuals
#with moderate to severe sleep apnea
hist(predictor_rdi4p[MtS_SA==1],col = rgb(0,0,1,0.25),add =T,breaks=30)
lines(c(8.54,8.54),c(0,400),col="red",lwd=3)
lines(c(5.92,5.92),c(0,400),col="red",lwd=3)
```

We also illustrate the same two risk thresholds that correspond to the second and third individuals in the SHHS using the vertical red lines. We discuss only the right threshold at $R_2 = 8.54$, which is indicated by the red line to the right. The estimated sensitivity of the test is $\widehat{P}(T+|D+)$, which is the number of individuals to the right of the red line in the blue distribution divided by the

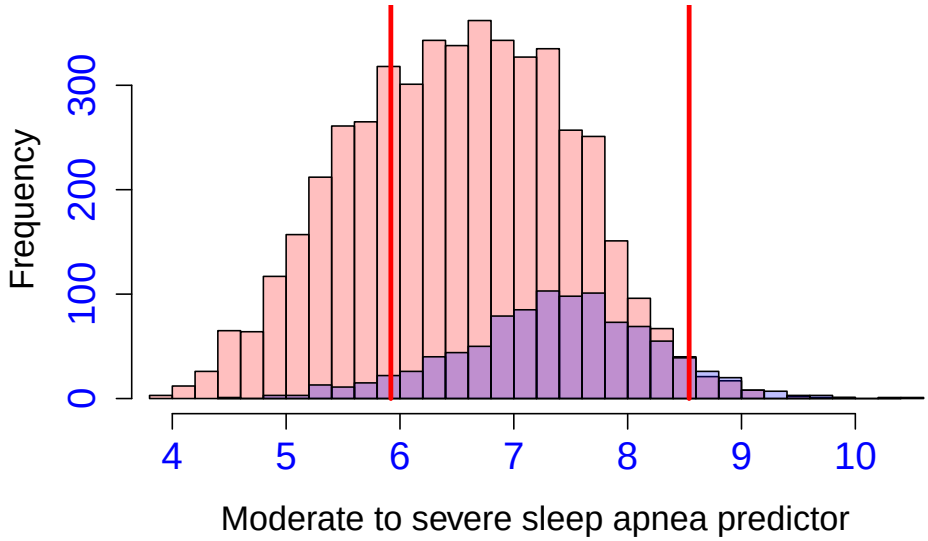


Figure 6.17: Histogram of risk scores stratified by healthy (red) and individuals with moderate to severe sleep apnea (blue).

number of individuals in the blue distribution. The specificity is $\widehat{P}(T-|D-)$, the number of individuals to the left of the red line in the red distribution divided by the number of individuals in the red distribution. The positive predictive value, $\widehat{P}(D+|T+)$, is the number of subjects to the right of the red line in the blue distribution divided by the number of subjects to the right of the red line in the red and blue distributions. A similar definition holds for specificity and we leave this as an exercise. Note that the same plot could be made using probability distributions, instead.

Figure 6.18 displays the same information as Figure 6.17, though the y-axis is expresssed in the ratio of subjects relative to the size of their group and not in absolute number of individuals per bin. The two distributions look remarkably different in this representation because they are both rescaled to integrate to 1. What is lost in Figure 6.18 is the number of subjects, the prevalence of the disease in the population, and the PPV and NPV, which cannot be calculated from the graph. However, the sensitivity can still be estimated as the area under the blue curve to the right of the red line, and the specificity can be estimated as the area under the red distribution to the left of the red line. Both these plots are complementary to Figures 6.15 and 6.16 and provide a different interpretation of the various quantities associated with the prediction performance of the risk score in the context of binary prediction. Which one is more helpful is in the eye of the beholder.

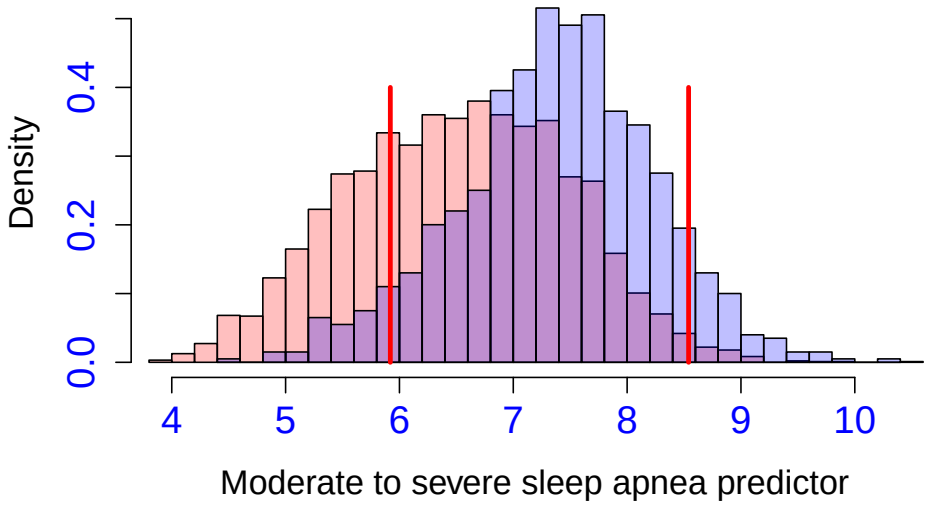


Figure 6.18: Histogram of risk scores stratified by healthy (red) and individuals with moderate to severe sleep apnea (blue).

6.3.4 Definition of the empirical ROC and AUC

The type of trade-off between sensitivity and specificity of a risk score is characteristic to a wide variety of problems. In practice, one may not have a good way of choosing a threshold or one may be interested in presenting the performance of the prediction performance of the risk score in one plot. In the SHHS example above, both the sensitivity and specificity of the risk score depend on the threshold.

The function $\{\text{Sensitivity}(R), 1-\text{Specificity}(R)\}$ is called the receiver operating characteristic (ROC) function and characterizes the risk score at any possible threshold, R. The area under the ROC is called the area under the receiver operating characteristic curve (AUC). In the SHHS example, both the sensitivity and specificity values are estimated from the data, which is why we refer to these functions as the empirical ROC and AUC, respectively. These functions, and not the theoretical ROC or AUC, are the ones used in practice. Thus, it is important to remember that both the empirical ROC and AUC depend on the data and will contain sampling variability, like any other statistic. We now show how to estimate both these quantities from scratch.

```
#Set the range of thresholds as the range of the predictor
thresh= seq(min(predictor_rdi4p,na.rm=TRUE),
            max(predictor_rdi4p,na.rm=TRUE),
            length=100)

#Sensitivity and specificity vectors
```

```
sensy=rep(NA,100)
specy=sensy

#PPV and NPV vectors
ppv=sensy
npv=sensy

#Proportion of positive tests
P_positive_tests=sensy

#Number of false positives
N_false_positives=sensy

#Number of false negatives
N_false_negatives=sensy

for (i in 1:length(thresh))
  {#begin iterating over thresholds
    R=thresh[i]
    #True positives
  red.points=(predictor_rdi4p>R) & (rdi4p>=15)
  #False positives
  green.points=(predictor_rdi4p>R) & (rdi4p<15)
  #False negatives
  orange.points=(predictor_rdi4p<=R) & (rdi4p>=15)
  #True negatives
  black.points=(predictor_rdi4p<=R) & (rdi4p<15)

  #Sensitivity and specificity at threshold R
    sensy[i] = sum(red.points, na.rm = TRUE) /
      (sum(red.points, na.rm = TRUE) + sum(orange.points, na.rm = TRUE))
    specy[i] = sum(black.points, na.rm = TRUE) /
      (sum(black.points, na.rm = TRUE) + sum(green.points, na.rm = TRUE))

    #PPV and NPV at threshold R
    ppv[i] = sum(red.points, na.rm = TRUE) /
      (sum(red.points,na.rm=TRUE) + sum(green.points, na.rm = TRUE))
    npv[i] = sum(black.points,na.rm = TRUE) /
      (sum(black.points,na.rm=TRUE) + sum(orange.points, na.rm = TRUE))

    #Number of positive tests
    P_positive_tests[i] =
      (sum(red.points,na.rm=TRUE) + sum(green.points,na.rm=TRUE)) /
      sum(!is.na(predictor_rdi4p))
```

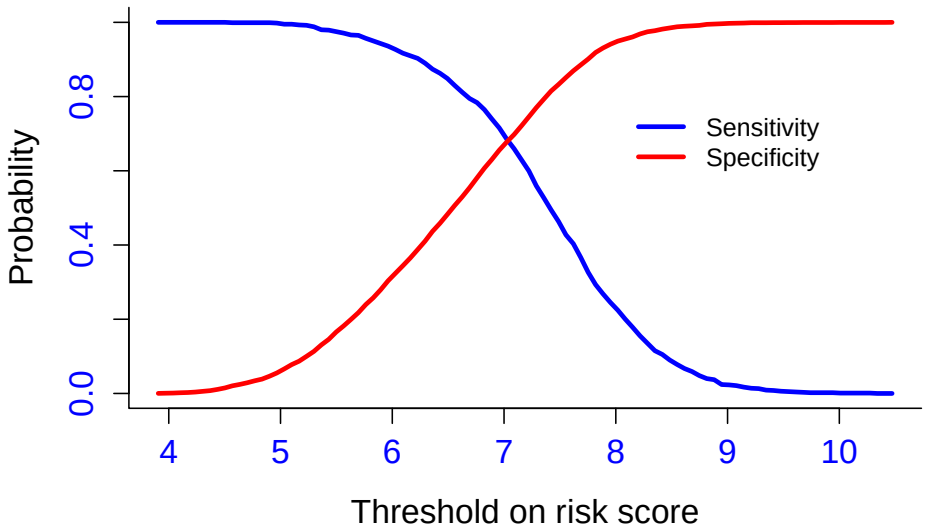


Figure 6.19: Sensitivity (blue line) and specificity (red line) as a function of the decision threshold.

```
    #Number of false positive tests
    N_false_positives[i] = sum(green.points, na.rm = TRUE)

    #Number of false negative tests
    N_false_negatives[i] = sum(orange.points, na.rm = TRUE)
 }#end iterating over thresholds
```

Figure 6.19 displays the sensitivity and specificity curves (blue and red lines, respectively) as a function of the threshold. Sensitivity is decreasing, while specificity is increasing as a function of the threshold. The maximum sensitivity is attained when the threshold $R = 3.91$, which is the minimum value of the risk score. At this threshold all subjects in the population are predicted to have the disease. The specificity at this threshold is 0 because there are no individuals who are predicted not to have the disease. Similarly, as the threshold approaches $T = 10.47$, the maximum value for the risk score, the sensitivity becomes 0 and the specificity becomes 1. It is interesting to observe the trade-off between sensitivity and specificity as a continuous function of the threshold.

The next step is to plot the ROC curve, which displays $1 - \text{specificity}(R)$ versus $\text{sensitivity}(R)$. Both these functions are decreasing functions of the threshold R and the estimated ROC curve is shown in Figure 6.20 as a solid red line. The black line is the 45 degrees line and the fact that the ROC is above the black line indicates that the risk score has some prognostic value that is better than chance alone. While a little counterintuitive, the points on the red line that are closer to $(0, 0)$ correspond to higher thresholds, while those closer to $(1, 1)$

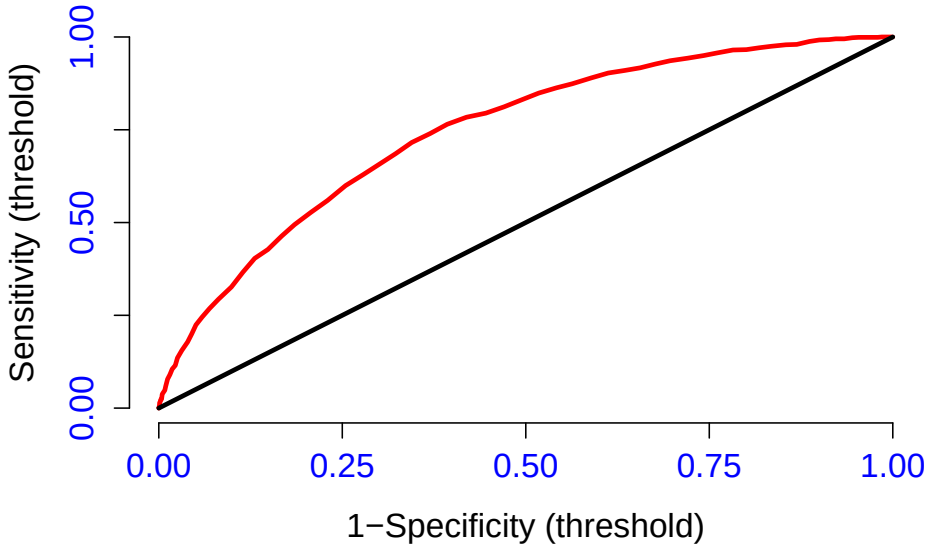


Figure 6.20: Receiver operating characteristic (ROC) curve displaying 1− specificity versus sensitivity. Points closer to $(0, 0)$ correspond to higher thresholds, while points closer to $(1, 1)$ correspond to lower thresholds.

correspond to lower thresholds. In some sense we could think that the red line is drawn from right to left as the threshold increases and spans the entire range of thresholds.

We would also like to calculate the area under the curve and we do this using the trapeizoidal rule for numerical approximations of the integral. The only small problem here is that the x-axis is not a standard axis, but the increments are in terms of differences in $1 - \text{Specificity}(R)$ and not in terms of increments of R. This makes our job only slightly more difficult as we need to calculate

$$\text{AUC} = \int \text{Sens}(R) d\{1 - \text{Spec}(R)\} \, .$$

Thus, we have the following numerical approximation to the integral

$$\text{AUC} \approx \sum_{k=1}^{n-1} \frac{\text{Sens}(R_k) + \text{Sens}(R_{k+1})}{2} \{\text{Spec}(R_{k+1}) - \text{Spec}(R_k)\} \, .$$

Here $R_k, k \in 1, \ldots, n = 100$ are the thresholds. The second term, $\text{Spec}(R_{k+1}) - \text{Spec}(R_k)$, is positive because the specificity function is increasing as a function of R_k and is equal to

$$\{1 - \text{Spec}(R_k)\} - \{1 - \text{Spec}(R_{k+1})\} \, .$$

This can be implemented in `R` as follows

```
  #number of thresholds
  n=length(specy)
  #Calculate AUC, which is the integral of the ROC
    ll=1-specy
  #Difference between specificity values
    uu=ll[1:(n-1)]-ll[2:n]
    #Average between two neighboring sensitivity values
    vv=(sensy[1:(n-1)]+sensy[2:n])/2

    #Approximate AUC using trapezoidal rule
    auc=sum(uu*vv)
    round(auc,digits=3)
```

```
[1] 0.745
```

This indicates that the estimated AUC for this particular risk score is 0.745, a value considered reasonably good for this type of problem. A perfect AUC would be equal to 1, indicating both perfect sensitivity and specificity, though such an ROC would not be realistic in most applications. A value of AUC above 0.5 indicates that the predictor is doing better than chance, though below we will provide a more precise definition of the interpretation of AUC and of what we mean by "chance."

6.3.5 Choosing the threshold for the risk score

In practice we often want to make a decision or prediction based on the risk score and not just present a risk score. How exactly do we do that? The answer, as often is the case in biostatistics, will depend on the particular problem. Here we present five choices that are reasonable and the pros and cons associated with them.

One choice is to select the threshold such that the percent of individuals who are predicted to have the disease is equal to the percent of individuals who actually have the disease. In our SHHS example, 1004 individuals have the disease among the 5804 individuals in SHHS for a prevalence of $\widehat{P}(D+) = 0.173$. Thus, we could choose a threshold to ensure that $P(T+) \approx 0.173$. The proportion of individuals who test positive for the disease is stored in the vector `P_positive_tests`. To obtain this particular threshold of interest the following code can be used

```
index_thresh_1=which.min(abs(P_positive_tests-0.173))
thresh_1<-round(thresh[index_thresh_1],digits=2)
sensy_1<-round(sensy[index_thresh_1],digits=2)
specy_1<-round(specy[index_thresh_1],digits=2)
```

This threshold is 7.62 and corresponds to a relatively low sensitivity of 0.4 and

reasonable specificity of 0.87.

Another choice would be to maximize the sum between sensitivity and specificity. This is a popular choice, though not always particularly easy to defend in practice. In our case this can be done as follows

```
index_thresh_2 = which.max(sensy + specy)
thresh_2<-round(thresh[index_thresh_2],digits=2)
sensy_2<-round(sensy[index_thresh_2],digits=2)
specy_2<-round(specy[index_thresh_2],digits=2)
n.green.points=N_false_positives[index_thresh_2]
```

This threshold is 6.82 and corresponds to a sensitivity of 0.76 and specificity of 0.61. In the SHHS such a low specificity is probably not reasonable, as it would lead to 1871 false positive tests.

A third choice that is rarely used in practice, but seems to make sense, is to maximize the product between sensitivity and specificity. This can be done as follows

```
index_thresh_3=which.max(sensy*specy)
thresh_3<-round(thresh[index_thresh_3],digits=2)
sensy_3<-round(sensy[index_thresh_3],digits=2)
specy_3<-round(specy[index_thresh_3],digits=2)
n.green.points=N_false_positives[index_thresh_3]
```

This threshold is 6.96 and corresponds to a sensitivity of 0.72 and specificity of 0.66. These results are close to the results obtained by maximizing the sum of sensitivity and specificity and, in this case, lead to slightly lower sensitivity and slightly higher specificity.

Yet, another strategy could be to minimize the sum of false positive and false negative tests, which is equivalent to minimizing the misclassification rate, where either a positive or a negative misclassification is weighted equally. This can be done as follows

```
index_thresh_4=which.min(N_false_positives+N_false_negatives)
thresh_4<-round(thresh[index_thresh_4],digits=2)
sensy_4<-round(sensy[index_thresh_4],digits=2)
specy_4<-round(specy[index_thresh_4],digits=2)
```

This threshold is 8.55 and corresponds to a sensitivity of 0.08 and specificity of 0.99. This test is almost identical to the one using the threshold 8.54, the risk score of the second subject in the SHHS. This score can actually be adapted to the case when the cost for making a false positive and false negative error is known. For example, if it was twice as expensive to make a false positive error than a false negative one, then we would simply change the criterion to `2*N_false_positives+N_false_negatives`, which now has a cost interpretation. Of course, more complex cost structures can be used and this type of

analysis is the basis for a risk/cost study of risk prediction.

Probably the most popular approach is to simply inspect the trade-off between sensitivity and specificity and choose a threshold that is reasonable for the particular study. For example, we would like to have at least 90% specificity because we know that the disease is relatively rare in the population and do not want to make too many false positive mistakes. This is equivalent to setting a maximum limit on the false negative errors. A similar approach can be used for sensitivity, but in our example the lower prevalence seems to be the driving force behind the choice of the threshold.

```
index_thresh_5=which.min(abs(specy-0.9))
thresh_5<-round(thresh[index_thresh_5],digits=2)
sensy_5<-round(sensy[index_thresh_5],digits=2)
specy_5<-round(specy[index_thresh_5],digits=2)
```

This threshold is 7.75 and corresponds to a sensitivity of 0.33 and specificity of 0.9.

We conclude that there are different approaches to choosing the threshold and that none of them is absolutely better than the other. The ROC curve characterizes the prediction performance of the risk score for the entire range of decision thresholds. However, the specific requirements of the problem may require different choices of strategy for choosing the specific threshold. We tend to prefer cost-based thresholds or fixing a high specificity threshold in cases when the disease is relatively rare. The later approach imposes a limit on the number (or cost) of false negative tests, which can be very large especially in low or moderate prevalence problems. Choosing a threshold to ensure that the proportion of individuals predicted to have the disease is equal to the proportion of disease who actually have the disease is, at first, appealing. However, for low to moderate AUCs and low to moderate disease prevalence, this seems to lead to a large number of false negatives. Thus, even if the number of test positives is equal to the number of disease positive individuals, this is not very helpful if the two do not overlap that much. Thus, it seems more reasonable to focus directly on the amount of overlap, or lack thereof.

6.3.6 Sampling variability of ROC and AUC

As we have emphasized, the empirical ROC and AUC are random variables and are affected by sampling variability. To see and quantify this, we can redo all calculations using a bootstrap of subjects and plot both the distribution of ROC curves as well as the distribution of AUC among the bootstrap samples.

```
#Sensitivity is a 100 by 100 matrix
#One row per boostrap sample
sensy=matrix(rep(NA,10000),nrow=100)
specy=sensy
```

```
ppv=sensy
npv=sensy

#AUc is a 100 dimensional vector
#One entry per boostrap sample
auc=rep(NA,100)
```

We will simulate 100 using bootstrap with replacement from the data. For every dataset we rerun the regression and obtain the risk score. Based on the new risk score we calculate the characteristics of the test.

```
set.seed(193618)
n.boot=100
for (k in 1:n.boot)
  {#Sample the index of subjects with replacement
  index_resample<-sample(1:length(age_s1),replace=TRUE)
  data.temp<-data.cv[index_resample,
                     c("gender", "age_s1", "bmi_s1", "HTNDerv_s1")]
  rdi4p_temp<-rdi4p[index_resample]
  outcome.temp<-MtS_SA[index_resample]
  fit<-glm(outcome.temp~gender+age_s1+bmi_s1+HTNDerv_s1,
           family="binomial",data=data.temp)

  #Obtain the linear predictors for every outcome
  #Note that we subtract the intercept because it does not matter
  #and because we want the predictors to be on the same scale with the original ones
  coefs = coef(fit)
  coef_names = names(coefs)
  var_names = intersect(colnames(data.temp), coef_names)
  data.temp = as.matrix(data.temp[, var_names])
  predictor_rdi4p<- data.temp %*% as.vector(coefs[var_names])
  thresh=seq( min(predictor_rdi4p,na.rm=TRUE),
              max(predictor_rdi4p,na.rm=TRUE),length=100)


    for (i in 1:length(thresh))
    {#begin calculating relevant functions for every threshold
      threshold = thresh[i]
      #True positives
      red.points = (predictor_rdi4p > threshold) & (rdi4p_temp >= 15)
      #False positives
      green.points = (predictor_rdi4p > threshold) & (rdi4p_temp < 15)
      #False negatives
      orange.points = (predictor_rdi4p <= threshold) & (rdi4p_temp >= 15)
      #True negatives
      black.points = (predictor_rdi4p <= threshold) & (rdi4p_temp < 15)
```

```
  #Sensitivity and specificity at threshold T
  sensy[k, i] = sum(red.points, na.rm = TRUE) /
    (sum(red.points, na.rm = TRUE) + sum(orange.points, na.rm = TRUE))
  specy[k, i] = sum(black.points, na.rm = TRUE) /
    (sum(black.points, na.rm = TRUE) + sum(green.points, na.rm = TRUE))

  #PPV and NPV at threshold T
  ppv[k, i] = sum(red.points, na.rm = TRUE) /
    (sum(red.points, na.rm = TRUE) + sum(green.points, na.rm = TRUE))
  npv[k, i] = sum(black.points, na.rm = TRUE) /
    (sum(black.points, na.rm = TRUE) + sum(orange.points, na.rm = TRUE))
    }#end iterations over thresholds for the kth bootstrap sample

  #number of thresholds
  n=dim(specy)[2]
  #Calculate AUC, which is the integral of the ROC
    ll=1-specy[k,]
  #Difference between specificity values
    uu=ll[1:(n-1)]-ll[2:n]
    #Average between two neighboring sensitivity values
    vv=(sensy[k,1:(n-1)]+sensy[k,2:n])/2

    #Estimate AUC for the kth dataset
    auc[k]=sum(uu*vv)
}#End bootstrap iterations
```

An investigation of the code indicates that we are incorporating the variability of the estimation the risk score in the procedure in addition to the variability induced by nonparametric bootstrap sampling. This is a more realistic representation of the variability, though we could have kept the risk score fixed, as well. We now plot the first ROC curves to understand their variability and the source of uncertainty in AUCs. Note that even for a relatively large sample size ($n = 5804$) and a reasonably large prevalence ($\widehat{P}(D+) = 0.173$) there is quite a bit of variability in estimated ROCs. This variability is often overlooked and rarely reported in applications. This is especially problematic in cases where one compares the improvement in prediction using an additional covariate or biomarker to a pre-existent risk score. Indeed, in many situations the observed improvement in one sample can be well within the confidence bands of the ROC curves.

```
plot(1-specy[1,],sensy[1,],xlab="1-Specificity",ylab="Sensitivity",
     type="l",lty=2,lwd=2,col="red",axes=FALSE,cex.lab=1.3)
lines(c(0,1),c(0,1),lwd=3)
axis(1,at=seq(0,1,length=5),cex.axis=1.3,col.axis="blue")
axis(2,at=seq(0,1,length=5),cex.axis=1.3,col.axis="blue")
```

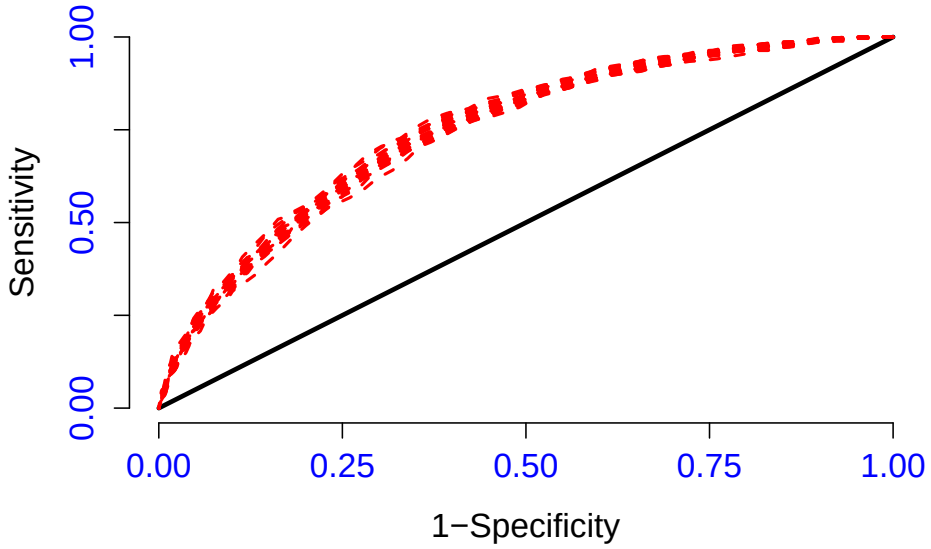


Figure 6.21: Variability of the receiver operating characteristic (ROC) bootstrap resampling of the SHHS (results shown for 20 resamples). For each bootstrap sample the model is refit and a different set of coefficients is used to calculate the ROC.

```
for (i in 2:20)
  {lines(1-specy[i,],sensy[i,],lty=2,lwd=2,col="red")}
```

To quantify visually the variability in estimated ROCs, Figure 6.22 displays the histogram of the 100 estimated AUCs using the nonparametric bootstrap. While on the original data we estimated an AUC of 0.745, we see that in repeated bootstrap samples the range of possible AUCs can vary quite a bit from around 0.72 to 0.77. This heterogeneity is due to sampling variability.

```
hist(auc,probability=TRUE,col=rgb(0,0,1,1/4),breaks=20,xlab="AUC",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

6.3.7 Calibration of risk scores

So far we have focused on the prediction (or discrimination performance), but in prediction we also need to assess calibration. Calibration of the risk score is the ability to accurately predict the absolute risk level (Crowson, Atkinson, and Therneau 2016). To assess calibration we use the Hosmer-Lemeshow test (Hosmer and Lemeshow 2013). More precisely, the study population is partitioned in several subgroups and in each subgroup the expected number of events is compared to the observed number of events. The expected number of events

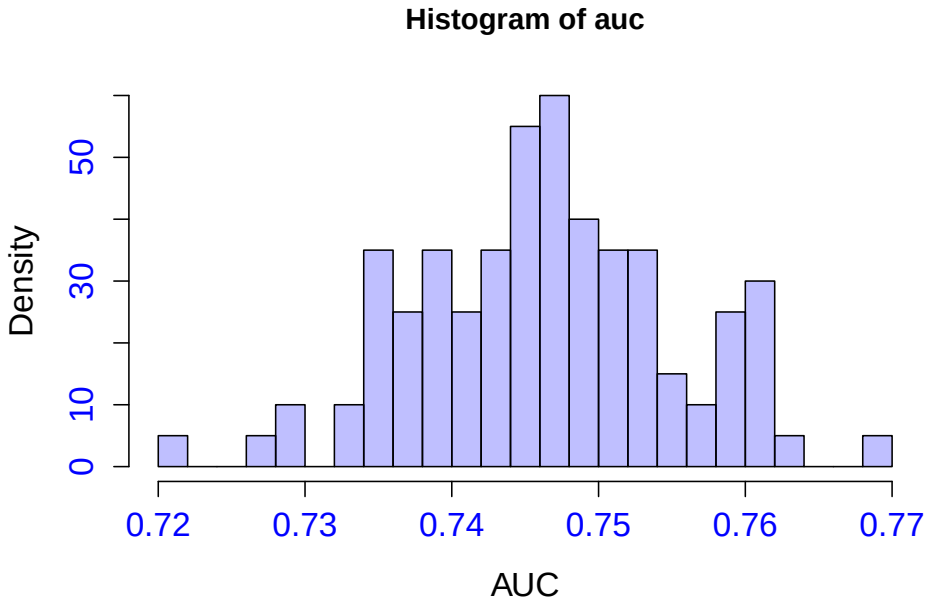


Figure 6.22: Histogram of the ROC obtained from bootstrap resampling of the SHHS. For each bootstrap sample the model is refit and a different set of coefficients is used to calculate the ROC.

is computed as the sum of the predicted probabilities for the patients in that group.

```
library(ResourceSelection)
fit<-glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1,family="binomial",data=data.cv)

#Identify subjects who do not have missing observations
#these are the subjects for which predictions can be calculated
is.not.missing<-!is.na(age_s1) & !is.na(bmi_s1) & !is.na(HTNDerv_s1)

#Predicted probabilities
pred.prob<-fitted(fit)


#Just outcomes who have no missing covariates
outcome_not_missing<-MtS_SA[is.not.missing]

#Apply the Hosmer-Lemeshow test
hoslem.test(outcome_not_missing, pred.prob,g=10)
```

```
    Hosmer and Lemeshow goodness of fit (GOF) test
```

```
data:  outcome_not_missing, pred.prob
X-squared = 2.0199, df = 8, p-value = 0.9804
```

The p-value of the test is 0.98 indicating that there is no evidence against the null hypothesis that the observed number of events is inconsistent with the predicted number of events. The Hosmer-Lemeshow test for calibration on the same dataset is a χ^2 test with $g-2$ degrees of freedom, where g is the number of groups. However, we would like to dig a little deeper and provide a visual representation of what exactly is calculated and fed into the test. Note that `pred.prob` contains the predicted probabilities for all complete case subjects (individuals who have no missing data). This vector is then split into 10 subvectors organized by increasing risk and then the number of events in each subgroup is calculated. More precisely

```
#Define the number of groups
g=10

#Observed number of events
n.observed<-rep(NA,g)

#Expected number of events
n.expected<-n.observed

#Number subjects per group
n.subjects<-n.observed

#Prediction interval
pred.interval<-cbind(n.observed,n.expected)

#Identify the quantiles of risk
risk_thr<-c(0,quantile(pred.prob,seq(0.1,1,0.1)))

for (i in 1:g)
{
  #identify all subjects whose predicted risk score is in the ith bin
  index_risk_set<-(pred.prob>=risk_thr[i]) & (pred.prob<risk_thr[i+1])
  #Number of observed events
  n.observed[i]=sum(outcome_not_missing[index_risk_set])
  #Number of expected events
  n.expected[i]=sum(pred.prob[index_risk_set])
  #Number of subjects per group
  n.subjects[i]=sum(index_risk_set)
  #Prediction interval for the number of observed events
  pred.interval[i,]<-qpois(c(0.025,0.975),n.expected[i])
}
```

```
results<-round(cbind(n.expected,pred.interval,n.observed),digits=2)
colnames(results)<-c("Expected number","Lower limit",
                     "Upper limit","Observed number")
rownames(results)=rep("",dim(results)[1])
results
```

Expected number	Lower limit	Upper limit	Observed number
16.12	9	24	15
28.12	18	39	27
40.40	28	53	43
54.84	41	70	52
71.72	56	89	71
91.22	73	110	89
114.80	94	136	124
140.95	118	165	145
177.86	152	204	174
262.10	231	294	259

We have listed the number of expected and observed events together with the 95% prediction interval for observed number of events given the expected number of events. This confidence interval is based on the assumption that the number of events have a Poisson distribution with the mean equal to the number of expected events in that particular group. The Hosmer-Lemeshow test simply compares the observed and expected number of events in each group. Figure 6.23 provides a visual representation of these results

```
require(ggplot2)
df <- data.frame(x =n.expected,
                 y =n.observed,
                 L =pred.interval[,1],
                 U =pred.interval[,2])

p<-ggplot(df, aes(x = x, y = y)) +
  geom_point(size = 4) +
  geom_errorbar(aes(ymax = U, ymin = L))
p + geom_abline(intercept = 0, slope = 1,col="blue") +
  labs(x = "Expected number of events",
       y="Observed number of events")+
  theme(axis.text=element_text(size=12,color="blue"),
        axis.title=element_text(size=14,face="bold",color="blue"))
```

We can make the same plot, but on the probability scale, or the risk of suffering from sleep apnea. Figure 6.24 provides similar information with Figure 6.23, but on the probability, not on the counts scale. The probability scale calibration plot is sometimes preferred in practice because the x- and y- scales remain consistent across applications and provide information about absolute risk as a function of

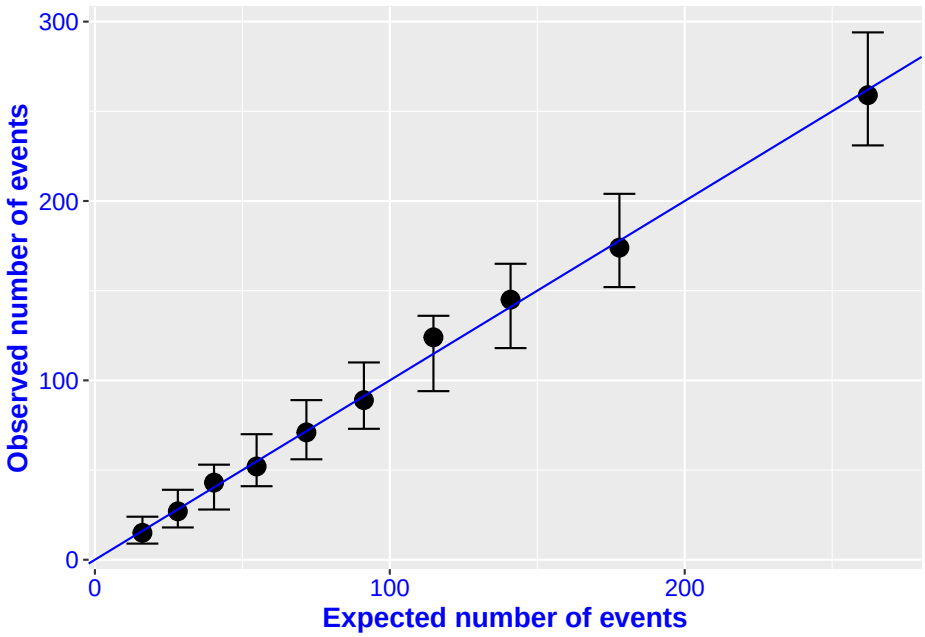


Figure 6.23: The observed versus the predicted number of events together with the 95% prediction interval for the observed given the expected number of events assuming a Poisson distribution of counts. The blue line is the 45 degrees line.

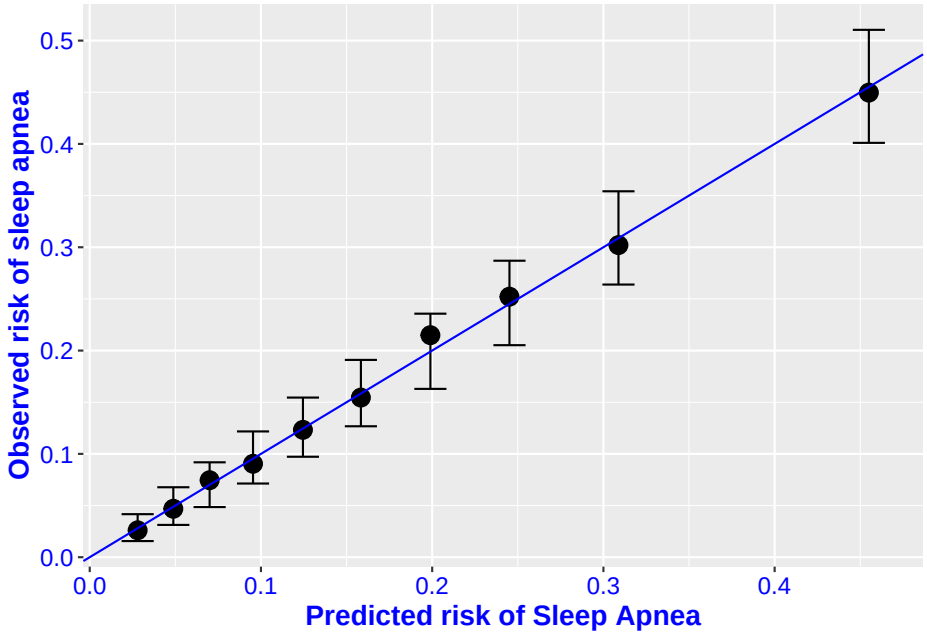


Figure 6.24: The observed versus the predicted risk of sleep apnea together with the 95% prediction interval for the observed given the expected number of events assuming a Poisson distribution of counts. The blue line is the 45 degrees line.

risk bin number. In contrast, the plot based on the number of events depends on the size of the population at risk, which may vary from study to study. In both plots the 45 degree (blue line) passes through the confidence intervals indicating that the prediction equation is well calibrated.

```
df <- data.frame(x =n.expected/n.subjects,
                 y =n.observed/n.subjects,
                 L =pred.interval[,1]/n.subjects,
                 U =pred.interval[,2]/n.subjects)

p<-ggplot(df, aes(x = x, y = y)) +
  geom_point(size = 4) +
  geom_errorbar(aes(ymax = U, ymin = L))
p + geom_abline(intercept = 0, slope = 1,col="blue") +
  labs(x = "Predicted risk of Sleep Apnea",
       y="Observed risk of sleep apnea")+
  theme(axis.text=element_text(size=12,color="blue"),
        axis.title=element_text(size=14,face="bold",color="blue"))
```

The confidence intervals on the probability plot could also be obtained using a

Binomial approach. For example, in the first bin there are 576 subjects with an expected probability of an event 0.028, because the number of expected events is 16.12. Since this is a Binomial$(576, 0.028)$ we need to predict how many events we could have observed given this distribution. This can be obtained as follows

```
L1<-qbinom(0.025,n.subjects[1],n.expected[1]/n.subjects[1])
U1<-qbinom(0.975,n.subjects[1],n.expected[1]/n.subjects[1])
c(L1,U1)
```

```
[1]  9 24
```

This is the same interval we obtained with the Poisson distribution because this Binomial distribution is well approximated by the Poisson(576×0.028). To obtain the probability interval we need to divide by the number of subjects at risk

```
round(c(L1,U1)/n.subjects[1],digits=3)
```

```
[1] 0.016 0.042
```

Both the Binomial and Poisson distributions could be approximated by a Normal distribution with mean and variance equal to `n.expected[1]` $= 16.12 \approx 576 \times 0.028$, where the rounding error in the probability estimation is responsible for the approximate sign. Thus, in general, the 95% calibration confidence interval could be obtained as

$$\mathrm{N}_E \pm 2\sqrt{\mathrm{N}_E}\,,$$

where N_E is the expected number of events for a bin. The corresponding probability confidence interval could be obtained as

$$\frac{\mathrm{N}_E \pm 2\sqrt{\mathrm{N}_E}}{\mathrm{N}_B}\,,$$

where N_B is the number of subjects per bin. In the case of our first bin $N_E = 16.12$ and $N_B = 576$. This approximation assumes that the number of subjects per bin is relatively large and that the expected number of events per bin is also relatively large. When this is not the case we suggest using the Binomial approach. When the expected number of events is small the Binomial approximation will not provide an exact confidence interval because it is a discrete variable and the exact tail probabilities may not match the nominal ones. While we have not yet discussed confidence intervals, we will discuss them in depth in later chapters. Here the confidence interval is an interval of possible number of events given an expected number of events. To obtain that we used basic probability distribution results.

6.4 Problems

Problem 1. Three tennis players, Serena, Maria, and Simona, are playing in a tournament according to the following rules: two of them are chosen at

random and play against each other. The winner of the match plays against the remaining player and the winner of the second match wins the tournament. Because players have met before there is historic information about probability of winning: Serena has a 4 to 1 odds of winning against Maria and a 9 to 2 odds of winning against Simona. Maria has a 3 to 2 odds of winning against Simona.

a. What is the probability that Simona will win the tournament if the first match is Serena versus Maria?
b. What is the probability that Simona will win the tournament if the first match is Simona versus Maria?
c. What is the probability that Simona will win the tournament?

Problem 2. The survival probability for a particular type of cancer depends on whether or not the individual has a particular genetic mutation. If the individual has the mutation, then the individual survival time is distributed exponentially with mean 2 years. If the individual does not have the mutation, then the survival time is a Gamma distribution with mean 5 years and variance 4 years. Also, it is known that only 2% of the population have this particular genetic mutation.

a. Knowing that a person survived for $t = 4$ years, what is the probability that he or she had the genetic mutation?
b. What is the variance of the conditional distribution of having the genetic mutation conditional on a surviving time t?

Problem 3. For a set of n random variables X_i, $i = 1, \ldots, n$ show that

$$[X_1, \ldots, X_n] = [X_1|X_2, \ldots, X_n][X_2|X_3, \ldots, X_n] \ldots [X_{n-1}|X_n][X_n] \, ,$$

where $[\cdot]$ and $[\cdot|\cdot]$ are notations for joint and conditional distributions, respectively.

Problem 4. Assume that in a particular population the survival time and the frequency of mutations at the baseline visit are independent conditional on age at the baseline visit. Also assume that the survival time has an exponential distribution with mean $60 - \text{age}$, that the percent of mutations has a Beta distribution with mean 0.002age and variance 0.005, and that the distribution of age at baseline is uniform from 1 to 50.

a. Write down the joint distribution of survival time, frequency of mutations, and age at baseline.
b. Simulate the marginal distribution of the mutation rate in the population.

Note: we say that two random variables, X_1 and X_2, are independent conditional on a third random variable, X_3, if $[X_1, X_2|X_3] = [X_1|X_3][X_2|X_3]$

Problem 5. The Chinese Mini-Mental Status Test (CMMS) is a test consisting of 114 items intended to identify Alzheimer's disease (AD) and dementia among people in China. An extensive clinical evaluation was performed of this instrument, whereby participants were interviewed by psychiatrists and nurses

and a definitive (clinical) diagnosis of AD was made. The table below shows the counts obtained on the subgroup of people with at least some formal education. Suppose a cutoff value of ≤ 20 on the test is used to identify people with AD.

CMMS score	AD-	AD+
0-5	0	2
6-10	0	1
11-15	3	4
16-20	9	5
21-25	16	3
26-30	18	1

a. What is the sensitivity and specificity of the CMMS test using the 20 cutoff?
b. Create a plot of the sensitivity by (1 - specificity), which is the true positive rate versus the false positive rate for all of the cut-offs from 0 to 30. This is called an ROC curve.
c. Graph the positive and negative predictive value as a function of the prevalence of AD.

Problem 6. A website for home pregnancy tests cites the following: "When the subjects using the test were women who collected and tested their own samples, the overall sensitivity was 75%. Specificity was also low, in the range 52% to 75%."

a. Interpret a positive and negative test result using diagnostic likelihood ratios using both extremes of the specificity.
b. A woman taking a home pregnancy test has a positive test. Draw a graph of the positive predictive value by the prior probability (prevalence) that the woman is pregnant. Assume the specificity is 63.5%.
c. Repeat the previous question for a negative test and the negative predictive value.

Problem 7. A new blood test for chronic kidney disease (CKD) is known to be correct 70% of the time when the person does not suffer from CKD and 80% of the time when the person suffers from CKD. She mentions that the test is only 60% accurate when the test is positive in a particular target population. What is the CKD prevalence in that target population?

Problem 8. In the SHHS provide evidence that `age_s1` and `rdi4p` may not be independent by focusing on events $A_1 = \{\texttt{rdi4p} \in I_1\}$ and $B_1 = \{\texttt{age} \in J_1\}$, where I_1 and J_1 are intervals.

Problem 9. In the SHHS risk score example we have defined the risk score as

$$R_i = 1.16 \text{ gender}_i + 0.033 \text{ age}_i + 0.14 \text{ BMI}_i + 0.19 \text{ HTN}_i \, .$$

Consider the case when the risk score is defined instead as

$$L_i = -8.48 + 1.16 \text{ gender}_i + 0.033 \text{ age}_i + 0.14 \text{ BMI}_i + 0.19 \text{ HTN}_i \, ,$$

or

$$P_i = \frac{e^{L_i}}{1+e^{L_i}} = \frac{1}{1+e^{-L_i}} \ .$$

a. Show that all three risk scores provide the same ranking of subjects.
b. Show that every risk score $D_i = f(R_i)$, where $f(\cdot)$ is strictly increasing provides the same ranking of subjects by risk scores.
c. Show that the estimated ROC and AUC do not depend on the choice of $f(\cdot)$, as defined in the previous point.
d. Show empirically using `R` that the estimated ROC and AUC are identical for all three risk scores.

Problem 10. The risk scores L_i and P_i defined in the previous problem can be obtained as follows

```
L<-fit$linear.predictors
P<-fit$fitted.values
```

a. Obtain these vectors for the SHHS example and explain why the length of vectors is shorter than the length of the outcome vector, `MtS_SA`.
b. Given the vector `MtS_SA` and the vector `P` indicate the correspondence between observed outcomes and their predictors, or `fitted.values`. Explain the role of missing data.
c. Check that, indeed, $1/(1+e^{-L}) = P$ and $\log\{P/(1-P)\} = L$ for every entry of the vector.
d. Discuss the pros and cons of using R_i, L_i, and P_i, respectively.

Problem 11. Using the `predictor_rdi4p` identify the highest and lowest 10 risk scores and list the characteristics of these individuals. Compare and explain why they ended up at the extreme ends of the risk score distribution. Check how many of the 10 individuals with the lowest and highest risk scores have moderate to severe sleep apnea.

Problem 12. Obtain the ROC and AUC for random sub-samples of the SHHS with data sizes 100, 500, 1000, and 2000. Plot and interpret your results. For subsamples of 1000 subjects selected with replacement repeat the same process for 100 subsets of data. This process will be called in this book "downstrapping", as it is strongly related to bootstrapping, but it uses a smaller subpopulation from the original population.

Problem 13. Sample with replacement 100 datasets from the SHHS with a number of subjects 1.5×5804 and obtain the ROC and AUC. Compare these results with those obtained using random samples with fewer subjects than in the original data. Draw conclusions about observed differences.

Problem 14. Recall the interpretation of the AUC as

$$P(R_i > R_j | D_i = 1, D_j = 0) \ .$$

Estimate this probability using the SHHS data for moderate to severe sleep apnea as outcome and the predictor discussed in this chapter as risk score. To

do that, calculate

$$\widehat{P}(R_i > R_j | D_i = 1, D_j = 0) = \frac{1}{N_P} \sum_{\{i:D_i=1\}} \sum_{\{j:D_j=0\}} I\{R_i > R_j\} \ ,$$

where $I\{R_i > R_j\} = 1$ if $R_i > R_j$ and 0 otherwise, N_P is the number of pairs of outcomes $D_i = 1$ and $D_j = 0$. Compare this result with the estimated ROC result derived in the Chapter.

Problem 15. Consider the following 10 subjects with their corresponding risk scores and true disease status

PID	risk_score	disease_status
1	2.3	1
2	-1.2	0
3	4.4	1
4	2.2	0
5	5.1	0
6	-0.2	0
7	3.7	1
8	1.6	1
9	3.7	1
10	0.8	0

Enumerate all pairs of patient IDs (i, j) such that the risk score for subject i is larger than the risk score for subject j and the subject i has the disease while subject j does not have the disease. Calculate N_P, the number of pairs of outcomes $D_i = 1$ and $D_j = 0$ and obtain the estimator of the AUC based on the formula in the previous problem.

Chapter 7

Likelihood

This chapter covers the following topics

- Likelihood definition and interpretation
- Maximum likelihood
- Interpreting likelihood ratios
- Likelihoods for multiple parameters
- Profile likelihood

7.1 Likelihood definition and interpretation

A common approach to statistics is to assume that data arise from a family of distributions indexed by a parameter (or set of parameters) that represents a useful summary of the distribution. The likelihood of the data is the joint density evaluated as a function of the parameters with the data fixed. Likelihood analysis of data uses the likelihood to perform inference regarding the unknown parameter. More precisely, given a statistical probability mass function or density, say $f(\mathbf{x}, \theta)$, where θ is an unknown parameter, the **likelihood** is f viewed as a function of θ for a fixed, observed value of $\mathbf{x}$

$$\mathcal{L}(\theta|x) = f(\mathbf{x}, \theta) \ .$$

Here both $\mathbf{x}$ and θ are in bold because they could both be vectors.

7.1.1 Example: Normal likelihood

Consider an experiment where three independent observations, X_1, X_2, X_3, are made from a $N(\mu, 1)$ distribution. For the sake of simplicity, we assume that the variance of the distribution is known and equal to 1. After the experiment

is run the following three values were observed $x_1 = 5$, $x_2 = 2$, and $x_3 = 3$ (note the use of lower case to indicate values obtained after the experiment is run). We denote by $\mathbf{x} = (x_1, x_2, x_3)$ the vector of observations, where we used bold font to indicate vectors. The Normal pdf for a single observation, x, is

$$f(x, \mu) = \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu)^2}{2}\right\} .$$

Therefore, the likelihood for the experiment is

$$\mathcal{L}(\mu | X_1 = 5, X_2 = 2, X_3 = 3) = f(5, \mu) f(2, \mu) f(3, \mu) .$$

After some calculations,

$$\mathcal{L}(\mu | \mathbf{x}) = \frac{1}{(2\pi)^{3/2}} \exp\left\{-\frac{(5-\mu)^2 + (2-\mu)^2 + (3-\mu)^2}{2}\right\} .$$

This is a function of μ and can be calculated for any given value of this parameter. For example, for $\mu = 4$ the likelihood is

```
#Set the parameter where to evaluate the likelihood
mu=4
#Set the observed data
bx=c(5,2,3)
#Calculate the exponent
ebx2=-sum((bx-mu)^2)/2
#Calculate and print the likelihood
like=exp(ebx2)/((2*pi)^(length(bx)/2))
round(like,digits=5)
```

```
[1] 0.00316
```

The values of the likelihood do not have an absolute interpretation, as the likelihood does not typically integrate to 1. However, the relative size of the likelihood (aka likelihood ratio) for two different parameters can be interpreted as the relative evidence in the data about one parameter to another. To better understand that, we calculate the likelihood on a grid of possible parameters and plot it in Figure 7.1. This is obtained by doing the same calculations as above, but at many more possible parameter values (201 to be precise)

```
#Set a fine enough grid of parameters
mu=seq(0,6,length=201)
#Initialize the vector containing the likelihood
likep=rep(0,201)
for (i in 1:201)
  {#begin calculating the likelihood on a grid
   ebx2=-sum((bx-mu[i])^2)/2
   likep[i]=exp(ebx2)/((2*pi)^(length(bx)/2))
  }#end calculating the likelihood at each parameter
plot(mu,likep,type="l",col="blue",lwd=3,xlab="Parameter",ylab="Likelihood")
```

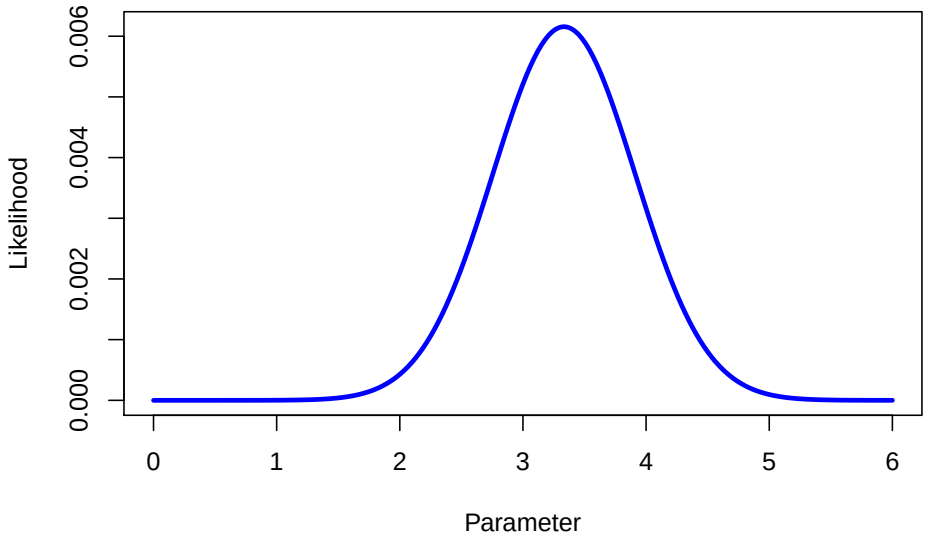


Figure 7.1: Normal likelihood for the mean parameter, μ, when three observations 5, 2, and 3 are available and the standard deviation is known and fixed $\sigma = 1$.

```
#Obtain the maximum likelihood estimator
mle<-mu[which.max(likep)]
round(mle,digits=3)
```

```
[1] 3.33
```

The point where the likelihood attains its maximum is called the maximum likelihood estimator (MLE) and plays a crucial role in biostatistics. In our example the MLE is $3.33 = (5+3+2)/3$, the average of the three observations. This point depends on the observed data because the likelihood does and will be different in a different experiment. Thus, the MLE, like all estimators, is a random variable. In our example the MLE obtained numerically is equal to the mean of the three observations $(5, 2, 3)$. This happens because the underlying distribution is the Normal distribution, though other distributions (e.g., Poisson, Binomial) have the average of the observations as the MLE. However this rule does not generalize to all distributions; the MLE of the population mean need not be the average of the observations.

Let us show formally that the mean is the MLE in the case of Normally distributed data. If $X_1 = x_1, \ldots, X_n = x_n$ are realizations of n iid $N(\mu, 1)$, then the likelihood of the observed data is

$$\mathcal{L}(\mu|\mathbf{x}) = \frac{1}{(2\pi)^{n/2}} \exp\left\{-\frac{\sum_i^n (x_i - \mu)^2}{2}\right\} .$$

In biostatistics one often uses the log-likelihood or minus twice the log-likelihood because log-likelihoods tend to be better behaved than likelihoods. For example, in the case of normal variables we have

$$-2\log\{\mathcal{L}(\mu|\mathbf{x})\} = \sum_i^n (x_i - \mu)^2 + \text{constant} ,$$

where the constant does not depend on the parameter. As an important aside, biostatisticians do not agree on much, but they do agree on the crucial role of likelihood in data analysis. *The reason is that given a data set and a model the likelihood contains all the information about the unknown parameter that is contained in the data.*

7.1.2 Likelihood interpretation

The law of likelihood is a rule that helps to interpret likelihoods. It states:

> **Law of the likelihood:** Ratios of likelihood values measure the relative **evidence** of one value of the unknown parameter to another.

In contrast, the likelihood principle is an arguably provable statement about likelihoods. It states:

> **Likelihood principle:** Given a statistical model and observed data, all of the relevant information contained in the data regarding the unknown parameter is contained in the likelihood.

The phrase "arguably provable" is necessary, as there is some discussion on the validity of the proof of the likelihood principle (Mayo 2014).

A third helpful fact about likelihoods is how they adjust to more data. If X_i, $i = 1, \ldots, n$ are independent random variables, then their likelihoods multiply. That is, the likelihood of the parameters given all of the X_i is simply the product of the individual likelihoods. Of course, this means that the log-likelihoods add.

Assume $X_1 = x_1, \ldots, X_n = x_n$ are iid with pdf $f(x, \theta)$. The likelihood for the observed data vector $\mathbf{x} = (x_1, \ldots, x_n)$ is

$$\mathcal{L}(\theta|\mathbf{x}) = \prod_{i=1}^n f(x_i, \theta) ,$$

while the log-likelihood is

$$\log\{\mathcal{L}(\theta|\mathbf{x})\} = \sum_{i=1}^n \log\{f(x_i, \theta)\} .$$

Note, again, that individual contributions to the log-likelihoods of independent observations add up, whereas they multiply on the likelihood scale. Of course, in practice, it is much easier to add than multiply, especially when numbers are very small, as is often the case with likelihoods. Indeed, when working with likelihoods one often has to be on guard for numerical problems due to rounding errors and very small probabilities.

7.1.3 Example: Bernoulli likelihood

Consider the case when one is interested in whether a person taken at random from the population has the flu and the true proportion of people with the flu is θ. If X denotes the random variable "the person has the flu" then the pmf for X is

$$f(x, \theta) = \theta^x (1-\theta)^{1-x} \quad \text{for} \quad \theta \in [0, 1] \, ,$$

where x is either 0 (does not have the flu) or 1 (has the flu). Suppose that the first individual sampled from the population has the flu. The likelihood for this data point evaluated at every parameter is

$$\mathcal{L}(\theta|1) = \theta^1 (1-\theta)^{1-1} = \theta \quad \text{for} \quad \theta \in [0, 1] \, .$$

Suppose that we are interested in the strength of evidence in favor of half the population being infected versus a quarter of the population being infected. To determine that we calculate the likelihood ratio $\mathcal{L}(.5|1)/\mathcal{L}(.25|1) = 2$, which indicates that there is twice as much evidence supporting the hypothesis that $\theta = .5$ compared to the evidence that $\theta = .25$.

Suppose now that we sample four subjects from the population and get the sequence 1, 0, 1, 1, that is, the first, third, and fourth individuals are infected, but the second is not. The likelihood is

$$\mathcal{L}(\theta|1,0,1,1) = \theta^1 (1-\theta)^{1-1} \theta^0 (1-\theta)^{1-0} \theta^1 (1-\theta)^{1-1} \theta^1 (1-\theta)^{1-1} = \theta^3 (1-\theta)^1 \, .$$

This likelihood only depends on the total number of individuals who are infected and the total number of people who are not infected. We might write $\mathcal{L}(\theta|1,3) = \mathcal{L}(\theta|1,0,1,1)$ to avoid writing really long sequences for data. In this example, the likelihood depends only on the summary statistic $(1, 3)$, which is referred to as the *sufficient statistic*. Sufficient statistics play an important role, because they can be much simpler than the entire dataset and encapsulate all the available information about the model parameter given the data and the model. Now consider $\mathcal{L}(.5|1,3)/\mathcal{L}(.25|1,3) = 5.33$, indicating that after seeing three individuals with the flu and one without there is over five times as much evidence supporting the hypothesis that $\theta = .5$ over the hypothesis that $\theta = .25$.

To plot the likelihood we want to consider all the values of θ from 0 to 1. Figure 7.2 displays the likelihood $\mathcal{L}(\theta|\text{Data})$ on the y-axis versus a grid of parameters θ on the x-axis. Dividing the likelihood by its maximum value provides a level of

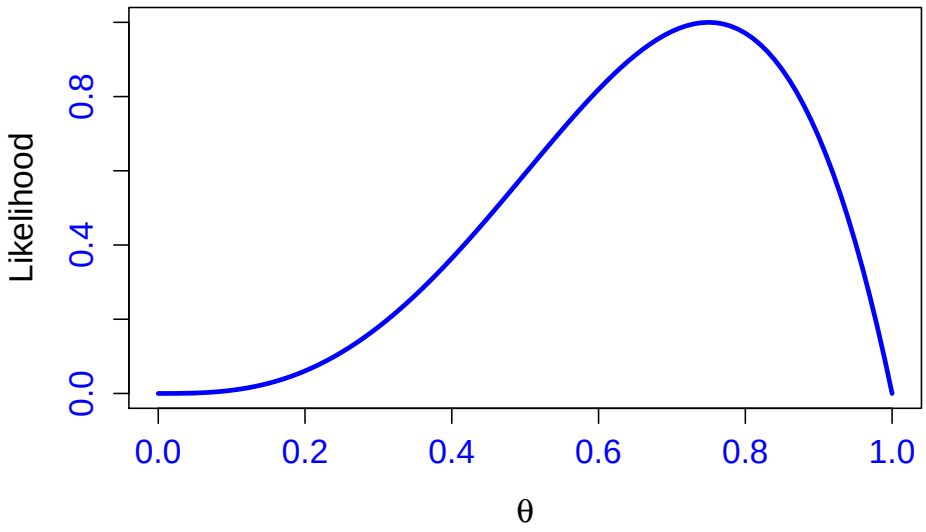


Figure 7.2: Bernoulli likelihood for the success probability when we observe three successes out of four trials.

standardization because its height is 1 (at the maximum likelihood estimator) and the relative strength of evidence can be read from the plot directly for any value of the parameter. Because the likelihood measures *relative evidence*, dividing it by its maximum value (or any other value for that matter) does not change its interpretation.

```
#Set a grid for the likelihood
theta=seq(0,1,by=0.01)
#Calculate the likelihood at every parameter value
like=theta^3*(1-theta)
like=like/max(like)
plot(theta,like,type="l",col="blue",lwd=3,xlab=expression(theta),
     ylab="Likelihood",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

The likelihood is tilted to the right, indicating that the evidence favors higher probabilities of infection in the population. However, the likelihood is still relatively dispersed with reasonable evidence for a wide range of values, anywhere from around 0.4 to around 0.95. This happens because there are only four observations from the population. The maximum evidence is attained at

```
MLE=theta[which.max(like)]
round(MLE,digits=2)
```

```
[1] 0.75
```

which is the maximum likelihood estimator and happens to be equal to 3/4, the number of individuals who were infected divided by the number of individuals

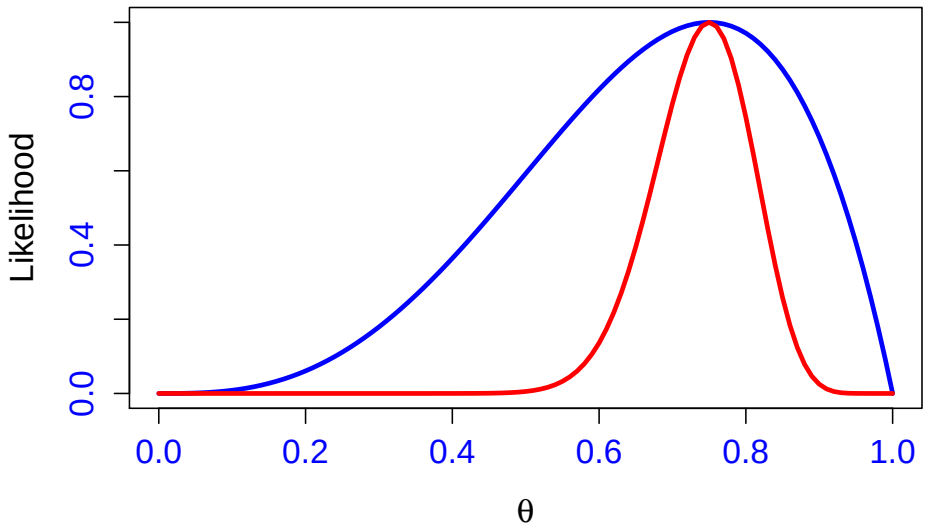


Figure 7.3: Bernoulli likelihood for the success probability when we observe three successes out of four trials (blue line) and thirty successes out of forty trials (red line).

sampled from the population. Indeed, consider the case when 30 out of 40 individuals are infected instead of 3 out of 4. Then the likelihood is $\mathcal{L}(\theta|10,30) = \theta^{30}(1-\theta)^{10}$ and is dsiplayed in Figure 7.3 as a red line, which can be compared to the likelihood for three successes out of four trials (blue line).

```
#Set a grid for the likelihood
theta=seq(0,1,by=0.01)
#Calculate the likelihood at every parameter value
like=theta^3*(1-theta)
like=like/max(like)
plot(theta,like,type="l",col="blue",lwd=3,xlab=expression(theta),
     ylab="Likelihood",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
like1<-theta^30*(1-theta)^10
like1<-like1/max(like1)
lines(theta,like1,lwd=3,col="red")
```

As anticipated, the interval of values of the θ parameter that are consistent with the observed data is much smaller and the concentration of likelihood is much more pronounced around the MLE, 0.75. Moreover, the shape of the likelihood is much better approximated by a Normal when the number of observations increases. This is a general property of the likelihood for independent observations, but we neither prove nor use this property of the likelihood in this book.

7.1.4 Example: Uniform distribution

Assume now that the three observations (5, 2, 3) are obtained from a Uniform distribution on $[0, \theta]$ (denoted as $U[0, \theta]$), where θ is an unknown scalar parameter. The pdf of the distribution is

$$f(x, \theta) = \begin{cases} \frac{1}{\theta} & \text{if } 0 \leq x \leq \theta \, ; \\ \\ 0 & \text{otherwise} \, . \end{cases}$$

The likelihood function is

$$\mathcal{L}(\theta|\mathbf{x}) = \frac{1}{\theta} I\{5 \leq \theta\} \frac{1}{\theta} I\{2 \leq \theta\} \frac{1}{\theta} I\{3 \leq \theta\} = \frac{1}{\theta^3} I\{5 \leq \theta\} \, ,$$

where $I\{\cdot\}$ is the indicator function. The last equality holds because θ is larger than 2, 3, and 5 if and only if θ is larger than 5. Here the likelihood depends on the parameter and on the data only through the largest observation, 5. This is the realization of the sufficient statistic, which is $\max(X_1, X_2, X_3)$. The support of the likelihood (the interval where the likelihood is not zero) depends on the data in the sample, which is different from the Normal distribution discussed earlier. Figure 7.4 displays the Uniform likelihood for this sample

```
#Set a grid for the likelihood
theta=seq(1,10,by=0.1)
#Calculate the likelihood at every parameter value
like=1/theta^3*(theta>=5)
#Normalize the likelihood
like=like/max(like)
plot(theta,like,type="l",col="blue",lwd=3,xlab=expression(theta),
     ylab="Likelihood",cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

The MLE can be obtained numerically, as in the previous example

```
theta[which.max(like)]
```

```
[1] 5
```

indicating that the MLE is 5. This can also be shown theoretically as the likelihood at every value of $\theta < 5$ is zero because if the true θ were smaller than 5, then the $U[0, \theta]$ could not have produced a 5. For every value of $\theta > 5$ we have $1/\theta^3 < 1/5^3$. The likelihood provides more information than just where the maximum is attained. Indeed, it provides information about how fast the likelihood (evidence) decreases away from the maximum

```
#Likelihood ratio for $\theta=6$ versus $\theta=5$
round(like[theta==6]/like[theta==5],digits=3)
```

```
[1] 0.579
```

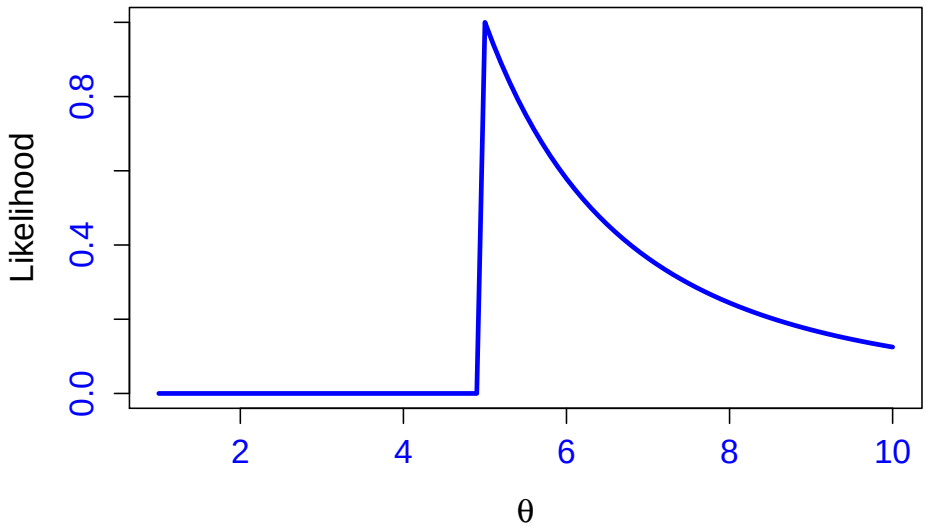


Figure 7.4: Uniform likelihood for the upper limit of the Uniform distribution with three independent observations 5, 2, and 3.

This indicates that there is close to half as much evidence about $\theta = 6$ relative to $\theta = 5$. However, in the case when comparing $\theta = 6$ with $\theta = 4$

```
#Likelihood ratio for $\theta=6$ versus $\theta=4$
round(like[theta==6]/like[theta==4],digits=3)
```

```
[1] Inf
```

```
# maximum likelihood
```

there is infinitely more evidence in favor of $\theta = 6$ relative to $\theta = 4$ simply because seeing a realization of 5 from a $U(0, \theta)$ distribution rules out $\theta = 4$.

7.1.5 Bayesian interpretation of the likelihood

A density $f(\mathbf{x}, \mu)$, which can be thought of as the conditional distribution of $\mathbf{x}$ (data) given μ (parameter), is a proper pdf with respect to $\mathbf{x}$. More precisely,

$$\int f(\mathbf{x}, \mu) d\mathbf{x} = 1 \ ,$$

for every μ. However, $\mathcal{L}(\mu|\mathbf{x})$ is not always a proper pdf with respect to μ. That is,

$$K(\mathbf{x}) = \int_{-\infty}^{\infty} \mathcal{L}(\mu|\mathbf{x}) d\mu \neq 1 \ .$$

Of course, if $K(X)$ is finite, the likelihood can be turned into a proper pdf by dividing by the constant $K(\mathbf{x})$, though this constant is often hard to calculate and does not help with the interpretation of the likelihood.

It is tempting to think about the pdf as the conditional distribution of $[\mathbf{x}|\mu]$ and of the likelihood as the conditional distribution $[\mu|\mathbf{x}]$. Specifically, we know from the Bayes rule that

$$[\mu|\mathbf{x}] \propto [\mathbf{x}|\mu][\mu] \ .$$

If, a priori, nothing is known about μ then we could assume that $[\mu] \propto 1$; that is, the prior distribution of the parameter is uniform. Therefore,

$$[\mu|\mathbf{x}] \propto [\mathbf{x}|\mu] = \mathcal{L}(\mu|\mathbf{x}) \ ,$$

which indicates that the posterior distribution of the parameter μ is equal to the conditional distribution of $[\mathbf{x}|\mu]$, up to a normalizing constant. This normalizing constant is $K(\mathbf{x})$, which is the marginal pdf of the data, after integrating over the parameters. This provides a useful Bayesian interpretation of the likelihood as the **posterior distribution of the parameter given the observed data if the prior distribution on the model parameters is uniform.** As mentioned earlier, $K(\mathbf{x})$ need not be finite, and $[\mu] \propto 1$ is not a proper prior if μ is unbounded. However, the uniform distribution places explicit bounds on μ, thus the prior is proper and typically $K(\mathbf{x})$ will also be finite. Therefore, the rule holds in these more difficult cases as well, though we may not know the explicit bounds which μ lies in.

Likelihood ratios are important in every branch of model-based statistics. Frequentists use likelihood ratios to create hypothesis tests. In the Bayesian context the ratios of posteriors evaluated at two parameter values are likelihood ratios times ratios of the prior. Therefore, it is on the importance of likelihood ratios that model based frequentists and Bayesians come closest to agreeing.

Appealing to the likelihood principle and the law of the likelihood, some statisticians directly use the likelihood for inference. Frequentists base their use of the likelihood on hypothesis tests and confidence intervals based on likelihood ratios. In the Bayesian context, the likelihood contains the information in the data for forming the posterior, and is proportional to the posterior if the prior is flat.

While the interpretations are quite different, the likelihood serves as a fundamental component in all three foundational philosophies (frequentist, Bayesian, likelihood). Therefore, philosophical differences in interpretation aside, if one has dramatically different inferences depending on which style of inference is used, it is worth investigating the source of the discrepancy. Since the key data ingredient in model-based statistics is the likelihood, it is useful to check if the prior, algorithms or calibration across the methods is the driver of the discrepancy.

7.2 Maximum likelihood

As we have already discussed, the value of θ parameter where the likelihood curve reaches its maximum has a special meaning. It is the value of θ that is most well supported by the data and is called the **maximum likelihood estimate** (or MLE) of θ

$$\widehat{\theta}_{\mathrm{ML}}(\mathbf{x}) = \mathrm{MLE} = \mathrm{argmax}_{\theta}\mathcal{L}(\theta|\mathbf{x}) \ .$$

Another interpretation of the MLE is that it is the value of θ that would make the data that we observed most probable. We have shown explicitly the dependence of the MLE on the observed data and reiterate that the MLE is a random variable, which has its own variability and properties.

7.2.1 Example: Bernoulli MLE

We consider again the case when one is interested whether a person taken at random from the population has the flu and the true proportion of people with the flu is θ. Consider the general case when we sample n independent individuals from this population and let X_i be the random variable "the ith person has the flu" and let x be the number of individuals who have the flu among the n sampled individuals. We already know that

$$\mathcal{L}(\theta|x) = \theta^x(1-\theta)^{n-x} \ ,$$

and the log-likelihood is

$$l(\theta, x) = \log\{\mathcal{L}(\theta|x)\} = x\log(\theta) + (n-x)\log(1-\theta) \ .$$

Taking the derivative of the log-likelihood we obtain

$$\frac{d}{d\theta}l(\theta, x) = \frac{x}{\theta} - \frac{n-x}{1-\theta} \ ,$$

and setting this equal to zero implies

$$\frac{x}{\theta} = \frac{n-x}{1-\theta} \Rightarrow x(1-\theta) = (n-x)\theta \Rightarrow x = n\theta \Rightarrow \theta = \frac{x}{n} \ .$$

Thus the log-likelihood has a critical point at x/n (point where the first derivative is zero). The second derivative of the log-likelihood function is

$$\frac{d^2}{d\theta^2}l(\theta, x) = -\frac{x}{\theta^2} - \frac{n-x}{(1-\theta)^2} < 0 \ ,$$

which is strictly negative for every value of $\theta \in (0, 1)$. This indicates that the log-likelihood is concave (its graph does not hold water), which completes the

proof that it has a unique maximum and that its maximum is attained at the critical point x/n.

When X is not equal to n or 0, we can also check that the likelihood, $\theta^x(1-\theta)^{n-x}$, is zero when θ is 0 and 1 and is strictly positive for $\theta \in (0,1)$. Therefore, we can say that x/n is the maximum, including the boundaries 0 and 1. When X is 0 or n, the likelihood is either $(1-\theta)^n$ or θ^n respectively. Consider the case where $X = 0$. The likelihood, $(1-\theta)^n$, is strictly decreasing on $[0,1]$ and therefore $\hat{\theta} = 0$. In contrast, when $X = n$, the likelihood, θ^n, is strictly increasing on $[0,1]$ and so $\hat{\theta} = 1$. Thus, no matter the setting, the MLE is $\hat{\theta} = x/n$.

As an aside, likelihoods that have the property that the log-likelihood is concave are referred to log-concave likelihoods; they tend to be better behaved numerically and inferentially. However, not all likelihoods are log-concave.

7.2.2 Example: Binomial likelihood

The binomial random variables are obtained as the sum of iid Bernoulli random variables. Let $X_1, \ldots, X_n$ be iid Bernoulli(p). Then $X = \sum_{i=1}^n X_i$ is a Binomial random variable with a probability mass function

$$P(X = k) = \left(\begin{array}{c} n \\ k \end{array} \right) p^k (1-p)^{n-k} ,$$

for $k = 0, \ldots, n$, where

$$\left(\begin{array}{c} n \\ k \end{array} \right) = \frac{n!}{x!(n-x)!}$$

reads "n choose k" and counts the number of ways of selecting k items out of n without replacement disregarding the order of the items. In particular cases

$$\left(\begin{array}{c} n \\ 0 \end{array} \right) = \left(\begin{array}{c} n \\ n \end{array} \right) = 1 ,$$

because, by convention, $0! = 1$. If X is a random variable with a Binomial distribution with probability p and number of trials n then we denote $X \sim$ Bernoulli(n,p). Note that if $X_1, \ldots, X_n$ are independent Bernoulli(p) then

$$\sum_{i=1}^n X_i = n\overline{X}_n \sim \text{Binomial}(n,p) ,$$

where $\overline{X}_n = \sum_{i=1}^n X_i/n$ is the empirical mean of the n independent Bernoulli random variables.

There is a very close connection, but important distinction, between the probability that a set of independent Bernoulli random variables results in a particular sequence of 0/1 outcomes and the probability that their sum results in exactly

k ones. Indeed, let us consider the example of how likely or unlikely a string of birth sexes is in a family (a question that biostatisticians get asked a lot!).

Consider a family who has eight children and that they were born in the following order

$$F, F, M, F, F, F, F, F \ ;$$

that is, the third is a boy, all other children are girls, and none are twins. Assume that the probability of having a girl is p and that having a boy or a girl are independent events. Then, the probability that 8 independent Bernoulli(p) random variables result in exactly this boy/girl configuration is

$$p^7(1-p) \ .$$

However, if we are interested in the probability of the event "the family has one boy out of 8 children," then all other possibilities of when the boy is born need to be listed. Indeed, the boy could have been born first, second, and so on and so forth giving a total of 8 possibilities. Each one of these possibilities has the same probability, $p^7(1-p)$, indicating that the probability of this event is $8p^7(1-p)$, or, eight times larger than the probability of seeing the exact outcome above. The factor 8 is "8 choose 1" from our formula. (This factor can be quite large, so it is important to notice the difference between the events "the third child is a boy, the rest were girls," and "there was exactly 1 boy".)

Suppose we now know that each sex has an independent 50% probability for each birth. We would like to know what is the probability of getting 7 or more girls out of 8 births? If X denotes the number of girls out of $n = 8$ births, then we need to calculate

$$P(X \geq 7) = P(X = 7) + P(X = 8) = \left(\begin{array}{c} 8 \\ 7 \end{array} \right) .5^7(1-.5)^1 + \left(\begin{array}{c} 8 \\ 8 \end{array} \right) .5^8(1-.5)^0 \approx 0.035 \ .$$

In R this probability can be obtained directly as

```
round(1-pbinom(6,8,0.5),digits=3)
```

```
[1] 0.035
```

or, equivalently,

```
round(pbinom(6, 8, 0.5, lower.tail = FALSE), digits = 3)
```

```
[1] 0.035
```

In both cases, we enter 6 instead of 7, since for upper tail probabilities R is giving $P(X > x)$ with a strict inequality. Therefore, if we want $P(X \geq 7)$ we calculate the equivalent $P(X > 6)$.

This calculation is an example of a p-value, the probability under a null hypothesis (in this case that the probability of having a girl is $p = 0.5$) of getting a result as extreme or more extreme than the one actually obtained (7 girls out

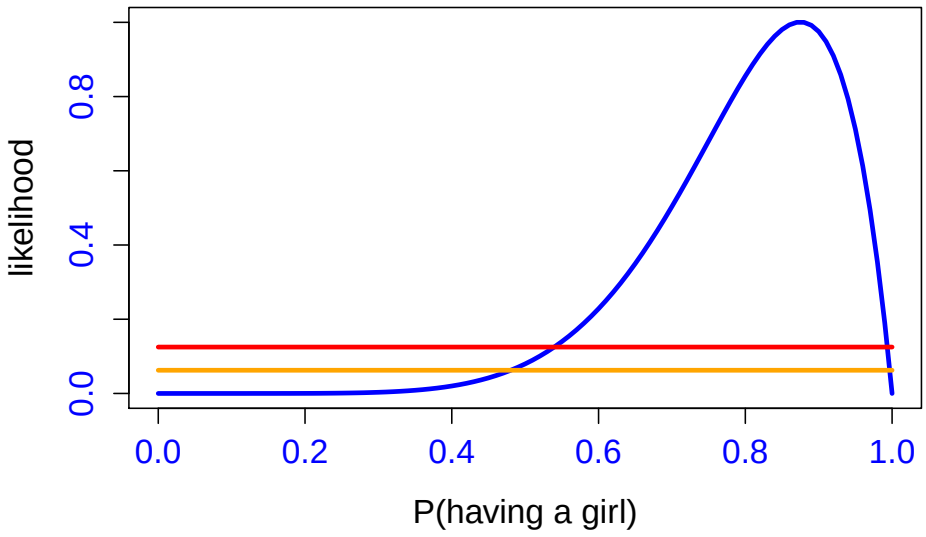


Figure 7.5: Binomial likelihood divided by its maximum for the probability of having a girl, p, for a family who has 7 girls and 1 boy. Thresholds for the ratio of the likelihood with respect to the maximum shown at 1/8 (red horizontal line) and 1/16 (orange horizontal line).

of 8 births). Another interpretation is that if $p = 0.5$, then 3.5% of the families with eight children will have at least 7 girls indicating that this is a rather rare event among families with 8 children.

If we do not know the probability p of having a girl, then the likelihood is

$$\mathcal{L}(p|\mathbf{x}) = \left(\begin{array}{c} 8 \\ 7 \end{array} \right) p^7(1-p) = \frac{8!}{7!1!} p^7(1-p) = 8p^7(1-p) \; .$$

Figure 7.5 displays this likelihood together with some important thresholds at 1/8 and 1/16. We'll build up the intuition over setting likelihood thresholds later on in this chapter.

```
p=seq(0,1,length=101)
like=8*p^7*(1-p)
plot(p,like/max(like),type="l",col="blue",lwd=3,
     xlab="P(having a girl)",ylab="likelihood",cex.lab=1.3,
     cex.axis=1.3,col.axis="blue")
lines(p,rep(1/8,101),col="red",lwd=3)
lines(p,rep(1/16,101),col="orange",lwd=3)
```

We now show a few uses of the Binomial distribution in R. The command below generates 17 independent identically distributed random variables, $X_1, \ldots, X_{17}$, from the distribution Binomial$(10, 0.2)$. Note that the realizations are relatively

small due to the total number of trials, 10, and the small probability of success of each trial, $p = 0.2$.

```
set.seed(3467326)
#Simulate 15 independent Binomial(10,.2)
rbinom(17,size=10,prob=0.2)
```

```
 [1] 3 1 2 1 3 2 2 3 1 1 2 5 2 1 2 1 2
```

R is quite flexible and allows generation of independent, but not identically distributed random variables, as well. Below, we have changed the distribution of the ith random variable to Binomial$(10, p_i)$, where $p_i = 0.1 + 0.05 * (i - 1)$. That is, we allow probabilities of success of each individual trial to increase from a very low 0.1 to a very large 0.9 in small increments. The number of successes towards the end of the series is much larger than the number of successes at the beginning of the series. However, the number of successes does not follow the same strict increasing pattern of the probabilities used to simulate the events. Here the length of the probability vector needs to be equal to the number of samples being simulated. Otherwise R recycles the probability vector if its length is smaller than the number of samples, or uses only the first part of the vector if the length is shorter.

```
# Simulate 15 independent Binomial(10,p)\\
rbinom(17,size=10,prob=seq(0.1,0.9,length=17))
```

```
 [1] 0 1 4 2 3 3 5 3 4 8 6 7 6 7 6 9 9
```

R also allows us to simulate independent Binomial variables with a different number of trials and different success probabilities. Below we show how to simulate independent $X_i \sim$ Binomial(n_i, p_i), where $n_i = i$ and $p_i = 0.1 + 0.05 * (i - 1)$. Both the `size` (containing the total number of samples) and the probability vector have to have the same length and the length be equal to the number of independent samples.

```
# Simulate 15 independent Binomial(n,p)
# First (n,p)=(1,0.1), last (n,p)=(17,0.9)\\
rbinom(17,size=1:17,prob=seq(0.1,0.9,length=17))
```

```
 [1]  0  0  0  2  1  1  3  4  4  8  6  8  9  8 10 12 16
```

7.3 Interpreting likelihood ratios

In this section we relate likelihood ratios to intuition based on basic coin flipping. This is used by statisticians employing the law of the likelihood to analyze data. Our treatment follows that of (Royall 2017).

Consider the same experiment, where we sample individuals at random from a population with the true proportion θ of people having the flu. We entertain

Table 7.1: Relative strength of evidence for three hypotheses depending on the outcome of an experiment.

Outcome	$P(X\|H_1)$	$P(X\|H_2)$	$P(X\|H_3)$	$\frac{L(H_1)}{L(H_2)}$	$\frac{L(H_3)}{L(H_2)}$
P	0	0.500	1	0	2
N	1	1.000	1	1	1
PP	0	0.250	1	0	4
NP	0	0.500	1	0	2
PN	0	0.500	1	0	2
NN	1	1.000	1	1	1
PPP	0	0.125	1	0	8
NPP	0	0.250	1	0	4
PNP	0	0.250	1	0	4
NNP	0	0.500	1	0	2
PPN	0	0.250	1	0	4
NPN	0	0.500	1	0	2
PNN	0	0.500	1	0	2
NNN	1	1.000	1	1	1

three hypotheses: $H_1 : \theta = 0$ (nobody has the flu), $H_2 : \theta = .5$ (half of the population has the flu), and $H_3 : \theta = 1$ (everybody has the flu). We will code by P a person who is positive for flu and by N a person who is negative. Table 7.1 displays the relative strength of evidence for the three hypotheses depending on the outcome of an experiment that samples one or two or three individuals from the population.

```
Warning: `data_frame()` is deprecated, use `tibble()`.
This warning is displayed once per session.
```

Let us go systematically through the table and the relative strength of evidence in favor or against the three distinct hypotheses. If only one person is selected from the population and that person has the flu, then the likelihood $P(X = P|H_1) = 0$ because the H_1 hypothesis is that nobody has the flu. Since the first person who was selected has the flu, the likelihood for this hypothesis ($\theta = 0$) is 0. In contrast, $P(X = P|H_3) = 1$ because the H_3 hypothesis is that everybody has the flu ($\theta = 1$) and we observed one case of flu from the first observation. The likelihood $P(X = P|H_2) = 0.5$ because the hypothesis H_2 is that half of the population has the flu ($\theta = 0.5$). Therefore the relative strength of evidence for hypothesis H_1 versus H_2 is $0/0.5 = 0$. Similarly, the relative strength of evidence for hypothesis H_3 versus H_2 is $1/0.5 = 2$, indicating that after seeing one case of the flu the hypothesis that everybody has the flu is twice as likely as the hypothesis that half of the population has it. If, instead of a positive first case one observes a negative first case, then the $P(X = N|H_1) = 1$, $P(X = N|H_2) = 0.5$, and $P(X = N|H_3) = 0$. Therefore the relative strength of evidence for hypothesis H_1 versus H_2 is $1/0.5 = 2$ and the relative strength

of evidence for hypothesis H_3 versus H_2 is $0/0.5 = 0$.

Consider now the case when we sample two subjects instead of one. We can have only four possible outcomes: PP, PN, NP, and NN. Here PN means that the first person selected had the flu and the second did not. Consider the first case in the table, PP, when both individuals selected have the flu. In this case the likelihood $P(X = PP|H_1) = 0$ because the H_1 hypothesis is that nobody has the flu. In contrast, $P(X = P|H_3) = 1$ because the H_3 hypothesis is that everybody has the flu and we observed two cases of the flu from the first two observations. The likelihood $P(X = PP|H_2) = 0.25$ because the hypothesis H_2 is that half of the population has the flu ($\theta = 0.5$) and $P(X = PP|\theta) = \theta^2$. Therefore the relative strength of evidence for hypothesis H_1 versus H_2 is $0/0.5 = 0$. The relative strength of evidence for hypothesis H_3 versus H_2 is $1/0.25 = 4$, indicating that after seeing two cases of the flu the hypothesis that everybody has the flu is four times as likely as the hypothesis that half of the population has it. If the outcome is either NP or PN (one person has the flu and one does not), then the probability of seeing these data under either hypothesis H_1 or hypothesis H_3 is 0, while under hypothesis H_2 it is 0.25. Indeed, $P(X = NP|\theta) = P(X = PN|\theta) = \theta(1 - \theta)$ and, under H_2, $\theta = 1 - \theta = 0.5$. Therefore, in both of these cases the relative strength of evidence for either hypothesis H_1 or H_3 versus H_2 is zero. However, if we observe two individuals who do not have the flu, then $P(X = NN|H_1) = 1$, $P(X = NN|H_2) = 0.25$, and $P(X = NN) = 0$ indicating that the hypothesis H_1 is four times more likely than hypothesis H_2, while the strength of evidence in favor of hypothesis H_3 relative to hypothesis H_2 is zero.

The case when we sample three individuals from the population follows the same logic. If all of them have the flu (PPP), then the strength of evidence for hypothesis H_3 is eight times the strength of evidence in favor of hypothesis H_2. This happens because $P(X = PPP|H_2) = \theta^3$ and, under H_2, $\theta = 0.5$. Moreover the strength of evidence in favor of H_1 versus H_2 is zero, because under H_1 one should not observe any case of the flu. If among the three individuals at least one has the flu and at least one does not have the flu (cases PPN, PNP, NPP, PNN, NPN, NNP) then the probability of seeing these data under either hypothesis H_1 or H_3 is 0 because neither of them allows for such outcomes. The last case, NNN, when we observe three individuals who do not have the flu the interpretation of the table should be self-explanatory.

7.3.1 Likelihood ratio benchmarks

Using this example as a guide, researchers tend to think of a likelihood ratio of 8 as being moderate evidence, of 16 as being moderately strong evidence, and of 32 as being strong evidence. Indeed, 32 would correspond to how much more likely it is in a population to have everybody infected than have half the population infected with the flu if the first 5 subjects drawn at random from the population all have the flu.

7.4 Likelihood for multiple parameters

So far, we have focused on the case when θ is a scalar, though many distributions depend on multiple parameters (e.g., Normal, Gamma, Beta, t). Definitions remain the same for the likelihood and MLE. Indeed, the likelihood is

$$\mathcal{L}(\theta|\mathbf{x}) = \prod_{i=1}^{n} f(x_i, \theta) \ ,$$

while the MLE is

$$\widehat{\theta}_{\text{ML}} = \text{MLE} = \text{argmax}_\theta \mathcal{L}(\theta|\mathbf{x}) \ .$$

The only difference is that the likelihood function depends on a vector of parameters; visualizing it can be much harder, and obtaining the MLEs may require more difficult numerical operations.

7.4.1 Example: the Normal distribution

A random variable is said to follow a Normal or Gaussian distribution with mean μ and variance σ^2 if the associated density is

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \ ,$$

which depends on two parameters (μ, σ^2). If X is a random variable with this density, then $E[X] = \mu$ and $\text{Var}(X) = \sigma^2$ and we write $X \sim \text{N}(\mu, \sigma^2)$. When $\mu = 0$ and $\sigma = 1$ the resulting distribution is called the standard normal distribution. The standard normal pdf is labeled $\phi(\cdot)$ and the standard normal cdf is labeled $\Phi(\cdot)$. By convention, standard normal random variables are often labeled Z, while standardizing a random variable is often referred to as the Z-score transform. More precisely, for any random variable W the Z-score transform is

$$Z_W = \frac{W - \mu_W}{\sigma_W} \ ,$$

where $\mu_W = E(W)$ and $\sigma_W^2 = \text{Var}(W)$. It is easy to prove that $E(Z_W) = 0$ and $\text{Var}(Z_W) = 1$.

7.4.1.1 Properties of the normal distribution

If $X \sim \text{N}(\mu, \sigma^2)$, then $Z = \frac{X-\mu}{\sigma}$ is a standard normal random variable. If Z is standard normal

$$X = \mu + \sigma Z \sim \text{N}(\mu, \sigma^2)$$

and has the pdf equal to

$$\phi\{(x - \mu)/\sigma\}/\sigma \ .$$

We now show that if $Z \sim N(0,1)$ then $X = \mu + \sigma Z \sim N(\mu, \sigma^2)$. To prove this, calculate the cdf of X (denoted as F_X) as a cdf of Z (denoted as $\Phi(\cdot)$) and take the derivatives to obtain the pdf of X, $f_X(\cdot)$, as a function of the pdf of Z, $\phi(\cdot)$. More precisely, for every t

$$F_X(t) = P(X \leq t) = P(\mu + \sigma Z \leq t) = P(Z \leq \frac{t-\mu}{\sigma}) = F_Z\left(\frac{t-\mu}{\sigma}\right).$$

Taking the derivatives with respect to t and using the chain rule we obtain

$$F'_X(t) = \frac{1}{\sigma} F'_Z\left(\frac{t-\mu}{\sigma}\right) \implies f_X(t) = \frac{1}{\sigma} f_Z\left(\frac{t-\mu}{\sigma}\right).$$

By simply substituting this into the pdf of Z we obtain the result. This approach is general if we are interested in obtaining the pdf of a transformation of a random variable with known pdf. As an exercise, try to obtain the pdf of Z^2 if $Z \sim N(0,1)$ (this is called the χ^2 distribution with one degree of freedom and will be used later on in the book). Also, if $U \sim \text{Uniform}(0,1)$ calculate the pdf of $1 + 2U^3$.

We show now that if $Z \sim N(0,1)$, then $E[Z] = 0$ and $\text{Var}[Z] = 1$. Recall that

$$E(Z) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} z e^{-\frac{z^2}{2}} dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \frac{\partial}{\partial z}\{-e^{\frac{-z^2}{2}}\} dz = -\frac{1}{\sqrt{2\pi}} e^{\frac{-z^2}{2}} \Big|_{-\infty}^{\infty} = 0 .$$

Therefore, $\text{Var(Z)} = E(Z^2) - \{E(Z)\}^2 = E(Z^2)$ because we just proved that $E(Z) = 0$. Now we calculate

$$\begin{aligned} E(Z^2) &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} z^2 e^{-\frac{z^2}{2}} dx \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} z \times \frac{\partial}{\partial z}\{-e^{\frac{-z^2}{2}}\} dz \\ &= -\frac{1}{\sqrt{2\pi}} z e^{\frac{-z^2}{2}} \Big|_{\infty}^{\infty} + \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dx . \end{aligned}$$

The first term in the last equality is zero because the limits at infinity are zero, while the last term is equal to 1 because it is the integral of the $N(0,1)$ distribution. Therefore, $E(Z^2) = 1$. Let $X = \mu + \sigma Z$ and using the rules of expectation we obtain

$$E(X) = E(\mu + \sigma Z) = \mu + \sigma E(Z) = \mu ,$$

and

$$\text{Var}(X) = \text{Var}(\mu + \sigma Z) = \sigma^2 \text{Var}(Z) = \sigma^2 .$$

The normal distribution plot can be easily obtained as follows

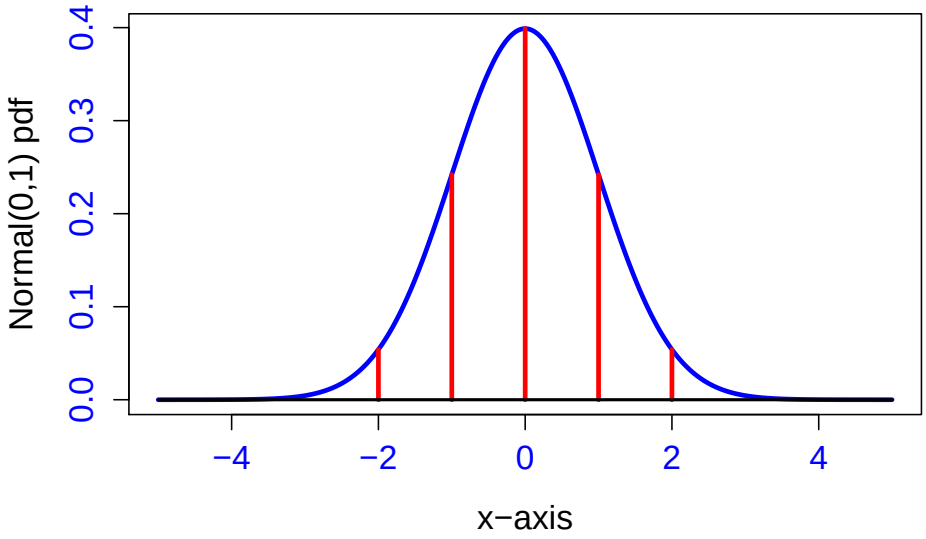


Figure 7.6: Normal pdf with mean zero and variance one together with thresholds at 1 and 2 standard deviations away from the mean (red horizontal lines).

```
x=seq(-5,5,by=0.01)
y=dnorm(x)
plot(x,y,type="l",col="blue",lwd=3,
     xlab="x-axis",ylab="Normal(0,1) pdf",cex.lab=1.3,
     cex.axis=1.3,col.axis="blue")
lines(c(-2,-2),c(0,dnorm(-2)),col="red",lwd=3)
lines(c(-1,-1),c(0,dnorm(-1)),col="red",lwd=3)
lines(c(0,0),c(0,dnorm(0)),col="red",lwd=3)
lines(c(1,1),c(0,dnorm(1)),col="red",lwd=3)
lines(c(2,2),c(0,dnorm(2)),col="red",lwd=3)
lines(c(-5,5),c(0,0),lwd=2)
```

The probability of being less than k-sigma away from the mean for a $N(\mu, \sigma^2)$ random variable is equal to

$$P(\mu - k\sigma \leq X \leq \mu + k\sigma) = P(-k \leq \frac{X-\mu}{\sigma} \leq k) = P(-k \leq Z \leq k),$$

which is equal to the probability that a standard Normal random variable, Z, is less than k away from 0. Below we provide a few such calculations (note that we do not need to keep books with Normal distribution tables)

```
#Probability of at least 1 standard deviation away from the mean
round(pnorm(1)-pnorm(-1),digits=3)
```

```
[1] 0.683
```

```
#Probability of at least 2 standard deviations away from the mean
round(pnorm(2)-pnorm(-2),digits=3)
```

```
[1] 0.954
```

```
#Probability of at least 3 standard deviations away from the mean
round(pnorm(3)-pnorm(-3),digits=3)
```

```
[1] 0.997
```

Figure 7.6 displays the pdf of the $N(0,1)$ distribution together with thresholds at 1 and 2 standard deviations away from the mean (red horizontal lines). The probabilities we just calculated can be visualized in Figure 7.6 as the areas under the curve contained between the corresponding red lines. For example, the area under the curve between the left-most and right-most red vertical line is 0.954, whereas the area between the second line from the left and the second line from the right is 0.683. Therefore, approximately 68%, 95% and 99% of the normal density lies within 1, 2, and 3 standard deviations from the mean, respectively. The 10^{th}, 5^{th}, 2.5^{th}, and 1^{st} percentiles of the standard normal distribution are -1.28, -1.645, -1.96, and -2.33. By symmetry, the 90^{th}, 95^{th}, 97.5^{th}, and 99^{th} percentiles of the standard normal distribution are 1.28, 1.645, 1.96, and 2.33, respectively.

Suppose that we are interested in what is the 95^{th} percentile of a $N(\mu, \sigma^2)$ distribution. We want the point x_0 so that $P(X \leq x_0) = .95$.

$$P(X \leq x_0) = P\left(\frac{X-\mu}{\sigma} \leq \frac{x_0-\mu}{\sigma}\right) = P\left(Z \leq \frac{x_0-\mu}{\sigma}\right) = .95\ .$$

Therefore,

$$\frac{x_0-\mu}{\sigma} = 1.645$$

or $x_0 = \mu + \sigma 1.645$. In general $x_0 = \mu + \sigma z_0$ where z_0 is the corresponding standard Normal quantile. It is nice that the quantile of the $N(\mu, \sigma^2)$ distribution can be obtained directly from the quantiles of a $N(0,1)$ distribution using the transformation $x_0 = \mu + \sigma z_0$.

We would like to know what is the probability that a $\mathrm{N}(\mu, \sigma^2)$ random variable is 2 standard deviations above the mean. Therefore, we have to calculate

$$P(X > \mu + 2\sigma) = P\left(\frac{X-\mu}{\sigma} > \frac{\mu+2\sigma-\mu}{\sigma}\right) = P(Z \geq 2) \approx 2.5\%\ .$$

We summarize here some of the properties of Normal random variables. The Normal distribution is symmetric and peaked about its mean, and its mean, median, and mode are all equal. A constant times a normally distributed random variable is also normally distributed. Sums of normally distributed random variables are normally distributed even if the variables are dependent. Sample means of normally distributed random variables are again normally distributed.

The square of a standard normal random variable follows what is called **chi-squared** (denoted as χ^2) distribution with one degree of freedom. The exponent of a normally distributed random variable follows what is called the **log-normal** distribution. As we will see later, many random variables, properly normalized, have an approximate normal distribution.

7.4.2 Likelihood of the Normal distribution

If X_i are iid $N(\mu, \sigma^2)$, then the likelihood is a function of the two parameters (μ, σ^2). The joint likelihood is proportional to

$$\mathcal{L}(\mu, \sigma^2|\mathbf{x}) = \exp\left\{-\frac{\sum_{i=1}^n (x_i - \mu)^2}{2\sigma^2} - \frac{n}{2}\log(\sigma^2)\right\} ,$$

where we basically ignored the component $(2\pi\sigma^2)^{-n/2}$, which contains no information about the parameters. The minus twice log-Likelihood has the form

$$-2l(\mu, \sigma^2|\mathbf{x}) = \frac{\sum_{i=1}^n (x_i - \mu)^2}{\sigma^2} + n\log(\sigma^2) ,$$

which is a much nicer form.

7.4.2.1 Maximum likelihood estimation for the Normal distribution

We would like to maximize the likelihood with respect to both μ and σ^2, which is equivalent to minimizing the function $-2l(\mu, \sigma^2|\mathbf{x})$. We first show that, for any value of σ^2, $\widehat{\mu}_n = \frac{1}{n}\sum_{i=1}^n x_i = \overline{x}_n$ minimizes the function $-2l(\mu, \sigma^2|\mathbf{x})$. Note that

$$\frac{\partial}{\partial\mu}\left\{-2l(\mu, \sigma^2|\mathbf{x})\right\} = -\frac{2}{\sigma^2}\sum_{i=1}^n (x_i - \mu) ,$$

and

$$\frac{\partial^2}{\partial\mu^2}\left\{-2l(\mu, \sigma^2|\mathbf{x})\right\} = \frac{2n}{\sigma^2} > 0 ,$$

indicating that the function is strictly convex. This shows that for every σ^2 and μ

$$-2l(\mu, \sigma^2|\mathbf{x}) \geq -2l(\overline{x}_n, \sigma^2|\mathbf{x}) .$$

Now, consider the following function of σ^2

$$-2l(\widehat{\mu}_n, \sigma^2|\mathbf{x}) = \frac{\sum_{i=1}^n (x_i - \overline{x}_n)^2}{\sigma^2} + n\log(\sigma^2) .$$

The first derivative of this function relative to σ^2 is

$$\frac{\partial}{\partial\sigma^2}\{-2l(\widehat{\mu}_n, \sigma^2|\mathbf{x})\} = -\frac{\sum_{i=1}^n (x_i - \overline{x}_n)^2}{\sigma^4} + \frac{n}{\sigma^2} = \frac{n}{\sigma^4}(\sigma^2 - s_n^2) ,$$

where

$$s_n^2 = \frac{\sum_{i=1}^n (x_i - \overline{x}_n)}{n} .$$

This indicates that $\frac{\partial}{\partial \sigma^2}\{-2l(\widehat{\mu}_n, \sigma^2|\mathbf{x})\} < 0$ if $\sigma^2 < s_n^2$ and $\frac{\partial}{\partial \sigma^2}\{-2l(\widehat{\mu}_n, \sigma^2|\mathbf{x})\} > 0$ if $\sigma^2 > s_n^2$. This indicates that the function is decreasing between $(0, s_n^2)$ and increasing between (s_n^2, ∞) and because the function is continuous its minimum is attained at $\widehat{\sigma}_n^2 = s_n^2$. Therefore, for every μ and σ^2 we have

$$-2l(\mu, \sigma^2|\mathbf{x}) \geq -2l(\overline{x}_n, s_n^2|\mathbf{x}) ,$$

which indicates that $(\overline{X}_n, S_n^2)$ is the MLE for (μ, σ^2).

The proof for this result is often presented incorrectly by first taking the derivative with respect to σ^2, calculating the critical point and then making an argument using the second derivative. The argument does not work because the second derivative is not positive everywhere. Indeed, the function has an inflection point (changes from convex to concave) at $2s_n^2$.

7.4.2.2 Plotting the Normal likelihood

We plot the bivariate Normal likelihoods first to get a sense of what they look like. We will look at the likelihood as a function of (μ, σ), but one could similarly do the same thing for (μ, σ^2). Figure 7.7 displays the normalized bivariate likelihood for the parameters (μ, σ) of the $N(\mu, \sigma^2)$ distribution based on 10 independent random samples drawn from a $N(3, 4)$ distribution (note that the parameterization of the Normal in `R` is done relative to the standard deviation σ and not the variance σ^2.)

```
#Plot the bivariate likelihood for a normal
#Set the sample size for the data
n=10

#Simulate n=10 independent N(3,4) rvs (these are the data)
x=rnorm(n,3,2)

#Set up a grid for the mean and standard deviation
mu=seq(0,5,length=100)
sigma=seq(0.1,5,length=100)

#Set the 100 by 100 matrix that will store the likelihood
like=matrix(rep(0,10000),ncol=100)

#Building a 3D plot of a distribution
#Calculate the likelihood
for (i in 1:100) {
  for (j in 1:100) {
```

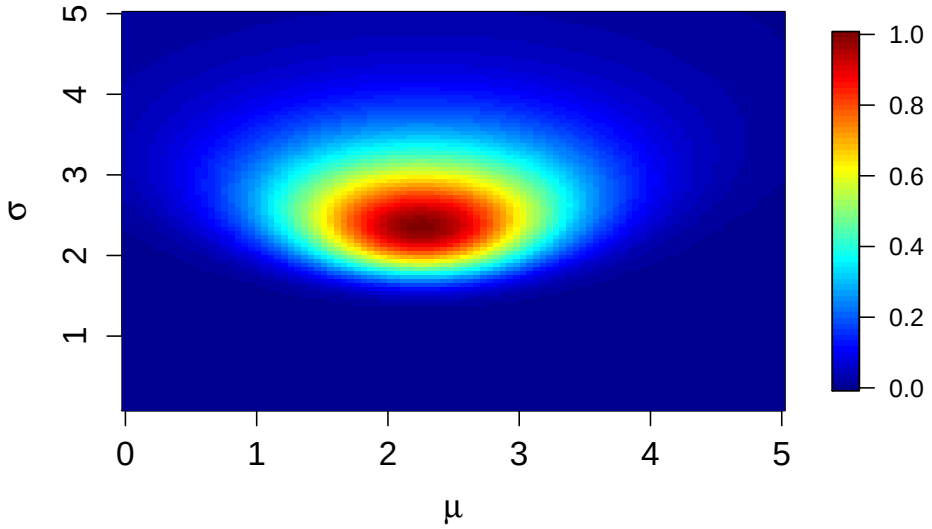


Figure 7.7: Bivariate likelihood for the parameters (μ, σ) of the $N(\mu, \sigma^2)$ distribution based on 10 independent random samples drawn from a $N(3, 4)$ theoretical distribution.

```
    like[i,j] = exp(-sum((x-mu[i])^2)/(2*sigma[j]^2) - n*log(sigma[j]))
  }
}

#Normalize the likelihood
like=like/max(like)

library(fields)
image.plot(mu,sigma,like,xlab=expression(mu),
           ylab=expression(sigma),cex.lab=1.3,cex.axis=1.3)
```

```
round(mean(x),digits=2)
```

```
[1] 2.26
```

```
round(sqrt(9*var(x)/10),digits=2)
```

```
[1] 2.36
```

The MLE for (μ, σ) in this sample is $(2.26, 2.36)$ and this can be seen as the more intense shades of red around the MLE. The likelihood is not symmetric around the MLE, especially in the y-axis direction (σ). The likelihood will vary with the sample and the sample size. To get a better visualization of these variations Figure 7.8 displays the bivariate likelihood on a 2D grid of values for (μ, σ) for four iid samples of size 10 from the $N(3, 4)$ distribution.

