R Markdown
The Definitive Guide
The R Series

Series Editors
John M. Chambers, Department of Statistics Stanford University Stanford, California, USA
Torsten Hothorn, Division of Biostatistics University of Zurich Switzerland
Duncan Temple Lang, Department of Statistics University of California, Davis, California, USA
Hadley Wickham, RStudio, Boston, Massachusetts, USA

Recently Published Titles

bookdown: Authoring Books and Technical Documents with R Markdown
Yihui Xie

Testing R Code
Richard Cotton

R Primer, Second Edition
Claus Thorn Ekstrøm

Flexible Regression and Smoothing: Using GAMLSS in R
Mikis D. Stasinopoulos, Robert A. Rigby, Gillian Z. Heller, Vlasios Voudouris, Fernanda De Bastiani

The Essentials of Data Science: Knowledge Discovery Using R
Graham J. Williams

blogdown: Creating Websites with R Markdown
Yihui Xie, Alison Presmanes Hill, Amber Thomas

Handbook of Educational Measurement and Psychometrics Using R
Christopher D. Desjardins, Okan Bulut

Oscar Perpinan Lamigueiro

Reproducible Finance with R
Jonathan K. Regensteirn, Jr

R Markdown: The Definitive Guide
Yihui Xie, J.J. Allaire, Garrett Grolemund

For more information about this series, please visit:
https://www.crcpress.com/go/the-r-series
R Markdown
The Definitive Guide

Yihui Xie
J. J. Allaire
Garrett Grolemund
To Jung Jae-sung (1982 – 2018),
a remarkably hard-working badminton player with a remarkably
simple playing style
## Contents

List of Tables .......................... xvii  
List of Figures .......................... xix  
Preface ................................... xxiii  
About the Authors ......................... xxxiii  
I Get Started .............................. 1  
1 Installation .............................. 3  
2 Basics ................................ 5  
2.1 Example applications ..................... 9  
2.1.1 Airbnb’s knowledge repository .......... 9  
2.1.2 Homework assignments on RPubs ......... 10  
2.1.3 Personalized mail ..................... 10  
2.1.4 2017 Employer Health Benefits Survey .... 11  
2.1.5 Journal articles .................... 12  
2.1.6 Dashboards at eeloo .................. 12  
2.1.7 Books ............................... 12  
2.1.8 Websites ............................ 13  
2.2 Compile an R Markdown document .......... 14  
2.3 Cheat sheets ........................... 16  
2.4 Output formats ......................... 16  
2.5 Markdown syntax ....................... 19  
2.5.1 Inline formatting ..................... 19  
2.5.2 Block-level elements .................. 20  
2.5.3 Math expressions .................... 23  
2.6 R code chunks and inline R code .......... 24  
2.6.1 Figures ............................. 27  
2.6.2 Tables ............................. 29  
2.7 Other language engines ................... 30
## Output Formats

