Use R!

Chris Chapman
Elea McDonnell Feit

R for Marketing Research and Analytics

Second Edition

Springer

Use R!

Use R!

This series of inexpensive and focused books on R will publish shorter books aimed at practitioners. Books can discuss the use of R in a particular subject area (e.g., epidemiology, econometrics, psychometrics) or as it relates to statistical topics (e.g., missing data, longitudinal data). In most cases, books will combine LaTeX and R so that the code for figures and tables can be put on a website. Authors should assume a background as supplied by Dalgaard's Introductory Statistics with R or other introductory books so that each book does not repeat basic material.

More information about this series at http://www.springer.com/series/6991

Chris Chapman · Elea McDonnell Feit

R for Marketing Research and Analytics

Second Edition

Springer

Chris Chapman
Google
Seattle, WA, USA

Elea McDonnell Feit
Drexel University
Philadelphia, PA, USA

ISSN 2197-5736 ISSN 2197-5744 (electronic)
Use R!
ISBN 978-3-030-14315-2 ISBN 978-3-030-14316-9 (eBook)
https://doi.org/10.1007/978-3-030-14316-9

Library of Congress Control Number: 2019932720

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

We are here to help you learn R for marketing research and analytics.

R is a great choice for marketing analysts. It offers unsurpassed capabilities for fitting statistical models. It is extensible and able to process data from many different systems, in a variety of forms, for both small and large data sets. The R ecosystem includes the widest available range of established and emerging statistical methods and visualization techniques. Yet its use in marketing lags other fields such as statistics, econometrics, psychology, and bioinformatics. With your help, we hope to change that!

This book is designed for two audiences: practicing marketing researchers and analysts who want to learn R and students or researchers from other fields who wish to review selected marketing topics in an R context.

What are the prerequisites? Simply that you are interested in R for marketing, are conceptually familiar with basic statistical models such as linear regression, and are willing to engage in hands-on learning. This book will be particularly helpful to analysts who have some degree of programming experience and wish to learn R. In Chap. 1, we describe additional reasons to use R (and a few reasons perhaps *not* to use R).

The *hands-on* part is important. We teach concepts gradually in a sequence across the first seven chapters and ask you to *type* our examples as you work; this book is *not* a cookbook-style reference. We spend some time (as little as possible) in Part I on the basics of the R language and then turn in Part II to applied, real-world marketing analytics problems. Part III presents a few advanced marketing topics. Every chapter shows the power of R, and we hope each one will teach you something new and interesting.

Specific features of this book are:

- It is organized around marketing research tasks. Instead of generic examples, we put methods into the context of marketing questions.
- We presume only basic statistics knowledge and use a minimum of mathematics. This book is designed to be approachable for practitioners and does not dwell on equations or mathematical details of statistical models (although we give references to those texts).
- This is a didactic book that explains statistical concepts and the R code. We want you to understand what we're doing and learn how to avoid common problems in both statistics and R. We intend the book to be *readable* and to fulfill a different need than references and cookbooks available elsewhere.
- The applied chapters demonstrate progressive model building. We do not present "the answer" but instead show how an analyst might realistically conduct analyses in successive steps where multiple models are compared for statistical strength and practical utility.
- The chapters include visualization as a part of core analyses. We don't regard visualization as a standalone topic; rather, we believe it is an integral part of data exploration and model building.
- You will learn more than just R. In addition to core models, we include topics such as structural models and transaction analysis that may be new and useful even for experienced analysts.
- The book reflects both traditional and Bayesian approaches. Core models are presented with traditional (frequentist) methods, while later sections introduce Bayesian methods for linear models and conjoint analysis.
- Most of the analyses use simulated data, which provides practice in the R language along with additional insight into the structure of marketing data. If you are inclined, you can change the data simulation and see how the statistical models are affected.
- Where appropriate, we call out more advanced material on programming or models so that you may either skip it or read it, as you find appropriate. These sections are indicated by * in their titles (such as *This is an advanced section**).