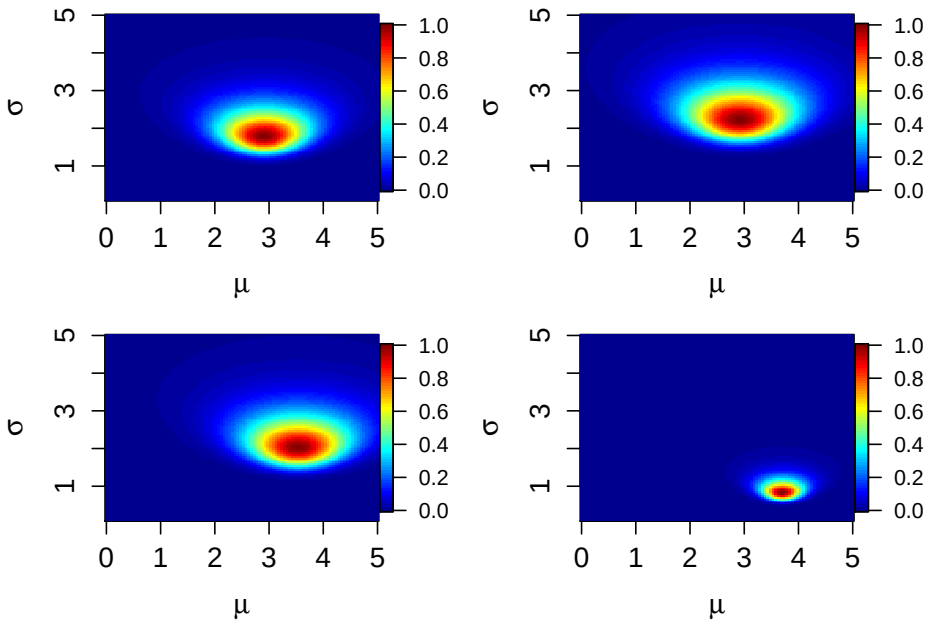


Figure 7.8: Bivariate likelihood for the parameters (μ, σ) of the $N(\mu, \sigma^2)$ distribution for four different independent samples of size 10 drawn from a $N(3, 4)$ theoretical distribution.

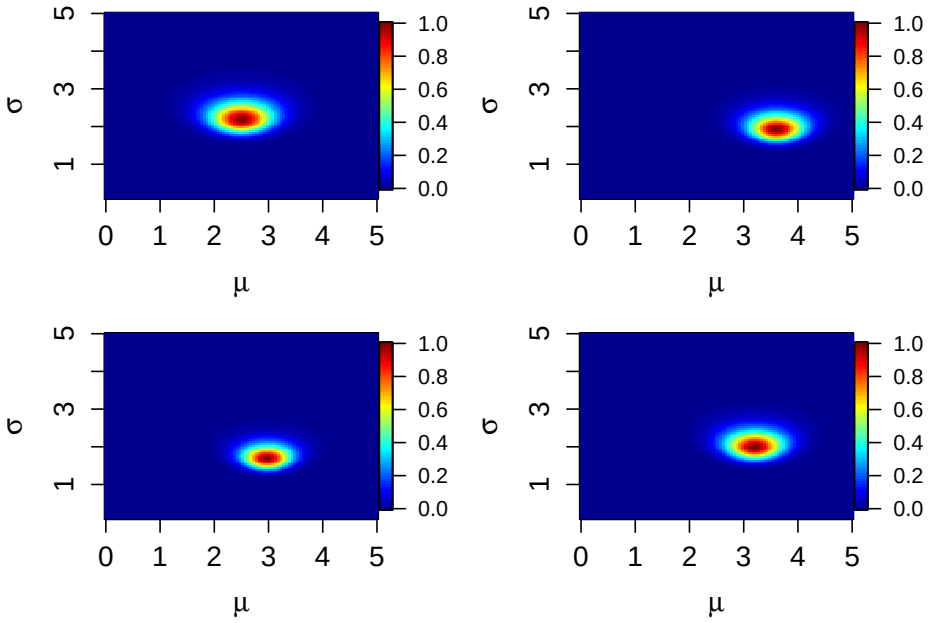


Figure 7.9: Bivariate likelihood for the parameters (μ, σ) of the $N(\mu, \sigma^2)$ distribution for four different independent samples of size 30 drawn from a $N(3, 4)$ theoretical distribution.

Even though the four likelihoods correspond to 10 independent samples from the same theoretical distribution, $N(3, 4)$, the location of their maxima and the spread of the likelihood around their maxima changes quite substantially. This is due to the sampling variability of the MLE as well as to the properties of the likelihood. For example, the right bottom plot has the likelihood much more concentrated around the MLE because the MLE for σ is quite small. To see the effects of sample size on the likelihood, Figure 7.9 displays the likelihood for a sample size equal to $n = 30$.

When comparing Figures 7.9 and 7.8 we observe that likelihoods look much more symmetric and concentrated around their MLEs, while the location of the maximum and the spread of the likelihood around it are more consistent and predictable from one plot to another. We repeat the process one more time, increase the sample size to $n = 100$, and display the results in Figure 7.10. We notice that the changes in the likelihood continue in the same direction, the likelihood becomes more symmetric (and bell-shaped) around the MLEs, the concentration of likelihood around the MLEs continues to increase, while the location of the maximum and the spread of the likelihood around it are more consistent and predictable from one plot to another.

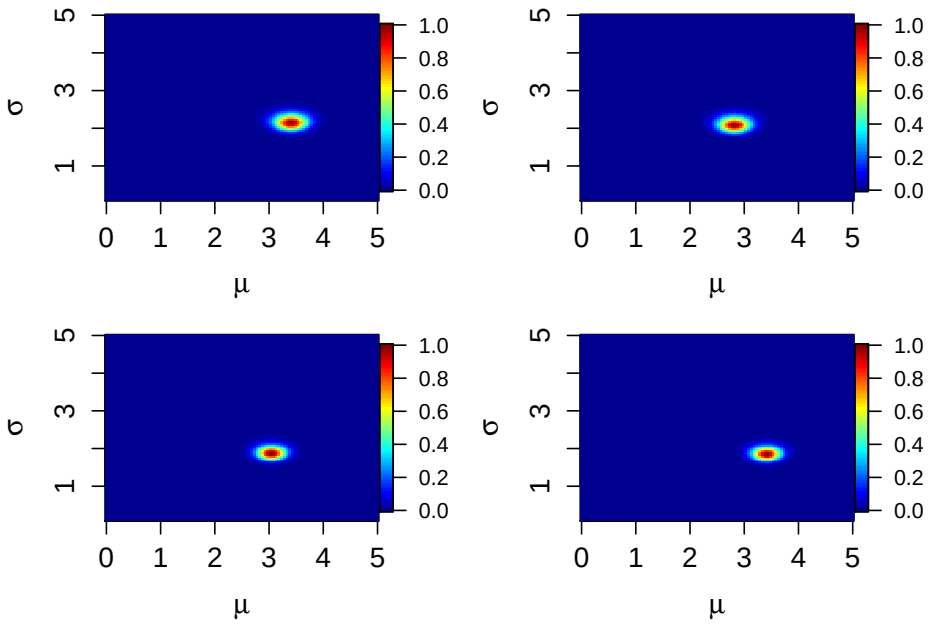


Figure 7.10: Bivariate likelihood for the parameters (μ, σ) of the $N(\mu, \sigma^2)$ distribution for four different independent samples of size 30 drawn from a $N(3, 4)$ theoretical distribution.

7.5 Profile likelihood

Consider the case when $X_1, \ldots, X_n$ are iid with pdf $f(x, \theta)$, where θ is a multivariate vector of parameters that can be partitioned as $\theta = (\mu, \eta)$. Typically, μ is a scalar parameter of interest and η is a scalar of vector of nuisance parameters. The idea of the profile likelihood is that we would like to isolate and present the information available in the data about the parameter μ of interest.

The procedure is quite simple and requires the following steps: (1) for each value of μ maximize the likelihood with respect to the rest of the parameters η; (2) denote by $\widehat{\eta}(\mu, \mathbf{x})$ the maximum likelihood estimator for η; and (3) define the profile likelihood function with respect to μ as

$$\mathcal{PL}(\mu|\mathbf{x}) = \mathcal{L}\{\mu, \widehat{\eta}(\mu, \mathbf{x})|\mathbf{x}\} \ .$$

The profile likelihood is especially useful when we are interested in one parameter and there are many other parameters that would make plotting and interpretation of a multivariate likelihood difficult.

7.5.1 Profile likelihood of the Normal

Let $X_1 = x_1, \ldots, X_n = x_n$ be iid $N(\mu, \sigma^2)$ with the likelihood

$$\mathcal{L}(\mu, \sigma^2|\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{\sum_i^n (x_i - \mu)^2}{2\sigma^2}\right\} \ .$$

We consider that μ is the parameter of interest. Thus, we fix μ and maximize the log likelihood with respect to σ^2. Using techniques already described in this chapter we obtain that the profile estimator of the variance is

$$\widehat{\sigma}^2(\mu, \mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2 \ .$$

It is easier to work with the profile log likelihood

$$2 \log \mathcal{PL}(\mu|\mathbf{x}) = 2 \log \mathcal{L}\{\mu, \widehat{\sigma}^2(\mu, \mathbf{x})|\mathbf{x}\} \ ,$$

where

$$\begin{aligned} 2 \log \mathcal{L}\{\mu, \widehat{\sigma}^2(\mu, \mathbf{x})|\mathbf{x}\} &= -\frac{\sum_i^n (x_i - \mu)^2}{\widehat{\sigma}^2(\mu, \mathbf{x})} - n \log\{\widehat{\sigma}^2(\mu, \mathbf{x})\} - n \log(2\pi) \\ &= -n \log\{\widehat{\sigma}^2(\mu, \mathbf{x})\} - n - n \log(2\pi) \ . \end{aligned}$$

Here $n + n \log(2\pi)$ is a constant that does not depend on the parameter μ and can be ignored. Therefore minus twice the profile likelihood for μ is

$$-2 \log \mathcal{L}\{\mu, \widehat{\sigma}^2(\mu, \mathbf{x})|\mathbf{x}\} = n \log\{\widehat{\sigma}^2(\mu, \mathbf{x})\} = n \log\left\{\frac{\sum_i^n (x_i - \mu)^2}{n}\right\} \ .$$

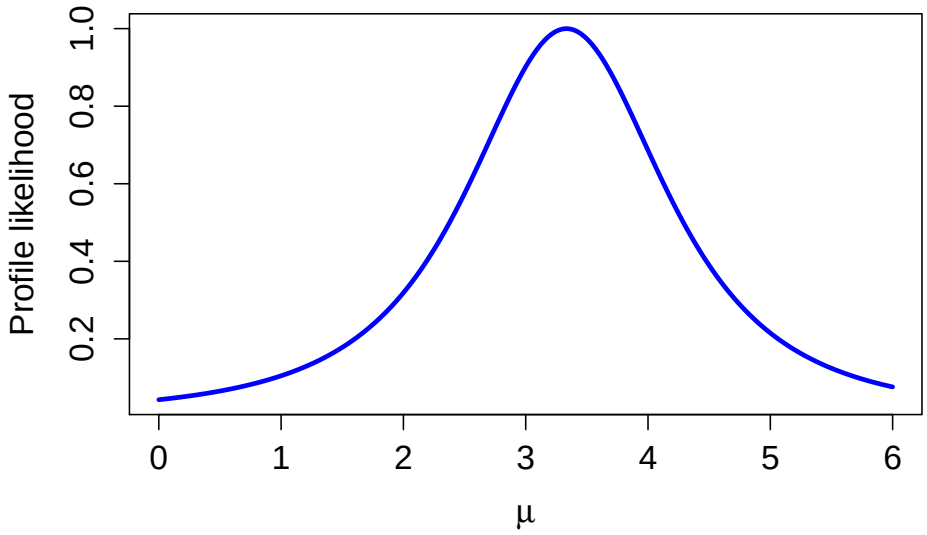


Figure 7.11: Profile likelihood for the mean parameter, μ, of a $N(\mu, \sigma^2)$ distribution when the observations are 5, 2, 3. The profile likelihood is normalized by dividing by the maximum likelihood.

7.5.1.1 Example: profile likelihood of a Normal

Consider the example when we have 3 iid observations, $(5, 2, 3)$, from a $N(\mu, \sigma^2)$. As we derived above,

$$-2\log\mathcal{L}\{\mu, \widehat{\sigma}^2(\mu, \mathbf{x})|\mathbf{x}\} = 3\log\left\{\frac{(5-\mu)^2 + (2-\mu)^2 + (3-\mu)^2}{3}\right\}.$$

```
#Data
bx=c(5,2,3)
#Grid where the profile likelihood is evaluated
mu=seq(0,6,length=201)

#Store -2*log profile likelihood
minus_2_p_loglike=rep(0,201)
for (i in 1:201)
   {minus_2_p_loglike[i]=3*log(sum((bx-mu[i])^2))}

profile_likelihood<-exp(-0.5*minus_2_p_loglike)
profile_likelihood<-profile_likelihood/max(profile_likelihood)
plot(mu,profile_likelihood,type="l",col="blue",lwd=3,
     xlab=expression(mu),ylab="Profile likelihood",
     cex.lab=1.3,cex.axis=1.3)
```

Figure 7.11 displays the profile likelihood for the mean parameter, μ, of a $N(\mu, \sigma^2)$ distribution. This plot is simpler than the bivariate plots shown earlier and focuses on the parameter of interest, μ, with the parameter σ^2 being profiled out.

7.6 Problems

Problem 1. Imagine that a person, say his name is Flip, has an oddly deformed coin and tries the following experiment. Flip flips his coin 10 times, 7 of which are heads. You think maybe Flip's coin is biased towards having a greater probability of yielding a head than 50%.

a. What is the maximum likelihood estimate of p, the true probability of heads associated with this coin?
b. Plot the likelihood associated with this experiment. Renormalize the likelihood so that its maximum is one. Does the likelihood suggest that the coin is fair?
c. What is the probability of seeing 7 or more heads out of 10 coin flips if the coin was fair? Does this probability suggest that the coin is fair? This number is called the p-value.
d. Suppose that Flip told you that he did not fix the number of trials at 10. Instead, he told you that he had flipped the coin until he obtained 3 tails and it happened to take 10 trials to do so. Therefore, the number 10 was random while the number 3 was fixed. The probability mass function for the number of trials, say y, to obtain 3 tails (called the negative binomial distribution) is
$$\left(\begin{array}{c} y-1 \\ 2 \end{array} \right) (1-p)^3 p^{y-3}$$
for $y = 3, 4, 5, 6, \ldots$. What is the maximum likelihood estimate of p now that we've changed the underlying pmf?
e. Plot the likelihood under this new mass function. Renormalize the likelihood so that its maximum is one. Does the likelihood suggest that the coin is fair?
f. Calculate the probability of requiring 10 or more flips to obtain 3 tails if the coin was fair. (Notice that this is the same as the probability of obtaining 7 or more heads to obtain 3 tails). This is the p-value under the new mass function.

(Aside) This problem highlights a distinction between the likelihood and the p-value. The likelihood and the MLE are the same regardless of the experiment. That is to say, the likelihood only seems to care that you saw 10 coin flips, 7 of which were heads. Flip's intention about when he stopped flipping the coin, either at 10 fixed trials or until he obtained 3 tails, is irrelevant as far as the likelihood is concerned. In contrast, the p-value depends on Flip's intentions.

Problem 2. A researcher is studying the number of sexual acts with an infected person until an uninfected person contracts a sexually transmitted disease. She assumes that each encounter is an independent Bernoulli trial with probability p that the subject becomes infected. This leads to the so-called geometric distribution

$$P(\text{Person is infected on contact } x) = p(1-p)^{x-1}$$

for $x = 1, \ldots$.

a. Suppose that one subject's number of encounters until infection is recorded, say x. Symbolically derive the ML estimate of p.
b. Suppose that the subject's value was 2. Plot and interpret the likelihood for p.
c. Suppose that it is often assumed the probability of transmission, p, is .01. The researcher thinks it is perhaps strange to have a subject get infected after only 2 encounters if the probability of transmission is really 1%. According to the geometric mass function, what is the probability of a person getting infected in 2 or fewer encounters if p truly is .01?
d. Suppose that she follows n subjects and records the number of sexual encounters until infection (assume all subjects became infected) $x_1, \ldots, x_n$. Symbolically derive the MLE of p.
e. Suppose that she records values $x_1 = 3$, $x_2 = 5$, $x_3 = 2$. Plot and interpret the likelihood for p.

Problem 3. In a study of aquaporins, six frog eggs received a protein treatment. If the treatment of the protein is effective, the frog eggs would implode. The experiment resulted in five frog eggs imploding. Historically, 10 percent of eggs implode without the treatment. Assuming that the results for each egg are independent and identically distributed:

a. What is the probability of getting five or more eggs imploding in this experiment if the true probability of implosion is 10%? Interpret this number.
b. What is the maximum likelihood estimate for the probability of implosion?
c. Plot and interpret the likelihood for the probability of implosion.

Problem 4. Suppose that IQs in a particular population are normally distributed with a mean of 110 and a standard deviation of 10.

a. What is the probability that a randomly selected person from this population has an IQ between 95 and 115?
b. What is the 65^{th} percentile of this distribution?
c. Suppose that 5 people are sampled from this distribution. What is the probability 4 (80%) or more have IQs above 130?
d. Suppose that 500 people are sampled from this distribution. What is the probability 400 (80%) or more have IQs above 130?
e. Consider the average of 100 people drawn from this distribution. What is the probability that this mean is larger than 112.5?

Problem 5. Suppose that 400 observations are drawn at random from a distribution with mean 0 and standard deviation 40.

a. What is the approximate probability of getting a sample mean larger than 3.5?
b. Was normality of the underlying distribution required for this calculation?

Problem 6. Suppose that the Diastolic Blood Pressure (DBP) drawn from a certain population is normally distributed with a mean of 90 $mmHg$ and standard deviation of 5 $mmHg$. Suppose that $1,000$ people are drawn from this population.

a. If you had to guess the number of people having DBPs less than 80 $mmHg$, what would you guess?
b. You draw 25 people from this population. What is the probability that the sample average is larger than 92 $mmHg$?
c. You select five people from this population. What is the probability that four or more of them have a DBP larger than 100 $mmHg$?

Problem 7. Let X_1, X_2 be independent, identically distributed coin flips (taking values 0 = failure or 1 = success) having success probability π. Give and interpret the likelihood ratio comparing the hypothesis that $\pi = .5$ (the coin is fair) versus $\pi = 1$ (the coin always gives successes) when both coin flips result in successes.

Problem 8. Let X be a random variable with a Binomial distribution with success probability p_1 and n_1 trials and Y be an independent random variable with a Binomial distribution with success probability p_2 and n_2 trials. Let $\hat{p}_1 = X/n_1$ and $\hat{p}_2 = Y/n_2$ be the associated sample proportions. What would be an estimate for the standard error for $\hat{p}_1 - \hat{p}_2$?

Problem 9. A sample of 40 United States men contained 25% smokers. Let p be the true prevalence of smoking amongst males in the United States. Write out, draw, and interpret the likelihood for p. Is $p = .35$ or $p = .15$ better supported given the data (why, and by how much)? What value of p is best supported (just give the number, do not derive)?

Problem 10. Consider a sample of n iid draws from an exponential density

$$\frac{1}{\beta}\exp(-x/\beta) \quad \text{for} \quad \beta > 0\,.$$

a. Derive the maximum likelihood estimate for β. b. Suppose that in your experiment, you obtained five observations

$$1.590,\ 0.109,\ 0.155,\ 0.281,\ 0.453$$

c. Plot the likelihood for β. Put in reference lines at 1/8 and 1/16.

Problem 11. Often infection rates per time at risk are modelled as Poisson random variables. Let X be the number of infections and let t be the person

days at risk. Consider the Poisson mass function

$$P(X = x) = (t\lambda)^x \frac{\exp(-t\lambda)}{x!} .$$

The parameter λ is called the population incident rate.

a. Derive the ML estimate for λ.
b. Suppose that five infections are recorded per 1000 person-days at risk. Plot the likelihood.
c. Suppose that five independent hospitals are monitored and that the infection rate (λ) is assumed to be the same at all five. Let X_i, t_i be the count of the number of infections and person days at risk for hospital i. Derive the ML estimate of λ.

Problem 12. Consider n iid draws from a gamma density where α is known

$$\frac{1}{\Gamma(\alpha)\beta^\alpha} x^{\alpha-1} \exp(-x/\beta) \quad \text{for} \quad \beta > 0, x > 0, \alpha > 0 .$$

a. Derive the ML estimate of β.
b. Suppose that $n = 5$ observations were obtained: 0.015, 0.962, 0.613, 0.061, 0.617. Draw the likelihood plot for β (assume that $\alpha = 1$).

Problem 13. Let $Y_1, \ldots, Y_N$ be iid random variables from a Lognormal distribution with parameters μ and σ^2. Note $Y \sim \text{Lognormal}(\mu, \sigma^2)$ if and only if $\log(Y) \sim N(\mu, \sigma^2)$. The log-normal density is given by

$$(2\pi\sigma^2)^{-1/2} \exp[-\{\log(y) - \mu\}^2/2\sigma^2]/y \quad \text{for} \quad y > 0 .$$

a. Show that the ML estimate of μ is $\hat{\mu} = \frac{1}{N}\sum_{i=1}^N \log(Y_i)$. (The mean of the log of the observations. This is called the "geometric mean".)
b. Show that the ML estimate of σ^2 is then the biased variance estimate based on the log observation

$$\frac{1}{N}\sum_{i=1}^N \{\log(y_i) - \hat{\mu}\}^2 .$$

Problem 14. Consider X_i for $i = 1, \ldots, n$ iid $N(\mu, \sigma^2)$ random variables.

a. Derive the profile likelihood function for σ^2.
b. If $n = 3$ and the three realizations are 5, 2, and 3 plot the normalized profile likelihood function for σ^2.

Problem 15. A study of blood alcohol levels (mg/100 ml) at post mortem examination of traffic accident victims involved taking one blood sample from the leg, A, and another from the heart, B. The results were:

Case	A	B	Case	A	B
1	44	44	11	265	277
2	265	269	12	27	39
3	250	256	13	68	84
4	153	154	14	230	228
5	88	83	15	180	187
6	180	185	16	149	155
7	35	36	17	286	290
8	494	502	18	72	80
9	249	249	19	39	50
10	204	208	20	272	271

a. Create a graphical display comparing a case's blood alcohol level in the heart to that in the leg. Comment on any interesting patterns from your graph.
b. Create a graphical display of the distribution of the difference in blood alcohol levels between the heart and the leg.
c. Do these results indicate that in general blood alcohol levels may differ between samples taken from the leg and the heart? Obtain confidence intervals using a bootstrap of subjects.
d. Create a profile likelihood for the true mean difference and interpret.
e. Create a likelihood for the variance of the difference in alcohol levels and interpret.

Chapter 8

Data visualization

This chapter covers the following topics

- Histograms
- Kernel density estimates (KDEs)
- Scatterplots
- Dotplots
- Boxplots
- Bar plots and stacked bar plots
- QQ-plots
- Heat maps

Visualization of data is a very powerful way to communicate information and much work has been dedicated in `R` to improving data visualization. This is an extremely dynamic area with many packages that create a diverse and thriving environment. Therefore, this will be one of the chapters that will probably be most often updated to keep up with the latest developments. Indeed, there are now monographs dedicated to `R` visualization; see, for example, (Chang 2013; Hilfiger 2016; Rahlf 2017; Wickham 2016). In our own book we extensively use data visualization and various types of plots are introduced progressively. We have found this to be the best way for teaching visualization because the context of the problem, knowledge of methods used to manipulate and analyze the data, and final choice of plot interact naturally when working with data. Below we go through a list of necessary, but far from exhaustive, graphics that are very useful in our data practice. We will use several different datasets, including the SHHS and datasets that already exist in `R`.

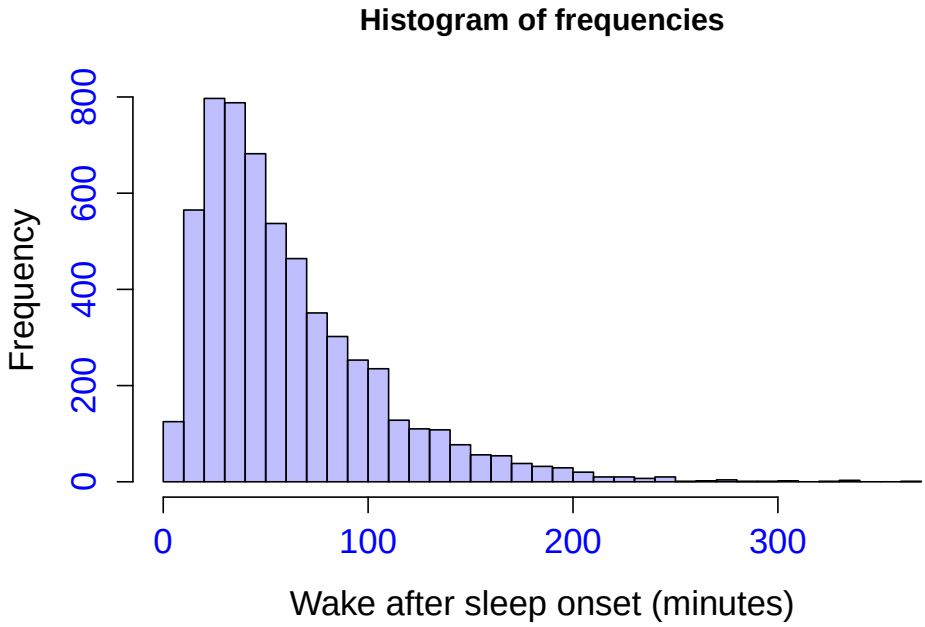


Figure 8.1: Histogram of Wake After Sleep Onset (WASO) in the SHHS. The y-axis is expressed in number of observations per bin.

8.1 Histograms

Histograms display a sample estimate of the density or mass function by plotting a bar graph of the frequency or proportion of times that a variable takes specific values, or a range of values for continuous data, within a sample. Throughout the book we have seen many histograms and here we focus on the variable `WASO` (Wake After Sleep Onset) that has been connected to sleep quality. Figure 8.1 displays the histogram of `WASO`, a variable expressed in minutes, and indicates that most individuals have less than 100 minutes of wake after sleep onset with a mode (point of maximum density) close to 20-30 minutes. To construct a histogram the range of the variable is divided into bins, which typically are of equal length, but do not have to be. In every bin one simply counts how many individuals have values of the variable in the interval defined by the bin. Figure 8.1 provides the visual representation of this histogram, where the y-axis represents the number of subjects in each bin. To indicate this scale, the y-axis is automatically labeled "Frequency" in `R`.

```
hist(WASO,col=rgb(0,0,1,1/4),breaks=30,
     xlab="Wake after sleep onset (minutes)",
     main="Histogram of frequencies",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

For example, there are 797 individuals who have a WASO between $(20, 30]$ minutes and there are 537 individuals who have a WASO score between $(50, 60]$ minutes. When the histogram plots the intervals versus the counts in each interval we say that this is a frequency histogram. One drawback of these plots is that they depend on the total number of individuals in the sample, making frequency histograms hard to compare across populations. Thus, sometimes it make sense to transform the numbers in each bin to the proportion of individuals from the population who fall into that bin. Technically, the number of subjects in each bin is divided by the total number of subjects in the sample. In this case, for the bin $(20, 30]$ the associated fraction of subjects is $797/5804 \approx 0.137$ and for the bin $(50, 60]$ the associated fraction is $537/5804 \approx 0.093$. However, these are not the numbers that are being plotted. Indeed, the area of each bin needs to be equal to the fraction because we want the total area to be equal to 1. Recall that the area of each bin is equal to the length of the bin on the x-axis times its length on the y-axis. In our case the length of each bin is 10 and to obtain the density estimator we need to divide the frequencies we obtained by 10, the length of the bin on the x axis. That makes the height of the bin $(20, 30]$ equal to 0.0137 and the height of the bin $(50, 60]$ equal to 0.0093. Figure 8.2 displays the histogram of WASO on this transformed scale, which in `R` can be obtained directly by using the option `probability=TRUE`. Note that there is no change in the shape of the histogram, but the y-axis units change substantially from the number of individuals in each bin to the number of individuals in each bin divided by the total number of individuals in the sample times the length of the bin on the x-axis.

```
hh<-hist(WASO,probability=TRUE,col=rgb(0,0,1,1/4),breaks=30,
     xlab="Wake after sleep onset (minutes)",
     main="Histogram of frequencies",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

The function `hist` produces by default a histogram of the data, but the numbers produced by the function can easily be accessed, as well. For example,

```
names(hh)
```

```
[1] "breaks"   "counts"   "density"  "mids"     "xname"    "equidist"
```

The first 7 breaks, counts, and density numbers can be accessed as

```
hh$breaks[1:7]
```

```
[1]  0 10 20 30 40 50 60
```

```
hh$counts[1:7]
```

```
[1] 125 565 797 788 682 537 464
```

```
round(hh$density[1:7],digits=4)
```

```
[1] 0.0022 0.0097 0.0137 0.0136 0.0118 0.0093 0.0080
```

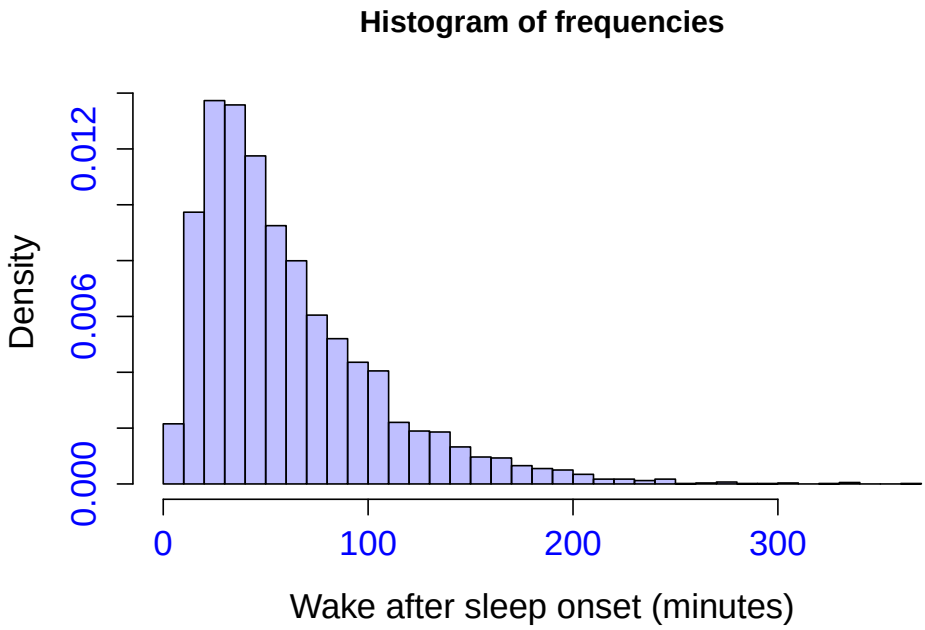


Figure 8.2: Histogram of Wake After Sleep Onset (WASO) in the SHHS. The y-axis is expressed in proportion of subjects in each bin divided by the total number of subjects and the length of the bin. The total area of the histogram (blue shaded area) is equal to 1.

The `hist` function is quite flexible and allows us to change the number of bins using `breaks`. In our case we used `breaks=30` to indicate that we want 30, but it can also take a vector of breakpoints for added flexibility. Here we provide an example of how to build a histogram, but the process is customizable and one can add lines, points, and legend, and change colors. For more information just type `?hist` in `R` and check out the many innovations that scientists show online.

8.2 Kernel density estimates (KDEs)

Kernel density estimates (KDEs) are nonparametric estimates of the pdf of a variable. They are designed to target the same estimand as histograms, but try to address some of the visual problems associated with the histograms. In particular KDEs smooth the data instead of binning it into groups. At every point in the domain of the variable a weighted average of the density of points is calculated. Figure 8.3 displays one KDE, while the code below indicates how the plot is constructed `R`

```
d<-density(WASO,bw="sj")
plot(d,col="blue",lwd=3,
     xlab="Wake after sleep onset (minutes)",
     main="Kernel density estimate",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
```

Here the option `bw="sj"` indicates that we asked for the Sheather & Jones method (Sheather and Jones 1991) for selecting the Kernel bandwidth (`bw` is shorthand for bandwidth). By comparing the outline of the KDE estimator with the histogram estimator that uses `probability=TRUE` we see that the results are comparable, with the KDE being smoother. One small problem is that the KDE assigns some small probability to values below 0 and we know that this is not possible, as `WASO` is a positive variable. Correcting this is not immediately clear, with one solution being to simply cut all values below zero and re-assign the probability assigned to negative values to positive ones. This type of edge effect is characteristic of KDEs. To better understand what KDEs are we will look at the definition. If $x_1, \ldots, x_n$ are realizations of independent random variables $X_1, \ldots, X_n$ then the KDE is

$$f(x; K, h) = \frac{1}{nh} \sum_{k=1}^{n} K\left(\frac{x - x_i}{h}\right) ,$$

where $K(\cdot)$ is a function, called Kernel, and h is the bandwidth. It is easy to show that if $K(\cdot)$ is a pdf then $f(x; K, h)$ is a pdf for any $K(\cdot)$ and $h > 0$. In some situations the function $K(\cdot)$ is not required to be positive, but almost

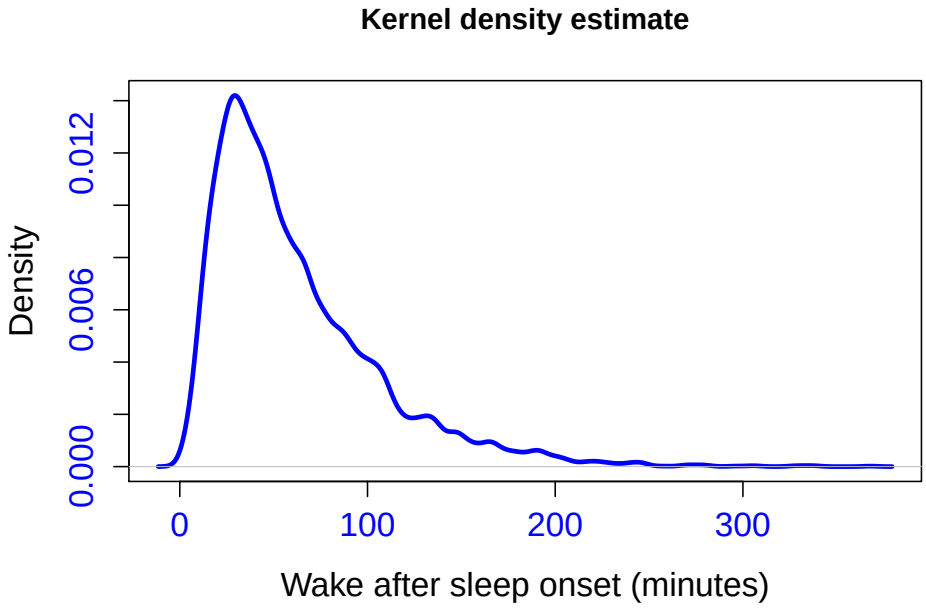


Figure 8.3: Kernel density estimator of the pdf of WASO in the SHHS. The y-axis is on the same scale as the histogram on the probability scale.

always it is required to integrate to 1, that is

$$\int_{-\infty}^{\infty} K(x)dx = 1 \, .$$

The standard choice of Kernel is the pdf of $N(0,1)$ distribution

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \, ,$$

but other Kernels have been used historically. To choose a different kernel one simply needs to change the option to, say `kernel="cosine"`, while many other options are available. However, it is known in practice that the shape of the kernel is not particularly important and what drives the smoothness of the estimator is the bandwidth h. Let us explore visually the dependence of the KDEs on the bandwidth. Figure 8.4 displays 3 KDE with three different bandwidths $h = 0.5, 3, 15$ (expressed in minutes because `WASO` is) corresponding to the blue, red, and orange lines, respectively. The blue KDE ($h = 0.5$) has a wiggly appearance because the bandwidth is too small and data tend to be undersmooth. A larger value of the bandwidth (red, $h = 3$) seems to look much more appealing while maintaining the main features of the histogram. At the other extreme a bandwidth that is too large (orange, $h = 15$) tends to oversmooth the data and exacerbate the boundary problems by substantially

overestimating the probability that WASO is negative. Note that the choice of what is a small, moderate, and large bandwidth depends on the scale of the problem, number of observations, and structure of the observed data. Do not assume that $h = 0.5$, $h = 3$, and $h = 10$ will always be provide this type of result.

There is a vast literature on bandwidth choice, though in practice trying multiple bandwidth parameters and comparing them seems to be the reasonable and practical thing to do. Cross-validation, a method that removes one observation at a time and predicts using the remaining observations seems to be the theoretical winner. However, implementing cross validation for KDEs is still problematic, at least computationally.

```
d1<-density(WASO,bw=0.5)
d2<-density(WASO,bw=3)
d3<-density(WASO,bw=15)
plot(d1,col="blue",lwd=3,
     xlab="Wake after sleep onset (minutes)",
     main="Kernel density estimate (Gaussian)",
     cex.lab=1.3,cex.axis=1.3,col.axis="blue")
lines(d2,lwd=3,col="red")
lines(d3,lwd=3,col="orange")
legend(200,0.015,c("Bandwidth = 0.5","Bandwidth = 3","Bandwidth = 15"),
       lwd=c(3,3,3), col=c("blue","red","orange"),bty = "n")
```

Let us unpack the KDE formula to better understand how it is constructed. Note that, at every point, the KDE receives contributions from every observation in the sample. Suppose that we are interested in estimating the KDE at $x = 100$ minutes with a bandwidth $h = 3$. The individual contribution of an observation x_i is

$$K\left(\frac{x - x_i}{h}\right) = \frac{1}{\sqrt{2\pi}h} e^{-\frac{(100-x_i)^2}{2h^2}} .$$

```
contribution<-round(dnorm(WASO,mean=100,sd=3),digits=5)
values<-cbind(WASO[1:10],contribution[1:10])
colnames(values)<-c("WASO","Kernel contributions")
values
```

```
      WASO Kernel contributions
 [1,]  65.0              0.00000
 [2,]  43.0              0.00000
 [3,]  73.0              0.00000
 [4,]  43.5              0.00000
 [5,] 100.5              0.13115
 [6,]  82.5              0.00000
 [7,]  41.5              0.00000
 [8,] 103.0              0.08066
 [9,]  52.5              0.00000
```

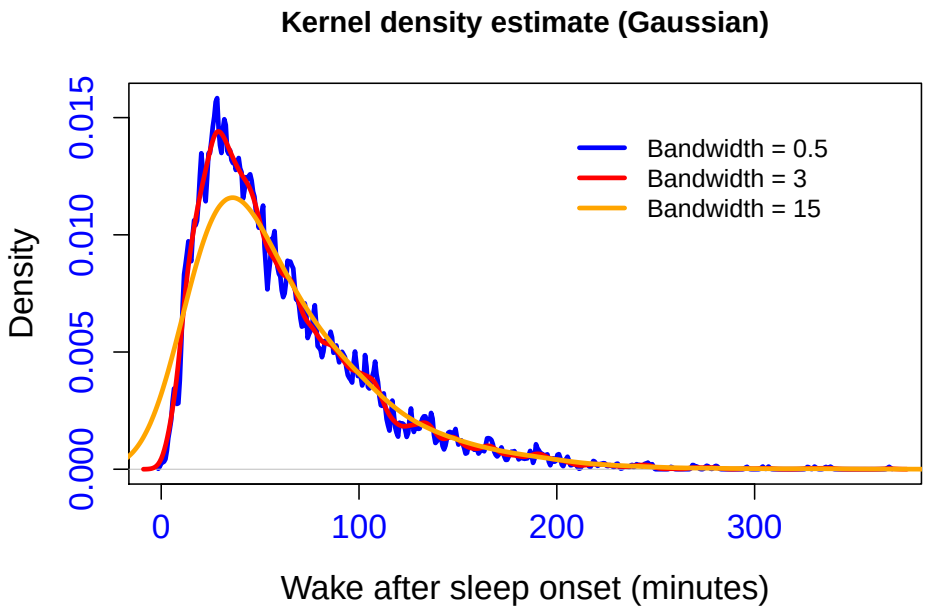


Figure 8.4: Kernel density estimators of the pdf of WASO in the SHHS. The three KDEs correspond to a small (blue line), moderate (red line), and large (orange line) bandwidth. The y-axis is on the same scale as the histogram on the probability scale.

```
[10,]  67.5              0.00000
```

Among the first 10 values of `WASO` only two, the fifth and the eighth, seem to have a sizeable relative contribution to the Kernel estimate at $x = 100$. The other contributions are not exactly zero, but are very small. The reason is that the fifth observation is 100.5 and the eighth is 103.0, which are relatively close to the mean of the Normal kernel, 100. Indeed, one can think about the Kernel as the pdf of a Normal density $N(x, h^2)$, indicating that all values that are farther than $3h$ from the mean x will contribute negligibly to the KDE. In our case the mean is 100 and $h = 3$, indicating that it is mainly `WASO` values between 91 minutes and 109 minutes that contribute to the KDE at $x = 100$. There are, in fact, 425 values of `WASO` between $(91, 109)$ minutes and 452 whose contribution is larger than 0.001. The idea here is that the KDE at the point $x = 100$ is a weighted average obtained by giving higher weight to observations closer to $x = 100$ and lower weight to observations that are further away. The weight assignment is done smoothly and the KDE can be estimated at any value x.

8.3 Scatterplots

Histograms and KDEs are used in practice to explore the distribution of individual variables. We call these distributions the marginal distributions to distinguish them from joint distributions, which refer to the distribution of multiple variables, and conditional distributions, which refer to the distribution of one or several random variables, given the fixed values of one or multiple other variables. Scatterplots are useful for visually exploring the joint and conditional distributions for two variables. This is sometimes used for three or more variables either by going to 3D plotting or by adding colors for the points. With the exception of textbook examples, we have found these approaches to be quite distracting and hard to understand both for the biostatistician and for the scientific researchers. In practice we recommend to do multiple 2-variable scatterplots to explore possibly unexpected associations between more than 2 variables.

Figure 8.5 displays `bmi_s1` versus `rdi4p` in the SHHS. There are 5761 individuals who do not have missing observations, as there are 43 individuals who have missing `bmi_s1` measurements. Also, the plot is cut at `rdi4p`< 30 for presentation purposes, though there are 349 subjects who have an `rdi4p` larger than 30 and do not appear in the plot. We have added two smooth fits using the `lowess` function in `R` (Cleveland 1979, 1981) (shown as an orange line) and penalized splines (O'Sullivan 1986; Eilers and Marx 1996; Ruppert, Wand, and Carroll 2003; Wood 2017), as implemented in the `mgcv` package developed and maintained by Simon Wood (Wood 2004, 2017) (shown as a red line).

```
plot(bmi_s1,rdi4p,pch=".",col=rgb(0,0,1,.2),cex=3,ylim=c(0,30),
     bty="l",cex.axis=1.3,col.axis="blue",main=NULL,xlab="BMI",
     cex.lab=1.3,ylab="RDI 4%")
```

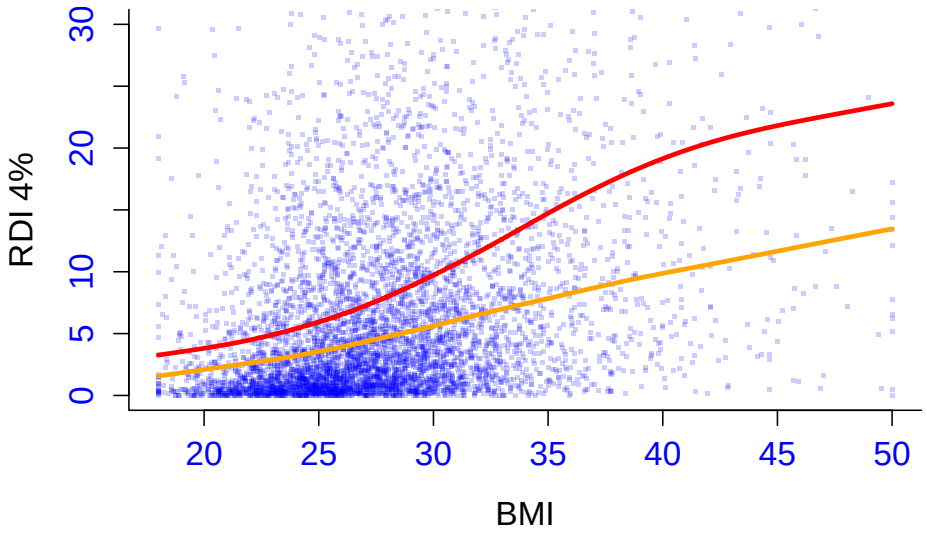


Figure 8.5: Scatterplot of BMI versus RDI in the SHHS. Two smooth lines are shown, one obtained from the lowess function (orange line) and one from the mgcv package (red line).

```
index_non_na<-!is.na(bmi_s1) & !is.na(rdi4p)
lines(lowess(x = bmi_s1[index_non_na], y= rdi4p[index_non_na]),
      col="orange",lwd=3)
library(mgcv)
fit_gam<-gam(rdi4p~s(bmi_s1))
fitted_sm<-fit_gam$fitted.values
table_fitted=cbind(bmi_s1[index_non_na],fitted_sm)
table_ordered<-table_fitted[order(table_fitted[,1]),]
lines(table_ordered[,1],table_ordered[,2],
      col="red",lwd=3)
```

It is quite surprising that the two smooth lines are that far from one another, which raises questions about which smoother to use in practice. To investigate this a little further, we calculate the mean of the `rdi4p` among individuals who have their `bmi_s1` measurements and obtain that it is 8.67 events per hour. When taking the average of the lowess smooth estimate we obtain 4.93 events per hour and when taking the mean of the penalized spline smoother we obtain 8.67 events per hour, the exact value we obtained taking the straight average. This seems to indicate that the `lowess` estimator is not well calibrated, at least in this dataset. It is not immediately clear why this should be the case, but it is clear what can be done to at least check that results make sense. In addition, we will calculate and plot the means in several `bmi_s1` intervals and superimpose them over the graph.

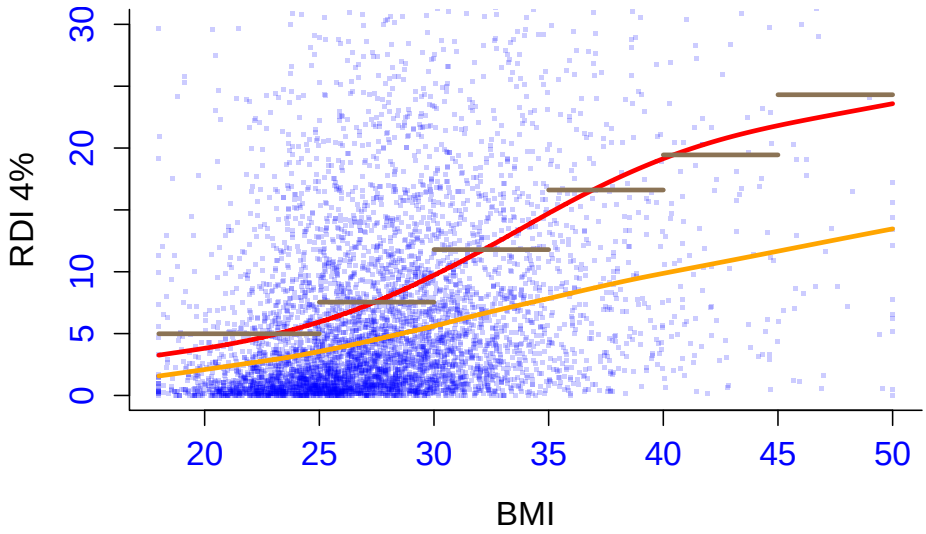


Figure 8.6: Scatterplot of BMI versus RDI in the SHHS. Two smooth lines are shown, one obtained from the lowess function (orange line) and one from the mgcv package (red line). Local means are shown in light brown to help indicate which smoother produces a reasonable estimator of the mean.

The light brown (coded `burlywood4` in `R`) colored lines in Figure 8.6 indicate the mean of all `rdi4p` values in a particular bin of `bmi_s1`. For example, the mean `rdi4p` for individuals with a `bmi_s1` between $(25, 30]$ is 7.54. These local means show close agreement with the penalized spline smooth and are quite far from the `lowess` fit. This problem with `lowess` may be more general than what we have seen in this example, but, at the very least, one should be careful about checking whether the fit passes simple checks, like those described here.

To further investigate why these differences occur we provide the same plot in Figure 8.7, but we indicate the means in each bin as a light brown (`burlywood4`) dot and the medians in each bin as a pink dot. It is now clear that the penalized spline smoother alligns very well with the local means, whereas the `lowess` estimator alligns well with the local medians. The reason for the difference is that the penalized spline smoother estimates the mean, whereas the `lowess` targets the median. Because our data are highly skewed these differences are more pronounced than what could be seen in other applications. It is likely that this is not a widely appreciated difference in biostatistics, as the function `lowess` has been used extensively in many applications. Figure 8.7 also displays the smooth estimator of the function `loess` (confusingly pronounced the same as `lowess`); the results are shown as a blue solid line, which closely tracks with the red line associated with the penalized spline smooth estimator from mgcv. Warning: functions with similar names may provide completely different results even if their description indicates that they do the same thing. It is always

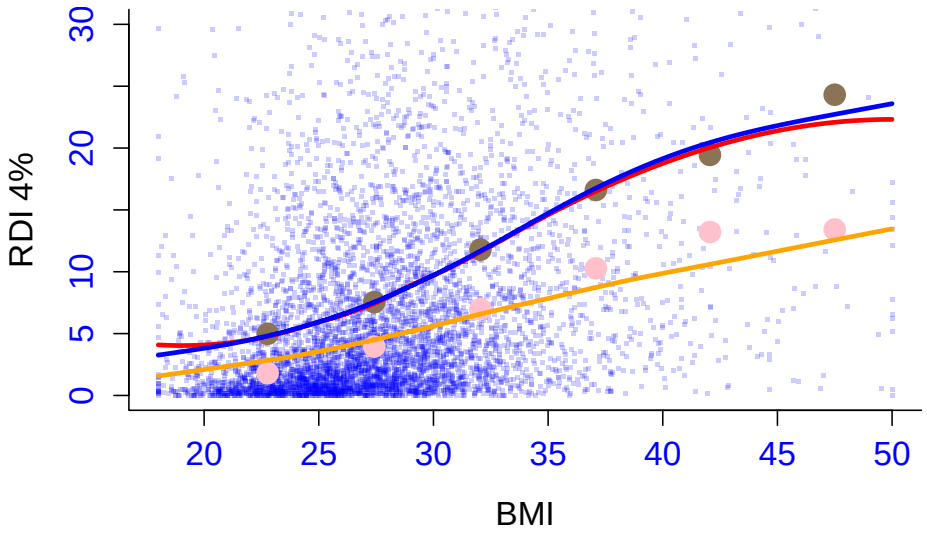


Figure 8.7: Scatterplot of BMI versus RDI in the SHHS. Three smooth lines are shown, one obtained from the lowess function (orange line), one from the mgcv package (red line), and one from the loess function (blue line). Local means and medians are shown as light brown and pink dots, respectively.

helpful to check, go back to basics, and plot some local means.

8.4 Dotplots

Dotplots simply display a dataset, one point per dot. The ordering of the dots and labeling of the axes can display additional information that can be used to better understand the data. Dotplots show a complete dataset and have high information density. However, it may be impossible or difficult to construct and interpret for datasets with lots of points. Figure 8.8 illustrate this using the `R` dataset `InsectSprays` (Beall 1942; McNeil 1977). The dataset contains two columns, the first one indicating the number of insects found on a plot after applying a particular insecticide and the second indicating the type of insecticide applied. Because the insecticide was applied on multiple plots there are multiple counts of insects for each insecticide. For example, there are 12 plots where insecticide `A` was applied.

```
set.seed(346672)
attach(InsectSprays)
#Setting the plot, the axes, and the labels
plot(c(.5, 6.5), range(count),xlab="Spray type",
     ylab="Insect count",bty="l",col="white",xaxt = 'n',
```

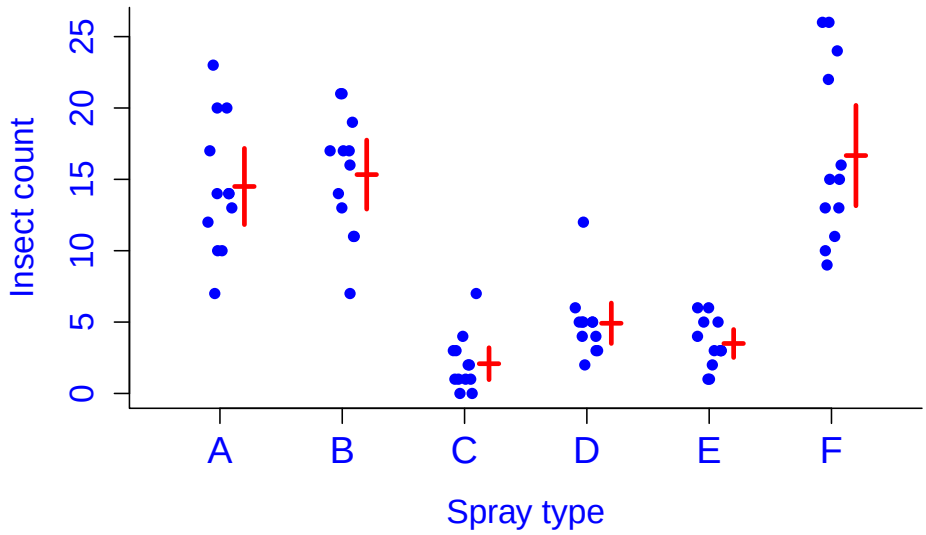


Figure 8.8: Dotplot of insecticide type by insect count (blue dots) using a slight jitter along the x-axis to avoid overplotting. Point estimators and 95% confidence of spray-type specific means are also shown (red).

```
     cex.axis=1.3,col.axis="blue",cex.lab=1.3,col.lab="blue")
axis(side=1,at=1:6,
     labels=c("A","B","C","D","E","F"),
     cex.axis=1.5,col.axis="blue")

#Identify all the unique spray labels, in this case the letters A through F
sprayTypes <- unique(spray)

#Plot the data by spray type
for (i in 1 : length(sprayTypes))
  {#begin loop over all insecticide types
  #Obtain the counts of insects for each spray type
  y <- count[spray == sprayTypes[i]]
  #Total number of plots for each spray type
  n <- sum(spray == sprayTypes[i])
  #Plot the counts, but jitter their x-value (i) to avoid overplotting
  points(jitter(rep(i, n), amount = .1), y,col="blue",pch=16)
  #Plot the mean count as a short horizontal line
  lines(i + c(.12, .28), rep(mean(y), 2), lwd = 3,col="red")
  #Plot the Normal 95% CI for the mean insect count
  lines(rep(i +.2,2), mean(y)+c(-1.96, 1.96)*sd(y)/sqrt(n),col="red",lwd=3)
  }#End loop over all insecticide types
```

We consider that, whenever possible, plotting the entire dataset is preferable, though adding the mean per group together with the confidence intervals for the mean can add information. The data were jittered slightly along the x-axis to avoid overplotting. Simply showing the mean and the confidence intervals for the mean would remove a lot of the information from the plot, especially about outliers (see data for insecticide C and D) or multimodality (see data for insecticide F). Boxplots and scatterplots become difficult to use when datasets grow in size.

8.5 Boxplots

Boxplots were introduced by John Tukey (Tukey 1970). Boxplots are useful for the same sort of display as the dotplot, but are used in instances when displaying the whole dataset is cumbersome or not possible. The centerline of the boxes represents the median while the box edges correspond to the quartiles. Therefore, 50% of the data is contained within the boxplot box. The whiskers of the boxplot extend out to a constant times the interquartile range (IQR). Sometimes, potential outliers are identified as points beyond the whiskers. Skewness of the distribution could be indicated by the centerline being near one of the box edges. Let us investigate a few ways of producing boxplots.

Consider the SHHS and we will focus on two groups. The first group has a `bmi` below 25 and age below 50 and the second group has a `bmi` above 35 and age above 70. We would like to compare these groups in terms of respiratory disturbance index, `rdi4p`. Because the data are highly skewed in both groups we choose to take the square root of `rdi4p`. Figure 8.9 displays the square root of `rdi4p` in the two groups.

```
index_g1<-(bmi_s1<25) & (age_s1<50)
index_g2<-(bmi_s1>35) & (age_s1>70)

sqrt_rdi_g1<-sqrt(rdi4p[index_g1])
sqrt_rdi_g2<-sqrt(rdi4p[index_g2])
data.box<-list(sqrt_rdi_g1,sqrt_rdi_g2)
par(bg="lightcyan")
plot(1.5, 1.5, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
boxplot(data.box,col=c(col=rgb(1,0,1,0.5),col = rgb(1,0,0,0.5)),
        cex.lab=1.3,cex.axis=1.3,col.axis="blue",col.lab="blue",
        names=c("Group 1","Group 2"),ylab="Square root of RDI 4%",
        border=c("blue","blue"),
        outpch=20,outcex=1,
        outcol=c(col=rgb(1,0,1,0.5),col = rgb(1,0,0,0.5)))
```

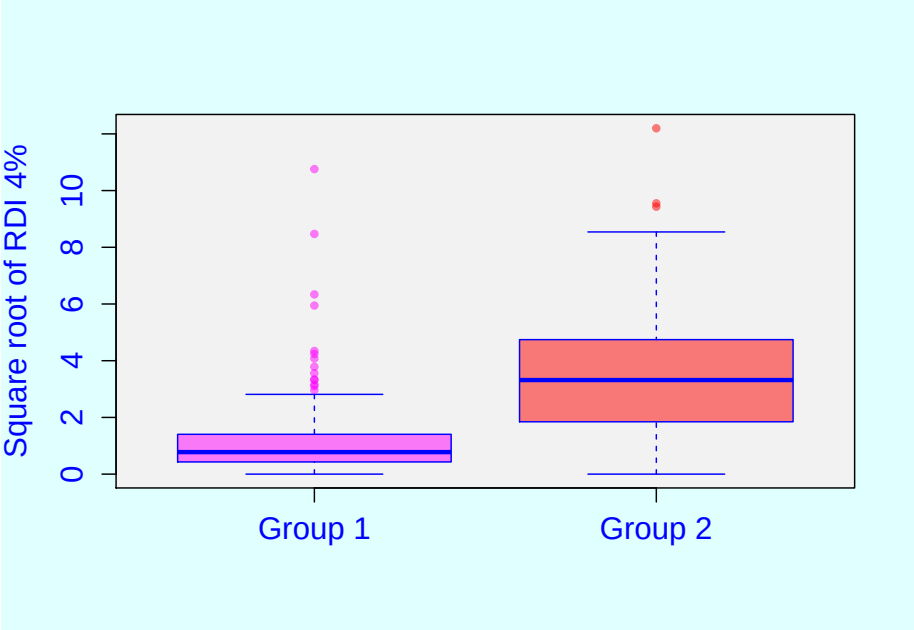


Figure 8.9: Boxplot of the square root of RDI in Group 1 (BMI< 25 and age< 50) and Group 2 (BMI> 35) and age> 70.

We have used several customizations, most of which are self-explanatory. An important property is that we can change the colors of the boxes using the option `col`, the border of the boxes using the option `border`, and of the outliers using `outcol`. We have also chosen to change the symbol for displaying the outliers using `outpch=20`, which provides a filled dot versus the empty circle. Here we kept the original size of the symbol for the outliers using `outcex=1` because we have few outliers. As we will see below, in cases when there are many outliers, we may choose to make the symbol smaller to improve appearance.

Figure 8.10 displays the boxplots of `rdi4p` separated in six groups by `bmi_s1` categories. We plot directly the `rdi4p` and not the square root because this time we want to retain the original interpretation of the `rdi4p`. We cut the plot at 100 events per hour for presentation purposes, which only cuts 4 subjects from the graph. Because there are many outliers in this plot we display them as filled points, but reduce the size of the symbol using `outcex=0.4`.

```
break_bmi<-c(min(bmi_s1,na.rm=TRUE)-0.01,25,30,35,40,45,
             max(bmi_s1,na.rm=TRUE)+0.01)
bmi_cut<-cut(bmi_s1,breaks=break_bmi,
             labels=c("[18,25]","(25,30]","(30,35]",
                      "(35,40]","(40,45]","[45,50]"))
```

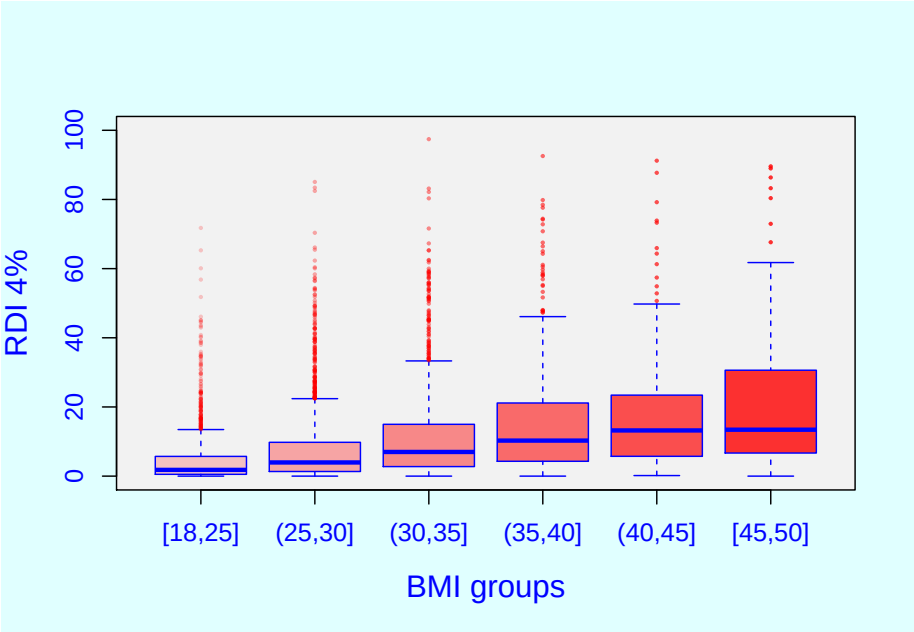


Figure 8.10: Boxplot of RDI in six groups defined by cutoff points on BMI.

```
par(bg="lightcyan")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
res<-boxplot(rdi4p~bmi_cut,ylim=c(0,100),
             col=rgb(1,0,0,seq(0.2,0.8,length=6)),
             cex.lab=1.3,cex.axis=1.1,
             col.axis="blue",col.lab="blue",
             ylab="RDI 4%",xlab="BMI groups",
             border=rep("blue",6),
             outpch=20,outcex=0.4,
             outcol=rgb(1,0,0,seq(0.2,0.8,length=6)))
```

The color scheme we have chosen for the boxplots is pleasing, but the shading of the colors may not be the best. Indeed, we have simply increased the transparency of red linearly as a function of the `bmi_s1` group, which may be interpreted as a linear increase in the median `rdi4p`. However, the plot suggests that the median seems to plateau in the last three BMI groups. Thus, it may make more sense to color the boxplots according to the median `rdi4p` in that particular age group. To do this, we assign transparency according to how far the `rdi4p` median of the group is from the `rdi4p` median of groups $[18, 25]$ and

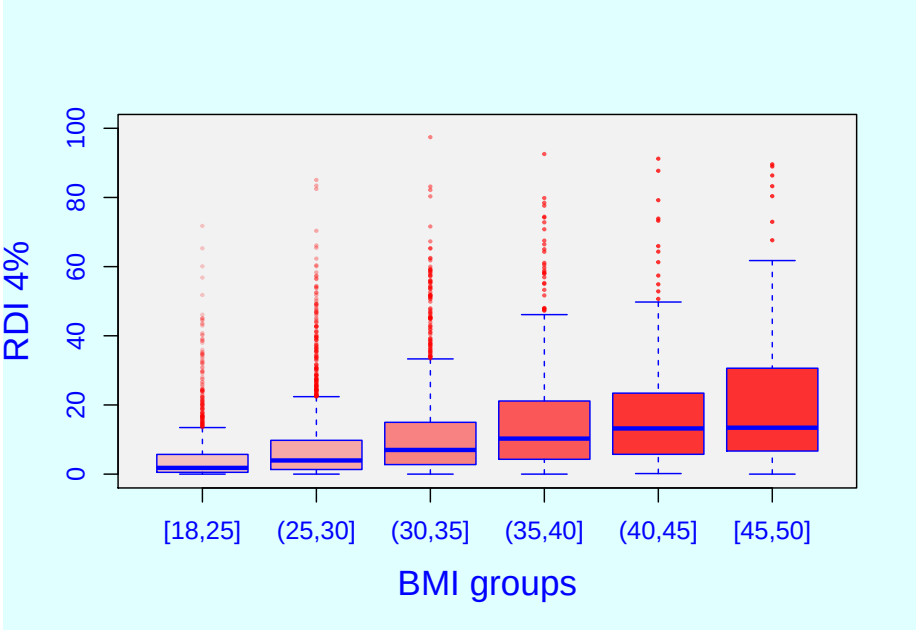


Figure 8.11: Boxplot of RDI in six groups defined by cutoff points on BMI. The coloring of the boxplots is refined to more closely reflect the changes in median RDI.

[45, 50], respectively. Figure 8.11 displays the same information as Figure 8.10, though the shading of the boxplots is connected to the median RDI in the corresponding group. A comparison of the two plots will indicate that the last three boxplots are now colored in closer shades of red, indicating that the increase in median `rdi4p` is not as strong among the last three groups.

```
medians<-res$stats[3,]
color_pal<-0.2+(0.8-0.2)/(medians[6]-medians[1])*(medians-medians[1])

par(bg="lightcyan")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
boxplot(rdi4p~bmi_cut,ylim=c(0,100),col=rgb(1,0,0,color_pal),
        cex.lab=1.5,cex.axis=1.1,col.axis="blue",col.lab="blue",
        ylab="RDI 4%",xlab="BMI groups",
        border=rep("blue",6),
        outpch=20,outcex=0.4,
        outcol=rgb(1,0,0,color_pal))
```

Suppose that we want to do something more complex, separating the 6 `bmi_s1` groups by gender. Because this creates 12 boxplots, labeling becomes an issue. Thus, we relabel the groups such that F1 stands for female in BMI group 1, that is, females with BMI between $[18, 25]$. The label M3 stands for male with BMI between $(30, 35]$, the third BMI group. Below we display these 12 boxplots and color the female groups magenta and male groups red to make them easier to compare. It makes sense to compare male versus female groups within the same BMI group (e.g. F1 versus M1 and F3 versus M3), compare females across BMI groups (e.g. F1 versus F4), compare males across BMI groups (e.g. M2 versus M6), and males versus females across BMI groups (e.g. M2 versus F5). Figure 8.12 displays the boxplots of RDI for these 12 groups, which separate each of the six groups in Figures 8.10 and 8.11 into two subgroups, one for females and one for males.

```
par(bg="lightcyan")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
res<-boxplot(rdi4p~gender*bmi_cut,ylim=c(0,100),
             col=c(col=rgb(1,0,1,0.5),col = rgb(1,0,0,0.5)),
             cex.lab=1.3,cex.axis=1,col.axis="blue",col.lab="blue",
             ylab="RDI 4%",xlab="BMI groups",
             names=c("F1","M1","F2","M2","F3","M3",
                     "F4","M4","F5","M5","F6","M6"),
             border=rep("blue",6),
             outpch=20,outcex=0.4,
             outcol=c(col=rgb(1,0,1,0.5),col = rgb(1,0,0,0.5)))
```

Ignoring the outliers the plot reveals that men tend to have higher median (and quartiles) `rdi4p` for every BMI group, that the increase in median `rdi4p` is much higher for men in the higher BMI groups, and that the distribution of `rdi4p` for women tends to be roughly equivalent to that in men in two groups lower for BMI (F3 versus M1, F4 versus M2, F5 versus M3, and F6 versus M4). We might want to have the intensity of the color reflect the size of the median `rdi4p` in a particular group. However, things get trickier when we have to deal with two variables (in our case `gender` and `bmi_s1`) that both have a strong effect on the median `rdi4p`. Indeed, there are several options that we could consider. The first could be to choose the color intensity within BMI groups separated for males and females. This will highlight the within-gender effect, but may visually reduce the between-gender effects. Indeed, with this choice the color intensity for F6 and M6 would be the same. An alternative would be to choose the color intensity by pooling all medians together. A potential problem with this approach is that the color intensity in M6 may visually dominate the other intensities, simply because the median for the M6 group is so large. This second choice is shown in Figure 8.13.

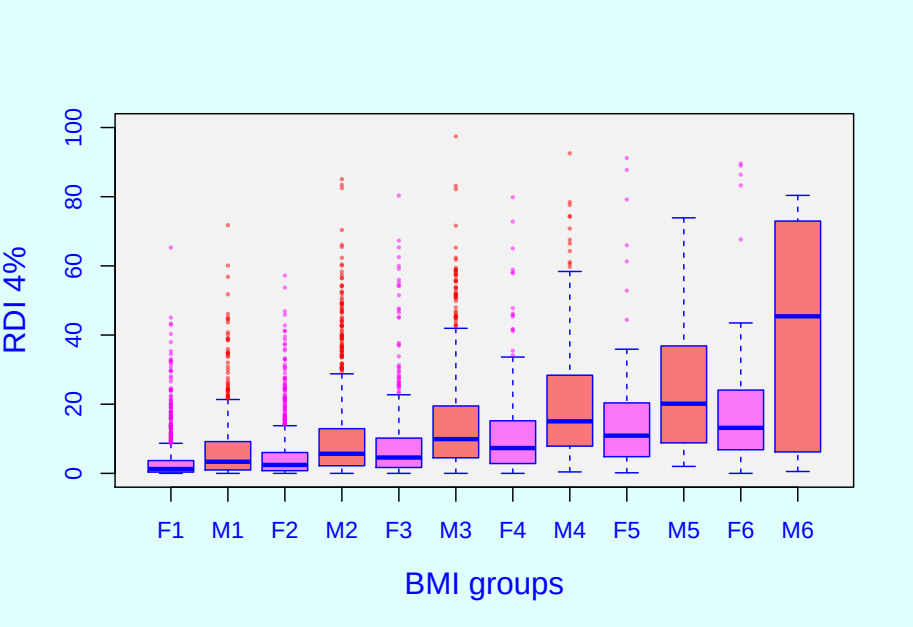


Figure 8.12: Boxplot of RDI in twelve groups defined by cutoff points on BMI and gender. We use the same color for all six men groups (light red) and all six women groups (light magenta).

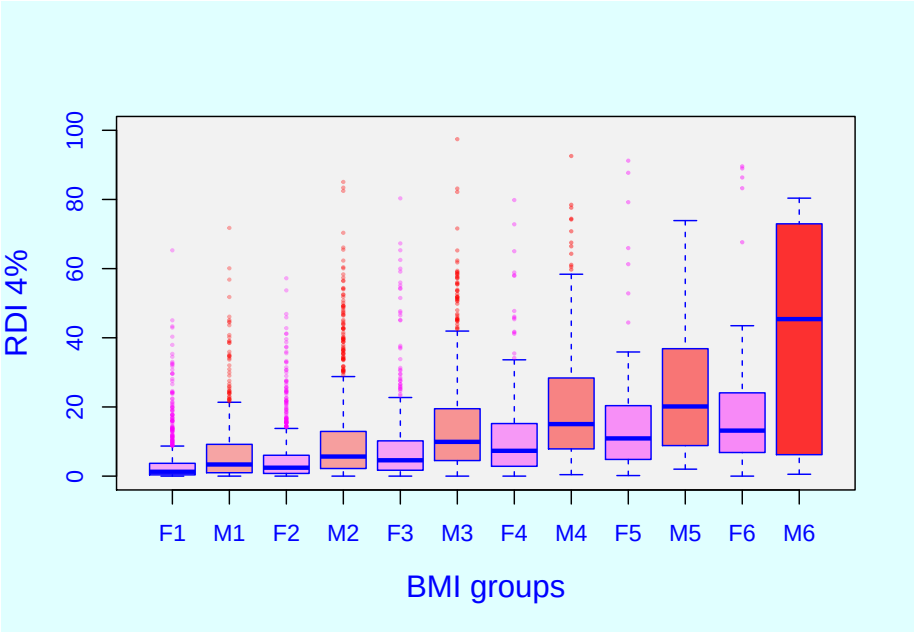


Figure 8.13: Boxplot of RDI in twelve groups defined by cutoff points on BMI and gender. We use the same color for all six men groups (light red) and all six women groups (light magenta). However, the color transparency is associated with the median RDI in each group.

```
medians<-res$stats[3,]
color_pal<-0.3+(0.8-0.3)/(medians[12]-medians[1])*(medians-medians[1])

par(bg="lightcyan")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
res<-boxplot(rdi4p~gender*bmi_cut,ylim=c(0,100),
             col=c(col=rgb(1,0,rep(c(1,0),6),color_pal)),
             cex.lab=1.3,cex.axis=1,col.axis="blue",col.lab="blue",
             ylab="RDI 4%",xlab="BMI groups",
             names=c("F1","M1","F2","M2","F3","M3",
                     "F4","M4","F5","M5","F6","M6"),
             border=rep("blue",6),
             outpch=20,outcex=0.4,
             outcol=c(col=rgb(1,0,rep(c(1,0),6),color_pal)))
```

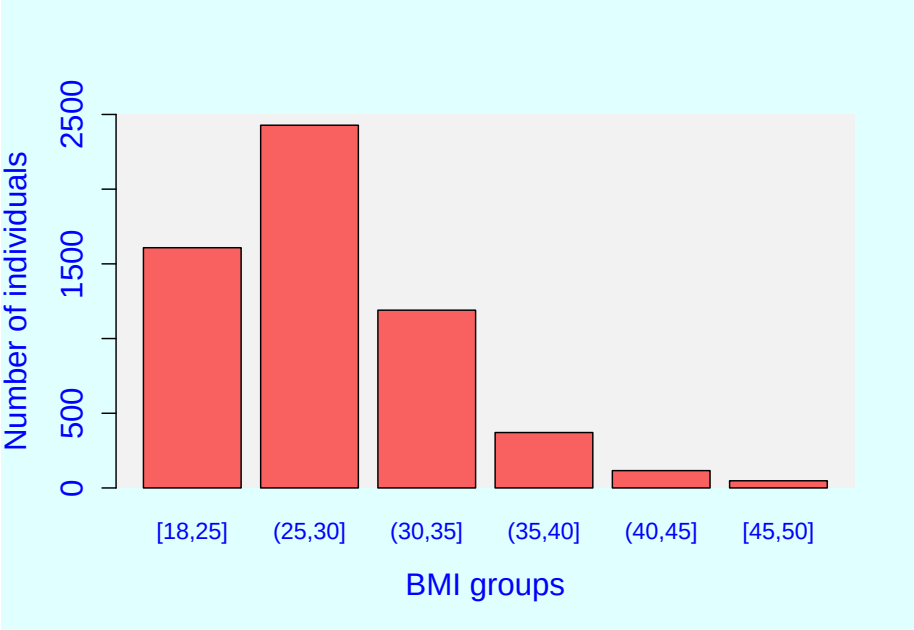


Figure 8.14: Bar plot of the number of subjects by BMI group in the SHHS.

8.6 Bar plots and stacked bar plots

Bar plots can be used in many situations, though they are most useful for indicating the number of observations in particular groups. Suppose, for example, that we would like to represent the number of subjects that have a BMI in one of the groups we have defined. Figure 8.14 displays the bar plot indicating the number of subjects in each BMI group from the SHHS

```
counts <- table(bmi_cut)
par(bg="lightcyan")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)

barplot(counts, main="", ylab="Number of individuals",
    xlab="BMI groups",col=rgb(1,0,0,0.6),ylim=c(0,2500),
    cex.axis=1.3,col.axis="blue",cex.lab=1.3,col.lab="blue")
```

The plot contains the same information as the table

```
counts
```

```
bmi_cut
```

```
[18,25] (25,30] (30,35] (35,40] (40,45] [45,50]
   1608    2428    1190     371     116      48
```

which shows, for example, that there are 1190 subjects who have a `bmi_s1` between $(30, 35]$. The bar plot is slightly less informative than the table, but it provides an intuitive representation of the data. Supposed that we would like to visually separate the number of subjects in each BMI group by smoking status. The Figure 8.15 displays the stackked bar plot indicating the number of subjects in every BMI group separated by "Former" (yellow), "Current" (light green), and "Never" (dark green) smoker.

```
smokstatus<-factor(smokstat_s1,
                   levels = c(0,1,2),
                   labels = c("Never", "Current", "Former"))
counts <- table(smokstatus,bmi_cut)

par(bg="oldlace")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
barplot(counts, main="",
        ylab="Number of individuals",
        xlab="BMI groups",
        col=c("darkslategray","darkseagreen3","khaki1"),
        ylim=c(0,2500),cex.axis=1.3,col.axis="darkslategray",
        cex.lab=1.3,col.lab="darkslategray",
        legend = rownames(counts))
```

The information contained in the plot is the same as in the two by two table shown below.

```
          bmi_cut
smokstatus [18,25] (25,30] (30,35] (35,40] (40,45] [45,50]
   Never       793    1118     541     165      49      22
   Current     206     222      95      17      10       5
   Former      603    1073     545     185      56      21
```

For example, there are 222 individuals who are current smokers and have a BMI between $(25, 30]$, shown as the middle color `darkseagreen3` in the second bar, and 541 individuals who were never smokers and have a BMI between $(30, 35]$, shown as the top color `khaki1` in the third bar. Because there are fewer individuals who are current smokers and with very high BMI values, it becomes hard to represent the numbers in the table as color bars. Indeed, the columns corresponding to BMI values between $(40, 45]$ and $[45, 50]$ are not particularly useful. To address this problem we transform the data from counts into fractions and provide stacked bar-plots for the fractions. The resulting plot is shown in Figure 8.16. If we want the same plots on a scale of proportions

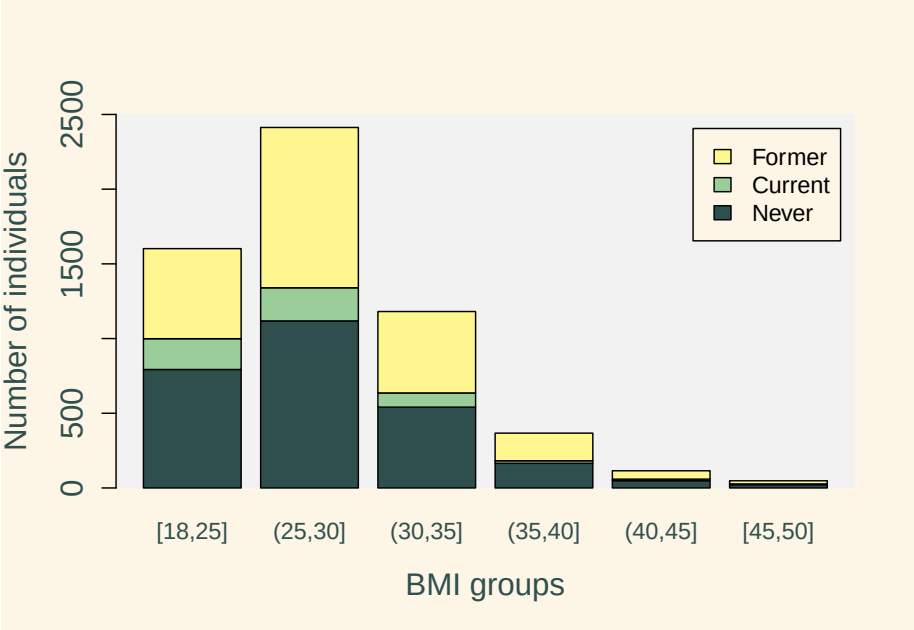


Figure 8.15: Bar plot of the number of subjects by BMI group in the SHHS separated by smoking status.

(from $[0, 100]\%$) instead of the scale of fractions (from $[0, 1]$) we need to multiply the `propto` variable below with 100.

```
counts <- table(smokstatus,bmi_cut)
totalc<-colSums(counts)
propt<-counts
for (i in 1:ncol(counts))
  {propt[,i]<-counts[,i]/totalc[i]}
```

The difference between Figure 8.16 and the stacked bar plot for counts in Figure 8.15 is substantial. Indeed, all bar plots have a total height of 1, irrespective of how many observations there are per BMI group. However, the plot provides additional information that was not readily available from the counts plot. Indeed, it seems that the proportion of never smokers decreases slightly from around 50% in the BMI category $[18, 25]$ to 42% in the BMI category $(40, 45]$. For the same BMI categories the proportion of former smokers increases slightly from around 40% to around 50%, while the proportion of current smokers hovers roughly around $8 - 10\%$. Because all bars have exactly the same height, we needed to add space to accommodate the legend to the right of the plot. One could, of course, relocate the legend, depending on what is in the eye of the beholder. The color scheme for bar plots is important for illustrative purposes.

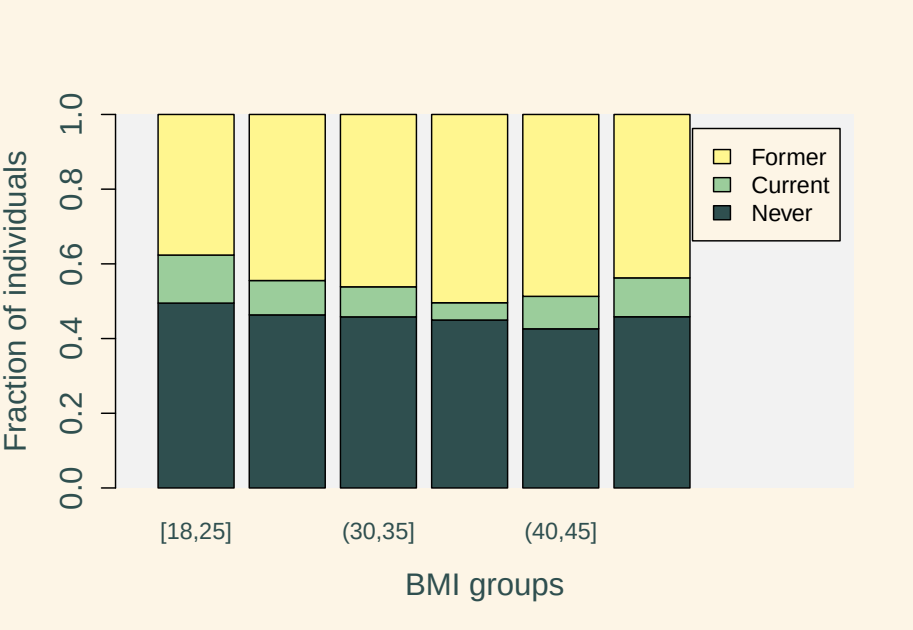


Figure 8.16: Bar plot of the proportion of subjects within BMI group corresponding to each smoking status.

One of the problems with the previous bar plot is that the total number of observations for each BMI group is lost. This can be recovered and posted on top of each bar, as we describe below. The corresponding plot is shown in Figure 8.17.

```
par(bg="oldlace")
plot(3, 3, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
stat<-barplot(propt, main="",
        ylab="Fraction of individuals",
        xlab="BMI groups",
        col=c("darkslategray","darkseagreen3","khaki1"),
        ylim=c(0,1.1),xlim=c(0,9),cex.axis=1.2,
        col.axis="darkslategray",
        cex.lab=1.2,col.lab="darkslategray",
        legend = rownames(counts))
text(x = stat, y = rep(1.05,6), label = totalc,
     col = "darkslategray")
```

Instead of stacking bar plots we could convey the same information by showing

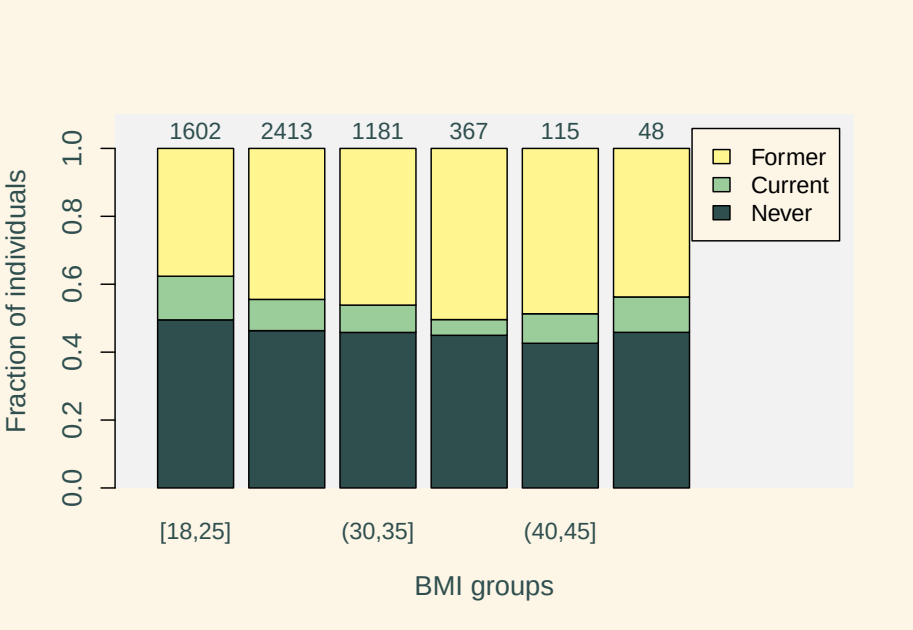


Figure 8.17: Bar plot of the proportion of subjects within BMI group corresponding to each smoking status. Adding total counts for each group and legend.

the within-group information as side-by-side bars. Figure 8.18 shows the same information as Figure 8.17, but arranged side-by-side instead of stacked bars.

Mosaic plots are a version of stacked bar plots that are sometimes used in practice. They are designed to represent the same information as a stacked bar plot for fractions. Mosaic plots display the stacked columns slightly differently with larger stacked rectangles that have less space between them. Some of the standard information in the stacked plots is suppressed. An example of the mosaic plot is shown in Figure 8.19.

We show one more mosaic plot on a classic data set published by R.A. Fisher (Fisher 1940) on the cross-classification of people in Caithness, Scotland, by eye and hair colour. The data set is a component of the `MASS` package in `R` (Venables and Ripley 2002) and the corresponding mosaic plot is shown in Figure 8.20.

```
       fair red medium dark black
blue    326  38    241  110     3
light   688 116    584  188     4
medium  343  84    909  412    26
dark     98  48    403  681    85
```

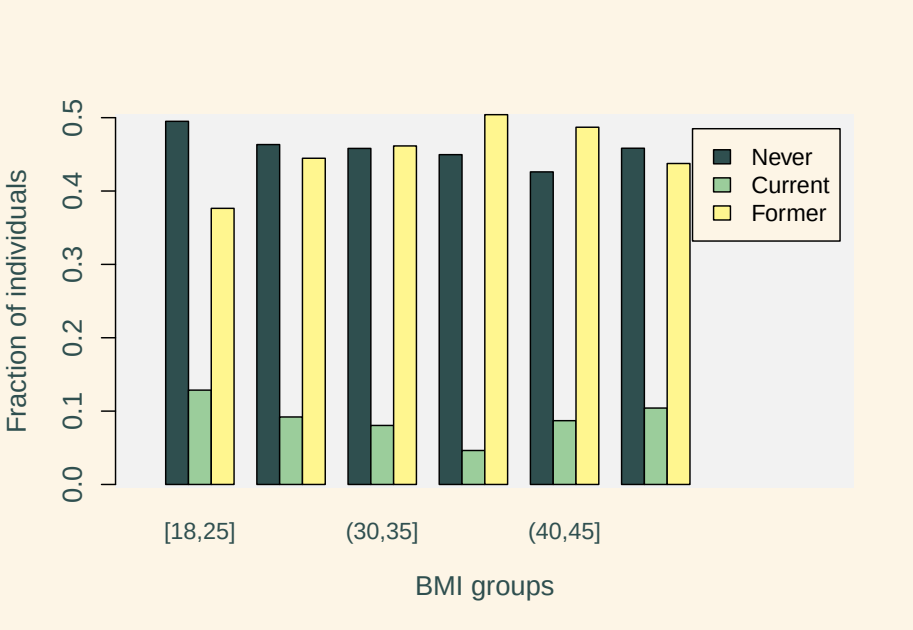


Figure 8.18: Side-by-side bar plot of the proportion of subjects within BMI group corresponding to each smoking status.

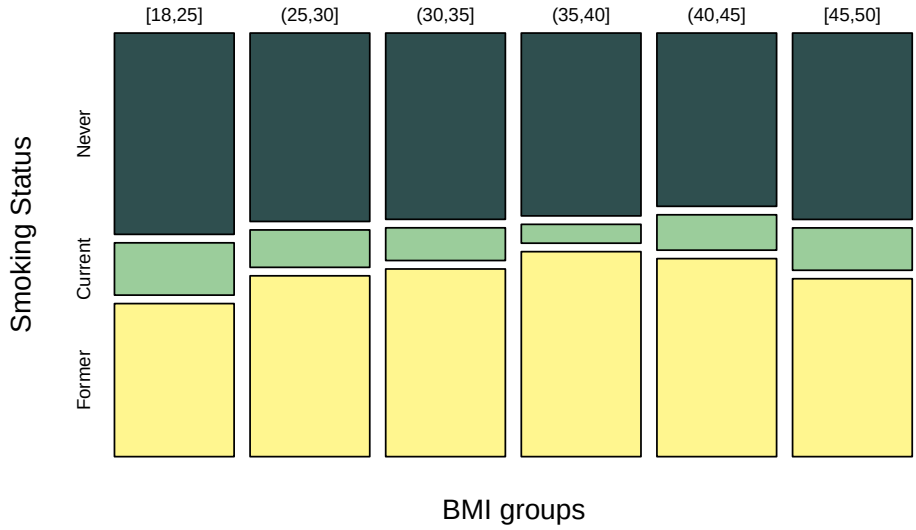


Figure 8.19: Mosaic plot, a very close form of plotting to the stacked bar plot. Displayed is the proportion of subjects within BMI group corresponding to each smoking status.

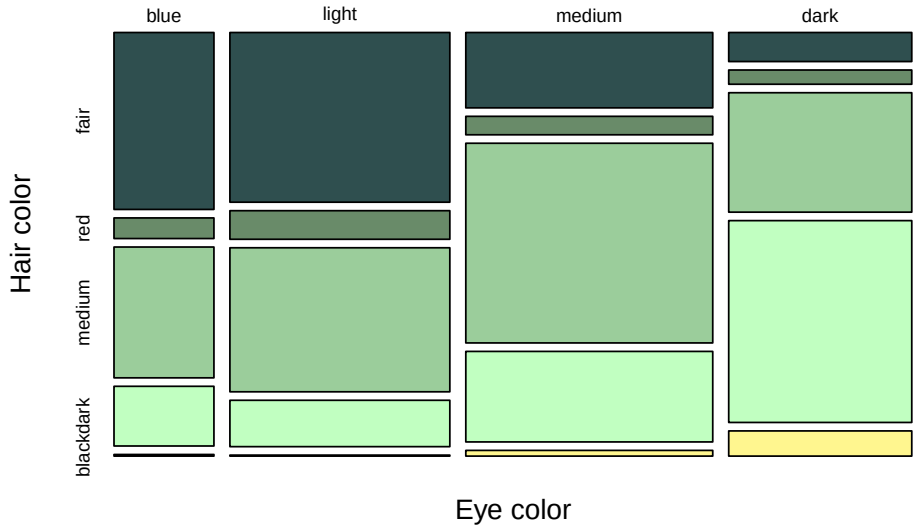


Figure 8.20: Mosaic plot, a very close form of plotting to the stacked bar plot. Displayed is the proportion of subjects with black, dark, fair, medium, red, and fair hair among individuals with a particular eye color.

8.7 QQ-plots

QQ-plots (for quantile-quantile) are useful for comparing data to a theoretical distribution as well as for comparing distributions with each other. The QQ-plot displays the quantiles of one distribution against the quantiles of the second distribution. Both distributions can be empirical or theoretical, though the most common approach is to compare the quantiles of a distribution with those of a $N(0,1)$ distribution.

Let z_p and x_p be the pth quantile from $N(0,1)$ and $N(\mu,\sigma^2)$ distributions, respectively. If $X \sim N(\mu,\sigma^2)$ is a random variable then

$$p = P(X \leq x_p) = P\left(\frac{X-\mu}{\sigma} \leq \frac{x_p-\mu}{\sigma}\right) .$$

Since $(X-\mu)/\sigma \sim N(0,1)$ it follows that

$$x_p = \mu + z_p\sigma .$$

This result implies that the quantiles of a $N(\mu,\sigma^2)$ distribution are linearly related to the standard normal quantiles. The intercept of the line is equal to the mean μ and the slope is equal to the standard deviation, σ. Thus, if we calculate the quantiles of a sample from a Normal distribution and plot them against the theoretical quantiles of a standard Normal distribution, then we expect the resulting graph to approximate a line.

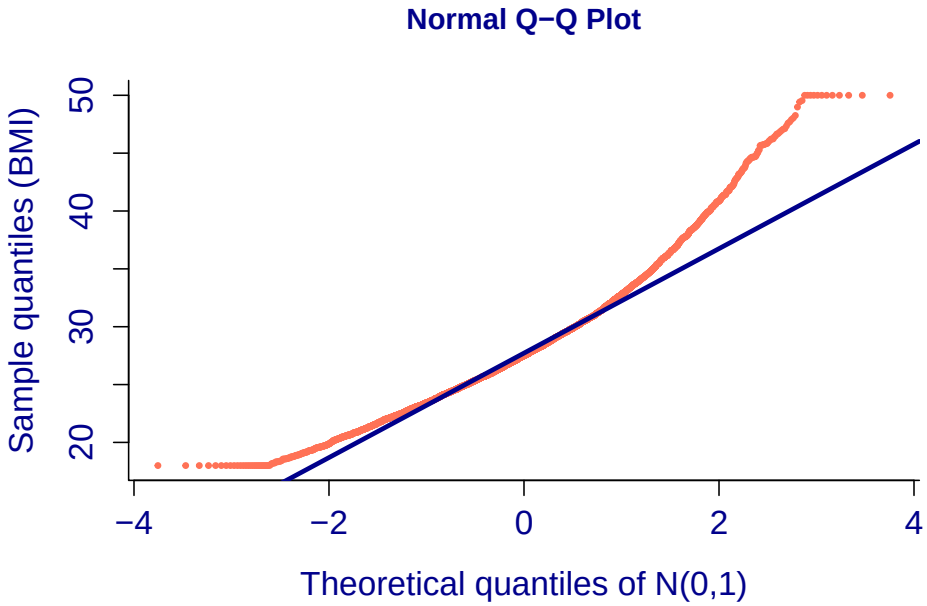


Figure 8.21: QQ-plot for the empirical quantiles of BMI distribution in SHHS versus the theoretical quantiles of the N(0,1) distribution.

A normal QQ-plot displays the empirical quantiles against the theoretical standard normal quantiles. In `R` the function `qqnorm` is used for a normal QQ-plot and the `qqplot` is used when checking against an arbitrary distribution (Becker, Chambers, and Wilks 1988). Let us investigate several distributions from the SHHS versus the Normal distribution. Figure 8.21 provides the QQ-plot for the empirical quantiles of the BMI (`bmi_s1`) distribution in SHHS versus the theoretical quantiles of the $N(0,1)$ distribution.

```
qqnorm(bmi_s1,pch=20,cex=0.7,col="coral1",
       ylab="Sample quantiles (BMI)",
       xlab="Theoretical quantiles of N(0,1)",
       cex.axis=1.3,col.axis="darkblue",col.main="darkblue",
       cex.lab=1.3,col.lab="darkblue",bty="l")
qqline(bmi_s1,col="darkblue",lwd=3)
```

We have changed quite a few parameters, especially the type of symbol for the points, `pch=20`, as the original empty circles look messier, especially for larger samples. The `qqline` function is the line that passes through the first and third quantiles of the $N(0,1)$ and of the BMI distribution, respectively. One could change the quantiles that are matched using the argument `prob=c(0.25,0.75)` in the `qqline` function. The interpretation is that there is some agreement between the two distributions, especially in the middle of the distribution, but the agreement is quite poor in both tails of the distribution. The right side of the

plot shows that the points are moving away and up from the line, indicating that the quantiles of the empirical distribution are larger than those of a theoretical Normal. For example, the 0.975 quantile (corresponding to 1.96 on the x-axis) would require a 0.975 quantile of BMI of about 35.

However, the 0.975 quantile of the `bmi_s1` variable is 40.56, which is much larger. This indicates that the right tail of the `bmi_s1` distribution is heavier than the right tail of a Normal distribution. This means that the probability of observing more extreme BMI values is higher than one would expect under a Normal distribution. We now show how to obtain the exact numbers, not just approximations, using plot inspection. The `qqline` intercept and slope can be calculated below

```
# Find the 1st and 3rd empirical quartiles
y <- quantile(bmi_s1,c(0.25, 0.75),na.rm=TRUE)
# Find the 1st and 3rd N(0,1) quartiles
x <- qnorm( c(0.25, 0.75))
# Compute the line slope
slope <- diff(y) / diff(x)
# Compute the line intercept
intercept <- y[1] - slope * x[1]
expectedq<-round(intercept+slope*qnorm(c(0.025,0.975)),digits=2)
```

Thus, the expected 0.975 quantile of a Normal matched at the 0.25 and 0.75 quantiles is 36.57, which, indeed, is close to the 35 we obtained by inspecting the plot, but not identical.

The left side of the plot shows that the points are moving away and up from the line, indicating that the quantiles of the empirical distribution are, again, larger than those of a theoretical Normal. This indicates that the left tail of the distribution is actually lighter than that of a Normal. Indeed, the BMI quantiles are larger than one would expect from a Normal distribution. For example, the 0.025 quantile of the matched Normal is 18.87, which is smaller than 20.13, the 0.025 empirical quantile of BMI. This type of behavior of the left tail of the distribution is often encountered in practice when dealing with positive or otherwise bounded random variables. The qq-plot reveals that both the right and upper tail quantiles of the `bmi_s1` distribution exhibit an unusual behavior as they seem to stabilize at 18 and 50, respectively. This happens because there are exactly 25 subjects with a `bmi_s1` equal to 18 and 12 subjects with a `bmi_s1` equal to 50, and 18 and 50 are the smallest and largest `bmi_s1` values in the SHHS, respectively.

```
qqnorm(age_s1,pch=20,cex=0.7,col="coral1",
       ylab="Sample quantiles (age)",
       xlab="Theoretical quantiles of N(0,1)",
       cex.axis=1.3,col.axis="darkblue",col.main="darkblue",
       cex.lab=1.3,col.lab="darkblue",bty="l")
qqline(age_s1,col="darkblue",lwd=3)
```

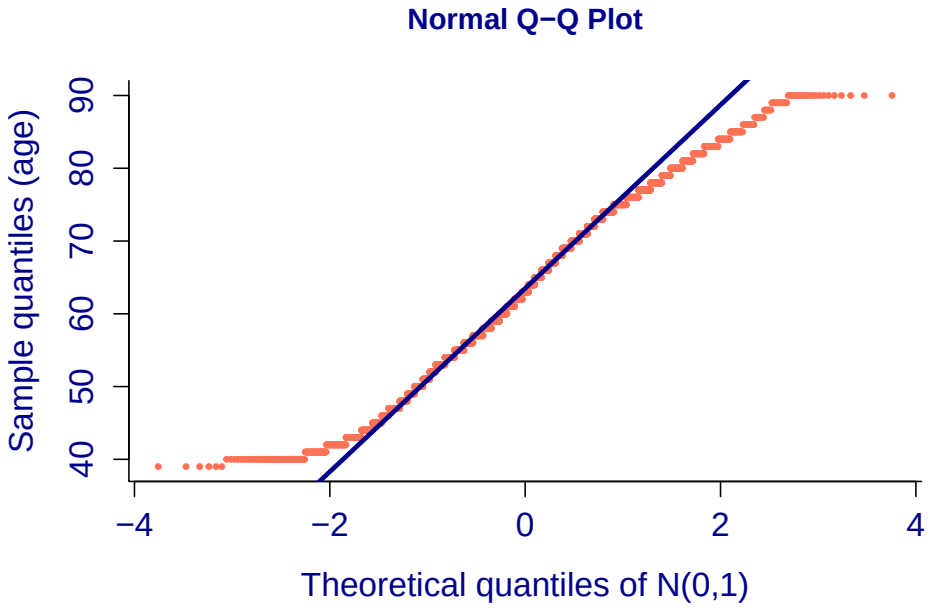


Figure 8.22: QQ-plot for the empirical quantiles of age distribution in SHHS versus the theoretical quantiles of the N(0,1) distribution.

Figure 8.22 displays the QQ-plot of the empirical distribution of age versus the quantiles of the theoretical $N(0,1)$ distribution. The S-shape of the QQ-plot indicates that the empirical distribution has lighter/shorter left and right tails than the Normal. This is quite unusual in practice as lighter tails than those of the Normal are quite uncommon. However, if data follow a Uniform distribution, then the tails are actually dissappearing beyond the range of the data. The horizontal lines in the plot are due to the fact that `age_s1` is a discrete variable in the SHHS because age is expressed in integers, without allowance for fractions of the year.

To check whether there is evidence that `age_s1` has a Uniform distribution we use the `qqplot` versus the quantiles of the Uniform distribution $U[39,90]$, where 39 is the minimum and 90 is the maximum age in the SHHS. Figure 8.23 displays the empirical quantiles of the age distribution in SHHS versus the theoretical quantiles of the $U[39,90]$ distribution.

```
mina=min(age_s1,na.rm=TRUE)
maxa=max(age_s1,na.rm=TRUE)

qqplot(qunif(ppoints(length(age_s1)),mina,maxa), age_s1,
       pch=20,cex=0.7,col="coral1",
       xlab="Theoretical quantiles for Uniform[39,90]",
       ylab="Sample quantiles (Age)",
```

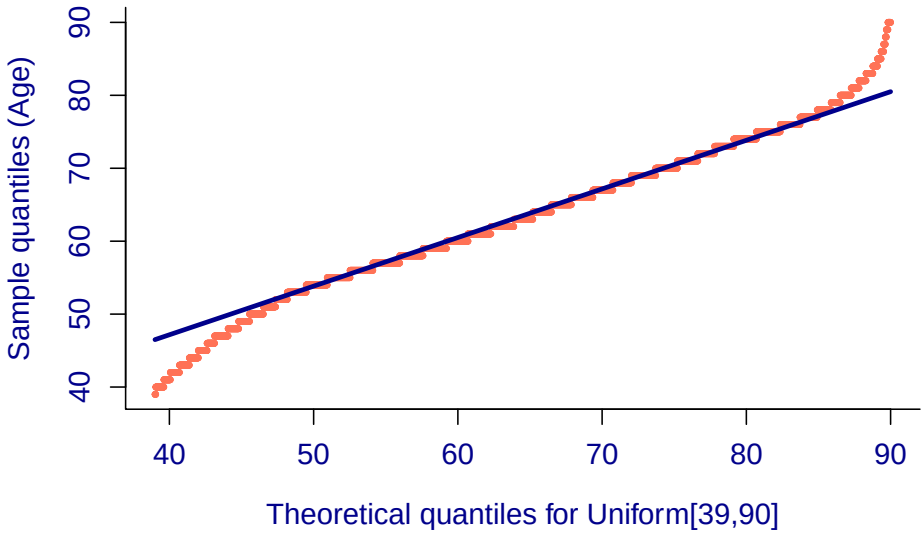


Figure 8.23: QQ-plot for the empirical quantiles of age distribution in SHHS versus the theoretical quantiles of the $U[39, 90]$ distribution.

```
       cex.axis=1.2,col.axis="darkblue",col.main="darkblue",
       cex.lab=1.2,col.lab="darkblue",bty="l")

#Obtain the qqline
y <- quantile(age_s1,c(0.25, 0.75),na.rm=TRUE)
x <- qunif(c(0.25, 0.75),min=39,max=90)
slope <- diff(y) / diff(x)
intercept <- y[1]-slope * x[1]
d=39:90
lines(d,intercept+slope*d,col="darkblue",lwd=3)
```

The shape of the `qqplot` is now an inverted S, which indicates that the empirical distribution of `age_s1` has thicker tails than a Uniform. This requires a little bit more explanation, because the Uniform has a constant density and the density of `age_s1` actually decreases towards age 40 and 90. The 0.25 and 0.75 quantiles of the `age_s1` are (55,72). The only Uniform distribution with these quartiles is the $U[46.5, 80.5]$ distribution, which has 0 probability between $[39, 46.5]$ and between $[80.5, 90]$. Because there are many individuals in the sample who are younger than 46.5 and who are older than 80.5, the tails of the empirical distribution are actually thicker than those of the particular Uniform distribution that matched the quartiles of the empirical distribution.

We would like to check what the QQ-plot should look like when we know that the two distributions are identical. One possibility is to simulate from known distributions. However, here we choose to use sub-sampling from the empirical

distributions of the variables we have observed. Figure 8.24 displays the QQ-plots for the empirical quantiles of three independent subsamples of size 100 from the BMI distribution in SHHS versus the empirical quantiles of the BMI distribution in SHHS.

```
set.seed(913)
bb1<-sample(bmi_s1,100)
bb2<-sample(bmi_s1,100)
bb3<-sample(bmi_s1,100)

qb1<-quantile(bb1,prob=seq(0.01,0.99,by=0.01),na.rm=TRUE)
qb2<-quantile(bb2,prob=seq(0.01,0.99,by=0.01),na.rm=TRUE)
qb3<-quantile(bb3,prob=seq(0.01,0.99,by=0.01),na.rm=TRUE)

qb<-quantile(bmi_s1,prob=seq(0.01,0.99,by=0.01),na.rm=TRUE)

plot(qb,qb1,pch=20,cex=0.7,col="coral1",
     xlab="Quantiles of BMI",
     ylab="Quantiles of BMI subsamples",
     xlim=c(18,47),ylim=c(18,50),
     cex.axis=1.3,col.axis="darkblue",col.main="darkblue",
     cex.lab=1.3,col.lab="darkblue",bty="l")
points(qb,qb2,pch=20,cex=0.7,col="darkgoldenrod3")
points(qb,qb3,pch=20,cex=0.7,col="darkred")
```

The quantiles of each of the subsamples of size 100 from the BMI distribution aligns pretty well with the quantiles of the distribution of BMI in the population, with some extra variability in the high range of BMI values. This happens because extreme quantiles are harder to estimate and each sample contains only 100 of the original 5761 BMI observations that are not missing. The QQ-plots do not perfectly align because of the sampling variability.