### Documents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>HTML document</td>
<td>49</td>
</tr>
<tr>
<td>3.1.1</td>
<td>Table of contents</td>
<td>50</td>
</tr>
<tr>
<td>3.1.2</td>
<td>Section numbering</td>
<td>51</td>
</tr>
<tr>
<td>3.1.3</td>
<td>Tabbed sections</td>
<td>51</td>
</tr>
<tr>
<td>3.1.4</td>
<td>Appearance and style</td>
<td>52</td>
</tr>
<tr>
<td>3.1.5</td>
<td>Figure options</td>
<td>54</td>
</tr>
<tr>
<td>3.1.6</td>
<td>Data frame printing</td>
<td>55</td>
</tr>
<tr>
<td>3.1.7</td>
<td>Code folding</td>
<td>56</td>
</tr>
<tr>
<td>3.1.8</td>
<td>MathJax equations</td>
<td>57</td>
</tr>
<tr>
<td>3.1.9</td>
<td>Document dependencies</td>
<td>58</td>
</tr>
<tr>
<td>3.1.10</td>
<td>Advanced customization</td>
<td>59</td>
</tr>
<tr>
<td>3.1.11</td>
<td>Shared options</td>
<td>62</td>
</tr>
<tr>
<td>3.1.12</td>
<td>HTML fragments</td>
<td>62</td>
</tr>
<tr>
<td>3.2</td>
<td>Notebook</td>
<td>63</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Using Notebooks</td>
<td>63</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Saving and sharing</td>
<td>72</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Notebook format</td>
<td>73</td>
</tr>
<tr>
<td>3.3</td>
<td>PDF document</td>
<td>78</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Table of contents</td>
<td>79</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Figure options</td>
<td>80</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Data frame printing</td>
<td>80</td>
</tr>
<tr>
<td>3.3.4</td>
<td>Syntax highlighting</td>
<td>81</td>
</tr>
<tr>
<td>3.3.5</td>
<td>LaTeX options</td>
<td>81</td>
</tr>
<tr>
<td>3.3.6</td>
<td>LaTeX packages for citations</td>
<td>82</td>
</tr>
<tr>
<td>3.3.7</td>
<td>Advanced customization</td>
<td>83</td>
</tr>
</tbody>
</table>
12.5.4 RStudio addins ............................................. 216
12.6 Publishing ................................................... 216
12.6.1 RStudio Connect ......................................... 217
12.6.2 Other services ............................................ 217
12.6.3 Publishers ................................................ 217

13 Journals ......................................................... 219
13.1 Get started .................................................. 219
13.2 Articles templates ......................................... 222
13.3 Using a template ........................................... 222
13.4 LaTeX content .............................................. 224
13.5 Linking with bookdown ................................. 224
13.6 Contributing templates ................................. 225

14 Interactive Tutorials .......................................... 227
14.1 Get started .................................................. 227
14.2 Tutorial types .............................................. 229
14.3 Exercises .................................................... 230
  14.3.1 Solutions .............................................. 230
  14.3.2 Hints .................................................. 231
14.4 Quiz questions ............................................. 233
14.5 Videos ..................................................... 234
14.6 Shiny components ........................................ 235
14.7 Navigation and progress tracking .................. 235

IV Other Topics .................................................. 237
15 Parameterized reports ...................................... 239
  15.1 Declaring parameters .................................. 239
  15.2 Using parameters ....................................... 240
  15.3 Knitting with parameters .............................. 241
    15.3.1 The Knit button .................................. 241
    15.3.2 Knit with custom parameters .................. 241
    15.3.3 The interactive user interface ................ 242
  15.4 Publishing .............................................. 244

16 HTML Widgets ................................................ 247
  16.1 Overview ................................................ 247
  16.2 A widget example (sigma.js) ....................... 248
    16.2.1 File layout ....................................... 249
    16.2.2 Dependencies .................................. 250
16.2.3 R binding ........................................ 250
16.2.4 JavaScript binding ................................. 252
16.2.5 Demo .............................................. 255
16.3 Creating your own widgets ......................... 256
16.3.1 Requirements ..................................... 256
16.3.2 Scaffolding ....................................... 257
16.3.3 Other packages .................................... 258
16.4 Widget sizing ......................................... 258
16.4.1 Specifying a sizing policy ........................ 259
16.4.2 JavaScript resize method ......................... 260
16.5 Advanced topics ...................................... 264
16.5.1 Data transformation ............................... 264
16.5.2 Passing JavaScript functions .................... 269
16.5.3 Custom widget HTML ............................. 270
16.5.4 Create a widget without an R package ......... 271

17 Document Templates ........................................ 273
17.1 Template structure .................................... 274
17.2 Supporting files ...................................... 276
17.3 Custom Pandoc templates ........................... 277
17.4 Sharing your templates ............................... 278

18 Creating New Formats ..................................... 279
18.1 Deriving from built-in formats ....................... 279
18.2 Fully custom formats ................................... 280
18.3 Using a new format ..................................... 282