What do we *not* cover? For one, this book teaches *R* for marketing and does not teach marketing research in itself. We discuss many marketing topics but omit others that would repeat analytic methods. As noted above, we approach statistical models from a conceptual point of view and skip the mathematics. A few specialized topics have been omitted due to complexity and space; these include customer lifetime value models and econometric time series models. In the R language, we do not cover the "tidyverse" (Sect. 1.5) because it is an optional part of the language and would complicate the learning process. Overall, we believe the topics here represent a great sample of marketing research and analytics practice. If you learn to perform these, you'll be well equipped to apply R in many areas of marketing.

Why are we the right teachers? We've used R and its predecessor S for a combined 35 years since 1997, and it is our primary analytics platform. We perform marketing analyses of all kinds in R, ranging from simple data summaries to complex analyses involving thousands of lines of custom code and newly created models.

We've also taught R to many people. This book grew from courses the authors have presented at American Marketing Association (AMA) events including the Academy of Marketing Analytics at Emory University and several years of the Advanced Research Techniques Forum (ART Forum). As noted in our Acknowledgements below, we have taught R to students in many workshops at universities and firms. At last count, more than 40 universities used the first edition in their marketing analytics courses. All of these students' and instructors' experiences have helped to improve the book.

What's New in the Second Edition

This second edition focuses on making the book more useful for students, self-learners, and instructors. The code has proven to be very stable. Except for one line (updated at the book's Web site), all of the code and examples from the first edition still work more than four years later. We have added one chapter, and otherwise, the marketing topics and statistical models are the same as in the first edition. The primary changes in this edition are:

- New **exercises** appear at the end of each chapter. Several of these use real-world data, and there are example solutions at the book's Web site.
- A **new chapter** discusses analysis of behavior sequences (Chap. 14) using Markov chains. These methods are applicable to many sources of behavioral and other data comprising sequences of discrete events, such as application usage, purchases, and life events, as well as non-marketing data including physical processes and genomic sequences. We use a published Web server log file to demonstrate the methods applied to real data.
- Classroom **slides** are available for instructors and self-learners at the book's Web site. These include the slides themselves, the raw code that they discuss, and Rmarkdown and LaTeX files that generate the slides and may be edited for your own use.
- For our various data sets, we present **additional details** about how such data might be acquired. For example, when a data set represents consumer survey data, we describe how the data might be gathered and a brief description of typical survey items.
- A new appendix describes options for **reproducible research** in R and explains the basics of R Notebooks (Appendix B). R Notebooks are a simple yet powerful way to create documents in R with integrated code, graphics, and formatted text. They may be used to create documents as simple as homework exercises,

or as complex as final deliverable reports for clients, with output in HTML, PDF, or Microsoft Word formats.
- We have updated other **content** as needed. This includes additional explanations, code, and charts where warranted; up-to-date references; and correction of minor errors.

Acknowledgements

We thank many people who made this book possible. First are many participants in our workshops and classes over the years, including students at Drexel University, Boston University, Temple University, the Wharton School of the University of Pennsylvania, and the University of Washington; practitioners at Google and URBN, Inc.; and workshop attendees at the Advanced Research Techniques Forum (ART Forum), the Sawtooth Software Conference, and the Academy of Marketing Analytics at Emory University. They provided valuable feedback, and we hope their questions and experiences will benefit you.

In the marketing academic and practitioner community, we had valuable feedback from Ken Deal, Fred Feinberg, Shane Jensen, Jake Lee, Hui Lin, Dave Lyon, Bruce McCullough, Bernd Skiera, Hiroshi Torii, and Randy Zwitch. Many readers of the book's first edition sent notes, reviewed it online, and reported errata. We appreciated the supportive and helpful comments.