We can now conduct an experiment where we simulate three independent samples of size $n = 100$ from a $N(4, 9)$ distribution. Figure 8.25 displays the empirical quantiles of the three samples versus the theoretical quantiles of a $N(0, 1)$ distribution. We found it more flexible to just use the quantiles instead of the `qqnorm` function. Each of the sample quantiles produces a slightly different QQ-plot, though they all visually align pretty close to a straight line. The observed variability (small differences between the lines that best approximate the QQ-plots) is due to sampling variability and closely reflects the variability in estimating the intercept, 4, and slope, 3.

```
rr1<-rnorm(100,4,3)
rr2<-rnorm(100,4,3)
rr3<-rnorm(100,4,3)

qr1<-quantile(rr1,prob=seq(0.01,0.99,by=0.01))
```

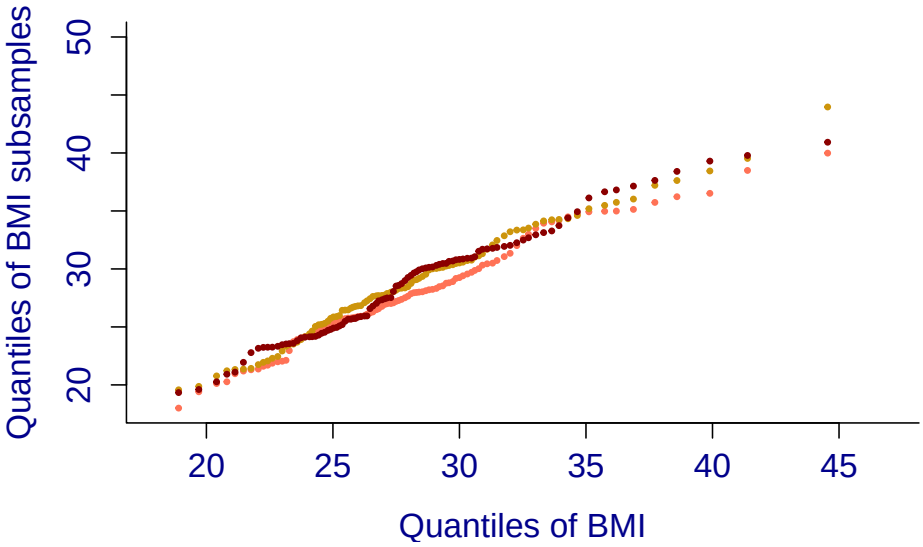


Figure 8.24: QQ-plots for the empirical quantiles of three independent subsamples of size 100 from the BMI distribution in SHHS versus the empirical quantiles of the BMI distribution in SHHS.

```
qr2<-quantile(rr2,prob=seq(0.01,0.99,by=0.01))
qr3<-quantile(rr3,prob=seq(0.01,0.99,by=0.01))

qn<-qnorm(seq(0.01,0.99,by=0.01))
plot(qn,qr1,pch=20,cex=0.7,col="coral1",
     xlab="Theoretical quantiles N(0,1)",
     ylab="Empirical quantiles",
     ylim=c(-3,12),
     cex.axis=1.3,col.axis="darkblue",col.main="darkblue",
     cex.lab=1.3,col.lab="darkblue",bty="l")
points(qn,qr2,pch=20,cex=0.7,col="darkgoldenrod3")
points(qn,qr3,pch=20,cex=0.7,col="darkred")
```

8.8 Heat maps

Heat maps are a type of display of the data in matrix format, where each axis could represent a vector of variables (e.g., the longitude/latitude), or a pair of subjects by vector of variables (e.g. subject ID by BMI, age, and gender). The intensity of the color represents the values taken at that particular pair of variables (e.g., altitude at the given latitude/longitude or correlation between BMI and age). We focus first on displaying the correlation matrix between

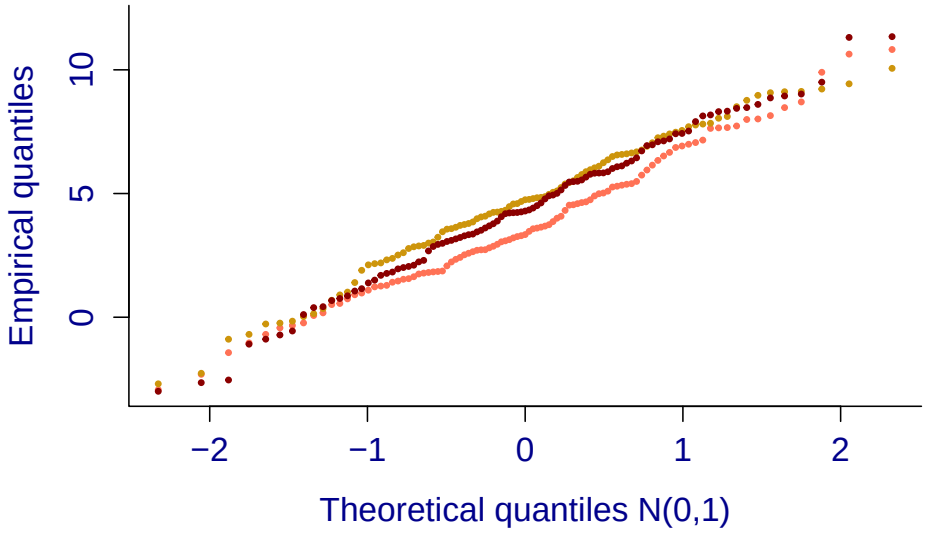


Figure 8.25: QQ-plots for the empirical quantiles of three independent samples of size 100 from the $N(4, 9)$ distribution versus the theoretical quantiles of the $N(4, 9)$ distribution.

several variables. Probably one of the best functions to do this in `R` is the `corrplot` function (Wei 2013). While this may not be classified by some as a heat map, we have found it to be particularly versatile and have particularly strong visualizations. Figure 8.26 displays the correlations for eight variables from the SHHS.

```
library(corrplot)
subset.data.cv=data.cv[,c(2,4,10:11,24,26:27,29)]
M=cor(subset.data.cv,use="pairwise.complete.obs")
corrplot(M, method = "square")
```

The colors on the main diagonal are very dark, indicating that correlations are equal to 1. The other correlations are all relatively small in absolute value, with the strongest association between `waist` and `rdi4p` at 0.33. The strongest negative association is between `timeremp` and `WASO` at -0.21. This should not be surprising in the context of observational studies, where many associations between variables are weak. Having a good representation of the correlation matrix is often useful in applications. Correlation matrix plots become less useful as the number of variables increases.

Figure 8.27 produces a similar plot using the `image.plot` function in the `R` package `fields` (Nychka et al. 2016), even though the function was not originally developed for correlation plots. The advantage of the `image.plot` function over the base `R` functions `image` and `heatmap` is that it produces a legend with the color key. This is quite useful, though a little bit of manipulation is necessary

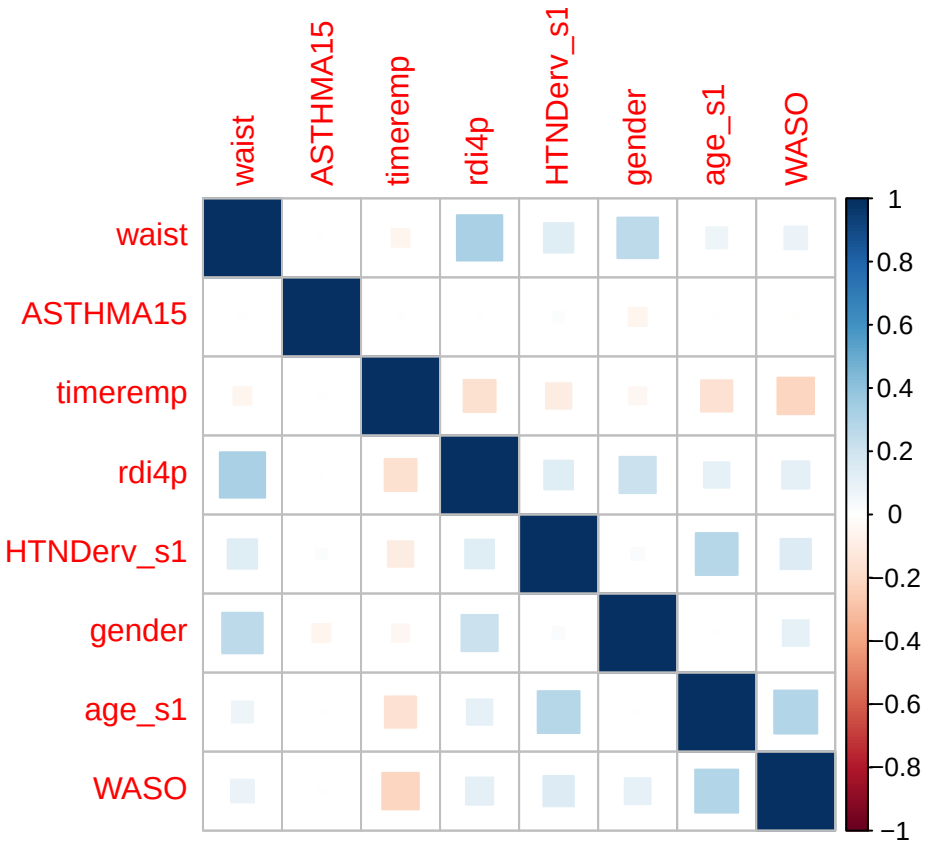


Figure 8.26: Correlation plot for 8 variables in the SHHS.

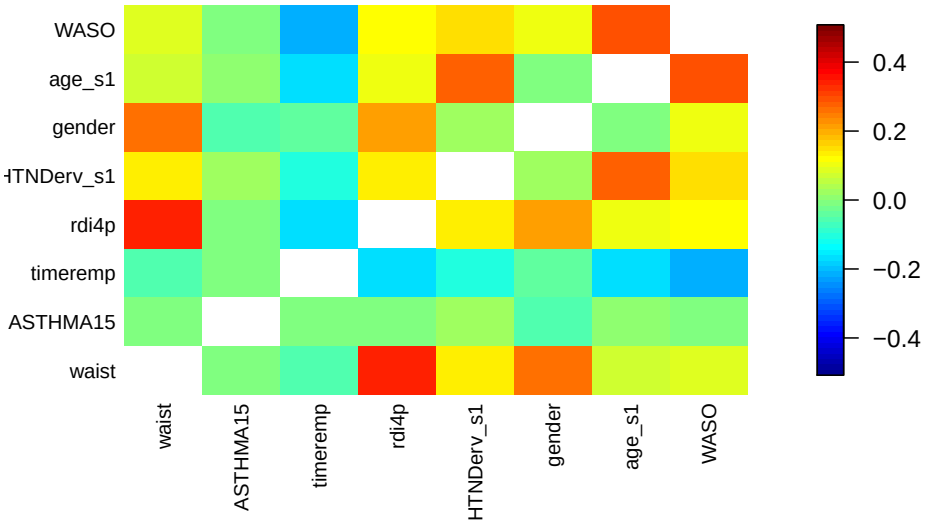


Figure 8.27: Correlation plot for 8 variables in the SHHS using the fields package.

to produce exactly what is needed.

```
library(fields)
par(oma=c(0,0,0,5))
image(M, zlim=c(-.5,.5), col=tim.colors(),axes=FALSE)
mtext(text=rownames(M), side=2, line=0.3,
      at=seq(0,1,length=8), las=1, cex=0.8)
mtext(text=rownames(M), side=1, line=0.3,
      at=seq(0,1,length=8), las=2, cex=0.8)
par(oma=c( 0,0,0,0))# reset margin to be much smaller
image.plot(M, col=tim.colors(),legend.only=TRUE, zlim=c(-.5,.5))
```

In the previous `R` code we used the `R` base `image` function, which allows us to add the text on the sides of the plot. This is quite useful for labeling the variables and connecting the pairs of variables to their corresponding correlations. We also changed the plotting parameters `par(oma=c(0,0,0,5))` to allow for more space to the right of the plot where we add the color key. The function `image.plot` is used here only to create the color key using `legend.only=TRUE`. We have also used a `zlim=c(-0.5,0.5)` both for the image and the legend because we want to avoid the large discrepancy between the autocorrelations on the diagonal, which are equal to 1 and the other correlations, which, in our case, are quite small in absolute value. This particular plot of the correlation does not actually correspond exactly to the `corrplot` in its orientation. Indeed, the matrix `M` needs to be rotated clockwise 90 degrees for the `image` function to produce the same orientation. Figure 8.28 does that and also changes the color palette and

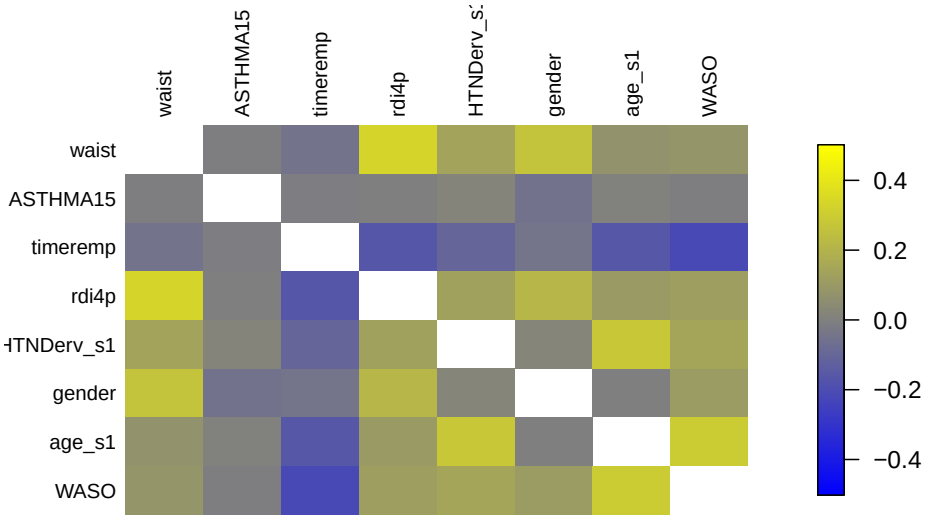


Figure 8.28: Correlation plot for 8 variables in the SHHS using the fields package.

the placement of the text to more closely resemble the `corrplot`.

```
library(fields)
par(oma=c(0,0,0,5))
rotate <- function(x) t(apply(x, 2, rev))
M1=rotate(M)

ColorRamp <- rgb( seq(0,1,length=256),   # Red
                  seq(0,1,length=256),   # Green
                  seq(1,0,length=256))   # Blue

image(M1, zlim=c(-.5,0.5), col=ColorRamp,axes=FALSE)
mtext(text=rev(rownames(M)), side=2, line=0.3,
      at=seq(0,1,length=8), las=1, cex=0.8)
mtext(text=rownames(M), side=3, line=0.3,
      at=seq(0,1,length=8), las=2, cex=0.8)
par(oma=c( 0,0,0,0))# reset margin to be much smaller
image.plot(M1, zlim=c(-.5,0.5),
           col=ColorRamp,legend.only=TRUE, zlim=c(-.5,0.5))
```

We have found the `R` base `heatmap` function to be quite inflexible, especially when one is interested in adding a color key. The `heatmap.2` function in the `gplots` package is designed with far more options and with a standard color key. Figure 8.29 displays the heat map for the correlation using the `heatmap.2` function.

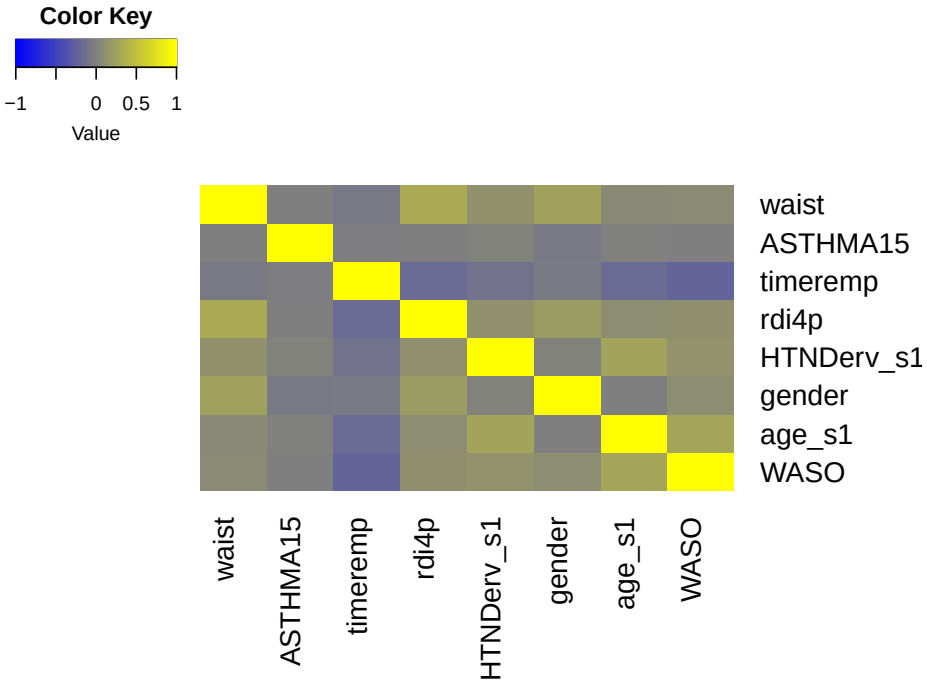


Figure 8.29: Correlation plot for 8 variables in the SHHS using the heatmap.2 function in the gplots package.

```
library(gplots)
heatmap.2(M,dendogram="none",
          Rowv=FALSE,Colv=FALSE,
          density.info="none",
          trace="none",
          col=ColorRamp,
          margins = c(8, 8))
```

The choice of the color key position is slightly different in `heatmap.2` from `corrplot`, probably because of the choice of variable placement. We find the layout of the `corrplot` to be more intuitive, with the close vertical bar to the right of the plot providing better visuals. Also, the `heatmap.2` is designed with the idea of conducting row and column clustering and plotting the dendrograms, which could be useful in certain situations.

Below we show how to use heatmaps implemented using the `image` and `image.plot` functions. First, we should recognize that variables are recorded on many different scales and we would like the visual display to not be dominated by the variables that have the largest and smallest observations. To reduce this effect we suggest plotting only the centered and scaled version

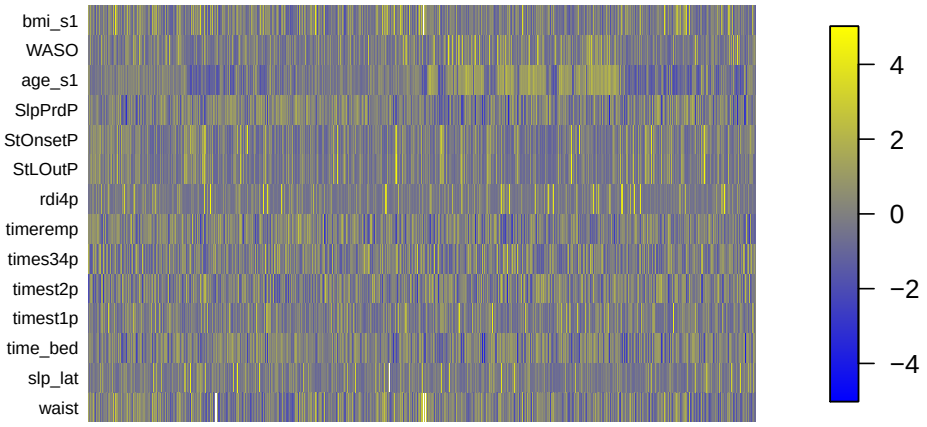


Figure 8.30: Showing the entire SHHS dataset for 14 covariates. Each covariate is shown on a row and each subject on a column.

of the variables. For example, in SHHS, instead of plotting `age_s1` we will plot `(age_s1-63.13)/11.22`, where 63.13 is the average `age_s1` and 11.22 is the standard deviation of `age_s1`. Here we will plot only continuous variables, as plotting 0/1 and categorical variables as part of a heatmap becomes quite difficult. Figure 8.30 displays the data for 5804 subjects (columns) from SHHS using 14 centered and scaled variables. Note that missing in the variable `slp_lat` is indicated in white, which readily identifies variables with more missing observations.

```
#The reordering of the rows is done so that
#first subjects appear to the left of the plot
subset.data.cv=data.cv[5804:1,c(2,5:14,27,29:30)]
subset_scaled<-scale(subset.data.cv)
d<-dim(subset_scaled)
par(oma=c(0,0,0,5))
image(1:d[1],1:d[2],as.matrix(subset_scaled),
      col=ColorRamp,axes=FALSE,
      xlab="",ylab="",zlim=c(-5,5))
mtext(text=colnames(subset.data.cv), side=2, line=0.3,
      at=1:d[2], las=1, cex=0.7)
par(oma=c(0,0,0,0))
image.plot(1:d[1],1:d[2],as.matrix(subset_scaled),
           col=ColorRamp,
           legend.only=TRUE,zlim=c(-5,5))
```

One could prefer to have the subjects on the rows and variables on the column, which is how they are typically analyzed. For that we need only change the x- and y-axes, transpose the matrix, write the text on `side=3`, and change its orientation from horizontal to vertical to avoid over-writing using `las=2`. Figure

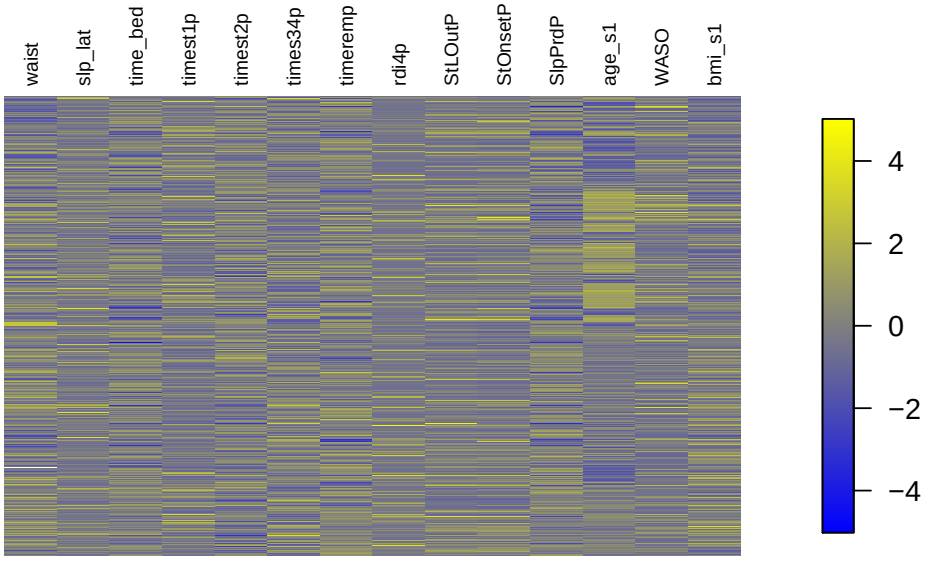


Figure 8.31: Showing the entire SHHS dataset for 14 covariates. Each covariate is shown on a column and each subject on a row.

8.31 displays the same information as Figure 8.30, but the information is now presented with variables shown on columns.

```
par(oma=c(0,0,0,5))
image(1:d[2],1:d[1],as.matrix(t(subset_scaled)),
      col=ColorRamp,axes=FALSE,
      xlab="",ylab="",zlim=c(-5,5))
mtext(text=colnames(subset.data.cv), side=3, line=0.3,
      at=1:d[2], las=2, cex=0.7)
par(oma=c(0,0,0,0))
image.plot(1:d[1],1:d[2],as.matrix(subset_scaled),
           col=ColorRamp,
           legend.only=TRUE,zlim=c(-5,5))
```

In practice we may want to order the data first according to one or more variables and then plot it. This can help identify patterns if the variables used for ordering are particularly well correlated with some of the variables. Below we order by `age_s1` first and by `bmi_s1` second, though ordering could be done for more/fewer variables. Figure 8.32 displays the same information as 8.30 and 8.31, but the rows are reordered first relative to age and then relative to BMI.

```
ordered_subset<-subset_scaled[
  order(subset_scaled[,12],subset_scaled[,14]),]
#This is done to ensure that first subjects in ordered_subset
#appear at the top
```

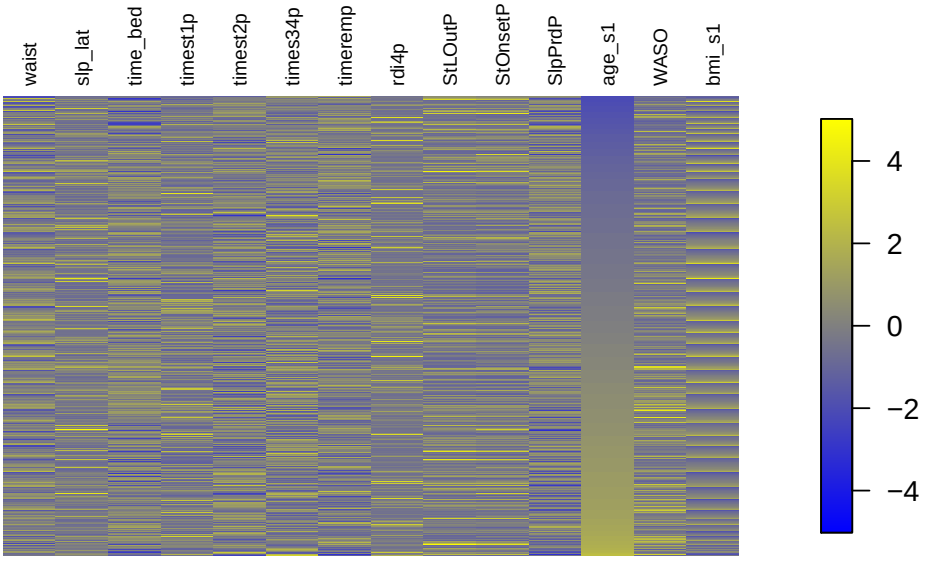


Figure 8.32: Showing the entire SHHS dataset for 14 covariates. Each covariate is shown on a row and each subject on a column. Data are ordered by age first and then by BMI.

```
ordered_subset<-ordered_subset[5804:1,]
par(oma=c(0,0,0,5))
image(1:d[2],1:d[1],as.matrix(t(ordered_subset)),
      col=ColorRamp,axes=FALSE,
      xlab="",ylab="",zlim=c(-5,5))
mtext(text=colnames(subset.data.cv), side=3, line=0.3,
      at=1:d[2], las=2, cex=0.7)
par(oma=c(0,0,0,0))
image.plot(1:d[1],1:d[2],as.matrix(ordered_subset),
           col=ColorRamp,
           legend.only=TRUE,zlim=c(-5,5))
```

However, clustering is often hard to identify for variables that are weakly correlated, as is often the case in health studies. Indeed, after ordering by age there is a clear indication that the corresponding column is now smoothly moving from low to large values. BMI has more of a streaky pattern because for each fixed age group `bmi_s1` increases and then it is reset to the lowest value within the next age category.
With this particular choice of ordering we may note that `rdi4p` follows a similar pattern with `age_s1` with the colors changing slowly from darker (lower) to lighter (higher). One could also notice an unusually low `timeremp` for advanced ages and an increase of `WASO` with `age_s1`. These results provide three impor-

tant take home messages: (1) ordering of variables can provide useful insights into the association between the variable we order on and the other variables; (2) ordering by more than one variable at a time may not produce much additional information; and (3) repeating the procedure with all variables may be helpful, though we suggest doing that with only up to 3-4 variables, as the number of plots can become quickly overwhelming.

8.9 Problems

Problem 1. Load the SHHS dataset and create a histogram and a smoothed histogram of `rdi4p` and `log(rdi4p + 1)`.

Problem 2. Compare `log(rdi4p + 1)` between men and women using histograms, smoothed histograms, dotplots, and boxplots. Discuss the advantages and disadvantages of each.

Problem 3. Create a normal quantile/quantile plot of `rdi4p` and `log(rdi4p+1)`. Does the distribution of `rdi` appear to be normal on either scale? Discuss the nature of any deviations from normality. Does the QQ plot help highlight the discrepancies?

Problem 4. Define healthy sleep as `rid4p < 7`, moderate sleep apnea as `7 <= rdi4p < 15`, sleep apnea as `15 <= rdi4p < 30`, and severe sleep apnea as `30 <= rdi4p`. Compare counts of sleep apnea status versus sex and the BMI groupings used in this chapter using bar plots, stacked bar plots, and mosaic plots.

Problem 5. Consider the `ldeaths`, `mdeaths`, and `fdeaths` data in R. (See `?ldeaths` for more detail). Create heat maps with year as the row and month as the column for all three datasets. Try combining them into a single heat map with useful annotations. Compare the information from the heat maps with time series plots overlaying separate lines. What do the heat maps tell you about these data?

Chapter 9

Approximation results and confidence intervals

This chapter covers the following topics

- Limits
- Law of Large Numbers (LLN)
- Central Limit Theorem (CLT)
- Confidence intervals

We often discuss why it is so hard to understand the principles and results that underlie the foundation of confidence intervals and their interpretation. There are many reasons, but probably the most important ones are: (1) the expectation that once one understands mathematical concepts the related Biostatistical concepts should be trivial; and (2) the intuition built on mathematical concepts do not apply to random variables. Thus, teaching the convergence of random variables and the subsequent use of this theory requires both the professor and the student to start from scratch and accept that this is not just mathematics. Our treatment of stochastic limits attempts to give a deep enough dive so that students will understand the theoretical foundations, while not venturing so deeply into the weeds that the sense of applicability is lost.

9.1 Limits

So, why should one be interested in limits? Why are they so widely used in mathematics? From a mathematical perspective, it is the beauty of a fundamental result and knowing that there is some sort of order in a series of numbers. From a Biostatistical perspective, a limit is an approximation. It simply says that given a very long series of numbers the series can be approximated by a finite

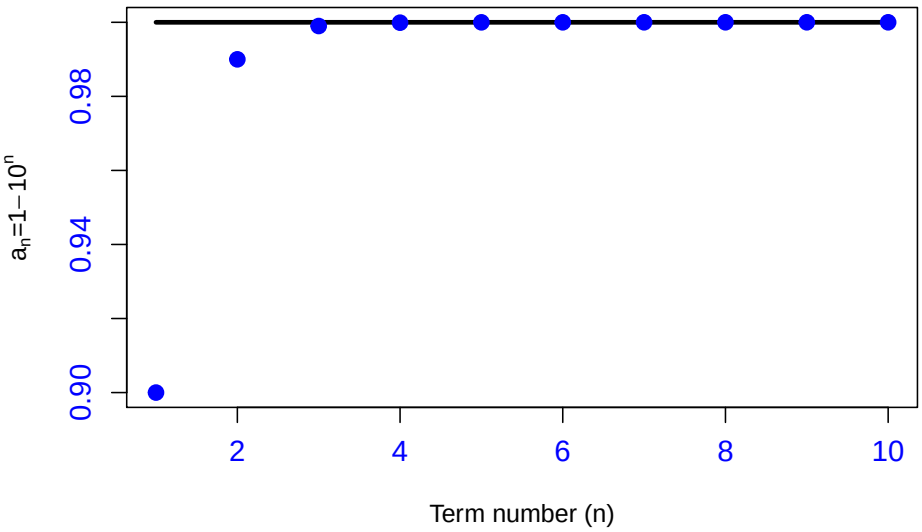


Figure 9.1: Example of a sequence of numbers that converges quickly to 1.

summary of numbers: the finite first observations, that do not have a defined behavior, and the last infinitely many numbers, which are all near the limit. In our view, interpreting limits as approximations is the most useful way to see them. In contrast, there are other uses of limits, such as for proving optimal theoretical results. However, we ignore those uses in this text and focus on limits as approximations, which is their most practical application.

9.1.1 Convergence of number series

Let us revisit the concept of convergent series from calculus. Consider the following sequence: $a_1 = 0.9$, $a_2 = 0.99$, $a_3 = 0.999$, and so on. Clearly, this sequence converges to 1, though no value in the series is exactly 1. Figure 9.1 displays the series on the y-axis as a function of the sequence term number (n) on the x-axis.

```
n=1:10
c=1-10^(-n )
plot(n,rep(1,10),type="l",lwd=3,xlab="Term number (n)",
     ylab=expression(paste(a[n],"=1",-10^n)),
     cex.lab=1,cex.axis=1.2,col.axis="blue",ylim=c(0.9,1))
points(n,c,col="blue",cex=2,pch=20)
```

The first three numbers are visually distinguishable from 1, though the rest of the series can be approximated by 1 for most practical purposes.

In mathematics the definition of a limit of a sequence is that for any fixed

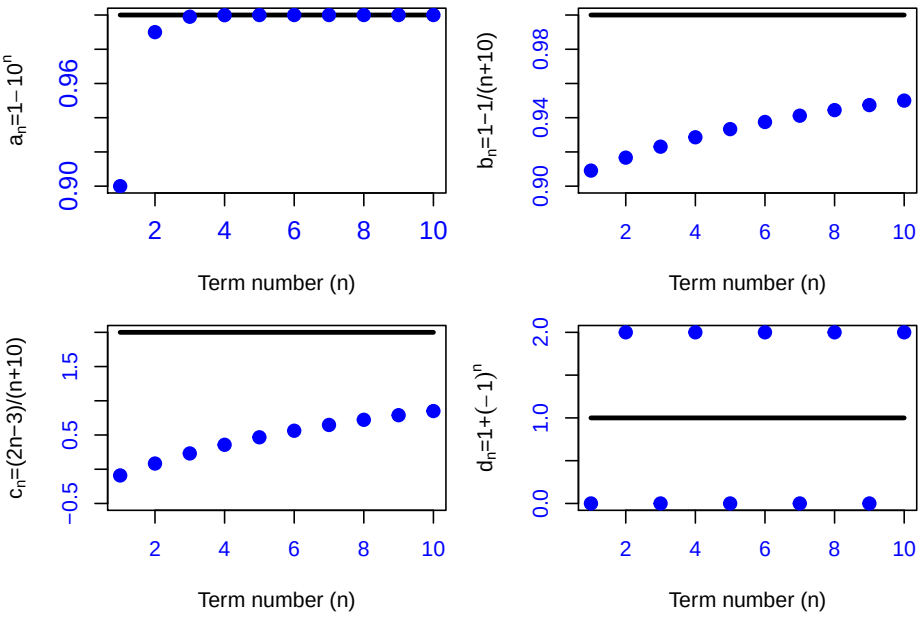


Figure 9.2: Example of a sequence of numbers, some that converge to a limit and one that does not (panel on the second row and column).

distance we can find a point in the sequence so that the sequence is closer to the limit than that distance from that point on. For our sequence

$$|a_n - 1| = 10^{-n} ,$$

indicating that the distance between a_n and 1 decreases exponentially fast to zero. Consider a fixed distance, say ϵ, then for every $n > -\log_{10}(\epsilon)$ we have $-n < \log_{10}(\epsilon)$ and $10^{-n} < \epsilon$. For example, if $\epsilon = 0.00123$ then $-\log_{10}(\epsilon) = 2.91$ indicating that for every $n \geq 3$ we have

$$|a_n - 1| < 0.00123 .$$

However, not all series converge so quickly to their limit and some are not even convergent. Consider, the following three additional series: $b_n = 1 - 1/(n+10)$, $c_n = (2n-3)/(n+10)$, and $d_n = 1 + (-1)^n$. It can be shown that b_n converges to 1, c_n converges to 2 and d_n is not convergent. Figure 9.2 displays all four number sequences as a function of the sequence term number.

The b_n series is shown in the top right panel as blue dots together with its limit shown as a horizontal solid black line at 1. In contrast to a_n, b_n converges much more slowly to its limit. The reason is that the convergence is polynomial not exponential. The third series, c_n, is shown in the bottom left panel and displays a similar behavior to b_n. The consequence is that to effectively characterize both b_n and c_n one needs a lot more numbers than for characterizing a_n. Indeed, one

may need 100 or 1000 numbers at the beginning of the series plus the limit to provide the same level of approximation that four numbers provide for the a_n series. The series d_n shown in the bottom right panel is not convergent, because it has two subsequences, each of each converges to a different number (limit).

9.1.2 Convergence in probability of random variables

The problem is much harder for random variables because sequences of random variables are not sequences of numbers. Consider $\overline{X}_n$ the sample average of the first n of a collection of iid observations. It could be the average number of individuals infected with HIV in a sample of n individuals. We say that $\overline{X}_n$ **converges in probability** to a limit if for any fixed distance the *probability* of $\overline{X}_n$ being closer (further away) than that distance from the limit converges to one (zero). In more precise terms

$$P(|\overline{X}_n - \text{limit}| < \epsilon) \to 1 \ .$$

This definition borrows some characteristics from the convergence of series, but it is fundamentally different. It says that the probability that a random variable is close to the limit as n goes to infinity goes to 1. It does not say that every realization of $\overline{X}_n$ will be between limit $\pm\, \epsilon$ starting from a particular value of n. It also does not say that a particular realization of the sequence of random variables will behave the same way as another realization of a sequence of random variables.

So, what does it say and why is this exotic looking definition useful? Imagine a scenario when one is sequentially sampling individuals and testing them for HIV and consider that the same experiment is conducted by three separate study groups on the same, high-risk, population. They all conduct tests on 10 individuals and compare their results. The first study group identifies three (or 30%) individuals with HIV, the second group identifies four individuals (or 40%), whereas the last research group identifies six (or 60%). They do not have the same subjects, and their data are not identical. However, they all continue to conduct the experiment and we know intuitively that these experiments must share something in common if the data are to ever become useful. Thus, we conclude that the individual data points *are not reproducible* from one experiment to another and the summary statistics (like the number of individuals who are HIV infected) will not be the same either. However, the research groups continue to collect data because they think that their studies are reproducible and their target of inference is the same. Indeed, what stays the same is what is estimated, the limit, and as more data are collected one should expect that the percent number of infected individuals will get closer to the limit for all three research groups. Note that, in contrast to series of numbers, the convergence in probability of random variables requires one to imagine an infinite number of experiments and sequences of experimental results all converging to the same limit. The convergence in probability says that the percent of experiments,

whose results after collecting n data points is between limit $\pm\,\epsilon$, converges to 100% as the number of observations, n, converges to infinity. At this point, for a given n (finite sample sizes), we do not know what percentage of experiments will fall into the interval limit $\pm\,\epsilon$, but the Central Limit Theorem will help us answer that question, as well.

9.2 Law of Large Numbers (LLN)

Establishing that a random sequence converges to a limit is hard. Fortunately, we have a theorem that does all the work for us, called the Law of Large Numbers (LLN). The LLN states that if $X_1, \ldots X_n$ are iid from a population with mean μ and variance σ^2, then $\overline{X}_n$ converges in probability to μ. There are many variations on the LLN; we are using a particularly easy to prove version. Recall that Chebyshev's inequality states that the probability that a random variable is more than k standard deviations away from its mean is less than $1/k^2$. Therefore for the sample mean

$$P\left\{|\overline{X}_n - \mu| \geq k\ \mathrm{sd}(\overline{X}_n)\right\} \leq 1/k^2\ .$$

Pick a distance ϵ and let $k = \epsilon/\mathrm{sd}(\overline{X}_n)$. It follows that

$$P(|\overline{X}_n - \mu| \geq \epsilon) \leq \frac{\mathrm{sd}^2(\overline{X}_n)}{\epsilon^2} = \frac{\sigma^2}{n\epsilon^2}\ ,$$

where the last equality is true because the variance of the sample mean is $\mathrm{Var}(\overline{X}_n) = \sigma^2/n$. This shows that

$$P(|\overline{X}_n - \mu| \geq \epsilon) \to 0\ ,$$

though it shows a lot more than that. Indeed, it shows that the limit is going slowly to zero (at polynomial rate) and that the probability is governed by the amount of dispersion, σ, in finite sample sizes.

9.2.1 Example: estimating the proportion of HIV infections

To further build intuition about the LLN let us revisit the HIV testing experiment and assume that the true infection proportion in the population is 50%. It is easy to simulate and plot the type of data obtained sequentially by three independent study groups.

We generate Bernoulli random variables and calculate the averages $\bar{X}_n$ for various values of n. We do this three times to illustrate that $\bar{X}_n, n \geq 1$ is a series of random variables, that convergence in probability is different from mathematical convergence of number series, and that convergence in probability is a

necessary concept. One could think of the three experiments simulated below as three different experiments run in three different labs. We do not expect that the labs will get the same results in the same order, but we do expect that something is reproducible, in this case the average success rate.

```
suppressWarnings(RNGversion("3.5.0"))
set.seed(14567688)
#Generate the first 100 independent Bernoulli(1,0.5)
x1=rbinom(100,1,0.5)
#Generate the second 100 independent Bernoulli(1,0.5)
x2=rbinom(100,1,0.5)
#Generate the third 100 independent Bernoulli(1,0.5)
x3=rbinom(100,1,0.5)
#Store the sequential means for experiment 1, 2, and 3
xbar1=rep(0,length(x1))
xbar2=xbar1
xbar3=xbar1
#Calculate averages of the observed data
for (i in 1:length(x1))
  {xbar1[i]=mean(x1[1:i])
     xbar2[i]=mean(x2[1:i])
     xbar3[i]=mean(x3[1:i])}

plot(1:100,xbar1-0.5,type="l",col=rgb(1,0,0,alpha=0.5),lwd=3,
     xlab="Number of subjects sampled",ylab="Distance to the mean",
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",ylim=c(-0.5,0.5))
lines(1:100,xbar2-0.5,col=rgb(0,0,1,alpha=0.5),lwd=3)
lines(1:100,xbar3-0.5,col=rgb(1,0,1,alpha=0.5),lwd=3)
lines(1:100,rep(0,100),lwd=3)
```

Figure 9.3 displays $\overline{X}_n - 0.5$ for $n = 1, \ldots, 100$ for each experiment separately (shown in different colors). The first experiment is shown in red and $\overline{X}_{11} = 1$ because the first individual was HIV positive. Because we plot $\overline{X}_{11} - 0.5 = 0.5$ the value shown in the plot is 0.5. We use the notation $\overline{X}_{nk}$, where n indicates over how many subjects the average is calculated and k indicates which of the three experiments we are referring to. The second individual was HIV negative, so $\overline{X}_{21} = (1+0)/2 = 0.5$ and because we are plotting $\overline{X}_{21} - 0.5$ this value is represented as 0 in the plot. The third subject was identified as HIV positive, so $\overline{X}_{31} = (1+0+1)/3 = 0.66$ and on the plot we display $\overline{X}_{31} - 0.5 = 0.16$. Every time the red line goes up it does so because a new HIV case was identified. Every time the line goes down corresponds to an HIV negative individual. Let us investigate the second experiment (shown as the blue solid line). In this experiment the first individual was HIV negative and $\overline{X}_{21} = 0$ and because we display $\overline{X}_{21} - 0.5 = -0.5$ the first y-value for the second experiment is -0.5. The second individual was also HIV negative, so $\overline{X}_{22} = (0+0)/2 = 0$ and the value shown is $\overline{X}_{22} - 0.5 = -0.5$, while the third individual was HIV positive, which

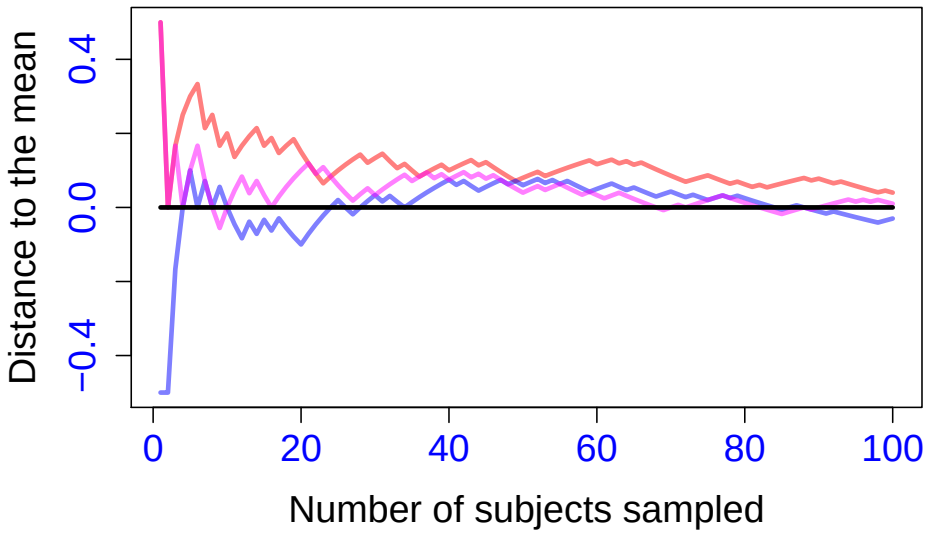


Figure 9.3: Examples of the percent number of infections obtained by three different studies when data are collected sequentially and the probability of infection for every individual in the population is 50%.

resulted in $\overline{X}_{23} = (0+0+1)/3 = 0.33$ and a plotted value of $\overline{X}_{23} - 0.5 = -0.17$ and so on.

These experiments show the characteristics of what happens if multiple experiments are run in parallel: (1) the observed data are not the same; (2) the observed means are not the same; (3) as the number of observations increases, the variability between the observed means decreases; (4) all means get closer to the target; and (5) the means are not equal to the target of inference.

Though in this case data were simulated, this type of behavior is intended to closely mimic what is seen in practical experiments. Thus, this raises the question about what exactly is meant by reproducibility and generalizability of an experiment. Indeed, all three experiments were run in exactly the same way, but resulted in different data and different summary statistics. However, the level of organization of the means of experiments as the number of subjects increases seems to suggest that something is reproducible. That something is the underlying true probability of infection, which remains unchanged, irrespective of the experiment number, and the observed sampling variability. Under the definition of convergence in probability, the fact that the means converge could be interpreted as "for a large enough sample size, if we run many experiments about the same target of inference most of them will produce estimates of the target of inference that are close to it."

The law of large numbers will establish that as n increases the averages are close to the target, while the central limit theorem will say how close and with

what probability the results of the experiment are to the true target. Of course, in practice one usually conducts only one experiment and we do not have the luxury of even two let alone an infinite number of experiments. So, the secret sauce is to make all these inferences from one experiment, generalize the results to a population of interest, and provide reasonable estimates for what other experiments would produce if they were conducted. Stated like this, it should become apparent that the problem we are trying to solve is truly daunting. Luckily the practical solution is simple and very easy to implement.

9.2.2 Convergence of transformed data

If $X_1, \ldots, X_n$ are iid random variables so that $E\{f(X)\}$ exists then under mild conditions (e.g. $E\{f^2(X)\}$) exists and is finite) then

$$\frac{1}{n}\sum_{i=1}^{n} f(X_i) \to E\{f(X)\} \; .$$

For example,

$$\frac{1}{n}\sum_{i=1}^{n} X_i^2 \underset{n}{\longrightarrow} E[X^2] \; ,$$

$$\frac{1}{n}\sum_{i=1}^{n} \exp(X_i) \underset{n}{\longrightarrow} E\{\exp(X)\} \; ,$$

and

$$\frac{1}{n}\sum_{i=1}^{n} \log(X_i) \underset{n}{\longrightarrow} E\{\log(X)\} \; ,$$

where the last result is for positive random variables. Moreover, functions of convergent random sequences converge to the function evaluated at the limit. These include sums, products, and differences. For example, $\overline{X}_n^2$ converges to μ^2. Note that this is different from $(\sum X_i^2)/n$, which converges to $E(X_i^2) = \sigma^2 + \mu^2$.

9.2.3 Convergence of the sample variance

Recall that the sample variance is

$$S_n^2 = \frac{\sum_{i=1}^{n}(X_i - \overline{X}_n)^2}{n-1} = \frac{\sum_{i=1}^{n}(X_i^2 - 2X_i\overline{X}_n + \overline{X}_n^2)}{n-1} \; .$$

Because

$$\sum_{i=1}^{n} X_i\overline{X}_n = \overline{X}_n \sum_{i=1}^{n} X_i = n\overline{X}_n^2 \; ,$$

where the last equality holds from the definition of $\overline{X}_n$, it follows that

$$S_n^2 = \frac{\sum_{i=1}^n X_i^2 - 2n\overline{X}_n^2 + n\overline{X}_n^2}{n-1} = \frac{n}{n-1}\left\{\frac{1}{n}\sum_{i=1}^n X_i^2 - \overline{X}_n^2\right\} .$$

Obviously, $n/(n-1) \to 1$ when n converges to infinity, $\sum_{i=1}^n X_i^2/n \to \sigma^2 + \mu^2$, and $\overline{X}_n^2 \to \mu^2$. Therefore

$$S_n^2 \to 1 \times (\sigma^2 + \mu^2 - \mu^2) = \sigma^2 .$$

It follows that $S_n \to \sigma$, that is, the empirical standard deviation converges to the theoretical standard deviation.

9.2.4 Unbiased and convergent estimators

It is important to understand the difference between unbiased and convergent estimators. Both are desirable properties, but they represent different characteristics of statistics. We have seen both definitions before, but it is useful to see them again, side by side. First, recall that every function of the data is called a statistic, or an estimator. If $\mathbf{X}$ are the data and θ is a parameter of interest, then we say that an estimator $\widehat{\theta}(\mathbf{X})$ is unbiased for θ if $E\{\widehat{\theta}(\mathbf{X})\} = \theta$. Note that for unbiased estimators we do not need a sequence of estimators, we just need the data. In contrast, if $\mathbf{X}_n$ is a vector of data that increases with n and $\widehat{\theta}_n(X)$ is a sequence of estimators that depend on n, then we say that $\widehat{\theta}_n(X)$ converges to θ if it converges in probability to θ. In practice, many sequences of estimators are both unbiased and convergent to the true target. For example, the sample mean is unbiased for the mean, $E(\overline{X}_n) = \mu$, and converges to the target, $X_n \to \mu$ in probability. However, this is not the case for all sequences.

Indeed, consider the sequence of iid random variables $X_1, \ldots, X_n$ with mean μ. Then X_n is unbiased for μ, $E(X_n) = \mu$, but it does not converge in probability to μ because X_n is a random variable. This provides a sequence of estimators that are unbiased but not convergent. However, $\overline{X}_n + 1/n$ is a biased estimator for the mean, $E(\overline{X}_n + 1/n) = \mu + 1/n$, with a bias equal to $1/n$, but converges in probability to μ. This is an example of a sequence of estimators that are biased but convergent.

The LLN basically states that the sample mean is consistent and we also showed that the sample variance and the sample standard deviation are consistent, as well. Recall that the sample mean and the sample variance are unbiased, though the sample standard deviation is biased.

9.3 Central Limit Theorem (CLT)

The **Central Limit Theorem** (CLT) is one of the most important theorems in statistics. For our purposes, the CLT states that the distribution of averages

of iid variables, properly normalized, becomes that of a standard Normal as the sample size increases. Just like any other limit results, the applicability of this result is that for a large enough sample size a potentially very complicated distribution can be very well approximated by a Normal. The CLT applies in an endless variety of settings.

9.3.1 Convergence in distribution of random variables

Consider a sequence of rvs $X_n, n \geq 1$. We say that X_n converges in distribution to X if

$$P(X_n \leq x) = F_n(x) \underset{n}{\longrightarrow} F(x) = P(X \leq x) \ ,$$

for every x. This is sometimes referred to as the *weak convergence of random variables*. This is useful in practice because cdfs are used to calculate probabilities, the cdfs $F_n(x)$ may be intractable or difficult to work with, and $F(x)$ is simple and provides a good approximation for $F_n(x)$ starting from a large enough n. The weak convergence of random variables is, in fact, the pointwise convergence of functions from calculus, where the functions are the cdfs of the random variables. That is why biostatisticians call it "weak." Consider, again, a sequence of iid variables $X_1, \ldots, X_n$ and note that, by definition the cdfs of this random variables are all equal $F_{X_1}(x) = \ldots = F_{X_n}(x)$ for every x and for every n. Thus, the sequence converges weakly to either random variable in the sequence, say X_1. However, the sequence is not convergent in probability. It can be shown that convergence in probability of a series of random variables implies convergence in distribution, but we won't be providing the proof here, as it is fairly technical. (Students who are interested can build on this text with more advanced work in probability theory.)

9.3.2 CLT formulation

Given $X_1, X_2, \ldots$ a sequence of iid random variables with mean μ and variance σ^2 then

$$\lim_{n\to\infty} P\left(\frac{\bar{X}_n - \mu}{\sigma/\sqrt{n}} \leq z\right) = \Phi(z) \ ,$$

for every real z, where $\Phi(z)$ is the cdf of a $N(0,1)$ distribution. This approximation is called the Central Limit Theorem (CLT) and it is often used to approximate the distribution of $\bar{X}_n$ when the number of samples, n, is relatively large and the distribution of $\bar{X}_n$ is complicated. The random variable

$$Z_n = \frac{\bar{X}_n - \mu}{\sigma/\sqrt{n}} = \frac{\sqrt{n}(\bar{X}_n - \mu)}{\sigma}$$

is referred to as the z-score of the mean random variable $\bar{X}_n$.

The random variable $Z_n = \frac{\bar{X}_n - \mu}{\sigma/\sqrt{n}}$ is a transformation of the sample average $\bar{X}_n$ that has mean zero (centered) and variance one (standardized). This type of transformation is widely spread in science to transform measurements to the same scale (note that Z_n is unitless) where they can be compared. The transformation is sometimes referred to as z-scoring. The CLT states that Z_n converges in distribution to $Z \sim N(0,1)$. The CLT essentially says that, as n increases, the distribution of the z-score Z_n becomes indistinguishable from the $N(0,1)$ distribution. This can be used to construct confidence intervals for the true mean of the distribution.

In general, z-scoring means that we subtract the mean and divide by the standard deviation of the random variable. In this case the random variable is $\overline{X}_n$ whose mean is μ and standard deviation is $\sigma/\sqrt{n}$. In practice, z-scoring is not limited to means and it can be applied to any random variable.

9.3.3 Example: the exponential distribution

We conduct some simulations in R to provide a better idea about exactly how CLT is working. Consider an independent random sample $X_1, X_2, \ldots$ from an exp(1) distribution. The mean of this distribution is $\mu = 1$ and the variance is also $\sigma^2 = 1$. Thus the standardized mean is $Z_n = (\bar{X}_n - 1)/(1/\sqrt{n}) = \sqrt{n}(\bar{X}_n - 1)$. We would like to see how the distribution of the mean of three and thirty realizations from such a distribution behave. Below we simulate 1000 times independent random variables from an exp(1). For each simulation we simulate two vectors of mutually independent random variables, one of length 3 and one of length 30. For each vector we calculate its mean, resulting in two vectors of length 1000: one of means of three and one of means of thirty independent random variables with exp(1) distribution.

```
#Set a grid to evaluate the exp(1) distribution
xh=seq(0,5,length=101)
#Calculate the pdf of the exp(1) distribution
he=dexp(xh,rate=1)

#Set the two sample sizes, $n=3$ and $n=30$
n=c(3,30)
#Matrix for storing the 1000 means for each n
mx=matrix(rep(0,2000),ncol=2)
for (i in 1:1000)
    {#begin simulations
    #Calculate the mean of 3 independent exp(1)
    mx[i,1]=mean(rexp(n[1], rate = 1))
    #Calculate the mean of 30 independent exp(1)
    mx[i,2]=mean(rexp(n[2], rate = 1))
    }#end simulations
```

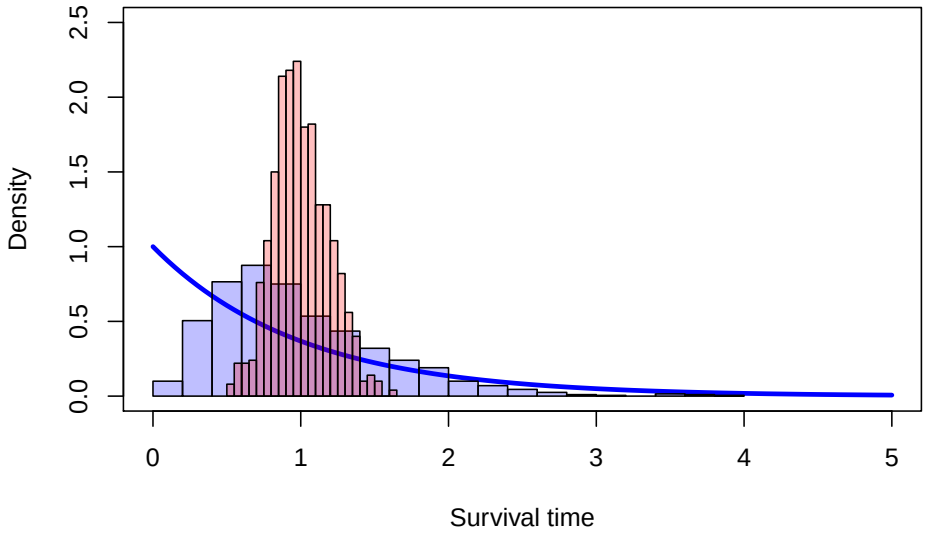


Figure 9.4: Histograms of 1000 means of three (light blue) and thirty (light red) independent random variables with exp(1) distribution. The original pdf of the exp(1) distribution is shown as a solid blue line.

Figure 9.4 displays the histograms of these two 1000-dimensional vectors. The histohram corresponding to the mean of three random variables with an exp(1) distribution is shown in light blue, while the one corresponding to the mean of thirty random variables with an exp(1) distribution is shown in light blue. Here we used transparent colors to overplot histograms and `add=T` to add new features to existent plots. The original pdf of the exp(1) distribution is shown as a solid blue line.

```
plot(xh,he,type="l",col="blue",lwd=3,ylim=c(0,2.5),
     xlab="Survival time",ylab="Density")
hist(mx[,1],prob=T,add=T,col=rgb(0,0,1,1/4),breaks=25)
hist(mx[,2],prob=T,add=T,col=rgb(1,0,0,1/4),breaks=25)
```

The distribution of the mean looks closer to being bell-shaped than the shape of the original exp(1) distribution (shown as blue line), which is highly skewed. For $n = 3$ the distribution of the means starts to have a shape closer to the Normal than to the original exponential and for $n = 30$ the distribution of the means is quite symmetric and relatively bell-shaped. This illustrates that convergence in this case is quite rapid, even though the shape of the original distribution was relatively far from a Normal.

Instead of the means we could have plotted the z-scores directly. Below we show the distribution of the z-scores relative to the theoretical density of a $N(0, 1)$ distribution. Figure 9.5 displays these two histograms together with the pdf of

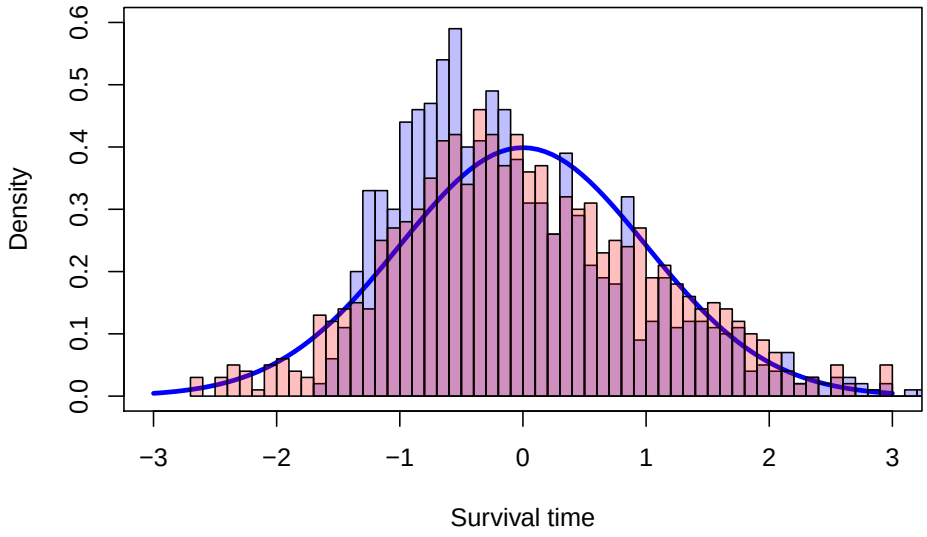


Figure 9.5: Histograms of 1000 z-scored means of three (light blue) and thirty (light red) independent random variables with exp(1) distribution. The $N(0,1)$ distribution is shown as a solid blue line.

the $N(0,1)$ distribution indicating just how close these distributions are even after taking the mean of a few iid variables. Again, just like limits are used to approximate series of numbers the Normal distribution (solid blue line) is used to approximate the distribution of means of random variables (the light red and blue histograms).

```
#Set a grid to evaluate the N(0,1) distribution
xx=seq(-3,3,length=101)
#Evaluate the pdf of the N(0,1) distribution
yx=dnorm(xx)

#Matrix to store the z-scores
zx=mx
for (j in 1:2)
    #Calculate the z-scores for the means
    {zx[,j]<-sqrt(n[j])*(mx[,j]-1)}

plot(xx,yx,type="l",col="blue",lwd=3,ylim=c(0,0.6),
     xlab="Survival time",ylab="Density")
hist(zx[,1],prob=T,add=T,col=rgb(0,0,1,1/4),breaks=50)
hist(zx[,2],prob=T,add=T,col=rgb(1,0,0,1/4),breaks=50)
```

9.3.4 Example: Bernoulli distribution

We now show the CLT for coin flipping using different types of coins and numbers of flips. Assume that we have n independent Bernoulli(p) trials $X_1, \ldots, X_n$. The estimated probability of success $\widehat{p}_n = \bar{X}_n$ can take values $0/n, 1/n, \ldots, n/n$ with probability

$$P\left(\widehat{p}_n = \frac{k}{n}\right) = \binom{n}{k} p^k (1-p)^{n-k} .$$

This is true because

$$P\left(\widehat{p}_n = \frac{k}{n}\right) = P(n\widehat{p}_n = k) = P\left(\sum_{i=1}^{n} X_i = k\right) ,$$

and $\sum_{i=1}^{n} X_i$ is a Binomial random variable with probability of success p out of n trials. Since $E[\widehat{p}_n] = p$ and $\text{Var}[\widehat{p}_n] = \frac{p(1-p)}{n}$ it follows that the z-score for $\widehat{p}_n$ is

$$Z_n = \frac{\widehat{p}_n - p}{\sqrt{p(1-p)/n}} = \frac{\sqrt{n}(\widehat{p}_n - p)}{\sqrt{p(1-p)}} .$$

Given the values that can be taken by $\widehat{p}_n$, Z_n takes the values $\frac{\sqrt{n}(k/n-p)}{\sqrt{p(1-p)}}$ with probability $\binom{n}{k} p^k (1-p)^{n-k}$. The values that Z_n can take cover the entire real line with a dense grid as $n \to \infty$. This case is particularly interesting, because the Z_n is a discrete variable that can take $n+1$ values. The CLT indicates that the discrete distribution of Z_n is well approximated by the continuous $N(0,1)$ distribution.

Figure 9.6 displays the exact distribution of the Z_n random variable for different values of n and p. Clockwise, panels show the distribution for $n = 5$, 10, 20, and 50. In each panel we display four distributions, one corresponding to the $N(0,1)$ distribution (blue), and three corresponding to the z-score of the Binomial distribution with different values of p: $p = 0.5$ (red), $p = 0.3$ (orange), and $p = 0.1$ (violet). The panels indicate just how fast the distribution of Z_n is approximated by a $N(0,1)$. Indeed, even for $n = 10$ and $p = 0.5$ or 0.3 the pmfs of Z_n are hard to distinguish from the $N(0,1)$. It takes a little longer for the approximation to work very well for $p = 0.1$ because the distribution of Z_n is more skewed and it takes longer for the variable to even take values below -2. However, by $n = 50$ all approximations are excellent. This is one of the possible illustrations of the CLT and should provide a clear idea about the quality of the approximation of the Binomial distribution by the Normal. A reasonable rule of thumb for the approximation to work well is that $np \geq 10$, though these plots indicate that for the range of values considered here the rule of thumb could be relaxed to $np \geq 5$.

```
#Set the value of the number of samples
nv=c(5,10,20,50)
#Set the parameters for plotting multiple panels
```

```
par(mfrow = c(2, 2),mai = c(0.4, 0.7, 0.2, 0.2))
#Set the grid for the Normal distribution
xx=seq(-3,3,length=101)
yx=dnorm(xx)

for (l in 1:length(nv))
    {#begin loop over sample sizes
    n=nv[l]
    #Set the value of the sample size
    plot(xx,yx,type="l",col="blue",lwd=3,xlab="Z-score",
         ylab=paste("n=",n),cex.lab=1.5,cex.axis=1.5,
         col.axis="blue",ylim=c(0,0.45))

    #Values taken by the sum of independent Bernoulli random variables
    k=0:n
    #Set the probability vector
    p=c(0.5,0.3,0.1)
    #Matrix of values that can be taken by the z-scores
    values=matrix(rep(0,(n+1)*3),ncol=3)
    #Matrix of probabilities for the z-scores
    pr=values
    for (i in 1:length(p))
          {#begin looping over probabilities
         #Calculate the z-store values
         values[,i]=sqrt(n)*(k/n-p[i])/sqrt(p[i]*(1-p[i]))
         #Calculate the z-score probabilities
           pr[,i]=dbinom(k,n,prob=p[i])/(values[2,i]-values[1,i])
        }#end looping over probabilities

lines(values[,1],pr[,1],lwd=3,col="red")
lines(values[,2],pr[,2],lwd=3,col="orange")
lines(values[,3],pr[,3],lwd=3,col="violet")
    }#end loop over sample sizes
```

9.4 Confidence intervals

One of the most important applications of the CLT in practice is to construct confidence intervals. Indeed,

$$P\left(\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} \leq z\right) \approx \Phi(z)\,.$$

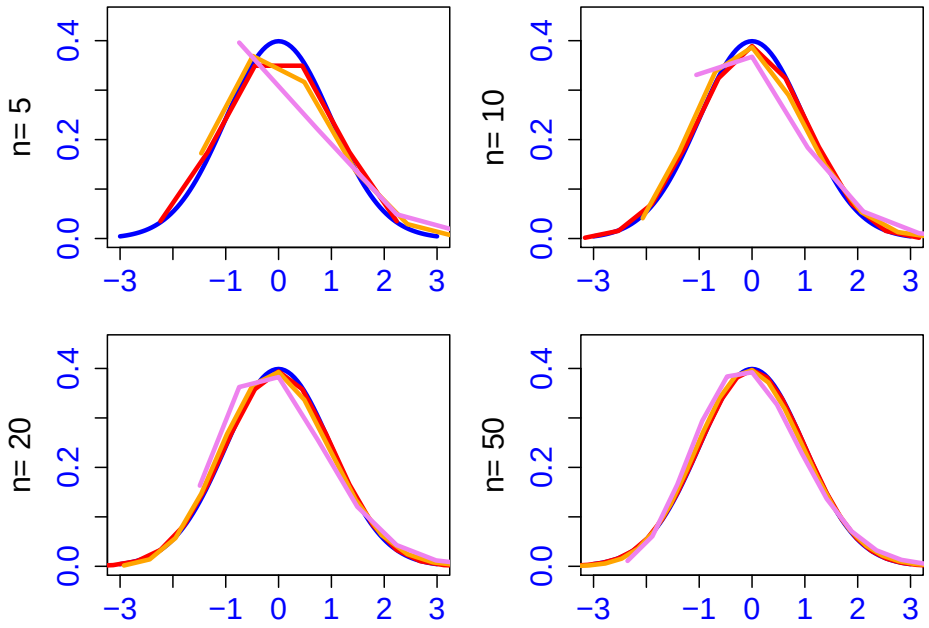


Figure 9.6: Exact distributions of the z-scored Binomial(n, p) for different values of the success probability, p: $p = 0.5$ (red), $p = 0.3$ (orange), and $p = 0.1$ (violet). Each panel corresponds to a different number of trials, n, corresponding clockwise to $n = 5$, 10, 20, and 50. The pdf of the $N(0, 1)$ distribution is shown in blue.

Recall that 1.96 is a good approximation to the 0.975 quantile of the standard Normal indicating that

$$.95 \approx P\left(-1.96 \leq \frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} \leq 1.96\right) = P\left(\overline{X}_n - 1.96\frac{\sigma}{\sqrt{n}} \leq \mu \leq \overline{X}_n + 1.96\frac{\sigma}{\sqrt{n}}\right) .$$

The last equality indicates that the random interval $\overline{X}_n \pm 1.96\sigma/\sqrt{n}$ contains the true parameter μ, with probability close to 0.95. This is called the 95% confidence interval for the true value of the parameter, μ. The approximation holds for any distribution of the original data and it is better with increased sample size. In terms of interpretation, recall that we interpret a random variable as the outcome of the experiment before the experiment is being run. Therefore, the interpretation of the confidence interval is that in repeated experiments 95% of the confidence intervals constructed from the observed data will contain the true parameter. For any individual experiment there is no guarantee whether or not the calculated confidence interval covers the true parameter. In fact, it makes no sense to calculate the confidence interval based on the data and assign a probability of covering the true value of the parameter. Indeed, after running the experiment the realization of the confidence interval is purely deterministic and it either covers or does not cover the true value of the parameter, which is also fixed.

Both in Statistics and sciences there is criticism of the frequentist interpretation of confidence intervals. It is worth mentioning that both alternatives, Bayesian and likelihood/evidence based, produce practically the same results and they differ only in their interpretation of these identical results. We consider that the criticism is unwarranted as the interpretation simply requires one to understand the difference between a random variable and its realization in an experiment. If one is not comfortable with that we suggest revisiting earlier chapters of the book or switching to a book that does not require understanding of the fundamental principles in biostatistics. We repeat that we did not promise this would be easy, just that we would do our best to explain hard to understand concepts.

9.4.1 Slutsky's theorem

While this confidence interval is extremely useful, its utility is quite limited by the fact that, in practice, we do not know what σ is. Therefore, we would like to replace σ by the empirical standard deviation, S_n. The reason is that once the data are collected we can replace σ by the realization of S_n, s_n. Slutsky's theorem allows us to do that.

Slutsky's theorem states that if X_n and Y_n are random sequences, such that X_n converges in distribution to X and Y_n converges in probability to a constant c then

a. $X_n + Y_n$ converges in distribution to $X + c$

b. X_nY_n converges in distribution to Xc
c. X_n/Y_n converges in probability to X/c

We know already that $Z_n = \sqrt{n}(\overline{X}_n - \mu)/\sigma$ converges in distribution to $Z \sim N(0,1)$ and that S_n/σ converges in probability to 1. It follows that $Z_n/(S_n/\sigma) = \sigma Z_n/S_n$ converges in distribution to Z. Therefore,

$$\sigma Z_n/S_n = \frac{\overline{X}_n - \mu}{S_n/\sqrt{n}}$$

converges weakly to Z. Using the same exact technique used above it follows that $\overline{X}_n \pm 1.96 S_n/\sqrt{n}$ is an approximate 95% confidence interval for the mean μ. This interval is more practical than the one obtained directly from CLT because it does not contain unknown parameters.

9.4.2 Example: sample proportions

In the event that each X_i is 0 or 1 with common success probability p, then $\sigma^2 = p(1-p)$. The $100(1-\alpha)\%$ confidence interval for the true mean p takes the form

$$\widehat{p} \pm z_{1-\alpha/2}\sqrt{\frac{p(1-p)}{n}} \ .$$

However, since p is unknown this confidence interval cannot be calculated once data are collected. Replacing p by $\widehat{p}$ in the standard error results in what is called a Wald confidence interval for p

$$\widehat{p} \pm z_{1-\alpha/2}\sqrt{\frac{\widehat{p}(1-\widehat{p})}{n}} \ .$$

It can also be shown that $p(1-p) \leq 1/4$ for $0 \leq p \leq 1$, which shows that

$$\widehat{p} \pm \frac{z_{1-\alpha/2}}{2\sqrt{n}} \ ,$$

is an asymptotically conservative $100(1-\alpha)\%$ confidence interval. Here asymptotically conservative means that for large enough n the probability that this interval contains the true mean is at least $1-\alpha$. In the case when $\alpha = 0.05$ $z_{1-\alpha/2} = 1.96$ and $z_{1-\alpha/2}/2 \approx 1$ the interval becomes

$$\widehat{p} \pm \frac{1}{\sqrt{n}} \ .$$

Now, let us revisit our HIV testing example and inspect what happens with the confidence intervals as a function of number of subjects sampled. Figure 9.7 displays a realization of $\bar{X}_n - 0.5$ (red solid line), where $\bar{X}_n$ is the mean of the first n independent Bernoulli(0.5) random variables. The shaded area around the red solid line are the 95% Wald confidence intervals for $\mu - 0.5$ instead of μ to keep the plotting consistent.

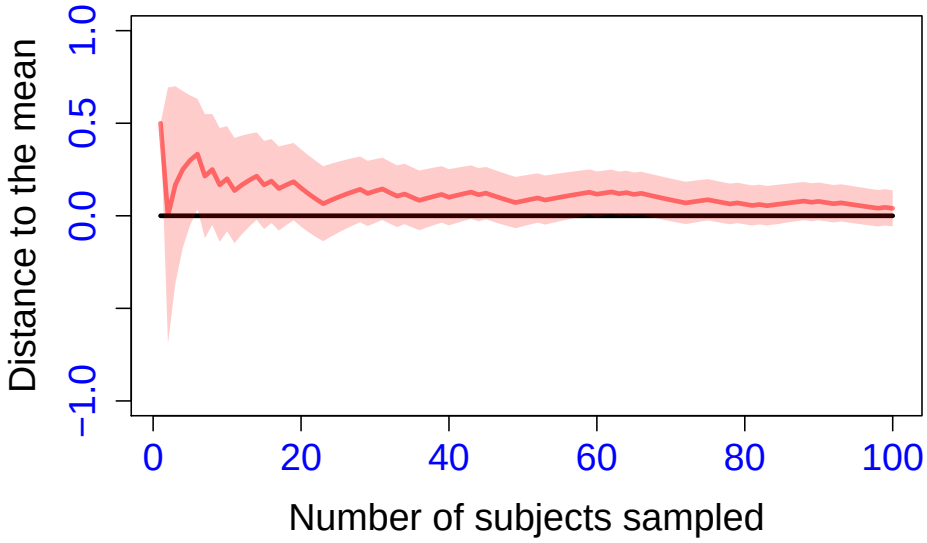


Figure 9.7: A realization of $\bar{X}_n - 0.5$ (red solid line), where $\bar{X}_n$ is the mean of the first n independent Bernoulli(0.5) random variables together with the 95% confidence intervals as a function of the number of individuals sampled, n.

```
plot(1:100,xbar1-0.5,type="l",col=rgb(1,0,0,alpha=0.5),lwd=3,
     xlab="Number of subjects sampled",ylab="Distance to the mean",
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",ylim=c(-1,1))
n=1:100
lower_limit=pmax(xbar1-0.5-1.96*sqrt(xbar1*(1-xbar1)/n),-1)
upper_limit=pmin(xbar1-0.5+1.96*sqrt(xbar1*(1-xbar1)/n),1)
lines(1:100,rep(0,100),lwd=3)
xn<-c(n,n[100:1])
yn<-c(lower_limit,upper_limit[100:1])
polygon(xn,yn,col=rgb(1,0,0,0.2),border=NA)
```

It is worth noting that the Wald confidence interval has a very strange behavior for $n = 1$ because $\widehat{p}_1(1 - \widehat{p}_1) = 0$. Indeed, in this case the length of the confidence interval is 0. The interval $\widehat{p} \pm 1/\sqrt{n}$ does not behave much better, as the interval is always $[0, 1]$. Indeed, there is not much that one can do in this situation, though an interval of the form $\widehat{p} \pm \sqrt{0.5(1 - 0.5)} = \widehat{p} \pm 0.5$ will not have the exact coverage probability, but at least it will be less silly than the previous two confidence intervals. For $n = 1$ we suggest not to construct confidence intervals. For $n > 1$ the confidence intervals make more sense and they tend to cover 0 (recall that we are focusing on $\widehat{p} - 0.5$, which is an estimator of 0). We would like to know exactly what confidence intervals do not cover 0. This can only happen if the `lower_limit` vector is above 0 or the `upper_limit` is below 0. The `R` code below identifies the exact sample sizes for which the

confidence intervals do not cover 0.

```
index_no_covering<-((lower_limit>0) | (upper_limit<0))
n[index_no_covering]
```

```
[1]  1  6 59 61 62 64 66
```

Clearly $n = 1$ should not be a surprise, as the confidence interval has length 0 and the case can simply be excluded. When $n = 6$ the confidence interval also does not cover zero, which is visible in the plot. This happened because the first 6 observations were

```
x1[1:6]
```

```
[1] 1 0 1 1 1 1
```

indicating that 5 out of the first 6 individuals tested for HIV were HIV positive. This number is unusualy high if the true proportion of individuals who are HIV positive in the target high risk population is 0.5. As the number of samples continues to increase, the estimated proportion converges to the target, though the convergence is a little slow. There is an additional cluster of sample sizes, 59, 61, 62, 64, and 66, for which the confidence interval does not cover 0, though the lower limit is very close to 0 for all these intervals. Again, this happened due to an unusually long run of HIV positive patients who were sequentially identified. The fact that 6 out of 99 confidence intervals do not cover 0 should not be surprising, as these are 95% confidence intervals. However, this interpretation is not perfect here, as the confidence intervals are highly correlated. Indeed, if the confidence interval for one n does not contain 0 then the confidence intervals for its neighbors are more likely not to contain 0. This is precisely what happened with the cluster of sample sizes that produced confidence intervals that do not cover 0. The correct interpretation is that at $n = 59$ this study was one of the unlucky ones, but if we continue to conduct additional studies then 95% of these studies will produce confidence intervals that cover 0 at $n = 59$.

Another important characteristic is that the confidence intervals decrease in length as the sample size increase. This should not be surprising, as the length of the confidence intervals is $\approx 4\sqrt{\widehat{p}_n(1-\widehat{p}_n)/n}$, which decreases at the rate of $\sqrt{n}$. As we discussed earlier, the CLT, which provides the basis for calculating the confidence intervals, controls how far the realization of the estimators should be from the true target. This is a general characteristic of confidence intervals, indicating that the length of confidence intervals decreases slowly with the sample size. For example, one needs four times as much data to obtain a confidence interval that has half the length.

Yet another important characteristic of confidence intervals is that they are random, that is, they depend on the specific data that one acquires. Figure 9.8 presents results for another experiment with the same exact characteristics as the one used in Figure 9.7. The confidence intervals for the second study look quite different from the ones from the first experiment.

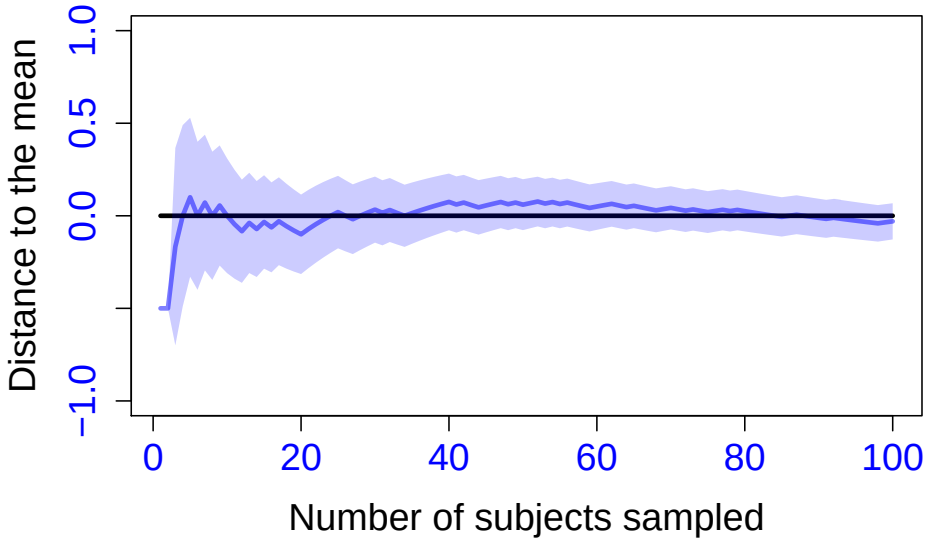


Figure 9.8: A realization of $\bar{X}_n - 0.5$ (blue solid line), where $\bar{X}_n$ is the mean of the first n independent Bernoulli(0.5) random variables together with the 95% confidence intervals as a function of the number of individuals sampled, n.

This happens because the data are different, the point estimators are different, and the length of the confidence intervals is different, though the last is not obvious by visual inspection of the plots. Note that in this study both for $n = 6$ and $n = 59$ the confidence intervals cover 0.

Biostatisticians have not invented confidence intervals and their interpretation to make students suffer. Instead, they were built to: (1) capture the natural variability that appears in data collected by different studies even when the study protocols and data collection protocols are identical; (2) provide generalizable information to the population and about other studies that could be conducted by only conducting one study; and (3) quantify how close or far from the target of estimation one can be in finite samples, which are the norm in realistic scenarios.

In this chapter we have learned about LLN and CLT, which, in spite of the assumptions, have withstood the test of time, billions of simulation scenarios, and scientific applications. One still has to be careful with the implied properties in finite, small samples, but the power of these results is truly enormous. We would like to finish with a simple example about the power of these results.

Consider that you have a truly large dataset with billions of subjects and tens of thousands of variables. Even storing such a dataset is daunting, not to mention analyzing it in any depth. The results here suggest a simple workaround. Define a parameter of interest, subsample 100000 subjects from the population, run a quick analysis, and adjust the length of the confidence intervals from $n = 100000$

to the true N in the population. This simple trick can save enormous amounts of time, resources, and debate about best algorithms for handling big data. Beating subsampling in the era of big data is hard.

9.5 Problems

Problem 1. Suppose that 400 observations are drawn at random from a distribution with mean 0 and standard deviation 40.

a. What is the approximate probability of getting a sample mean larger than 3.5?
b. Was normality of the underlying distribution required for this calculation?

Problem 2. Recall that `R` function `runif` generates (by default) random uniform variables that have means 1/2 and variance 1/12.

a. Sample 1000 observations from this distribution. Take the sample mean and sample variance. What numbers should these estimate and why?
b. Retain the same 1000 observations from part a. Plot the sequential sample means by observation number. Hint: if `x` is a vector containing the simulated uniforms, then the code

```
y <- cumsum(x) / (1 : length(x))
```

will create a vector of the sequential sample means. Explain the resulting plot.

c. Plot a histogram of the 1000 numbers. Does it look like a uniform density?
d. Now sample 1000 *sample means* from this distribution, each calculated for 100 observations. What numbers should the average and variance of these 1000 numbers be equal to and why? Hint: the command

```
x <- matrix(runif(1000 * 100), nrow = 1000)
```

creates a matrix of size 1000×100 filled with random uniforms. The command

```
y<-apply(x,1,mean)
```

takes the sample mean of each row.

e. Plot a histogram of the 1000 sample means appropriately normalized. What does it look like and why?
f. Now obtain 1000 *sample variances* from this distribution, each based on 100 observations. Take the average of these 1000 variances. What property does this illustrate and why?

Problem 3. Note that `R` function `rexp` generates random exponential variables. The exponential distribution with rate 1 (the default) has a theoretical mean of 1 and variance of 1.

a. Sample 1000 observations from this distribution. Take the sample mean and sample variance. What numbers should these estimate and why?
b. Retain the same 1000 observations from part a. Plot the sequential sample means by observation number. Explain the resulting plot.
c. Plot a histogram of the 1000 numbers. Does it look like an exponential density?
d. Now sample 1000 *sample means* from this distribution, each based on 100 samples from the exponential distribution. What numbers should the average and variance of these 1000 numbers approximate and why?
e. Plot a histogram of the 1000 sample means appropriately normalized. What does it look like and why?
f. Now sample 1000 *sample variances* from this distribution, each comprised of 100 observations. Take the average of these 1000 variances. What property does this illustrate and why?

Problem 4. Consider the distribution of a fair coin flip (i.e. a random variable that takes the values 0 and 1 with probability 1/2 each.

a. Sample 1000 observations from this distribution. Take the sample mean and sample variance. What numbers should these estimate and why?
b. Retain the same 1000 observations from part a. Plot the sequential sample means by observation number. Explain the resulting plot.
c. Plot a histogram of the 1000 numbers. Does it look like it places equal probability on 0 and 1?
d. Now sample 1000 *sample means* from this distribution, each based on 100 observations. What numbers should the average and variance of these 1000 numbers be equal to and why?
e. Plot a histogram of the 1000 sample means appropriately normalized. What does it look like and why?
f. Now sample 1000 *sample variances* from this distribution, each based on 100 observations. Take the average of these 1000 variances. What property does this illustrate and why?

Problem 5. Consider a density for the proportion of a person's body that is covered in freckles, X, given by $f(x) = cx$ for $0 \leq x \leq 1$ and some constant c.

a. What value of c makes this function a valid density?
b. What is the mean and variance of this density?
c. You simulated 100000 sample means, each based on 100 draws from this density. You then took the variance of those 100000 numbers. Approximately what number did you obtain?

Problem 6. You need to calculate the probability that a *standard Normal* is larger than 2.20, but have nothing available other than a regular coin. Describe how you could estimate this probability using only your coin. (Do not actually carry out the experiment, just describe how you would do it.)

Problem 7. The density for the population of increases in wages for assistant professors being promoted to associates (1 = no increase, 2 = salary has doubled)

is uniform on the range from 1 to 2.

a. What is the mean and variance of this density?
b. Suppose that the sample variance of 10 observations from this density was sampled, say 10000 times. What number would we expect the average value from these 10000 variances to be near? Briefly explain your answer briefly.

Problem 8. Suppose that the US intelligence quotients (IQs) are normally distributed with mean 100 and standard deviation 16.

a. What IQ score represents the 5^{th} percentile? Explain your calculation.
b. Consider the previous question. Note that 116 is the 84^{th} percentile from this distribution. Suppose now that 1000 subjects are drawn at random from this population. Use the central limit theorem to write the probability that less than 82% of the sample has an IQ below 116 as a standard Normal probability. Note, you do not need to solve for the final number. Show your work.
c. Consider the previous two questions. Suppose now that a sample of 100 subjects is drawn from a *new* population and that 60 of the sampled subjects had an IQ below 116. Give a 95% confidence interval estimate of the true probability of drawing a subject from this population with an IQ below 116. Does this proportion appear to be different than the 84% for the population from questions 1 and 2?

Problem 9. You desperately need to simulate standard Normal random variables yet do not have a computer available. You do, however, have 10 standard six-sided dice. Knowing that the mean of a single die roll is 3.5 and the standard deviation is 1.71, describe how you could use the dice to approximately simulate standard Normal random variables. Be precise.

Problem 10. Consider three sample variances, S_1^2, S_2^2 and S_3^2. Suppose that the sample variances obtained from iid samples of size n_1, n_2 and n_3 from Normal populations $N(\mu_1, \sigma^2)$, $N(\mu_2, \sigma^2)$ and $N(\mu_3, \sigma^2)$, respectively. Argue that

$$\frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2 + (n_3 - 1)S_3^2}{n_1 + n_2 + n_3 - 3}$$

is an unbiased estimate of σ^2.

Problem 11. You need to calculate the probability that a normally distributed random variable is less than 1.25 standard deviations below the mean. However, you only have an oddly shaped coin with a known probability of heads of 0.6. Describe how you could estimate this probability using this coin. (Do not actually carry out the experiment, just describe how you would do it.)

Problem 12. The next three questions (a., b., c.) deal with the following setting. Forced expiratory volume, FEV_1, is a measure of lung function that is often expressed as a proportion of lung capacity called forced vital capacity,

FVC. Suppose that the population distribution of FEV_1/FVC of asthmatic adults in the US has mean of 0.55 and standard deviation of 0.10.

a. Suppose that a random sample of 100 people is drawn from this population. What is the probability that their average FEV_1/FVC is larger than 0.565?
b. Suppose that the population of non-asthmatic adults in the US have a mean FEV_1/FVC of 0.8 and a standard deviation of 0.05. You sample 100 people from the asthmatic population and 100 people from the non-asthmatic population and take the difference in sample means. You repeat this process 10000 times to obtain 10000 differences in sample means. What would you guess the mean and standard deviation of these 10000 numbers would be?
c. Moderate or severe lung dysfunction is defined as $FEV_1/FVC \leq 0.40$. A colleague tells you that 60% of asthmatics in the US have moderate or severe lung dysfunction. To verify this you take a random sample of five subjects, only one of which has moderate or severe lung dysfunction. What is the probability of obtaining only one or fewer if your friend's assertion is correct? What does your result suggest about your friend's assertion?

Problem 13. A special study is conducted to test the hypothesis that persons with glaucoma have higher blood pressure than average. Two hundred subjects with glaucoma are recruited with a sample mean systolic blood pressure of $140mm$ and a sample standard deviation of $25mm$. (Do not use a computer for this problem.)

a. Construct a 95% confidence interval for the mean systolic blood pressure among persons with glaucoma. Do you need to assume normality? Explain.
b. If the average systolic blood pressure for persons without glaucoma of comparable age is $130mm$. Is there statistical evidence that the blood pressure is elevated?

Problem 14. Consider the previous question. Make a probabilistic argument that the interval

$$\left[\bar{X} - z_{.95}\frac{s}{\sqrt{n}}, \quad \infty\right]$$

is a 95% *lower bound* for μ.

Problem 15. Suppose that we wish to estimate the concentration μg/ml of a specific dose of ampicillin in urine. We recruit 25 volunteers and find that they have a sample mean concentration of 7.0 μg/ml with sample standard deviation 3.0 μg/ml. Let us assume that the underlying population distribution of concentrations is normally distributed.

a. Find a 90% confidence interval for the population mean concentration.
b. How large a sample would be needed to insure that the length of the confidence interval is 0.5 μg/ml if it is assumed that the sample standard

deviation remains at 3.0 μg/ml?

Chapter 10

The χ^2 and t distributions

This chapter covers the following topics

- The χ^2 distribution
- Confidence intervals for the variance of a Normal
- Student's t distribution
- Confidence intervals for Normal means

Previously, we discussed creating a confidence interval using the CLT. That confidence interval ended up using the empirical standard deviation instead of the often unknown true standard deviation. Slutsky's theorem allowed us to do this replacement, though the price paid is that the approximation holds only asymptotically. In many applications, the number of samples is small to moderate, which may raise questions about how good this asymptotic approximation actually is. Now we discuss the creation of better confidence intervals for small samples using Gosset's t distribution, which is the finite sample distribution for the t-statistic when the observations are independent Normal. For completeness, if $X_1, \ldots, X_n$ are iid random variables then the t-statistic is defined as

$$t_n = \frac{\overline{X}_n - \mu}{S_n/\sqrt{n}} \, .$$

Here $\overline{X}_n$ and S_n^2 are the empirical mean and variance of the observations, respectively.

To discuss the t distribution we must discuss the χ^2 (Chi-squared or, phonetically, kai-squared) distribution. Throughout we use the following general procedure for creating confidence intervals: (a) create a **pivot**: a function of data and parameters whose distribution does not depend on the parameter of interest; (b) calculate the probability that the pivot lies in a particular interval; and (c) re-express the confidence interval in terms of (random) bounds on the parameter of interest.

10.1 The χ^2 distribution

If $X_1, \ldots, X_n$ are independent $N(0,1)$ random variables then

$$V_n = \sum_{i=1}^{n} X_i^2$$

has a Chi-squared distribution with n degrees of freedom and we denote $V_n \sim \chi^2_n$. The Chi-squared distribution is skewed and has support $(0, \infty)$. We will show that the mean of the Chi-squared is its degrees of freedom and its variance is twice the degrees of freedom. That is, $E(V_n) = n$ and $\text{Var}(V_n) = 2n$.

We first derive the distribution of a χ^2_1 random variable. If $X \sim N(0,1)$ then

$$V = X^2 \sim \chi^2_1 \, .$$

Denote by $\Phi(x) = P(X \leq x)$ the cdf of the standard Normal distribution. If $F_V(v) = P(V \leq v)$ denotes the cdf of the χ^2_1 random variable, then we have

$$F_V(v) = P(X^2 \leq v) = P(-\sqrt{v} \leq X \leq \sqrt{v}) = \Phi(\sqrt{v}) - \Phi(-\sqrt{v}) = 2\Phi(\sqrt{v}) - 1 \, .$$

The second equality holds because $X^2 \leq v$ if and only if $-\sqrt{v} \leq X \leq \sqrt{v}$, the third equality holds because $[-\infty, \sqrt{v}] = (-\infty, -\sqrt{v}) \cup [-\sqrt{v}, \sqrt{v}]$ and $(-\infty, -\sqrt{v}) \cap [-\sqrt{v}, \sqrt{v}] = \emptyset$, and the fourth inequality holds because of the symmetry of the Normal distribution, which ensures that $\Phi(-\sqrt{v}) = 1 - \Phi(\sqrt{v})$. Recall that the pdf of a $N(0,1)$ is

$$\phi(x) = \Phi'(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \, .$$

Therefore, the pdf of the χ^2_1 distribution is

$$f_V(v) = F'_V(v) = \frac{\partial}{\partial v}\{2\Phi(\sqrt{v}) - 1\} = 2\frac{\partial}{\partial v}(\sqrt{v})\Phi'(\sqrt{v}) \, .$$

Because $\partial/\partial v(\sqrt{v}) = 1/(2\sqrt{v})$ and

$$\Phi'(\sqrt{v}) = \frac{1}{\sqrt{2\pi}} e^{-\frac{v}{2}}$$

it follows that

$$f_V(v) = 2\frac{1}{2\sqrt{v}}\frac{1}{\sqrt{2\pi}} e^{-\frac{v}{2}} = \frac{1}{\sqrt{2\pi}} v^{1/2-1} e^{-\frac{v}{2}} \, .$$

Recall that a Gamma(α, β) distribution has the pdf

$$f_\Gamma(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-x\beta} \, ,$$

where $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1}e^{-x}dx$ for every $\alpha > 0$. Inspecting the two distributions we can see that the χ^2_1 distribution is a Gamma$(1/2, 1/2)$ distribution. Indeed, both pdfs have to integrate to 1, which should make the two Normalizing constants equal, as well. That is

$$\frac{1/2^{1/2}}{\Gamma(1/2)} = \frac{1}{\sqrt{2\pi}}$$

indicating that $\Gamma(1/2) = \sqrt{\pi}$. This result can be obtained directly by computations, as well, but here we show a particularly simple solution. We know that the mean of a Gamma(α, β) distribution is α/β and the variance is α/β^2. Therefore $E(V) = 0.5/0.5 = 1$ and $\text{Var}(V) = 0.5/0.5^2 = 2$. Hence, for $V_n \sim \chi^2_n$ we have

$$E(V_n) = \sum_{i=1}^{n} E(X_i^2) = n$$

and

$$E(V_n) = \sum_{i=1}^{n} \text{Var}(X_i^2) = 2n \ .$$

It can be shown that

$$V_n \sim \text{Gamma}\left(\frac{n}{2}, \frac{1}{2}\right) \ ,$$

though the proof exceeds the level of detail and does not meet the importance criteria we set for proofs in this book. This result can be shown as a particular case of the result that if $X_1, \ldots, X_n$ are independent Gamma(α, β) random variables then

$$\sum_{i=1}^{n} X_i \sim \text{Gamma}(n\alpha, \beta) \ .$$

10.1.1 Pdfs of χ^2 distributions

Figure 10.1 displays the pdf for four different χ^2 distributions. The χ^2_1 pdf is displayed as a blue solid line indicating that the distribution is heavily skewed with $\lim_{v\downarrow 0} f_V(v) = \infty$. While the pdf has an infinite limit at 0 it does integrate to 1. The pdf of the χ^2_2 is $f_V(v) = 0.5 \exp -x/2$, which is the exponential distribution with mean equal to 2. This distribution is shown in red, has a limit $\lim_{v\downarrow 0} f_V(v) = 0.5$ and, just like all other distributions, has a limit equal to 0 at infinity. The pdfs both for the χ^2_1 and χ^2_2 distributions are decreasing on the domain $(0, \infty)$. The pdfs of the χ^2_3 (violet) and χ^2_4 (orange) distributions have the property $f_V(v) = 0$, which holds true for any value of the number of degrees of freedom larger than 2. Both the χ^2_3 and χ^2_4 distributions are heavily skewed with heavy right tails (tails that decrease slower than the tails of a Normal distribution).

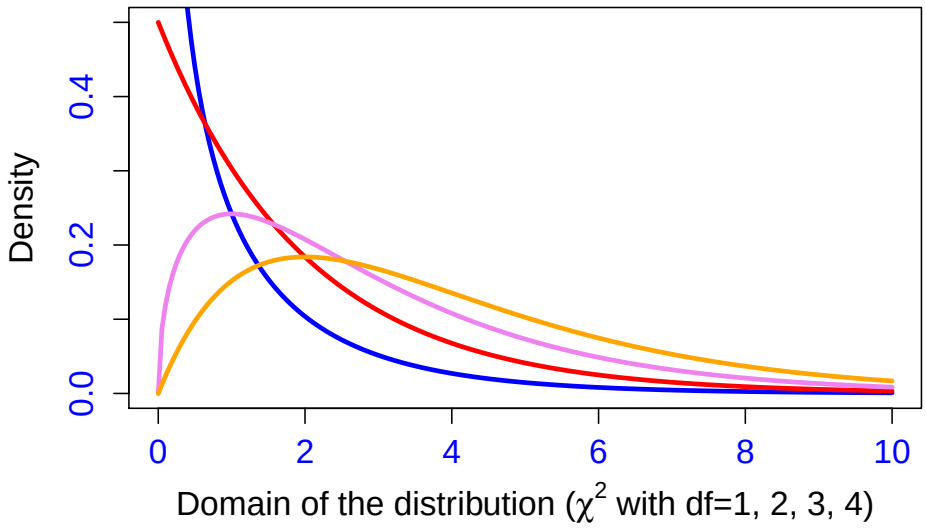


Figure 10.1: Pdfs of the χ^2_1 (blue), χ^2_2 (red), χ^2_3 (violet) and χ^2_4 (orange) distributions.

As the number of degrees of freedom, n, of the χ^2_n distribution increases, the distribution becomes more symmetric and can be very well approximated by a Normal$(n, 2n)$. This result follows immediately from the CLT applied to the variables $X_1^2, \ldots, X_n^2$, which are iid because $X_1, \ldots, X_n$ are iid. With this notation

$$\frac{V_n}{n} = \frac{\sum_{i=1}^n X_i^2}{n} ,$$

and, according to the CLT, the distribution of

$$Z_n = \frac{V_n/n - \mu}{\sigma/\sqrt{n}}$$

can be approximated by the $N(0,1)$ distribution for large enough n, where $\mu = E(X_1^2) = 1$ and $\sigma^2 = \text{Var}(X_1^2) = 2$. Replacing these values in the formula above we obtain that the distribution of

$$Z_n = \frac{V_n/n - 1}{\sqrt{2/n}} = \frac{V_n - n}{\sqrt{2n}}$$

can be approximated by a $N(0,1)$. From the properties of the Normal distribution it follows that the distribution of V_n can be approximated by a $N(n.2n)$ distribution. Because we know the exact distribution of V_n we can obtain the distribution of Z_n and compare it to that of a $N(0,1)$. Indeed if $F_{Z_n}(\cdot)$ and $F_{V_n}(\cdot)$ are the cdfs of the Z_n and V_n random variables, respectively, then we have

$$F_{Z_n}(z) = P\left(\frac{V_n - n}{\sqrt{2n}} \leq z\right) = P(V_n \leq n + z\sqrt{2n}) = F_{V_n}(n + z\sqrt{2n}) .$$

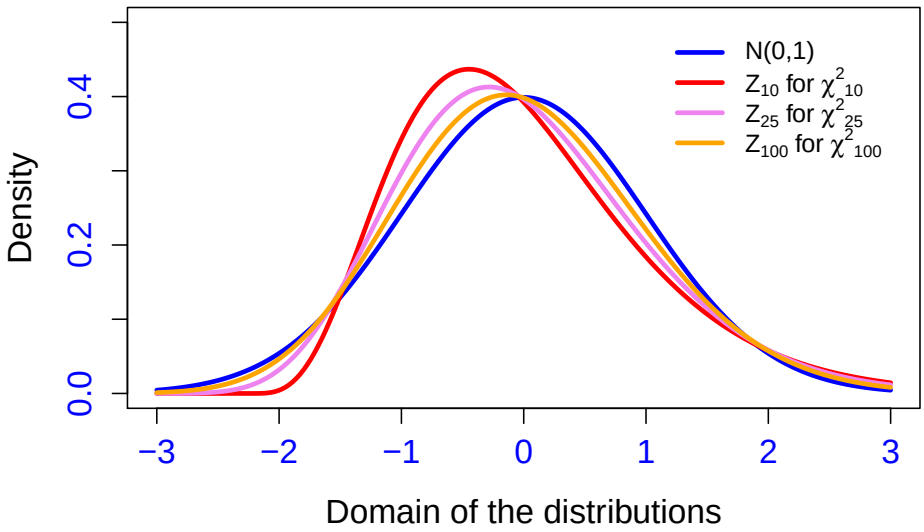


Figure 10.2: Pdfs of the Z-scores for the χ^2_{10} (red), χ^2_{25} (violet), and χ^2_{100} (orange) compared with the pdf of a $N(0,1)$ (blue).

Similarly, if $f_{Z_n}(\cdot)$ and $f_{V_n}(\cdot)$ are the pdfs of the random variables Z_n and V_n, respectively, then

$$f_{Z_n}(z) = F'_{Z_n}(z) = \frac{\partial}{\partial z} F_{V_n}(n + z\sqrt{2n}) = \sqrt{2n} f_{V_n}(n + z\sqrt{2n}) \, .$$

Of course, we could obtain an explicit formula for $f_{Z_n}(z)$ by plugging the corresponding values into the function $f_{V_n}(\cdot)$, which is the Gamma$(n/2, 1/2)$ density. Instead, we simply use this formula to calculate numerically the pdf of Z_n and plot it against the $N(0,1)$ density.

Figure 10.1 displays the $N(0,1)$ pdf in blue and the pdfs of the Z_n random variables for $n = 10$ (red), $n = 25$ (violet), and $n = 100$ (orange) indicating the improved approximation by the $N(0,1)$ random variable as n increases. By $n = 100$ the distribution of Z_n is almost indistinguishable from the $N(0,1)$, though it is interesting to observe the tilt of the Z_n distributions. They all have their mode to the left of 0 due to the skew of the Gamma distributions. In the `R` code above we hard coded the exact pdfs of the Z_n random variables obtained from the location-scale transformations of the Gamma distribution.

10.1.2 The χ^2 distribution in `R`

The day of the books of statistical tables is over. They have been replaced by computers, which can provide the information faster, with fewer errors, and allow us to avoid carrying two-pound books containing tables of numbers. That's

the good news. However, in order to obtain what is needed in practice we need to understand exactly what we are looking for and how to obtain it. Suppose, for example, that we are interested in obtaining an interval that contains 95% of the probability for the χ^2_4 distribution. There are many such intervals, but here we will focus on the one interval that leaves out equal-tail probabilities. In R this is simple:

```
#Set the number of degrees of freedom
df=4
#Set the probability that will be left out
alpha=0.05
#Specify that we need the interval with equal tail probability
interval<-c(alpha/2,1-alpha/2)
#Obtain the equal-tail probability interval
round(qchisq(interval,df),digits=2)
```

```
[1]  0.48 11.14
```

10.2 Confidence intervals for the variance of a Normal

Suppose that S_n^2 is the sample variance from a collection of iid $N(\mu, \sigma^2)$ data. We will show that

$$\frac{(n-1)S_n^2}{\sigma^2} \sim \chi^2_{n-1} \, ,$$

and we will use this result to construct confidence intervals for the true variance of $N(\mu, \sigma^2)$ distribution. Note that $(X_i - \mu)/\sigma \sim N(0, 1)$ and are independent indicating that

$$\frac{n}{\sigma^2} \sum_{i=1}^{n} \frac{(X_i - \mu)^2}{n} \sim \chi^2_n \, ,$$

where $\sum_{i=1}^{n}(X_i - \mu)^2/n$ is used instead of S_n^2. The often used intuition for explaining the distribution of the S_n^2 variable is that "one degree of freedom is lost because μ is replaced by $\overline{X}_n$ in the formula for the variance." This is true, but it does not explain the technical underpinnings of the result. We first use the result to show how to construct confidence intervals for the variance and then we provide a detailed proof of the result.

10.2.1 Deriving and using the confidence intervals for the variance of a Normal sample

Since $(n-1)S_n^2/\sigma^2\chi^2_{n-1}$, it follows that if $\chi^2_{n-1,\alpha}$ is the α quantile of the Chi-squared distribution, then

$$1-\alpha = P\left(\chi^2_{n-1,\alpha/2} \leq \frac{(n-1)S_n^2}{\sigma^2} \leq \chi^2_{n-1,1-\alpha/2}\right) = P\left\{\frac{(n-1)S_n^2}{\chi^2_{n-1,1-\alpha/2}} \leq \sigma^2 \leq \frac{(n-1)S_n^2}{\chi^2_{n-1,\alpha/2}}\right\}.$$

We say that the random interval

$$\left[\frac{(n-1)S_n^2}{\chi^2_{n-1,1-\alpha/2}}, \frac{(n-1)S_n^2}{\chi^2_{n-1,\alpha/2}}\right]$$

is a $100(1-\alpha)\%$ confidence interval for σ^2.

Consider an example, where the brain volume for 513 organo-lead manufacturing workers was measured with an average total brain volume of 1150.315cm^3 and a standard deviation of 105.977cm^3. Assuming normality of the underlying measurements, we would like to calculate a confidence interval for the population standard deviation in total brain volume. Below we provide the R code to obtain this result based on the previous results:

```
#Observed variance
s2 <- 105.977 ^ 2
#Number of subjects
n <- 513
#alpha level
alpha <- .05
#Obtain the quantiles of the chi-square(n-1) distribution
qtiles <- qchisq(c(1 - alpha/2,alpha/2), n - 1)
ival <- (n - 1) * s2 / qtiles
##interval for the sd
round(sqrt(ival),digits=2)
```

```
[1]  99.86 112.89
```

Thus, $[99.87, 112.89]$ is the realization of the confidence interval based on data. Its interpretation is that in repeated samples of 513 workers the realized confidence intervals will cover the true value of the standard deviation 95% of the time. The actual interval either covers or does not cover the true value of the parameter. This interval relies heavily on the assumed normality of the observations. Also, we obtained the 95% confidence interval by square-rooting the endpoints of the confidence interval for σ^2.