19 Shiny Documents .......................................... 283
19.1 Getting started ......................................... 283
19.2 Deployment ........................................... 287
  19.2.1 ShinyApps.io .................................. 287
  19.2.2 Shiny Server / RStudio Connect ............... 288
19.3 Embedded Shiny apps .................................. 289
  19.3.1 Inline applications ............................... 289
  19.3.2 External applications ............................ 290
19.4 Shiny widgets ........................................... 290
  19.4.1 The shinyApp() function ........................ 290
  19.4.2 Example: k-Means clustering ................... 291
  19.4.3 Widget size and layout ......................... 293
19.5 Multiple pages .......................................... 293
19.6 Delayed rendering ............................................. 294
19.7 Output arguments for render functions ......................... 294
  19.7.1 A caveat .................................................. 297

Bibliography 299

Index 303
List of Tables

3.1 The possible values of the df_print option for the html_document format ........................................ 55
3.2 The options for paged HTML tables. ................................ 56
3.3 The possible values of the df_print option for the pdf_document format ........................................ 81
3.4 Available top-level YAML metadata variables for LaTeX output ......................................................... 82
3.5 Markdown variants for some popular publishing systems ................................................................. 89
8.1 The currently supported reveal.js plugins. ............... 172
15.1 Possible input types and the associated Shiny functions for parameterized reports ............................................. 245
16.1 Options that can be specified within a sizing policy .... 261
List of Figures

2.1 A minimal R Markdown example in RStudio. . . . . . . . . . . 7
2.2 The output document of the minimal R Markdown example
in RStudio. . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
2.3 A screenshot of RPubs.com that contains some homework ass-
ignments submitted by students. . . . . . . . . . . . . . . . . 11
2.4 The output formats listed in the dropdown menu on the RStu-
dio toolbar. . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
2.5 Two plots side-by-side. . . . . . . . . . . . . . . . . . . . . . 28
2.6 The R Markdown hex logo. . . . . . . . . . . . . . . . . . . . 29
2.7 An R Markdown document with a leaflet map widget. . . . . 43
2.8 An R Markdown document with a Shiny widget. . . . . . . 45

3.1 Traditional tabs and pill tabs on an HTML page. . . . . . . 52
3.2 A paged table in the HTML output document. . . . . . . 56
3.3 An R Notebook example. . . . . . . . . . . . . . . . . . . . . 64
3.4 Send the R code chunk output to the console. . . . . . . . 65
3.5 Select the code to split into a new chunk. . . . . . . . . . 65
3.6 Insert a new chunk from the code selected before. . . . . 66
3.7 Insert a new chunk from the code selected before. . . . . 67
3.8 Output from an inline R expression in the notebook. . . . 69
3.9 Execute a code chunk read from an external R script. . . 70
3.10 Preview a notebook. . . . . . . . . . . . . . . . . . . . . . 71
3.11 Errors in a notebook. . . . . . . . . . . . . . . . . . . . . . 72

4.1 Two sample slides in an ioslides presentation. . . . . . . 94
4.2 Two sample slides in a Slidy presentation. . . . . . . . . 104
4.3 Two sample slides in a Beamer presentation. . . . . . . 107
4.4 Two sample slides with the AnnArbor theme in Beamer. . 108
4.5 A sample slide in a PowerPoint presentation. . . . . . . 111

5.1 A quick example of the dashboard layout. . . . . . . . . . 120
5.2 Multiple pages on a dashboard. . . . . . . . . . . . . . . 122
14.5 Keeping track of the student’s progress in a tutorial. . . . . . 236

15.1 Input parameter values interactively for parameterized re-
ports. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 243
15.2 Custom controls for parameters. . . . . . . . . . . . . . . . 245

16.1 A graph generated using the sigma.js library and the sigma
package. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 249