Chris's colleagues in the research community at Google provided extensive feedback on portions of the book. We thank the following current and former Googlers: Eric Bahna, Mario Callegaro, Marianna Dizik, Rohan Gifford, Tim Hesterberg, Shankar Kumar, Norman Lemke, Paul Litvak, Katrina Panovich, Joe Paxton, Marta Rey-Babarro, Kerry Rodden, Dan Russell, Angela Schörgendorfer, Jason Schwarz, Steven Scott, Rebecca Shapley, Bob Silverstein, Gill Ward, John Webb, Ercan Yildiz, and Yori Zwols for their encouragement and comments.

The staff and editors at Springer helped us smooth the process, especially Hannah Bracken and Jon Gurstelle for the first edition, and Lorraine Klimowich and Nicholas Philipson for the second edition. The UseR! series editors, Robert Gentleman, Kurt Hornik, and Giovanni Parmigiani, provided early feedback. They have nurtured a superb series of R texts, and we are honored to contribute to it.

Much of this book was written in public and university libraries, and we thank them for their hospitality alongside their literary resources. Portions of the book were written during pleasant days at the New York Public Library, Christoph Keller Jr. Library at the General Theological Seminary (New York), New Orleans Public Library, British Library (London), University of California San Diego Giesel Library, University of Washington Suzzallo and Allen Libraries, Sunnyvale Public Library (California), West Osceola Public Library (Florida), Howard County Library System (Maryland), Montgomery County Public Libraries (Maryland),

Kennett Library (Pennsylvania), Utica Public Library (Michigan), Clinton-Macomb Public Library (Michigan), San Juan Island Library (Washington), and in the dining hall of Holden, Washington (see Sect. 2.4.4). We give special thanks to the Tokyo Metropolitan Central Library, where the first words, code, and outline were written, along with much more in both the first and second editions.

Our families supported us in weekends and nights of editing, and they endured more discussion of R than is fair for any layperson. Thank you, Cristi, Maddie, Jeff, and Zoe.

Most importantly, we thank *you*, the reader. We're glad you've decided to investigate R, and we hope to repay your effort. Let's start!

Seattle, WA, USA/New York, NY, USA — Chris Chapman
Philadelphia, PA, USA — Elea McDonnell Feit
January 2019

Contents

Part I
Basics of R

Chapter 1
Welcome to R

1.1 What is R?

As a marketing analyst, you have no doubt heard of R. You may have tried R and become frustrated and confused, after which you returned to other tools that are "good enough." You may know that R uses a command line and dislike that. Or you may be convinced of R's advantages for experts but worry that you don't have time to learn or use it.

We are here to help! Our goal is to present *just the essentials*, in the *minimal necessary time*, with *hands-on learning* so you will come up to speed as quickly as possible to be productive in R. In addition, we'll cover a few advanced topics that demonstrate the power of R and might teach advanced users some new skills.

A key thing to realize is that *R is a programming language*. It is *not* a "statistics program" like SPSS, SAS, JMP, or Minitab, and doesn't wish to be one. The official R Project describes R as "a language and environment for statistical computing and graphics." Notice that "language" comes first, and that "statistical" is coequal with "graphics." R is a great programming language for doing statistics. The inventor of the underlying language, John Chambers received the 1998 Association for Computing Machinery (ACM) Software System Award for a system that "will forever alter the way people analyze, visualize, and manipulate data …" [5].

R was based on Chambers's preceding S language (S as in "statistics") developed in the 1970s and 1980s at Bell Laboratories, home of the UNIX operating system and the C programming language. S gained traction among analysts and academics in the 1990s as implemented in a commercial software package, S-PLUS. Robert Gentleman and Ross Ihaka wished to make the S approach more widely available and offered R as an open source project starting in 1997.

Since then, the popularity of R has grown geometrically. The real magic of R is that its users are able to contribute developments that enhance R with everything from additional core functions to highly specialized methods. And many do contribute!

C. Chapman and E. M. Feit, *R For Marketing Research and Analytics*, Use R!,
https://doi.org/10.1007/978-3-030-14316-9_1

Today there are over 13,000 packages of add on functionality available for R (see http://cran.r-project.org/web/packages for the latest count).