10.2.2 Likelihood of the σ^2 parameter

We know that $(n-1)S_n^2/\sigma^2 \sim \text{Gamma}\{(n-1)/2, 1/2\}$ which implies that

$$S_n^2 \sim \text{Gamma}\left(\frac{n-1}{2}, \frac{n-1}{2\sigma^2}\right) .$$

This can be obtained because if X is a random variable with a Gamma(α, β) distribution, then $Y = aX$ has a Gamma$(\alpha, \beta/a)$ distribution. By taking $a = \sigma^2/(n-1)$ we obtain the result. Let us prove this result. The cdf of Y is $F_Y(y) = P(aX \leq y) = P(X \leq y/a) = F_X(y/a)$, where $F_X(\cdot)$ is the cdf of X. Therefore the pdf of Y is $f_Y(y) = \frac{1}{a} f_X(y/a)$, where $f_X(\cdot)$ is the pdf of X. By plugging into the pdf of the Gamma(α, β) distribution we obtain

$$f_Y(y) = \frac{1}{a}\frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{y}{a}\right)^{\alpha-1} e^{-\frac{\beta}{\alpha}y} = \frac{(\beta/a)^\alpha}{\Gamma(\alpha)} y^{\alpha-1} e^{-\frac{\beta}{\alpha}y} ,$$

which is the pdf of the Gamma$(\alpha, \beta/a)$ distribution. The result is useful because it allows us to plot the likelihood of σ^2. We do this for the organo-lead worker example:

```
sigmaVals <- seq(90, 120, length = 1000)
likeVals <- dgamma(s2,shape=(n-1)/2,rate=(n-1)/(2*sigmaVals^2))
likeVals <- likeVals / max(likeVals)
plot(sigmaVals, likeVals, type = "l",lwd=3,col="blue",
     xlab=expression(paste(sigma^2," parameter")),
     ylab="Likelihood",cex.lab=1.3,cex.axis=1.3,
     col.axis="blue")
lines(range(sigmaVals[likeVals >= 1 / 8]),
      c(1 / 8, 1 / 8),col="red",lwd=3)
lines(range(sigmaVals[likeVals >= 1 / 16]),
      c(1 / 16, 1 / 16),col="orange",lwd="3")
```

Figure 10.3 displays the normalized likelihood (likelihood divided by its maximum) of σ^2 in blue and thresholds at 1/8 (red horizontal line) and 1/16 (orange horizontal line). The likelihood looks very close to the Normal distribution because we have 513 organo-lead workers and we already know that for $n > 100$ the approximation of the associated Gamma distribution by a Normal is excellent. In this case the results would be indistinguishable from results using a Normal approximation, though in cases when the number of observations is small to moderate some differences might actually become visible.

10.2.3 Distribution of the empirical variance of a Normal sample distribution

At the risk of becoming technical, we outline the proof of the result, the intuition behind it, and the interesting results obtained along the way. The sketch of the

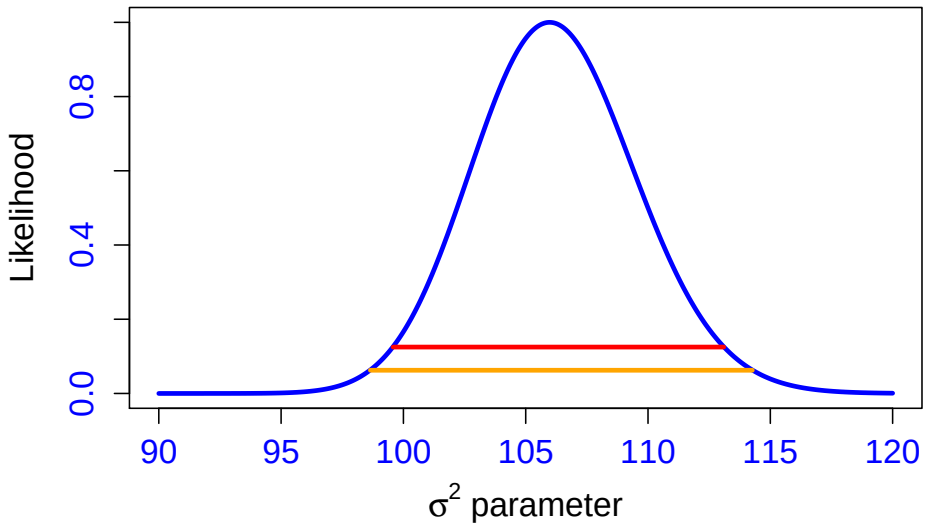


Figure 10.3: Normalized likelihood (to ensure that 1 is the maximum likelihood) of σ^2 (blue) when we assume that the $n = 513$ organo-lead workers brain volume measurements form an iid $N(\mu, \sigma^2)$ random sample. Also shown are thresholds at 1/8 (red horizontal line) and 1/16 (orange horizontal line).

proof has four important steps, some of which we are proving in detail and some of which we are skipping. Warning: technical details ahead.

A. The following decomposition holds

$$\sum_{i=1}^{n} \frac{(X_i - \mu)^2}{\sigma^2} = \sum_{i=1}^{n} \frac{(X_i - \overline{X}_n)^2}{\sigma^2} + \frac{n(\overline{X}_n - \mu)^2}{\sigma^2} = \frac{(n-1)S_n^2}{\sigma^2} + \frac{n(\overline{X}_n - \mu)^2}{\sigma^2} .$$

B. We have $\sum_{i=1}^{n}(X_i - \mu)^2/\sigma^2 \sim \chi^2_n$ and $n(\bar{X}_n - \mu)^2/\sigma^2 \sim \chi^2_1$.

C. The random variables $(n-1)S_n^2/\sigma^2$ and $n(\bar{X}_n - \mu)^2/\sigma^2$ are independent.

D. If a variable with a χ^2_n distribution can be written as the sum of two independent variables, one of which is distributed as a χ^2_1 distribution, then the second variable in the sum has a χ^2_{n-1} distribution.

Proving "D" is above the technical target of this book and will not be done. However, the intuition of the proof should be unmistakable. Indeed, the formula in "A" shows exactly how a random variable with a χ^2_n distribution is written as the sum of two random variables, one of which has a χ^2_1 distribution. This is what is referred in the literature as "losing one degree of freedom." The point "C" is a necessary technical point that allows us to *subtract the number of degrees of freedom* to obtain $n-1$, where n is the number of degrees of freedom of the variable on the left hand side of the equation and 1 is the number of degrees of freedom in one of the variables on the right hand side of the equation.

Independence also allows us to determine that the distribution of $(n-1)S_n^2/\sigma^2$ is χ^2.

We start by proving "A" and observe that we can simply ignore σ^2 everywhere. The idea is to write

$$X_i - \mu = (X_i - \overline{X}_n) + (\overline{X}_n - \mu) \ ,$$

which is obvious by simply adding and subtracting $\overline{X}_n$ and regrouping. Therefore,

$$(X_i - \mu)^2 = (X_i - \overline{X}_n)^2 + 2(X_i - \overline{X}_n)(\overline{X}_n - \mu) + (\overline{X}_n - \mu)^2 \ ,$$

where we simply used the result $(a+b)^2 = a^2 + 2ab + b^2$ for any a and b. Therefore,

$$\sum_{i=1}^n (X_i - \mu)^2 = \sum_{i=1}^n (X_i - \overline{X}_n)^2 + 2\sum_{i=1}^n (X_i - \overline{X}_n)(\overline{X}_n - \mu) + \sum_{i=1}^n (\overline{X}_n - \mu)^2 \ .$$

The first term on the right hand side is, by definition, equal to $(n-1)S_n^2$, while the last term is equal to $n(\overline{X}_n - \mu)^2$ because $(\overline{X}_n - \mu)^2$ does not depend on i and the sum is over n terms. Therefore to prove "A" we need to show only that the middle term is equal to zero. Again, because $(\overline{X}_n - \mu)$ does not depend on i we have

$$\sum_{i=1}^n (X_i - \overline{X}_n)(\overline{X}_n - \mu) = (\overline{X}_n - \mu)\sum_{i=1}^n (X_i - \overline{X}_n) = (\overline{X}_n - \mu)\{\sum_{i=1}^n X_i - n\overline{X}_n\} = 0 \ .$$

The last equality holds by the definition of $\overline{X}_n$.

Proving "B" is relatively easy. The first part was already proved in the beginning of the section. For the second part we will use the following useful result that we do not prove, but use throughout the book.

Result. If $(X_1, \ldots, X_n)$ is a multivariate Normal vector, then scalar or vector linear combinations of the entries of the vector are also Normal. In particular, if $X_1, \ldots, X_n$ are iid $N(\mu, \sigma^2)$ then $(X_1, \ldots, X_n)$ is a multivariate Normal random vector and $\overline{X}_n = \sum_{i=1}^n X_i/n$ is a linear combination of the entries of the vector. Therefore, $\overline{X}_n$ has a Normal distribution. Because $E(\overline{X}_n) = \mu$ and $\mathrm{Var}(\overline{X}_n) = \sigma^2/n$ it follows that

$$\overline{X}_n \sim N\left(\mu, \frac{\sigma^2}{n}\right) \ ,$$

which, by the properties of the Normal distribution, implies that

$$\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} = \frac{\sqrt{n}(\overline{X}_n - \mu)}{\sigma} \sim N(0,1) \ .$$

In turn, this implies that

$$\frac{n(\overline{X}_n - \mu)^2}{\sigma^2} \sim \chi_1^2 \ .$$

To prove "C" we apply again the result discussed above. Since $(X_1, \ldots, X_n)$ is a multivariate normal random vector then

$$(\overline{X}_n, X_1 - \overline{X}_n, \ldots, X_n - \overline{X}_n)$$

is also a multivariate normal random vector because it is a linear transformation of the vector $(X_1, \ldots, X_n)$. Because of this property, in order to prove that $\overline{X}_n$ is independent of the vector $(X_1 - \overline{X}_n, \ldots, X_n - \overline{X}_n)$ it is enough to show that $\overline{X}_n$ is uncorrelated to any of the entries of the vector. This is a general property of multivariate Normal random vectors that will not be proved here. While zero correlation among the entries of a Normal multivariate distribution implies independence of those entries this is not true in general, as zero correlation does not imply independence.

Thus, it is enough to show that

$$\text{Cov}(\overline{X}_n, X_1 - \overline{X}_n) = 0 \ ,$$

as the proof for all other entries will be identical. If we prove this result, then it follows that $\bar{X}_n$ is independent of any function of $(X_1 - \bar{X}_n, \ldots, X_n - \bar{X}_n)$, including S_n^2 and $(n-1)S_n^2/\sigma^2$. Recall that

$$\text{Cov}(\overline{X}_n, X_1 - \overline{X}_n) = E\{\overline{X}_n(X_1 - \overline{X}_n)\} - E(\overline{X}_n)E(X_1 - \overline{X}_n) = E\{\overline{X}_n(X_1 - \overline{X}_n)\} \ .$$

The last equality holds because $E(X_1 - \overline{X}_n) = E(X_1) - E(\overline{X}_n) = 0$. It follows that the covariance is equal to

$$E(\overline{X}_n X_1) - E\{\overline{X}_n^2\} = E(\overline{X}_n X_1) - [\text{Var}(\overline{X}_n) + \{E(\overline{X}_n)\}^2] = E(\overline{X}_n X_1) - (\sigma^2/n + \mu^2) \ .$$

Thus, all we need to show is that $E(\overline{X}_n X_1) = \sigma^2/n + \mu^2$. To obtain this result we follow the simple rules of algebra:

$$E(\overline{X}_n X_1) = \frac{1}{n}\sum_{i=1}^{n} E(X_1 X_i) = \frac{1}{n}\{E(X_1^2) + \sum_{i=2}^{n} E(X_1 X_i)\} = \frac{1}{n}\{E(X_1^2) + \sum_{i=2}^{n} E(X_1)E(X_i)\} \ .$$

The first equality holds from the definition of the empirical mean and the basic rules of expectation, while the second equality is simply splitting the expectations of products to acknowledge that X_1 plays a special role. Indeed, the $E(X_1 X_i) = E(X_1)E(X_i)$ because X_1 is independent of X_i for every $i = 2, \ldots, n$. Because $E(X_i) = \mu$ for every $i = 1, \ldots, n$ it follows that

$$E(\overline{X}_n X_1) = \frac{1}{n}\{E(X_1^2) + (n-1)\mu^2\} = \frac{1}{n}[\text{Var}(X_1) + \{E(X_1)\}^2 + (n-1)\mu^2] \ ,$$

where, for the last equality we used the equality $E(X_1^2) = \text{Var}(X_1) + \{E(X_1)\}^2$. Since $\text{Var}(X_1) = \sigma^2$ and $\{E(X_1)\}^2 = \mu^2$ we obtain

$$E(\overline{X}_n X_1) = \frac{1}{n}[\sigma^2 + \mu^2 + (n-1)\mu^2] = \frac{\sigma^2}{n} + \mu^2 \ .$$

10.2.4 Some implications of the proof

The proof is quite technical, but it also offers a few interesting points. First, the mean $\overline{X}_n$ is uncorrelated to the residuals $X_i - \overline{X}_n$ irrespective to the distribution of the independent random variables $X_1, \ldots, X_n$. This can be used in practice. Consider, for example, the case when we conduct a replication study to measure a particular biomarker. Denote by W_{ij} the measurement for subject $i = 1, \ldots, n$ at time $j = 1, 2$ and let X_i be the true value of the biomarker for subject i. Then a classical measurement error for replication studies can be written as

$$W_{ij} = X_i + U_{ij} \ ,$$

where X_i and U_{ij} are independent for every i and j and U_{ij} is the measurement error for subject i at visit j. Then an estimator of the true long term mean of the biomarker is $\widehat{X}_i = (W_{i1} + W_{i2})/2$ and the residuals are $W_{i1} - \widehat{X}_i = (W_{i1} - W_{i2})/2$ and $W_{i2} - \widehat{X}_i = (W_{i2} - W_{i1})/2$. Our results show that $\widehat{X}_i$ and $W_{i1} - W_{i2}$ are uncorrelated. Thus, plotting one versus the other should show no obvious sign of association. This plot is called the Bland-Altman plot (Bland and Altman 1986) and the associated paper is one of the most widely used and cited papers in the entire scientific literature. For more details on Statistical methods for measurement error models there are at least three monographs (Buonaccorsi 2010; Carroll et al. 2006; Fuller 1987).

10.3 Student's t distribution

This distribution was invented by William Gosset (under the pseudonym "Student") in 1908 (Gosset 1908). The practical need probably came about from some of the work that William Gosset did at the Guinness brewery, where he was employed, and where he dealt with small sample sizes. The distribution is indexed by degrees of freedom, resembles the $N(0, 1)$ distribution, is very well approximated by a $N(0, 1)$ for a moderate and large number of degrees of freedom, and has thicker tails than the Normal.

A random variable T is said to follow a t distribution with n degrees of freedom if

$$T = \frac{Z}{\sqrt{W/n}} \ ,$$

where $Z \sim N(0, 1)$ and $W \sim \chi^2_n$ are independent random variables. We will not derive here the pdf of the distribution and we will use `R` instead to obtain all the necessary characteristics of the distribution.

```
xx=seq(-5,5,length=201)
yx1=dt(xx, 1)
yx2=dt(xx, 2)
yx3=dt(xx, 10)
```

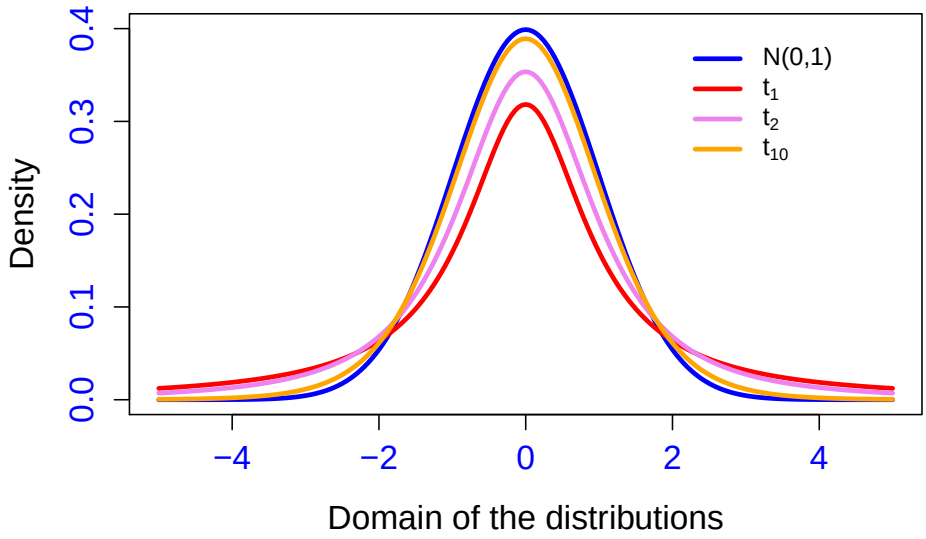


Figure 10.4: Pdfs of the t_1 (red), t_2 (violet), t_{10} (orange) distributions together with the pdf of the $N(0,1)$ distribution.

```
yx5=dnorm(xx)
plot(xx,yx5,type="l",lwd=3,
     xlab=expression(paste("Domain of the distributions")),
     ylab="Density",cex.lab=1.3,cex.axis=1.3,col.axis="blue",
     ylim=c(0,0.4),col="blue")
lines(xx,yx1,lwd=3,col="red")
lines(xx,yx2,lwd=3,col="violet")
lines(xx,yx3,lwd=3,col="orange")
legend(2, 0.4, c("N(0,1)",
                  expression(paste("t"["1"])),
                  expression(paste("t"["2"])),
                  expression(paste("t"["10"]))
                  ),
       col = c("blue","red","violet","orange"),
       lty = c(1,1, 1, 1),lwd=c(3,3,3,3),bty="n")
```

Figure 10.4 displays the $N(0,1)$ pdf in blue and the pdfs of the t_n distributions for $n = 1$ (red), $n = 2$ (violet), and $n = 10$ (orange) indicating the improved approximation by the $N(0,1)$ distribution as n increases. All distributions are symmetric and the thicker tails of the t_n distributions get thinner as n increases. It is interesting to follow how the probability towards the center of the distribution is re-assigned to the tails as n decreases. By $n = 10$ the distribution of Z_n is very close to the $N(0,1)$ indicating that for more than 10-20 degrees of freedom there is not much difference between the Normal and the t.

In practice the most important difference between the Normal and the t distribution is in terms of the quantiles, especially those corresponding to probabilities 0.025 and 0.975. Let us compare these quantiles for a range of degrees of freedom.

```
#Set a range of degrees of freedom
n=c(1,2,5,10,20,50)
alpha <- .05
round(c(qt(1-alpha/2,n),qnorm(1-alpha/2)),digits=2)
```

```
[1] 12.71  4.30  2.57  2.23  2.09  2.01  1.96
```

The 0.975 quantile for the t_1 distribution is 12.71 much larger than 1.96, the corresponding quantile for the Normal distribution. For t_2 the quantile becomes smaller at 4.30, which is more than twice the corresponding $N(0,1)$ quantile. However, as n increases there is less and less difference between the t and Normal quantiles. Indeed, for $t = 50$ the 0.975 quantile is 2.01, which is very close to 1.96. The conclusion is that when the number of degrees of freedom is larger than ~ 50 there really is no difference between the $N(0,1)$ and t_n quantiles. This will become especially useful when we are thinking about constructing confidence intervals using the t versus the Normal approximation based on the CLT. In practice, the CLT always holds for moderate to large n, the t distribution is exact for Normal observations, and the t always provides more conservative approximations to the distribution of the z-score statistic. This is most likely the reason why the t distribution is often used instead of the Normal as it adds a layer of robustness by slightly increasing the length of the confidence intervals.

10.4 Confidence intervals for Normal means

Suppose that $(X_1, \ldots, X_n)$ are iid $N(\mu, \sigma^2)$. We have shown already that

$$\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} \sim N(0,1) \ ,$$

that

$$\frac{(n-1)S_n^2}{\sigma^2} \sim \chi^2_{n-1} \ ,$$

and that $(\overline{X}_n - \mu)/(\sigma/\sqrt{n})$ and $(n-1)S_n^2/\sigma^2$ are independent. In fact, for the last point we have shown a lot more, that $(\overline{X}_n - \mu)/(\sigma/\sqrt{n})$ is independent of $X_i - \overline{X}_n$ for every i. As S_n^2 is a function of the vector $(X_1 - \overline{X}_n, \ldots, X_n - \overline{X}_n)$ it follows that the empirical mean, $\overline{X}_n$, and variance, S_n^2, are independent of each other. Therefore

$$\frac{\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}}}{\sqrt{\frac{(n-1)S_n^2}{(n-1)\sigma^2}}} = \frac{\overline{X}_n - \mu}{S_n/\sqrt{n}}$$

follows Student's t distribution with $n-1$ degrees of freedom, or

$$T_n = \frac{\overline{X}_n - \mu}{S_n/\sqrt{n}} \sim t_{n-1} \ .$$

The main idea behind taking this rather exotic ratio between a $N(0,1)$ and the square root of a random variable with a χ^2 distribution divided by its number of degrees of freedom was to get rid of the unknown standard deviation that appears in the denominator of the z-statistic. Slutsky's theorem allowed us to replace σ by S_n in large samples for any distribution of the random variables $X_1, \ldots, X_n$. The t_{n-1} is exact in finite samples and asymptotically when the random variables are Normally distributed. When the variables are not Normally distributed, the t approximation still provides a more conservative approximation to the true distribution of random variables. After billions of simulations and millions of applications we now know that the t distribution and its associated confidence intervals are extremely hard to beat in practice.

Just as in the case of the confidence intervals built on the CLT, note that the t statistic T_n is a pivot that can be used to construct confidence intervals for the mean. If $t_{n-1,\alpha}$ denotes the α^{th} quantile of the t distribution with $n-1$ degrees of freedom, then

$$1-\alpha = P\left(-t_{n-1,1-\alpha/2} \leq \frac{\overline{X}_n - \mu}{S_n/\sqrt{n}} \leq t_{n-1,1-\alpha/2}\right) = P\left(\overline{X}_n - t_{n-1,1-\alpha/2}S_n/\sqrt{n} \leq \mu \leq \overline{X}_n + t_{n-1,1-\alpha/2}S_n/\sqrt{n}\right) \ ,$$

indicating that $\overline{X}_n \pm t_{n-1,1-\alpha/2}S_n/\sqrt{n}$ is an exact $100(1-\alpha)\%$ confidence interval for the mean of the Normal distribution. The **only** practical difference between this confidence interval and the one based on the CLT plus Slutsky's approach is that it uses the $t_{n-1,1-\alpha/2}$ quantile instead of the $z_{1-\alpha/2}$ quantile of a Normal. Therefore, for $n > 50$ there is basically no difference between these intervals irrespective of the assumptions on which they are derived. For $n \leq 50$ the confidence intervals based on the t distribution are slightly longer than those based on the Normal asymptotic approximation.

The length of the confidence interval is $2t_{n-1,1-\alpha/2}S_n/\sqrt{n}$ based on the t_{n-1} distribution and
$2z_{1-\alpha/2}S_n/\sqrt{n}$ based on the CLT Normal approximation. Thus, the ratio of lengths is

$$\frac{t_{n-1,1-\alpha/2}}{z_{1-\alpha}} \ ,$$

because all other factors cancel out. These ratios can be calculated in `R` for a given number of degrees of freedom as follows:

```
#Set a range of degrees of freedom
n=c(1,2,5,10,20,50)
alpha <- .05
round(qt(1-alpha/2,n)/qnorm(1-alpha/2),digits=2)
```

```
[1] 6.48 2.20 1.31 1.14 1.06 1.02
```

For $n = 50$ the lengths of the t_{20} and t_{50} intervals are only 6% and 2% larger than the one based on the Normal, respectively. A difference of 31% and 14% can be observed for t_5 and t_{10}, respectively, with much larger differences for t_1 and t_2. These results are, of course, identical to the earlier results when we compared quantiles, but the interpretation is now done in terms of length of the confidence intervals.

10.4.1 Properties of the t confidence interval

The t interval technically assumes that the data are iid normal, though it is robust to this assumption. It works well whenever the distribution of the data is roughly symmetric and mound shaped. For large degrees of freedom, t quantiles become the same as standard Normal quantiles and the corresponding confidence interval becomes indistinguishable from the confidence interval based on the CLT.

For skewed distributions, the spirit of the t interval assumptions is violated, though the t confidence interval often continues to perform very well for distributions that have small to moderate skew. For heavily skewed distributions it may not make a lot of sense to center the interval at the mean. In this case, consider taking logs, applying a Box-Cox transformation, or using a different summary, such as the median. For highly discrete data, including binary, other intervals are available and will be discussed in subsequent chapters. Since we mentioned data transformations, the family of Box-Cox transformations (Box and Cox 1964) depends on a parameter λ. More precisely, if the observed data are $x_1, \ldots, x_n$ then the Box-Cox transformation is of the type

$$x_{i,\lambda} = \frac{x_i^\lambda - 1}{\lambda} \ .$$

One of the particular cases for this type of transformation is the log transformation, which corresponds to $\lambda = 0$. In practice, only one other transformation is used consistently, that being the square root transformation $x_{i,1/2} = \sqrt{x_i}$, where the flourish with the minus 1 and division by $\lambda = 1/2$ are often ignored. These transformations are typically designed for positive observations, as powers of negative observations or mixtures of positive and negative observations need to be treated very carefully. Probably the one other data transformation that is worth mentioning is log-one-plus transformation

$$y_i = \log(1 + x_i) \ ,$$

which is especially useful when the observations x_i have 0 values and/or many very small values. This transformation will leave 0 values unchanged, will tend not to induce skewness in the left tail of the distribution due to extreme small positive values, and will tend to substantially reduce the skewness in the right

tail of the distribution. In the end all transformations of the data come at the price of changing the scale on which results are interpreted. Sometimes the benefits outweigh the cost. But often they do not.

10.4.2 Example: sleep data

In R typing `data(sleep)` brings up the sleep data originally analyzed in Gosset's Biometrika paper (Gosset 1908), which shows the increase in hours of sleep for 10 patients on two soporific drugs. `R` treats the data as two groups rather than paired.

```
#Load the data
data(sleep)
print(sleep,row.names=FALSE)
```

```
 extra group ID
   0.7     1  1
  -1.6     1  2
  -0.2     1  3
  -1.2     1  4
  -0.1     1  5
   3.4     1  6
   3.7     1  7
   0.8     1  8
   0.0     1  9
   2.0     1 10
   1.9     2  1
   0.8     2  2
   1.1     2  3
   0.1     2  4
  -0.1     2  5
   4.4     2  6
   5.5     2  7
   1.6     2  8
   4.6     2  9
   3.4     2 10
```

This is not the most compact way the data could be formatted, but having this structure makes a lot of sense from a data analytic perspective. The first column, labeled `extra`, indicates the amount of extra sleep (in hours) when taking one of the treatments, the second column indicates the treatment group (1 or 2, corresponding to first or second treatment), and the third column indicates the individual ID. There are 10 individuals in the study and each one of them had both the first and second treatment. This type of data is called matched pair data because there are two nights on different treatments for each subject. To better visualize and understand the data we reformat it a little.

```
extra_T1=sleep$extra[sleep$group==1]
extra_T2=sleep$extra[sleep$group==2]
diff_T2_minus_T1=extra_T2-extra_T1
sleep_new<-data.frame(extra_T1,extra_T2,diff_T2_minus_T1)
print(sleep_new,row.names = FALSE)
```

```
 extra_T1 extra_T2 diff_T2_minus_T1
      0.7      1.9              1.2
     -1.6      0.8              2.4
     -0.2      1.1              1.3
     -1.2      0.1              1.3
     -0.1     -0.1              0.0
      3.4      4.4              1.0
      3.7      5.5              1.8
      0.8      1.6              0.8
      0.0      4.6              4.6
      2.0      3.4              1.4
```

We would like to construct a confidence interval for the mean of the difference between the two treatment groups under the assumptions that differences in the observations are normally distributed.

```
n <- length(diff_T2_minus_T1)
#calculate the empirical mean of observed difference
mn <- mean(diff_T2_minus_T1)
s <- sd(diff_T2_minus_T1)
#return the estimated mean and standard of the difference vector
round(c(mn,s),digits=2)
```

```
[1] 1.58 1.23
```

```
#Obtain the confidence interval based on explicit calculations
round(mn + c(-1, 1) * qt(.975, n-1) * s / sqrt(n),digits=2)
```

```
[1] 0.70 2.46
```

```
#Obtain the same confidence intervals using the built-in R function
round(t.test(diff_T2_minus_T1)$conf.int[1:2],digits=2)
```

```
[1] 0.70 2.46
```

In contrast the CLT confidence interval is

```
#Obtain the confidence interval based on explicit calculations
round(mn + c(-1, 1) * qnorm(.975) * s / sqrt(n),digits=2)
```

```
[1] 0.82 2.34
```

which is a bit shorter. It is a good idea to check and see how the bootstrap confidence intervals compare.

```
set.seed(236791)
n.boot<-100000
bootstraps<-rep(NA,n.boot)
for (i in 1:n.boot)
{ #Bootstrap the data
  bootstraps[i]<-mean(diff_T2_minus_T1[sample(1:10,replace=TRUE)])
}
quantile(bootstraps,probs=c(0.025,0.975))
```

```
 2.5% 97.5%
 0.95  2.38
```

Interestingly, the bootstrap-based confidence interval is 6% shorter than the CLT, which, in turn, is 14% shorter than the t confidence interval. This is likely due to the fact that both the bootstrap and the CLT confidence intervals have a lower than nominal probability (95%) of covering the true value of the parameter. This is a recognized problem of the bootstrap in small sample sizes and many corrections have been proposed. Probably one of the best solutions is to estimate the mean and standard deviation of the bootstrap means and then use the quantiles of the t_{n-1} distribution to construct the confidence interval. This is done below:

```
mb<-mean(bootstraps)
sdb<-sd(bootstraps)
#Obtain the confidence interval based on explicit calculations
round(mb + c(-1, 1) * qt(.975, n-1) * sdb ,digits=2)
```

```
[1] 0.74 2.42
```

This interval is much closer to the t interval, though it is still about 5% shorter. There is no theoretical justification for the bootstrap in small samples or for the approximations that we are showing here. However, in repeated simulations it has been shown that the t confidence interval performs very well, that the CLT based confidence interval tends to have lower than nominal coverage probability, and that the bootstrap does not perform very well in small samples. Here we provide a simple, albeit not theoretically justified, approach that replaces the use of the quantiles of the bootstrapped sample with a studentized approach where the standard deviation of the mean is multiplied by the corresponding t quantile.

10.4.3 The non-central t distribution

If X is a $N(0,1)$ random variable and W is a Chi-squared random variable with n degrees of freedom, then

$$\frac{X+\mu}{\sqrt{W/n}}$$

is called the non-central t random variable with n degrees of freedom and non-centrality parameter μ/σ. We denote its distribution $t_{n,\mu}$ and note that $t_n = t_{n,0}$, that is, the t_n distribution is a particular case of the non-central $t_{n,\mu}$ distribution for $\mu = 0$.

Recall that if $X_1, \ldots, X_n$ are iid $N(\mu, \sigma^2)$ then $\sqrt{n}(\overline{X}_n/\sigma - \mu/\sigma)$ has a $N(0,1)$ distribution and $(n-1)S_n^2/\sigma^2$ has a Chi-square distribution with $n-1$ degrees of freedom. Therefore

$$\frac{\sqrt{n}(\overline{X}_n/\sigma - \mu/\sigma) + \sqrt{n}\mu/\sigma}{\sqrt{\frac{(n-1)S_n^2}{(n-1)\sigma^2}}} = \frac{\sqrt{n}\overline{X}_n}{S_n}$$

is a non-central t distribution with $n-1$ degrees of freedom and noncentrality parameter $\sqrt{n}\mu/\sigma$, $t_{n-1,\sqrt{n}\mu/\sigma}$. The ratio μ/σ is the *signal to noise ratio* in engineering or the *effect size* in Biostatistics. The result above allows us to calculate and plot the likelihood of the μ/σ. Figure 10.5 displays this likelihood using the density of the non-central t distribution implemented in `R`. Indeed, the expression, `dt(x,n,np)` provides the pdf of a t distribution with `n` degrees of freedom and non-centrality parameter `np` at `x`.

```
tStat <- sqrt(n) * mn / s
esVals <- seq(0, 3, length = 1001)
likVals <- dt(tStat, n - 1, ncp = sqrt(n) * esVals)
likVals <- likVals / max(likVals)
plot(esVals, likVals,type="l",col="blue",lwd=3,
     xlab=expression(paste("Parameter = ", mu,"/",sigma)),
     ylab="Likelihood",cex.lab=1.3,cex.axis=1.3,
     col.axis="blue")
lines(range(esVals[likVals>1/8]), c(1/8,1/8),
      col="red",lwd=3)
lines(range(esVals[likVals>1/16]), c(1/16,1/16),
      col="orange",lwd=3)
```

The MLE of the effect size μ/σ is

```
MLE<-esVals[which.max(likVals)]
round(MLE,digits=2)
```

```
[1] 1.3
```

and the 1/8 likelihood confidence interval for μ/σ is

```
round(range(esVals[likVals>1/8]),digits=2)
```

```
[1] 0.44 2.21
```

The likelihood looks symmetric and bell-shaped, but we know that, in fact, its tails are heavier than the tails of the Normal, as we are dealing with a t distribution with $n-1$ degrees of freedom.

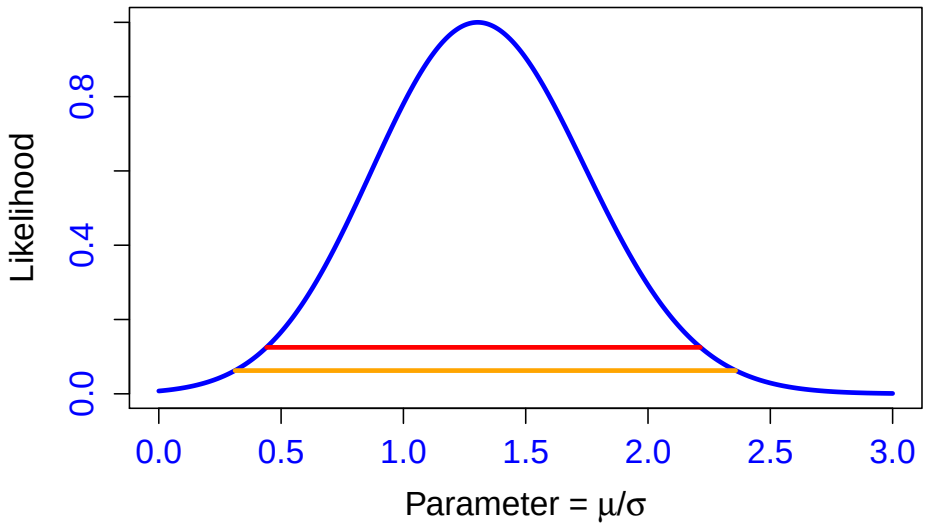


Figure 10.5: Normalized likelihood (to ensure that 1 is the maximum likelihood) of μ/σ (blue) where the differences between the sleep durations in the two treatment groups is assumed to be independent $N(\mu, \sigma^2)$ random variables. Also shown are thresholds at 1/8 (red horizontal line) and 1/16 (orange horizontal line).

10.5 Problems

Problem 1. Here we will verify that standardized means of iid normal data follow Gossett's t distribution. Randomly generate 1000×20 normals with mean 5 and variance 2. Place these results in a matrix with 1000 rows. Using two `apply` statements on the matrix, create two vectors, one of the sample mean from each row and one of the sample standard deviation from each row. From these 1000 means and standard deviations, create 1000 t statistics. Now use `R`'s `rt` function to directly generate 1000 t random variables with 19 df. Use `R`'s `qqplot` function to plot the quantiles of the constructed t random variables versus R's t random variables. Do the quantiles agree? Describe why they should. Repeat the same procedure using the theoretical quantiles of the t_{19} distribution.

Problem 2. Here we will verify the chi-squared result. Simulate 1000 sample variances of 20 observations from a Normal distribution with mean 5 and variance 2. Convert these sample variances so that they should be chi-squared random variables with 19 degrees of freedom. Now simulate 1000 random chi-squared variables with 19 degrees of freedom using `R`'s `rchisq` function. Use `R`'s `qqplot` function to plot the quantiles of the constructed chi-squared random variables versus those of R's random chi-squared variables. Do the quantiles agree? Describe why they should. Repeat the same procedure using the theo-

retical quantiles of the χ^2_{19} distribution.

Problem 3. If $X_1, \ldots, X_n$ are iid $N(\mu, \sigma^2)$ we know that $(n-1)S^2/\sigma^2$ is chi-squared with $n-1$ degrees of freedom. You were told that the expected value of a chi-squared is its degrees of freedom. Use this fact to verify the (already known fact) that $E[S^2] = \sigma^2$. (Note that S^2 is unbiased holds regardless of whether or not the data are Normally distributed. Here we are just showing that the chi-squared result for Normal data is not a contradiction of unbiasedness.)

Problem 4. A random sample was taken of 20 patients admitted to a hospital with a certain diagnosis. The lengths of stays in days for the 20 patients were

$$4, 2, 4, 7, 1, 5, 3, 2, 2, 4, 5, 2, 5, 3, 1, 4, 3, 1, 1, 3$$

a. Calculate a 95% t confidence interval for the mean length of hospital stay. Is your answer reasonable? What underlying assumptions were required for this method and are they reasonable?
b. Calculate a 95% percentile bootstrap interval and interpret.

Problem 5. Refer to the previous problem. Take logs of the data (base e)

a. Calculate a 95% confidence interval for the mean of the log length of stay.
b. Take inverse logs (exponential) of the endpoints of the confidence interval found in part a. Explain why that is a 95% confidence interval for the median length of stay if the data are lognormally distributed (lognormally distributed is when the logarithm of the data points has a normal distribution). Technically, under the lognormal assumption, is the confidence interval that you found in this equation also a confidence interval for the mean length of stay?

Problem 6. Forced expiratory volume FEV is a standard measure of pulmonary function. We would expect that any reasonable measure of pulmonary function would reflect the fact that a person's pulmonary function declines with age after age 20. Suppose we test this hypothesis by looking at 10 nonsmoking males, ages 35-39 and heights 68-72 inches, and measure their FEV initially and then once again two years later. We obtain the following data (expressed in liters)

Person	Year 0	Year 2	Person	Year 0	Year 2
1	3.22	2.95	6	3.25	3.20
2	4.06	3.75	7	4.20	3.90
3	3.85	4.00	8	3.05	2.76
4	3.50	3.42	9	2.86	2.75
5	2.80	2.77	10	3.50	3.32

a. Create the relevant confidence interval and interpret.
b. Create the relevant profile likelihood and interpret.
c. Create a likelihood function for the variance of the change in FEV.

Problem 7. In the SHHS consider the `rdi4p` of all subjects who are 40 and 45 years of age, who never smoked, and who have a BMI less than 25.

a. Create the confidence intervals based on the CLT, t, and bootstrap for the mean `rdi4p` for these study participants.
b. Create the confidence intervals for the standard deviation of the `rdi4p` for these study participants.
c. Plot and interpret the likelihood of σ and μ/σ.

Problem 8. Conduct the same analysis, but with the additional restriction that BMI is less than 22. Compare the results with the results from the previous problem and discuss similarities and discrepancies.

Problem 9. Produce Table 1 based on the data in the SHHS, where columns are age groups in 10-year increments and rows are the variables `sex`, `bmi`, `waist`, `COPD15`, `ASTHMA15`, `rdi4p`, `StOnsetP`, `HTN`, `CVD`, `CHD`, `smokstatus`. What confidence intervals can be constructed for each cell?

Problem 10. Consider the variable `time_bed` and construct a confidence interval for the standard deviation of the σ^2. Compare the confidence interval with the one obtained using a bootstrap of subjects. Under what conditions is the distribution of S_n^2 well approximated by a Normal distribution? Compare the distribution of the `time_bed` variable with the Normal distribution using the `qqplot` function in `R`.

Problem 11. We would like to compare whether the variances of `StOnsetP` (time from going to bed until falling asleep) for study participants who are 40 to 45 years of age versus subjects who are 60 to 65 years of age. Consider the test-statistic $T_n = S_{n,1}^2/S_{n,2}^2$, where $S_{n,1}^2$ and $S_{n,2}^2$ are the variances of the first and second group, respectively, and conduct a bootstrap of subjects in both groups separately to build a confidence interval for σ_1^2/σ_2^2.

Problem 12. Calculate the mean of `rdi4p` and `bmi` in the entire SHHS dataset. We will try to estimate these true means from subsamples of the data. For a range of sample sizes $n = 5, 10, \ldots, 100$ conduct the following experiments. Sample 10000 times with replacement n observations from SHHS.

a. For each sample construct the CLT, t, and bootstrap 95% confidence intervals for the mean and calculate the proportion of times the true sample mean is covered by the resulting confidence intervals.
b. Plot the coverage probability as a function of the sample size, n.
c. Interpret your results.

Problem 13. Calculate the standard deviation of `rdi4p` and `bmi` in the entire SHHS dataset. We will try to estimate these true means from subsamples of the data. For a range of sample sizes $n = 5, 10, \ldots, 100$ conduct the following experiments. Sample 10000 times with replacement n observations from SHHS.

a. For each sample construct the χ^2 and bootstrap 95% confidence intervals for the standard deviation and calculate the proportion of times the true

sample mean is covered by the resulting confidence intervals.
b. Plot the coverage probability as a function of the sample size, n.
c. Interpret your results.

Problem 14. We are interested in estimating and providing a confidence interval for the effect size μ/σ for the proportion of time spent in Rapid Eye Movement in SHHS. The variable containing this information is `timeremp`.

a. What would be a reasonable estimator for μ/σ?
b. Plot the histogram of `timeremp` and the `qqplot` versus a standard Normal distribution. What do you conclude?
c. Under the normality assumption derive and plot the likelihood of μ/σ.
d. Obtain the confidence interval for μ/σ using 1/8 the likelihood and the bootstrap of subjects approach.

Problem 15. Calculate the mean and standard deviation of `timeremp` in the entire SHHS dataset. We will try to estimate the ratio between the SHHS mean and standard deviation of `timeremp` from subsamples of the data. For a range of sample sizes $n = 5, 10, \ldots, 100$ conduct the following experiments. Sample 10000 times with replacement n observations from SHHS.

a. For each sample construct the 1/8 likelihood and bootstrap 95% confidence intervals for the effect size and calculate the proportion of times the effect size for `timeremp` in SHHS is covered by the resulting confidence intervals.
b. Plot the coverage probability as a function of the sample size, n.
c. Interpret your results.

Chapter 11

t and F tests

This chapter covers the following topics

- Independent group t confidence intervals
- t intervals for unequal variances
- t-tests and confidence intervals in `R`
- The F distribution
- Confidence intervals and testing for variance ratios of Normal distributions

The previous chapter focused on cases when we are interested in analyzing a single sample or comparing the means of two groups where observations are paired. A typical example is when the same subject is under two conditions (e.g,. treatment 1 versus treatment 2 or before and after treatment). In this case observations are paired and a typical solution is to construct a confidence interval based on the differences between the two paired measurements. However, in many situations observations are not paired and even the number of observations in each group may be different. Consider, for example, the case when we want to compare the mean blood pressure in two groups in a randomized trial: those who received the treatment and those who received a placebo. We cannot use the paired t confidence interval because the groups are independent and may have different sample sizes. This is why in this chapter we focus on methods for comparing independent groups.

11.1 Independent group t confidence intervals

Consider the case when

a. $X_1, \ldots, X_{n_x}$ are iid $N(\mu_x, \sigma^2)$;
b. $Y_1, \ldots, Y_{n_y}$ are iid $N(\mu_y, \sigma^2)$,

and they are also mutually independent. In this case we assume that the variances of the individual observations, σ^2, is the same (note the lack of indexing on the variances of the samples). Denote by $\overline{X}_n$, $\overline{Y}_n$ the means and $S_{n,x}$, $S_{n,y}$ the standard deviations of the two samples, respectively. Using the fact that linear combinations of entries of Normal vectors are Normal, it follows that

$$\overline{Y}_n - \overline{X}_n \sim N\left\{\mu_y - \mu_x, \sigma^2\left(\frac{1}{n_x} + \frac{1}{n_y}\right)\right\}$$

from the use of convolutions or characteristic functions.

Indeed, $E(\overline{Y}_n - \overline{X}_n) = E(\overline{Y}_n) - E(\overline{X}_n) = \mu_y - \mu_x$ and

$$\mathrm{Var}(\overline{Y}_n - \overline{X}_n) = \mathrm{Var}(\overline{Y}_n) + \mathrm{Var}(\overline{X}_n) = \frac{\sigma^2}{n_y} + \frac{\sigma^2}{n_x} = \sigma^2\left(\frac{1}{n_x} + \frac{1}{n_y}\right) .$$

The pooled variance estimator

$$S^2_{n,\mathrm{pooled}} = \{(n_x - 1)S^2_{n,x} + (n_y - 1)S^2_{n,y}\}/(n_x + n_y - 2)$$

is a good estimator of σ^2 because it combines all the available information about σ^2 and because it is unbiased for σ^2. Indeed,

$$E(S^2_{n,\mathrm{pooled}}) = \frac{1}{n_x + n_y - 2}\{(n_x - 1)E(S^2_{n,x}) + (n_y - 1)E(S^2_{n,y})\} ,$$

and since both $S^2_{n,x}$ and $S^2_{n,y}$ are unbiased for σ^2 it follows that

$$E(S^2_{n,\mathrm{pooled}}) = \frac{1}{n_x + n_y - 2}\{(n_x - 1)\sigma^2 + (n_y - 1)\sigma^2)\} = \sigma^2 .$$

$S^2_{n,\mathrm{pooled}}$ is a sufficient statistics for σ^2, that is, it contains all the information contained in the data that is pertinent to estimating σ^2. We will not prove this latest result, though it is nice to know that there is no unused residual information for estimating σ^2. The pooled variance estimator, $S^2_{n,\mathrm{pooled}}$, is a mixture of the group-specific variances, with greater weight on whichever group has a larger sample size. If the sample sizes are the same, then the pooled variance estimate is the average of the group-specific variances. Moreover, the pooled variance estimator, $S^2_{n,\mathrm{pooled}}$, is independent of $\overline{Y}_n - \overline{X}_n$ because $S^2_{n,x}$ is independent of $\overline{X}_n$ and $S^2_{n,y}$ is independent of $\overline{Y}_n$ and the two groups of observations are independent.

It can be shown that

$$\frac{(n_x + n_y - 2)S^2_{n,\mathrm{pooled}}}{\sigma^2} \sim \chi^2_{n_x+n_y-2} .$$

Indeed,

$$(n_x + n_y - 2)S^2_{n,\mathrm{pooled}}/\sigma^2 = (n_x - 1)S^2_{n,x}/\sigma^2 + (n_y - 1)S^2_{n,y}/\sigma^2 ,$$

and $(n_x - 1)S_x^2/\sigma^2 \sim \chi^2_{n_x-1}$ and $(n_y - 1)S_{n,y}^2/\sigma^2 \sim \chi^2_{n_y-1}$ are independent. The result follows because the sum of two independent χ^2 distributions with $n_x - 1$ and $n_y - 1$ degrees of freedom, respectively, is a random variable with a $\chi^2_{n_x+n_y-2}$ distribution. The reason that 2 degrees of freedom are lost is that we needed to estimate both μ_x and μ_y and use the empirical means, $\overline{X}_n$ and $\overline{Y}_n$, in the corresponding formulas for $S_{n,x}^2$ and $S_{n,y}^2$, respectively.

Putting this all together forms the statistic

$$\frac{\frac{\overline{Y}_n - \overline{X}_n - (\mu_y - \mu_x)}{\sqrt{\sigma^2(1/n_x + 1/n_y)}}}{\sqrt{\frac{(n_x+n_y-2)S^2_{n,\text{pooled}}}{(n_x+n_y-2)\sigma^2}}} = \frac{\overline{Y}_n - \overline{X}_n - (\mu_y - \mu_x)}{S_{n,\text{pooled}}\sqrt{(1/n_x + 1/n_y)}} \sim t_{n_x+n_y-2}$$

because it is a standard Normal divided by an independent $\chi^2_{n_x+n_y-2}$ distribution divided by $n_x + n_y - 2$, its number of degrees of freedom. Just as in the case of one sample t-statistic, the form of the expression is of the type

$$\frac{\text{estimator} - \text{true value}}{\text{SE}},$$

where, in this case estimator $= \overline{Y}_n - \overline{X}_n$, true value $= \mu_y - \mu_x$ and

$$\text{SE} = S_{n,\text{pooled}}\sqrt{(1/n_x + 1/n_y)}\,.$$

Using this expression as a pivot around $\mu_y - \mu_x$ we can obtain a $100(1-\alpha)\%$ confidence interval as

$$\overline{Y}_n - \overline{X}_n \pm t_{n_x+n_y-2,1-\alpha/2} S_{n,\text{pooled}} \left(\frac{1}{n_x} + \frac{1}{n_y}\right)^{1/2}.$$

This interval assumes that the two samples have the same variance. Also note that the variance is the same if the measure we care about is $\overline{Y}_n - \overline{X}_n$ or $\overline{X}_n - \overline{Y}_n$, though the intervals are different. If there is some doubt, we can assume a different variance per group, which we will be discussed later. We will also discuss tests for equal variance to gain insight into when it is reasonable to assume equal variance.

11.1.1 Example: age of smokers versus non-smokers

A common use for two-sample t-tests is to create a table comparing two groups of interest on a nuisance variable that is not of interest. This is done to see how alike the two groups are for subsequent comparisons. The paper by (Zhang et al. 2006) is focused on comparing current smokers to never smokers with respect to sleep characteristics. They first compare the ages of the two groups. The mean (sd) age (in years) for current smokers was 59.6 (9.5) while it was 63.5 (11.5) for never smokers. Even though the sample sizes are different in the paper, for

didactical reasons we assume that these data were observed for 10 smokers and 10 never smokers. We are interested in constructing a confidence interval for the difference in average age between the two groups.

The empirical pooled standard deviation is

$$s_{n,\text{pooled}} = \sqrt{\frac{9.5^2(10-1)+11.5^2(10-1)}{20-2}} = 10.55\ ,$$

where, as usual, we use lower case $s_{n,\text{pooled}}$ to denote the realization of the pooled standard deviation random variable, $S_{n,\text{pooled}}$. The 0.975 quantile of a t_{18} distribution is

```
round(qt(.975,18),digits=2)
```

```
[1] 2.1
```

Therefore the realization of the 95% confidence interval is

$$59.6 - 63.5 \pm 2.1 \times 10.55\sqrt{\frac{1}{10}+\frac{1}{10}}$$

```
[1] -13.81   6.01
```

This interval contains 0 indicating that there is not enough evidence to exclude a value of zero as a potential value of the true difference in means $\mu_y - \mu_x$. Equivalently, there is not enough evidence against the assumption that the two group means are equal, $\mu_x = \mu_y$.

11.1.2 Example: time in bed and percent REM

Consider the SHHS and we will focus on two groups. The first group has a `bmi` below 25 and age below 50 and the second group has a `bmi` above 35 and age above 70. We would like to compare these groups in terms of time in bed, `time_bed`, and respiratory disturbance index, `rdi4p`. First, we download the data.

```
file.name = file.path("data", "shhs1.txt")
data.cv<-read.table(file=file.name,header = TRUE,na.strings="NA")
```

Second, we use the `pipe` operator to create the two subgroups.

```
library(tidyverse)
file.name = file.path("data", "shhs1.txt")
data_cv <- read.table(file=file.name,header = TRUE,na.strings="NA")

data_cv = data_cv %>%
  mutate(group = case_when(
    (bmi_s1 < 25 & age_s1 < 50) ~ "Group 1",
```

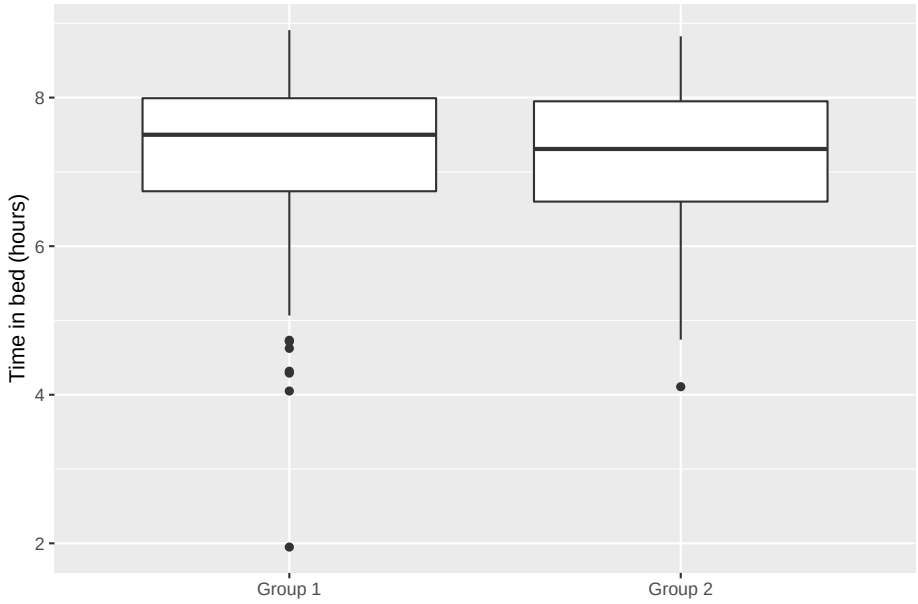


Figure 11.1: Boxplots for time in bed for Group 1, which contains all study participants with BMI less than 25 and age less than 50, and Group 2, which contains all study participants with BMI greater than 35 and age greater than 70.

```
    (bmi_s1 > 35 & age_s1 > 70) ~ "Group 2"))
time_in_bed = data_cv %>%
  filter(!is.na(group)) %>%
  mutate(time_bed = time_bed / 60)
time_in_bed_g1 = time_in_bed$time_bed[ time_in_bed$group == "Group 1"]
time_in_bed_g2 = time_in_bed$time_bed[ time_in_bed$group == "Group 2"]

# this is bad - gives same plot
data.box<-list(time_in_bed_g1,time_in_bed_g2)
```

Let us first plot the distributions of `time_bed` for the two groups to get a better idea about potential differences between the distributions.

```
time_in_bed %>%
  ggplot(aes(x = group, y = time_bed)) +
  geom_boxplot() +
  ylab("Time in bed (hours)") +
  xlab("")
```

Figure 11.1 indicates that the median number of hours in bed for both groups is around 7.5 hours, though the distributions are quite variable and include some

extreme outliers in both groups with less than 5 hours in bed. Even though the groups are quite different both in terms of `age` and `bmi`, the distributions of the time in bed seem to be remarkably similar. We would like to apply a t-test for the difference in the mean time in bed between the two groups. We will not perform any tests for Normality in this analysis.

The mean (standard deviation) of time in bed in hours is 7.29 (1.01) for group 1, which contains 248 subjects, and 7.2 (0.97) for group 2, which contains 97 subjects. In this case the empirical pooled standard deviation is

$$s_{n,\text{pooled}} = \sqrt{\frac{1.01^2(248-1) + 0.97^2(97-1)}{248+97-2}} = 1.00 \, .$$

The 0.975 quantile of a $t_{248+97-2} = t_{351}$ distribution is

```
round(qt(.975,343),digits=2)
```

```
[1] 1.97
```

Note that this is very close to the $Z_{0.975} = 1.96$ as the t distribution is approximately Normal when the degrees of freedom are large. Therefore the realization of the 95% confidence interval is

$$7.29 - 7.20 \pm 1.97 \times 1.00 \sqrt{\frac{1}{248} + \frac{1}{97}} \, ,$$

which results in an observed confidence interval of $[-0.15, 0.33]$ hours. The confidence interval covers 0 indicating there is not enough evidence against the assumption that the two true means of the two groups are equal. This information is something we may be able to get a feel for in the plot, but this approach gives us a quantifiable result to evaluate that visual assessment.

Now we will investigate whether there are differences in terms of `rdi4p` between the same two groups. Figure 11.2 displays the boxplots of `rdi4p` across the two groups:

```
time_in_bed %>%
  ggplot(aes(x = group, y = rdi4p)) +
  geom_boxplot() +
  ylab("RDI") +
  xlab("")
```

Because data in both groups are heavily skewed we choose to transform the `rdi4p` first by taking the square root transformation. This transforms 0 `rdi4p` into 0 for the square root, keeps the ordering of the quantiles, and makes both distributions less skewed. Other options could be performing a log transformation, such as `log(rdi4p + 1)` (adding `1` because `log(0) = -\infty` and `log(0+1) = 0`). In practice, one could argue that a test on the original scale would make more scientific sense, but one should also recognize that in that situation one may want to use a test for difference in medians or other quantiles. Figure 11.3 displays the boxplots for the square root of `rdi4p`:

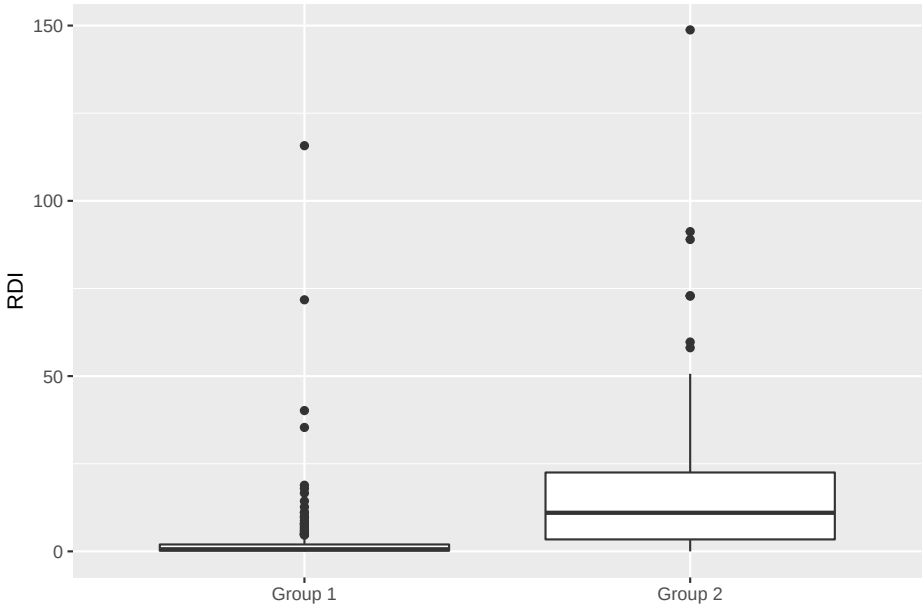


Figure 11.2: Boxplots for rdi for Group 1, which contains all study participants with BMI less than 25 and age less than 50, and Group 2, which contains all study participants with BMI greater than 35 and age greater than 70.

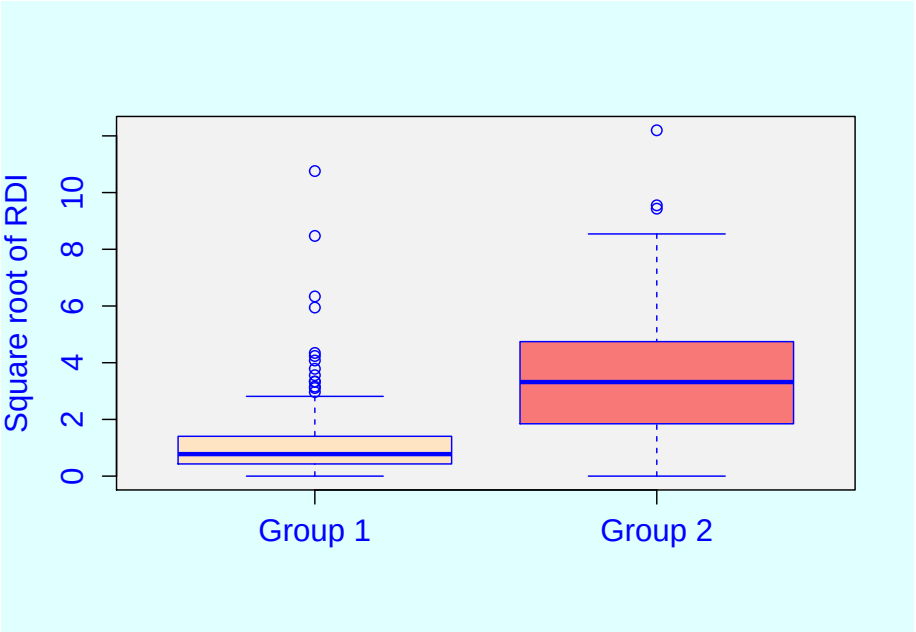


Figure 11.3: Boxplots for the square root of rdi for Group 1, which contains all study participants with BMI less than 25 and age less than 50, and Group 2, which contains all study participants with BMI greater than 35 and age greater than 70.

```
sqrt_rdi_g1 = sqrt(time_in_bed$rdi4p[ time_in_bed$group == "Group 1"])
sqrt_rdi_g2 = sqrt(time_in_bed$rdi4p[ time_in_bed$group == "Group 2"])
data.box<-list(sqrt_rdi_g1,sqrt_rdi_g2)
par(bg="lightcyan")
plot(1.5, 1.5, type="n", ann=FALSE, axes=FALSE)
u <- par("usr") # The coordinates of the plot area
rect(u[1], u[3], u[2], u[4], col="gray95", border=NA)
par(new=TRUE)
boxplot(data.box,col=c("bisque",col = rgb(1,0,0,0.5)),
        cex.lab=1.3,cex.axis=1.3,col.axis="blue",col.lab="blue",
        names=c("Group 1","Group 2"),ylab="Square root of RDI",
        border=c("blue","blue"))
```

```
dev.off()
```

```
null device
          1
```

Figure 11.4 is the same as Figure 11.3, but done using the `ggplot` function.

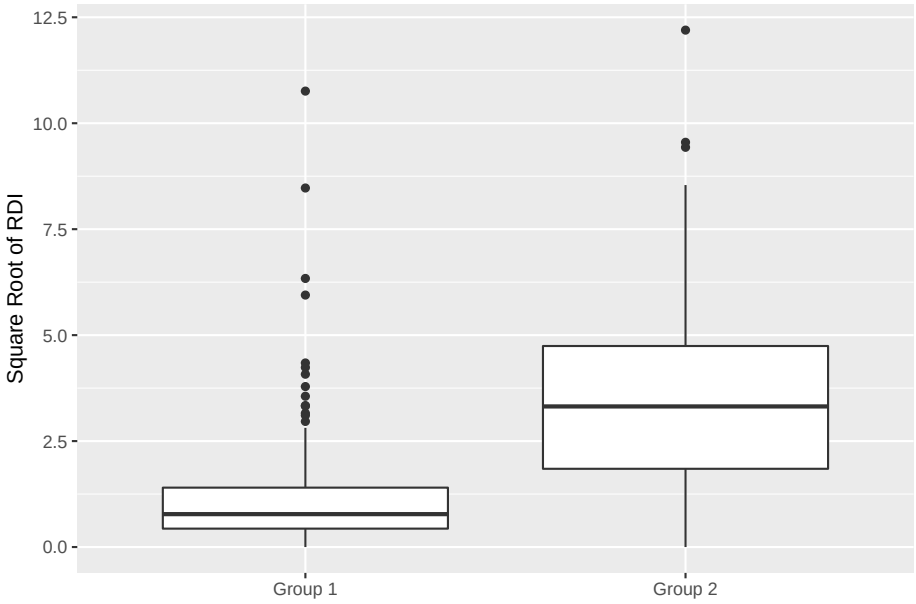


Figure 11.4: Boxplots for the square root of rdi for Group 1, which contains all study participants with BMI less than 25 and age less than 50, and Group 2, which contains all study participants with BMI greater than 35 and age greater than 70.

```
time_in_bed %>%
  mutate(sqrt_rdi = sqrt(rdi4p)) %>%
  ggplot(aes(x = group, y = sqrt_rdi)) +
  geom_boxplot() +
  ylab("Square Root of RDI") +
  xlab("")
```

The distribution of the square root of `rdi4p` is still skewed for Group 1, but less skewed than the original `rdi4p` data.

The mean (standard deviation) of the square root of `rdi4p` (in square root of events per hour) is 1.11 (1.23) for group 1, which contains 248 subjects, and 3.65 (2.31) for group 2, which contains 97 subjects. In this case the empirical pooled standard deviation is

$$s_{n,\text{pooled}} = \sqrt{\frac{1.23^2(248-1) + 2.31^2(97-1)}{248+97-2}} = 1.61 \, .$$

Therefore the realization of the 95% confidence interval is

$$1.11 - 3.65 \pm 1.97 \times 1.61 \sqrt{\frac{1}{248} + \frac{1}{97}} \, ,$$

which results in an observed confidence interval of $[-2.92, -2.16]$ square root of number of events per hour. The confidence interval does not cover 0 indicating that there is strong evidence against the assumption that the means of the two groups are equal. One could argue that the variance in the two groups may not be equal, that the distribution of the square root of `rdi4p` remains skewed, especially in Group 1, and that the scientific interpretation on the square root scale of the `rdi4p` is tenuous.

Therefore, one could be interested in approaches for calculating confidence intervals without the assumptions of Normality or equal variances. For example, we could be interested in the confidence interval for the difference in the mean or medians of the two groups. One could conduct a bootstrap approach to obtain these confidence intervals. To do that we bootstrap subjects with replacement in every group and calculate the statistics of interest for every sample:

```
#Setting the seed
set.seed(3566551)

#Extract the rdi for each group and remove the NAs
rdi_g1<-rdi4p[index_g1]
rdi_g1_non_NA<-rdi_g1[!is.na(rdi_g1)]
rdi_g2<-rdi4p[index_g2]
rdi_g2_non_NA<-rdi_g2[!is.na(rdi_g2)]

#Set the vectors that will store the mean and median differences
n_boot=10000
mean_diff_boot=rep(NA,n_boot)
median_diff_boot=rep(NA,n_boot)

v1<-1:length(rdi_g1_non_NA)
v2<-1:length(rdi_g2_non_NA)
#Start the bootstrap
for (i in 1:n_boot)
{
  ind1<-sample(v1,replace=TRUE)
  ind2<-sample(v2,replace=TRUE)
  mean_diff_boot[i]=mean(rdi_g2_non_NA[ind2])-mean(rdi_g1_non_NA[ind1])
  median_diff_boot[i]=median(rdi_g2_non_NA[ind2])-median(rdi_g1_non_NA[ind1])
}
```

```
#Setting the seed
set.seed(3566551)

samp = time_in_bed %>%
  group_by(group)
samp = samp %>%
  summarize(mean = mean(rdi4p),
```

```
            med = median(rdi4p))
orig_est = samp %>%
  summarize(mean = diff(mean),
            med = diff(med))
orig_est = orig_est %>% tidyr::gather(type, diff)

original_mean_diff = samp$mean[2] - samp$mean[1]
original_median_diff = samp$med[2] - samp$med[1]

#Set the vectors that will store the mean and median differences
n_boot=10000
mean_diff_boot = rep(NA,n_boot)
median_diff_boot = rep(NA,n_boot)

#Start the bootstrap
for (i in 1:n_boot)
{
  samp = time_in_bed %>%
    group_by(group) %>%
    sample_frac(replace = TRUE)
  samp = samp %>%
    summarize(mean = mean(rdi4p),
              med = median(rdi4p))
  mean_diff_boot[i] = samp$mean[2] - samp$mean[1]
  median_diff_boot[i] = samp$med[2] - samp$med[1]
}
diff_df = bind_rows(
  data_frame(
  diff = mean_diff_boot,
  type = "mean"),
  data_frame(
  diff = median_diff_boot,
  type = "median")  )
head(diff_df)
```

```
# A tibble: 6 x 2
   diff type
  <dbl> <chr>
1  18.3 mean
2  11.5 mean
3  11.3 mean
4  15.1 mean
5  16.4 mean
6  15.0 mean
```

Figure 11.5 displays the histograms of the bootstrap samples for the mean (red)

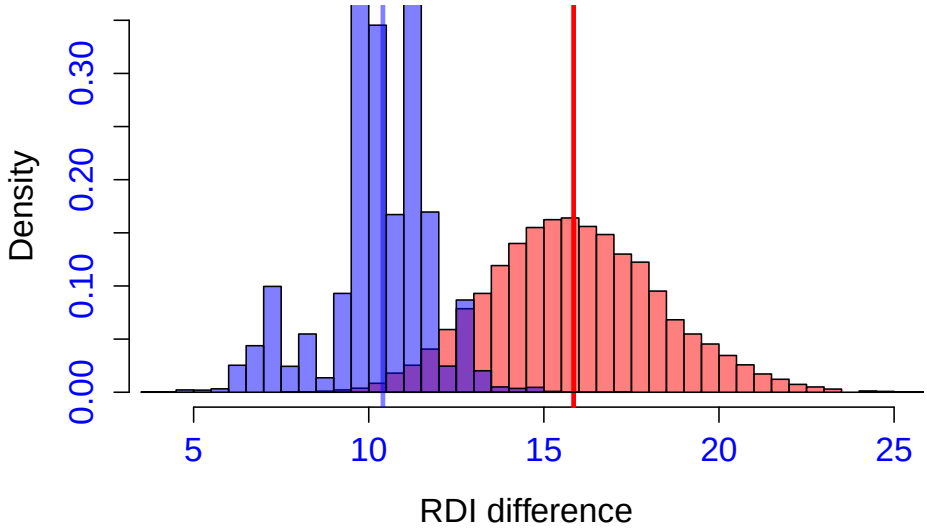


Figure 11.5: Histograms of the differences in the mean (red) and median (blue) rdi between Group 2 and Group 1.

and median (blue) difference between the RDI in group 2 and group 1. The distribution of bootstrap sample for the mean difference is more symmetric, almost normally distributed, whereas the distribution for the median difference is bi-modal with the first local maximum around 7 events per hour and the second one around 10 events per hour. The difference in the median number of events tends to be smaller than the difference in the mean number of events (the blue histogram is shifted to the left of the red histogram), while the variability of the difference in mean distribution is larger. This should not be surprising because the RDI distributions in both groups are highly skewed with many outliers. Figure 11.6 is the same as Figure 11.5, but done using the `ggplot` function instead of base `R`.

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
cis = diff_df %>%
  group_by(type) %>%
  summarize(
    q25 = round(quantile(diff, probs = 0.25), 2),
    q75 = round(quantile(diff, probs = 0.75), 2)
  )
```

A 95% bootstrap confidence interval for the mean is (11.35,20.95) and for the median is (6.64,12.83). An alternative for this example would be to test for the difference in medians using nonparametric tests, such as the Wilcoxon Rank-sum test. But again, the bootstrap can show you how to make few assumptions about the distribution of the data and get results. The tradeoff is that the

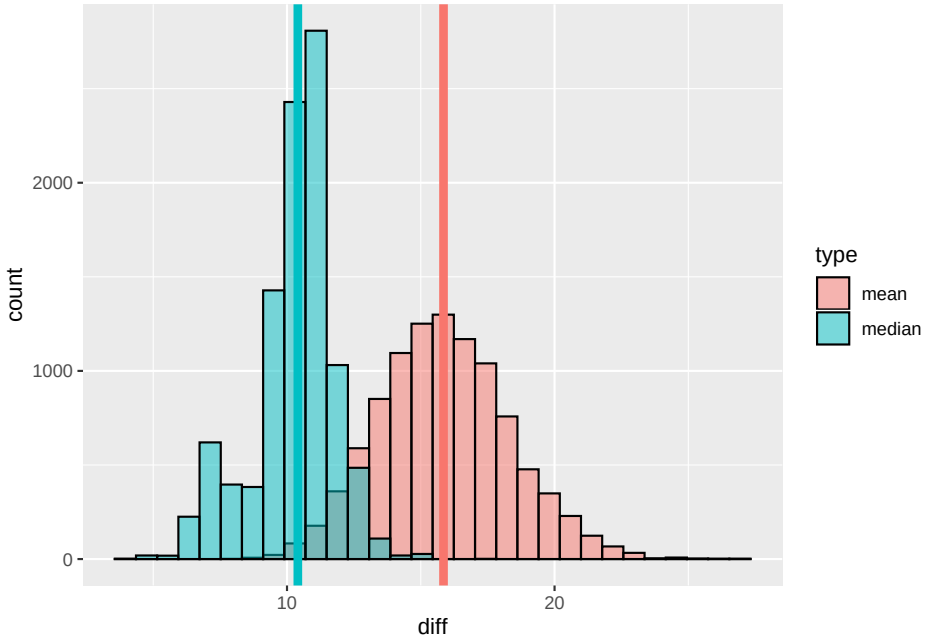


Figure 11.6: Histograms of the differences in the mean (red) and median (blue) rdi between Group 2 and Group 1.

sample needs to be similar to the population and that coverage of confidence intervals may have bad propoerties in small sample sizes.

11.2 t intervals for unequal variances

Consider now the case when

a. $X_1, \ldots, X_{n_x}$ are iid $N(\mu_x, \sigma_x^2)$
b. $Y_1, \ldots, Y_{n_y}$ are iid $N(\mu_y, \sigma_y^2)$

and they are also mutually independent. In this case we assume that the variances in the two groups are different (note different indices on the sample variances). In this case it can be shown that

$$\overline{Y}_n - \overline{X}_n \sim N\left(\mu_y - \mu_x, \frac{\sigma_x^2}{n_x} + \frac{\sigma_y^2}{n_y}\right)$$

The statistic

$$\frac{\overline{Y}_n - \overline{X}_n - (\mu_y - \mu_x)}{\left(S_{n,x}^2/n_x + S_{n,y}^2/n_y\right)^{1/2}}$$

approximately follows Gosset's t_d distribution with degrees of freedom equal to

$$d = \frac{\left(S_{n,x}^2/n_x + S_{n,y}^2/n_y\right)^2}{S_{n,x}^2/\{n_x^2(n_x-1)\} + S_y^2/\{n_y^2(n_y-1)\}} \ .$$

The exact distribution of the test statistic is not exactly this distribution. This approximation is due to Welch (Welch 1938), who provided the approximated distribution for the test statistic. In statistics this is the famous Behrens-Fisher (Behrens 1929; Fisher 1935) problem. The confidence intervals for $\mu_y - \mu_x$ are then obtained similarly to the case of the pooled variance with the only difference that the number of degrees of freedom, d, of the t_d distribution is estimated from the data, as above, and the standard error has the formula

$$\text{SE} = \left(S_{n,x}^2/n_x + S_{n,y}^2/n_y\right)^{1/2} \ .$$

The bootstrap procedure still works here for the mean and will give the same result as above. Bootstrapping other statistics such as the t value will be different depending on the variance estimate.

11.3 t-tests and confidence intervals in R

The function in R to construct confidence intervals based on the t-statistic is `t.test` and has a variety of options that make it applicable in a wide range of situations. The general form of the function in R is

```
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)
```

where `x` is the data from group 1 and `y` is the data from group 2. Other times we may use the `t.test` function with the syntax `t.test(formula, data)`, where the `formula = y ~ group` where `y` is the variable to test and `group` is a grouping variable (with 2 levels).

11.3.1 One sample t-tests and paired data

Let us see how the function works for the sleep data example first. The data are paired, so 10 individuals were given two different drugs:

```
#Load the data
data(sleep)
sleep = sleep %>%
  arrange(ID)
head(sleep)
```

```
  extra group ID
1   0.7     1  1
2   1.9     2  1
3  -1.6     1  2
4   0.8     2  2
5  -0.2     1  3
6   1.1     2  3
```

We can take the difference when the individuals are in each group:

```
x_sleep <- with(sleep, extra[group == 2] - extra[ group==1])
```

where the `with` functions allows us to work similar to `dplyr` functions where we do not need dollar signs or references. It knows we are referencing the `sleep` data set. We then pass this vector into a t-test:

```
#Conduct the t test
ttest_results <- t.test(x_sleep)
ttest_results
```

```

    One Sample t-test

data:  x_sleep
t = 4.0621, df = 9, p-value = 0.002833
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 0.7001142 2.4598858
sample estimates:
mean of x
     1.58
```

The `broom` package allows us to "tidy" up this result into a `data.frame`-type object:

```
library(broom)
tidy_results = tidy(ttest_results)
tidy_results
```

```
# A tibble: 1 x 8
  estimate statistic p.value parameter conf.low conf.high method
     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl> <chr>
1     1.58      4.06 0.00283         9    0.700      2.46 One S~
# ... with 1 more variable: alternative <chr>
```

The results indicate that this is a one-sample t-test (on the differences), which will be covered later, and the defaults are that the alternative is `two.sided` and the confidence intervals are at a confidence level `conf.level = 0.95`. The t-statistic is 4.06 and the number of degrees of freedom used to calculate the

null distribution of the test is a t_9, where $9 = 10 - 1$. The p-value for testing the null hypothesis $H_0 : \mu = 0$ versus the alternative $H_A : \mu \neq 0$ is equal to 0.0028, indicating that there is strong evidence against the hypothesis that the groups have equal means. Alas, we have not yet learned about testing, and for now we are only interested in the confidence intervals. The output contains the 95% confidence interval, which, for this data, was realized as $[0.70, 2.46]$, which does not include 0. The output also displays the mean of the variable `x_sleep`. If we want to only obtain the confidence intervals we could use the code:

```
ttest_results$conf.int
```

```
[1] 0.7001142 2.4598858
attr(,"conf.level")
[1] 0.95
```

which produces the confidence interval together with the level of confidence. The level of confidence can be adjusted in the `t.test` function using, for example, `conf.level=0.90`. We could obtain only the confidence interval if we used the expression

```
round(ttest_results$conf.int[1:2],digits=2)
```

```
[1] 0.70 2.46
```

To obtain all other objects produced by the `t.test` we can type:

```
names(ttest_results)
```

```
 [1] "statistic"   "parameter"   "p.value"     "conf.int"    "estimate"
 [6] "null.value"  "stderr"      "alternative" "method"      "data.name"
```

which gives output that can be referenced with a dollar sign (`$`). We can also use our `tidy_results` if that is more intuitive.

The same results can be obtained by creating the difference

```
ttest_results <- t.test(extra~group, paired = TRUE, data=sleep)
```

In this case we specify that the test is done on the variable `extra` and will compare `group` means when observations are paired. The pairing here is done in order, that is, the first observation from group 1 is paired with the first observation from group 2 and so on. The variable `ID` is not used for pairing, and so one should be careful to organize the data accordingly.

11.3.2 Using an unpaired t-test

If we do not specify `paired=TRUE`, the default is to run a non-paired t-test as below:

```
ttest_results<-t.test(extra~group, data=sleep)
ttest_results
```

```
    Welch Two Sample t-test

data:  extra by group
t = -1.8608, df = 17.776, p-value = 0.07939
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -3.3654832  0.2054832
sample estimates:
mean in group 1 mean in group 2
           0.75            2.33
```

The results are quantitatively and qualitatively different, indicating that, in this case, the 95% confidence interval for the difference in means between groups 1 and 2 is $[-3.36, 0.21]$, which contains 0 and indicates that there is no evidence against a difference between the two groups. This happens because we ignored the fact that observations were truly paired and the estimator of the variance of the difference in means is inflated. Indeed, the length of the confidence interval is 3.57 versus 3.16 when data are correctly treated as paired.

For the SHHS we have already shown how to derive the 95% confidence intervals for the difference in the mean time in bed between groups 1 and 2. The same results can be obtained directly using the `t.test` function in R. If we assume equal variances, the t-test with pooled variance estimator can be derived

```
ttest_results <- t.test(time_bed ~ group,
                        data = time_in_bed,
                        paired = FALSE, var.equal=TRUE)
conf_int<-round(ttest_results$conf.int[1:2],digits=2)
ttest_results
```

```
    Two Sample t-test

data:  time_bed by group
t = 0.77135, df = 343, p-value = 0.441
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1428636  0.3272091
sample estimates:
mean in group Group 1 mean in group Group 2
             7.287534              7.195361
```

The results are identical to those obtained from scratch and the 95% confidence interval is [-0.14, 0.33]. In this expression we did not need to specify that the test

is not paired, `paired=FALSE`, as this is the default option. However, specifying that the variances are equal, `var.equal=TRUE` is necessary, as the default is to assume they are not. Note the test being done is a `Two Sample t-test`. Let us compare these results with results based on the assumption that the variances are unequal. Recall that, in this case, the two variances were $1.01^2 = 1.02$ and $0.97^2 = 0.94$, making them very close.

```
ttest_results <- t.test(time_bed ~ group,
                        data = time_in_bed,
                        paired = FALSE, var.equal=FALSE)
conf_int<-round(ttest_results$conf.int[1:2],digits=2)
ttest_results
```

```
    Welch Two Sample t-test

data:  time_bed by group
t = 0.78326, df = 181.04, p-value = 0.4345
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.1400248  0.3243704
sample estimates:
mean in group Group 1 mean in group Group 2
             7.287534              7.195361
```

Note here the results are from a `Welch Two Sample t-test` indicating the degrees of freedom are adjusted using the unequal variance assumption.

When the variances are not assumed to be equal the number of degrees of freedom is smaller, changing from 343 to 181.04. The standard deviation of the empirical mean difference under the equal variance assumption is

$$\text{se}_{n,\text{equal}} = s_{n,\text{ pooled}}\sqrt{\frac{1}{n_x} + \frac{1}{n_y}}$$

and under the unequal variance assumption is

$$\text{se}_{n,\text{unequal}} = \sqrt{\frac{s_{n,x}^2}{n_x} + \frac{s_{n,y}^2}{n_y}}\ ,$$

where, again, we used the lower case $s_{n,x}^2$ and $s_{n,y}^2$ to indicate the realization of $S_{n,x}^2$ and $S_{n,y}^2$, respectively. This can be calculated in `R` as

```
overall_var = var(c(time_in_bed_g1, time_in_bed_g2), na.rm = TRUE)
#Number of observations in each group
nx=sum(!is.na(time_in_bed_g1))
ny=sum(!is.na(time_in_bed_g2))
```

```
#Estimated variance of the mean in each group
var_x_mean=var(time_in_bed_g1,na.rm=TRUE)/nx
var_y_mean=var(time_in_bed_g2,na.rm=TRUE)/ny

se_n_equal=round(overall_var*sqrt(1/nx+1/ny),digits=3)
se_n_unequal=round(sqrt(var_x_mean+var_y_mean),digits=3)
```

or in dplyr:

```
time_in_bed %>%
  # get overall variance
  mutate(overall_var = var(time_bed)) %>%
  group_by(group) %>%
  # get nx, sx^2/nx and 1/n_x
  summarize(n = n(),
            var = var(time_bed)/n,
            n_inverse = 1/n,
            overall_var = first(overall_var)) %>%
  # sqrt(sx^2/nx + sy^2/nx) and  sqrt( s^2(1/nx + 1/ny))
  summarize(unequal = sqrt(sum(var)),
            equal = sqrt(sum(n_inverse * overall_var)))
```

```
# A tibble: 1 x 2
  unequal equal
    <dbl> <dbl>
1   0.118 0.119
```

Thus, the estimated standard deviation of the mean difference is 0.119 under the equal variance assumption and is 0.118 under the unequal variance assumption. The difference in the t quantiles used is minimal and so is the difference between standard deviations, which results in an almost identical confidence interval [-0.14, 0.32] in the case when variances are not assumed equal.

Consider now the case when we test for the difference in square root of `rdi4p` between the same two groups. The visual inspection of the boxplots for the two distributions seems to indicate that the variability is larger in group 2 (estimated standard deviation 2.31) versus group 1 (estimated standard deviation 1.23). We obtain confidence intervals under the assumption of equal variances (which seems to be violated) and under unequal variances and compare the results.

```
#Calculate confidence intervals under the assumption of equal variance
ttest_equal = time_in_bed %>%
  mutate(sqrt_rdi = sqrt(rdi4p)) %>%
  t.test(sqrt_rdi ~ group, data = .,
         paired=FALSE,var.equal=TRUE
  )
tidy(ttest_equal)
```

```
# A tibble: 1 x 9
  estimate1 estimate2 statistic  p.value parameter conf.low conf.high
      <dbl>     <dbl>     <dbl>    <dbl>     <dbl>    <dbl>     <dbl>
1      1.11      3.65     -13.2 2.24e-32       343    -2.92     -2.16
# ... with 2 more variables: method <chr>, alternative <chr>
```

```
#Calculate confidence intervals under the assumption of unequal variance
ttest_unequal = time_in_bed %>%
  mutate(sqrt_rdi = sqrt(rdi4p)) %>%
  t.test(sqrt_rdi ~ group, data = .,
         paired=FALSE,var.equal=FALSE)
tidy(ttest_unequal)
```

```
# A tibble: 1 x 10
  estimate estimate1 estimate2 statistic  p.value parameter conf.low
     <dbl>     <dbl>     <dbl>     <dbl>    <dbl>     <dbl>    <dbl>
1    -2.54      1.11      3.65     -10.3 4.60e-18      118.    -3.03
# ... with 3 more variables: conf.high <dbl>, method <chr>,
#   alternative <chr>
```

The 95% confidence interval using equal variances is [-2.92, -2.16] and using unequal variances is [-3.03, -2.05], indicating that the interval based on equal variances is approximately 22% shorter than the interval based on the more realistic assumption that they are not. The effects can be much more pronounced in cases when the imbalance between the group 1 and 2 variances is even larger. In some cases, there may be a strong amount of knowledge outside of the data that the variances should be equal (e.g. they come from the exact same population). Otherewise, let us look at tests on the variances themselves.

11.4 The F distribution

One of the sticky points when constructing confidence intervals for the difference in the means of two populations is whether or not we can assume equal variance and use the pooled variance estimator. Thus, we would need a reasonable approach to decide whether the empirical variances of the two populations are sufficiently different to provide evidence against the assumption that the true variances are equal. To do this we need to obtain the distribution of the ratio of variances in group 1 and 2:

Consider the sample variances $S^2_{n,x}$ and $S_{n,y}$ for a collection of iid random variables $X_1, \ldots, X_{n_x} \sim N(\mu_x, \sigma_x^2)$ and $Y_1, \ldots, Y_{n_y} \sim N(\mu_y, \sigma_y^2)$. We know that

$$\frac{(n_x - 1)S^2_{n,x}}{\sigma_x^2} \sim \chi^2_{n_x - 1} \quad \text{and} \quad \frac{(n_x - 1)S^2_{n,y}}{\sigma_y^2} \sim \chi^2_{n_y - 1} \, .$$

If we take ratio of the variances, the distribution of this ratio can be obtained

using the so-called F distribution.

If $V_n \sim \chi^2_n$ and $U_m \sim \chi^2_m$ are independent random variables,

$$W_{n,m} = \frac{V_n/n}{U_m/m}$$

has an F distribution with (n, m) degrees of freedom and we denote $W_{n,m} \sim F_{n,m}$. Recall the mean of a χ^2_{df} is the degrees of freedom. Because both V_n/n and U_m/m are averages of independent χ^2_1 random variables with mean n and m, respectively, by the law of large numbers $\lim_{n\to\infty} \frac{V_n}{n} = 1$ and $\lim_{m\to\infty} U_m m = 1$. It is then easy to observe that $W_{n,m}$ is well approximated by 1 as both the number of degrees of freedom of the numerator and denominator, n and m, increase.

11.4.1 Pdfs of F distributions

Let us have a look at the shapes of the $F_{n,m}$ distributions. Figure 11.7 displays the $F_{5,2}$ pdf as a blue solid line. The distribution is heavily skewed with a pdf equal to 0 at 0 and a heavy right tail. The pdf of the $F_{2,5}$ is shown as a solid red line, which is equal to 1 at 0 and is decreasing throughout the range of the distribution. While the distribution is similar to an exponential, it is actually not an exponential pdf. The pdf of the $F_{20,10}$ distribution is shown in violet and it continues to be skewed, though its skew is less heavy than for $F_{5,2}$. Once the number of degrees of freedom starts to increase both in the numerator and denominator the F distribution looks increasingly Normal with a mode that converges to 1. Indeed, most of the probability for the $F_{50,50}$ distribution is in a tight range close to 1.

Let us say we want some rules of thumb for creating confidence intervals for the ratio of variances. For large m we can approximate the denominator by 1 and the distribution of the numerator V_n/n is that of a χ^2_n random variable divided by the number of degrees of freedom. As the distribution of V_n can be approximated by a $N(n, 2n)$ it follows that, for large m,

$$F_{n,m} \approx N\left(1, \frac{2}{n}\right) .$$

This indicates that 95% of the variability of $F_{n,m}$ is within $1 \pm 2\sqrt{2}/\sqrt{n}$. This approximation is very good for moderate n and m.

11.4.2 The F distribution in R

Suppose, for example, that we are interested in obtaining an interval that contains 95% of the probability for the $F_{5,2}$ distribution. There are many such

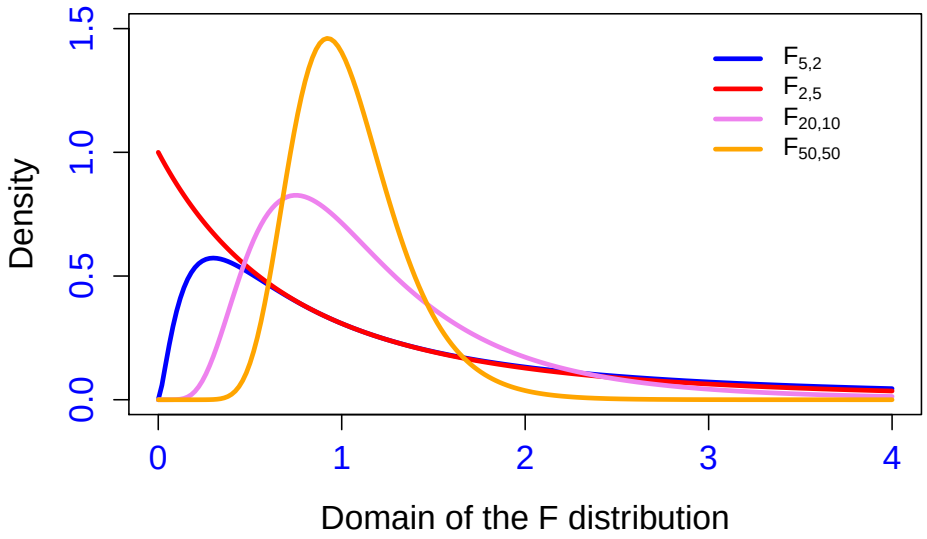


Figure 11.7: Pdfs of the distributions $F_{5,2}$ (blue), $F_{2,5}$ (red), $F_{20,10}$ (violet), and $F_{50,50}$ (orange).

intervals, but here we will focus on the one interval that leaves out equal-tail probabilities. In R this is simple:

```
#Set the number of degrees of freedom for the
#numerator and denominator
df1=5
df2=2
#Set the probability that will be left out
alpha=0.05
#Specify that we need the interval with equal tail probability
interval<-c(alpha/2,0.5,1-alpha/2)
#Obtain the equal-tail probability interval
round(qf(interval,df1,df2),digits=2)
```

```
[1]  0.12  1.25 39.30
```

The 95% confidence interval $[0.12, 39.30]$ is very wide. In the next section, if this was a distribution for a ratio of variances, then 1 would be contained in the interval (not much evidence of a difference). The 50 percentile is 1.25, which is close to 1. Below we provide the table of some representative quantiles. We do not do that because we believe that distributions should be published in textbooks. We just want to show a few examples, some more extreme than others, and what one should expect in practice. Note, for example, how quickly the median converges to 1 and how quickly the $100\alpha\%$ quantiles tighten around 1. In fact, they go to 1 at the rate $1/\sqrt{n}$, where n is the number of degrees of freedom of the numerator. For each example, think of the numerator and

denominator as the sample size of two different groups you are comparing:

Distribution	$q_{0.025}$	$q_{0.5}$	$q_{0.975}$
$F_{2,5}$	0.03	0.80	8.43
$F_{5,2}$	0.12	1.25	39.20
$F_{20,10}$	0.36	1.03	3.42
$F_{50,50}$	0.57	1.00	1.75

11.5 Confidence intervals for variance ratios of Normal distributions

Again, consider the sample variances $S^2_{n,x}$ and $S_{n,y}$ for a collection of iid random variables $X_1, \ldots, X_{n_x} \sim N(\mu_x, \sigma^2_x)$ and $Y_1, \ldots, Y_{n_y} \sim N(\mu_y, \sigma^2_y)$. We know that

$$\frac{(n_x - 1)S^2_{n,x}}{\sigma^2_x} \sim \chi^2_{n_x - 1} \quad \text{and} \quad \frac{(n_x - 1)S^2_{n,y}}{\sigma^2_y} \sim \chi^2_{n_y - 1} \, .$$

Because $S^2_{n,x}$ and $S^2_{n,y}$ are independent

$$\frac{\{(n_x - 1)S^2_{n,x}\}/\{(n_x - 1)\sigma^2_x\}}{\{(n_y - 1)S^2_{n,y}\}/\{(n_y - 1)\sigma^2_y\}} = \frac{\sigma^2_y}{\sigma^2_x}\frac{S^2_{n,x}}{S^2_{n,y}} \sim F_{n_x - 1, n_y - 1} \, ,$$

as this is the ratio between two independent χ^2 distributions with the corresponding number of degrees of freedom. If $f_{n_x-1,n_y-2,\alpha}$ is the α quantile of the F_{n_x-1,n_y-1} distribution then

$$1-\alpha = P\left(f_{n_x-1,n_y-1,\alpha} \leq \frac{\sigma^2_y}{\sigma^2_x}\frac{S^2_{n,x}}{S^2_{n,y}} \leq f_{n_x-1,n_y-1,1-\alpha} \right) = P\left\{ \frac{1}{f_{n_x-1,n_y-1,1-\alpha}}\frac{S^2_{n,y}}{S^2_{n,x}} \leq \sigma^2 \leq \frac{1}{f_{n_x-1,n_y-1,\alpha}}\frac{S^2_{n,y}}{S^2_{n,x}} \right\} \, .$$

We say that the random interval

$$\left[\frac{1}{f_{n_x-1,n_y-1,1-\alpha}}\frac{S^2_{n,y}}{S^2_{n,x}}, \frac{1}{f_{n_x-1,n_y-1,\alpha}}\frac{S^2_{n,y}}{S^2_{n,x}} \right]$$

is a $100(1-\alpha)\%$ confidence interval for σ^2_y/σ^2_x. Note that the inverse of the quantiles multiply the ratio of the observed variances. This is why the interpretation of the quantiles of the F distribution is important. For example, if $n_x = n_y = 51$ and $\alpha = 0.5$ then $1/f_{n_x-1,n_y-1,1-\alpha} = 1/1.75 = 0.57$ and $1/f_{n_x-1,n_y-1,\alpha} = 1/0.57 = 1.75$. This indicates that the confidence interval is

$$\left[0.57\frac{S^2_{n,y}}{S^2_{n,x}}, 1.75\frac{S^2_{n,y}}{S^2_{n,x}} \right] \, .$$

This interval includes 1 if and only if

$$0.57 S^2_{n,x} \leq S^2_{n,y} \leq 1.75 S^2_{n,x} \ ,$$

that is, if one of the observed variances is within 75% of the other one. This exact numbers hold for the case when $n_x = n_y = 51$, but similar interpretations can be obtained for other values. As the number of observations increases the confidence interval will contain 1 under increasingly stringent conditions on the proportion difference between the two empirical variances. The confidence intervals for the ratio of variances are often used for testing the equality of variances of two distributions. In biostatistics this is called analysis of variance (ANOVA).

11.5.1 Examples: comparing variances in the SHHS

We revisit the two examples from the SHHS and construct 95% confidence intervals for the variance ratio for `time_bed` and `rdi4p` in the two groups.

```
group_vars = time_in_bed %>%
  group_by(group) %>%
  summarize(n = n(),
            variance = var(time_bed))
group_vars
```

```
# A tibble: 2 x 3
  group       n variance
  <chr>   <int>    <dbl>
1 Group 1   248    1.01
2 Group 2    97    0.946
```

```
var_ratio = group_vars$variance[2]/group_vars$variance[1]
var_ratio
```

```
[1] 0.9325382
```

The variance for `time_bed` for group 1 is 1.01 and for group 2 is 0.95. Because $n_x = 248$ and $n_y = 97$ we are interested in the 0.025 and 0.975 quantiles of an $F_{96,247}$ distribution. These values are

```
round(qf(c(0.025,0.975),96,247),digits=2)
```

```
[1] 0.71 1.38
```

Therefore the realized 95% confidence interval for σ^2_y/σ^2_x is

$$\left[\frac{1}{1.38} \times \frac{0.95}{1.01}, \frac{1}{0.71} \times \frac{0.95}{1.01}\right] = [0.68, 1.32] \ ,$$

which includes 1 indicating that the equal variances hypothesis cannot be rejected for the `time_bed` for the two groups considered here.

In R confidence intervals can be computed and F tests can be conducted directly without going through every individual derivation using the `var.test` function. Specifically,

```
results_f_test<-var.test(time_bed ~ group, data = time_in_bed)
results_f_test
```

```
    F test to compare two variances

data:  time_bed by group
F = 1.0723, num df = 247, denom df = 96, p-value = 0.7018
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.7576116 1.4803826
sample estimates:
ratio of variances
          1.072342
```

The results of the F test are identical to the ones derived by hand. The F-statistic (ratio of the two variances) is 1.0723 with the number of degrees of freedom of the numerator equal to 247 and of the denominator equal to 96. The F test also produces the 95% confidence interval [0.76, 1.48]. We can again `tidy` the data using `broom`:

```
tidy(results_f_test)
```

```
Multiple parameters; naming those columns num.df, denom.df
```

```
# A tibble: 1 x 9
  estimate `num df` `denom df` statistic p.value conf.low conf.high method
     <dbl>    <int>      <int>     <dbl>   <dbl>    <dbl>     <dbl> <chr>
1     1.07      247         96      1.07   0.702    0.758      1.48 F tes~
# ... with 1 more variable: alternative <chr>
```

We now focus on the square root of `rdi4p` because we have tested the difference in means of these values in the two groups and the distributions are closer to being Normal.