17.1 Selecting R Markdown templates within RStudio. . . . . . 274

19.1 Create a new Shiny document in RStudio. . . . . . . . . . . 284
19.2 Increase the number of rows in the table in a Shiny document. 285
19.3 Change the number of bins of a histogram in a Shiny docu-
ment. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 286
19.4 Deploy a Shiny document to ShinyApps.io. . . . . . . . . . . 288
19.5 A Shiny widget to apply k-Means clustering on a dataset. . 291
The document format “R Markdown” was first introduced in the \texttt{knitr} package (Xie, 2015, 2018d) in early 2012. The idea was to embed code chunks (of R or other languages) in Markdown documents. In fact, \texttt{knitr} supported several authoring languages from the beginning in addition to Markdown, including LaTeX, HTML, AsciiDoc, reStructuredText, and Textile. Looking back over the five years, it seems to be fair to say that Markdown has become the most popular document format, which is what we expected. The simplicity of Markdown clearly stands out among these document formats.

However, the original version of Markdown invented by John Gruber\footnote{https://en.wikipedia.org/wiki/Markdown} was often found overly simple and not suitable to write highly technical documents. For example, there was no syntax for tables, footnotes, math expressions, or citations. Fortunately, John MacFarlane created a wonderful package named Pandoc \url{http://pandoc.org} to convert Markdown documents (and many other types of documents) to a large variety of output formats. More importantly, the Markdown syntax was significantly enriched. Now we can write more types of elements with Markdown while still enjoying its simplicity.

In a nutshell, R Markdown stands on the shoulders of \texttt{knitr} and Pandoc. The former executes the computer code embedded in Markdown, and converts R Markdown to Markdown. The latter renders Markdown to the output format you want (such as PDF, HTML, Word, and so on).

The \texttt{rmarkdown} package (Allaire et al., 2018c) was first created in early 2014. During the past four years, it has steadily evolved into a relatively complete ecosystem for authoring documents, so it is a good time for us to provide a definitive guide to this ecosystem now. At this point, there are a large number of tasks that you could do with R Markdown:

\begin{itemize}
  \item Compile a single R Markdown document to a report in different formats, such as PDF, HTML, or Word.
\end{itemize}
- Create notebooks in which you can directly run code chunks interactively.
- Make slides for presentations (HTML5, LaTeX Beamer, or PowerPoint).
- Produce dashboards with flexible, interactive, and attractive layouts.
- Build interactive applications based on Shiny.
- Write journal articles.
- Author books of multiple chapters.
- Generate websites and blogs.

There is a fundamental assumption underneath R Markdown that users should be aware of: we assume it suffices that only a limited number of features are supported in Markdown. By “features”, we mean the types of elements you can create with native Markdown. The limitation is a great feature, not a bug. R Markdown may not be the right format for you if you find these elements not enough for your writing: paragraphs, (section) headers, block quotations, code blocks, (numbered and unnumbered) lists, horizontal rules, tables, inline formatting (emphasis, strikeout, superscripts, subscripts, verbatim, and small caps text), LaTeX math expressions, equations, links, images, footnotes, citations, theorems, proofs, and examples. We believe this list of elements suffice for most technical and non-technical documents. It may not be impossible to support other types of elements in R Markdown, but you may start to lose the simplicity of Markdown if you wish to go that far.

Epictetus once said, “Wealth consists not in having great possessions, but in having few wants.” The spirit is also reflected in Markdown. If you can control your preoccupation with pursuing typesetting features, you should be much more efficient in writing the content and can become a prolific author. It is entirely possible to succeed with simplicity. Jung Jae-sung was a legendary badminton player with a remarkably simply playing style: he did not look like a talented player and was very short compared to other players, so most of the time you would just see him jump three feet off the ground and smash like thunder over and over again in the back court until he beats his opponents.