If you have experience in programming, you will appreciate some of R's key features right away. If you're new to programming, this chapter describes why R is special and Chap. 2 introduces the fundamentals of programming in R.

1.2 Why R?

There are many reasons to learn and use R. It is the platform of choice for the largest number of statisticians who create new analytics methods, so emerging techniques are often available first in R. R is rapidly becoming the default educational platform in university statistics programs and is spreading to other disciplines such as economics and psychology.

For analysts, R offers the largest and most diverse set of analytic tools and statistical methods. It allows you to write analyses that can be reused and that extend the R system itself. It runs on most operating systems and interfaces well with data systems such as online data and SQL databases. R offers beautiful and powerful plotting functions that are able to produce graphics vastly more tailored and informative than typical spreadsheet charts. Putting all of those together, R can vastly improve an analyst's overall productivity. Elea knows an enterprising analyst who used R to automate the process of downloading data and producing a formatted monthly report. The automation saved him almost 40 h of work each month…which he didn't tell his manager for a few months!

Then there is the community. Many R users are enthusiasts who love to help others and are rewarded in turn by the simple joy of solving problems and the fact that they often learn something new. R is a dynamic system created by its users, and there is always something new to learn. Knowledge of R is a valuable skill in demand for analytics jobs at a growing number of top companies.

R code is also inspectable; you may choose to trust it, yet you are also free to verify. All of its core code and most packages that people contribute are open source. You can examine the code to see exactly how analyses work and what is happening under the hood.

Finally, R is free. It is a labor of love and professional pride for the R Core Development Team, which includes eminent statisticians and computer scientists. As with all masterpieces, the quality of their devotion is evident in the final work.

1.3 Why Not R?

What's not to love? No doubt you've observed that not everyone in the world uses R. Being R-less is unimaginable to us yet there are reasons why some analysts might not want to use it.

One reason not to use R is this: until you've mastered the basics of the language, many simple analyses are cumbersome to do in R. If you're new to R and want a table of means, cross-tabs, or a t-test, it may be frustrating to figure out how to get them. R is about power, flexibility, control, iterative analyses, and cutting-edge methods, not point-and-click deliverables.

Another reason is if you do not like programming. If you're new to programming, R is a great place to start. But if you've tried programming before and didn't enjoy it, R will be a challenge as well. Our job is to help you as much as we can, and we will try hard to teach R to you. However, not everyone enjoys programming. On the other hand, if you're an experienced coder R will seem simple (perhaps deceptively so), and we will help you avoid a few pitfalls.

Some companies and their information technology or legal departments are skeptical of R because it is open source. It is common for managers to ask, "If it's free, how can it be good?" There are many responses to that, including pointing out the hundreds of books on R, its citation in peer-reviewed articles, and the list of eminent contributors (in R, run the `contributors()` command and web search some of them). Or you might try the engineer's adage: "It can be good, fast, or cheap: pick 2." R is good and cheap, but not fast, insofar as it requires time and effort to master.

As for R being free, you should realize that contributors to R actually do derive benefit; it just happens to be non-monetary. They are compensated through respect and reputation, through the power their own work gains, and by the contributions back to the ecosystem from other users. This is a rational economic model even when the monetary price is zero.

A final concern about R is the unpredictability of its ecosystem. With packages contributed by thousands of authors, there are priceless contributions along with others that are mediocre or flawed. The downside of having access to the latest developments is that many will not stand the test of time. It is up to you to determine whether a method meets your needs, and you cannot always rely on curation or authorities to determine it for you (although you will rapidly learn which authors and which experts' recommendations to trust). If you trust your judgment, this situation is no different than with any software. *Caveat emptor*.

We hope to convince you that for many purposes, the benefits of R outweigh the difficulties.