```
group_vars = time_in_bed %>%
  group_by(group) %>%
  summarize(n = n(),
            variance = var(sqrt(rdi4p)))
group_vars
```

```
# A tibble: 2 x 3
  group       n variance
  <chr>   <int>    <dbl>
1 Group 1   248     1.52
```

```
2 Group 2    97     5.34
```

```
var_ratio = group_vars$variance[2]/group_vars$variance[1]
var_ratio
```

```
[1] 3.511078
```

The variance for `rdi4p` for group 1 is 1.52 and for group 2 is 5.34. Because $n_x = 248$ and $n_y = 97$ we use the same 0.025 and 0.975 quantiles of an $F_{96,247}$ distribution. The realized 95% confidence interval for σ_y^2/σ_x^2 is

$$\left[\frac{1}{1.38} \times \frac{5.34}{1.52}, \frac{1}{0.71} \times \frac{5.34}{1.52}\right] = [2.55, 5.45] \,,$$

which does not include 1 indicating that the equal variances hypothesis can be rejected at the $\alpha = 0.05$ level for the square root of the `rdi4p` variable for the two groups. A similar result can be obtained for `rdi4p` instead of the square root of the `rdi4p`. We leave this as an exercise.

Similarly, results can be obtained for the square root of `rdi4p` using the `var.test` function. Specifically,

```
results_f_test <- var.test(sqrt(rdi4p) ~ group, data = time_in_bed)
results_f_test
```

```

    F test to compare two variances

data:  sqrt(rdi4p) by group
F = 0.28481, num df = 247, denom df = 96, p-value = 3.109e-15
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2012208 0.3931879
sample estimates:
ratio of variances
         0.2848128
```

Thus, we see the confidence interval does not overlap 1, so we conclude that the groups likely have different variances.

11.6 Problems

Problem 1. In a trial to compare a stannous fluoride dentifrice A, with a commercially available fluoride free dentifrice D, 260 children received A and 289 received D for a three-year period. The mean DMFS increments (the number of new Decayed Missing and Filled tooth Surfaces) were 9.78 with standard deviation 7.51 for A and 12.83 with standard deviation 8.31 for D. Is this good evidence that, in general, one of these dentifrices is better than the other at

reducing tooth decay? If so, within what limits would the average annual difference in DMFS increment be expected to be?

Problem 2. Suppose that 18 obese subjects were randomized, 9 each, to a new diet pill and a placebo. Subjects' body mass indices (BMIs) were measured at a baseline and again after having received the treatment or placebo for four weeks. The average difference from follow-up to the baseline (followup - baseline) was $-3\ kg/m^2$ for the treated group and $1\ kg/m^2$ for the placebo group. The corresponding standard deviations of the differences was $1.5\ kg/m^2$ for the treatment group and $1.8\ kg/m^2$ for the placebo group. Does the change in BMI over the two-year period appear to differ between the treated and placebo groups? Show your work and interpret results. Assume Normality and a common variance.

Problem 3. Suppose that systolic blood pressures were taken on 16 oral contraceptive (OC) users and 16 controls at baseline and again two years later. The average difference from follow-up SBP to the baseline (followup - baseline) was 11 $mmHg$ for oral contraceptive users and 4 $mmHg$ for controls. The corresponding standard deviations of the differences were 20 $mmHg$ for OC users and 28 $mmHg$ for controls.

a. Calculate and interpret a 95% confidence interval for the change in systolic blood pressure for oral contraceptive users; assume Normality.
b. Does the change in SBP over the two-year period appear to differ between oral contraceptive users and controls? Create the relevant 95% confidence interval and interpret. Assume Normality and a common variance.

Problem 4. In the previous problems does it make sense to assume equal variances? Construct the same confidence intervals without assuming equal variances and compare. Provide `R` code built from scratch and using built-in `R` functions for testing.

Problem 5. Will a Student's t or Z hypothesis test for a mean with the data recorded in pounds always agree with the same test conducted on the same data recorded in kilograms? (explain)

Problem 6. In the SHHS consider the square root of `rdi4p` for two groups: (1) individuals who are 40 to 45 years old, never smoked (`smokstat_s1==0`) and have a BMI below 25; and (2) individuals who are 70 to 80 years old, never smoked, and have a BMI above 35.

a. Calculate the 90% confidence interval for the difference in mean of the square root of `rdi4p` between the two groups under the equal variance assumption.
b. Calculate the 95% confidence interval for the ratio of variances of the square root of `rdi4p` between the two groups
c. Calculate the 90% confidence interval for the difference in mean of the square root of `rdi4p` between the two groups without assuming equal variance.
d. Discuss and interpret the differences.

Problem 7. We consider the same problem as above, where the second group is the same, while the first group comprises individuals who are 70 to 80 years old, are former smokers, and have a BMI above 35.

Problem 8. We consider the same problem as above, where the first group comprises women (`gender=0`) who are 70 to 80 years old, have never smoked, and have a BMI above 35, while the second group comprises men 70 to 80 years old, who have never smoked, and have a BMI above 35.

Problem 9. Compare the confidence intervals obtained in the previous 3 problems with the confidence intervals based on the bootstrap of subjects (using quantiles and mean plus/minus Normal quantiles times the standard deviation of the statistic.)

Problem 10. We would like to compare the performance of confidence intervals for ratio of variances. Simulate 10000 times N_1 random variables from a $N(3, 5^2)$ and N_2 random variables from a $N(1, 7^2)$. For every sample calculate the 95% confidence interval based on the F_{N_1-1,N_2-1} distribution as well as on the bootstrap (use 100000 bootstrap samples for every pair of samples). Consider the following pairs of (N_1, N_2): $(5, 10)$, $(10, 5)$, $(10, 10)$, $(10, 20)$, $(20, 10)$, $(20, 50)$, $(50, 10)$, $(50, 50)$, $(100, 200)$, and $(200, 100)$.

a. Calculate the percentage of times the confidence intervals cover the true value of the variance ratio $7^2/5^2$
b. Calculate and compare the average length of the confidence intervals for the F and bootstrap approximations
c. What are your conclusions?

Problem 11. Repeat the same experiment, but replace the $N(3, 5^2)$ distribution with the distribution of a $3 + t_{2.08}$ random variable (variance=26) and the $N(1, 7^2)$ distribution with $1 + t_{2.04}$ (variance=51). Here we chose the t-distributions to have close variances to the Normals from the previous example, but they are not identical. In general, the variance of the distribution of a t_n random variable is $n/(n-2)$ if $n > 2$ and is infinity if $n \leq 2$.

Problem 12. Show that $f_{n,m,\alpha} = 1/f_{m,n,1-\alpha}$, that is, the α-quantile of the F distribution with (n, m) degrees of freedom is the reciprocal of the $(1-\alpha)$ quantile of the F distribution with (m, n) degrees of freedom.

a. Where in this chapter can this result be observed?
b. What is the practical interpretation of this result?

Problem 13. Write an `R` function from scratch to calculate $100(1-\alpha)\%$ confidence intervals for the difference in mean, ratio of variances, and a recommendation of what type of interval to use (pooled versus un-pooled variance). Produce output that is as neat and intuitive as possible.

Problem 14. We would like to investigate whether the percent of time spent in stages 3 and 4 of sleep varies with age. The variable that contains this

information is `times34p`. Partition the age range into deciles. This can be done using the following `R` code:

```
q_age<-quantile(age_s1,probs=seq(0,1,length=11),na.rm=TRUE)
```

We would like to compare the average `times34p` across the 10 groups; this is often called stratification by a variable, in this case, `age_s1`. To do this we will set the first decile of `age_s1` as the reference category and we will compare every category against the reference category.

a. Conduct the t-test for the mean `times34p` for each age group versus the mean age group of `times34p` in the reference category
b. Verify whether the confidence interval for the variance ratio in each case contains 1
c. Suppose that we knew that there is no difference in average `times34p` as a function of age. Provide an expected number of tests for the difference in mean among the 9 conducted that would not include 0. Also, provide a confidence interval for the number of tests that would not include 0. Interpret.

Problem 15. Conduct the same type of analysis using `bmi_s1` as the stratification variable and the square root of the `rdi4p` as the variable on which we conduct t-testing for equality in the mean. In addition:

a. Calculate the variance of the means of the square root of the `rdi4p` in the 10 groups and the variance of the square root of `rdi4p` around the group means, and denote by F_0 the ratio of these two variances.
b. Repeat the same procedure 10000 times when every time the vector `rdi4p` is randomly permuted and collect the vector $F_1, \ldots, F_{10000}$ of F ratios obtained after each permutation.
c. What is the interpretation of the proportion of times $F_i > F_0$?

Chapter 12

Data resampling techniques

This chapter covers the following topics

- Jackknife and cross validation
- Bootstrap

In this chapter we focus on *data resampling* techniques. People will use these techniques to perform inference, assess bias or quantify uncertainty. Their popularity stems from their being easy to interpret and close to the data. We'll focus on the colorfully named jackknife and bootstrap procedures in this chapter. In another chapter, we'll focus on permutation testing, another resampling technique.

The jackknife and bootstrap were popularized by two of the most impactful modern statisticans: John Tukey for the jackknife (Tukey 1958) and Bradley Efron for the bootstrap (Efron and Tibshirani 1993; Efron 1979). The bootstrap, in particular, has become a revolutionary procedure underlying the basis of much of modern inference and machine learning. The names of the procedures underlie their utility. The jackknife, like its physical counterpart (think of a Swiss-army knife), is a handy and practical tool. The bootstrap is named after the impossible task of "pulling oneself up by their own bootstraps". (The phrase is often attributed to the accounts of the fictional Baron Munchausen. Though, the character, equally impossibly, pulled themselves up by their hair, not bootstraps.)

12.1 Jackknife and cross validation

The jackknife is, as was mentioned, a handy tool for statistical analysis. The technique was officially introduced by Quenouille (Quenouille 1949, 1956), but was popularized and expanded on by Tukey. The technique involves deleting

observations one at a time and calculating a statistic in turn. Imagine that we have a target to estimate (an estimand) and a procedure based on data that produces estimators given a particular size of a data set. Estimators can come from a parametric or nonparametric statistical model, any black-box algorithm, simulations or a large system of partial differential equations. The fundamental beauty of the idea is that it essentially says: "if you can produce an estimator based on a data set of size n then you can produce an estimator based on a data set of size $n-1$ by simply witholding one observation." This procedure can be repeated then by removing, one-at-a-time, the other observations from the data. The "leave-one-out" concept is probably one of the most important concepts in data analysis and the inspiration for the widely used cross-validation concept.

This is a rather strange concept, though new it is not. Indeed, why would anybody even think about using less data to obtain an estimator and then spending the time to do this seemingly sub-optimal procedure many times? The reasons are that: 1) the distribution of estimators obtained by removing one observation at a time provides a sense of the variability of the estimator when the data is just a little smaller; 2) the required time for running the procedure increases only n times (because there are n data sets of size $n-1$); 3) the procedure does not rely on the method used for estimation or prediction; 4) using a technical argument this approach can be used to numerically remove the bias of a large class of estimators (e.g. maximum likelihood estimators); and 5) the flavor, though not the intent, of the approach has deep connections to the widely used bootstrap (for estimating variability) and cross validation (for controlling over-fitting in machine learning). However, as interesting and connected as the jackknife is, it has not become one of the standard techniques in statistical practice. Keeping this in perspective, we focus now on the jackknife estimator of the bias and variance in *finite samples*.

12.1.1 Estimating bias

To go into details, let $X_1, \ldots, X_n$ be a collection of of univariate observations to estimate parameter θ. Let $\hat{\theta}_n$ be the estimate based on the full data and let $\hat{\theta}_{i,n}$ be the estimate of θ obtained by deleting observation i.

For example, if θ is the popuation mean and $\hat{\theta}_n$ is the sample mean $\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right)$, $\hat{\theta}_{i,n}$ would be the mean of the $n-1$ observations omitting observation i $\left(\frac{1}{n-1}\sum_{j\neq i} x_j\right)$. Let $\bar{\theta}_n = \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_{i,n}$ be the average of these leave one out estimators. We might think of the n leave one out estimators as an estimate of the population distribution of the sample statistic (based on $n-1$ observations). Then θ_n is the estimated population mean of this distribution.

Ideally, $\hat{\theta}_n$ and $\bar{\theta}_n$ would be close. In fact, their difference plays a fundamental

role in the *jackknife bias estimate*:

$$(n-1)\left(\bar{\theta}_n - \hat{\theta}_n\right).$$

Consider the sample mean, where $\bar{\theta}_n$ is equal to $\hat{\theta}_n$ suggesting that there is no bias. This is encouraging, since we know that the sample mean is unbiased, $E[\bar{X}] = \mu$ and the estimator of the bias is also $\widehat{b}_n = 0$.

Why does this work? Consider the case where there is an additive bias of order n. That is, $E[\hat{\theta}_n] = \theta + b/n$. Then:

$$E[\bar{\theta}_n] = \frac{1}{n}\sum_{i=1}^{n} E[\hat{\theta}_{i,n}] = \frac{1}{n}\sum_{i=1}^{n}\left\{\theta + \frac{b}{n-1}\right\} = \theta + \frac{b}{n-1}.$$

Note then that our bias estimate is then itself unbiased:

$$E[(n-1)(\bar{\theta}_n - \hat{\theta}_n)] = (n-1)\{E(\bar{\theta}_n) - E(\hat{\theta}_n)\} = (n-1)\left\{\frac{b}{n-1} - \frac{b}{n}\right\} = \frac{b}{n}.$$

The idea of constructing estimates with observations left out led to the key insight that (particular forms of) bias can be estimated using the data in hand. This style of thinking foreshadowed major future developments in data resampling strategies for bias and standard error estimation.

Since we have estimated the bias, it seems clear that we might want to correct for it. The jackknife estimate of θ is:

$$\hat{\theta}_n - (n-1)(\bar{\theta}_n - \hat{\theta}_n)\ ,$$

which is simply the estimate $\hat{\theta}_n$ minus the estimated bias $(n-1)(\bar{\theta}_n - \hat{\theta}_n)$. It is easy to see that the jackknife estimator is unbiased in finite samples:

$$E\left[\hat{\theta}_n - (n-1)(\bar{\theta}_n - \hat{\theta}_n)\right] = \theta + \frac{b}{n} - \frac{b}{n} = \theta$$

One motivation for the jackknife is that the leave one out estimators are a sample from the distribution of estimators (using $n-1$ observations instead of n). Thus, the sample mean from this distribution estimates the population mean of this statistic, leading to an ability to estimate bias.

Once students are exposed to the jackknife idea they often have a simple and reasonable question: "why do we assume that the bias is of the form b/n and what type of estimators have this form?" Indeed, we have just discussed that the mean is unbiased and the jackknife estimator of the zero bias is zero. It turns out that the mean is one of the very few estimators that is unbiased in *finite*

samples. The other very important example are the least square estimators of the regression parameters in a linear regression. But, almost every other reasonable estimator we can think of, can be shown to have a bias, which is typically of order n. Indeed, all Maximum Likelihood Estimators (MLE) have a bias of order n including the variance for a Normal distribution, the maximum for a Uniform on the interval $(0, \theta)$ distribution, the shape and scale of a Gamma distribution, etc. Indeed, while most estimators are *asymptotically unbiased*, that is, the bias goes to zero as the number of observations is going to infinity, they are biased in *finite samples*. The jackknife is a simple method to correct the finite samples bias.

The jackknife is designed to correct only for bias of the type b/n. In the case when the bias has additional, higher order components, the method does not eliminate the bias in finite samples. For example, consider the case when

$$E(\widehat{\theta}_n) = \theta + \frac{b}{n} + \frac{c}{n^2}$$

then

$$E(\bar{\theta}_n) = \frac{1}{n}\sum_{i=1}^{n} E[\hat{\theta}_{i,n}] = \frac{1}{n}\sum_{i=1}^{n}\left\{\theta + \frac{b}{n-1} + \frac{c}{(n-1)^2}\right\} = \theta + \frac{b}{n-1} + \frac{c}{(n-1)^2} \ ,$$

which shows that

$$E[(n-1)(\bar{\theta}_n - \hat{\theta}_n)] = \frac{b}{n} + \frac{(2n-1)c}{n^2(n-1)} \ .$$

Thus, the expected value of the jackknife estimator in this case is

$$\theta + \frac{b}{n} + \frac{c}{n^2} - \frac{b}{n} - \frac{(2n-1)c}{n^2(n-1)} = \theta - \frac{c}{n(n-1)} \ ,$$

indicating that the jackknife estimator has bias $-c/(n^2 - n)$. The linear part of the bias is removed and one could argue that the bias is reduced even though it is not eliminated.

12.1.2 Jackknife motivation using pseudo observations

Alternatively, one can motivate the jackknife using pseudo observations (pseudo meaning "fake"). Define the i^{th} pseudo observation as:

$$\mathrm{ps}_{i,n} = \hat{\theta}_n - (n-1)(\bar{\theta}_{i,n} - \hat{\theta}_n) = n\hat{\theta}_n - (n-1)\hat{\theta}_{i,n} \ .$$

Each of the pseudo observations is unbiased to the original target parameters even if the original observations are not. Indeed,

$$E(\mathrm{ps}_{i,n}) = E(\widehat{\theta}_n) - (n-1)\{E(\bar{\theta}_{i,n}) - E(\widehat{\theta}_n)\} = \theta + \frac{b}{n} - (n-1)\left\{\theta + \frac{b}{n-1} - \theta - \frac{b}{n}\right\} = \theta \ .$$

So, in some sense, the pseudo observations are trying to play the same role as the original observations. Indeed, pseudo observations replace the actual observations, X_i, with another set of n observations, $\text{ps}_{i,n}$, with the property that the new, pseudo, observations are unbiased to the true estimand, θ; that is $E(\text{ps}_{i,n}) = \theta$. In the case of the sample mean, pseudo observations are equal to the original observations as we show below, but this is not the case with all estimators. Surprisingly, this pseudo observation procedure can be done to any type of estimator with this type of finite sample bias and only requires running the algorithm n more times than usual. It is easy to show that the jackknife estimator of θ is the average of the pseudo observations.

Think of these values as what observation i contributes to the estimate. In the case where the estimate is the sample mean, the pseudo observations work out to be the observations themselves. Indeed, $n\widehat{\theta}_n = \sum_{k=1}^n X_k$ and $(n-1)\widehat{\theta}_{i,n} = \sum_{k=1}^n X_k - X_i$, which makes the pseudo observations for the mean equal to

$$\text{ps}_{i,n} = \sum_{k=1}^n X_k - \left(\sum_{k=1}^n X_k - X_i \right) = X_i \ .$$

In the general case, the mean of the pseudo observations is the bias corrected estimate. Indeed,

$$\begin{aligned} \frac{1}{n} \sum_{i=1}^n \{ n\widehat{\theta}_n - (n-1)\widehat{\theta}_{i,n} \} &= \frac{n^2 \widehat{\theta}_n}{n} - (n-1) \left\{ \frac{1}{n} \sum_{i=1}^n \widehat{\theta}_{i,n} \right\} \\ &= n\widehat{\theta}_n - (n-1)\bar{\theta}_n \\ &= \widehat{\theta}_n - (n-1) \left(\bar{\theta}_n - \widehat{\theta}_n \right) \ . \end{aligned}$$

12.1.3 Estimating standard errors using the jackknife

The jackknife is useful for estimating the standard error of a statistic as well. Thinking along the lines of the sample of leave-one out observations as being a sample of the statistic of interest (having $n-1$ observations instead of n), their standard deviation must yield information about the standard deviation of the statistic. More specifically, if we can assume that the variance approximately scales with $1/n$, i.e. $Var(\hat{\theta}_n)/Var(\hat{\theta}_{i,n}) = (n-1)/n$, then $Var(\hat{\theta}_n) = \frac{n-1}{n} Var(\hat{\theta}_{i,m})$. But, note we can estimate the variance of $Var(\hat{\theta}_{i,m})$, since we have n of them (obtained by leaving out each observation). Thus, an estimate of the standard deviation is:

$$\left[\frac{n-1}{n} \sum_{i=1}^n (\hat{\theta}_{i,n} - \bar{\theta}_n)^2 \right]^{1/2} ,$$

the estimated variance from the leave one out estimates. This also works out to be the standard deviation of the psuedo observations mentioned earlier!

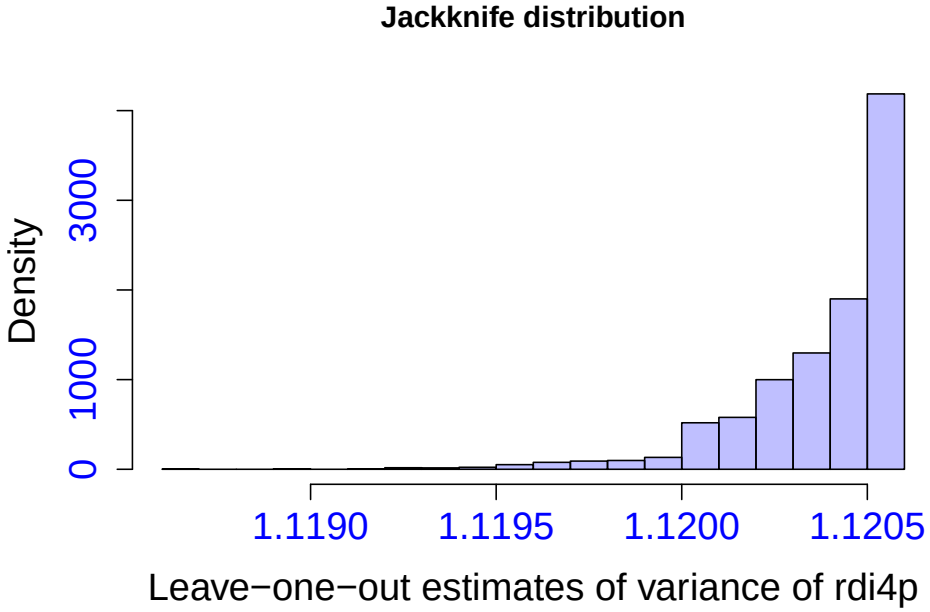
12.1.3.1 Example: RDI

Let us go over an example of using the jackknife in practice using the SHHS data. We will investigate the respiratory disturbance index (RDI). Because it is highly skewed we take the log transformation. However, because it has 0 values, a constant (1 in this case) is added to transform zeros into zeros.

```
## read in the data
dat = read.table(file = "data/shhs1.txt",
                 header = TRUE, na.strings = "NA")
logp1rdi = log(dat$rdi4p + 1)
```

Let us use the biased version of the variance as a statistic $\left(\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)$. First we will create our leave-one-out estimates and then plot their histogram.

```
n = length(logp1rdi)
bivar = function(x) {
  n = length(x)
  (n - 1) * var(x) / n
}
jk = sapply(1 : n, function(i) bivar(logp1rdi[-i]))
hist(jk,main="Jackknife distribution",
     xlab="Leave-one-out estimates of variance of rdi4p",
     probability=TRUE,col=rgb(0,0,1,1/4),breaks=20,
     cex.lab=1.5,cex.axis=1.5,col.axis="blue")
```



The distribution of the jackknife estimators of the variance of the log of `rdi4p`

is highly skewed and concentrated. Indeed, the minimum and maximum of the distribution are 1.1187 and 1.1206 indicating that no particular observation has a large effect on the variance estimator.

Let us perform the jackknife and estimate the bias.

```
theta = bivar(logp1rdi) # the biased estimate
theta
```

```
[1] 1.12039
```

```
thetaBar = mean(jk) # the mean of the jackknife estimates
biasEst = (n - 1) * (thetaBar - theta) # the estimate of the bias
biasEst
```

```
[1] -0.0001930709
```

Now let's calculate the bias corrected version of the statistic.

```
theta - biasEst ## the bias corrected version
```

```
[1] 1.120583
```

```
var(logp1rdi) ## using the unibiased variance estimate
```

```
[1] 1.120583
```

So, interestingly, the jackknife bias corrected version of the variance is exactly the unbiased variance formula. This is always the case, but it does show that our jackknife is doing exactly what it is trying to do, converts the biased estimate into an unbiased one. It does not always work this well. We will see that it fails particularly for sample quantiles.

Let us also calculate the jackknife estimated standard error of the variance:

```
sqrt((n - 1) * mean((jk - thetaBar)^2))
```

```
[1] 0.01668013
```

You do not have to program the jackkife, as it is included in the `bootstrap` package. The arguments are the vector of interest and a function that performs the estimation. The function outputs a list with the bias and standard error estimates.

```
library(bootstrap)
```

```
Attaching package: 'bootstrap'

The following object is masked from 'package:broom':

    bootstrap

The following object is masked from 'package:VGAM':
```

```
    hormone
out = jackknife(logp1rdi, bivar)
out$jack.bias
```

```
[1] -0.0001930709
```

```
out$jack.se
```

```
[1] 0.01668013
```

12.1.4 The jackknife and leave-one-out cross validation

So far, we have discussed the jackknife as a method for reducing bias and estimating the variance of an estimator. A related method that is often called the jackknife is used to assess the generalization performance of regression (Abdi and Williams 2010). Consider the case when we see pairs of random variables (X_i, Y_i) and we try to predict Y_i from X_i. For example, we would like to predict `rdi4p` from `bmi_s1` in the SHHS. A possible strategy is to do a linear regression (not yet covered) of the type

$$Y_k = a_n + b_n X_k + \epsilon_k \ , \text{ for } k = 1, \ldots, n \ ,$$

where $\epsilon_k \sim N(0, \sigma_\epsilon^2)$ are errors. A predictor of Y_k based on X_k can be obtained by fitting this model, obtaining the estimates $\widehat{a}_n$ and $\widehat{b}_n$ of a_n and b_n, respectively, and then calculating the prediction

$$\widehat{Y}_i = \widehat{a}_n + \widehat{b}_n X_i \ ,$$

where we used k as the generic index and i as a specific index that we are trying to emphasize. At first pass, this seems to be an excellent idea, but it is actually cheating. Indeed, both $\widehat{a}_n$ and $\widehat{b}_n$ contain information about Y_i because Y_i was used to calculate the coefficients to start with. This does not seem like a very good idea, as, if we are allowed to use Y_i in its own prediction then a much better predictor of Y_i is Y_i; also, perfectly non-generalizable to a new predictor observation, X_i. The jackknife procedure fits instead the model

$$Y_k = a_{i,n} + b_{i,n} X_k + \epsilon_k \ , \text{ for } k = 1, \ldots i-1, i+1, \ldots n \ ,$$

and produces the estimators $\widehat{a}_{i,n}$ and $\widehat{b}_{i,n}$ of $a_{i,n}$ and $b_{i,n}$, respectively and then produces the jackknife predictors

$$\widehat{Y}_{i,\text{jack}} = \widehat{a}_{i,n} + \widehat{b}_{i,n} X_i \ .$$

It is standard to calculate the squared error loss for the jackknife predictors as

$$\sum_{i=1}^{n} (Y_i - \widehat{Y}_{i,\text{jack}})^2 \ ,$$

though other losses can be considered, as well. Note that there is a strong connection between the two jackknife methods, but they are not the same. Indeed, it is the latter procedure that has been used extensively in machine learning (called leave one out cross-validation (LOOCV)), as ideas are widely generalizable. In the jackknife, we would get the bias-corrected versions of $\hat{\alpha}_n$ and $\hat{\beta}_n$ and getting $\hat{Y}$ and the loss from these estimates. In the second jackknife procedure, we calculate $\widehat{Y}_{i,\text{jack}}$ for each i and then get the losses from these predicted values. Also, the jackknife is computationally cheap for certain types of estimators where we can do algebraic tricks where we do not need to recompute estimates for each i.

Indeed, there is no need to use a linear regression or a regression. The method can be used with any algorithm that based on a set of predictors produces predictions about a variable of interest. Moreover, models can be compared, selected, or combined using the jackknife sum of square loss or any other loss function. In modern Biostatistics and Machine Learning few would call this procedure the jackknife; instead, the term leave-one-out cross validation is preferred. Versions of this approach include leave-k-out and k-fold cross validation, which may be more robust and also require fewer computations. Knowing that these ideas originated in the now little known works of Quenoullie and Tukey is one of the roles of this book.

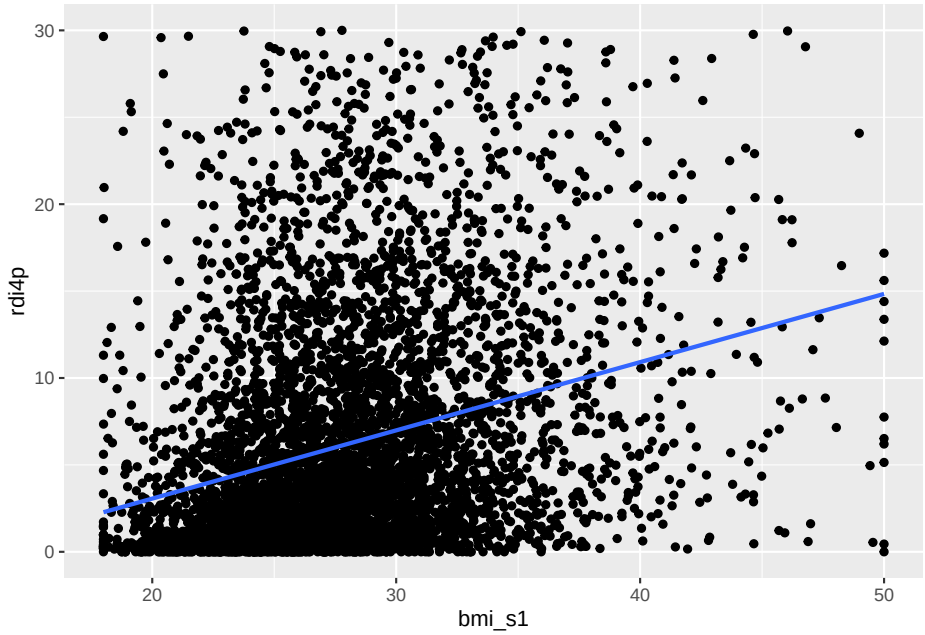
12.1.4.1 Example: jacknifing in regression

Let us investigate a little closer the association between `rdi4p` and `bmi_s1` in SHHS. We have seen the scatter plot many times before and we know that the association is not particularly linear, but the purpose of this example this imperfect assumption will do. Below we display the points indicating the (X_i, Y_i) pairs, where X_i stands for the ith observed `bmi_s1` value and Y_i stands for the ith observed `rdi4p` value. A few observations are skipped because they miss the `bmi_s1` value, but this is just a small problem we need to take care off when doing the accounting. We also fit a linear regression and display the best linear regression line as a red solid line. Not the most central curve to the data, but certainly the line that minimizes the sum of squares for errors.

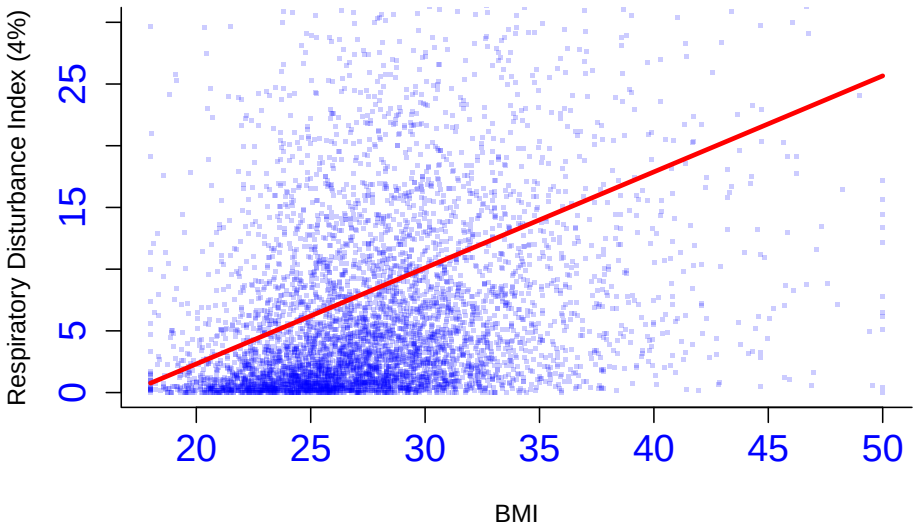
```
fit_df = dat %>%
  filter(!is.na(rdi4p) & !is.na(bmi_s1))
fit_df %>%
  ggplot(aes(x = bmi_s1, y= rdi4p)) +
  geom_point() +
  geom_smooth(se = FALSE, method = "lm") +
  ylim(c(0, 30))
```

```
Warning: Removed 347 rows containing non-finite values (stat_smooth).
```

```
Warning: Removed 347 rows containing missing values (geom_point).
```




```
plot(dat$bmi_s1,dat$rdi4p,pch=".",col=rgb(0,0,1,.2),cex=3,ylim=c(0,30),
     bty="l",cex.axis=1.5,col.axis="blue",main=NULL,xlab="BMI",
     ylab="Respiratory Disturbance Index (4%)")
fit<-lm(rdi4p~bmi_s1,data=dat)
fitt<-fit$fitted.values
bmi_non_na<-dat$bmi_s1[!is.na(dat$bmi_s1)]
rdi_non_na<-dat$rdi4p[!is.na(dat$bmi_s1)]
index_min<-which.min(bmi_non_na)
index_max<-which.max(bmi_non_na)
lines(c(bmi_non_na[index_min],bmi_non_na[index_max]),
      c(fitt[index_min],fitt[index_max]),
      col="red",lwd=3)
```



We would like to produce the jackknife, or leave-one-out, predictors and compare them to the predictors based on the entire data set . We will do this only for the first 5 observations, though this could be done the same way for all observations in the data (N = 5761). We do not do this here because it is relatively slow and there are very fast approaches for doing this much better; alas, they exceed the scope of the current book.

```
#Build the matrix of results that will be displayed
full_mod = lm(rdi4p ~ bmi_s1, data = fit_df)
results = fit_df %>%
  select(bmi_s1, rdi4p) %>%
  dplyr::slice(1:5)
results$pred = predict(full_mod, newdata = results)
results$pred_jack = NA

#Conduct the jackknife prediction, or leave one-out cross validation
for (i in 1:5){
  #Identify the point where prediction is made
  run_df = fit_df[-i, ]
  mod = lm(rdi4p ~ bmi_s1, data = run_df)
  results$pred_jack[i] = predict(mod, newdata = fit_df[i,])
}
colnames(results) = c("BMI", "rdi4p", "Complete case prediction", "Jackknife prediction")
round(results, digits = 4)
```

```
      BMI   rdi4p Complete case prediction Jackknife prediction
1 21.7776  1.4381                  3.7030               3.7040
2 32.9507 17.8022                 12.3963              12.3945
3 24.1141  4.8536                  5.5210               5.5212
```

```
4 20.1852  0.7973                2.4641                2.4651
5 23.3091  2.7568                4.8946                4.8953
```

The first and second columns contain the first 5 observations of `bmi_s1` and `rdi4p`, respectively, while the third column contains the prediction of `rdi4p` based on `bmi_s1` when all observations are considered in the regression equation. The last column contains the same prediction but obtained after removing the corresponding observation from the regression. For example, for the third observation the value 24.11 for `bmi_s1` and 4.85 for `rdi4p` were removed from the data before conducting the regression. In this case the two predictions are very close to each other, but not identical. Observations that would have a large effect on the linear regression if it were removed are called high leverage points. The reason why the two predictions are close for all cases is that none of the first 5 observations are a high leverage points. This is partially due to the fact that the data set is relatively large and there is nothing particularly peculiar about these 5 observations. Things are different in smaller data sets or when values in the predictors or outcomes are different from the rest of the points, which can make them highly influential.

12.2 Bootstrap

The bootstrap is a tremendously useful tool for calculating bias, constructing confidence intervals and calculating standard errors for difficult statistics. It is difficult to overstate the impact that the bootstrap has had on modern statistics. As a simple example, how would one create a confidence interval for the median? Note, asymptotic results and special cases exist to do exactly this. However, the bootstrap solves the problem generally without resorting to the mathematical difficulties of asymptotic statistics or specific models.

The bootstrap works on the so-called bootstrap principle, also often called the plug-in principle. Let us consider the task of finding the standard deviation of a statistic, let's say $\hat{\theta}(x)$, that is trying to estimate a parameter, θ. Here, x is our data and we write $\hat{\theta}(x)$ to denote that our estimator depends on our data. Imagine, if we could simulate from the distribution of the data. Then, as we have done countless times in the text, we could get the standard error of the statistic via simulation.

As an example, imagine if our data of 100 observations are drawn from a $N(\mu, \sigma^2)$ and we want to estimate the standard error of the median. Let us do an example where $\mu = 0$ and $\sigma^2 = 1$.

```
set.seed(32456)
n = 100
nosim = 1000
# make a matrix of 1000 samples of size 100
mat = matrix(rnorm(nosim * n), nosim, n)
```

```
# calculate the median of each sample
medns = matrixStats::rowMedians(mat)
# now we have a 1,000 medians
length(medns)
```

```
[1] 1000
```

```
# the standard deviation of the medians
sd(medns)
```

```
[1] 0.1245937
```

Here we just did 1000 simulations of the data, then calculated the median in each simulation. Our estimates are more accurate as we increase the number of simulations. Of course, this process isn't very useful in practice as we don't usually know the exact data-generating process. If we could simulate from the data generating distribution, we would not need our statistic at all! This is where the bootstrap principal comes in. Instead of simulating from the data generating distribution, we simulate from our best estimate of it. This is the so-called plug-in rule; we plug in the empirical distribution for the real distribution.

So, if we have a dataset of size 100, we simulate a lot of samples of size 100 from the empirical distribution. Of course, there are many ways to estimate the empirical distribution to simulate from it. The easiest way is to use a distribution that puts probability $1/n$ on every observed data point. This is the classic (non-parametric) bootstrap.

Steps for bootstrap standard error estimation given a data set of size n:

1. Simulate lots of samples of size n by drawing from the observed data *with replacement*.
2. Calculate the statistic value for each resample.
3. Calculate the standard error of the simulated statistics.

Let us calculate the median RDI for the SHHS data set. Recall, we read it in when discussing the jackknife. First, let us define our data

```
x = dat$rdi4p
any(is.na(x)) # Note there are no missing
```

```
[1] FALSE
```

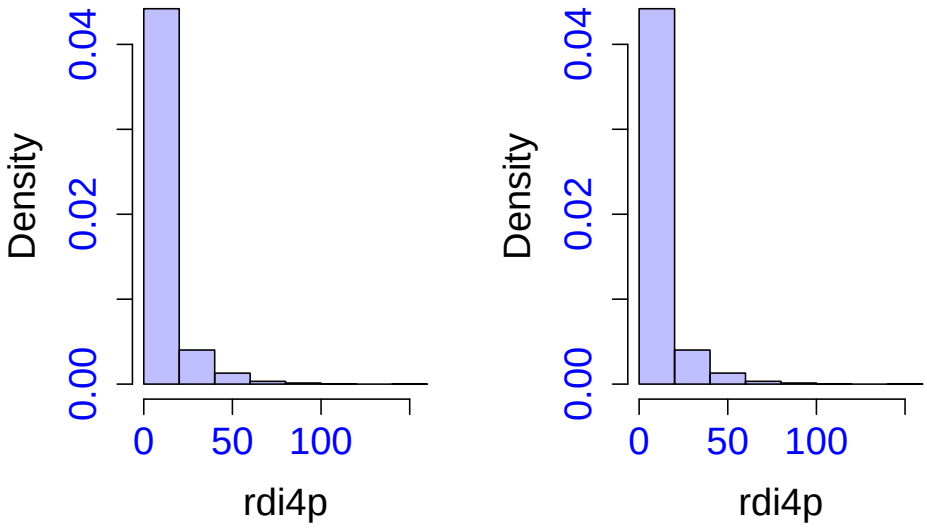
```
n = length(x)
nosim = 1000
thetaHat = median(x)
```

We perform the bootstrap resampling and plot a histogram of the simulated data across all samples and compare it to the histogram of the original data.

```
## sample nosim resamples of size n with replacement
## and put them in a matrix with nosim rows and n columns
```

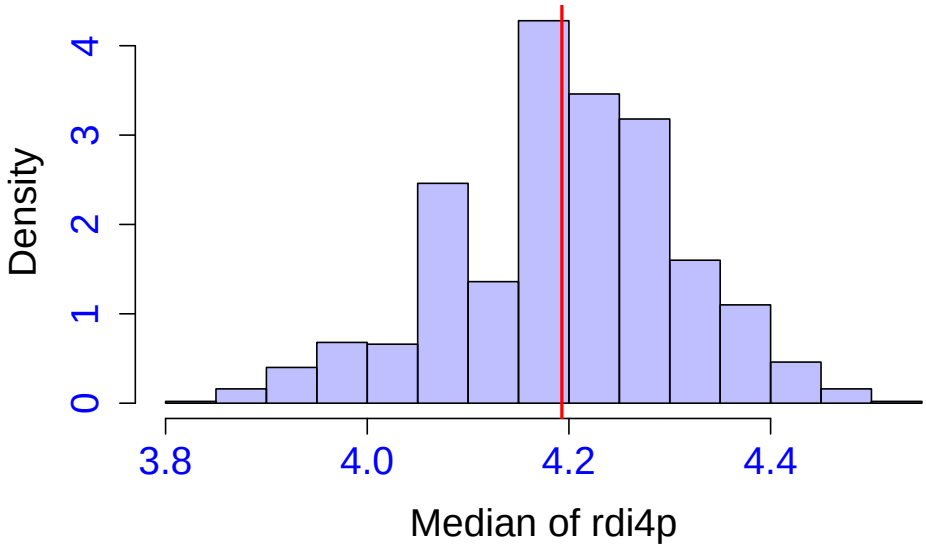
```
## so that each row is a resampled data set
bsResamples = matrix(sample(x, n * nosim, replace = TRUE),
                     nosim, n)

par(mfcol = c(1, 2))
## plot the histogram of the data
hist(x, probability=TRUE,col=rgb(0,0,1,1/4),breaks=10,
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",
     xlab="rdi4p",main="")
## plot a histogram of the resampled data
hist(bsResamples, probability=TRUE,col=rgb(0,0,1,1/4),breaks=10,
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",
     xlab="rdi4p",main="")
```



They look similar, as they should, because the right histogram was effectively created by simulating from the left histogram. To perform the bootstrap, we calculate the statistic (median) of each resample. Now let us plot a histogram of our 1000 bootstrap resamples. We place a red line where the observed statistic (full data median) is located

```
## calculate the median for each
bsStat = rowMedians(bsResamples)
## plot a histogram of our resampled statistics
hist(bsStat,probability=TRUE,col=rgb(0,0,1,1/4),
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",
     xlab="Median of rdi4p",main="")
abline(v = thetaHat, col = "red", lwd = 2)
```



By comparing the range of the X axis with that of the previous histograms it shows that the median is a lot less variable than the population. This because the variance of a statistic is going to be less than that of the population! (Just like the variance of $\bar{X}$ is σ^2/n.) This estimated distribution of the statistic is then used as if we had simulated from the real distribution. So, for example, the estimated standard error of the statistic is just the standard deviation of our bootstrap resampled medians.

```
sd(bsStat)
```

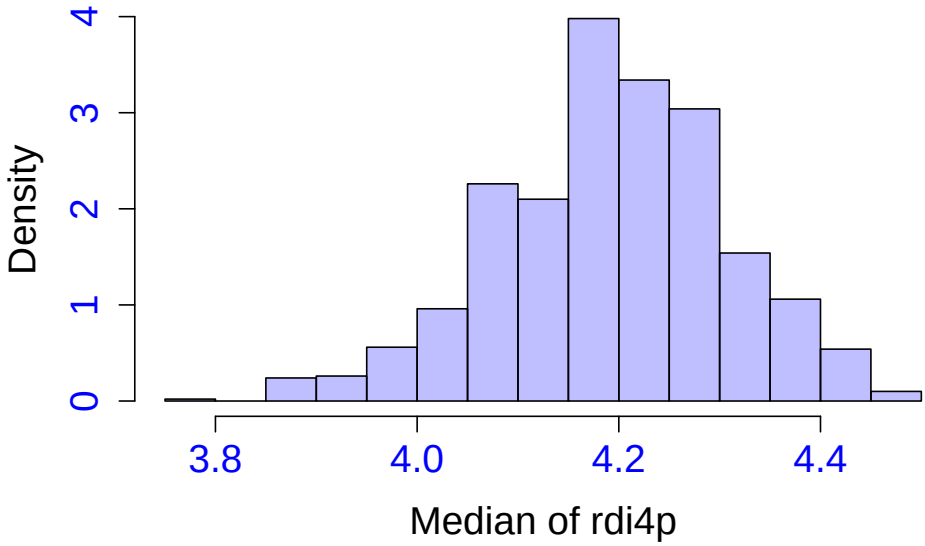
```
[1] 0.1156307
```

Technically, the bootstrap does not require simulation. The exact distribution simply equally weights all possible resamples of the data. This set is too unwieldy to handle exactly, so all bootstrap applications use simulation. Instead of programming this simulation ourselves, as above, we can use the `bootstrap` package for this calcuation.

```
out = bootstrap(x, 1000, median)
sd(out$thetastar)
```

```
[1] 0.1149204
```

```
hist(out$thetastar,probability=TRUE,col=rgb(0,0,1,1/4),
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",
     xlab="Median of rdi4p",main="")
```

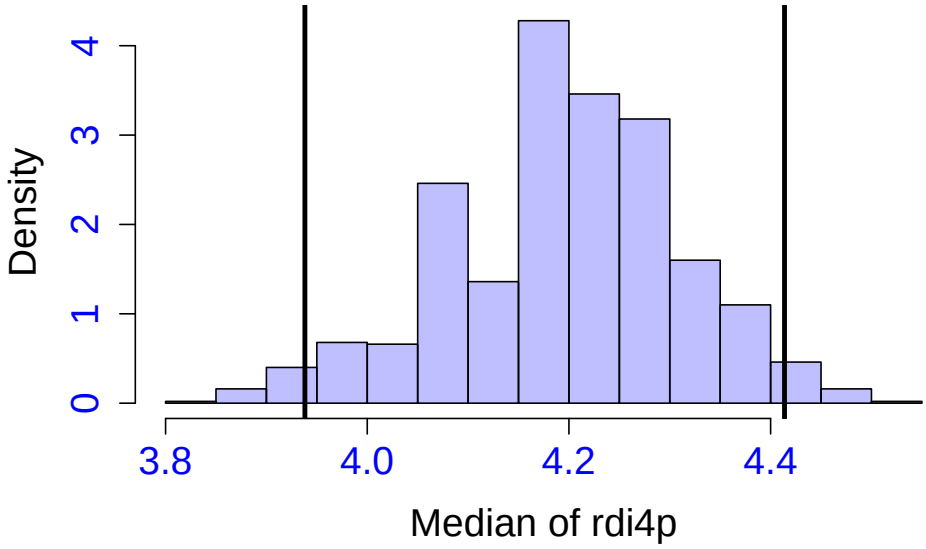


The bootstrap uses the logic of the plugin principle to estimate population quantities without onerous mathematics. The bootstrap relies on a theoretical foundation that shows that this style of analysis works, *provided your n is large enough*. This latter point is important, as the bootstrap does not have finite sample guarantees. Many students ask "how many bootstrap samples are enough?", which there is a definite, but unsatisfying answer. For each finite data set, there are n^n enumerable bootstrap samples, but this means there are a billion bootstrap samples when $n = 10$! The answer generally given is to take as many bootstrap samples as you can, but commonly people take anywhere from 10,000 to 100,000 samples depending on the complexity of calculation of the statistic. These numbers vary greatly also depending on the problem so you should see how the variability of your statistic changes with the number of bootstrap samples.

12.2.1 Bootstrap confidence intervals

The bootstrap not only gives us a way to estimate the standard error of a statistic, we can form confidence intervals. There are several procedures, but the easiest is to take percentiles of the bootstrapped distribution of the statistic. We show a picture of this below for the RDI data.

```
hist(bsStat,probability=TRUE,col=rgb(0,0,1,1/4),
     cex.lab=1.5,cex.axis=1.5,col.axis="blue",
     xlab="Median of rdi4p",main="")
abline(v = quantile(bsStat, c(.025, .975)), lwd = 3)
```



The interval is given below:

```
quantile(bsStat, c(.025, .975))
```

```
    2.5%    97.5%
3.938171 4.413793
```

Such bootstrap percentile intervals have the correct coverage asymptotically, under assumptions. However, a better interval can be constructed using a bias corrected confidence interval in the package `boot`. Let us try a bootstrapped interval for the standard deviation, since the correction does not help for the median we have been discussing so far. The `bcanon` function will perform non-parametric bias-corrected confiedence intervals:

```
sd(x)
```

```
[1] 12.43283
```

```
out = bcanon(x = x, nboot = 1000, theta = sd, alpha = c(.025, .975))
out$confpoints
```

```
     alpha bca point
[1,] 0.025  11.84043
[2,] 0.975  13.15082
```

12.2.2 Summary

The bootstrap is an incredibly useful tool for approximating the distribution of a statistic. We have also seen that it is extremely easy to apply, which lends itself to potential abuse. Remember, it is an asymptotic distribution. For example,

in our study, we investigated RDI, which has a strange distribution including many 0s (so called zero inflation). As such, the asymptotics of the bootstrap may apply as well.

12.3 Problems

Problem 1. Consider the case when $X_1, \ldots, X_n$ are iid $N(\mu, \sigma^2)$. We know that the MLE of σ^2 is

$$S_n^2 = \frac{1}{n}\sum_{i=1}^{n}(X_i - \overline{X}_n)^2 \, .$$

a. Show that

$$E(S_n^2) = \sigma^2 + \frac{b}{n}$$

and obtain b in closed form
b. Obtain the jackknife estimator of σ^2 based on the biased estimator S_n^2
c. Obtain the jackknife estimator for the standard deviation of the jackknife estimator

Problem 2. Consider the case when $X_1, \ldots, X_n$ are iid $\mathrm{U}(0, \theta)$. We know that the MLE of θ is

$$W_n = \max_{i=1,\ldots,n} X_i.$$

a. Show that

$$E(W_n) = \theta + \frac{b}{n}$$

and obtain b in closed form
b. Obtain the jackknife estimator of θ based on the biased estimator W_n
c. Obtain the jackknife estimator for the standard deviation of the jackknife estimator

Problem 3. List all possible bootstrap realizations of the vector $(1, 2, 3)$. How many of these realizations are unique up to a permutation? We say that $(1, 2, 3)$ and $(3, 2, 1)$ are the same up to a permutation.

a. What is the probability of obtaining $(2, 2, 2)$ in the first sample?
b. Suppose that we are running 6 bootstrap samples. What is the probability that at least two vectors are identical?

Problem 4. For a vector of sample size n how many bootstrap realizations are possible? How many of these realizations are unique up to a permutation?

a. What is the probability of obatining $(2, 2, \ldots, 2)$ in the 10-th bootstrap sample?
b. Suppose that we are running 6 bootstrap samples. What is the probability that at least two vectors are identical?

Problem 5. We would like to study the properties of the bootstrap confidence intervals in small samples.

a. Simulate a sample of size 10 from a $N(2, 9)$ distribution and construct the 95% confidence intervals for μ based on the t-statistic and using 10000 bootstrap samples. Repeat the experiment 100000 times and compare the frequency with which each confidence interval covers the true value $\mu = 2$.

b. Simulate a sample of size 10 from a $N(2, 9)$ distribution and construct the 95% confidence intervals for σ^2 based on the χ^2-statistic and using 10000 bootstrap samples. Repeat the experiment 100000 times and compare the frequency with which each confidence interval covers the true value $\sigma^2 = 9$.

c. Repeat the same simulation experiments by varying the sample size from 10, 50, to 100.

Problem 6. Provide a jackknife estimator that is unbiased in the case when the estimator has the following form of the bias

$$E(\widehat{\theta}_n) = \theta + \frac{b}{n} + \frac{c}{n^2} \, .$$

Problem 7. Conduct jackknife for prediction of `rdi4p` from `bmi_s1` using only the first 10 observations in the SHHS and compare your results with the ones obtained using the entire data set.

Problem 8. Show that the jackknife estimator is equal to the mean of the pseudo observations and that jackknife estimator of the standard deviation is the standard deviation of the psuedo observations.

Problem 9. Consider the regression where the outcome is moderate to severe sleep apnea defined as `rdi4p` greater or equal to 15 events per hour.

```
MtS_SA=dat$rdi4p>=15
```

Consider the regression where gender, age, hypertension, BMI, and the interaction between hypertension and BMI are predictors.

```
summary(glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1+bmi_s1*HTNDerv_s1,
            data=dat,family="binomial"))
```

```
Call:
glm(formula = MtS_SA ~ gender + age_s1 + bmi_s1 + HTNDerv_s1 +
    bmi_s1 * HTNDerv_s1, family = "binomial", data = dat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1134  -0.6559  -0.4437  -0.2633   2.8573
```

```
Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)       -8.741853   0.434147 -20.136   <2e-16 ***
gender             1.155232   0.079212  14.584   <2e-16 ***
age_s1             0.032565   0.003702   8.797   <2e-16 ***
bmi_s1             0.148589   0.010730  13.848   <2e-16 ***
HTNDerv_s1         0.704645   0.431328   1.634    0.102
bmi_s1:HTNDerv_s1 -0.017233   0.014119  -1.221    0.222
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5314.5  on 5760  degrees of freedom
Residual deviance: 4659.1  on 5755  degrees of freedom
  (43 observations deleted due to missingness)
AIC: 4671.1

Number of Fisher Scoring iterations: 5
```

Note that in this model neither hypertension nor the interaction between hypertension and BMI are significant at the level $\alpha = 0.05$. We would like to know what sample size would be necessary to obtain significant results for the interaction effect with high probability, say $1 - \beta = 0.90$. Conduct the following analysis

a. Consider a sequence of sample sizes $N^{\text{up}}(p) = (1 + p)N$, where $N = 5804 - 43$, the number of observations without missing covariates and $p = 0, 0.01, \ldots, 0.5$.
b. For every value of p sample with replacement from the data $N^{\text{up}}(p)$ observations, run the regression, and record whether the interaction effect is significant or not at $\alpha = 0.05$
c. Repeat the experiment 10000 times for each value of p and record the proportion of times, $S(p)$, the interaction effect is significant at $\alpha = 0.05$
d. Plot p versus $S(p)$ and obtain the smallest p that ensures $S(p) \geq 0.9$
e. Interpret $S(p)$ and explain what would happen if instead of 10000 simulations for each p we would do only 100 simulations

Problem 10. Consider a slightly different regression with the same outcome and covariates gender, age, hypertension, BMI.

```
summary(glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1,
            data=dat,family="binomial"))
```

```
Call:
glm(formula = MtS_SA ~ gender + age_s1 + bmi_s1 + HTNDerv_s1,
```

```
    family = "binomial", data = dat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0350  -0.6558  -0.4447  -0.2672   2.8334

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -8.478417   0.374489 -22.640   <2e-16 ***
gender       1.160011   0.079171  14.652   <2e-16 ***
age_s1       0.032785   0.003695   8.873   <2e-16 ***
bmi_s1       0.139142   0.007386  18.839   <2e-16 ***
HTNDerv_s1   0.186753   0.077047   2.424   0.0154 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5314.5  on 5760  degrees of freedom
Residual deviance: 4660.6  on 5756  degrees of freedom
  (43 observations deleted due to missingness)
AIC: 4670.6

Number of Fisher Scoring iterations: 5
```

Note that in this model hypertension is significant at the level $\alpha = 0.05$ with a p-value=0.05. We would like to know at what sample size the hypertension would stop being significant with high probability, say $1 - \beta = 0.9$. We will repeat the same experiment as before, though instead of $N^{\text{up}}(p)$ we will consider $N^{\text{down}}(p) = (1 - p)N$, where $p = 0, 0.01, \ldots, 0.5$.

The two problems show a very useful, though never used idea for sample size calculations when preliminary data are available. The idea is exactly the same as that of the bootstrap, but instead of sampling with replacement a data set of the same size we sample a data set of larger or smaller size to see what would happen with the significance of the test. Indeed, there is nothing set in stone about the size of the sample with replacment.

Chapter 13

Taking logs of data

This chapter covers the following topics

- Brief review
- Taking logs of data
- Interpret logged data
- Interpretation of inferences for logged data

13.1 Brief review

In this chapter, we discuss the issue of taking the logarithm, shortened to just log, of outcome data. Throughout, when we write log we are always referring to a logarithm base e. To briefly review, $\log(x)$ is the number y so that $e^y = x$ where e is Euler's number, 2.718... In other words, $\log(x)$ is the inverse of the function e^y. For each (or any) different base, say 10, we write $\log_{10}(x)$ as the number so that $10^y = x$. Thus, $\log_{10}(100) = 2$. The primary bases of use in statistics are: 10, 2 and e. Base 10 is useful as it gives us orders of magnitude from the number system that we are most used to (e.g., cost data in thousands of dollars). Base 2 is less used, but is perhaps more useful, being the smallest whole number for which we can define log and powers of 2 being easy to remember. Also, in genomics, base 2 is used to see two-fold changes in expression. Base e has many mathematical properties that make it attractive. For example, e^y is the only function that is its own derivative, has a simple Taylor expansion, and e is the limit of $(1 + 1/n)^n$, a useful expression used in calculating compound interest.

Recall, $\log(1) = 0$ and $\log(x)$ approaches $-\infty$ as x approaches 0. Notice, if $\log(x) = a$ and $\log(y) = b$, $xy = e^{a+b}$ and so $\log(x) + \log(y) = \log(xy)$. Also notice, if $x = e^a$ then $1/x = e^{-a}$ so that $\log(1/x) = -\log(x)$ and $\log(x/y) = \log(x) - \log(y)$. Similarly, $\log(x^y) = y\log(x)$, since if $e^b = x$ then $e^{yb} = x^y$.

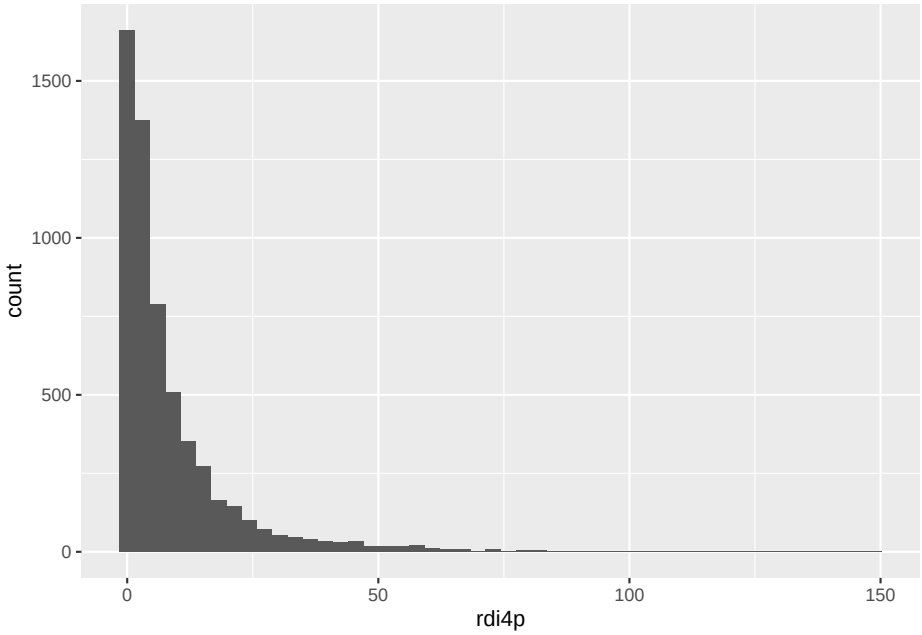


Figure 13.1: Histogram of `rdi4p`.

13.2 Taking logs of data

Typically, one takes logs of data to correct for skewness. Figure 13.1 provides the specific example of the histogram of the respiratory disturbance index, `rdi4p`, in the SHHS.

```
dat = read.table(file = "data/shhs1.txt",
                 header = TRUE, na.strings=".")
library(tidyverse)
dat %>% ggplot(aes(rdi4p)) + geom_histogram(bins = 50)
```

Consider now the distribution of $\log(r+1)$ where r is RDI. Here the plus 1 is useful as RDI is often 0 and $\log(0+1) = \log(1) = 0$. Figure 13.2 displays the transformed `rdi4p` data.

```
dat = dat %>% mutate(logp1rdi = log(dat$rdi4p + 1))
dat %>% ggplot(aes(logp1rdi)) + geom_histogram(bins = 50)
```

Notice the impact on the data. Taking the log spreads out the data and makes them more symmetric. In this case, it is clearly not perfectly symmetric or Gaussian-like. In addition, there is a zero inflation issue from the instances where subjects have a zero value of RDI. Nonetheless, the logged data can be preferable to work with for many reasons. For example, the long tails of subjects

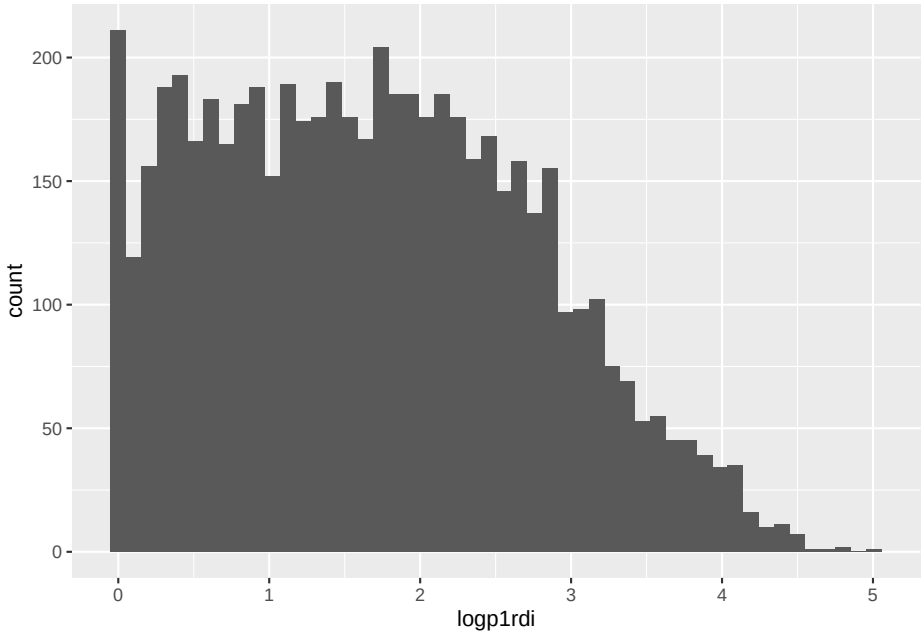


Figure 13.2: Histogram of log(1+`rdi4p`}.

with very high RDIs tend to squash the data together when plotting.

13.3 Interpreting logged data

A useful starting point for discussing methods of interpreting logged data is to discuss the geometric mean. The (sample) **geometric mean** of a dataset $X_1, \ldots, X_n$ is

$$\left(\prod_{i=1}^{n} X_i\right)^{1/n}.$$

This is in contrast to the typical (arithmetic) mean. Notice that the log of the geometric mean is

$$\frac{1}{n}\sum_{i=1}^{n} \log(X_i) = \frac{1}{n}\sum_{i=1}^{n} Y_i = \bar{Y}$$

where $Y_i = \log(X_i)$. Thus, the geometric mean is the exponent of the arithmetic mean of the logged data.

The geometric mean is the more natural quantity to use when considering multiplicative quantities, like percentages. As an example, suppose that in a population of interest, the prevalence of a disease rose 2% one year, then fell 1%

the next, then rose 2%, then rose 1%; since these factors act multiplicatively it makes sense to consider the geometric mean

$$(1.02 \times .99 \times 1.02 \times 1.01)^{1/4} = 1.01$$

for a 1% geometric mean increase in disease prevalence. This estimate is a more useful definition of average, since a 1% increase per year for four years is equal to the overall percentage increase in those four years: $(1.01)^4 = 1.02 \times .99 \times 1.02 \times 1.01$. The arithmetic mean, when added to itself four times, would match the sum of the numbers, a less useful summary in this case. The similarity between the arithmetic and geometric means is that if we multiply the arithmetic mean by n we get the sum of the values, whereas if we raise the geometric mean to the power n we get the product of the values.

13.3.1 Convergence

Via the law of large numbers, we know that (under iid sampling assumptions) the logged geometric mean converges to $E[\log(X)]$ and so the geometric mean converges to:

$$\theta = \exp\{E[\log(X)]\} .$$

We might call this quantity the *population geometric mean*. Since exp is not linear we know generally $\exp(E[\log(X)]) \neq E[\exp(\log(X))]$, so that we can see this is not equal to the population mean on the natural scale, $E[X]$:

$$\begin{aligned} E[X] &= E[\exp(\log(X))] \\ \exp(E[\log(X)]) &\neq E[\exp(\log(X))] \\ \exp(E[\log(X)]) &\neq E[X] \end{aligned}$$

Note, however, if *data are symmetric* the mean is equal to the median. Also, the log is a monotonic function, which implies that the median on the log scale is equal to the log of the median on the original scale.

If the data are symmetric on the log scale (called log-symmetric), then the mean is the median on the log scale (by symmetry), such that:

$$P\{\log(X) \leq E[\log(X)]\} = 0.5 .$$

We can use this identity to find that

$$P\{X \leq \theta\} = 0.5 .$$

Thus we arrive at the result that the population geometric mean is the median if the population is log-symmetric. One example of a log-symmetric distribution is the log-Normal distribution.

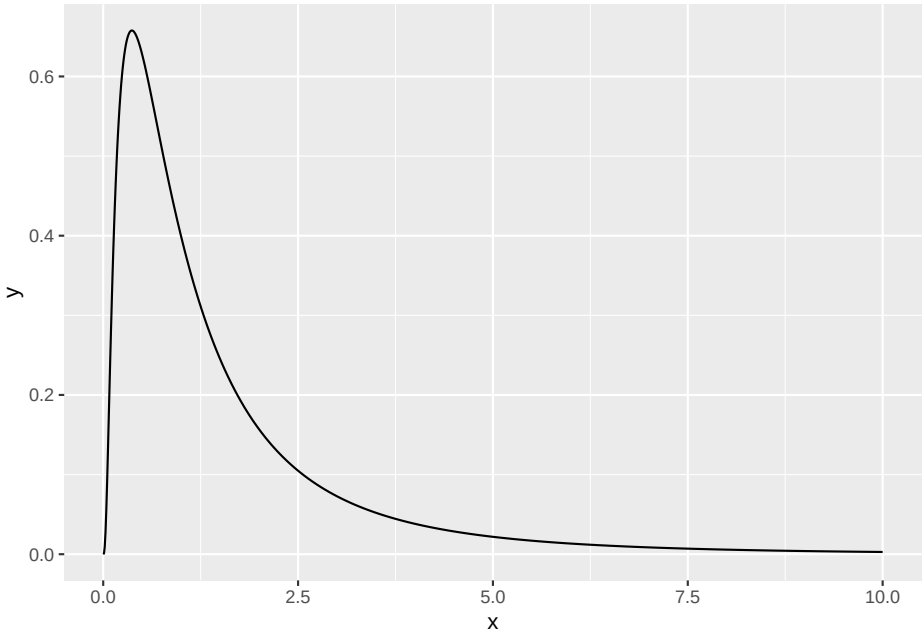


Figure 13.3: Pdf of the Lognormal$(0, 1)$.

13.4 Interpretation of inferences for logged data

All of the inferences that we have studied for arithmetic means apply to the geometric mean, since we can simply analyze the logged data. For confidence intervals, we need only to exponentiate the endpoints as the final step. We can use either the CLT or small sample t intervals on the log scale. If we assume that $\log(X)$ is Normally distributed to make t intervals, then X is said to be *log Normally distributed*. To be specific, we say $X \sim \text{Lognormal}(\mu, \sigma^2)$ if $\log(X) \sim N(\mu, \sigma^2)$. It should be emphasized that the log normal distribution is *not* the log of a Normal distribution. One cannot take the log of a Normally distributed random variable, since it is negative with positive probability.

Figure 13.3 provides a plot of the Lognormal$(0, 1)$ density:

```
t = data.frame(x = seq(0, 10, length = 1000)) %>%
  mutate(y = dlnorm(x))
ggplot(t, aes(x, y)) + geom_line()
```

Figure 13.4 provides the boxplots of `rdi4p` by COPD (chronic obstructive pulmonary disease) status.

```
ggplot(dat,aes(x = COPD15, y = rdi4p)) + geom_boxplot()
```

The 8 and NA categories are for unknown status. Also, notice the boxplots are

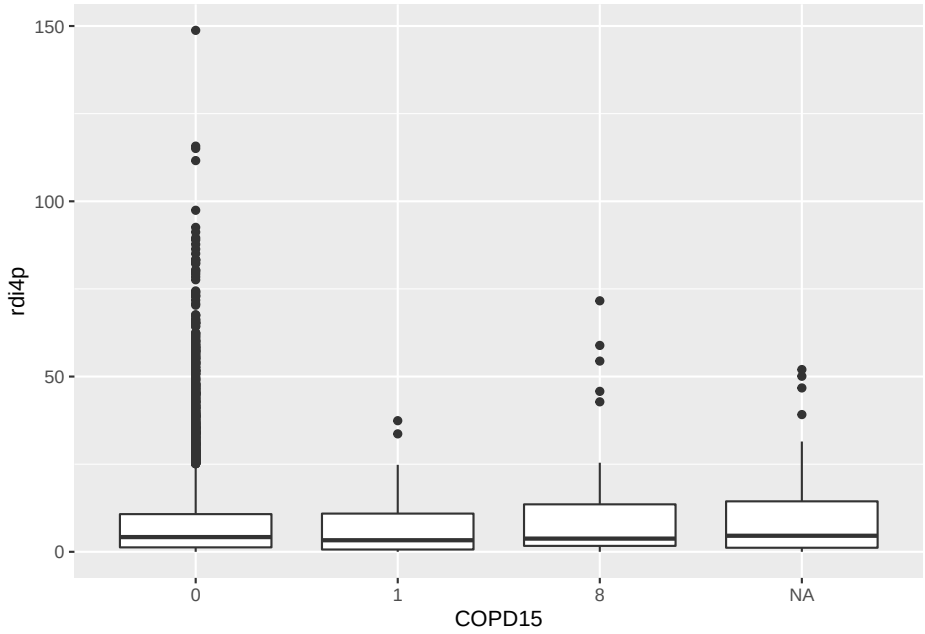


Figure 13.4: Boxplots of `rdi4p` by COPD status.

all squished up and skewed. So, let us both restrict our attention to the 0 and 1 categories and look at the logged data. Figure 13.5 provides the boxplots for the transformed data in these two categories.

```
subdat = filter(dat, COPD15 %in% c(0, 1))
ggplot(subdat,aes(x = COPD15, y = logp1rdi)) + geom_boxplot()
```

Surprisingly, the COPD positive group seems to have a slightly lower respiratory disturbance index. Now let's do a confidence interval on the log scale. Since there are so many observations, it is irrelevant whether we do a t or CLT based interval (so we will just use the CLT version). There are no missing observations from either `logp1rdi` or `COPD15` and we summarize the means and standard deviations within the groups.

```
summaryDat = group_by(subdat, COPD15) %>%
  summarise(mean = mean(logp1rdi), sd = sd(logp1rdi), n = n())
summaryDat
```

```
# A tibble: 2 x 4
  COPD15  mean    sd     n
  <fct>  <dbl> <dbl> <int>
1 0       1.69  1.06  5641
2 1       1.47  1.07    62
```

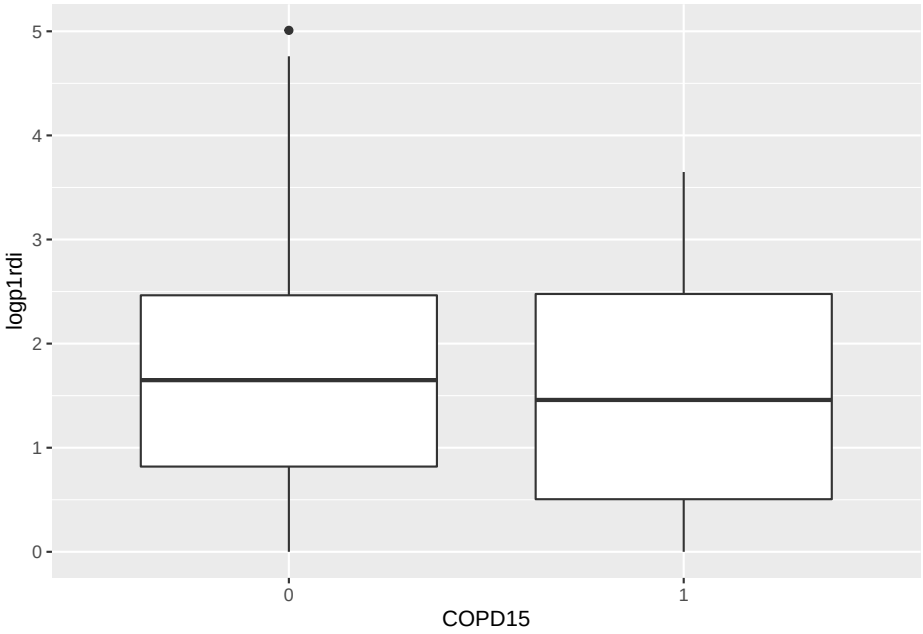


Figure 13.5: Boxplots of log of 1+ `rdi4p` by COPD status for categories 0 and 1.

Now let's construct our interval. We will do it manually.

```
ival = summaryDat %>%
  summarize(
    difference = diff(mean),
    sd = sqrt(sum(sd^2/n)))
ival = ival$difference + qnorm(c(0.5, .025, .975)) * ival$sd
out = rbind(ival, exp(ival))
rownames(out) = c("Log scale", "Exponentiated")
colnames(out) = c("Estimate", "Lower", "Upper")
round(out, 3)
```

```
              Estimate  Lower Upper
Log scale       -0.216 -0.485 0.052
Exponentiated    0.805  0.616 1.054
```

Here, the raw interval considers the difference in the log geometric mean of RDI+1 between the groups. To compare groups, consider whether 0 is in the interval. In this case, the interval (narrowly) contains 0. In contrast, the exponentiated interval shows the ratio of the geometric means of RDI+1 between the groups. Those who answered yes to having been diagnosed with COPD have an estimated geometric mean RDI+1 that is 80.5% that of those who answered no. The 95% interval for this ratio is from 61.6% to 105.4%. If we were willing

to assume that the population is log-symmetric, which is clearly *not* the case for these data, the interval could also be said to be an estimate for the ratio of the medians.

13.4.1 Summary

To summarize, if you log your raw data, then study the empirical mean, then your estimation is for logged population geometric means. When subtracting means across groups, one is studying the log of the ratio of geometric means. Furthermore, if you are willing to assume that the population level data are log-symmetric, then one can discuss medians and ratios of medians.

13.5 Problems

Problem 1. A random sample was taken of 20 patients admitted to a hospital. The lengths of stays in days for the 20 patients were:

```
los = c(4, 2, 4, 7, 1, 5, 3, 2, 2, 4,
        5, 2, 5, 3, 1, 4, 3, 1, 1, 3)
```

Calculate a 95% confidence interval for the mean length of stay.

Problem 2. Exponentiate your endpoints for the interval from Problem 1 and interpret the resulting interval. What is the interval estimating?

Problem 3. Calculate and interpret a 95% confidence interval for the geometric mean respiratory disturbance index plus one from the SHHS data. (Note that the plus one is necessary as RDI can be exactly 0 prohibiting logs. Adding a constant, typically one, is common in the sleep literature.)

Problem 4. Calculate and interpret a 95% confidence interval for the ratio of geometric means of the respiratory disturbance index plus one for subjects above and below the median split for body mass index.

Problem 5. Consider a random variable, X. Let $\mu = E[X]$. Differentiate between $\log(\mu)$ and $\exp(E[\log(X)])$. Given data, give a method for performing interval estimation of both.

Chapter 14

Interval estimation for binomial probabilities

This chapter covers the following topics

- Confidence intervals for a binomial proportion
- The Wald and Agresti-Coull intervals
- Bayesian intervals
- The exact, Clopper-Pearson interval
- Confidence intervals in `R`

14.1 Confidence intervals for a binomial proportion

In this chapter, we discuss the construction of confidence intervals for the probability of success, p, from data $X \sim \text{Binomial}(n, p)$. Here X is the number of successes obtained from n trials, each with probability of success p. This serves three purposes. First, it provides information about the uncertainty of the standard estimator, X/n, for the success probability, p. Second, it gives practical advice on constructing confidence intervals that have improved small to moderate sample properties. Third, it provides a practical solution to a general problem often encountered in practice. Indeed, consider the case when one is interested in constructing the confidence interval for the proportion of voters in Maryland for a presidential candidate or for the probability of death from stage IA Non-Small Cell Lung Cancer (NSCLC IA) in 10 years.

14.2 The Wald and Agresti-Coull intervals

Recall that if $X \sim \text{Binomial}(n, p)$, then the following hold:

1. $\widehat{p} = X/n$ is the maximum likelihood estimate (MLE) for p
2. $E[\widehat{p}] = p$
3. $\text{Var}(\widehat{p}) = p(1-p)/n$
4. For large n
$$\frac{\widehat{p} - p}{\sqrt{\widehat{p}(1-\widehat{p})/n}} \sim N(0,1) \; .$$

Point 4 allows us to construct the so-called Wald confidence interval for p

$$\widehat{p} \pm Z_{1-\alpha/2}\sqrt{\widehat{p}(1-\widehat{p})/n} \; .$$

This interval has several appealing properties. Most notably, it has a familiar form, is easy to remember, and has the property that

$$P\left(\widehat{p} - Z_{1-\alpha/2}\sqrt{\widehat{p}(1-\widehat{p})/n} \leq p \leq \widehat{p} + Z_{1-\alpha/2}\sqrt{\widehat{p}(1-\widehat{p})/n}\right) \approx 0.95 \; .$$

The interpretation of this formula is that before data are collected there is a 95% chance that the interval will cover the true, unknown, value of the parameter, p. An equivalent interpretation is that if a large number of experiments is run (each experiment consisting of n trials, each trial having success probability p), then 95% of the confidence intervals will cover the true value of the parameter, p. After a specific experiment is finished and the corresponding confidence interval is constructed, we do not know whether the resulting confidence interval covers the true parameter. Indeed, once the experiment is run, the calculated confidence interval based on the data from the experiment either does or does not contain the true value of the parameter.

The Bayesian interpretation of the corresponding 0.95 "credible interval" is more appealing. Indeed, the credible interval is a fixed (non-random) interval with the property that the parameter p is in that interval with probability 0.95 (or other designated probability). While this seemingly more palatable interpretation is sometimes viewed as a triumph of the Bayesian over frequentist philosophy, it comes at the price that parameters are random variables, an equally problematic pedagogical conundrum. Here we will continue to use statistical pragmatism, acknowledge that both philosophies lead to near identical practice, and point out the useful connections. When in doubt or when one obtains substantially different results using Bayesian analysis, check the prior and/or the Bayesian computations first.

Now, about that approximation sign, "≈". It is due to the central limit theorem (CLT) approximation, which holds only asymptotically. This means that the approximation is better when n is large. Of course, there is the question of "how large should n be for the approximation to hold?" There is no exact answer to this question, as the approximation depends both on the number of successes,

X, the true success probability, p, and the definition of "good approximation." In practice after conducting billions of simulations there is good agreement that the approximation is good if $n\widehat{p}(1-\widehat{p}) \geq 5$. However, this is a rule of thumb and should be treated as such. In practice, we recommend simulations to check how good specific approximations are.

The length of the Wald confidence interval is $2Z_{1-\alpha/2}\sqrt{p(1-p)/n}$, or twice the margin of error, $Z_{1-\alpha/2}\sqrt{p(1-p)/n}$. Since $p(1-p)$ is maximum when $p = 0.5$, this margin of error is at most $Z_{1-\alpha/2}/2\sqrt{n}$. For $\alpha = 0.05$, $Z_{1-\alpha/2} \approx 2$, so this is roughly $1/\sqrt{n}$, which yields the easy, quick, and often conservative interval estimate $\widehat{p} \pm 1/\sqrt{n}$.

Unfortunately, when $n\widehat{p}(1-\widehat{p}) < 5$ the Wald interval tends to be too short and have lower than nominal coverage probability. Remember, the 95% coverage only holds asymptotically and there are no guarantees in finite, small to moderate, samples. For finite samples the actual coverage of the interval may be much lower or higher than the desired 95%. Over-coverage can also be a problem in practice, as it is typically due to unnecessarily wide confidence intervals. However, in the case of this interval, under-coverage generally occurs, especially if the true parameter value, p, is close to 0 or 1. A simple simulation can show this. Below we simulate 10000 confidence intervals for a fine grid of possible values of p.

```
set.seed(237610)
## define a sequence of true values for the success probability
pVals = seq(.01, .99, by = .001)
## number of trials; number of simulations
n = 10; noSim = 10000
## loop over sequence of true values
coverageWald1 = sapply(pVals,
  function (p) {
    ## generate nosim proportions
    phat = rbinom(noSim, size = n, prob = p) / n
    ## the margin of error for each one
    moe = qnorm(.975) * sqrt(phat * (1 - phat) / n)
    ## the ll and ul of the confidence intervals
    ll = phat - moe; ul = phat + moe
    ## the estimated coverage
    mean( ll < p & p < ul )
  }
)

## Obtain the coverage probability at p=0.1, 0.5, 0.9
coverage_at_half<-round(coverageWald1[pVals==0.5],digits=2)
coverage_at_90percent<-round(coverageWald1[pVals==0.9],digits=2)
```

The black jagged line in Figure 14.1 displays the percentage of times the 95%

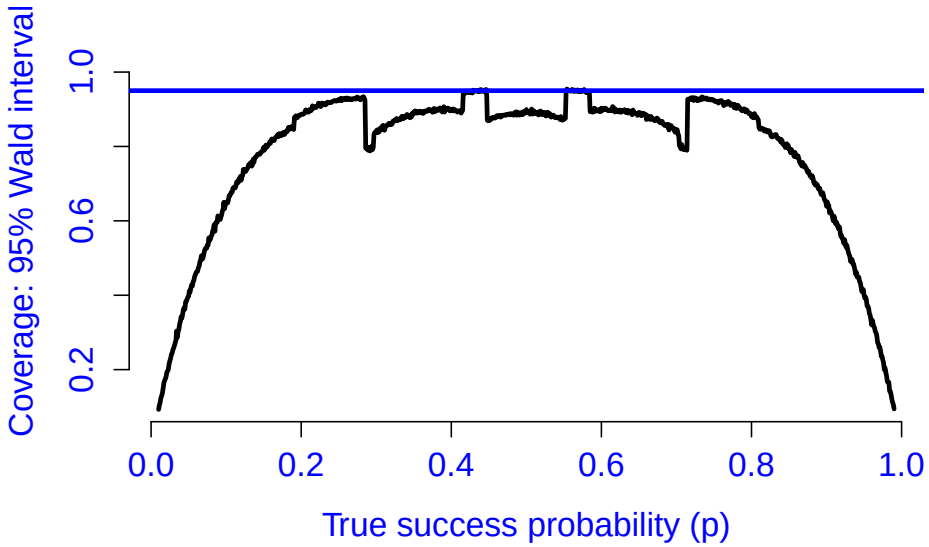


Figure 14.1: Coverage probability for the 95% Wald confidence interval when the number of trials is $n = 10$ as a function of the true success probability.

Wald confidence interval covers the true success probability, p, as a function of p (x-axis). The number of trials is $n = 10$ and the maximum value of $np(1-p)$ is obtained at $p = 0.5$, where it is equal to $2.5 < 5$.

```
plot(pVals, coverageWald1, type = "l",ylim=c(0.1,1.1),bty="n",
     xlab="True success probability (p)",ylab="Coverage: 95% Wald interval",
     lwd=3,cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main=NULL)
abline(h = .95,lwd=3,col="blue")
```

The coverage probability is smaller than the nominal 0.95 (blue horizontal line), though it is closer to nominal when p is close to 0.4 and 0.6. However, it dips to 0.9 for $p = 0.5$ and 0.64 for $p = 0.9$. The low coverage of the Wald interval is, at least in part, due to the small sample size and is particularly bad for p close to 0 and 1. We have seen that when $n\widehat{p}(1-\widehat{p}) \leq 5$, the performance of the Wald interval can be poor. Let us investigate what happens, for example, when $n = 100$ and $p \in [0.06, 0.94]$, which ensures that $np(1-p) \geq 5$.

The black jagged line in Figure 14.2 displays the percentage of times the 95% Wald confidence interval covers the true success probability, p, as a function of p (x-axis). The difference from Figure 14.2, where $n = 10$, is that now $n = 100$.

```
plot(pVals, coverageWald2, type = "l",ylim=c(0.1,1.1),bty="n",
     xlab="True Proportion (p)",ylab="Coverage: 95% Wald interval",
     lwd=3,cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main=NULL)
```

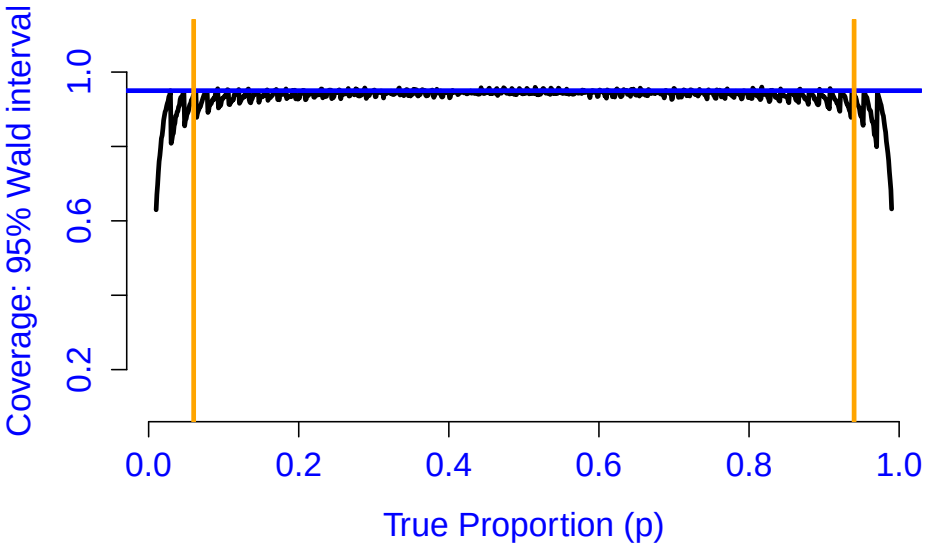


Figure 14.2: Coverage probability for the 95% Wald confidence interval when the number of trials is $n = 100$ as a function of the true success probability.

```
abline(h = .95,lwd=3,col="blue")
abline(v = .06,lwd=3,col="orange")
abline(v = .94, lwd=3,col="orange")
```

The plot shows much improved coverage probability of the Wald confidence interval, especially between the two orange lines that correspond to 0.06 and 0.94, respectively. Moreover, for most values of p the confidence interval neither under- nor over-shoots the target coverage probability. The small variations in coverage probabilities are due to sampling variability and could be reduced by increasing the number of simulations.

Of course, in practice one cannot always assume that $np(1-p) \geq 5$, and one sometimes has to deal with smaller sample sizes, such as $n = 10$ or 20. In this case, especially for p close to zero or one, obtaining a confidence interval with the nominal coverage is a surprisingly hard problem. A simple fix is to add two successes and two failures. That is, let $\widetilde{p} = (X+2)/(n+4)$ and $\widetilde{n} = n+4$. Then the so-called Agresti-Coull interval (Agresti and Coull 1998) is:

$$\widetilde{p} \pm Z_{1-\alpha/2}\sqrt{\widetilde{p}(1-\widetilde{p})/\widetilde{n}}\,.$$

The interval performs much better while retaining the simplicity and form of the Wald interval. Let us evaluate its coverage. We perform the exact same simulation as before, but replace the formula for calculating the confidence interval.

```
n = 10
coverageAC1 = sapply(pVals,
  function (p) {
    x = rbinom(noSim, size = n, prob = p)
    ## Use the shrunken estimator
    ptilde = (x + 2) / (n + 4)
    moe = qnorm(.975) * sqrt(ptilde * (1 - ptilde) / (n + 4))
    ll = ptilde - moe; ul = ptilde + moe
    mean( ll < p & p < ul )
  }
)
```

Figure 14.3 displays the percentage of times the 95% Agresti-Coull (dark-blue line) and Wald (black line) cover the true value of the parameter, p, as a function of p. As before, the horizontal blue line is the nominal coverage level of 0.95. Notice that the coverage is much better, though it may be a little conservative (high).

```
plot(pVals, coverageWald1, type = "l",ylim=c(0.1,1.1),bty="n",
     xlab="True Proportion (p)",ylab="Coverage: Wald, Agresti-Coull",
     lwd=3,cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main = "N = 10")
lines(pVals, coverageAC1, type = "l", lwd=3,col = "darkblue")
abline(h = .95,lwd=3,col="blue")
```

We also compare the coverage performance of the two intervals when $n = 100$. The R code is similar.

Let us see what happens when the sample size increases to $n = 100$, especially for $p < 0.06$ and $p > 0.94$ when the Wald test does not perform that well. Figure 14.3 displays the percentage of times the 95% Agresti-Coull (jagged, dark-blue line) and Wald (jagged, black line) intervals cover the true value of the parameter, p, as a function of p.

```
plot(pVals, coverageWald2, type = "l",ylim=c(0.1,1.1),bty="n",
     xlab="True Proportion (p)",ylab="Coverage: Wald, Agresti-Coull",
     lwd=3,cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main= "N = 100")
lines(pVals, coverageAC2, type = "l", lwd=3,col = "darkblue")
abline(h = .95,lwd=3,col="blue")
```

```
indices = c(1:51,951:981)
m_not_conforming_Wald<-round(mean(coverageWald2[indices]),
                             digits=3)
m_not_conforming_AC<-round(mean(coverageAC2[indices]),
                           digits = 3)
m_not_conforming_simple<-round(mean(coverage_simple[indices]),
```

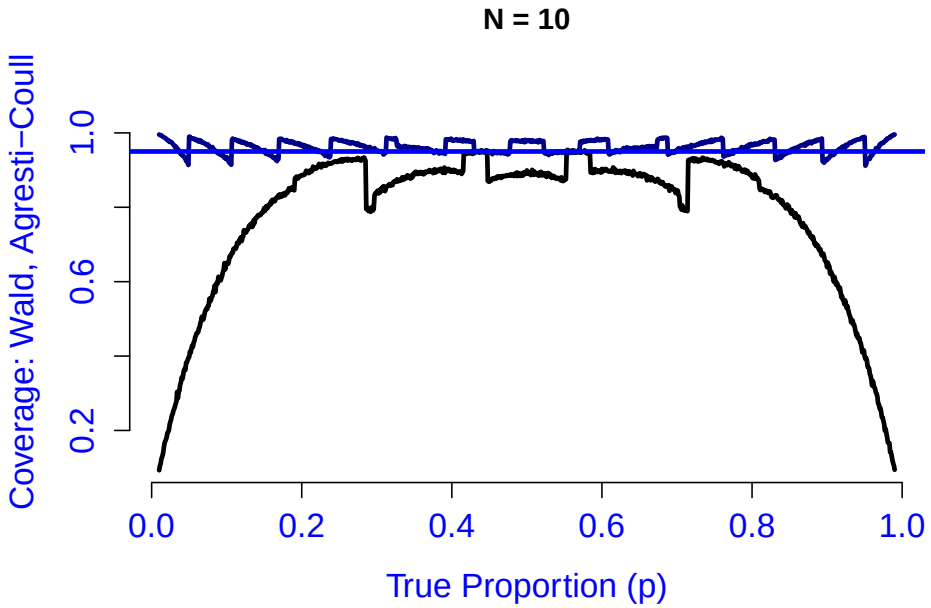


Figure 14.3: Coverage probability for the 95% Agresti-Coull (dark blue) and Wald (black) confidence intervals when the number of trials is $n = 10$ as a function of the true success probability.

```
                                    digits = 3)
```

Figure 14.4 indicates that the coverage probability of the two intervals closely agree for $p \in [0.1, 0.9]$ and that the Wald interval has a lower coverage probability than the Agresti-Coull interval. Moreover, for $p < 0.06$ and $p > 0.94$ the Wald interval has a smaller than nominal coverage probability (it is liberal), while the Agresti-Coull interval has a larger than nominal coverage probability (it is conservative). The average coverage probability for the Wald confidence interval for extreme probability values ($p \leq 0.06$ and $p \geq 0.96$) is 0.86 and for the Agresti-Coull confidence interval is 0.972. One could say "0.972 is much closer to 1 than 0.86 and most biostatisticians would agree that, as far as confidence intervals go, conservative is better than liberal." However, there are a couple of caveats. First, once one is conservative there is not much probability between 0.95 and 1 and it is easy to choose very long confidence intervals that have high coverage probability. For example, the interval $[0, 1]$ has coverage probability equal to 1 for any value of p and its average coverage is closer to 0.95 than 0.84, though it is not particularly useful in practice. Second, these discrepancies tend to be negligible when $np(1-p) \geq 5$ and absent when $np(1-p) \geq 10$. As a last example, let us consider the confidence interval $\widehat{p} \pm Z_{1-\alpha/2}/\sqrt{4n}$, which is known to be more conservative than the Wald confidence interval. Figure 14.5 displays the coverage probability for the Agresti-Coull (dark blue), Wald (black), and

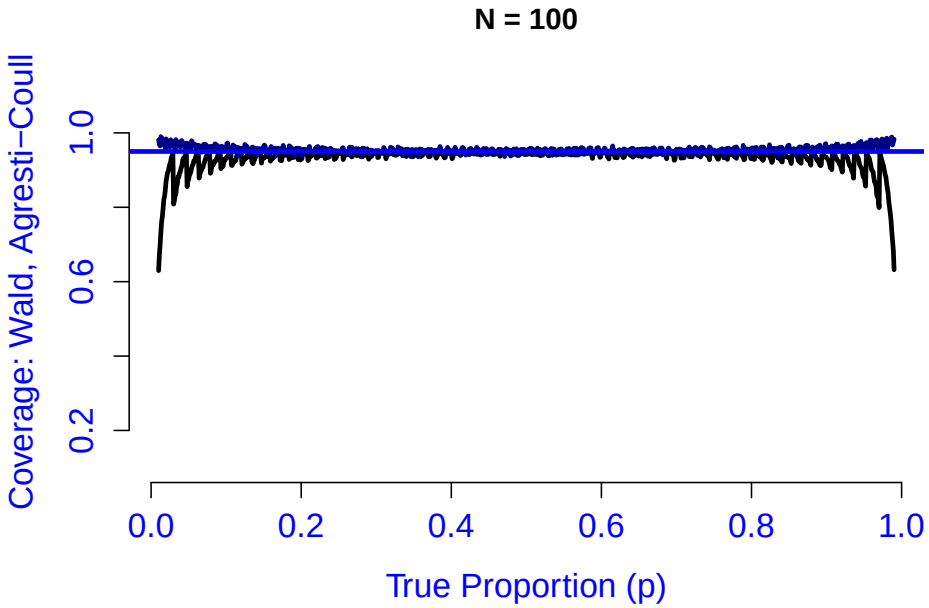


Figure 14.4: Coverage probability for the for the 95% Agresti-Coull (dark blue) and Wald (black) confidence intervals when the number of trials is $n = 100$ as a function of the true success probability.

$\widehat{p} \pm Z_{1-\alpha/2}/\sqrt{4n}$ (burlywood) confidence intervals when the number of trials is $n = 100$ as a function of the true success probability. We will refer to the $\widehat{p} \pm Z_{1-\alpha/2}/\sqrt{4n}$ as the simplified confidence interval.

```
plot(pVals, coverageWald2, type = "l",ylim=c(0.1,1.1),bty="n",
     xlab="True Proportion (p)",ylab="Coverage: Wald, AC, simplified",
     lwd=3,cex.axis=1.2,cex.lab=1.2,col.axis="blue",
     col.lab="blue",main=NULL)
lines(pVals, coverageAC2, type = "l", lwd=3,col = "darkblue")
lines(pVals,coverage_simple,lwd=3,col="burlywood4")
abline(h = .95,lwd=3,col="blue")
```

Look how the difference between the brown and blue coverage probabilities appears smaller than the one between the black and the blue, especially in the extreme probability area. However, the average coverage probability for the $\widehat{p} \pm Z_{1-\alpha/2}/\sqrt{4n}$ interval is estimated to be 1 for $p \in [0, 0.06] \cup [0.96, 1]$, the area of extreme probabilities. This coverage is perfect, but it comes at the expense of building an interval that is just too long relative to the other two confidence intervals in this area.

It is appealing that the Agresti-Coull interval tends to perform better than the Wald interval, especially in smaller sample sizes or when the true probabilities

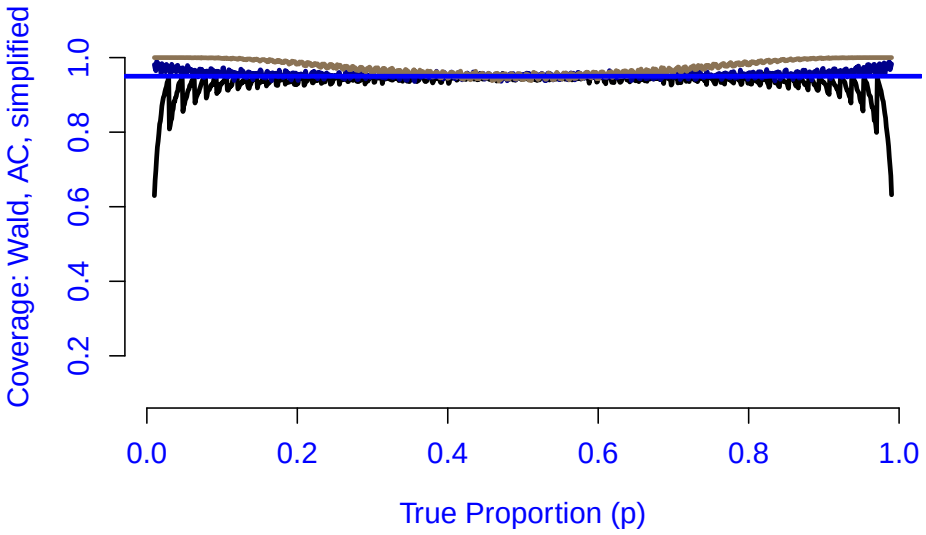


Figure 14.5: Coverage probability for the 95% Agresti-Coull (dark blue), Wald (black), and simplified (burlywood) confidence intervals when the number of trials is $n = 100$ as a function of the true success probability.

are quite extreme. However, the take home messages from this section are:

1. The Wald confidence interval works well when there is strong reason to believe that $np(1-p) \geq 5$
2. When $np(1-p) < 5$ it is reasonable to consider alternative intervals and the Agresti-Coull interval is one such alternative
3. The Agresti-Coull interval tends to be too conservative at extreme probabilities
4. It is a good idea to produce multiple confidence intervals and compare them. *Most of the time these intervals will be in close agreement*
5. Exceptions are important to understand, but should not paralyze us into data analysis inaction

It is a little surprising that making up and adding fake data would actually improve the performance of a statistical analysis. This happens because of small samples and extreme probabilities, when borrowing strength (or using statistical shrinkage) is a good idea. In the next section we provide more insight into how this is accomplished by deriving the motivation for the Agresti-Coull confidence interval using Bayesian inference. This will provide a more general strategy inspired by Bayesian inference for constructing confidence intervals. This is rather ironic given that Bayesians do not construct confidence intervals and frequentists do not use priors. But, it is useful and that is good enough for us.

14.3 Bayesian intervals

The discussion of improving on the Wald interval leads naturally into Bayesian intervals. In fact, estimating a binomial proportion is an ideal setting to introduce Bayesian analysis. Modern Bayesian analysis is incredibly diverse, and thus we can only present a subset. Here we discuss "subjective Bayesianism", the classical view of Bayes analysis. Henceforth in this chapter, we will not draw a distinction and simply refer to our analysis as Bayesian, omitting the "subjective" term.

Bayesian analysis requires three entities: the data, a model linking the data to a population of interest (likelihood), and a prior. The prior represents our beliefs regarding parameters of the likelihood that we are interested in (and ones that we are not, which we do not need to cover here.) We perform Bayesian analysis by combining these three entities into the so-called posterior distribution. The prior represents our beliefs in the absence of data, the posterior represents our beliefs after the evidence presented by the data.

Bayesian analysis treats parameters as random variables. This is because the probability in Bayesian analysis represents beliefs, not long run frequency behavior. In this way, one can think of Bayes analysis as a sort of formal calculus of beliefs. To illustrate, let our data, X, be assumed to be Binomial(n, p). Let $\pi(p)$ be a distribution with support on $[0, 1]$ representing our prior. Assuming our likelihood is $\mathcal{L}(p; Y)$, then the generalization of Bayes rule is:

$$\pi(p \mid x) = \frac{\mathcal{L}(p; x)\pi(p)}{\int_0^1 \mathcal{L}(\widetilde{p}; x)\pi(\widetilde{p})d\widetilde{p}} ,$$

where $\pi(p \mid x)$ is our posterior, the distribution of the parameter given the data. Notice that the denominator does not involve p, so we can write:

$$\pi(p \mid x) \propto \mathcal{L}(p; x)\pi(p) ,$$

where $\propto$ is "proportional to." We summarize this relationship as:

$$\text{Posterior} \propto \text{Likelihood} \times \text{Prior} .$$

Let us go through the binomial example, specifying a prior. To do this, we need a density defined on the support of the parameter, $[0, 1]$. The Beta distribution satisfies this requirement. Specifically, a random variable, p, is said to be Beta distributed and it is denoted Beta(α, β) if it has the pdf:

$$\frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1 - p)^{\beta-1} \quad \text{for } \ 0 \leq p \leq 1 ,$$

where Γ is the Gamma function: $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1}e^{-x}dx$. When α is a positive integer, then $\Gamma(\alpha) = (\alpha-1)!$. The mean of the Beta density is $\alpha/(\alpha+\beta)$ and the variance is:

$$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)} .$$

Notice that the uniform density is a special case of the Beta(α, β) family when $\alpha = \beta = 1$. It is worth thinking a little bit about the *interpretation* of the amount and type of information encapsulated by the Beta(α, β) distribution. Note its extraordinary likeness with the Binomial distribution with x successes and $n - x$ failures, whose pmf is

$$\binom{n}{x} p^x (1-p)^{n-x} .$$

So, the parameter $\alpha - 1 = x$ can be interpreted as the number of successes and $\beta - 1 = n - x$ as the number of failures encapsulated in the prior. Another way of thinking is that α is the number of successes plus one, $x + 1$, and β is the number of failures plus one, $n - x + 1$. The intuition breaks down for $\alpha < 1$ and $\beta < 1$, but it remains useful. Note that the mean of a Beta distribution is $\alpha/(\alpha+\beta) = (x+1)/(n+2)$, which is the maximum likelihood estimator (MLE) if we add one success and one failure to the data.

We will use the notation of likelihoods as above, where p is the parameter of interest. While we have used the Binomial likelihood and the success probability, p, as parameter of interest, the Bayesian formalism applies generally to any type of likelihood and parameter. Let us return to our setting where $Y \sim$ Binomial(n, p) and let us specify that $\pi(p)$ is a Beta density with parameters α (number of prior assumed successes plus one) and β (number of prior assumed failures plus one). Thus, we choose values of α and β so that the Beta prior is indicative of our degree of belief regarding p in the absence of data. Using the rule that the posterior is the likelihood times the prior, and throwing out anything that does not depend on p, we have that

$$\pi(p \mid x) \propto p^x(1-p)^{n-x} \times p^{\alpha-1}(1-p)^{\beta-1} = p^{x+\alpha-1}(1-p)^{n-x+\beta-1} .$$

We recognize that this density is just another Beta density with parameters $\widetilde{\alpha} = x + \alpha$ and $\widetilde{\beta} = n - x + \beta$. This was no accident. The Beta density is the so-called conjugate prior for the binomial, which means that if we have a Beta prior for the success probability of a Binomial, we get a Beta posterior. Note that the *interpretation* still holds. We had $\alpha-1$ prior successes and $\beta-1$ failures before the data were collected. To that we added x additional successes and $n - x$ failures from the experiment (likelihood) which led to a Beta distribution with $x + \alpha - 1$ successes and $n - x + \beta - 1$ failures. The particular case when $\alpha = \beta = 1$ (zero successes and failures added to the data) corresponds to a uniform prior, which makes the posterior equal to the likelihood.

For the general case when we use a Beta(α, β) the posterior mean is:

$$\begin{aligned} E[p \mid x] =& \frac{x+\alpha}{n+\alpha+\beta} \\ =& \frac{x}{n} \times \frac{n}{n+\alpha+\beta} + \frac{\alpha}{\alpha+\beta} \times \frac{\alpha+\beta}{n+\alpha+\beta} \\ =& \text{MLE} \times \pi + \text{Prior Mean} \times (1-\pi), \text{ where } \pi = \frac{n}{n+\alpha+\beta} . \end{aligned}$$

That is, the posterior mean is a weighted average of the MLE and the prior mean. The weight on the MLE is $n/(n+\alpha+\beta)$ so that as n gets larger, it gets more weight. That is, as we collect more data, we place more weight on it and less on our prior beliefs. In contrast, if our prior number of observations, i.e., $\alpha+\beta$, is large then it takes more data to shake us off of our beliefs.

Another interesting aspect to Bayesian analysis is that it is consistent with information updating as data are collected. For example, consider the case when $X_1 \sim \text{Binomial}(n_1, p)$ is collected first and $X_2 \sim \text{Binomial}(n_2, p)$ is collected second. You could imagine using the posterior, $\pi(p \mid x_1)$, as the prior for analyzing x_2. This is allowed because

$$\pi(p \mid x_1, x_2) \propto \mathcal{L}(p; x_2)\pi(p \mid x_1) \propto \mathcal{L}(p; x_2)\mathcal{L}(p; x_1)\pi(p) \propto \mathcal{L}(p; x_1, x_2)\pi(p) .$$

In other words, the posterior we get from using our current posterior as the prior yields the same subsequent posterior as if we had collected all data at once. You can think of this process as the new data updating the posterior. Note that the intuition we started with continues to hold in this context. Indeed, the posterior will encapsulate information from $\alpha + x_1 + x_2 - 1$ successes and $\beta + n_1 - x_1 + n_2 - x_2 - 1$ failures. This, and many other aspects, of internal logic and consistency of Bayesian analysis makes it very appealing to many. Nonetheless, Bayesian analysis is not a panacea. One must specify a prior (and likelihood) to perform the analysis. Moreover, frequency error rates are inherently useful quantities. This is why many modern Bayesian statisticians use the Bayes rule to create statistics, but evaluate those statistics using frequency error rates. The Agresti-Coull interval could be thought of as such an example. Thus, it is important for modern statisticians to be well versed in both Bayesian and frequency statistical approaches.

14.3.1 Connections with the Agresti-Coull interval

We can use Bayesian analysis to provide the motivation for the Agresti-Coull interval. Specifically, note that the posterior mean is

$$\widetilde{p} = \frac{x + \alpha}{n + \alpha + \beta} ,$$

which is $\widetilde{p}$ from the Agresti-Coull interval when $\alpha = \beta = 2$. Now, let us consider the variance.

$$\text{Var}(p \mid x) = \frac{(x + \alpha)(n - x + \beta)}{(n + \alpha + \beta)^2(n + \alpha + \beta + 1)} ,$$

which, when $\alpha = \beta = 2$ and $\widetilde{n} = n + 4 = n + \alpha + \beta$ becomes

$$\text{Var}(p \mid x) = \frac{\widetilde{p}(1 - \widetilde{p})}{\widetilde{n} + 1} .$$

This is almost identical to the Agresti-Coull interval variance, with the exception of the +1 in the denominator. However, for large n the difference is irrelevant. To obtain the interval, note that the normal density can approximate a Beta density as the α and β parameters are large, with the approximation being better when they are closer to being equal. Our posterior Beta parameters are $y + \alpha$ and $n - y + \beta$. Therefore, as long as we have a large n and do not have too many successes or failures, the approximation will be accurate. Using this approximation leads to the Agresti-Coull interval (with the extra +1 in the variance.)

14.3.2 Conducting Bayesian inference

The Agresti-Coull interval just uses Bayesian calculations to construct a confidence interval. If we want to perform a Bayesian analysis, we use the posterior directly for inference. In Bayesian inference the first step is plotting the posterior. Since this chapter is discussing interval estimates, if one is desired, a Bayesian analysis simply finds numbers a and b satisfying:

$$P(p \in [a, b] \mid x) = 1 - \alpha .$$

The important distinction here is that p, the true success probability, is a random variable, not a number, while the bounds of the confidence intervals, a and b, are fixed numbers. For frequentist confidence intervals p is a fixed, unknow, parameter and the bounds of the confidence intervals are random variables.

We call these intervals *credible intervals*, as opposed to confidence intervals, since they are interpreted differently. The interpretation of the above equation

is that the probability that the true parameter being in this interval is $1 - \alpha$. Interestingly, Bayesian intervals often have very good frequency performance, so that they could be interpreted as confidence intervals. However, one still needs to choose a and b to satisfy the above equation. Two methods for picking a and b are the equal-tail and the highest posterior density (HPD) interval. The equal-tail interval simply sets a and b to be the relevant lower and upper quantiles of the posterior distribution. In contrast, the HPD interval selects a and b to ensure that $b - a$ (the length of the credible interval) is minimized for the given coverage probability target. We will demonstrate this via an example below.

14.3.3 Example

Suppose that in a random sample of an at-risk population 13 of 20 subjects had hypertension. Based on these data we would like to estimate the prevalence of hypertension in this population. First, let us calculate the Wald and Agresti-Coull intervals,

```
n = 20; x = 13; ntilde = n + 4
phat = x / n
ptilde = (x + 2) / ntilde
```

Here $\widehat{p} = 0.65$ and $\widetilde{p} = 0.625$. Let us now calculate both intervals:

```
z = qnorm(c(0.025, 0.975))
# Wald
se_Wald = sqrt(phat * (1 - phat) / n)
ci_Wald = round(phat + z * se_Wald, 3)
ci_Wald
```

```
[1] 0.441 0.859
```

```
length_Wald <- round(diff(ci_Wald),digits=3)
# Agresti-Coull
se_AC = sqrt(ptilde * (1 - ptilde) / n)
ci_AC = round(ptilde + z * se_AC, 3)
ci_AC
```

```
[1] 0.413 0.837
```

```
length_AC <- round(diff(ci_AC),digits=3)
```

Now let us do our Bayesian analysis. First, plot the posterior. Consider setting $\alpha = \beta = 0.5$. This prior has some theoretical benefits, though the interpretation is a little difficult. Note that smaller α and β are less informative in the sense that the posterior is closer to the likelihood and $\alpha = 0$ or $\beta = 0$ would correspond to an improper prior (it is not a pdf and does not integrate to 1.) Note, however, that the improper Beta$(0, 0)$ is an acceptable prior for the success probability, p, and leads to a proper posterior.

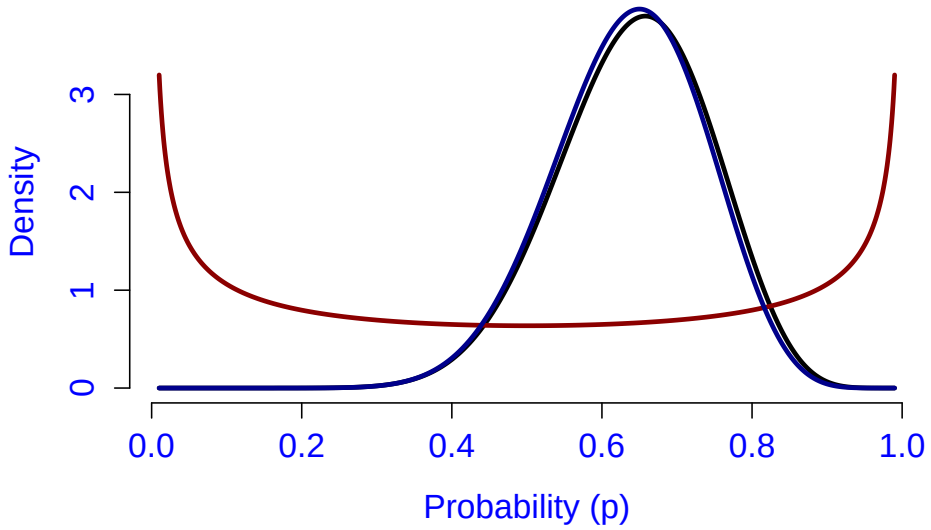


Figure 14.6: The prior distribution Beta(0.5,0.5) (dark red), likelihood for 13 out of 20 study participants having hypertension (dark blue), and posterior distribution (black).

```
alpha = 0.5; beta = 0.5
## Plot the posterior
plot(pVals, dbeta(pVals, x + alpha, n-x + beta), type = "l",
     bty = "n", xlab = "Probability (p)", ylab = "Density",
     lwd=3,cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main=NULL)
## Add the prior
lines(pVals, dbeta(pVals, alpha, beta), lwd = 3, col = "darkred")
## Show the likelihood
lines(pVals, dbeta(pVals, x + 1, n - x + 1), lwd=3, col = "darkblue")
```

Figure 14.6 displays the Beta$(0.5, 0.5)$ prior (dark red), the likelihood of the observed data (dark blue), and the posterior distribution (black). Here we are using the fact that the likelihood is proportional to a Beta density to avoid having to write out the function for the likelihood. Note that the likelihood and the posterior are very close together, even though the prior distribution is not particularly "flat." This is due to the fact that non-informative Beta priors correspond to small $\alpha < 1$ and $\beta < 1$, which do not correspond to flat priors. Let us obtain an equal-tail confidence interval.

```
round(qbeta(c(0.025, 0.975), x + alpha, n-x + beta), 3)
```

```
[1] 0.432 0.828
```

The odd U shape of the prior is related to the properties described above, which

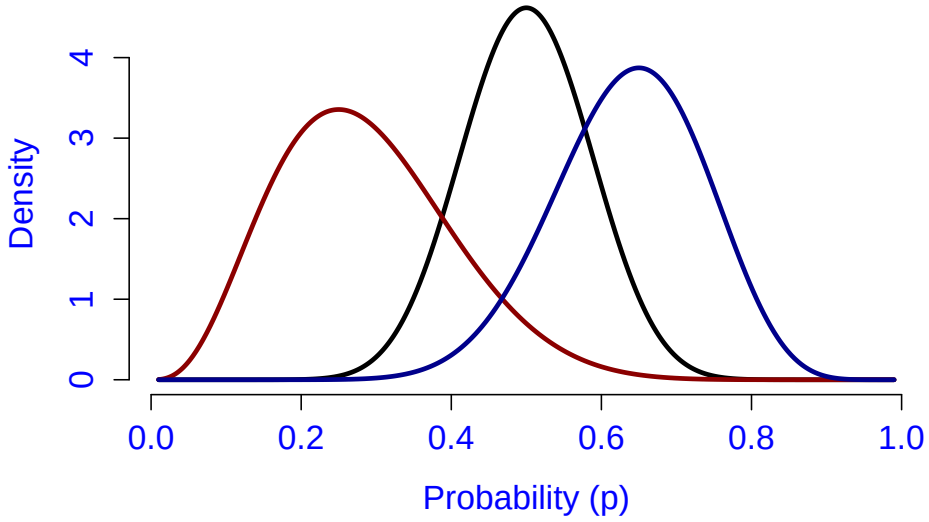


Figure 14.7: The prior distribution Beta(4,10) (dark red), likelihood for 13 out of 20 study participants having hypertension (dark blue), and posterior distribution (black).

makes the prior minimally informative. Let us look at a case with a more informative prior (larger α and β, $\alpha = 4, \beta = 10$) and see how the posterior combines our prior beliefs and our data.

```
alpha = 4; beta = 10
## Plot the posterior
plot(pVals, dbeta(pVals, x + alpha, n-x + beta), type = "l", lwd = 3,
      bty= "n", xlab = "Probability (p)", ylab = "Density",
      cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main=NULL)
## Add the prior
lines(pVals, dbeta(pVals, alpha, beta), lwd = 3, col = "darkred")
## Show the likelihood
lines(pVals, dbeta(pVals, x + 1, n - x + 1), lwd=3, col = "darkblue")
```

Figure 14.7 is similar to Figure 14.6, except that the prior used is Beta$(4, 10)$ instead of Beta$(0.5, 0.5)$. Thus, the prior belief or information (dark red) is that the prevalence of hypertension is quite low with a mean of $\alpha/(\alpha + \beta) = 4/(10 + 4) \approx 0.29$. The likelihood of the data, represented by the dark blue line, is in disagreement with a mean observed prevalence of $13/20 = 0.65$, which is about twice as large. Moreover, the number of prior successes ($\alpha - 1 = 3$) and failures ($\beta - 1 = 9$) is of the same order of magnitude with the number of successes ($x = 13$) and failures ($n - x = 7$) observed in the experiment. Because of that, the prior (dark red) and the likelihood (dark blue) have comparable

variability, which is why the posterior (black) is heavily influenced by both the prior and the likelihood. With such a strong prior the resulting interval is quite different because of the discrepancy between our prior and likelihood.