1.4 When R?

There are a few common use cases for R:

- You want access to methods that are newer or more powerful than available elsewhere. Many R users start for exactly that reason; they see a method in a journal article, conference paper, or presentation, and discover that the method is available only in R.
- You need to run an analysis many, many times. This is how the first author (hereafter, Chris) started his R journey; for his dissertation, he needed to bootstrap existing methods in order to compare their typical results to those of a new machine learning model. R is perfect for model iteration.
- You need to apply an analysis to multiple data sets. Because everything is scripted, R is great for analyses that are repeated across datasets. It even has tools available for automated reporting.
- You need to develop a new analytic technique or wish to have perfect control and insight into an existing method. For many statistical procedures, R is easier to code than other programming languages.
- Your manager, professor, or coworker is encouraging you to use R. We've influenced students and colleagues in this way and are happy to report that a large number of them are enthusiastic R users today.

By showing you the power of R, we hope to convince you that your current tools are *not* perfectly satisfactory. Even more deviously, we hope to rewrite your expectations about what *is* satisfactory.

1.4.1 R Versus Python, Julia, and Others

If you are new to programming, you might wonder whether to learn R or Python ... or Julia, Matlab, Ruby, Go, Java, C++, Fortran, or others. Each language has a somewhat unique value.

For interactive analyses and data visualization, with access to the latest developments in statistics, R is unmatched. On the other hand, if you want your analytic work to go into *production* and integrate with a larger system (such as a product or a web site), Python is a great choice [176]. If high performance is essential to you, such as working with massive data sets or models with high mathematical complexity, Julia is an excellent option [210]. Go is also designed for massive scalability.

Another factor is whether you want to program more generally beyond analytics, such as writing apps. Python is an excellent general purpose language. Many find Python more approachable than C or C++, and it has broader support for statistics and analytics than Go, Java, or Ruby.

If you often do a lot of directly mathematical work—such as writing equations for models—then R is a fine choice, although you might be more comfortable with Julia, Matlab, or even venerable Fortran (whose name abbreviates *for*mula *tran*slation).

If you work with other programmers, you might want to choose a language they know, so they can help you. At the same time, most languages interact well with others. For example, it is easy to write analytic code in R and to access it from Python (and vice versa). Similarly, it is easy in R to include code from C, C++ [49], Fortran, and SQL (Appendix C.1.4), among others. Many programmers end up using several languages and find that transitioning among them is not difficult.

In short, for analyses with high flexibility and a straightforward programming environment, R is a great choice.

1.5 Which R? Base or Tidy?

As the R language has evolved, it has begun to show diversity of syntax and commands that is analogous to linguistic dialects. In recent years, a significant distinction has appeared: *base R* (the core language) versus the *tidyverse*. The tidyverse is a vast set of add-on capabilities that extend base R with many new operators and functions, inspired by a powerful philosophy of data organization (Wickham and Grolemund [200]). It provides simple and efficient ways to manipulate, aggregate, and slice data; to visualize data; and to perform a wide range of analytic tasks from summarization to data mining.

Despite the power of the tidyverse, in this book we instead focus on programming in base R. Why? For several reasons:

- Fluency in base R is essential for all R users, so you must learn it. It is the basis for all R commands, packages, language structures, and analyses. Base R will always work, even when a particular section of code using it is less compact than a tidyverse alternative.
- The tidyverse introduces functions that duplicate many capabilities and approaches of base R, such as different commands to summarize data. We believe it is easier to learn a single dialect of a language first rather than to learn two dialects simultaneously.
- There are significant syntactic differences in the tidyverse. In particular, the tidyverse often uses "pipe" operators that cause program flow to be read *left-to-right*, whereas base R operations read *right-to-left*, as do most programming languages. In earlier chapters where we teach programming, covering both styles would be overly complicated for new programmers. (Imagine trying to read English in variable direction from one sentence to the next. For example, read this in reverse: Context in confusing *is* it but, read to difficult not is sentence this.)