```
round(qbeta(c(0.025, 0.975), x + alpha, n-x + beta), 3)
```

```
[1] 0.335 0.665
```

Even in this simple case, HPD intervals require numerical evaluation. Fortunately, the package `binom` can perform the calculations for us.

```
library(binom)
out = binom.bayes(13, 20, type = "highest")
l = out$lower; u = out$upper
round(c(l, u), 3)
```

```
[1] 0.442 0.836
```

The default in the `binom.bayes` for obtaining the HPD interval (option `type = "highest"`) is the Beta$(0.5, 0.5)$ prior. We can best see what the HPD interval is doing using graphics.

```
alpha = 0.5; beta = 0.5
## plot the posterior
plot(pVals, dbeta(pVals, x + alpha, n-x + beta),
     type = "l",
     lwd = 3,
     bty="n", xlab = "Probability (p)", ylab = "Density",
     cex.axis=1.3,cex.lab=1.3,col.axis="blue",
     col.lab="blue",main=NULL)
pVals2 = seq(l, u, length = 1000)
## plot the area represented by the HPD interval
polygon(c(l, pVals2, u),
        c(0, dbeta(pVals2, x + alpha, n - x + beta), 0),
        col = "salmon")
## plot the interval as a horizontal line
lines(c(l, u), c(dbeta(l, x + alpha, n - x + beta),
                 dbeta(u, x + alpha, n - x + beta)),
      lwd = 3,
      col = "darkblue")
```

Figure 14.8 displays the posterior distribution for the prior Beta$(0.5, 0.5)$ and the likelihood of an experiment with 13 (HTN) positive and 7 (HTN) negative results. The shaded area corresponds to the 95% HPD credible interval, which is the shortest interval with this property. The light red color shaded area represents 95% probability under the posterior distribution. Numerically, the interval is obtained by moving the dark blue horizontal line up and down to match the 0.95 target probability. The actual interval is given as the horizontal-axis endpoints of the blue line. HPD credible intervals can be obtained for any

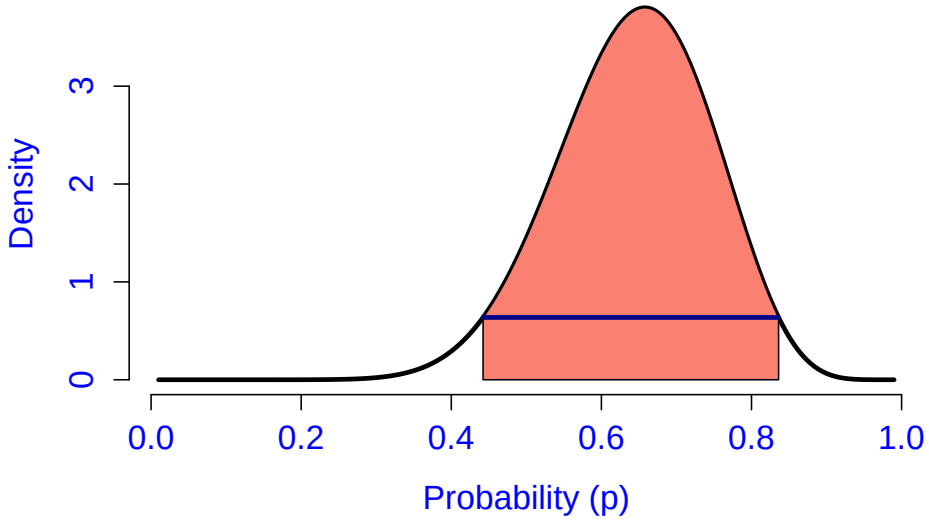


Figure 14.8: The posterior distribution for the prior Beta(0.5,0.5) and the likelihood of an experiment with 13 (HTN) positive and 7 (HTN) negative results. The shaded area corresponds to the 95% credible interval and the corresponding interval is the shortest interval with this property.

probability. For example, consider the 60% interval shown in Figure 14.9.

```
out = binom.bayes(13, 20, type = "highest", conf.level = .6)
l = out$lower; u = out$upper
plot(pVals, dbeta(pVals, x + alpha, n-x + beta), type = "l",
     lwd = 3,
     bty="n", xlab= "Probability (p)", ylab = "Density",
     cex.axis=1.2,cex.lab=1.2,col.axis="blue",
     col.lab="blue",main=NULL)
pVals2 = seq(l, u, length = 1000)
polygon(c(l, pVals2, u),
        c(0, dbeta(pVals2, x + alpha, n - x + beta), 0),
        col = "salmon")
lines(c(l, u), c(dbeta(l, x + alpha, n - x + beta),
                 dbeta(u, x + alpha, n - x + beta)),
      lwd = 3,
      col = "darkblue")
```

Note that the HPD intervals are not symmetric around the estimated p, but they are shorter and have the same coverage probability, at least in theory.

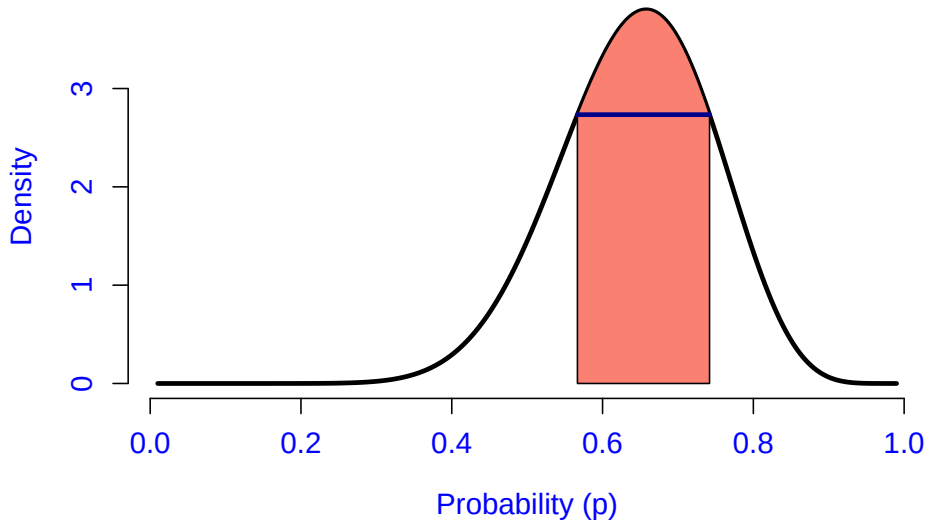


Figure 14.9: The posterior distribution for the prior Beta(0.5,0.5) and the likelihood of an experiment with 13 (HTN) positive and 7 (HTN) negative results. The shaded area corresponds to the 60% credible interval and the corresponding interval is the shortest interval with this property.

14.4 The exact, Clopper-Pearson interval

The Clopper-Pearson confidence interval was introduced very early on (Clopper and Pearson 1934). The approach is to estimate the probabilities $\widehat{p}_L$ and $\widehat{p}_U$ based on the data such that

$$P(X \leq x|p = p_U) = \sum_{k=0}^{x} \binom{n}{k} p_U^k (1 - p_U)^{n-k} = \alpha/2$$

and

$$P(X \geq x|p = p_L) = \sum_{k=x}^{n} \binom{n}{k} p_L^k (1 - p_L)^{n-k} = \alpha/2 \, .$$

These formulae look relatively complicated, but are easy to implement, especially using a computer and `R`. The rationale for calculating the lower limit, p_L, on the confidence interval is that we have already observed a fraction of $\widehat{p} = 13/20 = 0.65$ individuals who have hypertension. Given this observed fraction, we are interested in what small probabilities are not likely to be close to the true, but unknown, probability p. For example, we know that a probability $p = 0$ of having HTN in the population is incompatible with observing 13 cases out of 20 study participants. Of course, $p = 0.01$ is not incompatible with 13 out of 20 subjects having HTN. But it would be extremely rare to observe 13 or more events if the true event probability were 0.01. Thus, it makes sense to continue to increase p until we find one, say p_L, such that $P(X \geq x|p = p_L) = \alpha/2$.

This is a very small probability, but not zero. A similar intuition holds for calculating p_U. Below we show how to calculate from scratch the Clopper-Pearson confidence interval for a random sample from an at-risk population where 13 out of 20 subjects had hypertension.

```
alpha=0.05
exact_prob_U<-pbinom(x,n,pVals)
exact_prob_V<-1-pbinom(x-1,n,pVals)
pU<-pVals[which.min(abs(exact_prob_U-alpha/2))]
pL<-pVals[which.min(abs(exact_prob_V-alpha/2))]
length_PK=pU-pL
```

Thus, the Clopper-Pearson confidence interval for this example is [0.408,0.846]. This interval has a length of 0.438, making it slightly longer than the Agresti-Coull (0.424) and Wald intervals (0.418), respectively. Interestingly, this is a case that is pretty close to the rule of thumb for when the Wald interval is expected to perform well. Indeed, $20 * (13/20) * (7/20) = 4.55$, below 5, but pretty close.

14.5 Confidence intervals in R

For those frightful days when we want to forget everything we learned about biostatistics and just need to get some confidence intervals, fear not. We have a push button approach that will make everything better, the function `binom.confint` in the `R` package `binom` (Sundar 2014). Here it is a quick way of getting all 95% confidence intervals discussed in this chapter and even a few more,

```
binom.confint(x, n, conf.level = 0.95, methods = "all")
```

```
          method  x  n      mean     lower     upper
1  agresti-coull 13 20 0.6500000 0.4315888 0.8200736
2     asymptotic 13 20 0.6500000 0.4409627 0.8590373
3          bayes 13 20 0.6428571 0.4423068 0.8360884
4        cloglog 13 20 0.6500000 0.4030012 0.8153049
5          exact 13 20 0.6500000 0.4078115 0.8460908
6          logit 13 20 0.6500000 0.4256049 0.8231570
7         probit 13 20 0.6500000 0.4289545 0.8288619
8        profile 13 20 0.6500000 0.4320692 0.8316361
9            lrt 13 20 0.6500000 0.4320599 0.8316972
10     prop.test 13 20 0.6500000 0.4094896 0.8369133
11        wilson 13 20 0.6500000 0.4328543 0.8188082
```

One should recognize the Wald (labeled "asymptotic"), Agresti-Coull (labeled "agresti-coull"), the Clopper-Pearson (labeled "exact"), and the Bayes (labeled "bayes") confidence intervals. Here the Bayes confidence interval is obtained

using the default Beta$(0.5, 0.5)$ prior and the Agresti-Coull interval would be closest to the Bayes confidence interval with the Beta$(2, 2)$ prior. A few other confidence intervals are produced, but we cannot spend an entire book on each and every single one of them. If one wants to obtain only one specific confidence interval, simply call

```
binom.confint(x, n, conf.level = 0.95, methods = "agresti-coull")
```

while, if a more specific Bayes confidence interval is needed, the function `binom.bayes` in the same package can be used. For example,

```
binom.bayes(x,n,type="highest",conf.level=0.95,
            prior.shape1=2,prior.shape2=1)
```

produces the 95% confidence interval with the highest posterior density (HPD, `type="highest"`) and a Beta$(2, 1)$ prior. If a symmetric interval is needed instead of the HPD interval, one should use the option `type="central"`.

Recall that if $np(1-p) \geq 5$, the Wald (a.k.a. asymptotic, textbook) confidence interval tends to perform very well in practice. The Agresti-Coull and exact confidence intervals perform better in small samples, though they tend to be a little conservative. Generally, there should be pretty close agreement between intervals when there is enough information in the data. However, as a general rule, one should not run all confidence intervals and choose the one that is most convenient to support or reject a particular scientific hypothesis. Indeed, in practice *one should decide what interval will be used before the experiment is run*, based on the known properties of confidence intervals described here.

14.6 Problems

Problem 1. Consider the case when we compare the Wald, Agresti-Coull, and simplified confidence intervals based on the $\widehat{p} \pm Z_{1-\alpha/2}/\sqrt{4n}$ formula. However, we are interested in comparing the relative length of these confidence intervals. Using simulations compute and plot the average and relative lengths for each confidence interval for $n = 10, 20, \ldots, 1000$ and $p = 0.01, 0.02, \ldots, 0.99$.

Problem 2. Consider the case when the probability of having hypertension in a population is 13/20. Compare the coverage probabilities and length of the confidence intervals using the Wald, Agresti-Coull, Bayesian, and Clopper-Pearson methods. Provide a summary of your findings.

Problem 3. Using a computer, generate 1000 Binomial random variables for $n = 10$ and $p = 0.3$

a. Calculate the percentage of times that

$$\widehat{p} \pm 1.96\sqrt{\widehat{p}(1-\widehat{p})/n}$$

contains the true value of p. Here $\widehat{p} = X/n$, where X is each binomial variable. Do the intervals appear to have the coverage that they are supposed to?

b. Repeat the calculation, only now use the interval

$$\widetilde{p} \pm 1.96\sqrt{\widetilde{p}(1-\widetilde{p})/n}$$

where $\widetilde{p} = (X+2)/(n+4)$. Does the coverage appear to be closer to 0.95?

c. Repeat this comparison for $p = 0.1$ and $p = 0.5$. Which of the two intervals appears to perform better?

Problem 4. In a random sample of 100 subjects with low back pain, 27 reported an improvement in symptoms after exercise therapy.

a. Give and interpret an interval estimate for the true proportion of subjects who respond to exercise therapy.
b. Provide all confidence intervals using the `binom.confint` function in `R`.
c. Compare the confidence intervals and discuss.

Problem 5. A laboratory experiment found that in a random sample of 20 frog eggs having aquaporins, 17 imploded when put into water.

a. Plot and interpret the posteriors for p assuming a Beta prior with parameters $(2, 2)$, $(1, 1)$, and $(0.5, 0.5)$.
b. Calculate and interpret the credible intervals for each of the Beta prior parameter settings. Note that the `R` package `binom` may be of use.

Problem 6. A study of the effectiveness of streptokinase in the treatment of patients who have been hospitalized after myocardial infarction involves a treated and a control group. In the streptokinase group, 20 of 150 patients died within 12 months. In the control group, 40 of 190 died within 12 months.

a. Construct the 95% confidence intervals that you consider appropriate for the probability of death in the control and treatment groups, respectively.
b. Do either of these confidence intervals cover 0.3? Interpret the fact that they do or do not cover 0.3.
c. Do confidence intervals for treatment and control overlap? Can we conclude something from this?

Problem 7. We have not discussed yet how to construct confidence intervals for the relative risk, p_T/p_C, where p_T and p_C are the probabilities of death in the treated and control groups, respectively. For the data in the previous problem use the bootstrap to construct a 95% confidence interval for p_T/p_C.

Problem 8. We would like to construct an 80% confidence interval for the prevalence of moderate to severe sleep disrupted breathing (`rdip4`≥ 15) in the SHHS, for men and women, respectively. Compare and contrast the various confidence intervals described in this chapter for these two prevalences.

Problem 9. Consider the estimated prevalence of moderate to severe sleep disrupted breathing (`rdip4`≥ 15) for men and women in the previous problem.

Follow the procedure below to quantify the small sample properties of the confidence intervals introduced in this chapter.

a. Repeat the following procedure 10000 times: sample 100 men and 100 women at random from the SHHS and calculate various confidence intervals for the prevalence based on these subsamples.
b. Compare the proportion of times the confidence intervals cover the corresponding prevalence estimated based on all SHHS data.
c. Compare the lengths of the confidence intervals.
d. Provide your code, description of the problem and conclusions in an Rmarkdown document

Problem 10. Same problem as before, but now vary the sample sizes from 10 to 500. Provide plots of relevant quantities (e.g., average coverage probability, average interval length) as a function of the sub-sample size.

Problem 11. Assume that a new treatment of lung cancer is effective in a proportion p of individuals and let X be the number of patients who are prescribed the drug until one patient does not respond to treatment.

a. What is the pmf of the variable X, its expected value, and the maximum likelihood estimator for p?
b. We would like to create a confidence interval using Bayesian priors for p. Obtain the posterior distribution of p if the prior is Beta(α, β).
c. Build these confidence intervals if the observed number of patients until one does not respond to treatment is $x = 5$ for various α and β parameters.
d. What parameters α and β are likely to be uninformative?
e. Plot the prior, posterior and likelihood in different scenarios.

Problem 12. In the previous problem we would like to obtain and contrast the performance of Bayesian confidence intervals.

a. For a grid of values of the proportion p between 0 and 1 simulate 10000 realizations of X.
b. For each realization build Bayesian confidence intervals using Beta distributions with parameters $(0, 0)$, $(0.5, 0.5)$, $(1, 1)$, $(2, 2)$, $(5, 5)$.
c. Compare the coverage probability of the confidence intervals of the true probability and their average length.

Problem 13. We would like to evaluate the confidence intervals for the success probability using the nonparametric bootstrap. Consider the case when we conduct $n = 10$ independent experiments, each with success probability p.

a. For $p = 0.5$ generate a number of successes, X, out of $n = 10$ experiments. For this realization of X resample the successes and failures with replacement $B = 10000$ times. Obtain a 95% confidence interval using equal tail probabilities based on the mean number of successes in the B samples
b. Redo the experiment $K = 10000$ times and check how often the confidence interval described in a. covers the true value of the parameter, $p = 0.5$.

c. Repeat the same experiment for a grid of values of the success probability, p, between $[0, 1]$ and plot the coverage and average length of the confidence intervals as a function of p.

Problem 14. Repeat the same simulation experiment from above, but replace the estimator X/n by $(X + 2)/(n + 4)$. This is the estimator suggested by the Agresti-Coull confidence interval of adding two successes and two failures to each experiment. What do you conclude?

Problem 15. Redo the same experiments from the previous two problems, but vary both the number of experiments, n, and the success probability. Say,

```
n=seq(10,300,by=5)
p=seq(0,1,by=0.01)
```

Problem 16. Propose a method for improving the coverage probability of the bootstrap confidence interval and compare your method with other confidence interval methods introduced in this chapter. Provide a report of your findings in Rmarkdown.

Chapter 15

Building a figure in ggplot2

This chapter covers the following topics

- The `qplot` function
- The `ggplot` function
- Strategies for improving plots
- Saving figures: devices
- Interactive graphics with one function
- Conclusions

The `ggplot2` package (Wickham 2016) adapted the philosophy of the "Grammar of graphics" (Wilkinson 2006) and implemented these concepts into a rigorous and versatile plotting framework in `R`. The main authority of how to do things on `ggplot2` is the ggplot2 book (Wickham 2016); there is additional information at http://ggplot2.org/. To some, especially those who have extensively used the base graphics in `R`, the syntax and process of building plots in `ggplot2` are not intuitive. Another concern is that `ggplot2` makes default graphs "nice enough" so that new users believe these are sufficient for distribution or publication without customizing of the graph. The base graphics in `R` can be quite basic and therefore users are less likely to distribute these without changing default parameters, except as a first rough draft.

We believe that there is one solid foundation for creating understandable, visually pleasing figures: planning, investing time, and putting in the required work. No default in any system is usually sufficient for your needs. Whether you want to change the size or font of labels, color aspects of the design, or add additional annotations for readers, these customizations take iterations and revisions like all aspects of research.

Here, we will discuss how to create a figure in `ggplot2` and adapt certain elements of the design. In no way is this an exhaustive tutorial, but it shows how to make a simple set of graphs and then customize the output to have a different

design.

15.1 The qplot function

We have shown many examples of how to create figures with the `plot` function in base `R`. The `qplot` function bridges the syntax of `R` with the functionality of `ggplot2`, which allows some users to more easily transition to `ggplot2`. If you are new to `R` and have not extensively used base `R` graphics, we suggest reading this section and trying `qplot` but mainly using the `ggplot` function in Section ggplot function. This suggestion is due to some of the limitations of the `qplot` function, which is a helpful function, but users prefer more customization.

To illustrate how to simply use `ggplot2` initially, we will use the `gapminder` dataset (www.gapminder.org), which contains:

> Health and income outcomes for 184 countries from 1960 to 2016. It also includes two character vectors, oecd and opec, with the names of OECD and OPEC countries from 2016.

We first load four important packages. The package `dslabs` contains the `gapminder` dataset, which will be used extensively in this chapter.

```
library(dplyr)
library(ggplot2)
library(tidyverse)
library(dslabs)
```

Let us select only the data from the United States for the first example. Here we select the columns of the country, year, and life expectancy and then `filter` the specific rows for the United States:

```
usa = gapminder %>%
  select(country, year, life_expectancy) %>%
  filter(country == "United States")
```

Next we pass in the data to `qplot` and map the variables for the `x` and `y` axes:

```
qplot(data = usa, x = year, y = life_expectancy)
```

In `qplot`, we put in the dataset and map variables from that dataset into some **aesthetics** of the plot. Note that unlike in `plot`, we do not have to use the `$` or subset the specific vectors we wish to map; `qplot` knows that they come from the dataset. Also, when we specify both `x` and `y`, `qplot` assumes we want to do a scatterplot. If we want to do a line plot, we simply have to change the `geom` (for geometry) in `qplot`.

```
qplot(data = usa, x = year, y = life_expectancy, geom = "line")
```

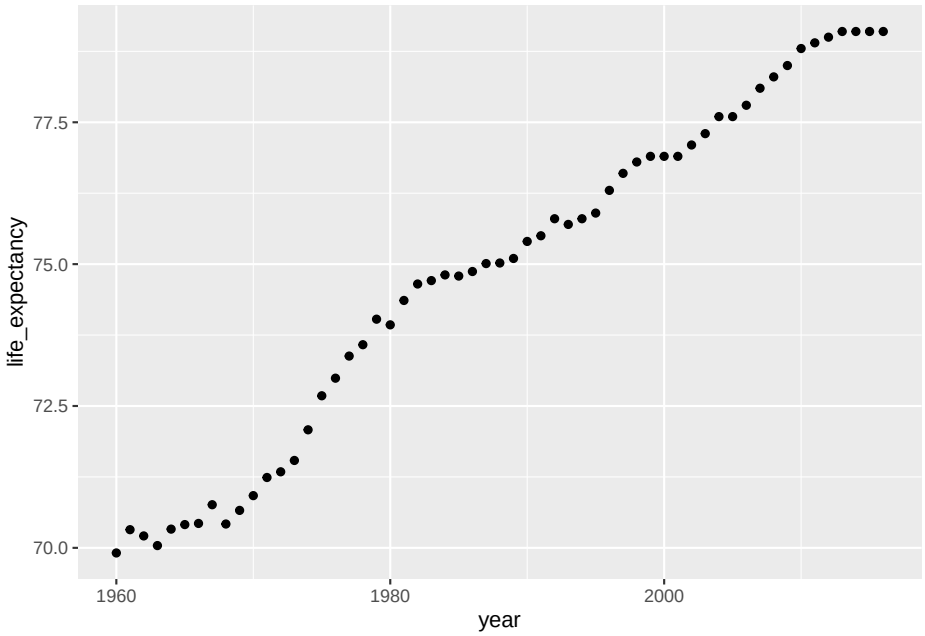


Figure 15.1: Life expectancy from 1960 to 2016 in USA.

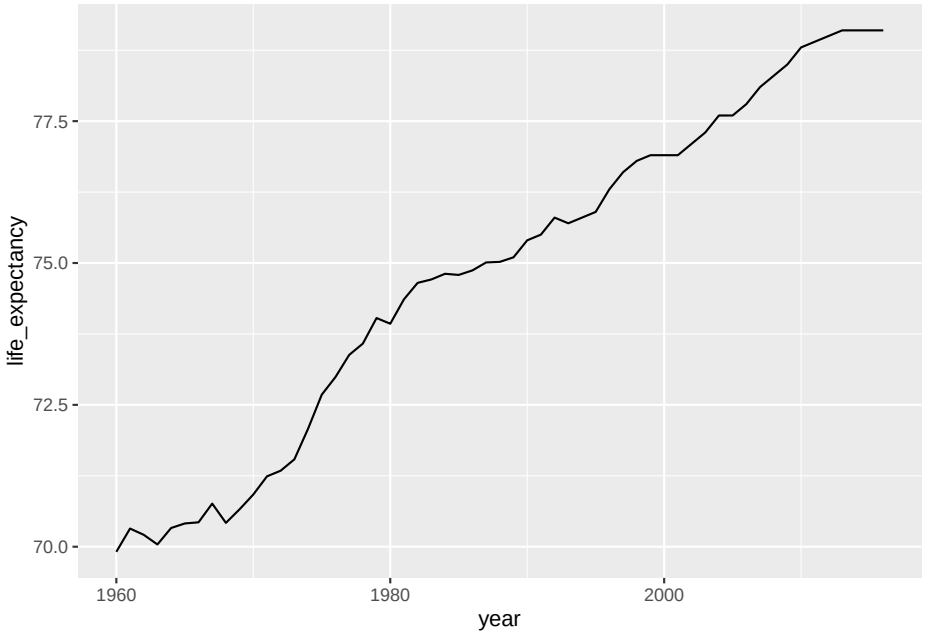


Figure 15.2: Life expectancy from 1960 to 2016 in USA shown as a line instead of dots.

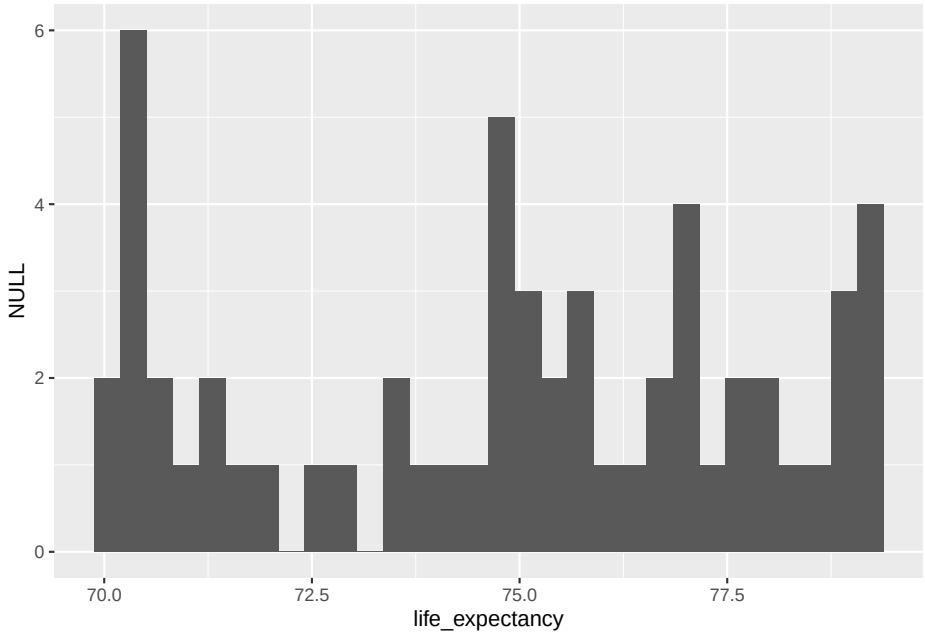


Figure 15.3: Histogram of life expectancy from 1960 to 2016 in USA.

When we only specify `x` in `qplot`, `qplot` assumes we want to do a histogram.

```
qplot(data = usa, x = life_expectancy)
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Again, we can change the type of plot by the `geom` argument, which indicates what type of geometry we want to add to the plot. Let us do a density plot instead of a histogram:

```
qplot(data = usa, x = life_expectancy, geom = "density")
```

Different from base graphics and the `plot` function, we can save this output to an object, let us call it `p`:

```
p = qplot(data = usa, x = life_expectancy, geom = "density")
```

Notice how when we assign the output to an object, nothing is plotted. It is like when we assigned a vector to an object `x`, nothing is printed out. Plot objects of `ggplot2` need to be "printed out" the same way as if we want them to be displayed:

```
print(p) # or just p
```

The ability to save this plot into an object will make it highly useful for changing parts of the plot, adding things to a plot, and making many versions of the same

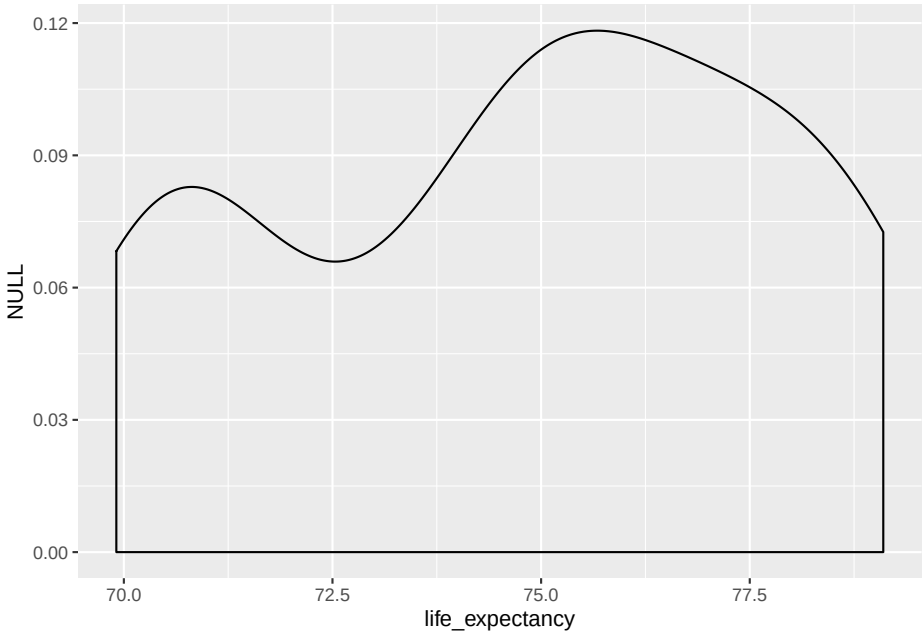


Figure 15.4: Probability density function of life expectancy from 1960 to 2016 in USA.

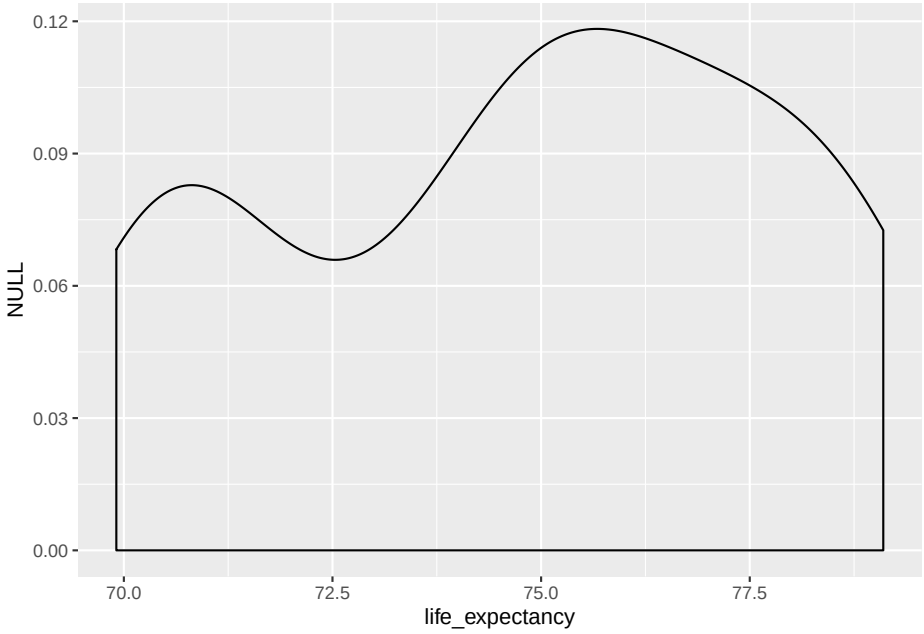


Figure 15.5: Probability density function of life expectancy from 1960 to 2016 in USA.

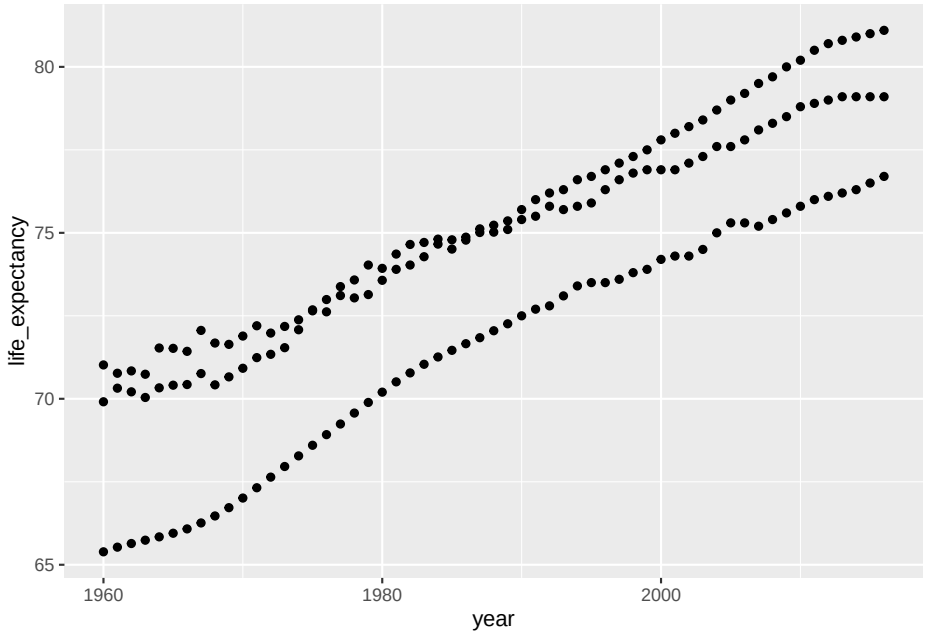


Figure 15.6: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina using `qplot`.

plot with different aesthetics or different datasets.

15.1.1 Mapping other aesthetics

Here we filter only the United States, United Kingdom, and Argentina from all the countries to illustrate how to plot over groups:

```
df = gapminder %>%
  select(country, year, life_expectancy, continent) %>%
  filter(country %in% c("United States", "United Kingdom", "Argentina"))
```

If we remake the plot like we did before, we see that the data are not automatically separated by country. Indeed, in Figure 15.6 data are plotted for all three countries, but dots are not separated by country:

```
qplot(data = df, x = year, y = life_expectancy)
```

We have to map the `country` variable to another aesthetic. Let us map the countries to different colors:

```
qplot(data = df, x = year, y = life_expectancy, colour = country)
```

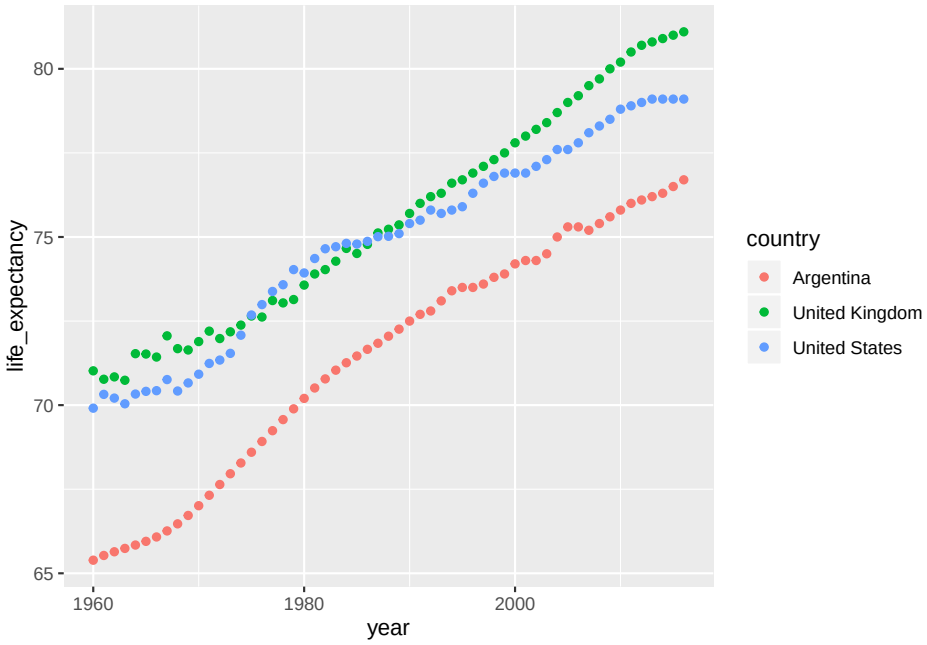


Figure 15.7: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina using `qplot` colored by country.

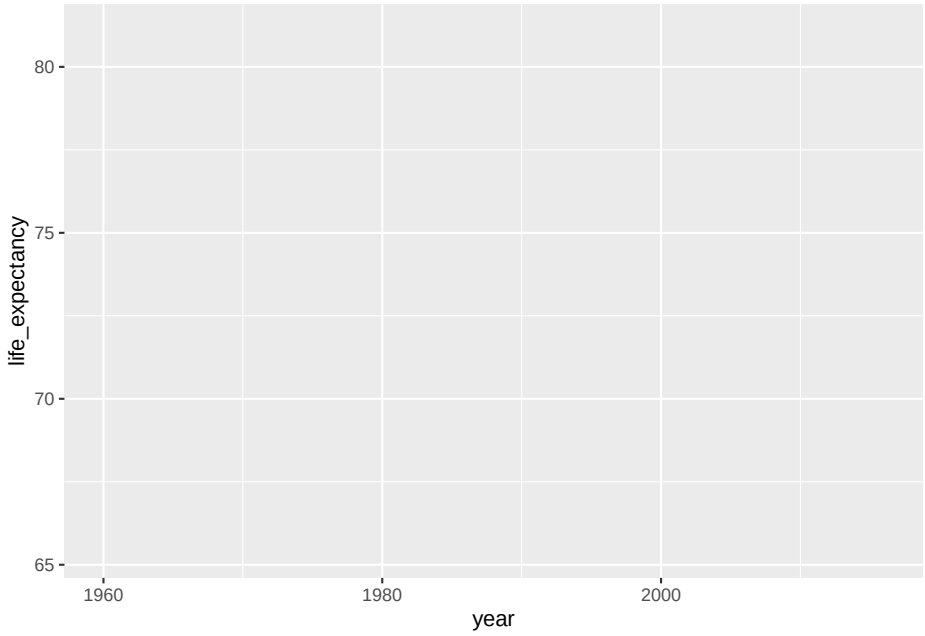


Figure 15.8: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina using `ggplot`, colored by country, but with no geometry.

Note, you can use `color` or `colour` to map onto color aesthetics. We see that we get different colors by countries as well as a legend (also known as a "guide" in the `ggplot2` lingo). We will show later how to manually set the colors (`scale_colour_manual`) or define a different palette.

15.2 The `ggplot` function

The generic function for creating a `ggplot2` plot is the `ggplot` function. As indicated above, it may be useful to get started with `qplot`, and http://ggplot2.org/book/qplot.pdf provides a great introduction to `qplot`. When creating general plots, however, we believe it's useful to use the `ggplot` function.

Let us create a `ggplot2` plot object using the `ggplot` function, passing in the data. You have to pass any aesthetics that are mapped to variables of your dataset inside the `aes()` function. **The most common error when starting creating plots in `ggplot2` is not putting variables in `aes()`.**

```
g = ggplot(data = df, mapping = aes(x = year, y = life_expectancy, colour = country))
print(g)
```

Consider the code that gave rise to the plot in Figure 15.8. Even though we

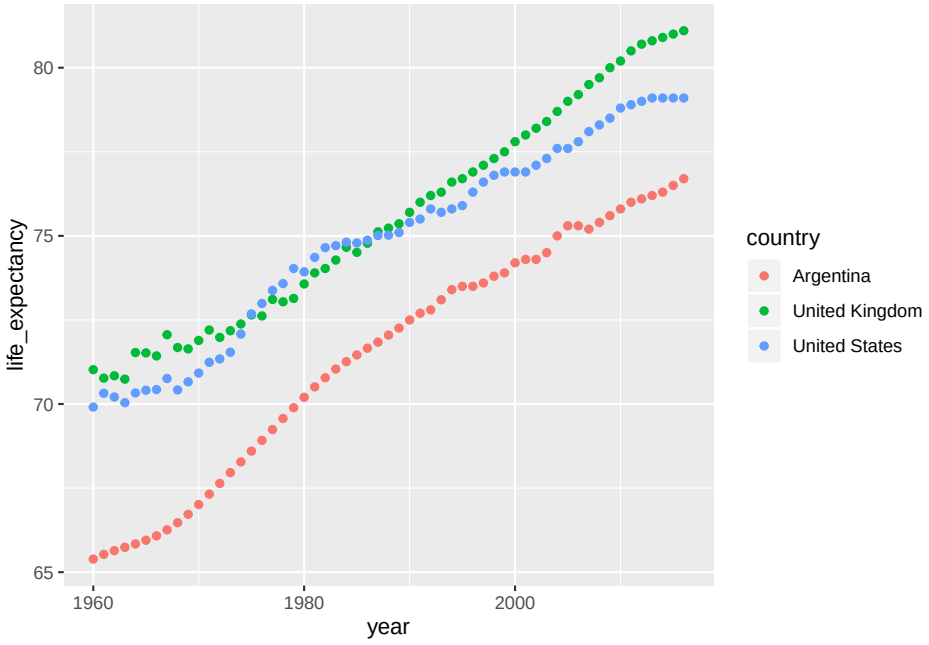


Figure 15.9: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

printed the object, nothing was plotted in the figure! That is because the `ggplot` function makes no assumptions about what you are plotting; it essentially sets the canvas, to which you have to add on top **layers**. Let us add a layer of points with `geom_point()`:

```
g + geom_point()
```

This shows that in `ggplot2` one literally "adds" layers using the addition sign (`+`). *Aside: the `ggplot2` package was created before the pipe (`%>%`) operator became popular and `ggplot2` is not planning on incorporating the pipe (See https://github.com/tidyverse/ggplot2/issues/1954 for some options to use `ggplot2` and the pipe.*

To break down what is going on, here is what `R` interprets (more or less):

1. Make a container/canvas with the data using the `ggplot` function.
2. Use the `data.frame` `df` : `data = df`.
3. Map certain "aesthetics" with the `aes` to three different aesthetics (`x`, `y`, `colour`) to certain variables from the dataset: `year`, `life_expectancy`, `country`, respectively.
4. Add a layer of geometric things to display, in this case points (`geom_point`).

Implicitly, ggplot2 notices that the x and y aesthetics are continuous, so maps them onto the plot using a "continuous" scale. We see that country is a factor or character column; the plot has a color bar but a "discrete" scale. Thus, the data types of the columns affect how ggplot2 creates the plot.

15.2.1 Adding multiple geometries

When setting aesthetics in ggplot(aes()), these are **global** aesthetics for your plot. If you add a layer and would like to change these aesthetics, you can add them inside a specific layer/geometry. This will be relevant when there are multiple layers (points and lines). If you look at the help of a geom (such as geom_point), you should note that each has the mapping and data arguments, like the ggplot function, and so we can change these within each geometry if desired.

```
og = ggplot(data = df, aes(x = year, y = life_expectancy))
```

Above we have created the og object, mapping our x and y variables. We will add some points, colored by country, and lines colored by continent:

```
g = og +
  geom_point(aes(colour = country)) +
  geom_line( aes(colour = continent) )
g
```

Figure 15.10 added a line connecting the dots for United States and Argentina, both countries belonging to Americas. The plot is not that informative with respect to the lines, though it illustrates how to map colors to different aspects of the data for different geometries. Here we will map the color of a point to the country, as well as the type of line plotted:

```
og +
  geom_point(aes(colour = country)) +
  geom_line( aes(linetype = country) )
```

Here we see that the different countries have different colors and different line types, but the line types are not colored differently. This would be different if we had put colour = country in the global ggplot aesthetics.

15.2.2 Passing in multiple datasets

Moreover, you can pass in different **datasets** to different geometries. For example, let us calculate the median life expectancy at each year for all countries.

```
medians = gapminder %>%
  select(year, life_expectancy) %>%
```

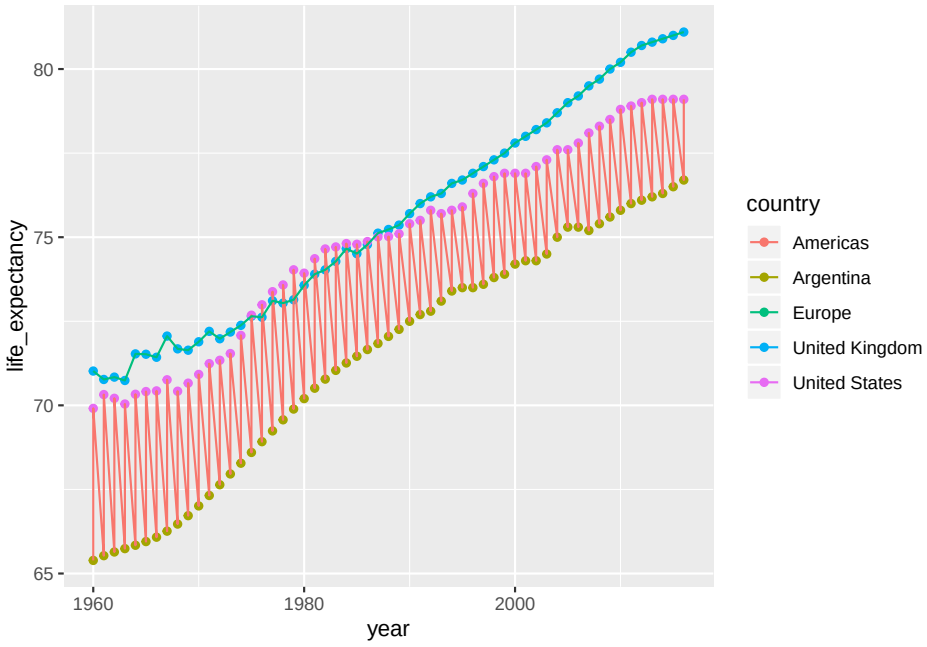


Figure 15.10: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

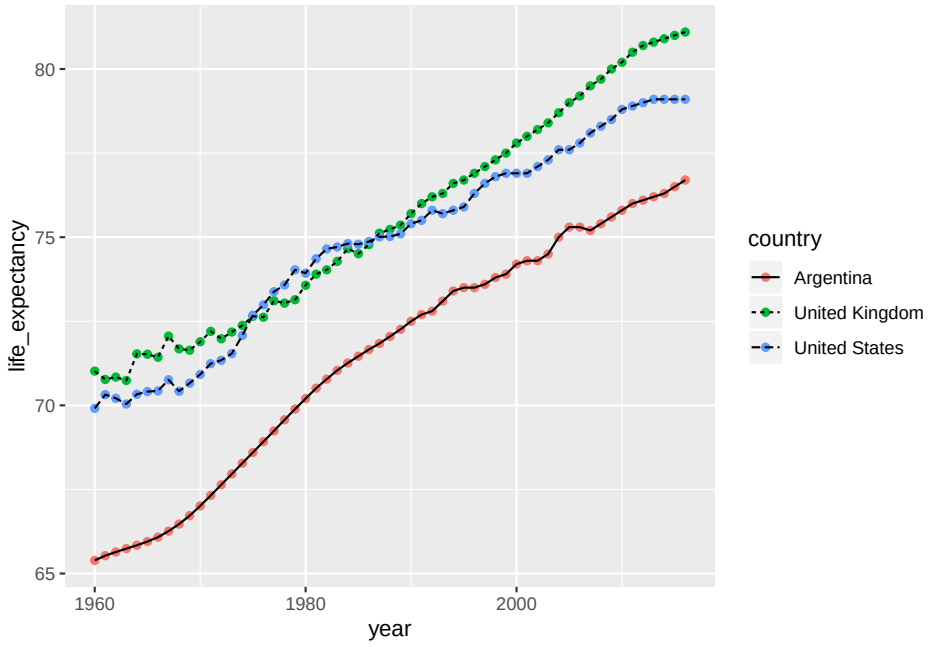


Figure 15.11: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

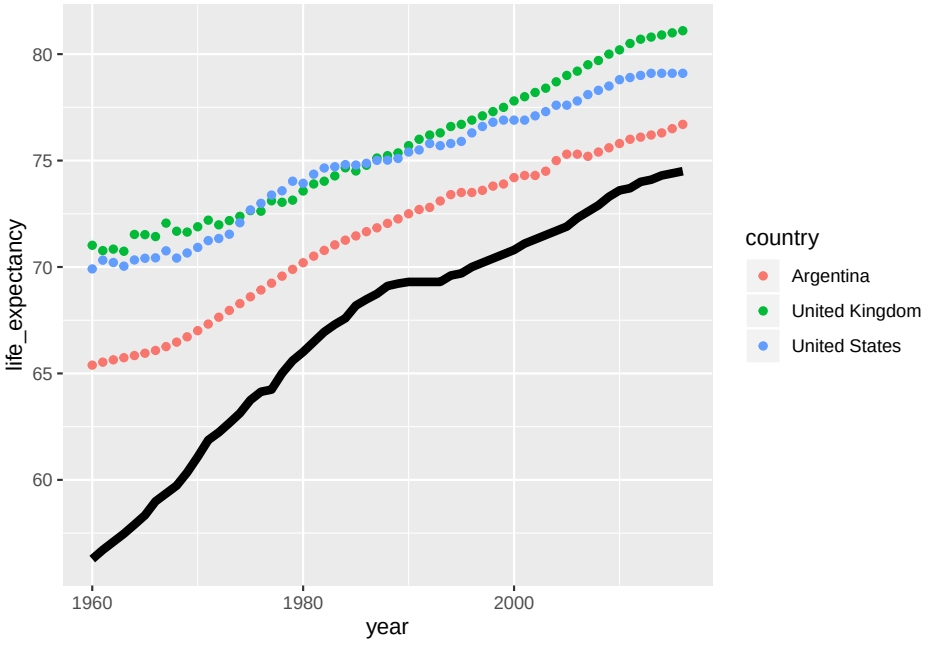


Figure 15.12: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina and median life expectancy for all countries.

```
  group_by(year) %>%
  summarise(life_expectancy = median(life_expectancy, na.rm = TRUE))
```

Let us plot this with our other data from the subset of countries we had:

```
og +
  geom_point(aes(colour = country)) +
  geom_line(data = medians, size = 2)
```

We see that we did not have to specify the `x` or `y` variables in the `geom_line` argument. This convenience is due to the fact that our median life expectancy dataset has a `year` and `life_expectancy` column, and the `ggplot` object `og` knows to map those to `x` and `y`. If we do not have the same column names, then we will get an error. Let us rename the `life_expectancy` column to `lexp` and rerun the same code with this data set (it should error):

```
renamed_meds = medians %>%
  dplyr::rename(lexp = life_expectancy)
```

```
og +
  geom_point(aes(colour = country)) +
  geom_line(data = renamed_meds, size = 2)
```

```
Error in FUN(X[[i]], ...): object 'life_expectancy' not found
```

This error does not imply that you need to have the same column names all the time when passing in multiple datasets, but you would have to re-define the mapping in those areas. The same code will run if we change the `aes` for that `geom_line`:

```
og +
  geom_point(aes(colour = country)) +
  geom_line(data = renamed_meds, aes(y = lexp), size = 2)
```

Note, you can assign aesthetics within a `geom` **outside** of using `aes()`, but these are not assigned from a variable. In the above examples, we set `size = 2` to indicate the line width to be size 2 for the overall median life expectancy. This aesthetic was not mapped to a variable, but directly set for that geometry.

For example, we can have the data colored by different countries for each point, but have the lines for each country separated, though all of them colored red:

```
ggplot(data = df, aes(x = year, y = life_expectancy,
                      colour = country)) +
  geom_point() + geom_line(colour = "red")
```

The assignment of `colour = "red"` will override the `colour` aesthetic set in `ggplot` for all the points (as it was set in `geom_point`), but it will still be inherited when we run `geom_line`.

Now it is much easier to work with plot as an object. Saving this as an object

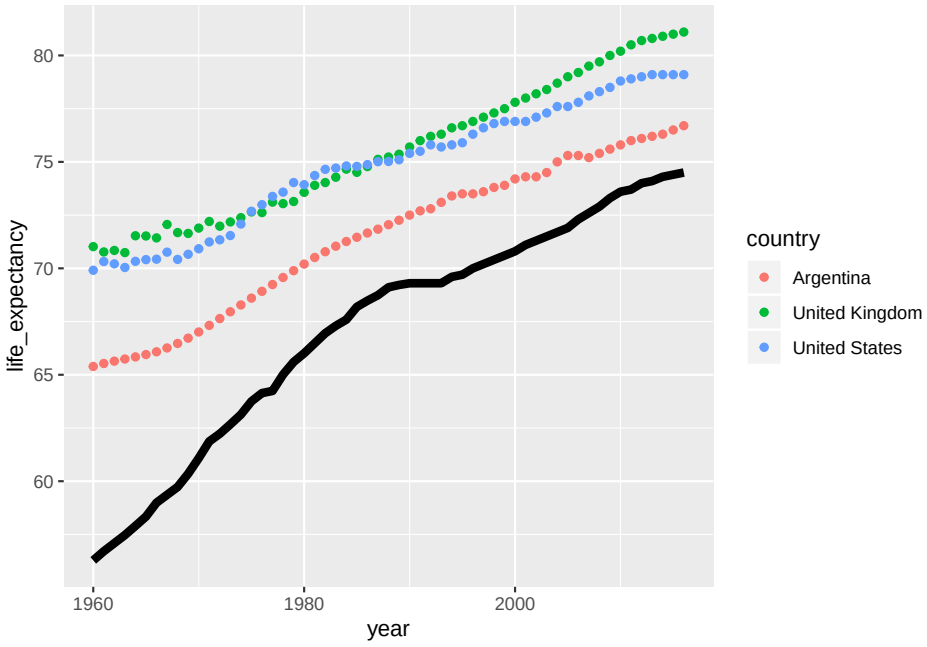


Figure 15.13: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina and median life expectancy for all countries.

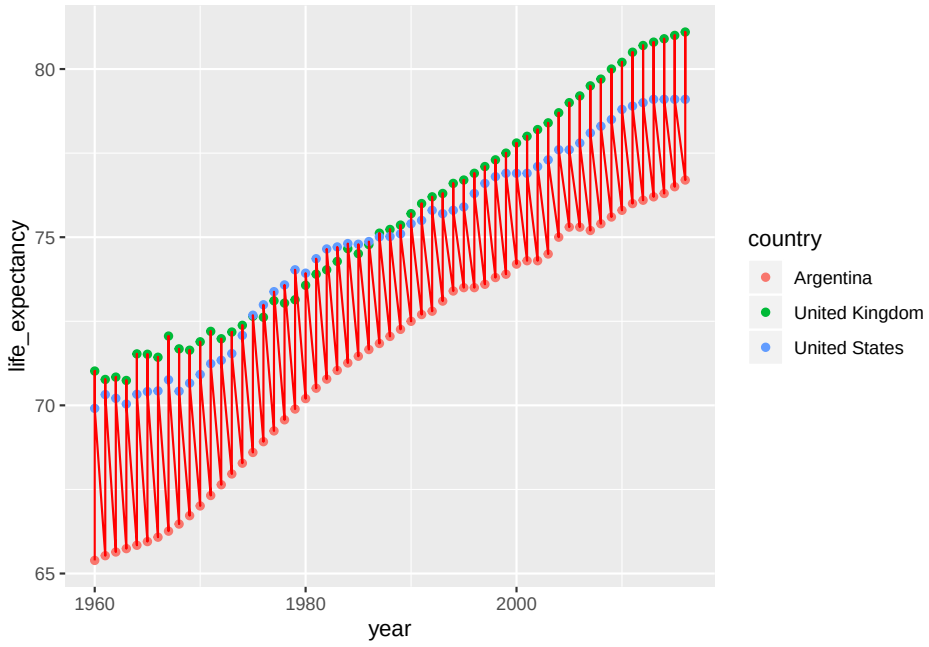


Figure 15.14: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

also makes it less likely for copy/paste errors and reduces the time to fix a plot if large changes need to be made.

15.3 Strategies for improving plots

In the above plots, there are elements that are required to make this plot more "production ready":

1. make the axes/labels bigger
2. use full names for labels
3. maybe add a title
4. change the legend title
5. improve the legend
6. place the legend INSIDE the plot

As such, we go through each step and show how to do implement each operation

15.3.1 Make the axes/labels bigger

First off, let us assign this plot to an object, called `g`:

```
g = ggplot(data = df,
           aes(x = year, y = life_expectancy, colour = country)) +
  geom_point()
```

Now, you can simply call `print(g)` to show the plot, but the assignment will not do that by default. If you simply call `g`, it will print/show the object (as other R objects do), and plot the graph.

15.3.2 The `theme` function - get to know it

One of the most useful `ggplot2` functions is `theme`. Read the documentation (`?theme`). There is a slew of options, but we will use a few of them for this and expand on them in the next sections.

15.3.3 Setting a global text size

We can use the `text` argument in `theme` to change **ALL** the text sizes to a value. Now this is where users who have never used `ggplot2` may be a bit confused. The `text` argument (input) in the `theme` command requires that `text` be an object of class `element_text`. If you look at the `theme` help for the `text` argument, it says "`all text elements (element_text)`". This means you can not just say `text = 5`, you must specify `text = element_text()`.

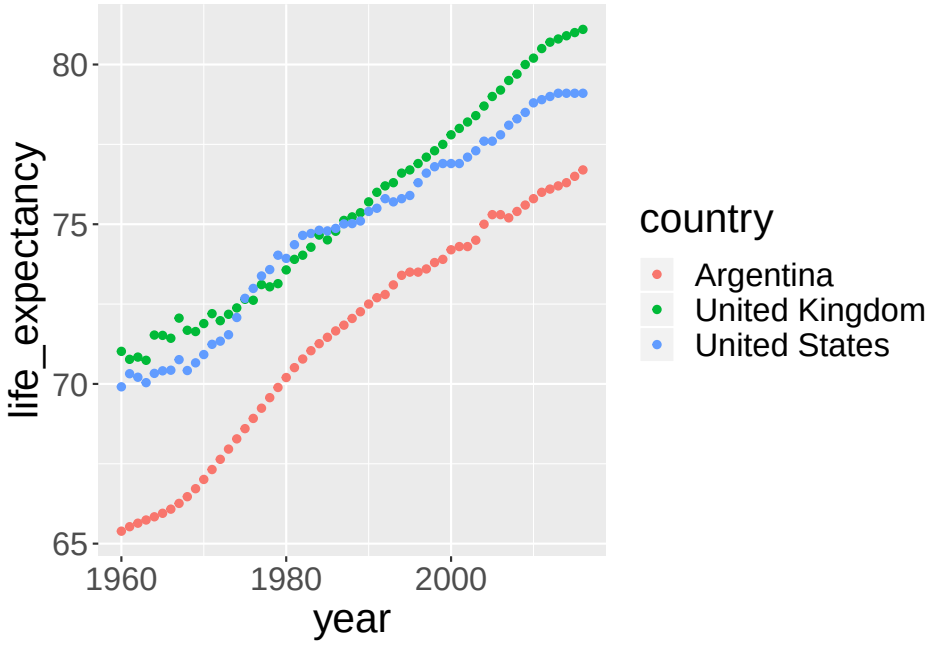


Figure 15.15: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

As text can have multiple properties (`size`, `color`, etc.), `element_text` can take multiple arguments for these properties. One of these arguments is `size`:

```
g + theme(text = element_text(size = 20))
```

Again, note that the `text` argument/property of theme changes all the text sizes. Let's say we want to change the axis tick text (`axis.text`), legend header/title (`legend.title`), legend key text (`legend.text`), and axis label text (`axis.title`) each to a different size:

```
gbig = g + theme(axis.text = element_text(size = 18),
                 axis.title = element_text(size = 20),
                 legend.text = element_text(size = 15),
                 legend.title = element_text(size = 15))
gbig
```

Now, we still have the plot `g` stored, but we make a new version of the graph, called `gbig`, with the larger text sizes.

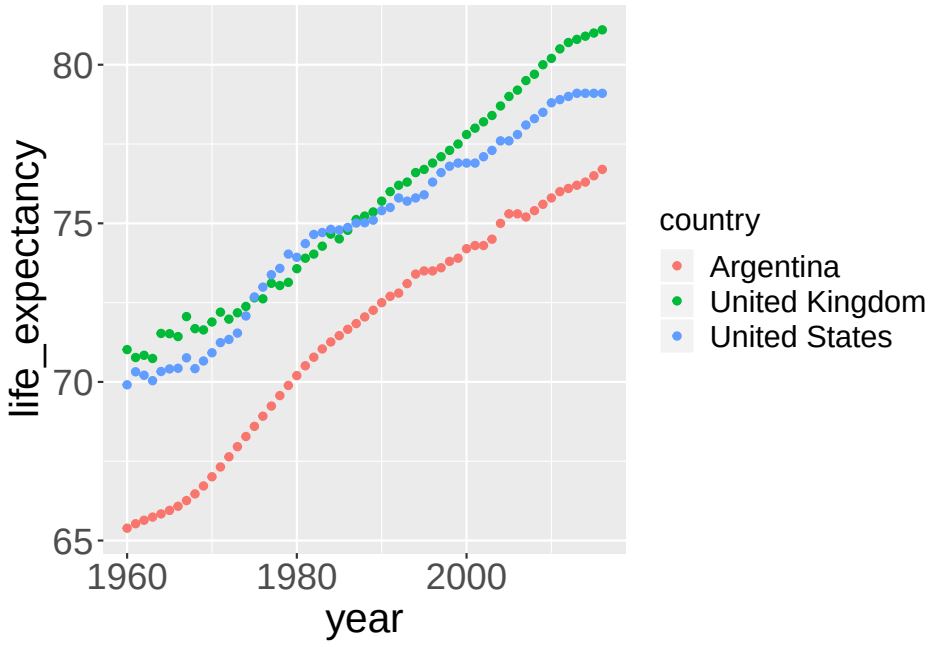


Figure 15.16: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

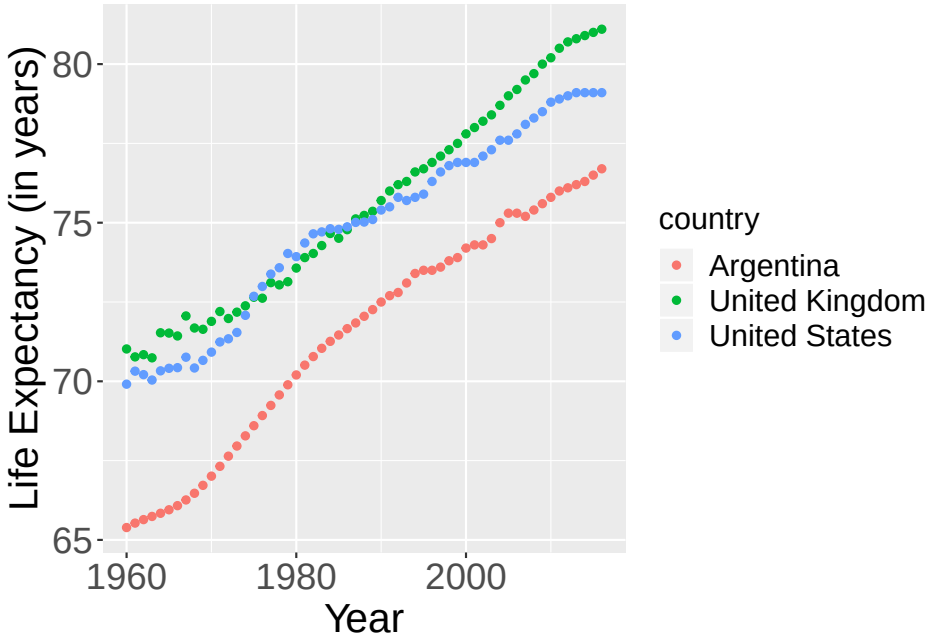


Figure 15.17: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

15.3.4 Use full names for labels

To change the x or y labels, you can just use the `xlab`/`ylab` functions:

```
gbig = gbig + xlab("Year") + ylab("Life Expectancy (in years)")
gbig
```

We want to keep these labels for future plots, building on `gbig`, so we overwrote `gbig`.

15.3.5 Maybe add a title

Now, one may assume there is a `main()` function from `ggplot2` to give the title of the graph (as this is the argument in `plot`), but that function is `ggtitle()`. Note, there is a `title` command in `base` R, so this was not overridden in `ggplot2`. The title can be added in the same way as adding a layer and other labels:

```
gbig + ggtitle("Life Expectancy Changes over Time")
```

Note, the title is smaller than the specified axes label sizes by default. Again if we wanted to make that title bigger, we can change that using `theme`:

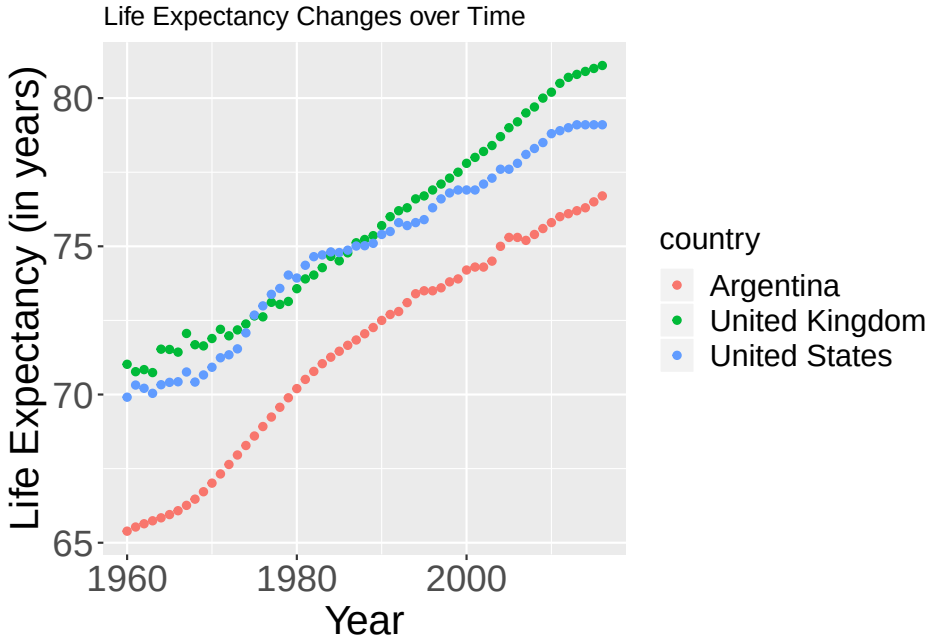


Figure 15.18: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

```
gbig +
  ggtitle("Life Expectancy Changes over Time") +
  theme(title = element_text(size = 20))
```

I will not reassign this to a new graph, because in some publication-ready figures the title is placed in the figure legend instead of the graph itself.

15.3.6 Improve the legend

Now let us change the header/title of the legend. We can do this using the `guides` function:

```
gbigleg_orig = gbig +
  guides(colour = guide_legend(title = "Country"))
gbigleg_orig
```

Here, `guides` takes arguments that are the same as the aesthetics (e.g., color, linetype) from before in `aes`. Also note, that `color` and `colour` are aliased so that you can spell it either way (the `ggplot2` author is from NZ).

Let us also adjust the horizontal justification, so the legend title is centered:

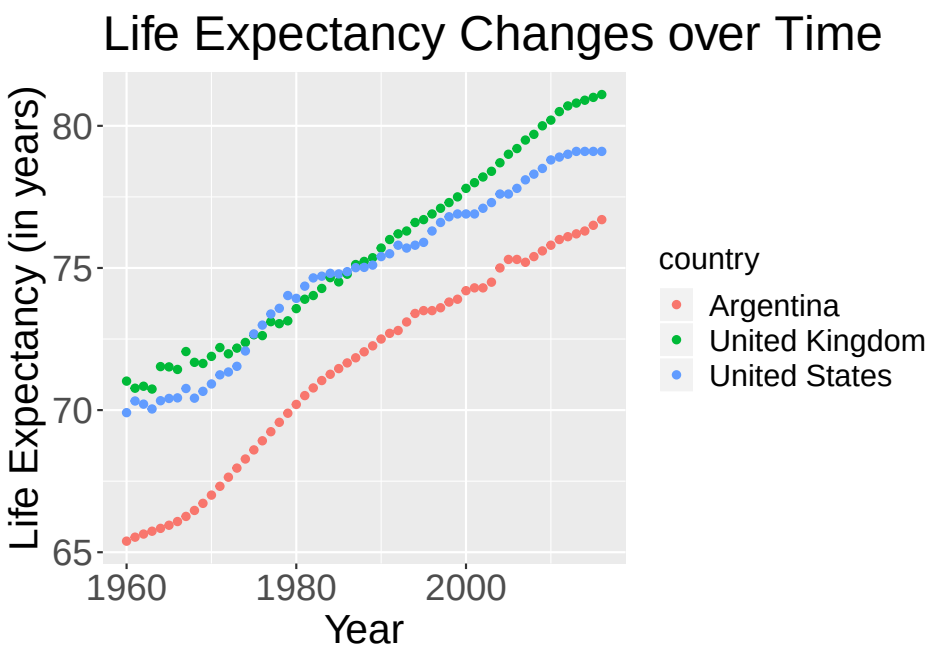


Figure 15.19: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

```
gbigleg = gbig +
  guides(colour = guide_legend(title = "Country",
                               title.hjust = 0.5))
gbigleg
```

That looks better for the legend in our opinion, but we still have a lot of wasted space.

15.3.6.1 Place the legend INSIDE the plot

In many cases having the legend **inside** the plot increases the size of actual figure. To do this, we can use the `legend.position` from the themes:

```
gbigleg +
  theme(legend.position = c(0.35, 0.3))
```

Now, there seem to be a few problems in Figure 15.22:

1. There may not be enough space to place the legend
2. The legend may mask out points/data

For problem 1., we can either make the y-axis longer, the legend smaller, or do

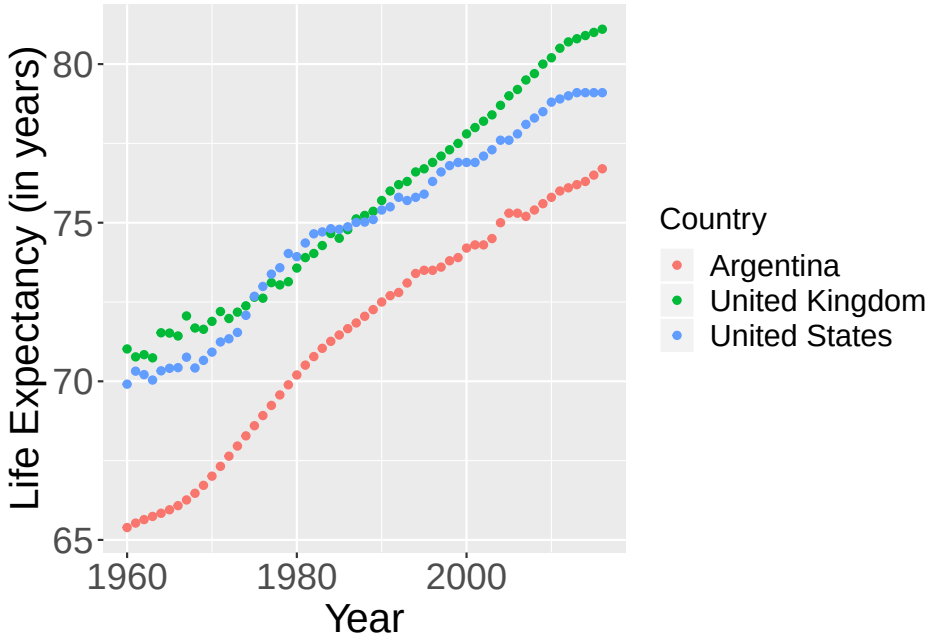


Figure 15.20: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

both. In this case, we do not have to change the axes, but you can use `ylim` to change the y-axis limits:

```
gbigleg +
  theme(legend.position = c(0.3, 0.35)) +
  xlim(c(2000, max(df$year)))
```

```
Warning: Removed 120 rows containing missing values (geom_point).
```

It is best not to do this as a large area without information has been added to the plot. We show now how to make the legend "transparent" so we can at least see if any points are masked out and to make the legend look like a more inclusive part of the plot.

15.3.6.2 Making a transparent legend

There is a helper "function" `transparent_legend` that will make the box around the legend (`legend.background`) transparent and the boxes around the keys (`legend.key`) transparent as well. Like `text` before, we have to specify boxes/backgrounds as an `element` type, but these are rectangles (`element_rect`) compared to text (`element_text`).

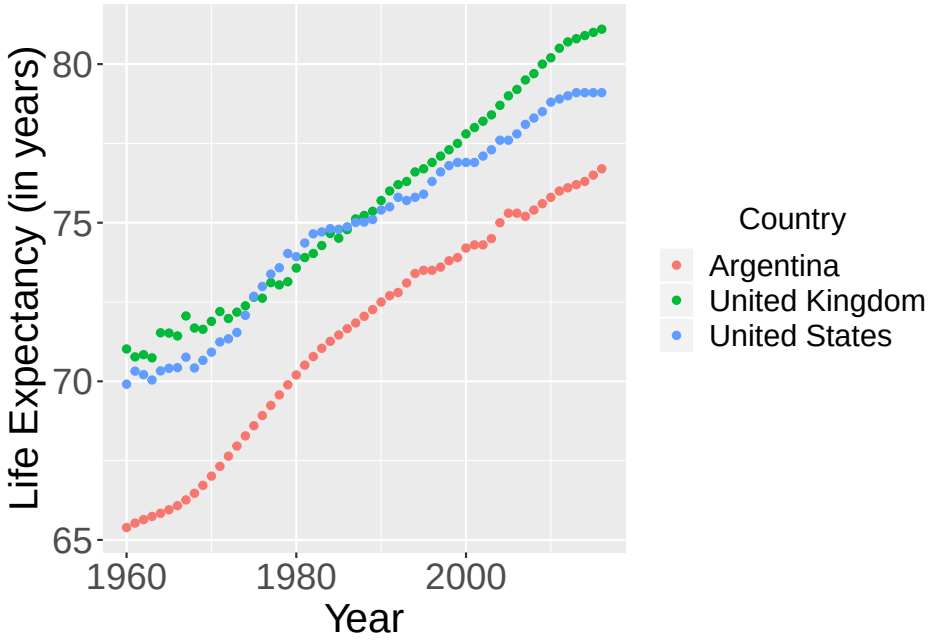


Figure 15.21: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

```
transparent_legend =  theme(
  legend.background = element_rect(fill = "transparent"),
  legend.key = element_rect(fill = "transparent",
                            color = "transparent") )
```

One nice thing is that we can save this as an object and simply "add" it to any plot for which we want a transparent legend. Let us add this to what we had and see the result:

```
gtrans_leg = gbigleg +
  theme(legend.position = c(0.7, 0.35)) +
  transparent_legend
gtrans_leg
```

15.3.6.3 Moving the title of the legend

Now, everything in `gtrans_leg` looks acceptable (to us) except for the legend positioning and title. We can move the title of the legend to the left hand side:

```
gtrans_leg + guides(colour = guide_legend(title.position = "left"))
```

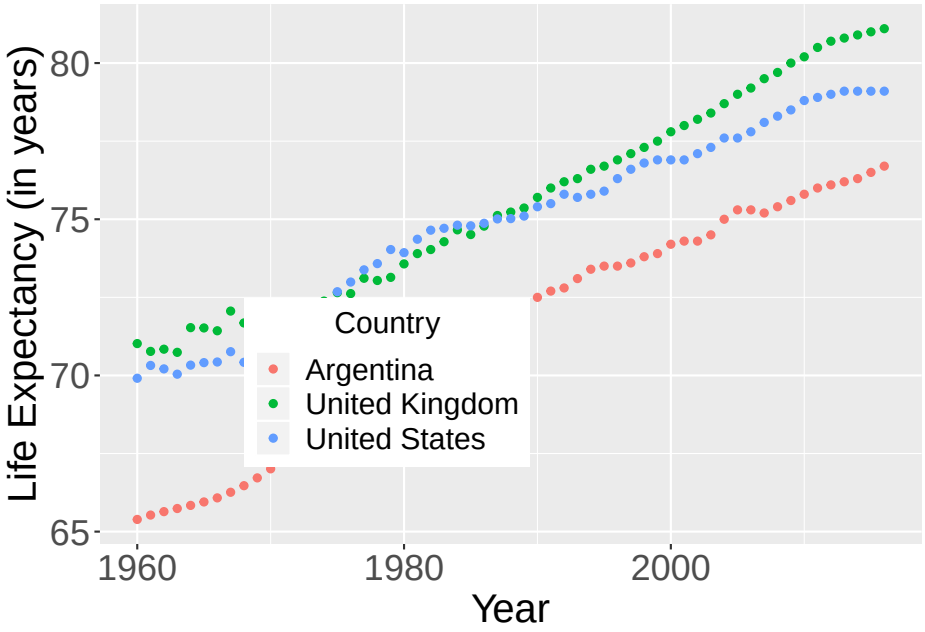


Figure 15.22: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

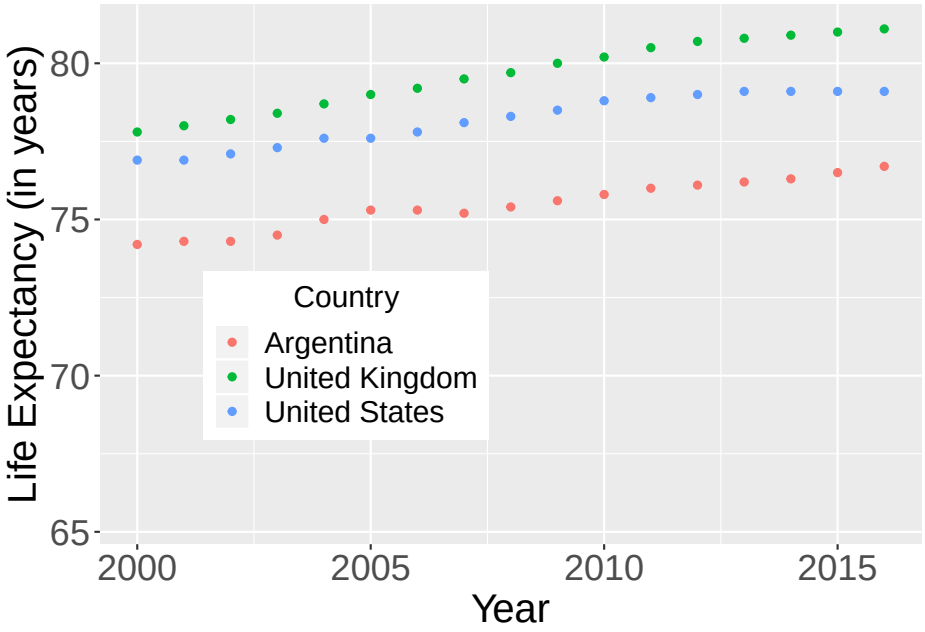


Figure 15.23: Life expectancy from 2000 to 2016 in USA, United Kingdom and Argentina with a legend.

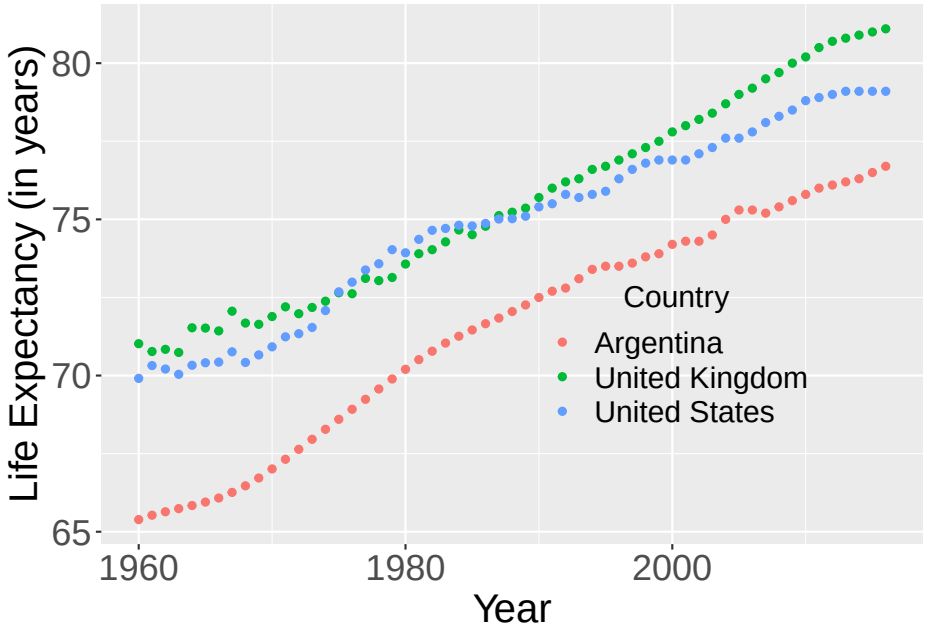


Figure 15.24: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a transparent legend inside the plot.

We lost our other changes to `guides`! Note, that if you **respecify** the guides, you must make sure you do it all in one shot (easiest way):

```
gtrans_leg + guides(
  colour = guide_legend(title = "Country",
                        title.hjust = 0.5,
                        title.position = "left"))
```

15.3.6.4 A little more advanced

Having to change `guides` all at once is not entirely accurate, as we could dig into the `ggplot2` object and assign a different `title.position` property to the object after the fact.

```
gtrans_leg$guides$colour$title.position = "left"
gtrans_leg
```

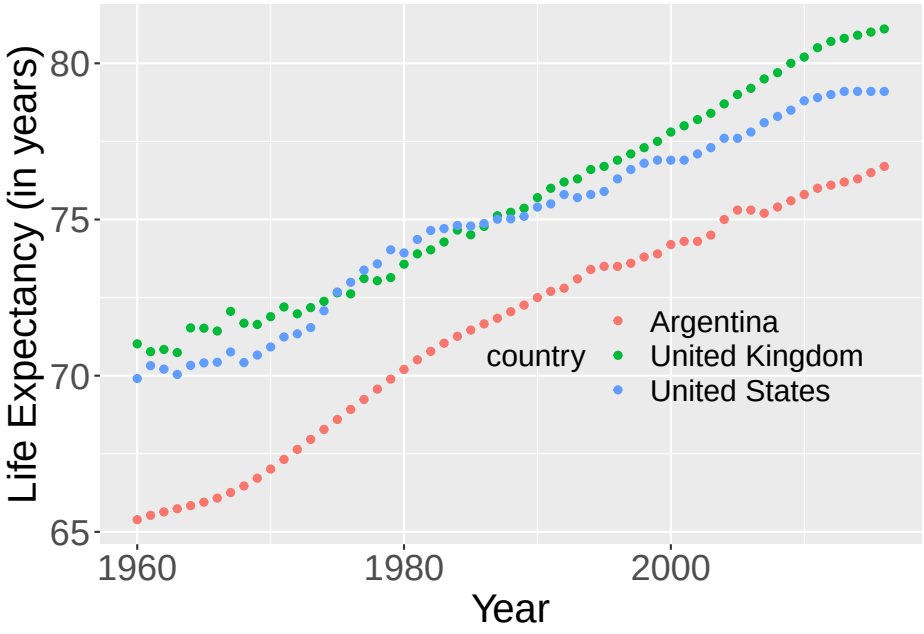


Figure 15.25: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a transparent legend inside the plot, but the title on the left side.

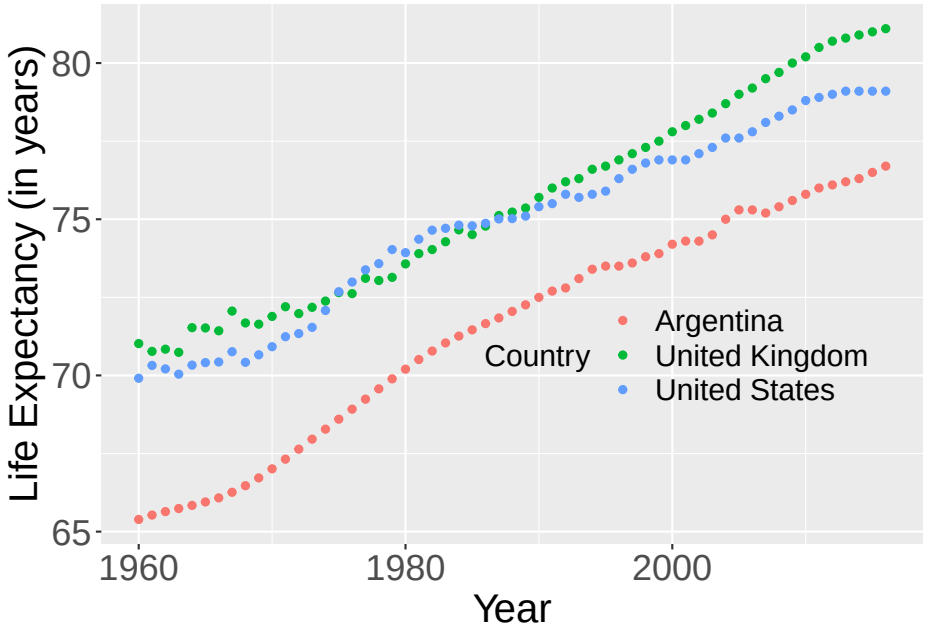


Figure 15.26: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a transparent legend with a recoded title.

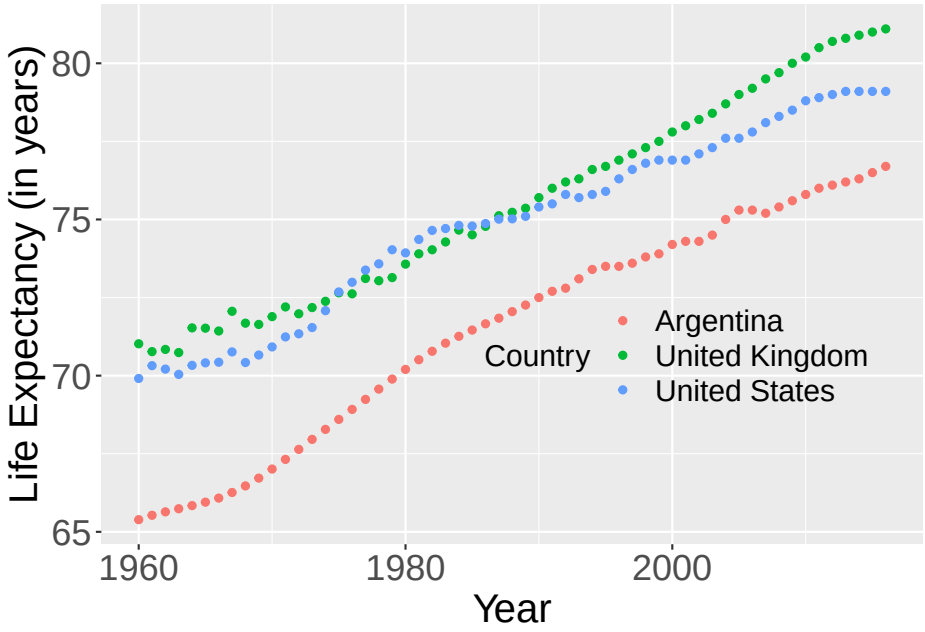


Figure 15.27: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina.

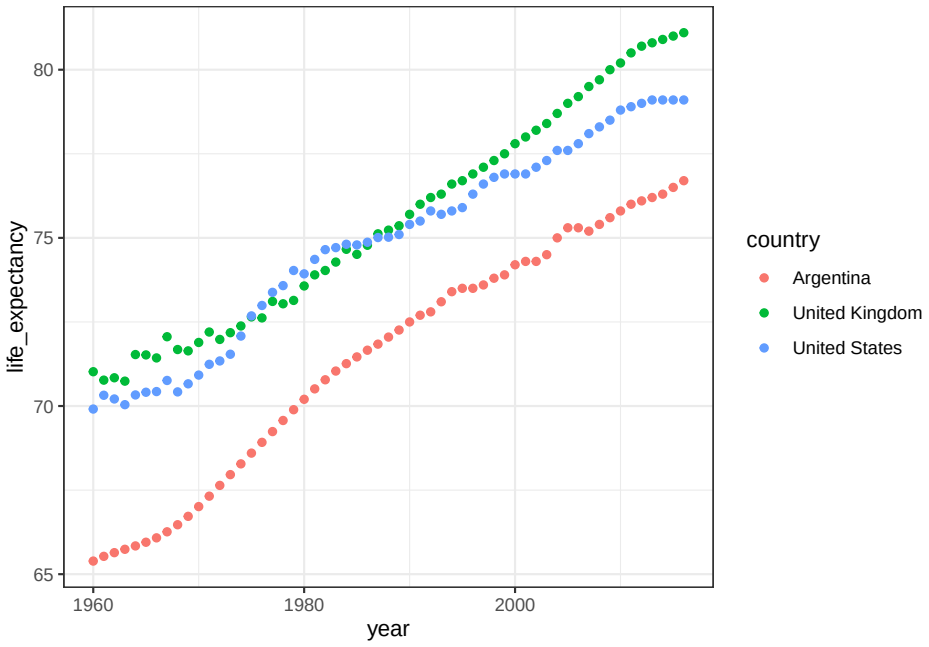


Figure 15.28: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a black and white theme.

15.3.7 "I don't like that theme"

People often like the grammar of `ggplot2` but not the default theme. The `ggthemes` package has some good extensions of theme from `ggplot2`, but there are also many themes included in `ggplot2`, which should be specified before changing specific elements of `theme` as done above. Here we will add the black and white theme:

```
g + theme_bw()
```

the dark theme:

```
g + theme_dark()
```

a minimalistic theme:

```
g + theme_minimal()
```

and the classic base `R` theme:

```
g + theme_classic()
```

Look at the documentation of the `ggthemes` package for more options.

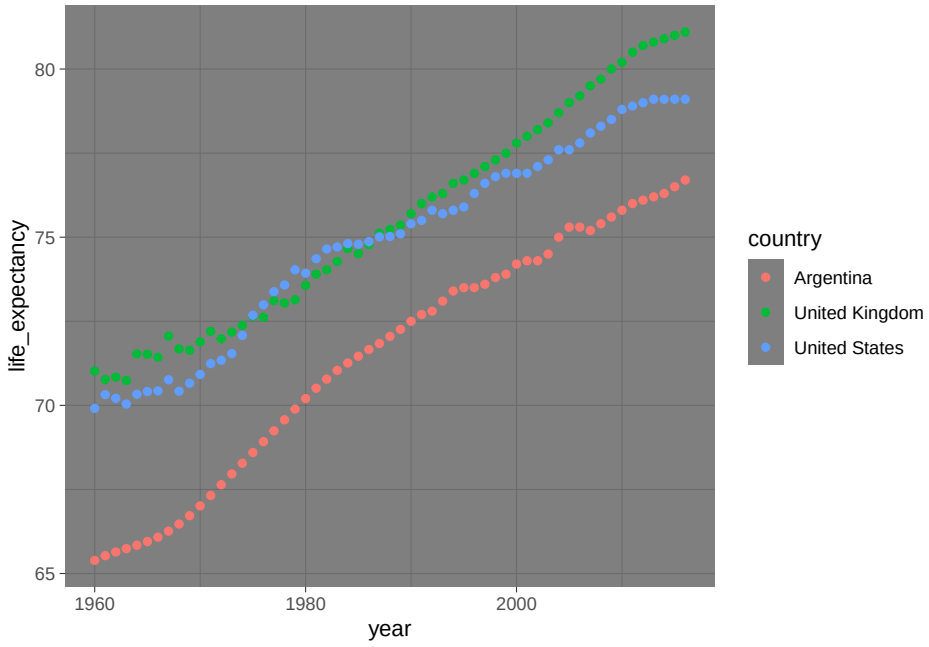


Figure 15.29: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a dark theme.

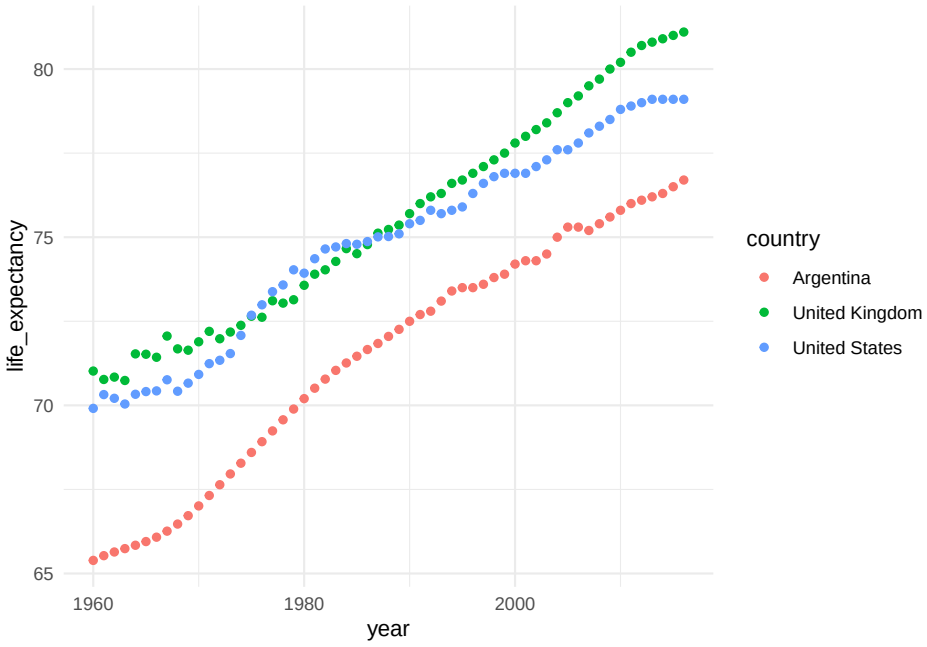


Figure 15.30: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a minimal theme.

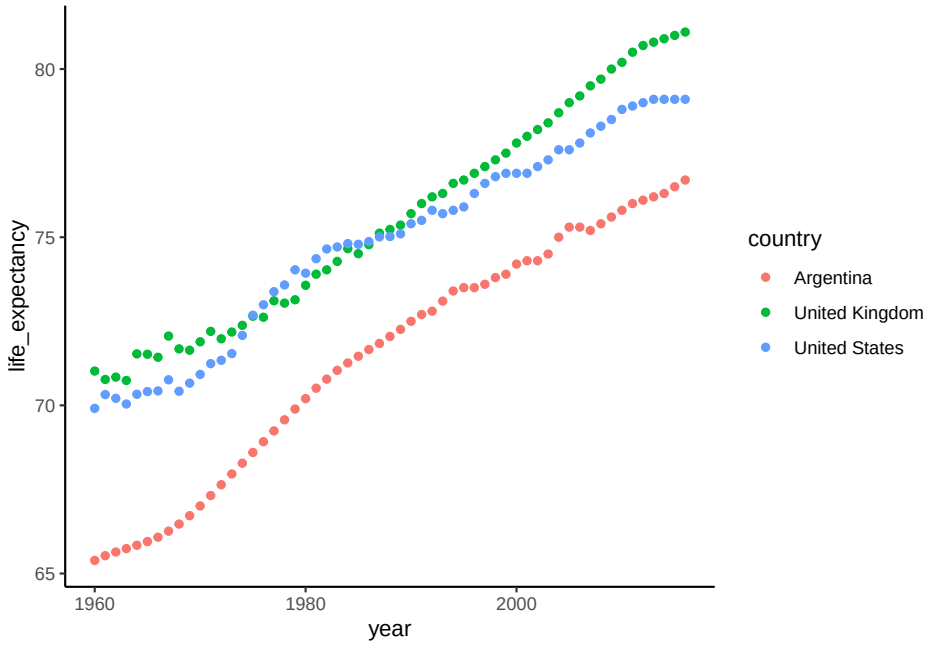


Figure 15.31: Life expectancy from 1960 to 2016 in USA, United Kingdom and Argentina with a classic theme.

15.4 Saving figures: devices

By default, R displays plots in a separate panel or in your viewer in RStudio. From there, you can export the plot to a variety of image file types, or copy it to the clipboard. However, you should be saving plots using code. The way R does this as follows:

1. You "open a device": usually using `pdf()`, `bmp()`, `jpeg()`, `png()`, or `tiff()`.
2. You print out a plot, using `print(ggplot2 object)` or base R plots.
3. You close the device using `dev.off()`.

See the help file using `?png` for the arguments. In each device there are different arguments for how to save the plot. Those include the height/width of the plot and the image resolution. In the `pdf` device, you can have multiple plots. For example:

```
pdf("filename.pdf", width=8, height=8) # inches -open device
plot() # plot 1
plot() # plot 2
dev.off()
```

Basically, you are creating a pdf file, and telling R to write any subsequent plots to that file. Once you are done, you turn the device off. Note that failing to turn the device off will create a pdf file that is corrupt and cannot be open.

For the other devices, you can only have one plot in that file (it can have multiple facets, but only one plot call). If you performed the same operation with a `png` versus a `pdf`:

```
png("filename.png") # inches -open device
plot() # plot 1
plot() # plot 2
dev.off()
```

Then the resulting `png` would only have the result of plot 2 in the output file. If you are using RMarkdown, you simply need to print the plots and these are embedded/saved in the background. Many times, you need to customize the output for certain file sizes/resolutions. This customization is possible in RMarkdown, but can be more intuitive using devices. Also, you may not be using RMarkdown but simply an R script, which does not have the same document format, and then you need to use the device open/close methods.

If you forget to close a device (or an error occurs) in an interactive R session and then do a plot, nothing will come up in your viewer because R still thinks you want to plot to that device. If this occurs, you can check on which device you are using `dev.cur()`, or simply turn off any open devices using `dev.off()` until you get to the "null device" (default plotting window).

In ggplot2, there is also a convenient function for saving a plot called ggsave. Please look at the documentation (using ?ggsave) for more information.

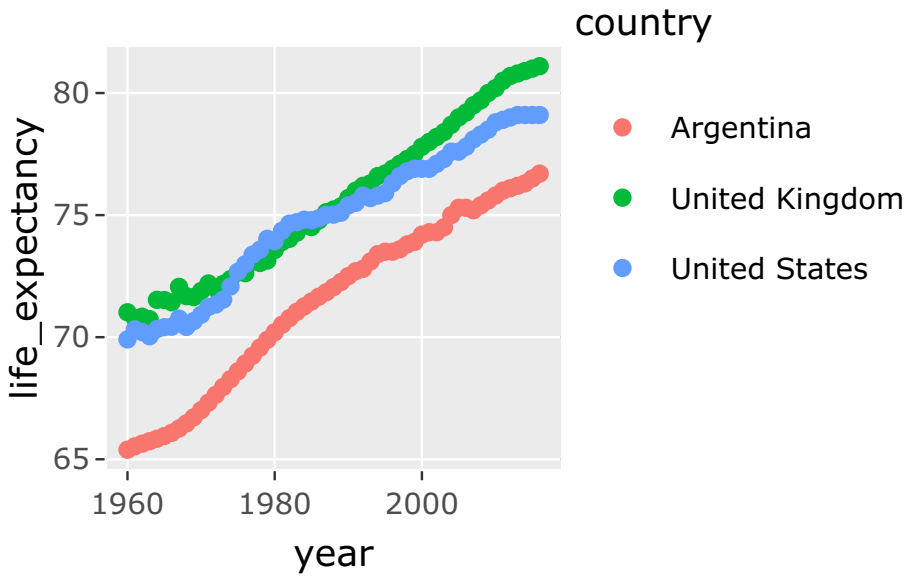
15.5 Interactive graphics with one function

WARNING: this section may shock you.

If the aforementioned aspects of ggplot2 did not convince you it was worth the effort of a new syntax, especially if you have used base R graphics, this section may do that. The plotly package (Sievert et al. 2017) in R, created by Carson Sievert, creates versatile ways to get plots into the plotly.js JavaScript library framework (https://plot.ly/). The plotly package has its own syntax, slightly different from ggplot2. One amazing function is the ggplotly function. This function alone is a reason to learn some ggplot2 even if you are an expert on base R graphics.

Recall that we have a ggplot2 object named g that was built using ggplot2 syntax. Suppose that we wanted an interactive graphic? You can simply load plotly and run the ggplotly(g) command and voila! It's interactive!

```
plotly::ggplotly(g)
```



Note, this interactivity requires a session where a web browser is available (such as in RStudio), or the output of the dynamic document needs to support HTML (e.g., PDFs will not work). If you are using the interactive HTML version of the book, you should be able to select and manipulate the plot above.

Let us step back and note the magic that just happened. We had a plot in ggplot2, used the ggplotly function, and out came a plotly graph! Although that seems straightforward, that task is remarkably difficult and many other plotting systems require completely different syntax to make things interactive.

15.6 Conclusions

The ggplot2 framework is a powerful plotting tool for visualization and data exploration. Users switching from base R graphics may want a gentle introduction using qplot. New users should jump directly into the ggplot functionality as this method gives more flexibility. That said, ggplot2 can deceive new users by making graphs that look "good"-ish. This may be a detriment as users may believe the graphs are good enough, when they truly need to be changed. The changes are available in base or ggplot2 and the overall goal was to show how the recommendations can be achieved using ggplot2 commands.

Below, we discuss some other aspects ggplot2 plotting, where you can make quick-ish exploratory plots. We believe, however, that ggplot2 is not the fastest framework when you want to simply learn the basic exploratory plots. This learning curve is not very steep in our opinion. Moreover, we believe that using ggplot2 when starting will allow users to learn one system for all their types of plots. When the complexity of the plots increases dramatically, such as multiple groups and smoothers, the learning curve of the ggplot2 framework pays itself off by orders of magnitude with respect to the time needed to implement these graphs.

15.6.1 How to make quick exploratory plots

We think that exploratory plots should be done **quickly** and with **minimal code**.

The base R plot is a great function. However, in some instances one can create many quick plots using ggplot2, which can be faster than using base plot. Consider the specific example of a binary y variable and multiple continous x variables. Let us say that we want to plot jittered points, a fit from a binomial glm (logistic regression), and one from using the loess smoother. We will use mtcars dataset and the binary outcome variable, y, is whether or not the car is American or foreign (am variable).

```
g = ggplot(aes(y = am), data = mtcars) +
  geom_point(position = position_jitter(height = 0.2)) +
  geom_smooth(method = "glm",
              method.args = list(family = "binomial"), se = FALSE) +
  geom_smooth(method = "loess", se = FALSE, col = "red")
```

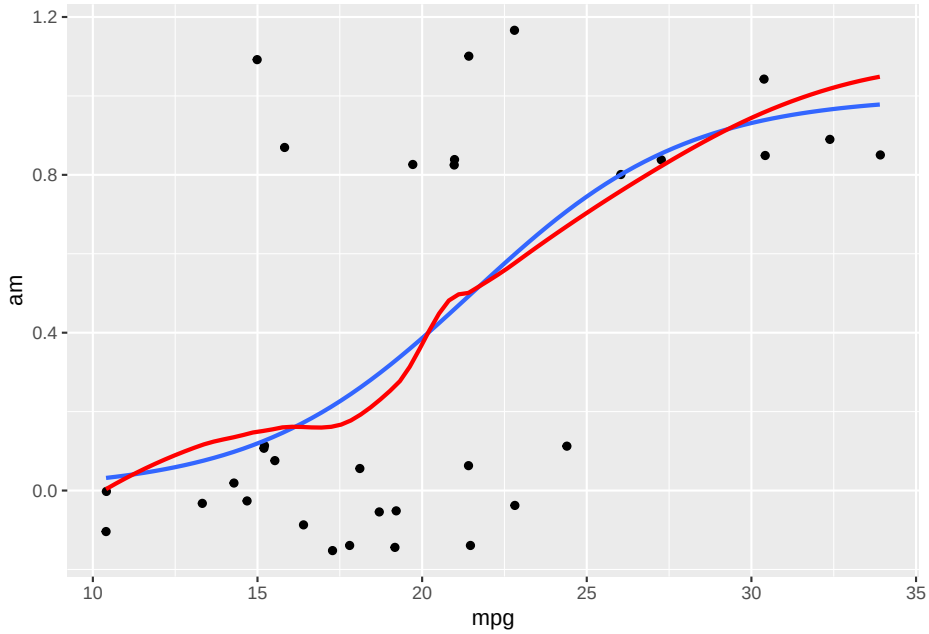


Figure 15.32: Miles per galon versus American (am=0) or foreign (am=1). Dots are slightly jittered around their 0/1 values. Blue line is the logistic regression and the red line is loess smoother.

Then we can simply add the `x` variables as aesthetics to look at the association between American/foreign (`am`) and miles per galon (`mpg`) in Figure 15.32, rear axle ratio (`drat`) in Figure 15.33, and number of seconds in the quarter mile (`qsec`) in Figure 15.34.

```
g + aes(x = mpg)
```

```
g + aes(x = drat)
```

```
g + aes(x = qsec)
```

We would like to emphasize the simplicity of working with using `ggplot2` object to create this sequence of plots. The same could be done in base `R`, but it would require to create a function. Here we do not take sides, just provide the options.

15.7 Problems

Problem 1. Use the `gapminder` dataset from the `dslabs` package. Perform the following operations on the data.

a. Filter the following countries: `"Vanuatu", "Zimbabwe", "Tanzania",`

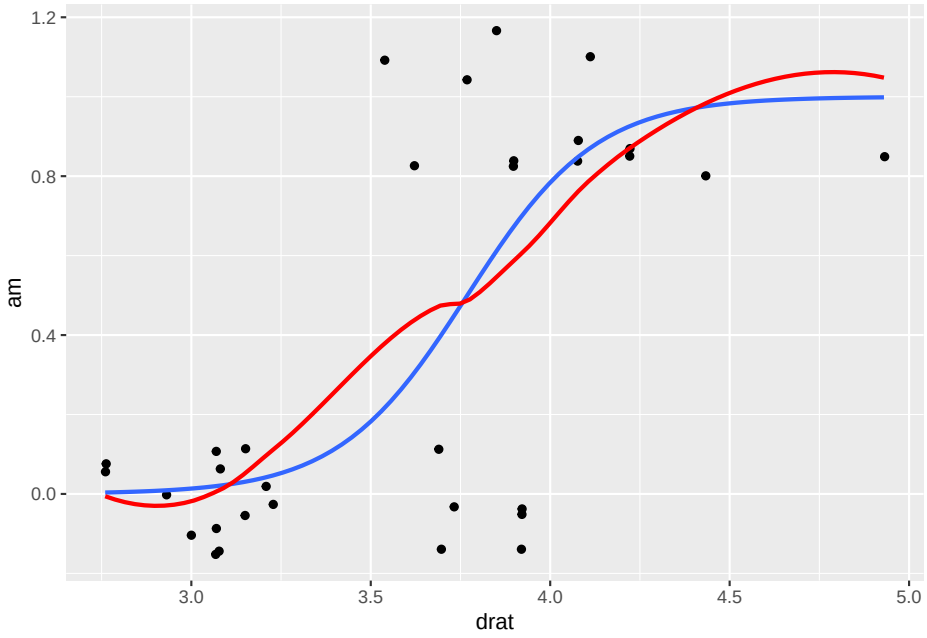


Figure 15.33: Rear axle ratio (drat) versus American (am=0) or foreign (am=1). Dots are slightly jittered around their 0/1 values. Blue line is the logistic regression and the red line is loess smoother.