- Many of the analyses in later chapters would not benefit from the tidyverse; they use packages that depend only on base R. Thus, learning the tidyverse approach would have relatively little benefit as the book progresses.
- Whereas this book focuses on statistical approaches to marketing problems, at the time of writing the tidyverse is optimized more for data manipulation and visualization. Thus we view it as complementary but somewhat outside the focus of this book.

There is one situation in which we recommend that you start with the tidyverse instead of base R: if your interest is primarily in routine data manipulation and visualization with little or no focus on statistical methods. For example, if you expect to produce many reports and charts summarizing data, and are not especially interested in statistical modeling or programming, the tidyverse approach may be especially productive for you at the beginning. Then you can learn more about base R later.

For most users, we recommend to become fluent in base R. With that under your belt, we recommend then to learn the tidyverse approach from a text that focuses on it, such as the excellent text from Wickham and Grolemund [200].

1.6 Using This Book

This book is intended to be *didactic* and *hands-on*, meaning that we want to teach you about R and the models we use in plain English, and we expect you to engage with the code interactively in R. It is designed for you to type the commands as you read. (We also provide code files for download from the book's web site; see Sect. 1.6.3 below.)

1.6.1 About the Text

R commands for you to run are presented in code blocks like this:

```
> citation()

To cite R in publications use:

  R Core Team (2018). R: A language and environment for statistical
  computing. R Foundation for Statistical Computing, Vienna, Austria.
  URL https://www.R-project.org/.
...
```

We describe these code blocks and interacting with R in Chap. 2. The code generally follows the Google style guide for R (available at https://google.github.io/styleguide/Rguide.xml) except when we thought a deviation might make the code or text clearer. (As you learn R, you will wish to make your code readable; the Google guide is very useful for code formatting.)

When we refer to R commands, add-on packages, or data in the text outside of code blocks, we set the names in monospace type like this: `citation()`. We include parentheses on function (command) names to indicate that they are functions, such as the `summary()` function (Sect. 2.4.1), as opposed to an object such as the `Groceries` data set (Sect. 12.2.1).

When we introduce or define significant new concepts, we set them in italic, such as *vectors*. Italic is also used simply for *emphasis*.

We teach the R language progressively throughout the book, and much of our coverage of the language is blended into chapters that cover marketing topics and statistical models. In those cases, we present crucial language topics in *Language Brief* sections (such as Sect. 3.4.5). To learn as much as possible about the R language, you'll need to read the Language Brief sections even if you only skim the surrounding material on statistical models.

Some sections cover deeper details or more advanced topics, and may be skipped. We note those with an asterisk in the section title, such as *Learning More**.

1.6.2 About the Data

Most of the data sets that we analyze in this book are *simulated* data sets. They are created with R code to have a specific structure. This has several advantages:

- It allows us to illustrate analyses where there is no publicly available marketing data. This is valuable because few firms share their proprietary data for analyses such as segmentation.
- It allows the book to be more self-contained and less dependent on data downloads.
- It makes it possible to alter the data and rerun analyses to see how the results change.
- It lets us teach important R skills for handling data, generating random numbers, and looping in code.
- It demonstrates how one can write analysis code while waiting for real data. When the final data arrives, you can run your code on the new data.

There are two exceptions to our usage of simulated data. First, many end-of-chapter exercises use an actual e-commerce data set (Sect. 3.8.1). Second, we use actual store transaction data in Chap. 12; such data is complex to create and appropriate data has been published [23].

We recommend you work through data simulation sections where they appear; they are designed to teach R and to illustrate points that are typical of marketing data. However, when you need data quickly to continue with a chapter, it is available for download as noted in the next section and again in each chapter.

Whenever possible you should also try to perform the analyses here with your own data sets. We work with data in every chapter, but the best way to learn is to adapt

the analyses to other data and work through the issues that arise. Because this is an educational text, not a cookbook, and because R can be slow going at first, we recommend to conduct such parallel analyses on tasks where you are not facing urgent deadlines.

At the beginning, it may seem overly simple to repeat analyses with your own data, but when you try to apply an advanced model to another data set, you'll be much better prepared if you've practiced with multiple data sets all along. The sooner you apply R to your own data, the sooner you will be productive in R.

1.6.3 Online Material

This book has a companion website: http://r-marketing.r-forge.r-project.org. The website exists primarily to host the R code and data sets for download, although we encourage you to use those sparingly; you'll learn more if you type the code and create the data sets by simulation as we describe.

On the website, you'll find:

- A welcome page for news and updates: http://r-marketing.r-forge.r-project.org
- Code files in .R (text) format: http://r-marketing.r-forge.r-project.org/code
- Slides for classroom usage, along with R Markdown files used to create the slides: http://r-marketing.r-forge.r-project.org/slides
- Copies of data sets that are used in the book: http://r-marketing.r-forge.r-project.org/data. These are generally downloaded directly into R using the `read.csv()` command (you'll see that command in Sect. 2.6.2, and will find code for an example download in Sect. 3.1)
- A ZIP file containing all of the data and code files: http://r-marketing.r-forge.r-project.org/data/chapman-feit-rintro.zip

Links to online data are provided in the form of shortened `goo.gl` links to save typing. More detail on the online materials and ways to access the data are described in Appendix E.

1.6.4 When Things Go Wrong

When you learn something as complex as R or new statistical models, you will encounter many large and small warnings and errors. Also, the R ecosystem is dynamic and things will change after this book is published. We don't wish to scare you with a list of concerns, but we do want you to feel reassured about small discrepancies and to know what to do when larger bugs arise. Here are a few things to know and to try if one of your results doesn't match this book:

- **With R**. The basic error correction process when working with R is to check everything very carefully, especially parentheses, brackets, and upper- or lowercase letters. If a command is lengthy, deconstruct it into pieces and build it up again (we show examples of this along the way).
- **With packages** (add-on libraries). Packages add functionality to R and are regularly updated. Sometimes they change how they work, or may not work at all for a while. Some are very stable while others change often. If you have trouble installing one, do a web search for the error message. If output or details are slightly different than we show, don't worry about it. The error `"There is no package called ..."` indicates that you need to install the package (Sect. 2.2). For other problems, see the remaining items here or check the package's help file (Sect. 2.4.2).
- **With R warnings and errors**. An R "warning" is often informational and does not necessarily require correction. We call these out as they occur with our code, although sometimes they come and go as packages are updated. If R gives you an "error," that means something went wrong and needs to be corrected. In that case, try the code again, or search online for the error message. Also check the errata page on the book's website (Sect. 1.6.3), where we post any necessary updates to the code.
- **With data**. Our data sets are simulated and are affected by random number sequences. If you generate data and it is slightly different, try it again from the beginning; or load the data from the book's website (Sect. 1.6.3).
- **With models**. There are three things that might cause statistical estimates to vary: slight differences in the data (see the preceding item), changes in a package that lead to slightly different estimates, and statistical models that employ random sampling. If you run a model and the results are very similar but slightly different, you can assume that one of these situations occurred. Just proceed.
- **With output**. Packages sometimes change the information they report. The output in this book was current at the time of writing, but you can expect some packages will report things slightly differently over time.
- **With names that can't be located**. Sometimes packages change the function names they use or the structure of results. If you get a code error when trying to extract something from a statistical model, check its help file (Sect. 2.4.2); it may be that something has changed names.
- **When things turn out differently than expected**. For various reasons, R or RStudio may give results or errors that differ from previous occasions. For example, a plot command might not work although it has worked in the past. If none of the preceding tips help, we suggest to exit R or RStudio altogether, restart it, and repeat your steps from the beginning of a section.

Our overall recommendation is this. If a difference is small—such as the difference between a mean of 2.08 and 2.076, or a p-value of 0.726 versus 0.758—don't worry too much about it; you can usually safely ignore these. If you find a large difference—such as a statistical estimate of 0.56 instead of 31.92—try the code block again in the book's code file (Sect. 1.6.3).