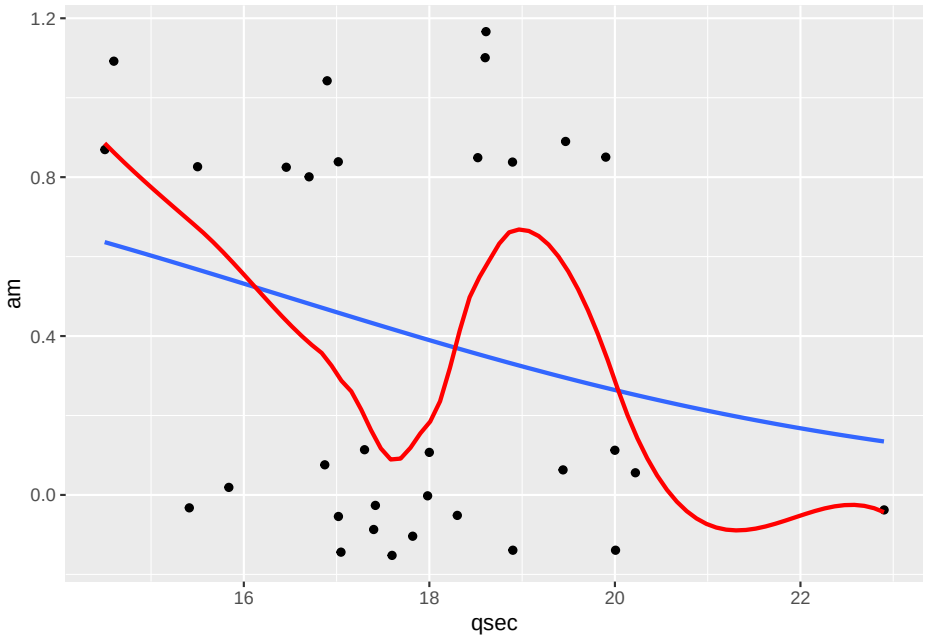


Figure 15.34: Number of seconds in quarter mile versus American (am=0) or foreign (am=1). Dots are slightly jittered around their 0/1 values. Blue line is the logistic regression and the red line is loess smoother.

`"Romania"` for the years 2000 to 2010.

b. Plot the `gdp` versus the `fertility` in a scatterplot and map the `population` to the size attribute for the year 2010. Use a different color for each country.
c. Plot the same information as above for all years, but different panels for the different years.

Problem 2. Use the `gapminder` dataset from the `dslabs` package. Perform the following operations on the data. Filter the following countries: `"Vanuatu", "Zimbabwe", "Tanzania", "Romania"` for the years 2000 to 2010.

a. Plot the `gdp` over time (over years) in a scatterplot and map the `population` to the size attribute with different colors for each country.
b. Add lines to the plot above, with the color to be black for each line.
c. Place the legend **inside** the plot and make the guide horizontal.
d. Remove the legend from the plot.
e. Add a smoother using the default in `geom_smooth` for all countries (you need to make this data yourself).

Problem 3. Use the `gapminder` dataset from the `dslabs` package. Perform the following operations on the data. Filter the years from 2000 to 2010.

a. Plot boxplots of `infant_mortality` over each year.
b. Plot separate boxplots by continent where each continent is a different color.
c. Plot separate boxplots by continent where each continent is in a different panel/facet.
d. Plot the facets with a free y-axis.
e. Instead of boxplots in the first plot, try using `geom_violin` for each year.

Problem 4. Use the `gapminder` dataset from the `dslabs` package. Perform the following operations on the data. Filter the years from 2000 to 2010.

a. Plot histograms of `fertility` over all countries, facetted by each separate year. Save this file as `fertility_hists.png` that is 4 by 4 inches with a resolution of 600ppi.
b. Plot a density of `fertility` over each year, where each year is a different color (using `geom_density`).
c. Plot the same as above using `geom_line(stat = "density")`.

Problem 5. Use the `gapminder` dataset from the `dslabs` package. Perform the following operations on the data. Filter the years from 2000 to 2010. Call this data set `df`.

a. Keep only data from 2010. Make a bar chart using `geom_col`, where the `x` axis is the continent and `y` axis is the fertility. Help:

```
g = df %>%
  filter(year == 2010 ) %>%
```

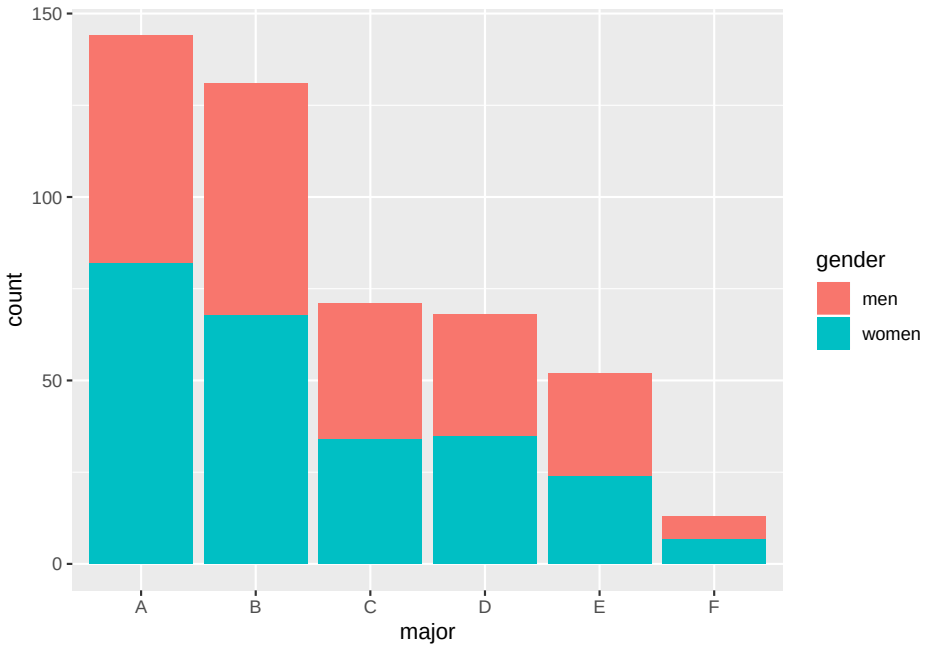
```
  ggplot(aes(x = XXX, y = XXX )) +
  geom_col()
```

b. Add `coord_flip` to the last plot to see how it changes. Is this better for viewing the labels? If so, why?

Problem 5. Use the `admissions` data from the `dslabs` package.

a. Make a stacked bar chart using `geom_bar` and map the `weight` aesthetic to the `admitted` variable, `gender` to the x-axis and fill each bar using the `major`:

```
admissions %>%
  ggplot(aes(x = major, weight = admitted, fill = gender)) +
  geom_bar()
```



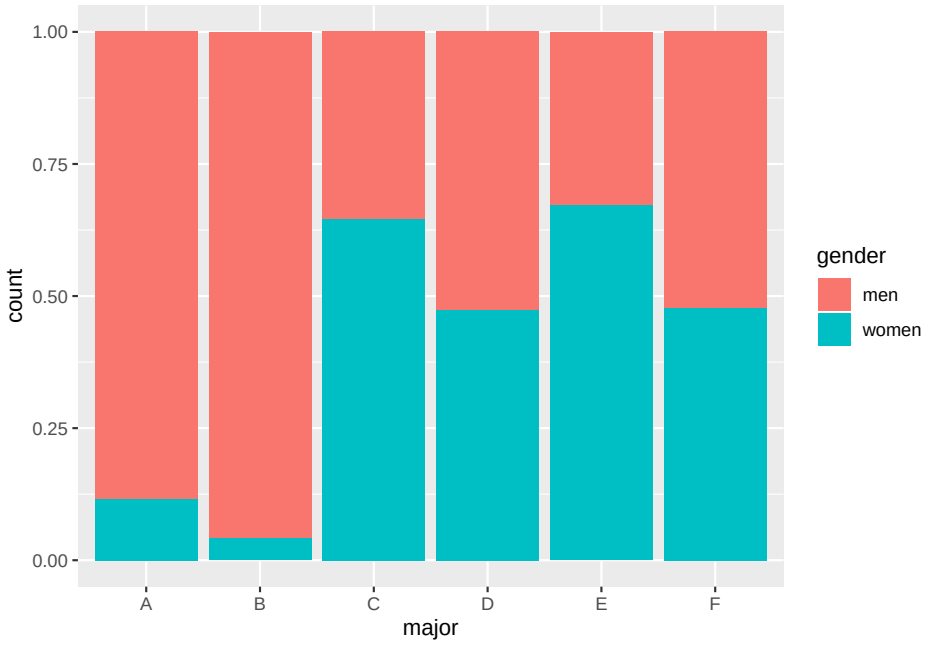
b. Perform the same operation, but weight the data by the number of applicants.

c. Run the same graph as in a., but switch `gender` and `major`.

d. Create a variable called `pct_app` using `mutate` and `group_by`, which calculates the percentage applications by each major. Make sure to `ungroup` the data. Create a stacked bar chart similar to c. using `pct_app` as the weight

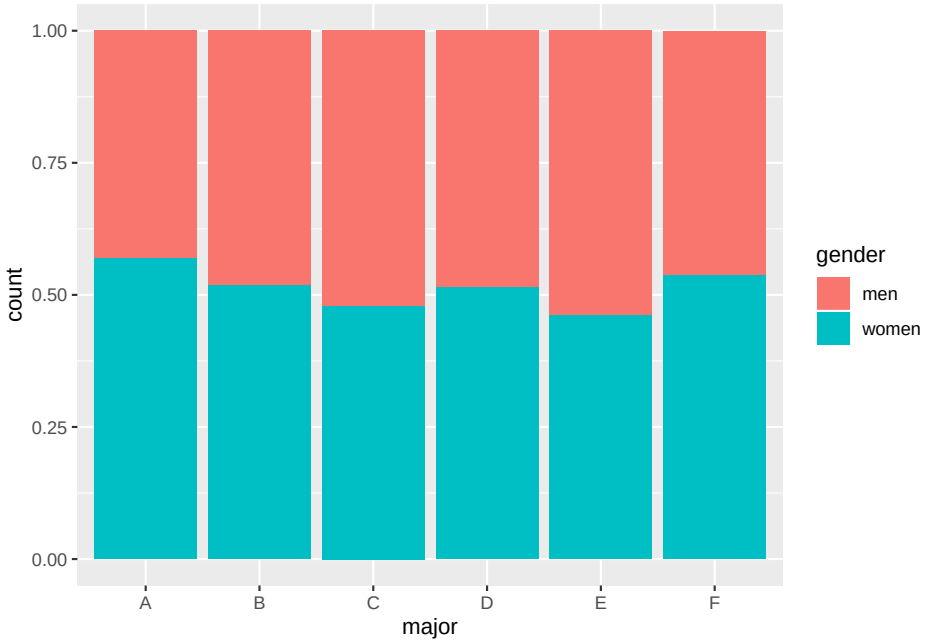
```
admissions %>%
  group_by(major) %>%
  mutate(pct_app = applicants / sum(applicants)) %>%
```

```
  ungroup() %>%
  ggplot(aes( x = major, weight = pct_app, fill = gender)) +
  geom_bar()
```



e. Look at the same for admitted (creating `pct_adm` for percent admitted):

```
admissions %>%
  group_by(major) %>%
  mutate(pct_adm = admitted / sum(admitted)) %>%
  ungroup() %>%
  ggplot(aes( x = major, weight = pct_adm, fill = gender)) +
  geom_bar()
```


1.00
0.75
0.50
0.25
0.00
count
A
B
C
D
E
F
major
gender
men
women

Chapter 16

Hypothesis testing

This chapter covers the following topics

- Introduction and general discussion
- General presentation of hypothesis tests
- Hypothesis test connections with confidence intervals
- P-values
- Discussion

16.1 Introduction

Hypothesis testing is one of the most popular and fundamental techniques in statistics. In hypothesis testing, one evaluates the evidence in favor of and against two hypotheses and then uses a procedure to control errors (see Neyman and Pearson 1933 for the origins of hypothesis testing). Consider the following example. Imagine seeing a funny shaped coin and deciding between whether it is fair, labeled hypothesis H_0 (i.e., hypothesizing that $p = 0.5$, where p is the probability of a head), versus biased in a particular way, labeled H_1, say $p = 0.75$. Suppose that we flip the coin and see three consecutive heads. The probability of three consecutive heads is $1/8$ under H_0 and $27/64$ under H_1. Thus, the relative support of H_1 to H_0 is $27/8 = 3.375$, indicating that the evidence supporting H_1 is 3.4 times the evidence supporting H_0. One could then plot the relative support of H_1 relative to H_0 for every value of H_1 from 0 to 1.

```
pSeq = seq(0, 1, length = 1000)
pH1 = pSeq ^ 3; pH0 = 1/8; relativeEvidence = pH1 / pH0
plot(pSeq, relativeEvidence, type = "l", frame = FALSE, lwd = 2,
     cex.axis=1.3,col.axis="blue",cex.lab=1.3,
     xlab = "H1 value of p", ylab = "P(3 heads|H1)/P(3 heads|H0)")
```

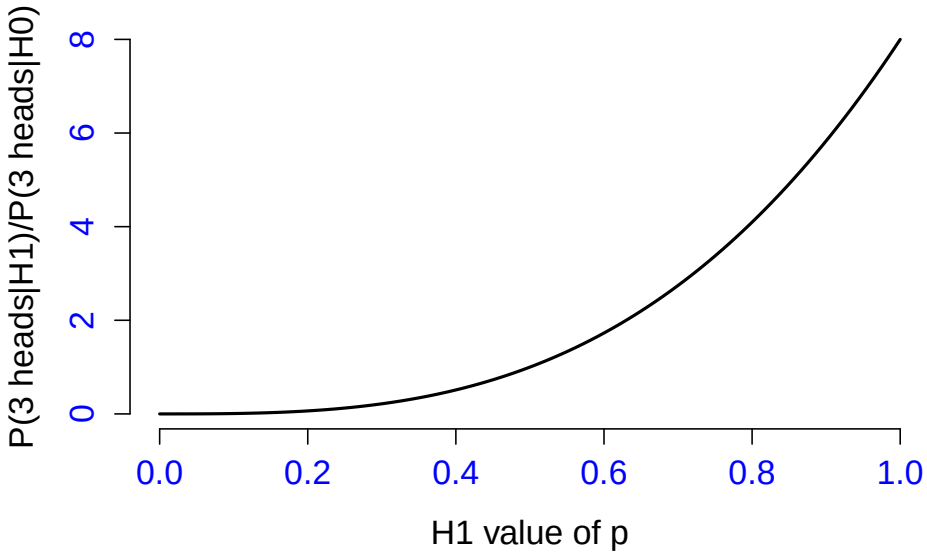


Figure 16.1: Ratio of probabilities for observing three heads under the alternative and null hypotheses (y-axis) as a function of the probability under the alternative hypothesis.

Figure 16.1 displays p_1^3/p_0^3, the ratio of probabilities for observing three heads under the alternative and null hypotheses, where $p_0 = 1/2$ and for every value of p_1 between 0 and 1. Results indicate that the best supported value of p is the MLE ($p = 1$), which is 8 times better supported than a fair coin. This strategy, while based on formal notions of statistical evidence, does not yield a strategy for deciding between any particular value of H_1 and H_0. Instead, it displays the relative evidence to H_0 for a collection of H_1 values. The prevailing method for subsequent decision making in this setting is to control probabilistic error rates. Purely likelihood based methods have error rates that are controlled, called "misleading evidence." However, these methods are not popular. Nor are Bayesian hypothesis testing procedures based on controlling expected posterior-based error rates.

Instead, we will describe the overwhelmingly most popular variation of hypothesis testing called null hypothesis significance testing (NHST). In this (frequetist) procedure one declares one of the hypotheses as a default or status quo hypothesis, usually labeled H_0 and referred to as the "null hypothesis." The alternative hypothesis is the research hypothesis, and is typically labeled H_a or (less often) H_1. For example, in a drug development trial, the null hypothesis would be that the drug has no therapeutic effect on disease symptoms, whereas the alternative hypothesis is that it has a positive effect. It is perhaps useful to think in terms of a criminal court case, an example that we will refer back to frequently. The

null hypothesis is that the defendant is innocent and evidence is required to convict.

In NHST, calculations are performed assuming that the null hypothesis is true, and evidence is required to reject the null hypothesis in favor of the alternative. Setting up the problem this way has several benefits, but also introduces several limitations; we will cover both.

When deciding between two hypotheses, there are two kinds of correct decisions that can be made and two kinds of errors. In NHST there are two types of correct decisions and two types of error:

- If H_0 is true and we conclude H_0, then we have correctly accepted the null
- If H_0 is true and we conclude H_a, then we have made a **Type I** error
- If H_a is true and we conclude H_a, then we have correctly rejected the null
- If H_a is true and we conclude H_0, then we have made a **Type II** error

Note that some specific language and concepts, which we will cover shortly, must be adhered to during the decision process. NHST controls the probabilities of the errors, the *error rates*, to make decisions. Specifically, the procedures ensure that the probability of a Type I error is small. Recall our court case, H_0 is that the defendant is innocent, H_a is that the defendant is guilty. A Type I error is convicting an innocent defendant. A Type II error is failing to convict a guilty defendant.

If we require a lot of evidence to convict defendants, then we will let a lot of guilty criminals go free (commit many Type II errors). In contrast, if we require very little evidence, then more innocents would be convicted (Type I errors). Most courts are set up to require a lot of evidence to convict, thus making the probability of Type I errors small. We adhere to a similar procedure in NHST; we ensure that the probability of Type I errors is small and thus require a lot of evidence to reject the null hypothesis.

The probabilities of the various error rates are so often spoken of that we have given most of them symbols or names as follows (in the same order as before):

- The probability of correctly accepting the null is called $1 - \alpha$.
- The probability of a Type I error is called α.
- The probability of correctly rejecting the null is called *Power* or $1 - \beta$.
- The probability of a Type II error is called β.

NHST forces α to be small; we want to avoid convicting innocent defendants. Thus, philosophically, we are stating a strong preference for the status quo as the default. If we meet this standard of evidence, then we say that we "reject the null hypothesis (in favor of the alternative)." That is, we say that there was enough evidence to find the defendant guilty. Because of the strong standard required, we can say that we "conclude H_a" (conclude that the defendant is actually guilty and lock him or her up).

On the other hand, if we do not meet the standard of evidence, we do not

conclude that the defendant is innocent, but rather that there was not enough evidence to convict. To see this, imagine a court that puts a near impossible standard of evidence for conviction and never convicts any defendants. Both innocent and guilty defendants will be cleared of charges. Therefore, being cleared of charges does not imply a great deal of evidence of innocence, rather only a failure to convict. This is summarized in the famous phrases "Absence of evidence is not evidence of absence" and "You can't prove a negative." Because of this, we tend to say that we "fail to reject" the null hypothesis, rather than saying we accept it. We emphasize that this is mostly a byproduct of how we implement NHST: performing calculations assuming the null hypothesis to be true and requiring a large amount of evidence to reject it.

How can one have faith that the null is actually true when failing to reject the null hypothesis? Ultimately, this comes down to having faith that the study was set up for success and would have rejected the null if an effect was truly present. Thus, invariably, when a study fails to reject H_0, discussion focuses on the main method for preventing Type II errors: study design. This includes the power of the study, its design, the sample size, the potential for biases, the sample being studied, and so on.

Another component of hypothesis testing is choosing a statistic that represents evidence. Often, this will be a likelihood ratio, though often not. The choice of statistic is important, as it can impact power. As a simple example, one could generate a uniform random variable and reject if it is less than 0.05. This procedure will always control the Type I error rate at 5%! However, it does not make any use of the data and will always have a power (the probability of rejecting if H_a if true) of 5%, as well. Of course, this is a terrible procedure, but it highlights the role that the statistic plays in designing an NHST. There is some theory that says the likelihood ratio is optimal as a statistic in certain settings (the Neyman/Pearson lemma). However, this theory is restrictive in that there are limited settings under which it applies. In general, we will have to evaluate our statistic using things like Monte Carlo studies or mathematics if we want to see how they perform over a variety of settings. In the cases we investigate in this chapter, a good statistic will be sort of obvious, so we will not discuss the choice of the test statistic very much.

As NHST is a deep topic, we will revisit the ideas frequently. However, for now, let us build up our intuition with some simple examples.

16.1.1 A simple example

Consider again our coin example. Suppose that we want to set up a rejection region for our problem. A standard in NHST is to control the error rate at 5%. Let us also set our null hypothesis to be that the coin is fair: $H_0 : p = 0.5$. Further, let us consider $H_a : p > 0.5$, comparing the hypothesis that the coin is fair to the hypothesis that it is biased towards heads. A reasonable statistic

results from simply counting the number of heads, with more heads being flipped pointing toward H_a. Thus, we need a value C so that the probability $X \geq C$ is smaller than α where X is the number of heads. That is, we want

$$P_{H_0}(\text{We reject } H_0) = P(X \geq C \mid p = 0.5) \leq \alpha \, .$$

Then we will reject the null hypothesis if we get greater than C heads. Clearly if a C satisfies this probability statement, then $C + 1$ does as well, as does $C + 2$ and so on. We then want to find the smallest such C that satisfies this statement. We can do this using R. Let us assume that the number of trials is $N = 10$.

```
out = round(
  rbind(0 : 10,
    100 * pbinom((0 : 10) - 1, size = 10, prob = 0.5, lower.tail = FALSE))
  )
rownames(out) = c("C", "P(X>=C) %")
out
```

```
          [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11]
C            0    1    2    3    4    5    6    7    8     9    10
P(X>=C) %  100  100   99   95   83   62   38   17    5     1     0
```

A small technical point on the code; R's upper tail probability calculation for discrete variables (like `pbinom`) gives the strict inequality $P(X > a)$. So, you have to supply `a-1` as an argument to include the probability of `a`. That is, `pbinom(4, size = 10, prob = 0.5, lower.tail = FALSE)` yields $P(X > 4) = P(X \geq 5)$. Note `pbinom(a, size = 10, prob = 0.5)` gives $P(X \leq a)$.

Thus, if we want an α of 5%, then we need to set $C = 8$. So, we reject the null hypothesis if we get 8 or more heads out of 10 flips. This procedure has an approximate probability of 5% under the null hypothesis. As a second technical note, the probability for $C = 8$ is actually over 5% (we rounded above). If you want strict control at 5% (that is, $\alpha < 0.05$), you would have to set $C = 9$. This is a byproduct of the discreteness of the binomial distribution, which does not allow to obtain an exact 5% level test. In continuous settings this will not be a problem.

We should also discuss whether the null is $H_0 : p = 0.5$ or $H_0 : p \leq 0.5$. It turns out, it does not matter. The hypothesis $H_0 : p = 0.5$ is the most extreme case towards the alternative and so if you set your test up for this case, it covers the more inclusive null $H_0 : p \leq 0.5$ as well. That is, $P(X \geq C \mid p = p_0)$ is the largest for $p_0 \leq 0.5$ at the boundary $p_0 = 0.5$ (check this yourself!). Thus, in most one sided testing settings, one typically specifies the boundary case. We will discuss one sided versus two sided tests more later.

16.1.2 A second simple example

Imagine studying a population of subjects for sleep apnea using the Respiratory Disturbance Index (RDI). An RDI of 30 is considered severe sleep apnea. This is one sleep breathing disturbance every two minutes on average for a night. (Hopefully, this drives home the importance of studying this disease, as you can imagine the negative health and well-being effects for being chronically deprived of oxygen that often while sleeping.)

You would like to know if a particular population has a mean RDI greater than 30. To do this, you sample 100 subjects from the population and measure their RDI. To make the problem easier for pedagogy, let us assume the standard deviation is known to be 10 and that RDI is normally distributed. A reasonable rejection strategy would be to reject if our sample mean is greater than some number, say C. We will pick C to ensure that our Type I error rate is α. Specifically, we want:

$$P_{H_0}(\text{We reject } H_0) = P(\bar{X} \geq C \mid \mu = 30) = \alpha$$

where μ is the population mean RDI. Under $H_0 : \mu = 30$ we know that $\bar{X}$ is normally distributed with a mean of 30 and a variance of $10/\sqrt{100} = 1$. Thus, C needs to be the upper α quantile of a $N(30, 1)$ distribution.

```
alpha = c(0.001, 0.01, 0.05, 0.1)
out = rbind(alpha, qnorm(alpha, 30, 1, lower.tail = FALSE))
rownames(out) = c("alpha", "C")
round(out, 3)
```

```
        [,1]   [,2]   [,3]   [,4]
alpha  0.001  0.010  0.050  0.100
C     33.090 32.326 31.645 31.282
```

So, for a 5% level test, we would need to see a sample average of 31.645 to reject.

16.1.3 Z scoring our test statistic

We can save ourselves a lot of trouble by taking the null Z-score of our test statistic. Specifically, note for all tests of the form $H_0 : \mu = \mu_0$ versus $H_a : \mu > \mu_0$ we get that the probability that we reject under the null hypothesis is:

$$P(\bar{X} \geq C \mid \mu = \mu_0) = \alpha = P\left(\frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}} \geq \frac{C - \mu_0}{\sigma/\sqrt{n}} \mid \mu = \mu_0\right) = P(Z \geq Z_{1-\alpha}) \,,$$

where Z is a $N(0, 1)$ random variable and $Z_{1-\alpha}$ is the $1 - \alpha$ quantile of the standard normal. Thus, $C = Z_{1-\alpha}\sigma/\sqrt{n} + \mu_0$; the rejection boundary is $Z_{1-\alpha}$ standard errors above the null mean. Equivalently, the equation above shows

that we could just standardize our test statistic. The resulting rule is that we reject if:

$$\frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}} \geq Z_{1-\alpha}.$$

You can derive for yourself the one sided case in the other direction. To simply present the result, if $H_0 : \mu = \mu_0$ versus $H_a : \mu \leq \mu_0$ we reject if:

$$\frac{\bar{X} - \mu_0}{\sigma/\sqrt{n}} \leq Z_{\alpha}.$$

These rules also apply if σ is estimated and one is applying the CLT, as well. However, of course, the Type I error only holds approximately in that case.

16.1.4 Two sided hypotheses

Often we are interested in whether the null is a specific value or is different from it. As an example, imagine if we had two groups of sleep disordered breathing patients, all with a starting RDI of 30, and we randomized half to a treatment, say a calorie restricted diet, while the other half serve as controls. A reasonable hypothesis is $H_0 : \mu_T - \mu_C = 0$ where μ_T is the mean for the treated and μ_C is the mean for the controls. Consider as a statistic, the difference in th eaverage RDI after six months of treatment: $\bar{X}_T - \bar{X}_C$. Given the randomization, it is reasonable to assume a constant variance, σ^2, so that the standard error of our statistic is

$$\text{Var}(\bar{X}_T - \bar{X}_c) = 4\sigma^2/n$$

where there are $n/2$ subjects in each group. Thus our Z-score standardized statistic is:

$$Z = \frac{\sqrt{n}(\bar{X}_T - \bar{X}_C)}{2\sigma}.$$

In general, if $H_0 : \mu_T - \mu_C = a$ and there are non-constant variances and different sample sizes, we would arrive at:

$$Z = \frac{\bar{X}_T - \bar{X}_C - a}{\sqrt{\frac{\sigma_T^2}{n_T} + \frac{\sigma_C^2}{n_C}}}.$$

We would like to reject if our Z statistic is either too large or too small. We will distribute our Type I error rate as αp in each tail where p is any number between 0 and 1:

$$P(Z \leq Z_{p\alpha} \text{ or } Z \geq Z_{1-(1-p)\alpha}) = p\alpha + (1-p)\alpha = \alpha.$$

Typically, we have no preference for requiring stricter evidence for either a too large or too small statistic, so we simply set $p = 0.5$, placing half of our Type I error rate in either tail.

Thus, we reject if our normalized statistic is smaller than $Z_{\alpha/2}$ or larger than $Z_{1-\alpha/2}$. But, since $Z_{1-\alpha/2} = -Z_{\alpha/2}$ we arrive at the rule, reject if $|Z|$ is larger than $Z_{1-\alpha/2}$.

We can arrive at this procedure from a different line of thinking. We could set our starting statistic as Z^2 and reject for large values, which would correspond to either large positive or small negative differences from the mean. However, if Z is the standard normal, then Z^2 is Chi-squared with 1 degree of freedom. Thus, to force a Type I error rate of α, we would reject if $Z^2 \geq \chi^2_{1-\alpha}$ where $\chi^2_{1-\alpha}$ is the upper α quantile of the Chi-squared distribution. Check for yourself that the square root of $\chi^2_{1-\alpha}$ is $Z_{1-\alpha/2}$ so that this leads to the same rule as above!

16.2 General hypothesis tests

Consider testing a mean parameter, μ via null hypothesis $H_0 : \mu = \mu_0$. Consider an estimator of μ, S, which will serve as your statistic. Assume that S has standard error $\mathrm{SD}(S)$, which is consistently estimated by $\widehat{\mathrm{SD}(S)}$ and that

$$Z = \frac{S - \mu_0}{\widehat{\mathrm{SD}(S)}}$$

either follows a normal distribution or limits to one via the central limit theorem. Then you reject under the following rules:

- $H_a : \mu > \mu_0$ reject if $Z \geq Z_{1-\alpha}$
- $H_a : \mu < \mu_0$ reject if $Z \leq Z_{\alpha}$
- $H_a : \mu \neq \mu_0$ reject if $|Z| \geq Z_{1-\alpha/2}$

Here are some examples of the test statistic and setting:

- Single mean: $Z = \sqrt{n}(\bar{X} - \mu_0)/S$
- Difference in two means: $Z = \frac{\bar{X}_1 - \bar{X}_2 - (\mu_1 - \mu_2)}{\sqrt{S_1^2/n_1 + S_2^2/n_2}}$
- Binomial proportion: $Z = \sqrt{n}(\hat{p} - p_0)/\sqrt{\hat{p}(1-\hat{p})}$

For data that can reasonably be assumed to be normal, the standardized statistic will follow a Student's t distribution, and one simply follows the same rules but with the normal quantiles replaced by t quantiles.

16.3 Connection with confidence intervals

There is a direct connection between confidence intervals and hypothesis tests. Even better, the connection works exactly as one would hope. You can perform a hypothesis test by simply rejecting if the null parameter value lies outside of the interval, and you can construct a confidence interval by taking the values of the null parameter that you fail to reject. We will elaborate.

Consider a $1-\alpha$ confidence interval, $[C_L, C_U]$, for a parameter, μ. (It does not have to be a mean parameter.) The fact that the interval is a confidence interval requires that

$$P\left(\mu \in [C_L, C_U]\right) = 1 - \alpha \ .$$

Here note that it is the interval that is random, not μ. So this is the probability that the random interval encompasses the true value, μ. Therefore, the probability that it does not contain μ is α. Then, given a null hypothesis value of μ, say μ_0, define a test statistic as the indicator of the interval not containing μ_0:

$$TS = I(\mu_0 \notin [C_L, C_U]) \ .$$

Then the probability that we reject assuming that $\mu = \mu_0$ is $P(TS = 1 \mid \mu = \mu_0) = \alpha$. We have thus constructed a test that clearly has the desired Type I error rate. This doesn't necessarily mean it is a good test; it may have low power. In this case, low power results if the confidence interval is unnecessarily wide. This can especially happen in discrete data settings or cases where the confidence interval is constructed asymptotically.

What about the reverse direction? Given a test, can we construct a confidence interval with the values of the parameter for which we fail to reject? Absolutely. Let $TS(\mu_0)$ be a test statistic, where we write it here as a function of μ_0, the null parameter value. For simplicity's sake, let us assume our test statistic follows some distribution under the null, $F(\mu_0)$, and we reject if our test statistic is larger than its upper quantile, say $Q_{1-\alpha}$. Therefore, consider the points for which we fail to reject:

$$CI = \{\mu_0 \mid TS(\mu_0) \leq Q_{1-\alpha}\} \ .$$

Therefore, $P(TS(\mu_0) \leq Q_{1-\alpha} \mid \mu = \mu_0) = 1 - \alpha$ for every point in the CI. In particular, this holds for the actual value of μ, whatever it is. Let us say that μ_1 is the actual value of μ to help with our notation. Note that $\mu_1 \in CI$ if and only if $TS(\mu_1) \leq Q_{1-\alpha}$. Thus,

$$P(\mu_1 \in CI \mid \mu = \mu_1) = P\{TS(\mu_1) \leq Q_{1-\alpha} \mid \mu = \mu_1\} = 1 - \alpha \ .$$

A couple of points are in order. Technically, this is a confidence set, since there is no guarantee that it will be an interval. (More on this in a moment.) Secondly, the test statistic has to actually depend on the null hypothesized values. For example, if we create a useless hypothesis test that rejects if a uniform random

variable is less than α (which results in an α level test), then we would have an infinitely large confidence interval. This interval has coverage larger than $1-\alpha$ (it is 1), but it is also useless.

If the test statistic is a Z-score statistic, then inverting a hypothesis test will always result in a confidence interval. For example, we fail to reject if

$$Z^2 = \left\{ \frac{S - \mu_0}{\widehat{\mathrm{SD}(S)}} \right\}^2$$

is smaller than a Chi-squared cutoff. Therefore, we reject if $Z^2 < \chi^2_{1,1-\alpha} = Z^2_{1-\alpha/2}$ or, equivalently, fail to reject if $Z \in [-Z_{1-\alpha/2}, Z_{1-\alpha/2}]$. This is equivalent to saying that the set of values of μ_0 for which we fail to reject is:

$$\mu_0 \in S \pm Z_{1-\alpha/2}\widehat{\mathrm{SD}(S)} \ .$$

This is how we arrive at the general form for 95% confidence intervals of

$$\text{Estimate} \pm 2 \times \text{Standard Error of Estimate} \ ,$$

where the 2 is an approximation of $Z_{.975} = 1.96$.

16.4 Data example

First, let us read in the data.

```
dat = read.table(file = "data/shhs1.txt",
                 header = TRUE, na.strings=".")
```

Now, let us test whether or not different genders have different average RDIs. We will use the CLT and a mean test to do it. Let us do it manually first. It is always a good idea to specify your hypotheses first. So, our hypotheses are $H_0 : \mu_{Men} - \mu_{Women} = 0$ versus $H_a : \mu_{Men} - \mu_{Women} \neq 0$. Let's get the summary information that we need.

```
library(tidyverse)
smry = dat %>%
  group_by(gender) %>%
  summarise(mean = mean(rdi4p), sd = sd(rdi4p), n = n())
smry
```

```
# A tibble: 2 x 4
  gender  mean    sd     n
   <int> <dbl> <dbl> <int>
1      0  6.15  10.2  3039
2      1 11.4   14.0  2765
```

The SHHS is a combination of studies and the variable gender was obtained from the parent study; there is no information to distinguish gender and sex. For pedagogical purposes, we will ignore the issue, but a more careful study of the precise questions asked by the parent studies would need to be performed to thoroughly investigate sex or gender differences in RDI.

The average for men is higher, but is it statistically significantly higher? We will use the statistic:

$$\frac{\bar{X}_{men} - \bar{X}_{women}}{\sqrt{\frac{s^2_{men}}{n_{men}} + \frac{s^2_{women}}{n_{women}}}}.$$

```
estimate = smry$mean[2] - smry$mean[1]
mu0 = 0
se = sqrt(smry$sd[2]^2 / smry$n[2] + smry$sd[1]^2 / smry$n[1])
ts = (estimate - mu0) / se
ts
```

```
[1] 16.20031
```

Our test statistic of 16.2003059 is much larger than the 0.975 quantile of the normal distribution (1.96), so, we would reject the null hypothesis. A confidence interval would be given by

```
round(
  estimate + c(-1, 1) * qnorm(.975) * se,
  2
)
```

```
[1] 4.62 5.89
```

Of course, these calculations are kind of tedious. We can just use the function `t.test`. Since there are so many subjects, whether it is a T or a Z test is irrelevant. To agree with our standard error calculation, we will set the variances to be unequal.

```
t.test(dat$rdi4p ~ dat$gender, var.equal = FALSE)
```

```
    Welch Two Sample t-test

data:  dat$rdi4p by dat$gender
t = -16.2, df = 5007.2, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -5.886221 -4.615395
sample estimates:
mean in group 0 mean in group 1
       6.154055       11.404863
```

Notice that the results agree. The T statistic is 16.2 and the confidence interval is the same (except that they are subtracted in the reverse order). Let us consider a small sample example. The `sleep` data included with `R` are from the original manuscript on the T test. Oddly, the `t.test` documentation treats the data as if the groups are independent, yet they are in fact paired.

```
levels(sleep$group) = c("Dextro", "Laevo")
sleep2 = sleep %>% spread(group, extra) %>% mutate(Diff = Dextro - Laevo)
sleep2
```

```
   ID Dextro Laevo Diff
1   1    0.7   1.9 -1.2
2   2   -1.6   0.8 -2.4
3   3   -0.2   1.1 -1.3
4   4   -1.2   0.1 -1.3
5   5   -0.1  -0.1  0.0
6   6    3.4   4.4 -1.0
7   7    3.7   5.5 -1.8
8   8    0.8   1.6 -0.8
9   9    0.0   4.6 -4.6
10 10    2.0   3.4 -1.4
```

Here the study was of two sleep medications (Dextro and Laevo) and the outcome is the extra hours of sleep on the medication. Each patient received both drugs. Let us do a T-test on the difference. Thus, we are testing $H_0 : \mu_D = 0$ versus $H_a : \mu_D \neq 0$ where μ_D is the population mean difference in extra hours slept. We will do the T-test calculations manually first, then use `t.test`.

```
mn = mean(sleep2$Diff); sd = sd(sleep2$Diff); n = 10
ts = abs(sqrt(n) * mn / sd)
ts
```

```
[1] 4.062128
```

Thus we get a test statistic of 4.0621277. We take the absolute value since we are doing a two sided test. Compare this to the upper 0.975 quantile from a Student's T distribution with 9 degrees of freedom

```
qt(0.975, 9)
```

```
[1] 2.262157
```

Since our test statistic lies in the rejection region of being larger than 2.2621572 or smaller than -2.2621572 we reject the null hypothesis. Let us also calculate a confidence interval:

```
round(mn + qt(c(0.025, 0.975), 9) * sd / sqrt(n), 3)
```

```
[1] -2.46 -0.70
```

Of course, there are automated methods for doing this calculation in R.

```
t.test(sleep2$Diff)
```

```
    One Sample t-test

data:  sleep2$Diff
t = -4.0621, df = 9, p-value = 0.002833
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -2.4598858 -0.7001142
sample estimates:
mean of x
    -1.58
```

In fact, it is not necessary to even perform the subtraction:

```
t.test(sleep2$Dextro, sleep2$Laevo, paired = TRUE)
```

```
    Paired t-test

data:  sleep2$Dextro and sleep2$Laevo
t = -4.0621, df = 9, p-value = 0.002833
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.4598858 -0.7001142
sample estimates:
mean of the differences
                  -1.58
```

16.5 P-values

In the previous example, we rejected for $\alpha = 0.05$ for a two sided test, but what about $\alpha = 0.01$ or $\alpha = 0.001$? We could report the smallest Type I error rate for which we fail to reject, a number called the *attained significance level*. In our previous example, the attained significance level is reported to be 0.002833. The benefit of reporting this number is that the readers can then perform the test at whatever α level they want. If your attained significance level is smaller than α then you reject.

The attained significance level is mathematically equivalent to the so-called *P-value*. Popuralized by R.A. Fisher (Fisher 1925), the P-value is perhaps the most famous statistic ever created. The P-value is defined as **the probability under the null hypothesis of obtaining a test statistic the same as the observed results or more extreme in favor of the alternative.**

Assuming that our statistical model is correct, if a P-value is small, then either the null is false or we observed an unlikely extreme test statistic in favor of the alternative. It is relatively easy to see that the P-value is equivalent to the attained significance level. For ease of discussion, assume a one sided test and that smaller values of the test statistic favor the alternative hypothesis. Then, the P-value is defined as

$$P_{H_0}(TS \leq TS_{Obs}) \, ,$$

where TS_{obs} is the observed value of the test statistic. In contrast, we reject if our TS is less than or equal to Q_α, where Q_α is the α quantile of the null distribution of our test statistic. Clearly, the smallest value of α, hence the smallest value of Q_α, for which we reject is when $Q_\alpha = TS_{obs}$ and the associated α is the P-value. Because of this, practitioners, and many textbooks, do not differentiate between the P-value and the attained significance level.

Let us calculate a P-value for a simple example. We hypothesize that a coin is slightly biased towards tails, $H_0 : p = 0.5$ versus $H_a : p < 0.5$. Let X count the number of heads in 10 coin flips. Suppose we got 2 heads. If the null is true, the probability of getting 2 or fewer heads in 10 coin flips is

```
pbinom(2, size = 10, prob = 0.5)
```

```
[1] 0.0546875
```

Thus we would (barely) fail to reject at a Type I error rate at 5%, but would reject at a Type I error rate of 10%.

In our reanalysis of Student's data, we obtained a T-test statistic of 4.0621277. Because our test is two sided, we would reject if our test statistic is too large or small. The easiest way to do this is to double the smaller of the two one sided P-values.

```
2 * min(pt(ts, df = 9), pt(-ts, df = 9))
```

```
[1] 0.00283289
```

Of course, we typically do this calculation using software.

```
t.test(sleep2$Dextro, sleep2$Laevo, paired = TRUE)
```

```

    Paired t-test

data:  sleep2$Dextro and sleep2$Laevo
t = -4.0621, df = 9, p-value = 0.002833
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -2.4598858 -0.7001142
sample estimates:
mean of the differences
```

```
-1.58
```

16.6 Discussion

16.6.1 Practical versus statistical significance

Now that we have introduced NHST testing, it is worth discussing some of the controversy surrounding this technique. First, as you might have noticed, our calculations are made much simpler via a very constrained null hypothesis. The highly constrained nature of the null has drawn criticism in that one can reject the null hypothesis for trivially small effects if the sample size is large enough. This is often emphasized with the phrase "statistical significance is not practical significance." A simple fix is to have confidence intervals accompany tests, therefore the size of the effect will also be shown along with the result of the significance test.

However, it is also worth emphasizing that the importance of the size of the effect is a nuanced topic (Lindquist, Caffo, and Crainiceanu 2013, @confidence-precision). Small effects in large observational studies can be important, since often many covariates are included in a model specifically to challenge the effect and see if it survives the onslaught. Therefore, it is difficult to say anything about the size of an effect without considering the context including the study design and conduct of the analysis.

Another point worth discussing for two group testing is the possibility that the natural null is that the mean for the two groups is different. As an example, consider the problem of bioequivalence. When testing a new generic drug, it must be shown that the the observed blood concentrations are equivalent with those observed for the brand name. In this case, the natural null hypothesis is to assume that the mean concentration is different and require evidence to establish equivalence. We will not investigate this sort of testing further and point interested readers to the literature on "equivalence testing" for further reading.

Moreover, we disagree that statistical and scientific significance cannot be reconciled. On the contrary, suppose that we look again at the test for RDI in a treatment and control group. However, instead of worrying about whether or not treatment and control are indistinguishable, we would like to test whether the treatment reduces on average the number of events by 1 per hour, which is viewed as a "scientifically significant improvement". Then the null hypothesis $H_0 : \mu_T - \mu_C = 0$ where μ_T is the mean for the treated and μ_C is the mean for the controls would change to

$$H_0 : \mu_T - \mu_C \geq -1 \, .$$

Thus, scientific significance can be directly translated into statistical significance by changing the null hypothesis accordingly. Note that, in general, establishing scientific significance as described above requires more evidence than statistical significance when testing for equality.

16.6.2 Fishing for significance

A potential problem with significance tests is that one can simply keep performing them until significance is obtained. Such techniques are often called fishing expeditions, data dredging, P-hacking, multiple testing and multiple comparisons (see Benjamini and Hochberg 1995 for a modern discussion). The phenomena can occur explicitly from unscrupulous analyses, or accidentally through unaccounted for multiple comparison problems. Multiple comparison issues can be adjusted using Bonferroni or False Discovery Rate (FDR) approaches Benjamini and Hochberg (1995).

An obvious but unscrupulous use of testing is to perform many hypothesis tests on a data set, find a significant one, paint a credible story around the significance and present the results as if this test was the only one considered. Accidental cases can occur when a researcher considers multiple models or has legitimate confusion over what collection of tests should be grouped together and adjusted for multiple comparisons.

Recently, several efforts have been made to combat these and other related problems seen as a source of lack of scientific reproducibility. For example, the idea of pre-registering studies has gained momentum. In a pre-registered study, one lays out the analysis plan prior to collecting the data or performing the analysis. Such pre-registration already occurs in clinical trials, but would also help in observational research to increase transparency for analyses.

However, all of these problems are problems of usage, understanding, and implementation and not fundamental problems with the statistical testing framework and the associated P-values. Therefore, in practice one should be very careful about addressing the problem (e.g. unscrupulously fishing for significance) without throwing away a true, tried, and understood scientific technique.

16.6.3 Evidence and hypothesis tests

Consider the issue of a barely significant test. What impact does the size of the study have on interpreting that result? For a large study, is the fact that it is barely significant problematic, since one would hope for strong significance for a large study? Alternatively, for a small study, are we content that significance was obtained or concerned over the lack of power of the study and the possibility that this was a Type I error?

This latter point has received a great deal of attention of late. Specifically, in a Bayesian context one can discuss the probability of the null hypothesis (Katherine et al. 2013b):

$$P(H_a \mid \text{Significance}) ,$$

which decreases as the sample size decreases. Indeed, using the Bayes rule this probability is equal to

$$\frac{P(\text{Significance} \mid H_a)P(H_a)}{P(H_a)P(\text{Significance} \mid H_a) + P(H_0)P(\text{Significance} \mid H_0)} = \frac{\text{Power} \times P(H_a)}{\text{Power} \times P(H_a) + \alpha \times P(H_0)} .$$

Therefore, holding the other terms constant, as power goes up so does $P(H_a \mid \text{Significance})$ and vice versa. Therefore, the evidence associated with a positive test result has less force if the study was designed with low power.

16.7 Problems

Problem 1. Forced expiratory volume FEV is a standard measure of pulmonary function. We would expect that any reasonable measure of pulmonary function would reflect the fact that a person's pulmonary function declines with age after age 20. Suppose we test this hypothesis by looking at 10 nonsmoking males, ages 35-39, heights 68-72 inches, and measure their FEV initially and then once again two years later. We obtain these data (expressed in liters)

Person	Year 0	Year 2	Person	Year 0	Year 2
1	3.22	2.95	6	3.25	3.20
2	4.06	3.75	7	4.20	3.90
3	3.85	4.00	8	3.05	2.76
4	3.50	3.42	9	2.86	2.75
5	2.80	2.77	10	3.50	3.32

a. Perform and interpret the relevant test. Give the appropriate null and alternative hypotheses. Interpret your results, state your assumptions, and give a P-value.
b. We are interested in calculating the sample size for detecting a change in FEV over two years at least as large as that detected for males, ages 35-39. Use the data above for any relevant constants that you might need.

Problem 2. Another aspect of the preceding study involves looking at the effect of smoking on baseline pulmonary function and on change in pulmonary function over time. We must be careful since FEV depends on many factors, particularly age and height. Suppose we have a comparable group of 15 men in the same age and height group who are smokers and we measure their FEV at

year 0. The data are given (For purposes of this exercise assume equal variance where appropriate).

Person	Year 0	Year 2	Person	Year 0	Year 2
1	2.85	2.88	9	2.76	3.02
2	3.32	3.40	10	3.00	3.08
3	3.01	3.02	11	3.26	3.00
4	2.95	2.84	12	2.84	3.40
5	2.78	2.75	13	2.50	2.59
6	2.86	3.20	14	3.59	3.29
7	2.78	2.96	15	3.30	3.32
8	2.90	2.74			

Test the hypothesis that the change in FEV is equivalent between non-smokers and smokers. State relevant assumptions and interpret your result. Give the relevant P-value.

Problem 3. Perform the following simulation. Randomly simulate 1000 sample means of size 16 from a normal distribution with mean 5 and variance 1. Calculate 1000 test statistics for a test of $H_0 : \mu = 5$ versus $H_a : \mu < 5$. Using these test statistics calculate 1000 P-values for this test. Plot a histogram of the P-values. This exercise demonstrates the interesting fact that the distribution of P-values is uniform if the null is true for all tests.

Problem 4. In the SHHS, test whether men and women have the same average `rdi4p`. For every sample size $n = 5, 10, 25, 50, 100$ sample at random n men and n women and conduct the relevant test. Repeat this 10000 times and plot the histogram of P-values for every n. Compare the histograms among themselves and with the theoretical probability density function of a uniform random variable.

Problem 5. Suppose that systolic blood pressure (SBP) was recorded for 16 oral contraceptive (OC) users and 16 controls at baseline and again two years later. The average difference from follow-up SBP to the baseline (followup - baseline) was 11 $mmHg$ for oral contraceptive users and 4 $mmHg$ for controls. The corresponding standard deviations of the differences was 20 $mmHg$ for OC users and 28 $mmHg$ for controls. Does the change in SBP over the two-year period appear to differ between oral contraceptive users and controls? Perform the relevant hypothesis test and interpret. Give a P-value. Assume normality and a common variance.

Problem 6. Will a Student's T or Z hypothesis test for a mean with the data recorded in pounds always agree with the same test conducted on the same data recorded in kilograms? (explain)

Problem 7. A researcher consulting you is very concerned about falsely rejecting her null hypothesis. As a result the researcher decides to increase the

sample size of her study. Would you have anything to say? (explain)

Problem 8. Researchers studying brain volume found that in a random sample of 16 65-year-old subjects with Alzheimer's disease, the average loss in gray matter volume as a person aged four years was 0.1 mm^3 with a standard deviation of 0.04 mm^3. Calculate and interpret a P-value for the hypothesis that there is no loss in gray matter volumes as people age. Show your work.

Problem 9. A recent *Daily Planet* article reported on a study of a two-week weight loss program. The study reported a 95% confidence interval for weight loss from baseline of [2 lbs, 6 lbs]. There was no control group, all subjects were on the weight loss program. The exact sample size was not given, though it was known to be over 200. What can be said of an $\alpha = 5\%$ hypothesis test of whether or not there was any weight change from baseline? Can you determine the result of an $\alpha = 10\%$ test without any additional information? Explain your answer.

Problem 10. Suppose that 18 obese subjects were randomized, 9 each, to a new diet pill and a placebo. Subjects' body mass indices (BMIs) were measured at a baseline and again after having received the treatment or placebo for four weeks. The average difference from follow-up to baseline (followup−baseline) was -3 kg/m^2 for the treated group and 1 kg/m^2 for the placebo group. The corresponding standard deviations of the differences was 1.5 kg/m^2 for the treatment group and 1.8 kg/m^2 for the placebo group. Does the change in BMI over the two-year period appear to differ between the treated and placebo groups? Perform the relevant test and interpret. Give a P-value. Assume normality and a common variance.

Problem 11. In a study of aquaporins, 120 frog eggs were randomized, 60 to receive a protein treatment and 60 controls. If the treatment of the protein was effective, the frog eggs would implode. The resulting data were

	Imploded	Did not	Total
Treated	50	10	60
Control	20	40	60
Total	70	50	120

State the appropriate hypotheses and report and interpret the P-value.

Problem 12. Two drugs, A and B, are being investigated in a randomized trial, with the data given below. Investigators would like to know if the Drug A has a greater probability of side effects than drug B.

	None	Side effects	N
Drug A	10	30	40
Drug B	30	10	40

State the relevant null and alternative hypotheses and perform the relevant test.

Problem 13. You are flipping a coin and would like to test if it is fair. You flip it 10 times and get 8 heads. Specify the relevant hypotheses and report and interpret an exact P-value.

Problem 14. In the SHHS we want to test whether the mean of `rdi4p` is 5. Write down the hypotheses and explain the relevant quantities. Suppose that you sample at random $n = 5, 10, 25, 50, 100$ subjects from SHHS and calculate the relevant P-values. Repeat the process 10000 times and plot the histograms of P-values for every n. What do you conclude?

Problem 15. The same problem as before, except that we test in subsamples whether the mean is equal to the observed mean `rdi4p` in the entire sample. How many times do you reject the null hypothesis? What do you conclude?

Chapter 17

R Programming in the Tidyverse

This chapter covers the following topics

- Data objects
- `dplyr`
- Grouping data
- Merging datasets
- Reshaping datasets
- Cleaning strings: the `stringr` package

The `tidyverse` is a collection of packages that work on a tidy data principle. It contains functions for manipulating datasets, such as `dplyr` and `tidyr`, transforming objects into datasets, such as `broom`, and plotting datasets, such as `ggplot2`. The `tidyverse` package itself is essentially a wrapper that loads this entire collection of packages. Here we can load the `tidyverse` package:

```
library(tidyverse)
```

As many of the `tidyverse` packages are created or maintained by many of the employees of RStudio, they provide many great cheatsheets for new users.

17.1 Data objects in the `tidyverse`: `tibbles`

We will mostly use `tibbles`, as the `tibble` package is included in the `tidyverse`. Another reason we will be using `tibbles` is that the `readr` package we will use for reading data outputs `tibbles` from its reading functions (Wickham, Hester, and Francois 2017). Some of the `dplyr` functions will return a `tibble` even if a

`data.frame` is passed in, thus most work will be on `tibble` objects. We must first read in the data.

17.1.1 `readr`: reading in data

The `readr` in the `tidyverse` package can handle many different types of data input formats. `R` inherently can read most of these formats, but `readr` provides better defaults (for most users) and can be significantly faster than the standard counterparts in the base `R` packages.

```
file.name = file.path("data", "shhs1.txt")
df = read_tsv(file.name)
class(df)
```

```
[1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

Here data are tab-delimited, so we use the `read_tsv` function and see the output is indeed a `tibble`. The analogous `read_csv` function is very useful for spreadsheets that are comma separated. Now that we have the data read into `R`, we will show how to manipulate it. Whenever reading in any dataset using `readr`, it is good to check that there were no parsing problems. The `problems` function allows you to see any of the problems, and the `stop_for_problems` function will throw an error if there are any problems in reading your data.

```
stop_for_problems(df)
```

17.1.2 Other data formats

The `readxl` package can read in `xlsx/xls` files directly into `R` using the `read_excel` function. There are packages that write out `xlsx` files directly from `R`, but we do not condone that workflow directly. Mostly, using CSV file outputs after manipulation in `R` is sufficient to get into Excel or other pieces of software for formatting tables and can be version-controlled much easier than `xlsx` files.

The `haven` package (Wickham and Miller 2017) provides the ability to read data formats from other statistical software such as SAS, SPSS, and Stata. This provides the ability to go directly from other software formats to `R` without going through an intermediary format such as CSV. This direct reading is useful because many times you are receiving these formats from a collaborator or online source and would like a reproducible way to perform an analysis. Creating an intermediate CSV is not as reproducible. Moreover, converting to an intermediate CSV file may lose pieces of information about the data, such as labels.

17.2 dplyr: pliers for manipulating data

The dplyr package has a multitude of functions, and we will cover the most fundamental ones here. These can be considered as the dplyr "verbs." If you think of R as a language with syntax, the dplyr dialect thinks of datasets as "nouns" and the actions done to the data as the "verbs." The dataset "nouns" we will be using will be things like data.frames, mainly tibbles.

Every function in dplyr that takes in a dataset has that dataset as the **first** argument, which is why the dplyr verbs work seamlessly with the pipe operator (%>%).

17.2.1 Selecting columns from a dataset

The first verb we will introduce is the select function, which subsets the columns of the data. We work with the SHHS dataset and we would like to extract only the patient identifier (pptid), age, BMI, and gender of the patient:

```
df = df %>% select(pptid, age_s1, bmi_s1, gender)
```

```
# A tibble: 3 x 4
  pptid age_s1 bmi_s1 gender
  <dbl>  <dbl>  <dbl>  <dbl>
1     1     55   21.8      1
2     2     78   33.0      1
3     3     77   24.1      0
```

Note that dplyr verbs, such as select, understand that you are referring to columns of a dataset when passing in arguments, so you do not have to use other operators such as the dollar sign ($) or the quote for the column names. There are several helper functions for select, which can be found in the help file (running the ?select_helpers command). For example, starts_with/ends_with functions allow you to select columns which start/end with a string. You can also use contains/matches to select columns based on a literal string/regular expression. Lastly, you can reorder the columns of a dataset by using the everything helper, such as:

```
df %>% select(col1, col2, everything())
```

so that the first two columns of the data are now col1 and col2. You can also remove columns using select by using the hyphen or negative sign. For example the following code will drop column pptid, but keep all other columns:

```
df %>% select(-pptid)
```

```
# A tibble: 3 x 3
  age_s1 bmi_s1 gender
   <dbl>  <dbl>  <dbl>
```

```
1     55   21.8      1
2     78   33.0      1
3     77   24.1      0
```

In every example above, nothing actually happened to the datasets as we did not **reassign** the output of these commands to `df`; the results were simply printed out. If we wanted to create another dataset or overwrite `df`, we would run something like:

```
df = df %>% select(-pptid)
```

This last operation was not actually run and `df` continues to contain `pptid` for the remainder of the chapter.

17.2.2 Renaming columns of a dataset

The next verb we will introduce is the `rename` function, which renames the columns of the data. Here, we wish to rename the `age_s1` and `bmi_s1` columns to `age` and `bmi`, respectively.

```
df = df %>% rename(age = age_s1, bmi = bmi_s1)
```

```
# A tibble: 3 x 4
  pptid   age   bmi gender
  <dbl> <dbl> <dbl>  <dbl>
1     1    55  21.8      1
2     2    78  33.0      1
3     3    77  24.1      0
```

Now we see the column names have been changed.

17.2.3 Subsetting rows of a dataset

Many times, we would like to subset our data based on a logical condition. For example, let us create a dataset named `under` that has only the individuals which have a BMI of less than 18.5. We would use the `filter` command. The `filter` command takes in a logical condition and returns only the rows where that condition is `TRUE`. If the logical condition returns missing values (such as `NA`), these rows will **not** be returned.

```
under = df %>% filter(bmi < 18.5)
```

```
# A tibble: 3 x 4
  pptid   age   bmi gender
  <dbl> <dbl> <dbl>  <dbl>
1    44    58    18      0
2   186    50    18      0
```

```
3   261    60    18      1
```

```
nrow(under)
```

```
[1] 38
```

If you would like to filter on multiple conditions, you can use additional logical operations. If we would like to subset data with BMI less than 18.5 and age above 30 years, we would run:

```
df %>% filter(bmi < 18.5 & age > 30)
```

```
# A tibble: 38 x 4
   pptid   age   bmi gender
   <dbl> <dbl> <dbl>  <dbl>
 1    44    58  18         0
 2   186    50  18         0
 3   261    60  18         1
 4   304    77  18         0
 5   397    85  18.4       0
 6   644    42  18.4       0
 7   652    69  18.1       1
 8   653    73  18.0       0
 9   717    48  18         0
10   821    41  18         0
# ... with 28 more rows
```

```
# A tibble: 3 x 4
  pptid   age   bmi gender
  <dbl> <dbl> <dbl>  <dbl>
1    44    58    18      0
2   186    50    18      0
3   261    60    18      1
```

In the `filter` function, you can pass in multiple logical indices. This implicitly assumes that you would like the intersection (or `AND`) of all these conditions. So the above code is equivalent to:

```
df %>% filter(bmi < 18.5, age > 30)
```

```
# A tibble: 38 x 4
   pptid   age   bmi gender
   <dbl> <dbl> <dbl>  <dbl>
 1    44    58  18         0
 2   186    50  18         0
 3   261    60  18         1
 4   304    77  18         0
 5   397    85  18.4       0
 6   644    42  18.4       0
```

```
 7   652    69  18.1      1
 8   653    73  18.0      0
 9   717    48  18        0
10   821    41  18        0
# ... with 28 more rows
```

```
# A tibble: 3 x 4
  pptid   age   bmi gender
  <dbl> <dbl> <dbl>  <dbl>
1    44    58    18      0
2   186    50    18      0
3   261    60    18      1
```

If you would like to use the `OR` operator (`|`), you must explicitly do that:

```
df %>% filter(bmi < 18.5 | age > 30)
```

```
# A tibble: 5,804 x 4
   pptid   age   bmi gender
   <dbl> <dbl> <dbl>  <dbl>
 1     1    55  21.8      1
 2     2    78  33.0      1
 3     3    77  24.1      0
 4     4    48  20.2      1
 5     5    66  23.3      0
 6     6    63  27.2      1
 7     7    52  30.0      1
 8     8    63  25.2      0
 9     9    69  25.8      1
10    10    40  27.8      1
# ... with 5,794 more rows
```

```
# A tibble: 3 x 4
  pptid   age   bmi gender
  <dbl> <dbl> <dbl>  <dbl>
1     1    55  21.8      1
2     2    78  33.0      1
3     3    77  24.1      0
```

17.2.4 Adding new columns to your data

Although we have shown how to drop columns using `select` above, we do not use `select` to create new columns. The `mutate` function will allow us to add and change columns of the data. Here we will add a column of indicators that BMI is under 18.5.

The action of the `mutate` function is to create a new variable, `low_bmi`, where

each entry corresponds to the indicator of whether or not the BMI of that person is less than 18.5, and add this variable to the `df` dataset.

17.2.5 Summarizing data

In most data analyses, you want to summarize variables by some statistic or measure. For example, let us calculate the average BMI for the entire population. To do this, we will use the `dplyr` verb `summarize` (the `summarise` is the same for those who prefer that spelling).

```
df %>%
  summarize(mean(bmi, na.rm = TRUE))
```

```
# A tibble: 1 x 1
  `mean(bmi, na.rm = TRUE)`
                      <dbl>
1                      28.2
```

Here we see the result is a `tibble` with one row and one column. The column name defaulted to the executed code, which is a bit messy. We can explicitly set the name of the output similarly to the way we did with `mutate`:

```
df %>%
  summarize(mean_bmi = mean(bmi, na.rm = TRUE))
```

```
# A tibble: 1 x 1
  mean_bmi
     <dbl>
1     28.2
```

We can summarize multiple columns and perform multiple summarizations on the same data:

```
df %>%
  summarize(mean_bmi = mean(bmi, na.rm = TRUE),
            mean_age = mean(age, na.rm = TRUE),
            sd_bmi = sd(bmi, na.rm = TRUE))
```

```
# A tibble: 1 x 3
  mean_bmi mean_age sd_bmi
     <dbl>    <dbl>  <dbl>
1     28.2     63.1   5.09
```

The `n()` function will allow you to count the number of cases:

```
df %>%
  summarize(mean_bmi = mean(bmi, na.rm = TRUE),
            n = n())
```

```
# A tibble: 1 x 2
  mean_bmi     n
     <dbl> <int>
1     28.2  5804
```

Though `n()` counts the number of records, it does not indicate how many records had a non-missing BMI:

```
df %>%
  summarize(mean_bmi = mean(bmi, na.rm = TRUE),
            n = n(),
            n_bmi = sum(!is.na(bmi)))
```

```
# A tibble: 1 x 3
  mean_bmi     n n_bmi
     <dbl> <int> <int>
1     28.2  5804  5761
```

We see that although there were 5804 records, only 5761 had BMI values.

The `tally` function essentially wraps `summarize(n = n())` if you are interested only in the count:

```
df %>%
  tally()
```

```
# A tibble: 1 x 1
      n
  <int>
1  5804
```

Also, if you simply want to add a column of the `n` to the data, you can use the `add_tally`:

```
df %>%
  add_tally() %>%
  select(pptid, bmi, n) %>%
  head()
```

```
# A tibble: 6 x 3
  pptid   bmi     n
  <dbl> <dbl> <int>
1     1  21.8  5804
2     2  33.0  5804
3     3  24.1  5804
4     4  20.2  5804
5     5  23.3  5804
6     6  27.2  5804
```

The utility of this functionality will be clearer when we compute these values

after grouping the data.

17.3 Grouping data

In many instances above, such as with `mutate` and `summarize`, we would like to perform operations separately for different groups. The `group_by` function in `dplyr` allows us to create a grouped `data.frame`/`tibble`. Let us group the data by `gender`:

```
class(df)
```

```
[1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
```

```
df = df %>%
  group_by(gender)
class(df)
```

```
[1] "grouped_df" "tbl_df"     "tbl"        "data.frame"
```

We see that `df` now has an additional class of `grouped_df` added to it after grouping. Let us print out the dataset:

```
head(df)
```

```
# A tibble: 6 x 5
# Groups:   gender [2]
  pptid   age   bmi gender low_bmi
  <dbl> <dbl> <dbl>  <dbl> <lgl>
1     1    55  21.8      1 FALSE
2     2    78  33.0      1 FALSE
3     3    77  24.1      0 FALSE
4     4    48  20.2      1 FALSE
5     5    66  23.3      0 FALSE
6     6    63  27.2      1 FALSE
```

The printout now contains at the top an indicator of grouping: `# Groups: gender [2]`. We can see the groups of a dataset by the `groups` function:

```
groups(df)
```

```
[[1]]
gender
```

17.3.1 Ungrouping data

We can remove all the groups of a dataset using the `ungroup` function:

```
head(ungroup(df))
```

```
# A tibble: 6 x 5
  pptid   age   bmi gender low_bmi
  <dbl> <dbl> <dbl>  <dbl> <lgl>
1     1    55  21.8      1 FALSE
2     2    78  33.0      1 FALSE
3     3    77  24.1      0 FALSE
4     4    48  20.2      1 FALSE
5     5    66  23.3      0 FALSE
6     6    63  27.2      1 FALSE
```

Note that there are no groups printed. Also, as we did not re-assign `df`, only the result was printed; the `df` object is still grouped. Grouping is a very important procedure because it will dictate the behavior of certain functions. Groups do not need to be only one variable. For example, you may do `group_by(x1, x2)`, and so on. In many cases, you may perform an operation/summarization on a set of grouping variables, ungroup the data, and then compute other metrics on a different set of grouping variables.

17.3.2 Summarizing grouped data

Now that `df` is grouped, let us calculate the mean BMI and age as before:

```
df %>%
  summarize(mean_bmi = mean(bmi, na.rm = TRUE),
            mean_age = mean(age, na.rm = TRUE)
  )
```

```
# A tibble: 2 x 3
  gender mean_bmi mean_age
   <dbl>    <dbl>    <dbl>
1      0     28.0     63.2
2      1     28.3     63.1
```

Note that this output has summarization for each separate group.

17.4 Merging datasets

You may have multiple datasets/tables from a study. For example, you may have baseline demographic data and baseline clinical data. In order to create a dataset where we look at the demographic and clinical data together, we need to merge or join the data together. To do this, we need variables that match up in both datasets, called "key" variables. Many times these key variables are a patient identifier and are simply one variable.

We will be using the `dplyr` package to join datasets as well. To see different types of joining for `dplyr`, run `?join`. The type of joins we will discuss here are:

- `inner_join(x, y)` - only rows that match for `x` and `y` are kept
- `full_join(x, y)` - all rows of `x` and `y` are kept
- `left_join(x, y)` - all rows of `x` are kept even if not merged with `y`
- `right_join(x, y)` - all rows of `y` are kept even if not merged with `x`

You can think of the `left_join` and `right_join` functions as the same functions with the `x` and `y` datasets switched. They can give the data back using different ordering for the rows and columns, but the overall data would be the same result. To be clear, if you used `left_joint(x, y)` and `right_join(y, x)`, the same inherent output would be given, but not necessarily if you did `left_join(y, x)`.

There are great cheatsheets for merging data: the Data Wrangling Cheatsheet and a more up-to-date Data Transformation Cheatsheet from RStudio and collaborators.

We are going to use some toy data in this section to illustrate the different type of joins. Here we have baseline data on three people, with identifiers 1 to 3 and their ages at baseline. Suppose that no `outcome` data were collected of any of these individuals at the baseline visit.

```
base = data_frame(
  id = 1:3,
  age = seq(55,60, length=3))
base
```

```
# A tibble: 3 x 2
     id   age
  <int> <dbl>
1     1  55
2     2  57.5
3     3  60
```

Suppose that we have additional data on a subset of these people at two visits. Let us say ID 3 did not show up to any of these visits. Also, another person (ID 4) missed the baseline visit, but showed up at visit 1.

```
visits = data_frame(
  id = c(rep(1:2, 2), 4),
  visit = c(rep(1:2, 2), 1),
  outcome = rnorm(5))
visits
```

```
# A tibble: 5 x 3
     id visit outcome
  <dbl> <dbl>   <dbl>
```

```
1     1     1  -2.10
2     2     2   0.820
3     1     1   0.107
4     2     2   0.838
5     4     1  -0.866
```

The baseline data are thus stored in 3×2 dimensional `tibble` called `base` and the visit (longitudinal) data are stored in a 5×3 dimensional `tibble` called `visit`.

17.4.1 Inner join

We start with an inner join (keep only people with baseline and follow-up visits). Without specifying the variable to join upon, the join functions will look at the column names of the `x` dataset and the column names of the `y` dataset and join on **all** the columns that have the same name.

```
ij = inner_join(base, visits)
```

```
Joining, by = "id"
```

As you can see, `dplyr` prints out a message saying `Joining, by = "id"`.

```
ij
```

```
# A tibble: 4 x 4
     id   age visit outcome
  <dbl> <dbl> <dbl>   <dbl>
1     1  55       1  -2.10
2     1  55       1   0.107
3     2  57.5     2   0.820
4     2  57.5     2   0.838
```

The output has only the records with baseline data **and** follow-up data. The IDs missing from either of those datasets are removed in the `inner_join`.

```
unique(ij$id)
```

```
[1] 1 2
```

```
nrow(ij)
```

```
[1] 4
```

Note, the joining functions are case sensitive like all other aspects of `tibble`s in `R` (e.g., column `ID` and `id` are different). Let us rename the identifier column to `ID` in the baseline data but not in the follow-up data:

```
base %>%
  rename(ID = id) %>%
  inner_join( visits)
```

```
`by` required, because the data sources have no common variables
```

We see that `dplyr` did not find any variables in common (exact same name) and gave an error. In many cases, we want to be explicit about which variables we are merging on. Let's specify `by = "id"`:

```
inner_join(base, visits, by = "id")
```

```
# A tibble: 4 x 4
     id   age visit outcome
  <dbl> <dbl> <dbl>   <dbl>
1     1  55       1  -2.10
2     1  55       1   0.107
3     2  57.5     2   0.820
4     2  57.5     2   0.838
```

We get the same result as `ij` above, but we do not get any message, as we explicitly passed in which variables to join upon.

17.4.2 Left join

Suppose that we want to keep any record if it has baseline data, regardless of whether or not it has follow-up data. We will use a `left_join` here and **must** have `base` as the `x` dataset (otherwise it is a different operation):

```
lj = left_join(base, visits)
```

```
Joining, by = "id"
```

```
lj
```

```
# A tibble: 5 x 4
     id   age visit outcome
  <dbl> <dbl> <dbl>   <dbl>
1     1  55       1  -2.10
2     1  55       1   0.107
3     2  57.5     2   0.820
4     2  57.5     2   0.838
5     3  60      NA  NA
```

```
nrow(lj)
```

```
[1] 5
```

In this case the follow-up data in the joined dataset for those IDs that had no follow-up data are set to missing.

17.4.3 Right join

Suppose that we want to keep any record if it has follow-up data, regardless of whether or not it has baseline data. We will use a `right_join` here and **must** have `visits` as the `y` dataset (otherwise it is a different operation):

```
rj = right_join(base, visits)
```

```
Joining, by = "id"
```

```
rj
```

```
# A tibble: 5 x 4
     id   age visit outcome
  <dbl> <dbl> <dbl>   <dbl>
1     1  55       1  -2.10 
2     2  57.5     2   0.820
3     1  55       1   0.107
4     2  57.5     2   0.838
5     4  NA       1  -0.866
```

```
nrow(rj)
```

```
[1] 5
```

17.4.4 Right join: switching arguments

Here we show that reversing the arguments of `x` and `y` and conducting a `right_join` provides the same results as a `left_join` up to a rearrangement.

```
rj2 = right_join(visits, base)
```

```
Joining, by = "id"
```

```
rj2
```

```
# A tibble: 5 x 4
     id visit outcome   age
  <dbl> <dbl>   <dbl> <dbl>
1     1     1  -2.10   55  
2     1     1   0.107  55  
3     2     2   0.820  57.5
4     2     2   0.838  57.5
5     3    NA  NA      60  
```

```
nrow(rj2)
```

```
[1] 5
```

This gives the same output as the `left_join` from before, after rearrangement.

```
rj2 = arrange(rj2, id, visit) %>% select(id, visit, outcome, age)
lj = arrange(lj, id, visit) %>% select(id, visit, outcome, age)
```

We can test if the output is identical using the `identical()` command:

```
identical(rj2, lj) ## after some rearranging
```

```
[1] TRUE
```

Thus, you can use `left_join` or `right_join` and get the same output. Simply choose the one that makes the most sense conceptually and gives the data ordered the way you want (you can always rearrange though).

17.4.5 Full join

The `full_join` will give us all the data, regardless of whether an ID has baseline or follow-up data:

```
fj = full_join(base, visits)
```

```
Joining, by = "id"
```

```
fj
```

```
# A tibble: 6 x 4
     id   age visit outcome
  <dbl> <dbl> <dbl>   <dbl>
1     1  55       1  -2.10
2     1  55       1   0.107
3     2  57.5     2   0.820
4     2  57.5     2   0.838
5     3  60      NA  NA
6     4  NA       1  -0.866
```

```
nrow(fj)
```

```
[1] 6
```

Sometimes you msy want to merge on multiple variables at the same time, in which case the syntax remains the same; you simply pass in multiple key variable names.

We would like now to join datasets that have a different outcome measure. Let us call this `hr_visits` because it contains heart rate measurements.

```
hr_visits = data_frame(
  id = rep(1:2, 2),
  visit = rep(1:2, 2),
  hr = rpois(4, lambda = 100))
hr_visits
```

```
# A tibble: 4 x 3
     id visit    hr
  <int> <int> <int>
1     1     1   106
2     2     2   100
3     1     1   121
4     2     2   109
```

The person with ID 4 missed this portion of the visit, however, and has no heart rate data. We can merge this with the follow-up outcomes:

```
out = full_join(visits, hr_visits)
```

```
Joining, by = c("id", "visit")
```

```
out
```

```
# A tibble: 9 x 4
     id visit outcome    hr
  <dbl> <dbl>   <dbl> <int>
1     1     1  -2.10    106
2     1     1  -2.10    121
3     2     2   0.820   100
4     2     2   0.820   109
5     1     1   0.107   106
6     1     1   0.107   121
7     2     2   0.838   100
8     2     2   0.838   109
9     4     1  -0.866    NA
```

We see that the data are lined up with the correct heart rate/outcome data for each visit. We could have also specified this using:

```
full_join(visits, hr_visits, by = c("id", "visit"))
```

```
# A tibble: 9 x 4
     id visit outcome    hr
  <dbl> <dbl>   <dbl> <int>
1     1     1  -2.10    106
2     1     1  -2.10    121
3     2     2   0.820   100
4     2     2   0.820   109
5     1     1   0.107   106
```

```
6     1     1   0.107   121
7     2     2   0.838   100
8     2     2   0.838   109
9     4     1  -0.866    NA
```

17.4.6 Non-joining columns with identical names

What if we messed up and forgot that these datasets should be merged with the `visit` as well as the `id`:

```
full_join(visits, hr_visits, by = "id")
```

```
# A tibble: 9 x 5
     id visit.x outcome visit.y    hr
  <dbl>   <dbl>   <dbl>   <int> <int>
1     1       1  -2.10        1   106
2     1       1  -2.10        1   121
3     2       2   0.820       2   100
4     2       2   0.820       2   109
5     1       1   0.107       1   106
6     1       1   0.107       1   121
7     2       2   0.838       2   100
8     2       2   0.838       2   109
9     4       1  -0.866      NA    NA
```

We see the data now have a `visit.x` and `visit.y` variable. In some cases, we may want `R` to perform this. For example, let us say we had 2 raters provide scores to the same individuals, that each rater had spreadsheets with the exact same individual names, and that we wanted to merge the spreadsheets and have rater 1 information in 1 column and rater 2 information in another column. Then we can do:

```
full_join(rater1, rater2, by = "id", suffix = c("_1", "_2"))
```

Using this approach the columns that have identical names would have suffixes on them.

17.5 Reshaping datasets

See http://www.cookbook-r.com/Manipulating_data/Converting_data_between_wide_and_long_format/ for more tutorials on reshaping data if this section is not extensive enough.

17.5.1 Data formats: wide versus long

In a "wide" dataset, there are usually multiple columns per observation. For example for multiple visits, we may have the dataset up as follow:

```
# A tibble: 2 x 4
     id visit1 visit2 visit3
  <int>  <dbl>  <dbl>  <dbl>
1     1     10      4      3
2     2      5      6     NA
```

In the terminology of the `tidyverse`, this is not tidy data. In a "long" dataset, there are multiple rows per observation; for example:

```
# A tibble: 5 x 3
     id visit   value
  <dbl> <chr>   <dbl>
1     1 visit_1    10
2     1 visit_2     4
3     1 visit_3     3
4     2 visit_1     5
5     2 visit_2     6
```

These are the common ways of thinking about wide or long data. More accurately, however, you should think of data that are wide or long **with respect** to certain variables.

17.5.2 Reshaping data in R using tidyr

The `reshape` command exists in base `R`. It is a **confusing** function. Do not use it. The `tidyr` allows you to "tidy" your data (Wickham and Henry 2018). The functions we will cover are:

- `gather` - make multiple columns into variables (wide to long)
- `spread` - make a variable into multiple columns (long to wide)
- `separate` - separate one string column into multiple columns
- `unite` - paste multiple columns into one string column

The `gather` function takes a set of columns and puts the column names into the `key` variable and the column values into the `value` variable. Here, we take the example with data in wide format and gather all columns that start with `"visit"` using the `starts_with` helper function. When gathering columns you can use the `select_helpers` as before:

```
gather(ex_wide, key = "visit", value = "outcome", starts_with("visit"))
```

```
# A tibble: 6 x 3
     id visit  outcome
```

```
  <int> <chr>    <dbl>
1     1 visit1      10
2     2 visit1       5
3     1 visit2       4
4     2 visit2       6
5     1 visit3       3
6     2 visit3      NA
```

The `spread` function takes a set of rows, puts the column names **from** the `key` variable and the values from the `value` variable to fill it in. Here, we take the example long data and spread the `visit` column:

```
spread(ex_long, key = "visit", value = "value")
```

```
# A tibble: 2 x 4
     id visit_1 visit_2 visit_3
  <dbl>   <dbl>   <dbl>   <dbl>
1     1      10       4       3
2     2       5       6      NA
```

Using `separate`, we can break up the `visit` column into a column that just says "visit" (calling it `tmp` and we will remove it later) and another column of the `visit_number`:

```
sep = ex_long %>%
  separate(visit, into = c("tmp", "visit_num"), sep = "_")
sep
```

```
# A tibble: 5 x 4
     id tmp   visit_num value
  <dbl> <chr> <chr>     <dbl>
1     1 visit 1            10
2     1 visit 2             4
3     1 visit 3             3
4     2 visit 1             5
5     2 visit 2             6
```

We could then `mutate` the `visit_num` column into a numeric column using `as.numeric`. Note that, by default, `separate` removes the column you separated. If you want to keep that variable, set the argument `remove = FALSE`.

```
ex_long %>%
  separate(visit, into = c("tmp", "visit_num"), sep = "_", remove = FALSE)
```

```
# A tibble: 5 x 5
     id visit   tmp   visit_num value
  <dbl> <chr>   <chr> <chr>     <dbl>
1     1 visit_1 visit 1            10
2     1 visit_2 visit 2             4
```

```
3     1 visit_3 visit 3             3
4     2 visit_1 visit 1             5
5     2 visit_2 visit 2             6
```

If you do not remove the column containing the names of the column you are separating, make sure to not create a column with the same name. For example, we are splitting `visit`, not removing it, but creating a new column named `visit` in the `into` argument, and will therefore override the original `visit` column:

```
ex_long %>%
  separate(visit, into = c("tmp", "visit"), sep = "_", remove = FALSE)
```

```
# A tibble: 5 x 4
     id value tmp   visit
  <dbl> <dbl> <chr> <chr>
1     1    10 visit 1
2     1     4 visit 2
3     1     3 visit 3
4     2     5 visit 1
5     2     6 visit 2
```

If we wish to perform the opposite operation, pasting columns into one column with a separator, we can use the `unite` function:

```
sep %>%
  unite(col = "new_visit_name", tmp, visit_num, sep = ".")
```

```
# A tibble: 5 x 3
     id new_visit_name value
  <dbl> <chr>          <dbl>
1     1 visit.1           10
2     1 visit.2            4
3     1 visit.3            3
4     2 visit.1            5
5     2 visit.2            6
```

17.6 Recoding variables

The `recode` function from `dplyr` allows you to recode values of type character or factor. The `recode_factor dplyr` function performs the same operation but matches the order of the replacements when replacing factor levels. The `forcats` package was specifically made for categorical variables/factors.

The `levels` function exists to extract levels from a factor. Do **not** use the levels for assignment (e.g. `levels(x) = c("level1", "level2")`) as this can completely reset your levels. To change the levels, either use the `factor(x,`

`levels = c("level1", "level2"))` if reordering the whole factor or use the `relevel` function to change the reference level.

17.7 Cleaning strings: the `stringr` package

One of the hardest things in data cleaning is cleaning text fields. These are usually free text fields where users can input completely unstructured information. We will use the `dslabs reported_heights` dataset, where the description of the data is:

> Students were asked to report their height in inches and sex in an online form. This table includes the results from four courses.

Let us just copy the data over so that we are sure that we do not change anything in the original data:

```
library(dslabs)
bad_strings = reported_heights
```

Similarly, we will make a duplicate height column to compare to if anything goes wrong during string manipulation:

```
bad_strings =  bad_strings %>%
  mutate(original_height = height)
```

We see that the height column is a character vector:

```
class(bad_strings[, "height"])
```

```
[1] "character"
```

If we were very hopeful and made the height a numeric variable straight away, we would not have known whether some of the values are missing values or are below a certain biological threshold, say 48. Moreover, we would not know whether 48 refers to inches or centimeters.

```
bad_strings = bad_strings %>%
  mutate(num_height = as.numeric(height))
```

```
Warning: NAs introduced by coercion
```

```
bad_strings %>%
  filter(is.na(num_height) | num_height < 48) %>%
  select(height) %>%
  head(10)
```

```
   height
1       6
2   5' 4"
```

```
3     5.3
4   165cm
5       6
6       2
7     5'7
8   >9000
9    5'7"
10   5'3"
```

We will try to fix some of these heights using informed guesses and the `stringr` package. The `stringr` package:

- Makes string manipulation more intuitive
- Has a standard format for most functions
  - the first argument is a string like first argument is a `data.frame` in `dplyr`
- Almost all functions start with `str_`

Here are some of the functions we will use from `stringr`:

- `str_detect` - returns `TRUE` if `pattern` is found
- `str_subset` - returns only the strings whose patterns were detected
  - convenient wrapper around `x[str_detect(x, pattern)]`
- `str_extract` - returns only strings whose patterns were detected, but ONLY the pattern, not the whole string
- `str_replace` - replaces `pattern` with `replacement` the first time
- `str_replace_all` - replaces `pattern` with `replacement` as many times matched

Let us look at records that have quotes in them:

```
bad_strings %>%
  filter(str_detect(height, "'") | str_detect(height, '"')) %>%
  head(10)
```

```
            time_stamp    sex height original_height num_height
1  2014-09-02 15:16:28   Male  5' 4"           5' 4"         NA
2  2014-09-02 15:16:52   Male    5'7             5'7         NA
3  2014-09-02 15:16:56   Male   5'7"            5'7"         NA
4  2014-09-02 15:17:09 Female   5'3"            5'3"         NA
5  2014-09-02 15:19:48   Male   5'11            5'11         NA
6  2014-09-04 00:46:45   Male  5'9''           5'9''         NA
7  2014-09-04 10:29:44   Male 5'10''          5'10''         NA
8  2014-09-11 01:02:37   Male     6'              6'         NA
9  2014-09-18 21:40:23   Male  5' 10           5' 10         NA
10 2014-10-19 13:08:30   Male   5'5"            5'5"         NA
```

We will use some regular expressions to match specific things in the data. The first `str_detect` will look for any spaces in the height column and the second

search looks for any alphabetical characters:

```
# find a space
bad_strings %>%
  filter(str_detect(height, "\\s")) %>%
  head(10)
```

```
            time_stamp    sex                 height
1  2014-09-02 15:16:28   Male                  5' 4"
2  2014-09-02 15:18:00   Male 5 feet and 8.11 inches
3  2014-09-18 21:40:23   Male                  5' 10
4  2014-10-08 19:19:33 Female Five foot eight inches
5  2015-04-21 16:40:25 Female               5' 7.78"
6  2015-05-27 08:03:32   Male         5 feet 7inches
7  2015-08-28 12:27:14   Male                  5 .11
8  2015-11-28 16:16:36   Male                   5 11
9  2016-01-26 09:55:07   Male                 5' 11"
10 2016-01-26 11:55:08   Male                  5' 7"
          original_height num_height
1                    5' 4"         NA
2  5 feet and 8.11 inches         NA
3                    5' 10         NA
4  Five foot eight inches         NA
5                 5' 7.78"         NA
6           5 feet 7inches         NA
7                    5 .11         NA
8                     5 11         NA
9                   5' 11"         NA
10                   5' 7"         NA
```

```
bad_strings %>%
  filter(str_detect(height, "[:alpha:]")) %>%
  head(10)
```

```
            time_stamp    sex                 height        original_height
1  2014-09-02 15:16:37 Female                  165cm                  165cm
2  2014-09-02 15:18:00   Male 5 feet and 8.11 inches 5 feet and 8.11 inches
3  2014-10-08 19:19:33 Female Five foot eight inches Five foot eight inches
4  2015-05-12 19:23:51 Female                    yyy                    yyy
5  2015-05-27 08:03:32   Male         5 feet 7inches         5 feet 7inches
6  2016-01-26 15:19:32   Male           5ft 9 inches           5ft 9 inches
7  2016-01-26 15:19:51   Male          5 ft 9 inches          5 ft 9 inches
8  2016-03-18 00:45:39   Male        5 feet 6 inches        5 feet 6 inches
9  2017-06-19 04:20:32   Male                 170 cm                 170 cm
  num_height
1         NA
2         NA
```

```
3         NA
4         NA
5         NA
6         NA
7         NA
8         NA
9         NA
```

17.7.1 Escape characters

The backslash \ is referred to as an escape character in regular expressions. You can use that when looking for "special" strings that mean something in regular expressions. `R`, however, has special strings that mean something, like `\n` for a new line. Therefore, if you want to use it as an escape character, you have to do a "double escape" `\\`. For example, if you want to find a dollar sign `$` in a string, you have to escape it first so that `R` knows you do not mean the regular expression meaning for `$`, which is the end of a string:

```
str_detect(c("$4.29", "5.99"), "$")
```

```
[1] TRUE TRUE
```

```
str_detect(c("$4.29", "5.99"), "\\$")
```

```
[1]  TRUE FALSE
```

The `stringr` package also has a set of `modifiers`, which allow you to tell it if you are looking for a `fixed` string (do not use regular expressions) or `regrex` (use regular expressions). See `?modifiers` for other modifiers. For example, we can use the `fixed` command to say we want to match exactly the dollar sign:

```
str_detect(c("$4.29", "5.99"), fixed("$"))
```

```
[1]  TRUE FALSE
```

Back to cleaning the `height` variable. We now will make the `height` variable lower case (using `tolower()`) and remove any white space from the beginning and end of the string (using `trimws()` in `base` or `str_trim` in `stringr`):

```
bad_strings = bad_strings %>%
  mutate(height = tolower(height),
         height = trimws(height))
```

For the remainder of the section, we will use the `match_replace_string` function below:

```
match_replace_string = function(pattern, replacement = "", n = 3) {
  d = bad_strings %>%
    filter(str_detect(height, pattern)) %>%
```

```
    select(height, original_height)
  if (nrow(d) > 0) {
    d = d %>%
      mutate(replaced_height = str_replace(height, pattern, replacement))
    print(head(d, n = n))
  }
}
```

The purpose of this function is to print out the rows of the data that match a specific pattern, then show the output when that pattern is replaced. For example, if we are removing double spaces, we can run:

```
match_replace_string("\\s\\s+", " ")
```

where `\\s` means a space, and then `\\s+` means any number of additional spaces (can be zero spaces). These multiple spaces will be changed to only one space. This code has done nothing to the `bad_strings` object. The code we will execute will reassign the output to `bad_strings`, as described below:

```
bad_strings = bad_strings %>%
  mutate(height = str_replace_all(height, "\\s\\s+", " "),
         height = trimws(height) )
```

Notice, nothing is printed out here, but the changes have been made as in standard `mutate` calls. We will use this `match_replace_string` to simplify our code and demonstrate the changes being made to the `bad_strings` dataset. We used `str_replace_all` because we wanted to replace all instances of the double spacing as opposed to the first instance (which `str_replace` does). The difference can be demonstrated using a simple example:

```
x = " I have  many   double    spaces  "
str_replace(x, "\\s\\s+", " ")
```

```
[1] " I have many   double    spaces  "
```

Note that the first double space in between `have` and `many` has been removed. Let us replace all the multiple spaces:

```
str_replace_all(x, "\\s\\s+", " ")
```

```
[1] " I have many double spaces "
```

```
x %>% str_replace_all("\\s\\s+", " ") %>% trimws
```

```
[1] "I have many double spaces"
```

We trimmed the whitespace again from the result using `trimws`. We now will loop over the numbers from 1 to 12 (as there are 12 inches in 1 foot) to replace these values with their numeric version. Also, note we must use `as.character(word)` in the loop, as `str_replace` requires the replacement to

be a character, not a number:

```
num_words <- c("one" = 1, "two" = 2, "three" = 3, "four" = 4,
               "five" = 5, "six" = 6, "seven" = 7, "eight" = 8, "nine" = 9,
               "ten" = 10, "eleven" = 11, "twelve" = 12)

for (iword in seq_along(num_words)) {
  word = num_words[iword]
  match_replace_string(names(word), as.character(word))
  bad_strings = bad_strings %>%
    mutate(height = str_replace(height, names(word), as.character(word))
    )
}
```

```
                  height        original_height     replaced_height
1 five foot eight inches Five foot eight inches 5 foot eight inches
               height        original_height replaced_height
1 5 foot eight inches Five foot eight inches 5 foot 8 inches
```

We will remove inches from the text and replace any demarcation of feet with an apostrophe/single quote to standardize:

```
match_replace_string("inches", "")
```

```
                  height        original_height  replaced_height
1 5 feet and 8.11 inches 5 feet and 8.11 inches 5 feet and 8.11
2        5 foot 8 inches Five foot eight inches        5 foot 8
3         5 feet 7inches         5 feet 7inches         5 feet 7
```

```
match_replace_string("and", "")
```

```
                  height        original_height     replaced_height
1 5 feet and 8.11 inches 5 feet and 8.11 inches 5 feet  8.11 inches
```

```
match_replace_string("ft|foot|feet", "'")
```

```
                  height        original_height     replaced_height
1 5 feet and 8.11 inches 5 feet and 8.11 inches 5 ' and 8.11 inches
2        5 foot 8 inches Five foot eight inches        5 ' 8 inches
3         5 feet 7inches         5 feet 7inches         5 ' 7inches
```

```
bad_strings = bad_strings %>%
  mutate(height = str_replace(height, "inches", ""),
         height = str_replace(height, "and", ""),
         height = str_replace(height, "ft|foot|feet", "'"),
         height = trimws(height) )
```

We will remove the spaces from before or after an apostrophe, denoting the number of feet:

```
match_replace_string("' ", "'")
```

```
     height        original_height replaced_height
1     5' 4"                 5' 4"            5'4"
2 5 '  8.11 5 feet and 8.11 inches        5 ' 8.11
3     5' 10                 5' 10            5'10
```

```
match_replace_string(" '", "'")
```

```
     height        original_height replaced_height
1 5 '  8.11 5 feet and 8.11 inches        5'  8.11
2     5 ' 8 Five foot eight inches            5' 8
3     5 ' 7         5 feet 7inches            5' 7
```

Here we replace these spaces with nothing so that we further standardize the formats:

```
bad_strings = bad_strings %>%
  mutate(
    height = str_replace_all(height, " ' ", "'"),
    height = str_replace_all(height, "' | '", "'"),
    height = trimws(height) )
```

We also have some records with the double quote (") denoting inches; we will remove these. Also, we have some records that use two single quotes/apostrophes ('') to denote inches. We will remove these, making sure they are at the end of the string using the $ regular expression operator. Note that we can use the escape operator \ to be able to use a double quote pattern inside a double quoted string:

```
match_replace_string('"', "")
```

```
  height original_height replaced_height
1   5'4"           5' 4"             5'4
2   5'7"            5'7"             5'7
3   5'3"            5'3"             5'3
```

```
match_replace_string("''$", "")
```

```
  height original_height replaced_height
1  5'9''           5'9''             5'9
2 5'10''          5'10''            5'10
3 5'10''          5'10''            5'10
```

```
bad_strings = bad_strings %>%
  mutate(height = str_replace(height, "(\")|(''$)", ""),
         height = trimws(height)
  )
```

We have not made any changes to the num_height variable or made the data

ready to be turned into a numeric just yet. Let's try to create a column indicating the units of the data. We will assume that these are all adults, and will assume that no one is above 7 feet tall (84 inches) and that values over 84 are expressed in centimeters (but must be less than 214 cm $\approx$ 7 ft). We will also assume that anyone who has a value for height less than 10 had given his/her height in feet only. We will create units based on the above rules using `case_when`, which takes in logical arguments, then maps them to the replacement when the logical condition is `TRUE`:

```
bad_strings = bad_strings %>%
  mutate(num_height = as.numeric(height))
```

```
Warning: NAs introduced by coercion
```

```
bad_strings = bad_strings %>%
  mutate(unit = case_when(
    str_detect(height, "cm") ~ "cm",
    str_detect(height, "'") ~ "feet",
    num_height < 1.2 ~ "unknown",
    num_height >= 1.2 & num_height <= 3 ~ "check",
    num_height > 3 & num_height <= 10 ~ "feet",
    num_height > 10 & num_height < 48  ~ "check",
    num_height >= 48 & num_height < 84  ~ "in",
    num_height > 84 & num_height <= 121.92 ~ "check",
    num_height > 121.92 & num_height < 214 ~ "cm",
    num_height >= 214 ~ "unknown"
    )
  )
table(bad_strings$unit, useNA = "ifany")
```

```
  check      cm    feet      in unknown    <NA>
     20      94     150     804      12      15
```

There may be some individuals who recorded a number either under 1.2 for meters, or under 48 inches, or between 84 and 121.92 centimeters that we can check later. Likely, we will not be able to infer their height and will need to drop the data. Let us remove the `cm` string from the data and them remove whitespace:

```
match_replace_string("cm", "")
```

```
  height original_height replaced_height
1  165cm           165cm             165
2 170 cm          170 cm            170
```

```
bad_strings = bad_strings %>%
  mutate(height = str_replace(height, "cm", ""),
         height = trimws(height)
```

```
  )
```

There are some records that still have spaces in the height data. Here we will show you the function `stopifnot`, which takes in a logical statement and will error if that statement is `FALSE`. When we checked these data, we saw the cases that had spaces. If new data come in or the data change in any way that results in a different number of records with spaces, this will prompt an error message to alert the analyst that something has changed and to re-check the data.

```
match_replace_string(" ", "", n = 10)
```

```
  height original_height replaced_height
1  5 .11           5 .11            5.11
2   5 11            5 11             511
3   6 04            6 04             604
```

```
have_space = str_detect(bad_strings$height, " ")
stopifnot(sum(have_space) == 3)
```

We are assuming that these cases are all in feet, and we can add in the quote accordingly and update the unit. We will use the `if_else` function that takes in a logical statement, and then returns the next argument if that statement is true or the next argument if that statement is false. Here we will change the `height` variable to add in the `'` and then change the unit:

```
bad_strings = bad_strings %>%
  mutate(
    height = if_else(
      str_detect(height, " "),
      str_replace(height, " ", "'"),
      height),
    unit = if_else(is.na(unit) & str_detect(height, "'"), "feet", unit))
table(bad_strings$unit, useNA = "ifany")
```

```
  check      cm    feet      in unknown    <NA>
     20      94     153     804      12      12
```

What about commas? These are some of the oddest of the bunch with respect to format:

```
match_replace_string(",", "'", n = 10)
```

```
     height original_height replaced_height
1       5,3             5,3             5'3
2       6,8             6,8             6'8
3       5,4             5,4             5'4
4       5,8             5,8             5'8
5      1,70            1,70            1'70
```

```
6   708,661         708,661         708'661
7   649,606         649,606         649'606
8   728,346         728,346         728'346
9 7,283,465       7,283,465       7'283,465
```

The only types that seem to make sense are those that are the cases such as `5,3` which indicate 5 feet, 3 inches, but again this is an assumption. Let us subset the data with commas and split them based on the comma:

```
comma_data = bad_strings %>%
  filter(str_detect(height, ",") & is.na(unit)) %>%
  select(height) %>%
  unlist
comma_data = strsplit(comma_data, ",")
```

Now we have them separated. If there is more than one comma, we cannot infer what the units are or what the height is. If the first value is between 3 and 10 feet, then we will make the output the "feet" format (e.g. `5,3` becomes `5'3`):

```
comma_data = map_chr(comma_data, function(x) {
  x = as.numeric(x)
  if (length(x) > 2) {
    x = NA
  }
  if (!is.na(x[1]) & between(x[1], 3, 10)) {
    x = paste0(x, collapse = "'")
  } else {
    x = NA
  }
  x
})
```

Now we replace these comma data with the replaced values (even the `NA`) and change the units for those we have changed to "feet":

```
bad_strings = bad_strings %>%
  mutate(comma = str_detect(height, ",") & is.na(unit),
         height = ifelse(comma, comma_data, height),
         unit = ifelse(comma & !is.na(height), "feet", unit),
         unit = ifelse(comma &  is.na(height), "unknown", unit)
         )
```

We can now check some of our progress with those missing or unknown units:

```
bad_strings %>%
  filter(is.na(unit) | unit %in% c("check", "unknown")) %>%
  head(10)
```

```
            time_stamp    sex height original_height num_height    unit
```

```
1  2014-09-02 15:16:41   Male    511             511      511.0 unknown
2  2014-09-02 15:16:50 Female      2               2        2.0   check
3  2014-09-02 15:16:56   Male  >9000           >9000         NA    <NA>
4  2014-09-03 23:55:37   Male  11111           11111    11111.0 unknown
5  2014-09-04 14:23:19   Male  103.2           103.2      103.2   check
6  2014-09-06 21:32:15   Male     19              19       19.0   check
7  2014-09-07 20:53:43   Male    300             300      300.0 unknown
8  2014-09-09 20:00:38   Male   <NA>             5,3         NA unknown
9  2014-09-15 14:38:58   Male   <NA>             6,8         NA unknown
10 2014-11-09 15:43:21 Female   <NA>             5,4         NA unknown
   comma
1  FALSE
2  FALSE
3  FALSE
4  FALSE
5  FALSE
6  FALSE
7  FALSE
8   TRUE
9   TRUE
10  TRUE
```

We also define a function that converts the feet to inches. This is done by multiplying the number of feet by 12 and adding the corresponding number of inches. We are going to take in a vector of heights in `x`, then split it on the single quote for the feet/inches delimiter. We will take this list and map it to a double/numeric using `map_dbl`. We define a function taking in one of these split strings, calling it `r`. If there is no `'`, then `r` will have only one element, and `r[2]` (the second element) will return `NA`, so no inches were specified.

```
feet_to_inches = function(x) {
  ss = str_split(x, pattern = "'")
  num = map_dbl(ss, function(r) {
    if (is.na(r[2])) {
      r[2] = 0
    }
    if (r[2] == "") {
      r[2] = 0
    }
    r = as.numeric(r)
    r[1] = r[1] * 12
    r = sum(r)
    })
  num
}
cm_to_inches = function(x) {
```

```
  x = as.numeric(x)
  x / 2.54
}
```

we now convert feet to inches using our function from above:

```
bad_strings = bad_strings %>%
  mutate(
    feet = unit == "feet",
    height = ifelse(feet,
                    feet_to_inches(height),
                    height),
    unit = ifelse(feet, "in", unit)
    ) %>%
  select(-feet)
```

```
Warning in .f(.x[[i]], ...): NAs introduced by coercion

Warning in .f(.x[[i]], ...): NAs introduced by coercion

Warning in .f(.x[[i]], ...): NAs introduced by coercion
```

We will do the same for centimeters to inches:

```
bad_strings = bad_strings %>%
  mutate(
    cm = unit == "cm",
    height = ifelse(cm,
                    cm_to_inches(height),
                    height),
    unit = ifelse(cm, "in", unit)
    ) %>%
  select(-cm)
```

Figure 17.1 provides the histogram of these numeric heights for quality control purposes:

```
bad_strings %>%
  ggplot(aes(x = num_height)) +
  geom_histogram()
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Clearly, there is something wrong with the distribution, most likely because we include cases that had a numeric value, but unknown unit. Let us plot the histogram of numeric heights to see the distribution in inches only:

```
bad_strings %>%
  filter(unit %in% "in") %>%
```

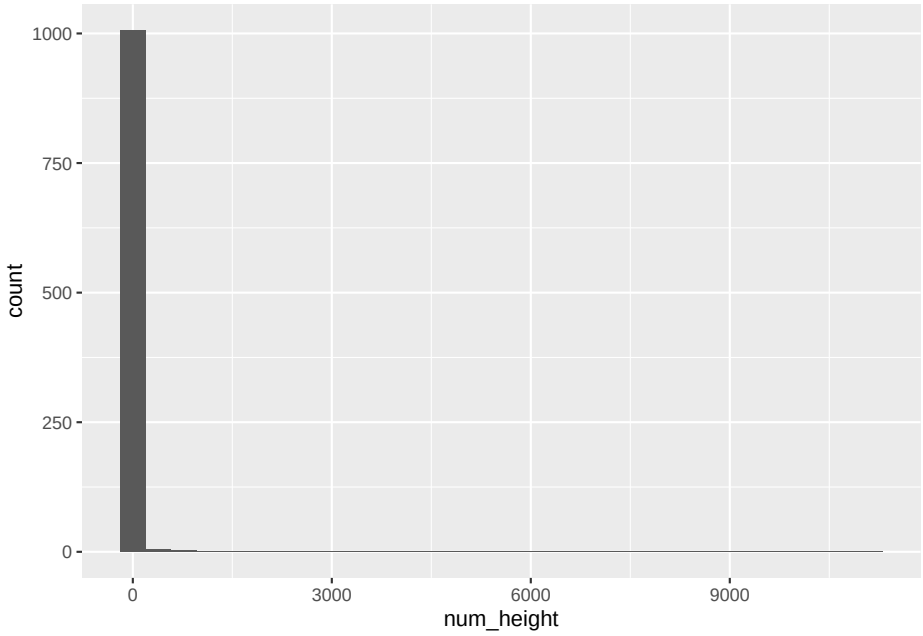


Figure 17.1: Histogram of numeric heights for quality control purposes.

```
  ggplot(aes(x = num_height)) +
  geom_histogram()
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

We can check our progress overall for the data:

```
bad_strings = bad_strings %>%
  mutate(num_height = as.numeric(height))
bad_strings %>%
  filter(is.na(num_height) | unit == "check") %>%
  select(height, original_height, num_height, unit) %>% head(10)
```

```
   height original_height num_height    unit
1       2               2        2.0   check
2    <NA>           >9000         NA    <NA>
3   103.2           103.2      103.2   check
4      19              19       19.0   check
5    <NA>             5,3         NA unknown
6    <NA>             6,8         NA unknown
7    <NA>             5,4         NA unknown
8    <NA>             5,8         NA unknown
9      87              87       87.0   check
```

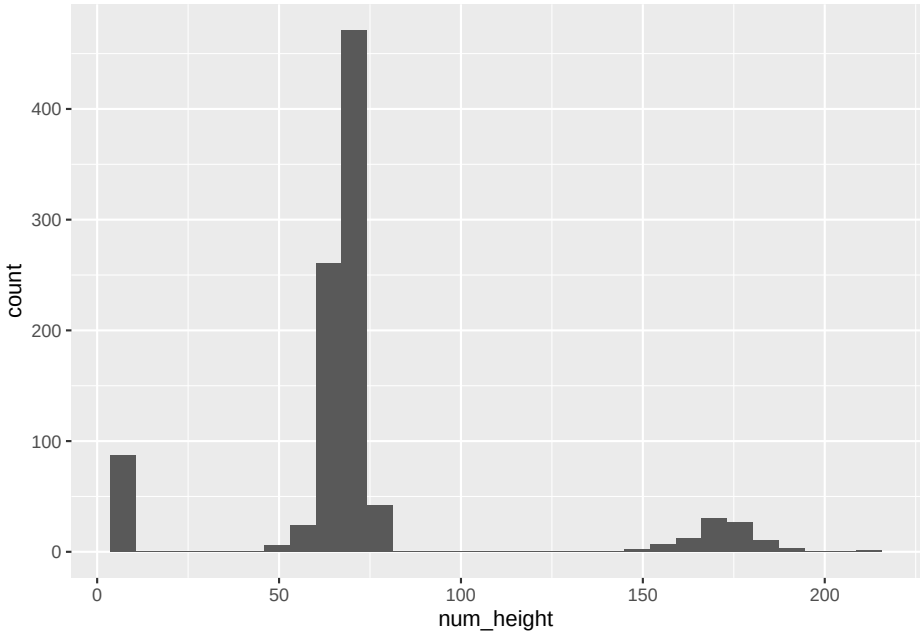


Figure 17.2: Histogram of numeric heights in inches for quality control purposes.

```
10    111             111      111.0   check
```

We can now see the `NA` values in the `height` variable corresponding to the bad comma data in the `original_height` variable. Let us look at the data that that did not have commas:

```
bad_strings %>%
  filter(is.na(num_height) | unit == "check") %>%
  filter(!is.na(height)) %>%
  select(height, original_height) %>%
  head(10)
```

```
   height original_height
1       2               2
2   103.2           103.2
3      19              19
4      87              87
5     111             111
6      12              12
7      89              89
8      34              34
9      25              25
10     22              22
```

For most of these data we cannot infer the numeric height and we will have to keep them as missing. There may be some cases expressed in meters that we are still missing, but let us look at how many such cases exist:

```
tab = table(bad_strings$unit, useNA = "ifany")
tab
```

```
  check      in unknown    <NA>
     20    1055      17       3
```

```
ptab = prop.table(tab)
round(ptab,digits=3)
```

```
  check      in unknown    <NA>
  0.018   0.963   0.016   0.003
```

We see that we are reasonably sure that 96% of the data are converted to inches. But to be clear, our cutoffs for the numeric values are somewhat arbitrary. You may get a different mean height given different assumptions of the cutoffs for inches or centimeters. Without further information (or more assumptions of the data), we cannot get to 100% "clean" data. This is common in cleaning exercises and illustrates the importance of working with a team of people who collect and manage the data to get additional information instead of excluding data or making additional assumptions.

17.8 Problems

Problem 1. In the `reported_heights` dataset from `dslabs`:

a. Display the records with double spaces in `height` using `filter`.
b. Replace the double spaces with single spaces.
c. Remove the first single quotes with a semicolon.
d. Replace all the double quotes with nothing (empty string `""`).

Problem 2. With the following vector:

```
vec = c("x^4", "y^5", "J**2", "3.5", "4,200,234")
```

a. Replace a `^` with `**`. (Note `^` is a special character)
b. Replace all `,` with nothing (empty string `""`).
c. Replace any upper case to the lower case.
d. Extract all numbers from the string.

Problem 3. In the `admissions` dataset from `dslabs`:

a. Create a `pct_adm` variable by dividing the `admitted` by number of `applicants`.

b. Spread the `pct_adm` variable where the result has a column for `men` and one for `women`. You may have to select only certain variables.
c. Gather the `admitted`, `applicants`, and `pct_adm` variables into one column called `value` with the numbers in there and a column called `variable` for the variable name.

Problem 4. In the `admissions` dataset from `dslabs`:

a. Filter the data on the `gender == "men"` and call that dataset `men`. Do the same for `women`. Keep the `major` and `admitted` columns.
b. Perform an `inner_join` on the `men` and `women` datasets. Do not specify on what to join on.
c. Perform an `inner_join` on the `men` and `women` datasets, by `major`. Call this `joined`.

d. Perform an `inner_join` on the `men` and `women` datasets, by `major`. Replace `joined` with this and create suffixes of `c("_men", "_women")`, respectively.

Problem 5. Use the `us_contagious_diseases` dataset from `dslabs`:

a. Create a `data.frame` named `means`, where you summarize each `state` and disease by the counts per 100000 of the population (you need to create variables). Summarize over all years using the mean and median incidence per 100000 people and count the number of non-missing years.
b. Plot a scatterplot of the mean incidence `Measles` versus `Mumps`. You may need to reshape `means` before plotting.
c. Plot a histogram of the mean incidence for each disease, colored by the disease.

Problem 6. In the `admissions` dataset from `dslabs`:

a. Subset the columns of `gender` and `admitted` using `select`.
b. Rename the `gender` column to `sex` using `rename`.
c. Make the column `log_app` which is the `log(applicants)` using `mutate`.

Chapter 18

Power

This chapter covers the following topics

- Introduction
- Power calculation for Normal tests
- Power for the t test
- Discussion

18.1 Introduction

In chapter 16 we introduced hypothesis tests and the concept of power. Power is the probability of rejecting the null hypothesis when it is actually false. Thus, as its name would suggest, power is a good thing; we want studies with more power. In fact, we saw in chapter 16 that the (Bayesian) probability that a rejection of the null hypothesis is correct involved power:

$$P(H_a \mid \text{Significance}) = \frac{\text{Power} \times P(H_a)}{\text{Power} \times P(H_a) + \alpha \times P(H_0)} \, .$$

Interestingly, if we convert this statement to odds we obtain the following:

$$\text{Odds}(H_a \mid \text{Significance}) = \frac{\text{Power}}{\alpha} \times \text{Odds}(H_a) \, ,$$

That is, the posterior odds of H_a, given that you reject, is the prior odds of H_a multiplied by the ratio of power to the Type I error rate. For a power of 0.05, it is an even bet (odds of 1) that a significant hypothesis test is actually significant. A rule of thumb to remember is that your odds that a significant test is actually significant is 20 times the power times that of the prior odds.

How can we control power? Power can be related to variability, the value of the parameter under the alternative, the study design, the sample size, the style of analysis, the Type I error rate, and other factors. Design considerations and sample size are the two components that are most under our control. In fact nearly all National Institutes of Health (NIH) funded studies require a power or sample size calculation to receive funding.

We will describe the basics of both here. Broadly, in a power calculation, one fixes the sample size and other aspects of the study and calculates an estimate of the power. In a sample size calculation, one fixes the power, then determines the minimum sample size to achieve that power. We will do some examples of both, but we will also dedicate a complete chapter to sample size calculations.

18.1.1 Simple example

Consider testing in the binomial example from chapter 16. We found that for testing $H_0 : p = 0.5$ versus $H_a : p > 0.5$ that rejecting if we got 8 or more heads on 10 coin flips yielded a Type I error rate of just over 5%. What is the power of this test? To do this, we need to evaluate power for a value of p under the alternative. This is probably best done with a plot. Figure 16.1 displays the probability of getting 8 or more heads on 10 coin flips for various values of p.

```
p = seq(0.5, .99, by = 0.01)
## recall that we have to subtract 1 to get 8 or more for the upper tail
## probabilities
power = pbinom(7, size = 10, prob = p, lower.tail = FALSE)
plot(p, power, type = "l", frame = FALSE, lwd = 2,
     cex.axis=1.3,col.axis="blue",cex.lab=1.3,
     xlab = "True success probability", ylab = "Power")
```

Thus, if we wanted to get 80% power, say, we would need `min(p[power >= .8])` which is 0.85. However, the true value of p is not under our control. So now, let us look at how the power changes as a function of the sample size. First, we need to find the rejection region for several values of n. Let us calculate the smallest number of heads for which you reject for several possible coin flips where we are strict about the 5% error rate (adding the +1 to the quantile).

```
nVals = seq(5, 100, by = 1)
rejectionRegion = qbinom(.95, size = nVals, prob = 0.5) + 1
plot(nVals, rejectionRegion, type = "l", frame = FALSE, lwd = 2,
     cex.axis=1.3,col.axis="blue",cex.lab=1.3,
     xlab = "Number of trials (n)",
     ylab = "Smallest number to reject")
```

Now let us calculate the power over both the value of n and the value of p under H_a.

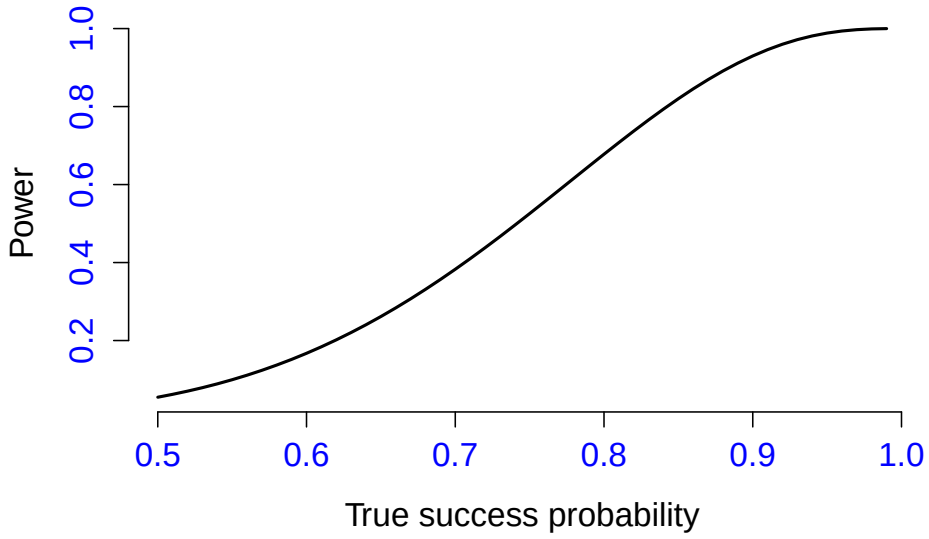


Figure 18.1: Power of rejecting the null hypothesis $H_0 : p = 0.5$ as a function of the true success probability, p, when observing 8 or more heads on 10 coin flips.

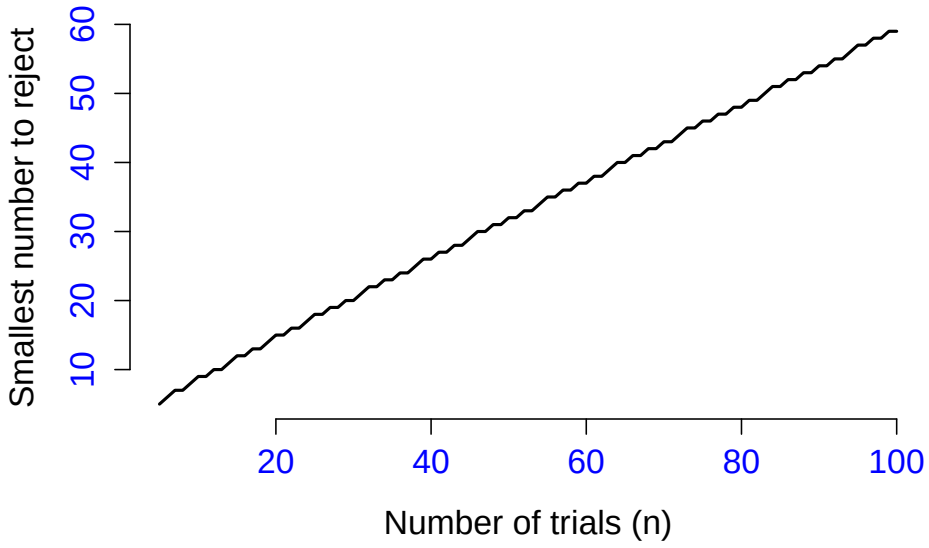


Figure 18.2: Smallest number of successes for which one rejects the null hypothesis $H_0 : p = 0.5$ as a function of the number of trials, n.

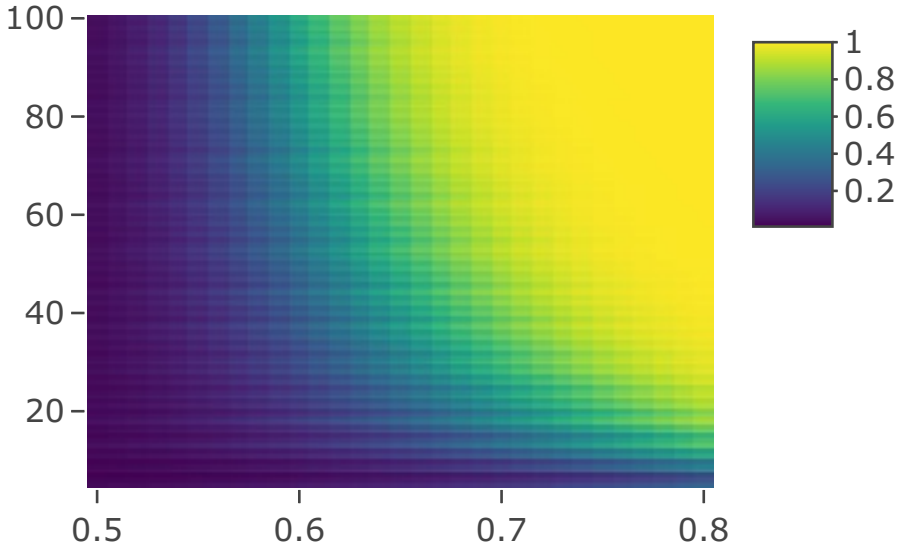
```
pSeq = seq(0.5, .8, by = 0.01)
power = sapply(pSeq, function(p)
  pbinom(rejectionRegion-1, size = nVals, prob = p, lower.tail = FALSE)
)
library(plotly)
```

```
plot1 = plot_ly(x = pSeq, y = nVals, z = power) %>% add_surface() %>%
  layout(title = "Power as a function of N and P",
         scene = list(
           xaxis = list(title = "p"),
           yaxis = list(title = "n"),
           zaxis = list(title = "Power")
           )
  )
```

```
plot1
```


```
plot2 = plot_ly(x = pSeq, y = nVals, z = round(power, 3)) %>% add_heatmap()
```

```
plot2
```



Both Figures @ref(fig:plotly_graphs) and @ref(fig:plotly_graphs_print) are interactive (in HTML), so you can play around with them to find the value of n for a given value of p that produces a specified power. So, for example, for a $p = 0.7$ under H_a we need a sample size of around 42 to achieve 80% power. Note that this problem is slightly unusual because of the discreteness of the binomial. For certain sample sizes, we can get closer to the 5% error rate and this results in a sawtooth looking function for power that is non monotonic. Thus, we can get higher power for smaller sample sizes. In this case at an $n = 42$ the power function is permanently above 0.8. Figure 18.3 displays the power to reject the null hypothesis (y-axis) as a function of the number of trials (x-axis), n, when the true success probability is $p = 0.7$.

```
plot(nVals, pbinom(rejectionRegion-1, size = nVals,
                   prob = 0.7, lower.tail = FALSE),
     type = "l", frame = FALSE, lwd = 2,
     cex.axis=1.3,col.axis="blue",cex.lab=1.3,
     xlab = "Sample size (n)",
     ylab = "Power")
abline(h = .8)
abline(v = 42)
```

Let us now cover power for continuous variables with Normal distributions.

18.2 Power calculation for Normal tests

Consider our previous example from Chapter 16 involving RDI. We were testing $H_0 : \mu = 30$ versus $H_a : \mu > 30$ where μ was the population mean respiratory

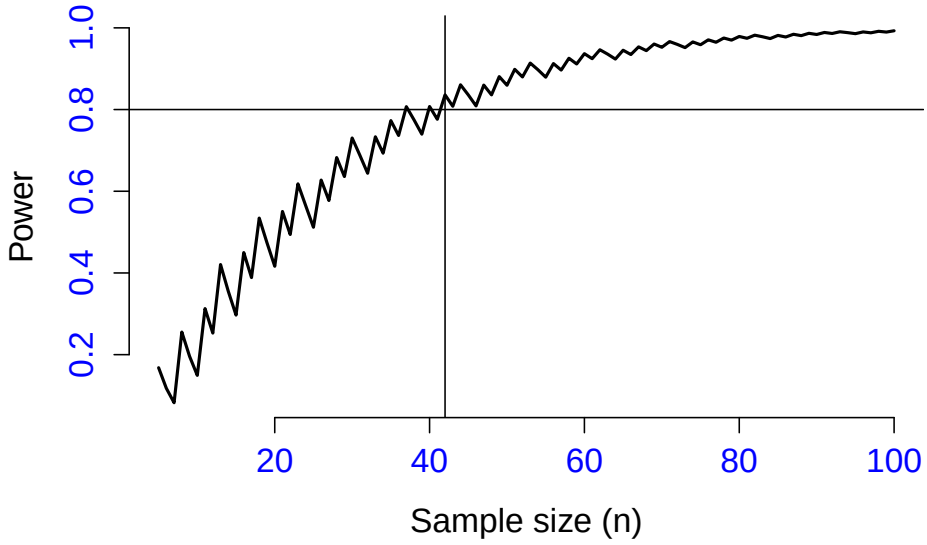


Figure 18.3: Power to reject the null hypothesis as a function of the number of trials, n, when the true success probability is $p = 0.7$.

disturbance index. Our test statistic was

$$Z = \frac{\bar{X} - 30}{\sigma/\sqrt{n}}$$

and we reject if $Z \geq Z_{1-\alpha}$. Assume that n is large and that Z is well approximated by a Normal distribution and σ is known. Let μ_a be a value of μ under H_a that we are assuming to be the true value for the calculations. Then consider:

$$\begin{aligned}
\text{Power} &= P\left(\frac{\bar{X} - 30}{\sigma/\sqrt{n}} > z_{1-\alpha} \mid \mu = \mu_a\right) \\
&= P\left(\frac{\bar{X} - \mu_a + \mu_a - 30}{\sigma/\sqrt{n}} > z_{1-\alpha} \mid \mu = \mu_a\right) \\
&= P\left(\frac{\bar{X} - \mu_a}{\sigma/\sqrt{n}} > z_{1-\alpha} - \frac{\mu_a - 30}{\sigma/\sqrt{n}} \mid \mu = \mu_a\right) \\
&= P\left(Z > z_{1-\alpha} - \frac{\mu_a - 30}{\sigma/\sqrt{n}} \mid \mu = \mu_a\right) .
\end{aligned}$$

Thus, we can relate our power to a standard Normal again. Suppose that we wanted to detect an increase in mean RDI of at least 2 events / hour (above 30). Assume normality and that the sample in question will have a standard

deviation of 4. What would the power be if we took a sample size of 16? We have that $Z_\alpha = 1.645$ and $\frac{\mu_a - 30}{\sigma/\sqrt{n}} = 2/(4/\sqrt{16}) = 2$ and therefore $P(Z > 1.645 - 2) = P(Z > -0.355) = 64\%$.

Of course, `R` will do the calculation directly for us. Specifically, our test statistic has mean $E[Z] = \sqrt{n}(\mu_a - \mu_0)/\sigma$ and variance 1 (check yourself!) under H_a . Thus, the probability that we reject is.

```
mu0 = 30; mua = 32; n = 16; sigma = 4
rprob<-pnorm(1.645,
       mean = sqrt(n) * (mua - mu0) / sigma ,
       sd = 1,
       lower.tail = FALSE)
round(rprob,digits=3)
```

```
[1] 0.639
```

Consider now a sample size calculation. What value of n would yield 80% power? That is, we want

$$0.80 = P\left(Z > z_{1-\alpha} - \frac{\mu_a - 30}{\sigma/\sqrt{n}} \mid \mu = \mu_a\right) .$$

Therefore, we want to set $z_{1-\alpha} - \frac{\mu_a - 30}{\sigma/\sqrt{n}} = z_{0.20}$ and solve for n yielding

$$n = \sigma^2 \left(\frac{z_{1-\alpha} - z_{.2}}{\mu_a - \mu_0}\right)^2 .$$

```
mu0 = 30; mua = 32; sigma = 4
ceiling(sigma^2 * (qnorm(.95) - qnorm(.2))^2 / (mua-mu0)^2)
```

```
[1] 25
```

We take the `ceiling` function as we want to round up. Thus, we need 25 subjects to get 80% power for these settings.

The general formula is for a one sided alternative of $H_a : \mu > \mu_0$

$$n = \left\lceil \sigma^2 \left(\frac{z_{1-\alpha} - z_\beta}{\mu_a - \mu_0}\right)^2 \right\rceil$$

where β is the Type II error rate (1 minus the power) and $\lceil \cdot \rceil$ is the ceiling function. One can use the same formula for a test of $H_a : \mu < \mu_0$, since the denominator is squared and the numerator works out to be $(z_\alpha - z_{1-\beta})^2 = (z_{1-\alpha} - z_\beta)^2$.

For a two sided alternative, consider that we reject for our RDI example if

$$\begin{aligned}\text{Power} &= P\left(\frac{\bar{X}-30}{\sigma/\sqrt{n}} > z_{1-\alpha/2} \text{ or } \frac{\bar{X}-30}{\sigma/\sqrt{n}} < z_{\alpha/2} \mid \mu = \mu_a\right) \\ &= P\left(\frac{\bar{X}-30}{\sigma/\sqrt{n}} > z_{1-\alpha/2} \mid \mu = \mu_a\right) + P\left(\frac{\bar{X}-30}{\sigma/\sqrt{n}} < z_{\alpha/2} \mid \mu = \mu_a\right) .\end{aligned}$$

Typically, one of these two probabilities is very small for an assumed μ_a. If μ_a is larger than 30, then the probability that we get a test statistic two standard deviations below 30 is small. The largest the smaller of these two probabilities can be is $\alpha/2$ as μ_a approaches μ_0 from one side or the other. Therefore, an underestimate of power throws out the smaller of the two and just calculates the one sided power using $\alpha/2$ instead of α. This underestimates power by a maximum of $\alpha/2$, which is typically not a relevant amount. The same strategy can be used for sample size calculations, since an underestimate of power gives a conservatively large n.

So, considering our sleep example, we would calculate the two sided sample size with:

```
mu0 = 30; mua = 32; sigma = 4
ceiling(sigma^2 * (qnorm(.975) - qnorm(.2))^2 / (mua-mu0)^2)
```

```
[1] 32
```

We will go into more detail in the next chapter, which is dedicated to sample size calculations.

18.3 Power for the t test

For smaller sample sizes where we would naturally use a t-test, we want to calculate the exact power. To do this, we would need to discuss the non-central t distribution, which is perhaps unnecessarily deep in the weeds for our purposes. Instead, let us demonstrate how the power can be calculated. For our RDI example, our one sided power based on the t-test would be:

$$
\begin{aligned}
\text{Power} &= P\left(\frac{\bar{X}-30}{S/\sqrt{n}} > t_{1-\alpha,n-1} \mid \mu = \mu_a\right) \\
&= P\left(\sqrt{n}(\bar{X}-30) > t_{1-\alpha,n-1}S \mid \mu = \mu_a\right) \\
&= P\left(\frac{\sqrt{n}(\bar{X}-30)}{\sigma} > t_{1-\alpha,n-1}\frac{S}{\sigma} \mid \mu = \mu_a\right) \\
&= P\left(\frac{\sqrt{n}(\bar{X}-\mu_a)}{\sigma} + \frac{\sqrt{n}(\mu_a-30)}{\sigma} > \frac{t_{1-\alpha,n-1}}{\sqrt{n-1}} \times \sqrt{\frac{(n-1)S^2}{\sigma^2}}\right) \\
&= P\left(Z + \frac{\sqrt{n}(\mu_a-30)}{\sigma} > \frac{t_{1-\alpha,n-1}}{\sqrt{n-1}}\sqrt{\chi^2_{n-1}}\right)
\end{aligned}
$$

where Z and χ^2_{n-1} are independent standard Normal and chi-squared (df $n-1$) random variables. (You will have to trust us on the independence.) An important point to emphasize is that this calculation only depends on n and the quantity

$$(\mu_a - \mu_0)/\sigma \, ,$$

which is referred to as the effect size. This is extremely beneficial since, as a unit free quantity, it has some hope of being consistently interpretable across experiments. One can thus feel more comfortable assuming an effect size over assuming both μ_a and σ.

The actual probability calculation above requires the aforementioned non-central t distribution. Instead let us use Monte Carlo for our RDI example.

```
nosim = 100000; n = 16; sigma = 4; mu0 = 30; mua <- 32
z = rnorm(nosim)
xsq = rchisq(nosim, df = n-1)
t = qt(.95, n-1)
mcprob<- mean(z + sqrt(n) * (mua - mu0) / sigma >
         t / sqrt(n - 1) * sqrt(xsq))
round(mcprob,digits=3)
```

```
[1] 0.607
```

Of course, we do not do this in practice. Instead, we user `power.t.test`. As mentioned before, we can just assume $\sigma = 1$ and specify the effect size.

```
power.t.test(n = 16, delta = (mua - mu0)/sigma,
             type = "one.sample",
             alt = "one.sided")
```

```
     One-sample t test power calculation
```

```
              n = 16
          delta = 0.5
             sd = 1
      sig.level = 0.05
          power = 0.6040329
    alternative = one.sided
```

Just to show you that you get the same result,

```
power.t.test(n = 16, delta = (mua - mu0),
             sd = sigma,
             type = "one.sample",
             alt = "one.sided")
```

```
     One-sample t test power calculation

              n = 16
          delta = 2
             sd = 4
      sig.level = 0.05
          power = 0.6040329
    alternative = one.sided
```

18.4 Discussion

Power is dependent on many factors. It is useful to use the Normal calculations to build our intuition in this regard. For example, power goes up as our Type I error goes up (it is easier to convict the guilty if we require less evidence). As the variance goes up, the power goes up (the higher the quality of the evidence, the more likely that we will convict the guilty). As the size of the effect gets larger, $\mu_a - \mu_0$, power goes up (we are more likely to convict those who are guilty). Finally, power goes up as the sample size goes up. Control of the sample size represents our primary method for ensuring that a study is well powered.

In addition, in a given setting one test statistic can be more powerful than another. For example, a test statistic that randomly throws out some of our data will be less powerful than one that uses all of the data. Of course, in many settings, there is no obviously optimal test and hence theoretical and simulation studies must be done to investigate different approaches.

Finally, it is worth mentioning that calculating power or sample size is dependent on the model being accurate. In extreme cases, one might be testing for a parameter that does not exist, such as the mean of a very heavy tailed distribution. In addition, even if the model is accurate, power calculations are dependent on unknown quantities, like the effect size. If estimates are used, it

is important to factor the error of estimating these quantities into the power calculations.

18.5 Problems

Problem 1. Consider the case when we conduct an exact binomial two sided test for proportions

$$H_0 : p = 0.4 \quad \text{vs.} \quad H_A : p \neq 0.4 \, .$$

a. For a sequence of sample sizes $n = 10, 11, \ldots, 100$ calculate the rejection region for the test; consider only tests that have less than 0.05 α levels and pick the one closest to 0.05.
b. For every rejection region calculate and display the α-level of the test.
c. For each n calculate the power function $P(n, p_a)$, where $p_a \in 0.01, 0.02, \ldots, 0.99$.

Problem 2. For the previous problem construct an interactive graph of the power function as a function of the two variables $P(n, p_a)$. Interpret the graph.

Problem 3. Consider the same set up as in the first problem, but consider the asymptotic test instead of the exact binomial test.

a. Construct the rejection region for the test and explain why, in theory, all tests are $\alpha = 0.05$-level.
b. Show that none of these tests is exactly a 0.05-level tests.
c. For each n calculate the power function $P(n, p_a)$, where $n \in 10, 11, \ldots, 100$ and $p_a \in 0.01, 0.02, \ldots, 0.99$.

Problem 4. Compare the power and α levels of the exact and asymptotic tests. What do you conclude?

Problem 5. Consider a two-sided t-test for testing

$$H_0 : \mu = \mu_0 \quad \text{vs.} \quad H_A : \mu \neq \mu_0 \, .$$

a. Obtain the power to reject the null hypothesis, $P(n, f)$, where $f = (\mu_a - \mu_0)/\sigma$ where $n = 10, 11, \ldots, 100$ and $f = 0.1, 0.11, \ldots, 1.5$. Use the function `power.t.test`.
b. Build an interactive plot of $P(n, f)$ in `R` and interpret it.

Problem 6. Same problem, but use a one-sided alternative instead of the two-sided alternative.

Problem 7. Consider the square root of `rdi4p` and estimate the effect size of gender on `rdi4p` in the entire SHHS data.

a. Using a nonparametric bootstrap estimate the effect size distribution.

b. For an $\alpha = 0.05$-level calculate the distribution of the power by applying the `power.t.test` function to the effect sizes obtained via the nonparametric bootstrap in the previous point.
c. Calculate and compare the distribution of power, as obtained in the previous point, as a function of different α levels.

Problem 8. Consider testing for equality in the mean of square root of `rdi4p` between men and women. For a sample of 10 men and 10 women taken at random from the SHHS population conduct an $\alpha = 0.05$ t-test for equality of means against the two-sided alternative.

a. Repeat this experiment 10000 times and record how many times the null hypothesis was rejected.
b. Calculate the theoretical power in a two-sided t-test of size $\alpha = 0.05$ to reject the null hypothesis assuming the effect size observed in the overall dataset and a sample size of $n = 10$.
c. At what level α is the observed proportion of rejections equal with the theoretical power?

Problem 9. The effect size in t-tests is defined as the function

$$f = \frac{\mu_a - \mu_0}{\sigma} .$$

a. Calculate the power under the "local alternative"

$$f = \frac{\mu_a - \mu_0}{\sigma/\sqrt{n}} .$$

b. How do you interpret the "local" versus the "global" power?

Problem 10. Here we would like to compare the theoretical and observed power using simulations.

a. Simulate 100000 times $n = 10$ $N(0,1)$ random variables and $n = 10$ $N(\mu,1)$ variables, where $\mu = 0, 0.01, \ldots, 1$ and calculate the percent of rejections of a size $\alpha = 0.05$ two sided t-test; hint, you can simulate $N(\mu,1)$ simultaneously using

```
nsim=100000
mu=seq(0,1,by=0.01)
sim_vec=mu+rnorm(nsim)
```

b. Compare the proportion of rejections with the theoretical probability of rejection.

Problem 11. Same problem description as Problem 10, but vary the number of observations per group from $n = 10$ to 500 in increments of 10. What do you observe in terms of agreement between the theoretical and observed power when n increases?

Problem 12. Same problem description as Problem 10, but change the data generating distribution from Normal to t with 2, 3, and 5 degrees of freedom, respectively.

Problem 13. Same problem description as Problem 10, but change the data generating distribution to a double exponential

$$f(x) = \frac{1}{2}\exp(-|x-\mu|) \; .$$

Problem 14. Consider a two sided t-test for testing

$$H_0 : \mu = \mu_0 \quad \text{vs.} \quad H_A : \mu \neq \mu_0$$

and sample sizes $n = 10, 11, \ldots, 100$, and test sizes $\alpha = 0.01, 0.02, \ldots, 0.99$. For two effect sizes $f = 0.1$ and $f = 0.5$:

a. Obtain the power to reject the null hypothesis, $P(n, \alpha, f)$. Use the function `power.t.test`.
b. Build two interactive plots of $P(n, \alpha, f)$ in `R` and interpret them.
c. Explain the relationship between power, sample size, and test size.

Problem 15. In a discussion with a collaborator you are told: we use a new, more powerful test because we obtain more statistically significant results.

a. Explain, in detail, to your collaborator why his or her statement is completely incorrect, but in a way that you keep the door open for future collaborations.
b. Provide an example of the test that is the most powerful in the world against any alternative and in any circumstance.
c. How would you propose to proceed with the investigation about comparing the previous test and the new, apparently, more "significant result producing" test.

Chapter 19

Sample size calculations

This chapter covers the following topics

- Sample size calculation for continuous data
- Sample size calculation for binary data
- Sample size calculation using exact tests
- Sample size calculation with preliminary data

19.1 Introduction

Sample size calculation is often treated unjustly in Biostatistics textbooks. It is extremely important, but, inexplicably, considered too "boring" or "standard." Moreover, in practice, many Biostatistician positions depend on the mastery of this subject. Far from being easy, sample size calculation is subject not only to science, but to cost and policy constraints. To highlight these limitations many an experienced Biostatistician would state in private that "sample size is the total budget divided by the cost of the trial per person." Here we try to provide Biostatistical insight and examples that both highlight *how* to calculate sample sizes based on previous knowledge and *what* the potential pitfalls might be. We take the view that in their career most Biostatisticians encounter multiple cases where sample size calculation is crucial and he or she may be the only authority to talk about it at their research room table. Imagine, for example, the design of a clinical trial for stroke where a new treatment is considered versus standard of care. Suppose that we consider a two-arm randomized clinical trial and that the follow-up time is three years. After some in-depth cost calculations we know that it will cost about $\$50,000$ to enroll, track, and treat each subject in the treatment arm. Therefore, the difference in cost between 100 subjects and 200 subjects in the treatment arm is exactly $\$5,000,000$. Do we have your attention now? If not and you care a lot more about methodological problems than hard, cold ca$$,

we would like to say that, with the exception of standard clinical trials, sample size calculations are often far from standard. Consider, for example, sample size calculations for testing for a regression coefficient in a linear regression, an interaction term in a generalized linear model, or a fixed effect parameter in a longitudinal mixed effects model. Here we will provide enough details to explain the general area of sample size calculation with emphasis on two-sample tests. However, this is a large scientific area that requires domain-specific knowledge and thinking. We have found the monograph by (Chow, Shao, and Wang 2008) and the comprehensize associated `R` package `TrialSize` (Zhang et al. 2013) to be excellent starting points for learning more about sample size calculation.

19.2 Sample size calculation for continuous data

We have already seen how to obtain confidence intervals and conduct tests for the mean of two populations. The sample size calculation is turning the hypothesis testing problem around and asking the question: "given an effect size (not yet defined) what is the sample size that will be detected with high probability (power)." Consider the case when we observe n subjects both in the first and second group for a total sample size of $2n$. Denote by $X_1, \ldots, X_n \sim N(\mu_1, \sigma^2)$ and $Y_1, \ldots, Y_n \sim N(\mu_2, \sigma^2)$ the outcomes of the experiment in the first and second group, respectively. We consider the case of equal and known variances in the two groups, but similar approaches can be used for unequal variances. We are interested in testing the null hypothesis

$$H_0 : \mu_2 - \mu_1 = 0 \quad \text{versus} \quad H_A : \mu_2 - \mu_1 > 0 \, .$$

19.2.1 Sample size calculation based on the Normal approximation

Here we consider the one-sided alternative because the math is a little easier to follow, but we will discuss the two-sided alternative, as well. Recall that

$$\overline{Y}_n - \overline{X}_n \sim N(\mu_2 - \mu_1, 2\sigma^2/n) \, .$$

Here the 2 in front of σ^2 appears because the variances of $\overline{Y}_n$ and $\overline{X}_n$ add up. The rejection of the null hypothesis happens if $\overline{Y}_n - \overline{X}_n > C$, where the constant C is determined such that the probability of rejecting the null, if the null is true, is small. This probability is denoted by α and is referred to as the size of the test. Under the null hypothesis, $\overline{Y}_n - \overline{X}_n \sim N(0, 2\sigma^2/n)$ and the constant C can be obtained from the formula

$$\alpha = P(\overline{Y}_n - \overline{X}_n > C | \mu_1 = \mu_2) = P\left\{ \frac{\sqrt{n/2}(\overline{Y}_n - \overline{X}_n)}{\sigma} > \frac{\sqrt{n/2}C}{\sigma} \right\} \, .$$

Because $\sqrt{n/2}(\overline{Y}_n - \overline{X}_n)/\sigma \sim N(0,1)$ it follows that

$$\frac{\sqrt{n/2}C}{\sigma} = z_{1-\alpha} \Rightarrow C = \frac{z_{1-\alpha}\sigma}{\sqrt{n/2}} ,$$

where $z_{1-\alpha}$ denotes the $1-\alpha$ quantile of the standard Normal distribution. The idea of the sample size calculation is to find the group sample size, n, that ensures a large probability of rejecting the null when the alternative is true $\mu_2 > \mu_1$. This probability is called the power of the test and is denoted by $1-\beta$. The rejection probability is

$$\begin{aligned} 1-\beta &= P\left(\overline{Y}_n - \overline{X}_n > \frac{z_{1-\alpha}\sigma}{\sqrt{n/2}} | \mu_2 > \mu_1\right) \\ &= P\left(\overline{Y}_n - \overline{X}_n - \mu_2 + \mu_1 > \frac{z_{1-\alpha}\sigma}{\sqrt{n/2}} - \mu_2 + \mu_1 | \mu_2 > \mu_1\right) . \end{aligned}$$

By dividing both the left and right side of the inequality by $\sigma/\sqrt{n/2}$ we obtain

$$1-\beta = P\left\{\frac{\sqrt{n/2}(\overline{Y}_n - \overline{X}_n - \mu_2 + \mu_1)}{\sigma} > z_{1-\alpha} - \frac{\sqrt{n/2}(\mu_2 - \mu_1)}{\sigma} | \mu_2 > \mu_1\right\} .$$

Because

$$\frac{\sqrt{n/2}(\overline{Y}_n - \overline{X}_n - \mu_2 + \mu_1)}{\sigma} \sim N(0,1)$$

it follows that

$$z_{1-\alpha} - \frac{\sqrt{n/2}(\mu_2 - \mu_1)}{\sigma} = z_\beta \Rightarrow \sqrt{n/2} = (z_{1-\alpha} + z_{1-\beta})\frac{\sigma}{\mu_2 - \mu_1} ,$$

where we used the fact that $z_{1-\beta} = -z_\beta$ due to the symmetry of the standard Normal distribution. Thus, the sample size needed for one group is

$$2(z_{1-\alpha} + z_{1-\beta})^2 \frac{\sigma^2}{(\mu_2 - \mu_1)^2}$$

and the total sample size for the two groups is

$$4(z_{1-\alpha} + z_{1-\beta})^2 \frac{\sigma^2}{(\mu_2 - \mu_1)^2} .$$

This formula shows what the ingredients are for conducting the sample size calculation. First, we need the size of the test, α, and the power of the test, $1-\beta$. Suppose that we fix the size at $\alpha = 0.05$ and the power of detecting the alternative at $1-\beta = 0.9$. Then the first part of the formula can be calculated in `R` as follows:

```
#Size
alpha=0.05
#One minus power
beta=0.1
#Corresponding quantiles of the standard N(0,1)
z_1_minus_alpha=qnorm(1-alpha)
z_1_minus_beta=qnorm(1-beta)
#Calculate the multiplier in front of the fraction
multiplier<-4*(z_1_minus_alpha+z_1_minus_beta)^2
```

This indicates that the constant $4(z_{1-\alpha} + z_{1-\beta})^2$ is equal to 34.255 for these choices of size and power of the test. The other factor that the sample size depends on is the effect size

$$f = \frac{\mu_2 - \mu_1}{\sigma} = \frac{\text{Mean of group 2} - \text{Mean of group 1}}{\text{Standard deviation}} .$$

Thus, the sample size can be written more compactly as

$$\frac{M}{f^2} ,$$

where $M = 4(z_{1-\alpha} + z_{1-\beta})^2$ and f is the effect size. Note that the square of the effect size is in the denominator, indicating that smaller effect sizes will have a much larger effect on sample sizes. Let us consider $f = 0.5$, which is considered to be a large effect size in Biostatistics. Then $1/f^2 = 4$ and the total sample size is $4 \times 34.255 = 137.02$, rounded up to 138; that is, 69 subjects per group. However, if the effect size is $f = 0.3$, which is considered moderate, then $1/f^2 = 11.111$ and we would need $11.111 \times 34.255 = 380.61$ subjects, rounded up to 381; that is, 191 subjects per group. Thus, going from a large to a moderate effect size is associated with a large increase in sample size and cost of the experiment.

For the single sample design (when we have a single sample and conduct a test about the mean of the sample), the same derivations can be used. In this case it is easy to show that the sample size formulas simply use half the sample size for the corresponding two-sample design.

19.2.2 Sample size calculation based on the t approximation

If a t-test is used instead of the asymptotic Normal one, there are no explicit formulas, though the sample size can still be calculated. Let us dive a little deeper into what needs to be done in the t-test case. Note that we still need to solve the equation

$$\sqrt{n/2} = (t_{1-\alpha,2n-2} + t_{1-\beta,2n-2})\frac{\sigma}{\mu_2 - \mu_1} ,$$

where $t_{\alpha,k}$ is the α quantile of the t distribution with k degrees of freedom. The reason we have $k = 2n - 2$ degrees of freedom is because we are using a pooled estimator of the standard deviation. In the Normal case the two quantiles in the right hand side of the equation do not depend on n and we obtain n by squaring both sides of the equation. In the case of the t test this cannot be done because of the slight dependence of the quantiles on n. However, the equation can be solved numerically. Indeed, consider a grid of sample sizes, n, and check when the equality above is met. Another approach could be to find the zero of the function

$$f(x) = \sqrt{x/2} - (t_{1-\alpha,2x-2} + t_{1-\beta,2x-2}) \frac{\sigma}{\mu_2 - \mu_1} = \sqrt{x/2} - \frac{t_{1-\alpha,2x-2} + t_{1-\beta,2x-2}}{f},$$

where f is the effect size. This function is strictly increasing because $\sqrt{x/2}$ is increasing and $t_{1-\alpha,2x-2}$ and $t_{1-\beta,2x-2}$ are both decreasing. Thus, the function $f(\cdot)$ crosses 0 only once at the sample size.

When x is large (say above 50) there is very little difference between $t_{1-\alpha,2x-2}$ and $z_{1-\alpha}$ and $z_{1-\beta}$. In general, small effect sizes will require larger samples, which is exactly when the Normal approximation will work best. Larger effect sizes will require smaller sample sizes, and some differences between the t and Normal approximations of the test statistic distribution may provide different results. Let us visually investigate what happens to the sample size for two effect sizes, $f = 0.3$ and 0.5.

```
#Set the size and power of the test
alpha=0.05
beta=0.1
#Set the effect sizes
f1=0.3
f2=0.5
#
grid=seq(20,1000,by=0.1)
f_grid_1<-sqrt(grid/2)-(qt(1-alpha,2*grid-2)+qt(1-beta,2*grid-2))/f1
f_grid_2<-sqrt(grid/2)-(qt(1-alpha,2*grid-2)+qt(1-beta,2*grid-2))/f2
```

Figure 19.1 displays the function $\{x, f(x)\}$, where x is the sample size and $f(x)$ was described above. Th ezero of the $f(\cdot)$ function is the sample size for detecting the corresponding effect size. The blue line corresponds to the effect size $f = 0.3$ and the orange line corresponds to $f = 0.5$.

```
plot(grid,f_grid_1,type="l",col="blue",lwd=3,ylim=c(-10,15),
     xlab="Sample size",ylab="f(x)",
     bty="l",cex.axis=1.3,cex.lab=1.3,col.lab="blue",
     col.axis="blue",main=NULL)
lines(grid,f_grid_2,col="orange",lwd=3)
abline(h=0,lwd=3)
```

Going from the plot to obtaining the actual sample size can be done directly

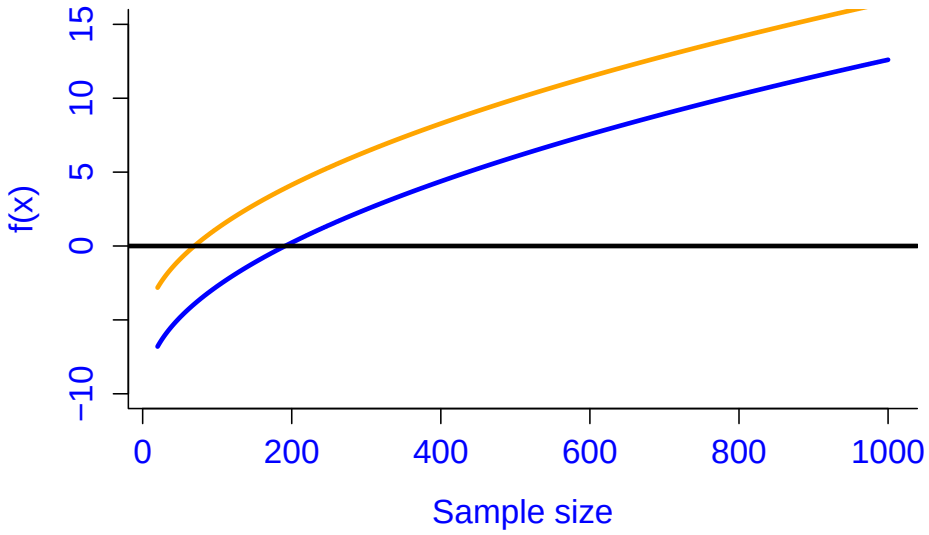


Figure 19.1: Function $f(x)$ (y-axis) whose zero represents the exact sample size for a t-test at the sample size (x-axis).

using the following R code:

```
#Obtain the sample size for the t-test
sample_size_f1<-grid[which.min(abs(f_grid_1))]
sample_size_f2<-grid[which.min(abs(f_grid_2))]
```

Thus, when using the t-test approach with an effect size $f = 0.5$ we require $n = 69.3$, which rounded up is 70 subjects per group, one more than when using the Normal approximation. Similarly, when using an effect size $f = 0.3$ we require $n = 69.3$, rounded up to 192, again one more subject per group. These very small differences are why in practice we can simply use the explicit Normal approximation formula. If we want to get more refined evaluations of the sample size, then we can increase the density of the grid.

19.2.3 Two-sided alternative hypothesis

We have investigated the case of a one-sided alternative because calculations are slightly easier to explain. In practice, one often encounters two-sided alternative hypotheses of the type

$$H_0 : \mu_2 - \mu_1 = 0 \quad \text{versus} \quad H_A : \mu_2 - \mu_1 \neq 0 \, .$$

For example, in clinical trials the Food and Drug Administration (FDA) would typically not accept a one-sided alternative hypothesis (no effect of drug versus a positive effect) and would insist on a two-sided alternative (no effect of drug

versus positive or adverse effect of drug). The net effect is that the two-sided hypotheses are harder to reject and/or require larger sample sizes to reject the null hypothesis. Using the same example as before in the case of a two-sided hypothesis we reject when $|\overline{Y}_n - \overline{X}_n|$ is large compared with the one-sided hypothesis where we reject if $\overline{Y}_n - \overline{X}_n$ is large. Using similar reasoning with the one-sided hypothesis it follows that we reject if

$$|\overline{Y}_n - \overline{X}_n| > \frac{z_{1-\alpha/2}\sigma}{\sqrt{n/2}} .$$

This formula shows that it is harder to reject because $z_{1-\alpha/2} > z_{1-\alpha}$. Calculating the sample size follows a very similar recipe, but it has a small twist. Indeed, as before,

$$\begin{aligned} 1-\beta &= P\left(|\overline{Y}_n - \overline{X}_n| > \frac{z_{1-\alpha/2}\sigma}{\sqrt{n/2}} | \mu_2 \neq \mu_1\right) \\ &= P\left(\overline{Y}_n - \overline{X}_n > \frac{z_{1-\alpha/2}\sigma}{\sqrt{n/2}} | \mu_2 \neq \mu_1\right) + P\left(\overline{Y}_n - \overline{X}_n < -\frac{z_{1-\alpha/2}\sigma}{\sqrt{n/2}} | \mu_2 \neq \mu_1\right) . \end{aligned}$$

This formula throws a monkey wrench into our explicit algebra, as the right hand side of the equation contains the sum of two probabilities and the sample size solution cannot be obtained by explicitly solving for n. Instead, the solution is based on the observation that one of the two terms in the right hand side of the equation is much smaller than the other. Indeed, consider the case when $\mu_2 > \mu_1$. In this case the first term is equal to

$$\begin{aligned} &P\left\{\frac{\sqrt{n/2}(\overline{Y}_n - \overline{X}_n - \mu_2 + \mu_1)}{\sigma} > z_{1-\alpha/2} - \frac{\sqrt{n/2}(\mu_2-\mu_1)}{\sigma} | \mu_2 > \mu_1\right\} = \\ &1 - \Phi\left\{z_{1-\alpha/2} - \frac{\sqrt{n/2}(\mu_2-\mu_1)}{\sigma} | \mu_2 > \mu_1\right\} , \end{aligned}$$

where $\Phi(\cdot)$ is the cdf of a standard normal distribution. Similarly, the second term is equal to

$$\begin{aligned} &P\left\{\frac{\sqrt{n/2}(\overline{Y}_n - \overline{X}_n - \mu_2 + \mu_1)}{\sigma} > -z_{1-\alpha/2} - \frac{\sqrt{n/2}(\mu_2-\mu_1)}{\sigma} | \mu_2 > \mu_1\right\} = \\ &\Phi\left\{-z_{1-\alpha/2} - \frac{\sqrt{n/2}(\mu_2-\mu_1)}{\sigma} | \mu_2 > \mu_1\right\} . \end{aligned}$$

When $\mu_2 > \mu_1$ we have $\sqrt{n/2}(\mu_2 - \mu_1)/\sigma > 0$, which indicates that

$$\Phi\left\{-z_{1-\alpha/2} - \frac{\sqrt{n/2}(\mu_2-\mu_1)}{\sigma} | \mu_2 > \mu_1\right\} < \Phi(-z_{1-\alpha/2}) = \alpha/2 ,$$

which is small. In most cases the term is much, much smaller than $\alpha/2$, and increases with n and with the effect size. To get a better idea let us calculate these numbers for a few cases.

For an effect size of 0.3 and a sample size of $n = 10$ the probability is 0.0043, while for $n = 20$ the probability is 0.0018. This is why this term can be ignored in calculations and we obtain

$$1 - \beta = 1 - \Phi\left\{ z_{1-\alpha/2} - \frac{\sqrt{n/2}(\mu_2 - \mu_1)}{\sigma} | \mu_2 > \mu_1 \right\} ,$$

which, using the same calculations as for the one-sided test, results in a total sample size for the two groups of

$$4(z_{1-\alpha/2} + z_{1-\beta})^2 \frac{\sigma^2}{(\mu_2 - \mu_1)^2} .$$

The only difference is that the term $z_{1-\alpha/2}$ replaces $z_{1-\alpha}$. Thus, for the same effect size a two-sided test would require

$$100 \frac{(z_{1-\alpha/2} + z_{1-\beta})^2 - (z_{1-\alpha} + z_{1-\beta})^2}{(z_{1-\alpha} + z_{1-\beta})^2} \%$$

larger sample size. For $\alpha = 0.05$ and $\beta = 0.9$ this is a 22.7% increase in the sample size. A similar reasoning holds for the case when $\mu_2 < \mu_1$, while the solution for t-tests can be obtained solving similar equations. We skip these details, though we will show later how to use `R` to conduct calculations in a variety of scenarios.

19.2.4 Different group variances

In some situations we may not want to assume that the two variances are equal. To understand the scenario we consider the one-sided normal test, with the two-sided Normal and t-tests being treated as described above. Denote by $X_1, \ldots, X_n \sim N(\mu_1, \sigma_X^2)$ and $Y_1, \ldots, Y_n \sim N(\mu_2, \sigma_Y^2)$ the outcomes of the experiment in the first and second group, respectively. We are interested in testing the null hypothesis

$$H_0 : \mu_2 - \mu_1 = 0 \quad \text{versus} \quad H_A : \mu_2 - \mu_1 > 0 .$$

Recall that

$$\overline{Y}_n - \overline{X}_n \sim N\{\mu_2 - \mu_1, (\sigma_X^2 + \sigma_Y^2)/n\} .$$

The rejection of the null hypothesis happens if $\overline{Y}_n - \overline{X}_n > C$, where the constant C is determined such that the probability of rejecting the null, if the null is true,

is small. Under the null hypothesis $\overline{Y}_n - \overline{X}_n \sim N\{0, (\sigma_X^2 + \sigma_Y^2)/n\}$ The constant C can be obtained from the formula

$$\alpha = P(\overline{Y}_n - \overline{X}_n > C | \mu_1 = \mu_2) = P\left\{ \frac{\sqrt{n}(\overline{Y}_n - \overline{X}_n)}{\sqrt{\sigma_X^2 + \sigma_Y^2}} > \frac{\sqrt{n}C}{\sqrt{\sigma_X^2 + \sigma_Y^2}} \right\} .$$

It follows that

$$\frac{\sqrt{n/2}C}{\sigma} = z_{1-\alpha} \Rightarrow C = \frac{z_{1-\alpha}\sqrt{\sigma_X^2 + \sigma_Y^2}}{\sqrt{n}} .$$

Following the exact same calculations as above it follows that the sample size for one group is

$$(z_{1-\alpha} + z_{1-\beta})^2 \frac{\sigma_X^2 + \sigma_Y^2}{(\mu_2 - \mu_1)^2}$$

and the total sample size for the two groups is

$$2(z_{1-\alpha} + z_{1-\beta})^2 \frac{\sigma_X^2 + \sigma_Y^2}{(\mu_2 - \mu_1)^2} .$$

The only difference from the equal variance case is that $2\sigma^2$ is replaced by $\sigma_X^2 + \sigma_Y^2$. For the two-sided test $z_{1-\alpha}$ is replaced by $z_{1-\alpha/2}$ and for the t-test we need to solve the problem numerically. As we have seen, the t-test calculation either agrees with the Normal calculation or requires one more subject per group. This is the reason why some software will add one or two subjects per group to the formulas above.

19.2.5 Different group sizes

While less common, there are situations when the number of subjects in the two groups are different. In this case it is common to set the sample size for group 1, say n, and to say that the group 2 will have rn, where r is the ratio of sample size in group 2 relative to group 1. Thus, the total sample size in the two groups will be $n(1 + r)$. We will explain the calculations in the one-sided hypothesis Normal test with unequal variances and unequal sample sizes. Denote by $X_1, \ldots, X_n \sim N(\mu_1, \sigma_X^2)$ and $Y_1, \ldots, Y_{rn} \sim N(\mu_2, \sigma_Y^2)$ the outcomes of the experiment in the first and second group, respectively. As before,

$$\overline{Y}_{rn} - \overline{X}_n \sim N(\mu_2 - \mu_1, (\sigma_X^2 + \sigma_Y^2/r)/n) .$$

From here on all calculations are the same where σ^Y is replaced by σ_Y^2/r, which leads to a sample size for the first group equal to

$$(z_{1-\alpha} + z_{1-\beta})^2 \frac{\sigma_X^2 + \sigma_Y^2/r}{(\mu_2 - \mu_1)^2}$$

and a total sample size of

$$(z_{1-\alpha} + z_{1-\beta})^2 \frac{\sigma_X^2 + \sigma_Y^2/r}{(\mu_2 - \mu_1)^2}(1 + r) \ .$$

Note how for equal sample sizes, $r = 1$, we obtain the sample size formulas from the previous section.

Despite the mathematical formulas lavishly displayed here, sample size calculation is not a perfect science. Indeed, one needs the effect size, which is typically obtained from previous study, experience, and negotiation with regulatory agencies. Often, however, the required sample size is driven more by total costs and feasibility (e.g., one cannot have more than 20 rats in a particular experiment) than by Biostatistical computations. It is perfectly fair to reverse the sample calculation system and provide the power expected for a cost-restricted sample size calculation.

19.2.6 R code

Now that we understand where all the sample size calculations for equality in mean are coming from, we can forget everything, and focus on R. One function in R that calculates power in R for a variety of scenarios is `power.t.test`. To get more information about the function simply type

```
?power.t.test
```

Here it is how to conduct the same calculations described above for the case of two-sample testing with a one sided alternative

```
power.t.test(power = .90, sig.level=0.05, delta = 0.3,
             sd=1, alternative = "one.sided")
```

```
     Two-sample t test power calculation

              n = 190.9879
          delta = 0.3
             sd = 1
      sig.level = 0.05
          power = 0.9
    alternative = one.sided

NOTE: n is number in *each* group
```

The result provides virtually the same result as the one we reported for an effect size $f = 0.3$ with the same size and power. The R function uses slightly different notation than in the previous sections, where `delta` stands for the difference in means $\mu_2 - \mu_1$ and `sd` stands for σ. Here `sd` is actually redundant

for those cases when we want to input separately the difference in means and the standard deviation. If we do not specify `sd` then it is set to the default of 1 and the parameter `delta` is the difference in means, $\mu_2 - \mu_1$, and the effect size, f. The `power=0.90` and `alternative = "one.sided"` statements are self-explanatory, while the size of the test is specified as `sig.level=0.05`. Getting the two-sided hypothesis sample size is as simple as

```
power.t.test(power = .90, sig.level=0.05, delta = 0.3,
             sd=1, alternative = "two.sided")
```

```
     Two-sample t test power calculation

              n = 234.4628
          delta = 0.3
             sd = 1
      sig.level = 0.05
          power = 0.9
    alternative = two.sided

NOTE: n is number in *each* group
```

Unfortunately, the `power.t.test` does not handle unequal variances or sample sizes in the two groups. However, because we know the meaning of the sample size formulas, it is straightforward to trick the function to calculate sample size for the case when the number of subjects is unbalanced and the variances may be unequal. Consider, for example, the case when $\sigma_X^2 = 1$, $\sigma_Y^2 = 4$ and we want twice as many subjects in group 2 versus group 1, that is $r = 2$ and a difference in means of 0.3. Let us revisit the formula for sample size, which contains the effect size in the form

$$\frac{\mu_2 - \mu_1}{\sqrt{\sigma_X^2 + \sigma_Y^2/r}} .$$

Thus, *we can treat* $\sqrt{\sigma_X^2 + \sigma_Y^2/r}$ *as a "pooled" standard error and proceed with the calculations as if we have equal variances.* This will provide the sample size for group 1, while the sample size for group 2 will be the sample size for group 1 multiplied by r, which, in our case, is 2.

```
#Set up the variance parameters
var_1=1
var_2=4
ratio=2
#Calculate the variance for unequal variances and unequal sample sizes
pooled_var=var_1+var_2/ratio

#This ensures that we get the correct effect size
pooled_sd=sqrt(pooled_var)
```

```
#Calculate the sample size of group 1
pwr<-power.t.test(power = 0.90, sig.level=0.05,
                  delta = 0.3, sd=pooled_sd,
                  alternative = "two.sided")
#Power size of group 1
n1<-round(pwr$n,digits=2)

#Power size of group 2
n2<-ratio*n1

#Total sample size
n_total<-n1+n2
```

Thus, the sample size for group 1 is 701.46 and for group 2 is 1402.92 with a total sample size of 2104.38.

19.3 Sample size calculation for binary data

We now focus on the two-sample problem for binary data with equal group sample sizes. Denote by $X_1, \ldots, X_n \sim \text{Bernoulli}(p_1)$ and $Y_1, \ldots, Y_n \sim \text{Bernoulli}(p_2)$. We would like to test

$$H_0 : p_2 - p_1 = 0 \quad \text{versus} \quad H_A : p_2 - p_1 > 0 \, .$$

There are two approaches for calculating sample size, depending on how the variance of the difference in means is calculated. In Chow, Shao, and Wang (2008) these two methods are called conditional and unconditional methods, respectively. We go through both derivations. We show that the conditional approach always requires a larger sample size given the true underlying success probabilities in the two groups, though the differences are small.

19.3.1 Two-sample test for proportion using the conditional approach

Just as in the continuous case, we reject the null if the difference $\overline{Y}_n - \overline{X}_n > C$, where C is a constant that ensures that the α level of the test is preserved. Denote by $p = (p_1 + p_2)/2$. More precisely,

$$\alpha = P(\overline{Y}_n - \overline{X}_n > C | p_1 = p) \, .$$

Since asymptotically

$$\overline{Y}_n - \overline{X}_n \sim N\left\{p_2 - p_1, \frac{p_2(1-p_2)}{n} + \frac{p_1(1-p_1)}{n}\right\}$$

it follows that under the null distribution $\overline{Y}_n - \overline{X}_n \sim N(0, 2p(1-p)/n)$. Therefore, the previous equality can be rewritten as

$$\alpha = P\left\{ \frac{\sqrt{n}(\overline{Y}_n - \overline{X}_n)}{\sqrt{2p(1-p)}} > \frac{\sqrt{n}C}{\sqrt{2p_1(1-p_1)}} | p_1 = p \right\} ,$$

indicating that

$$\frac{\sqrt{n}C}{\sqrt{2p(1-p)}} = z_{1-\alpha} \Rightarrow C = z_{1-\alpha}\sqrt{\frac{2p(1-p)}{n}} .$$

Thus, the power to detect $p_2 > p_1$ is

$$1 - \beta = P\left(\overline{Y}_n - \overline{X}_n - p_2 + p_1 > z_{1-\alpha}\sqrt{\frac{2p(1-p)}{n}} - p_2 + p_1 | p_2 > p_1 \right) .$$

Under the alternative, the mean of $\overline{Y}_n - \overline{X}_n - p_2 + p_1$ is 0 and the variance is $v(p_1, p_2)/n$, where

$$v(p_1, p_2) = p_1(1-p_1) + p_2(1-p_2) .$$

Therefore

$$1-\beta = P\left(\frac{\sqrt{n}(\overline{Y}_n - \overline{X}_n - p_2 + p_1)}{\sqrt{v(p_1,p_2)}} > z_{1-\alpha}\sqrt{\frac{2p(1-p)}{v(p_1,p_2)}} - \frac{\sqrt{n}(p_2-p_1)}{\sqrt{v(p_1,p_2)}} | p_2 > p_1 \right) .$$

Because, asymptotically, $\sqrt{n}(\overline{Y}_n - \overline{X}_n - p_2 + p_1)/\sqrt{v(p_1,p_2)} \sim N(0,1)$, it follows that

$$-z_{1-\beta} = z_{1-\alpha}\sqrt{\frac{2p(1-p)}{v(p_1,p_2)}} - \frac{\sqrt{n}(p_2-p_1)}{\sqrt{v(p_1,p_2)}} ,$$

which leads to the sample size per group

$$n = \frac{\left\{ z_{1-\alpha}\sqrt{2p(1-p)} + z_{1-\beta}\sqrt{p_1(1-p_1) + p_2(1-p_2)} \right\}^2}{(p_2-p_1)^2} .$$

In the case when we test for a two-sided alternative $z_{1-\alpha}$ is replaced by $z_{1-\alpha/2}$. The case of unequal group size allocation is treated similar to the continuous case.

19.3.2 Two-sample test for proportion using the unconditional approach

In this case we reject the null hypothesis if

$$\frac{\sqrt{n}(\overline{Y}_n - \overline{X}_n)}{\sqrt{v(p_1,p_2)}} \geq z_{1-\alpha} ,$$

that is, if $\sqrt{n}(\overline{Y}_n - \overline{X}_n) \geq z_{1-\alpha}\sqrt{v(p_1, p_2)}$. For a fixed p_1 and p_2 the power to detect the alternative is

$$1 - \beta = P\left\{ \frac{\sqrt{n}(\overline{Y}_n - \overline{X}_n) - \sqrt{n}(p_2 - p_1)}{\sqrt{v(p_1, p_2)}} \geq z_{1-\alpha} - \frac{\sqrt{n}(p_2 - p_1)}{\sqrt{v(p_1, p_2)}} \right\} .$$

Solving this for n we obtain that the sample size for one of the groups is

$$(z_{1-\alpha} + z_{1-\beta})^2 \frac{p_1(1-p_1) + p_2(1-p_2)}{(p_2 - p_1)^2} .$$

This is the formula most often referenced in practice. For a two-sided test $z_{1-\alpha}$ is changed to $z_{1-\alpha/2}$ and for unequal allocation of subjects per group small changes also need to be made. Note that the conditional approach always produces larger sample sizes than the unconditional approach because

$$2p(1-p) \geq p_1(1-p_1) + p_2(1-p_2) \Leftrightarrow (p_1 - p_2)^2 \geq 0 .$$

To make this more concrete and see how much the sample sizes differ, consider two examples $(p_1, p_2) = (0.2, 0.4)$ and $(p_1, p_2) = (0.6, 0.8)$, where we kept the difference in probability between the null and alternative hypotheses equal. The R code below calculates both sample sizes.

```
p1=0.2
p2=0.4
p=(p1+p2)/2
#Calculate the variance for unconstrained calculations
vp1p2<-p1*(1-p1)+p2*(1-p2)
#Set the size and power of the test
alpha=0.05
beta=0.1
za<-qnorm(1-alpha)
zb<-qnorm(1-beta)
#Calculate the unconstrained sample size
ss_u0<-round((za+zb)^2*vp1p2/(p2-p1)^2,digits=2)

#Calculate the constrained sample size
ss_c0<-round((za*sqrt(2*p*(1-p))+zb*sqrt(vp1p2))^2/(p1-p2)^2,digits=2)

perc_diff0<-round(100*(ss_c0-ss_u0)/ss_u0,digits=2)
```

The conditional approach requires a sample size of 88.03 compared with the unconditional approach which requires 85.64, or 2.79% more samples. We conduct the same calculations for $p_1 = 0.6$ and $p_2 = 0.8$. In this case the conditional approach requires a sample size of 88.03 compared with the unconditional approach which requires a sample size of 85.64, or 2.79% more samples.

19.4 Sample size calculations using exact tests

Using exact tests is particularly useful when the effect size is larger and the asymptotic distribution of the test statistic may be far from Normal. In the case of continuous data, using the t-test instead of the Normal test is quite reasonable. We have already shown how to obtain sample sizes for the t-test. Consider now the case when we are trying to obtain the sample size for testing for a Bernoulli proportion. We are using here a slightly modified example from our own experience. Suppose that a particular type of lung biopsy intervention has a historical probability of success $p_0 = 0.53$, and a new, imaging-assisted, surgical biopsy intervention was successful in 16 out of 20 surgeries. The estimated success rate in the new treatment arm was $\widehat{p} = 16/20 = 0.8$. We would like to find the sample size required to conduct a test of size $\alpha = 0.05$ with the power $1 - \beta = 0.9$ to detect a true proportion of successful biopsies of $p = 0.8$. Of course, we could use the Normal approximation test and obtain the sample size

$$n = (z_{1-\alpha} + z_{1-\beta})^2 \frac{p(1-p)}{(p-p_0)^2} ,$$

where we use a one-sample test. This results in a sample size of 18.8, which is rounded up to 19.

Now, we can conduct the same calculations using exact distributions. First, for every sample size n we need to find the number of successes, $s(n)$, that would be sufficient to reject the null hypothesis at that particular size. Let $\overline{X}_n$ be the proportion of successes among n surgeries and $n\overline{X}_n$ the total number of successes. We know that $n\overline{X}_n \sim$ Binomial(n, p_0) under the null hypothesis. One of the technical problems with the exact Binomial test is that we cannot construct an exact $\alpha = 0.05$ test. Indeed, consider the example when $n = 20$ and $p_0 = 0.53$. The rejection probability under the null if we see 14 or more successes is 0.096 and if we see 15 or more successes is 0.038. The reason is that the Binomial distribution is discrete. Of course, we could add a randomization rule to reject with some probability if we see 14 successes and always reject if we see 15 or more successes. This would create a correct α-level test, but would result in a test that is difficult to explain and we have never seen implemented in practice. Thus, for this n we choose the critical value to be $s(n) = 15$, the first number of successes that would provide a rejection probability less than 0.05. That is, we choose the critical value that ensures that we construct the least conservative test with a size $\alpha < 0.05$. This procedure can be repeated for every value of the sample size n. Figure 19.2 displays the graph of these critical values, $s(n)$ as a function of n,

```
#build a grid of points for the sample size
n=10:100
#calculate the critical value for every sample size
sn=qbinom(0.05,n,0.53,lower.tail=FALSE)+1
plot(n,sn,type="l",col="blue",lwd=3,
```

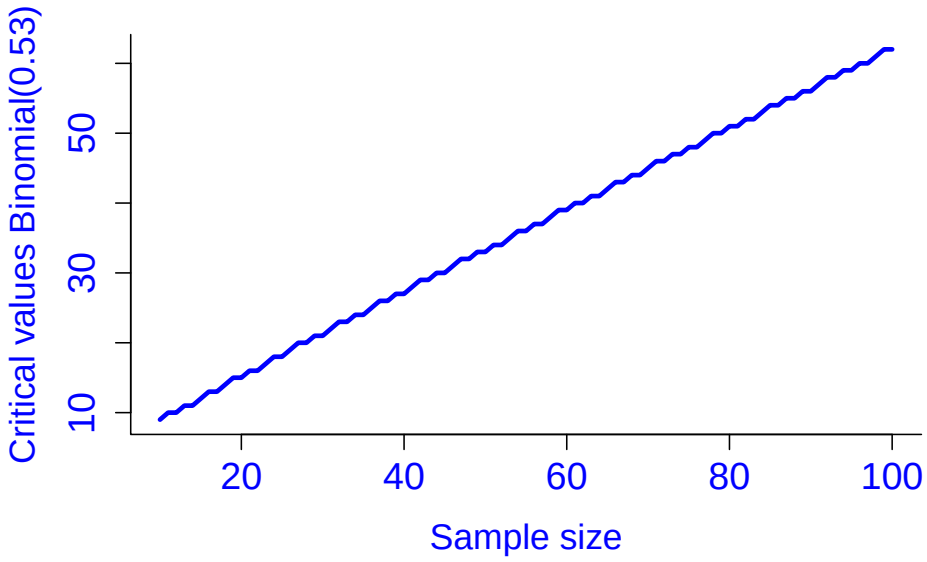


Figure 19.2: Critical value (y-axis) versus sample size (x-axis) for rejecting the null hypothesis $H_0 : p = 0.53$.

```
     xlab="Sample size",ylab="Critical values Binomial(0.53)",
     bty="l",cex.axis=1.5,cex.lab=1.4,col.lab="blue",
     col.axis="blue",main=NULL)
```

```
#Find the alpha level for n=100
alpha_100<-round(sum(dbinom(sn[length(sn)]:100,100,0.53)),digits=4)
```

The plot in Figure 19.2 is roughly linear with small bumps due to the discrete nature of the data. As the sample size increases the α level of the test is getting much closer to 0.05 because the normal approximation due to the Central Limit Theorem starts to kick in. Indeed, for $n = 100$ we obtain a value of $s(n)$ equal to 62 and a corresponding α level of the test equal to 0.0437. Note that this is still a test that does not have a size of $\alpha = 0.05$. Once the critical values are available for each sample size, we need to identify the sample size at which the power becomes equal to at least 0.9. To do that we calculate the probability of exceeding the critical values for a Binomial distribution with n samples and probability of success equal to 0.8.

```
power_binom<-rep(NA,length(n))
#We need to subtract 1 from sn because pbinom calculates the
#probability of strictly exceeding the critical value
power_binom<-pbinom(sn-1, n, 0.8, lower.tail = FALSE, log.p = FALSE)
```

Figure 19.3 displays the power for rejecting the null hypothesis $H_0 : p = 0.53$

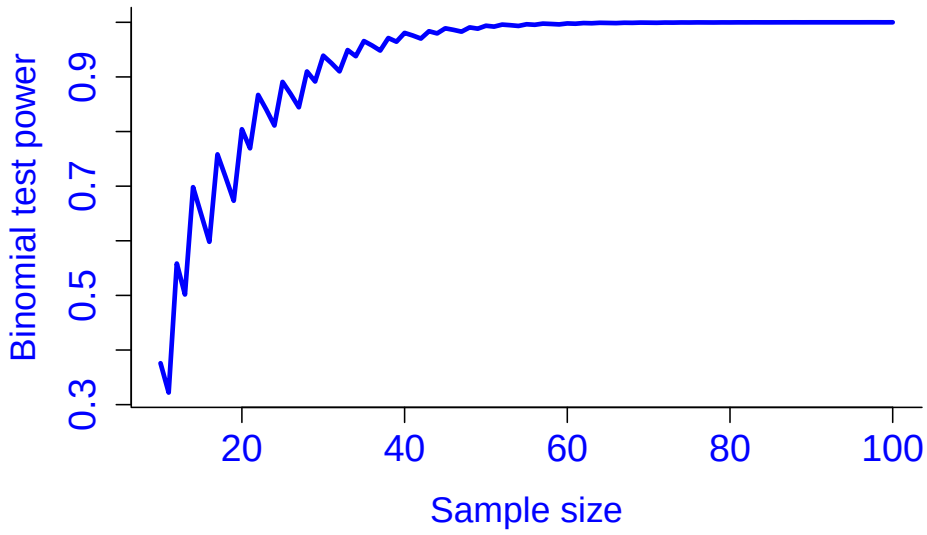


Figure 19.3: Power for rejecting the null hypothesis $H_0 : p = 0.53$ when the true value of the success probability is 0.8 as a function of the sample size.

when the true value of the success probability is 0.8 as a function of the sample size, n. The critical value is chosen so that the *alpha*-level of the test is as close as possible to 0.05.

```
plot(n,power_binom,type="l",col="blue",lwd=3,
     xlab="Sample size",ylab="Binomial test power",
     bty="l",cex.axis=1.5,cex.lab=1.4,col.lab="blue",
     col.axis="blue",main=NULL)
```

We are now interested in finding the sample size that will provide 90% power to detect a true success probability $p = 0.8$ when testing for the null hypothesis $H_0 : p = 0.53$ using an α-level as close as possible to 0.05.

```
#Calculate the first time the power exceeds 0.9
exceed_diff<-power_binom>0.9

#Get the sample size and corresponding critical value
sample_size_finite<-n[which.max(exceed_diff)]
critical_value_finite<-sn[which.max(exceed_diff)]

#Calculate the proportion difference between exact and asymptotic tests
ratio_exact_asymptotic<-round(100*(sample_size_finite-ss_o_c)/ss_o_c,digits=2)
```

Thus, the sample size is calculated as the first time the power exceeds 0.9 and that happens for 28 at a critical value of 20. Thus, the exact test would require 47.37% more subjects than the asymptotic test. The sample size is still small,

but it is much larger than the one required by the asymptotic test. A closer inspection of the power curve indicates that it has a rather unusual behavior as it is not strictly increasing. This is also due to the discrete nature of the Binomial distribution and to the fact that subsequent tests do not have the same size. Let us be concrete. Consider, for example, the case of $n = 15$ and $n = 16$, which both have a corresponding critical value of 12. The α level for the Binomial$(15, 0.53)$ test is 0.0303, while for the Binomial$(16, 0.53)$ it is 0.0194. This happens because the tests are chosen to be conservative and the tests are discrete. At this threshold the first test is less conservative (rejects easier) than the second test. Therefore, the power of the first test is 0.648, which is actually larger than the power of the second test 0.598. We conclude that the non-monotonic pattern of the power is due to the fact that we are comparing tests of different sizes. The problem becomes less obvious for larger values of the sample size n because the size of tests becomes closer to 0.05.

This is a simple example of exact calculation of sample sizes for the simplest case of testing for a given probability of success against an alternative. While calculations are more involved and do not follow a clearly prescribed formula, they are relatively easy to implement and follow in `R`. The difference between the sample size based on the asymptotic and exact tests is substantial in this case, which is very close to the real example encountered in practice. This shows one more time that truly understanding the concepts can make the Biostatistician more useful to his or her collaborators, more insightful about what needs to be done, and more constructive during the planning phase of an experiment.

19.5 Sample size calculation with preliminary data

So far, we have focused on relatively simple problems (e.g., two-sample t-test) and the case when the effect size, means, and standard deviations under the null and alternative hypotheses are known. In practice this is not always the case. Indeed, in most situations these quantities are unknown and are estimated from the data. The problem is that estimators of effect size are themselves subject to sampling variability, which is not typically taken into account. Accounting for sampling variability leads to larger sample sizes, sometimes much larger. We provide a couple of examples and a general strategy for dealing with such cases.

19.5.1 One-sample exact Binomial test

Consider, for example, the lung biopsy case. Your collaborator just told you that the new procedure was successful on 16 out of 20 study participants, for an estimated success probability of $\widehat{p} = 0.8$. However, you are a cautious Biostatistician and notice that this is the point estimate and, given that only 20

subjects were used in the preliminary study, it is likely that the true success probability of the new procedure can be quite a bit larger or smaller. Just how much smaller or larger? Luckily, we have learned how to obtain a confidence interval for proportions

```
library(binom)
x=16
n=20
binom.confint(x, n, conf.level = 0.95, methods = "all")
```

```
          method  x  n      mean     lower     upper
1  agresti-coull 16 20 0.8000000 0.5782364 0.9250885
2     asymptotic 16 20 0.8000000 0.6246955 0.9753045
3          bayes 16 20 0.7857143 0.6134811 0.9421974
4        cloglog 16 20 0.8000000 0.5511456 0.9198179
5          exact 16 20 0.8000000 0.5633860 0.9426660
6          logit 16 20 0.8000000 0.5721531 0.9228665
7         probit 16 20 0.8000000 0.5852912 0.9289199
8        profile 16 20 0.8000000 0.5946321 0.9331375
9            lrt 16 20 0.8000000 0.5946356 0.9331592
10     prop.test 16 20 0.8000000 0.5573138 0.9338938
11        wilson 16 20 0.8000000 0.5839826 0.9193423
```

All these intervals seem to agree that even values of $p = 0.6$ can actually be consistent with the observed probability of success among the 20 biopsies. The sample size calculations will, of course, lead to much larger sample sizes when testing against the alternative $p = 0.6$ versus $p = 0.8$.

For example, the asymptotic Normal test leads to a sample size of 420 for $p = 0.6$, which is much larger than the sample size 19 based on the point estimator $p = 0.8$. The exact test sample size calculations provides a sample size of 431, larger than the sample size using the asymptotic approximation, but of a similar magnitude. Such a large difference should have a sobering effect on the Biostatistician and scientist and could provide information about the sample size for the preliminary study, as well. Given the fickleness of the sample size calculations, it is important to be open and clear about the various assumptions that go into the calculation. We have found that the variability of the preliminary data can lead to substantially larger sample sizes.

19.5.2 Two-sample t-tests with preliminary data

Consider the case when we would like to compare the respiratory disruptance index `rdi4p` between men and women. Suppose that we have only the data from the first 10 men and 10 women from the SHHS study; the advantage of having a large study, such as SHHS, is that we can play all sorts of data games with subsamples to highlight real-life scenarios. We extract the data, transform them

using the square root transform to reduce the skewness of `rdi4p`, and conduct a t-test for equality of means between the two samples.

```
## read in the data
dat = read.table(file = "data/shhs1.txt",
                 header = TRUE, na.strings="NA")

sqrt_rdi4p_male<-sqrt(dat$rdi4p[dat$gender==1])[1:10]
sqrt_rdi4p_female<-sqrt(dat$rdi4p[dat$gender==0])[1:10]

#Obtain the means of the groups
m_sqrt_rdi4p_male<-round(mean(sqrt_rdi4p_male),digits=2)
m_sqrt_rdi4p_female<-round(mean(sqrt_rdi4p_female),digits=2)


t.test(sqrt_rdi4p_male,sqrt_rdi4p_female,
       alternative="two.sided",var.equal = TRUE)
```

```
    Two Sample t-test

data:  sqrt_rdi4p_male and sqrt_rdi4p_female
t = 1.2315, df = 18, p-value = 0.234
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.5021601  1.9246981
sample estimates:
mean of x mean of y
 2.626047  1.914778
```

The mean of the square root `rdi4p` for the first 10 men is 2.63, which is larger than the mean of the square root of `rdi4p` for the first 10 women, 1.91. The confidence interval covers zero and the p-value for the t-test of equality of means is $p = 0.235$, indicating that the null hypothesis cannot be rejected at the level $\alpha = 0.05$ by a two-sided t-test. This raises questions about what to do and how to estimate the sample size. Of course, a solution is to take the point estimators for the mean and standard deviations in each group, treat them as fixed, and obtain the sample size.

```
delta<-m_sqrt_rdi4p_male-m_sqrt_rdi4p_female
pooled_sd<-sd(c(m_sqrt_rdi4p_male,m_sqrt_rdi4p_female))
pwr<-power.t.test(power = .90, sig.level=0.05, delta = delta, sd=pooled_sd,
                  alternative = "two.sided")
sample_size_group<-ceiling(pwr$n)
```

After conducting the power calculation we obtain that the sample size per group is 12, slightly larger than 10, the number of subjects that we have used to estimate the sample size. Now, the nice thing is that we actually can sample

the SHHS dataset and obtain pairs of 12 men and 12 women and test whether the difference is significant. Note that typically we do not have this luxury, but a large dataset comes with its own perks. One of them is that we can really investigate the behavior of various statistical concepts in smaller sample sizes. We do this below,

```
#Obtain vectors of rdi4p for males and females
sqrt_rdi4p_male<-sqrt(dat$rdi4p[dat$gender==1])
sqrt_rdi4p_female<-sqrt(dat$rdi4p[dat$gender==0])

nsim=10000
rejections<-rep(NA,nsim)

set.seed(256423)
for (i in 1:nsim)
  {#begin sampling from the SHHS
  #Sample the groups of males and females with replacement
  #This could be done with or without replacement
  create_male_subsample<-sample(sqrt_rdi4p_male,size=sample_size_group,replace=TRUE)
  create_female_subsample<-sample(sqrt_rdi4p_female,size=sample_size_group,replace=TRUE)

  ttest_results<-t.test(create_male_subsample,create_female_subsample,
                        alternative="two.sided",var.equal = TRUE)
  rejections[i]<-ttest_results$p.value<0.05
  }#end sampling from the SHHS

rejection.rate<-round(mean(rejections),digits=3)
```

After conducting these simulations we end up with a rejection rate of 0.256, which is much, much smaller than the nominal power $1 - \beta = 0.9$. This must be sobering and should raise fundamental questions about what it is that we are doing. Of course, one of the problems could be that we chose the first 10 observations and that the sample means and standard deviations are quite variable. Let us compare these values more closely to see what actually happens.

```
#Find the mean, mean difference, pooled standard deviation
# and effect size at the population level
mean_male<-round(mean(sqrt_rdi4p_male),digits=2)
mean_female<-round(mean(sqrt_rdi4p_female),digits=2)
delta_pop<-mean_male-mean_female
sd_pooled_pop<-round(sd(c(sqrt_rdi4p_male,sqrt_rdi4p_female)),digits=2)
f_pop<-round((mean_male-mean_female)/sd_pooled_pop,digits=2)

#Calculate the percent difference between the population delta
#and delta for the first 10 males and 10 females
percent_delta_diff<-100*(delta_pop-delta)/delta
```

```
#Effect size using only the first 10 males and 10 females
f_first_10<-round(delta/pooled_sd,digits=2)
```

The mean square root of `rdi4p` for the entire SHHS male population is 2.87, which is larger, but relatively close to 2.63, the mean in the first 10 samples of male `rdi4p`. Similarly, the mean square root of `rdi4p` for the entire SHHS female population is 1.97, which is larger, but relatively close to 1.91, the mean in the first 10 samples of female `rdi4p`. Thus, the difference in means at the population level is 0.9, which is 25 percent larger than the difference in means observed in the first 10 samples. The real difference, however, comes from the estimator of the pooled standard error. Indeed, this is equal to 1.71 at the population level and 0.51 among the first 10 samples, or more than 3 times larger. Therefore, the effect size at the population level is 0.53 compared with the effect size 1.41 based on the first 10 males and females in the sample. Thus, the true effect size is roughly 3 times smaller than the one based on the first 10 observations, which is a major reason why there is a large discrepancy between the observed and nominal power of detecting the alternative. This is not a problem limited to the fact that we chose the first 10 observations, but it represents the serious problems associated with sample size variability when the effect size is estimated from a small to moderate sample.

19.5.3 Complex scenarios with preliminary data

We have discussed sample size calculations in specific, though relatively simple scenarios. In practice such scenarios appear regularly, but sometimes more complex problems occur, as well. Consider the case of a regression problem that has some interactions and we are interested in the sample size that would allow us to identify a particular effect with very large probability. For example, consider the following model using all the SHHS data without missing covariates ($n = 5761$),

```
MtS_SA=dat$rdi4p>=15
fit<-glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1+age_s1*HTNDerv_s1,
           family="binomial",data=dat)
summary(fit)
```

```
Call:
glm(formula = MtS_SA ~ gender + age_s1 + bmi_s1 + HTNDerv_s1 +
    age_s1 * HTNDerv_s1, family = "binomial", data = dat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0443  -0.6559  -0.4441  -0.2671   2.8556
```

```
Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)       -8.710927   0.421770 -20.653  < 2e-16 ***
gender             1.156897   0.079181  14.611  < 2e-16 ***
age_s1             0.036892   0.005000   7.378 1.61e-13 ***
bmi_s1             0.138369   0.007412  18.669  < 2e-16 ***
HTNDerv_s1         0.758377   0.473129   1.603    0.109
age_s1:HTNDerv_s1 -0.008748   0.007146  -1.224    0.221
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5314.5  on 5760  degrees of freedom
Residual deviance: 4659.1  on 5755  degrees of freedom
  (43 observations deleted due to missingness)
AIC: 4671.1

Number of Fisher Scoring iterations: 5
```

The point estimator of `HTNDerv_s1` based on the entire dataset is 0.758 with a p-value of 0.109. We would like to know at what sample size we would reject the null hypothesis $100(1-\beta)\%$. To do that we fix a sample size, say 20% larger, and sample the data with replacement, but with a sample size 20% larger than the original data. This is exactly like the bootstrap, except that we are not re-sampling a data set of the same size with the original; we sample more, though we could also sample less. Surprisingly, it is nearly everyone's mantra that the bootstrap should sample the same number of subjects. We agree to disagree and we have found that sampling with replacement with larger or smaller sample sizes could actually be extremely helpful. Of course, we can simulate many datasets that are 20% larger and for each one of them we count how many times we reject the null that the HTN effect is equal to zero. Below we provide the necessary code to do that, though we run it separately (`eval=FALSE`) because it takes time to run thousands of models.

```
n_total=dim(dat)[1]
length_upstrap=21
nboot=10000
percent_upstrap=seq(from=1, to=5,length=length_upstrap)
check<-matrix(rep(NA,nboot*length_upstrap),ncol=length_upstrap)

for (j in 1:length_upstrap)
  {#Each loop corresponds to an increase in the sample size
  print(j)
  for (i in 1:nboot)
  {#Each loop corresponds to one sample with a specified sample size
```

```
  temp_index<-sample(1:n_total,n_total*percent_upstrap[j],replace=TRUE)
  temp_data<-dat[temp_index,]
  MtS_SA=temp_data$rdi4p>=15
  fit<-glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1+age_s1*HTNDerv_s1,
           family="binomial",data=temp_data)
  #Obtain the p-values for HTN
  check[i,j]<-coef(summary(fit))[,4][5]<0.05
  }
}

power_check<-colMeans(check)

plot(percent_upstrap,power_check,type="l",col="blue",lwd=3,
     xlab="Factor by which the sample size is multiplied",
     ylab="Power to detect the HTN effect",
     bty="l",cex.axis=1.5,cex.lab=1.4,col.lab="blue",
     col.axis="blue",main=NULL)
abline(h=0.8,lwd=3,col="orange")
abline(h=0.9,lwd=3,col="red")
```

The Figure below provides the frequency with which the test for no HTN effect is rejected in the model

```
glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1+age_s1*HTNDerv_s1,
           family="binomial",data=dat)
```

as a function of the multiplier of the sample size. For example, for the multiplier 2 we produced 10000 samples with replacement from the SHHS with twice the number of subjects $2n = 11608$. For each up-sampled dataset we ran the model and recorded whether the p-value for HTN was smaller than 0.05. At this value of the sample size we obtained that the HTN effect was identified in 62% of the up-samples. We also obtained that the power was equal to 0.794 at the sample size multiplier 3.0 and 0.817 at multiplier 3.2, indicating that the power 0.8 would be attained at $3.1 * n \approx 17500$.

There are very few methods to estimate the sample size in such examples and we contend that the upstrap is a powerful and general method to conduct such calculations. Similar approaches could be used in many other situations, including estimating a fixed effect (e.g., treatment) using longitudinal data in the context of a clinical trial or the sample size necessary to detect gene by gene and gene by environment interactions in genomics studies.

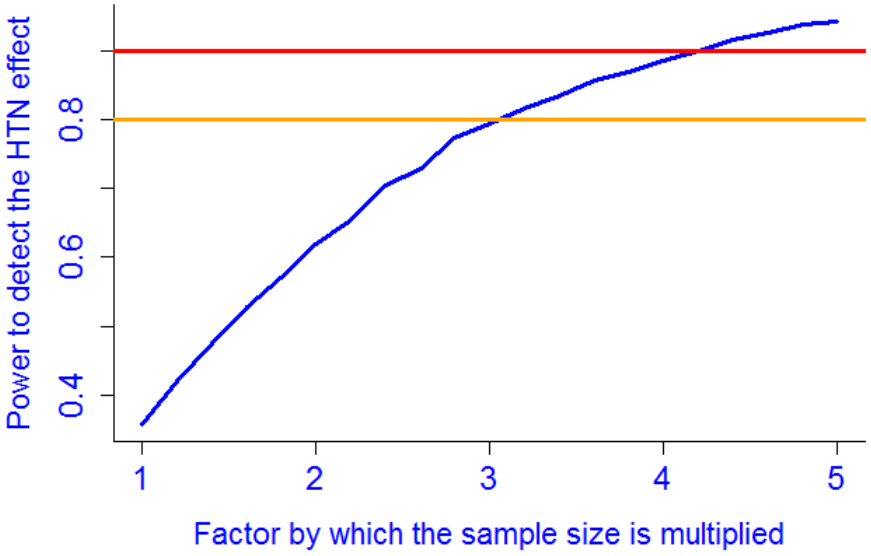


Figure 19.4: Power of detecting the HTN effect in an upstrap (sample with replacement of the data with a sample size different from the sample size of the original data) as a function of the multiplier of the original sample size.

19.6 Problems

Problem 1. Consider the case when we observe a single sample $X_1, \ldots, X_n \sim N(\mu, \sigma^2)$ and we are interested in testing the null hypothesis $H_0 : \mu = \mu_0$ versus $H_A : \mu > \mu_0$, where μ_0 is a fixed and known constant.

a. Calculate the sample size for the corresponding Normal test of size α to achieve a power $1 - \beta$ at a value $\mu_1 > \mu_0$.
b. What is a reasonable definition of effect size in this context?
c. Calculate the sample size for $\alpha = 0.05$, $\beta = 0.1$ and an effect size $f = 0.4$.

Problem 2. Solve the same problem in the case of the same null hypothesis with the two-sided alternative $H_A : \mu \neq \mu_0$.

Problem 3. Derive the sample size calculation for the two-sample testing problem with Bernoulli outcomes with unequal group sizes. Calculate the sample size needed to detect the alternative hypothesis using a one-sided test of size $\alpha = 0.05$ with probability 0.9, when the true probabilities in the two groups are $p_1 = 0.5$ and $p_2 = 0.65$ and there are twice as many subjects in group 1 versus group 2. Do this both for the conditional and unconditional approach.

Problem 4. For testing the difference in probability between two groups against a one-sided alternative, use simulations to compare the size and power of the conditional and unconditional tests.

Problem 5. Derive the sample size calculations for the one-sample problem when testing for the null hypothesis $H_0 : p = p_1$ against the alternative $H_A : p > p_1$. Obtain both the conditional and unconditional sample sizes. Compare the sample sizes required by the conditional and unconditional approaches and indicate when the difference is larger.

Problem 6. Set up simulations to compare the power and sample size of the Normal approximation tests (conditional and unconditional) for the one-sample test of probability.

Problem 7. Sometimes proving superiority of a new drug may not be possible. Therefore, in practice one sometimes uses a test for the null hypothesis of non-inferiority. For the two-sample the non-inferiority hypothesis is

$$H_0 : \mu_2 - \mu_1 \leq -\epsilon \quad \text{versus} \quad H_A : \mu_2 - \mu_1 > -\epsilon \, ,$$

where ϵ is a small positive constant. The idea is that under the null the mean of group 2, μ_2, is inferior to the mean in group 1, μ_1, by a margin ϵ. That is $\mu_2 \leq \mu_1 - \epsilon$. For an α level and power $1 - \beta$ derive the sample size necessary to detect an alternative. Use the Normal test and t-test for equality of means.

Problem 8. Another type of testing is that of equivalence. In this scenario we still have two groups, say of the same size, n, but we are interested in testing the equivalence of two treatments/drugs. In this case we are testing

$$H_0 : |\mu_2 - \mu_1| \geq \epsilon \quad \text{versus} \quad H_A : |\mu_2 - \mu_1| < \epsilon \, .$$

The idea is that under the null the mean of group 2, μ_2, is substantially different from the mean in group 1, μ_1, by a margin ϵ. The alternative is that the difference between the two means is small, that is, the two groups/treatments are equivalent. For an α level and power $1-\beta$ derive the sample size necessary to detect an alternative. Use the Normal test and t-test for equality of means.

Problem 9. Describe the non-inferiority and equivalence tests for the two-sample problem for proportions and derive the sample size for detecting the alternative at a given size α and power $1-\beta$.

Problem 10. Consider the case when we want to test

$$H_0 : p = 0.53 \quad \text{versus} \quad H_A : p > 0.53 \, ,$$

and we have $n = 20$ trials. Construct an exact $\alpha = 0.05$ test by rejecting with a probability p_r if there are exactly 14 successes and always reject if there are 15 or more successes.

Problem 11. Consider the case when we want to test

$$H_0 : p = 0.53 \quad \text{versus} \quad H_A : p \neq 0.53 \, .$$

For an $\alpha = 0.05$ size of the exact test calculate the power to detect an alternative $p \in (0, 1)$ as a function of p and sample size n.

a. Provide a report written in Rmarkdown explaining the findings. Hint: show power plots for several sample sizes of interest.
b. For a fixed probability difference $|p - p_0|$ obtain the sample size that would ensure a power $1 - \beta = 0.9$.
c. Plot the size of the tests as a function of n and discuss why the size is not equal to 0.05.

Problem 12. Consider the same example of the lung biopsy, but your collaborator tells you that he or she thinks that the new procedure is truly superior to the previous procedure. When asked specifically about what is meant by "truly superior," your collaborator says that he or she thinks that the new procedure has a probability of success at least 20% larger than the historical probability $p_0 = 0.53$. Set up the non-inferiority hypothesis and calculate the sample size necessary to show superiority (not non-inferiority) if the true probability of success of the new procedure is 0.8. Use the exact test with a $\alpha = 0.05$ size and power $1 - \beta = 0.9$.

Problem 13. Given that we observe 16 biopsy successes out of 20 surgeries, what is the probability of observing this many or more successes if the true value of the success probability is $p = 0.7$? For $p = 0.7$ calculate the sample size based on the asymptotic and exact tests and provide the necessary changes in your code.

a. Plot the critical values for the Binomial test versus the critical values for the asymptotic test.

b. Plot the power of the exact Binomial test as a function of sample size
c. What are the differences from the case when the true value of the success probability was 0.8?
d. What is the probability of observing this many successes if the true value of the success probability is $p = 0.8$?

Problem 14. Using the first 10 `rdi4p` observations from SHHS from men and women, calculate the sample size required to detect the observed empirical effect size using a two-sided t-test with unequal variances, size $\alpha = 0.05$, and power $1 - \beta = 0.9$. Here the observed empirical effect size is obtained by plugging in the point estimators of the mean and standard deviations obtained from the samples.

Problem 15. Calculate the gender-specific means and pooled standard deviation of the square root of `rdi4p` and calculate the sample size necessary to detect such a large difference from the data. With the estimated sample size, subsample groups of males and females and calculate with this estimated sample size and compare that with the nominal power.

Problem 16. Using a random sample of 10 females and 10 males from the SHHS, calculate the empirical effect size using plug-in estimators of the means and pooled standard deviation.

a. For a t-test of equality in means calculate the sample size to detect the observed effect size ($\alpha = 0.05$, $1 - \beta = 0.9$).
b. Repeat this procedure 10000 times and plot the histogram of estimated sample sizes.
c. Interpret and report your findings.

Problem 17. Using the model

```
MtS_SA=dat$rdi4p>=15
fit<-glm(MtS_SA~gender+age_s1+bmi_s1+HTNDerv_s1+age_s1*HTNDerv_s1,
           family="binomial",data=dat)
summary(fit)
```

```
Call:
glm(formula = MtS_SA ~ gender + age_s1 + bmi_s1 + HTNDerv_s1 +
    age_s1 * HTNDerv_s1, family = "binomial", data = dat)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0443  -0.6559  -0.4441  -0.2671   2.8556

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)       -8.710927   0.421770 -20.653  < 2e-16 ***
gender             1.156897   0.079181  14.611  < 2e-16 ***
```

```
age_s1             0.036892   0.005000   7.378 1.61e-13 ***
bmi_s1             0.138369   0.007412  18.669  < 2e-16 ***
HTNDerv_s1         0.758377   0.473129   1.603    0.109
age_s1:HTNDerv_s1 -0.008748   0.007146  -1.224    0.221
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5314.5  on 5760  degrees of freedom
Residual deviance: 4659.1  on 5755  degrees of freedom
  (43 observations deleted due to missingness)
AIC: 4671.1

Number of Fisher Scoring iterations: 5
```

a. Find the sample size at which the gender effect would be found in 80% of the samples.
b. Find the sample size at which the age effect would be found in 80% of the samples.

References

Abdi, H., and L.J. Williams. 2010. "Jackknife." *Neil Salkind (Ed.), Encyclopedia of Research Design*.

Agresti, A., and B.A. Coull. 1998. "Approximate Is Better Than 'Exact' for Interval Estimation of Binomial Proportions." *The American Statistician* 52: 119–26.

Bartlett, R.H., D.W. Roloff, R.G. Cornell, A.F. Andrews, P.W. Dillon, and J.B. Zwischenberger. 1985. "Extracorporeal Circulation in Neonatal Respiratory Failure: A Prospective Randomized Study." *Pediatrics* 76 (4): 479–87.

Bayes, T., R. Price, and J. Canton. 1763. *An Essay Towards Solving a Problem in the Doctrine of Chances*. C. Davis, Printer to the Royal Society of London.

Beall, G. 1942. "The Transformation of Data from Entomological Field Experiments." *Biometrika* 29: 243–62.

Becker, R.A., J.M. Chambers, and A.R. Wilks. 1988. *The New S Language*. Pacific Grove, CA, USA: Wadsworth & Brooks/Cole.

Behrens, W.U. 1929. "A Contribution to Error Estimation with Few Observations." *Landwirtschaftliche Jahrbücher* 68: 807–37.

Benjamini, Y., and Y. Hochberg. 1995. "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing." *Journal of the Royal Statistical Society. Series B (Methodological)*, 289–300.

Bland, J.M., and D.G. Altman. 1986. "Statistical Methods for Assessing Agreement Between Two Methods of Clinical Measurement." *Lancet* 327: 307–10.

Box, G.E.P., and D.R. Cox. 1964. "An Analysis of Transformations." *Journal of the Royal Statistical Society, Series B* 26: 211–52.

Buonaccorsi, J.P. 2010. *Measurement Error: Models, Methods and Applications*. Boca Raton, FL, USA: Chapman & Hall/CRC.

Burhenne, L.J., and H.J. Burhenne. 1993. "The Canadian National Breast Screening Study: A Canadian Critique." *AJR. American Journal of Roentgenology* 161 (4): 761–63.

Carroll, R.J., D. Ruppert, L.A. Stefansky, and C.M. Crainiceanu. 2006. *Measurement Error in Nonlinear Models: A Modern Perspective*. Boca Raton, FL, USA: Chapman & Hall/CRC.

Chang, W. 2013. *R Graphics Cookbook*. Sebastopol, Ca, USA: O'Reilly Media, Inc.

Chow, S.-C., J. Shao, and H. Wang. 2008. *Sample Size in Clinical Research*. Boca Raton, FL, USA: Chapman & Hall/CRC.

Cleveland, W.S. 1979. "Robust Locally Weighted Regression and Smoothing Scatterplots." *Journal of the American Statistical Association* 74: 829–36.

———. 1981. "LOWESS: A Program for Smoothing Scatterplots by Robust Locally Weighted Regression." *The American Statistician* 35: 54.

Clopper, C., and E.S. Pearson. 1934. "The Use of Confidence or Fiducial Limits Illustrated in the Case of the Binomial." *Biometrika* 26: 404–13.

Crowson, C.S., E.J. Atkinson, and T.M. Therneau. 2016. "Assessing Calibration of Prognostic Risk Scores." *Statistical Methods in Medical Research* 25: 1692–1706.

Dean, D.A., A.L. Goldberger, R. Mueller, M. Kim, M. Rueschman, D. Mobley, S.S. Sahoo, et al. 2016. "Scaling up Scientific Discovery in Sleep Medicine: The National Sleep Research Resource." *Sleep* 5: 1151–64.

Efron, B. 1979. "Bootstrap methods: Another look at the jackknife." *The Annals of Statistics* 7 (1): 1–26.

Efron, B., and R. Tibshirani. 1993. *An Introduction to the Bootstrap*. Boca Raton, FL, USA: Chapman & Hall/CRC.

Eilers, P.H.C., and B.D. Marx. 1996. "Flexible smoothing with B -splines and penalties." *Statistical Science* 11: 89–121.

Fisher, R.A. 1925. *Statistical Methods for Research Workers*. Edingburgh, UK: Oliver & Boyd.

———. 1935. "The Fiducial Argument in Statistical Inference." *Annals of Eugenics* 8: 391–98.

———. 1940. "The Precision of Discriminant Functions." *Annals of Eugenics* 10: 422–29.

Fuller, W.A. 1987. *Measurement Error Models*. New York, USA: Wiley.

Gosset, W.S. 1908. "The Probable Error of a Mean." *Biometrika* 6: 1–25.

Hilfiger, J.J. 2016. *Graphing Data with R*. O'Reilly Media, Inc.

Hosmer, D.W., and S. Lemeshow. 2013. *Applied Logistic Regression*. Wiley.

Katherine, S.B., J.P.A. Ioannidis, C. Mokrysz, B.A. Nosek, J. Flint, E.S.J. Robinson, and M.R. Munafò. 2013a. "Confidence and Precision Increase with High Statistical Power." *Nature Reviews Neuroscience* 14: 585.

———. 2013b. "Power Failure: Why Small Sample Size Undermines the Reliability of Neuroscience." *Nature Reviews Neuroscience* 14: 365.

Landman, B.A., A.J. Huang, A. Gifford, D.S. Vikram, and others. 2011. "Multi-Parametric Neuroimaging Reproducibility: A 3-T Resource Study." *Neuroimage* 54: 2854–66. https://www.nitrc.org/projects/multimodal/.

Lindquist, M.A., B. Caffo, and C. Crainiceanu. 2013. "Ironing Out the Statistical Wrinkles in "Ten Ironic Rules"." *Neuroimage* 81: 499–502.

Mayo, D.G. 2014. "On the Birnbaum Argument for the Strong Likelihood Principle." *Statistical Science*, 227–39.

McNeil, D. 1977. *Interactive Data Analysis*. New York: Wiley.

Miller, A.B., T. To, C.J. Baines, and C. Wall. 2000. "Canadian National Breast Screening Study-2: 13-Year Results of a Randomized Trial in Women Aged 50–59 Years." *Journal of the National Cancer Institute* 92: 1490–9.

———. 2002. "The Canadian National Breast Screening Study-1: Breast Cancer Mortality After 11 to 16 Years of Follow-up: A Randomized Screening Trial of Mammography in Women Age 40 to 49 Years." *Annals of Internal Medicine* 137: 305–12.

Miller, A.B., C. Wall, C.J. Baines, P. Sun, T. To, and S.A. Narod. 2014. "Twenty Five Year Follow-up for Breast Cancer Incidence and Mortality of the Canadian National Breast Screening Study: Randomised Screening Trial." *British Medical Journal* 348: g366.

Millner, A., and R. Calel. 2012. "Are First-Borns More Likely to Attend Harvard?" *Significance* 9: 37–39.

Muschelli, J., J.P. Fortin, B. Avants, B. Whitcher, J.D. Clayden, B. Caffo, and C.M. Crainiceanu. 2019. "Neuroconductor: an R platform for medical imaging analysis." *Biostatistics* 20: 218–39.

Neyman, J., and E.S. Pearson. 1933. "On the Problem of the Most Efficient Tests of Statistical Hypotheses." *Philosophical Transactions of the Royal Society London, Series A* 231: 289–337.

Nychka, D., R. Furrer, J. Paige, and S. Sain. 2016. *Fields: Tools for Spatial Data*. http://CRAN.R-project.org/package=fields.

O'Sullivan, F. 1986. "A Statistical Perspective on Ill-Posed Inverse Problems (with Discussion)." *Statistical Science* 1: 505–27.

Portnoy, S., and R. Koenker. 1997. "The Gaussian Hare and the Laplacian Tortoise: Computability of Squared-Error Vs. Absolute-Error Estimators, with Discussion." *Statistical Science* 12: 279–300.

Punjabi, N.M. 2008. "The Epidemiology of Adult Obstructive Sleep Apnea." *Proceedings of the American Thoracic Society* 5: 136–43.

Quan, S.F., B.V. Howard, C. Iber, J.P. Kiley, F.J. Nieto, O'ConnorG.T., D.M. Rapoport, et al. 1997. "The Sleep Heart Health Study: design, rationale, and methods." *Sleep* 20 (12): 1077–85.

Quenouille, M.H. 1949. "Problems in Plane Sampling." *The Annals of Mathematical Statistics* 20: 355–75.

———. 1956. "Notes on Bias in Estimation." *Biometrika* 43: 353–60.

Rahlf, T. 2017. *Data Visualisation with R: 100 Examples*. Springer.

R Core Team. 2016. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Redelmeier, D.A., and S.M. Singh. 2001. "Survival in Academy Award–Winning Actors and Actresses." *Annals of Internal Medicine* 134: 955–62.

Redline, S., M.H. Sanders, B.K. Lind, S.F. Quan, C. Iber, D.J. Gottlieb, W.H. Bonekat, D.M. Rapoport, P.L. Smith, and J.P. Kiley. 1998. "Methods for Obtaining and Analyzing Unattended Polysomnography Data for a Multicenter Study. Sleep Heart Health Research Group." *Sleep* 7: 759–67.

Rosenberger, W.F., and J.M. Lachin. 2015. *Randomization in Clinical Trials: Theory and Practice*. Hoboken, New Jersey, USA: John Wiley & Sons.

Rossouw, J.E., G.L. Anderson, R.L. Prentice, A.Z. LaCroix, C. Kooperberg, M.L. Stefanick, R.D. Jackson, et al. 2002. "Writing Group for the Women's Health Initiative Investigators. Risks and Benefits of Estrogen Plus Progestin in Healthy Postmenopausal Women: Principal Results from the Women's Health Initiative Randomized Controlled Trial." *Journal of the American Medical Association* 288: 321–33.

Royall, R. 2017. *Statistical Evidence: A Likelihood Paradigm*. Boca Raton, Florida, USA: Chapman & Hall/CRC.

Ruppert, D., M.P. Wand, and R.J. Carroll. 2003. *Semiparametric Regression*. Cambridge, UK: Cambridge University Press.

Sandel, M.J. 2010. *Justice: What's the Right Thing to Do?* New York, USA: Farrar, Straus; Giroux.

Selvin, S. 1975. "A Problem in Probability (Letter to the Editor)." *American Statistician* 29 (1): 67.

Sheather, S.J., and M.C. Jones. 1991. "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation." *Journal of the Royal Statistical Society. Series B* 53: 683–90.

Sievert, C., C. Parmer, T. Hocking, S. Chamberlain, K. Ram, M. Corvellec, and P. Despouy. 2017. *plotly: Create Interactive Web Graphics via 'plotly.js'*. https://CRAN.R-project.org/package=plotly.

Sundar, D.-R. 2014. *Binom: Binomial Confidence Intervals for Several Parameterizations*. https://CRAN.R-project.org/package=binom.

Sylvestre, M.-P., E. Huszti, and J.A. Hanley. 2006. "Do Oscar Winners Live Longer Than Less Successful Peers? A Reanalysis of the Evidence." *Annals of Internal Medicine* 145: 361–63.

Tibshirani, R. 1996. "Regression Shrinkage and Selection via the Lasso." *Journal of the Royal Statistical Society. Series B* 58: 267–88.

Tukey, J.W. 1958. "Bias and Confidence in Not Quite Large Samples (Abstract)." *The Annals of Mathematical Statistics* 29 (2): 614.

———. 1970. *Exploratory Data Analysis*. Addison-Wesley.

Venables, W.N., and B.D. Ripley. 2002. *Modern Applied Statistics with S. Fourth edition*. Springer.

Wei, T. 2013. *Corrplot: Visualization of a Correlation Matrix*. http://CRAN.R-project.org/package=corrplot.

Welch, B.L. 1938. "The Significance of the Difference Between Two Means When the Population Variances Are Unequal." *Biometrika* 29: 350–62.

Whitcher, B., V.J. Schmid, and A. Thornton. 2011. "Working with the DICOM and NIfTI Data Standards in R." *Journal of Statistical Software* 44: 1–28. http://www.jstatsoft.org/v44/i06/.

Wickham, H. 2014. "Tidy Data." *Journal of Statistical Software* 59 (10): 1–23.

———. 2016. *ggplot2: Elegant Graphics for Data Analysis*. New York, USA: Springer. http://ggplot2.org.

Wickham, H., and L. Henry. 2018. *tidyr: Easily Tidy Data with 'Spread()' and 'Gather()' Functions*. https://CRAN.R-project.org/package=tidyr.

Wickham, H., J. Hester, and R. Francois. 2017. *readr: Read Rectangular Text Data*. https://CRAN.R-project.org/package=readr.

Wickham, H., and E. Miller. 2017. *haven: Import and Export SPSS, Stata and SAS Files*. https://CRAN.R-project.org/package=haven.

Wilkinson, Leland. 2006. *The Grammar of Graphics*. Springer Science & Business Media.

Wood, S.N. 2004. "Stable and Efficient Multiple Smoothing Parameter Estimation for Generalized Additive Models." *Journal of the American Statistical Association* 99: 673–86.

———. 2017. *Generalized Additive Models: An Introduction with R (second edition)*. Boca Raton, FL, USA: Chapman; Hall/CRC.

Zhang, E., V.Q. Wu, S.-C. Chow, and H.G. Zhang. 2013. *TrialSize: R Functions in Chapter 3,4,6,7,9,10,11,12,14,15*. https://CRAN.R-project.org/package=TrialSize.

Zhang, L., J. Samet, B. Caffo, and N.M. Punjabi. 2006. "Cigarette Smoking and Nocturnal Sleep Architecture." *American Journal of Epidemiology* 164: 529–37.