Please do not underestimate the customizability of R Markdown because of the simplicity of its syntax. In particular, Pandoc templates can be surprisingly powerful, as long as you understand the underlying technologies such as LaTeX and CSS, and are willing to invest time in the appearance of your output documents (reports, books, presentations, and/or websites). As one
example, you may check out the PDF report\(^2\) of the 2017 Employer Health Benefits Survey\(^3\). It looks fairly sophisticated, but was actually produced via bookdown (Xie, 2016), which is an R Markdown extension. A custom LaTeX template and a lot of LaTeX tricks were used to generate this report. Not surprisingly, this very book that you are reading right now was also written in R Markdown, and its full source is publicly available in the GitHub repository https://github.com/rstudio/rmarkdown-book.

R Markdown documents are often portable in the sense that they can be compiled to multiple types of output formats. Again, this is mainly due to the simplified syntax of the authoring language, Markdown. The simpler the elements in your document are, the more likely that the document can be converted to different formats. Similarly, if you heavily tailor R Markdown to a specific output format (e.g., LaTeX), you are likely to lose the portability, because not all features in one format work in another format.

Last but not least, your computing results will be more likely to be reproducible if you use R Markdown (or other knitr-based source documents), compared to the manual cut-and-paste approach. This is because the results are dynamically generated from computer source code. If anything goes wrong or needs to be updated, you can simply fix or update the source code, compile the document again, and the results will automatically updated. You can enjoy reproducibility and convenience at the same time.

---

**How to read this book**

This book may serve you better as a reference book than a textbook. It contains a large number of technical details, and we do not expect you to read it from beginning to end, since you may easily feel overwhelmed. Instead, think about your background and what you want to do first, and go to the relevant chapters or sections. For example:

- I just want to finish my course homework (Chapter 2 should be more than enough for you).

\(^3\)https://www.kff.org/health-costs/report/2017-employer-health-benefits-survey/
• I know this is an R Markdown book, but I use Python more than R (Go to Section 2.7.1).

• I want to embed interactive plots in my reports, or want my readers to be able change my model parameters interactively and see results on the fly (Check out Section 2.8).

• I know the output format I want to use, and I want to customize its appearance (Check out the documentation of the specific output format in Chapter 3 or Chapter 4). For example, I want to customize the template for my PowerPoint presentation (Go to Section 4.4.1).

• I want to build a business dashboard highlighting some key figures and indicators (Go to Chapter 5).

• I heard about yolo = TRUE from a friend, and I’m curious what that means in the xaringan package (Go to Chapter 7).

• I want to build a personal website (Go to Chapter 10), or write a book (Go to Chapter 12).

• I want to write a paper and submit to the Journal of Statistical Software (Go to Chapter 13).

• I want to build an interactive tutorial with exercises for my students to learn a topic (Go to Chapter 14).

• I’m familiar with R Markdown now, and I want to generate personalized reports for all my customers using the same R Markdown template (Try parameterized reports in Chapter 15).

• I know some JavaScript, and want to build an interface in R to call an interested JavaScript library from R (Learn how to develop HTML widgets in Chapter 16).

• I want to build future reports with a company branded template that shows our logo and uses our unique color theme (Go to Chapter 17).

If you are not familiar with R Markdown, we recommend that you read at least Chapter 2 to learn the basics. All the rest of the chapters in this book can be read in any order you desire. They are pretty much orthogonal to each other. However, to become familiar with R Markdown output formats, you may want to thumb through the HTML document format in Section 3.1, because many other formats share the same options as this format.
Structure of the book

This book consists of four parts. Part I covers the basics: Chapter 1 introduces how to install the relevant packages, and Chapter 2 is an overview of R Markdown, including the possible output formats, the Markdown syntax, the R code chunk syntax, and how to use other languages in R Markdown.

Part II is the detailed documentation of built-in output formats in the `rmarkdown` package, including document formats and presentation formats.

Part III lists about ten R Markdown extensions that enable you to build different applications or generate output documents with different styles. Chapter 5 introduces the basics of building flexible dashboards with the R package `flexdashboard`. Chapter 6 documents the `tufte` package, which provides a unique document style used by Edward Tufte. Chapter 7 introduces the `xaringan` package for another highly flexible and customizable HTML5 presentation format based on the JavaScript library remark.js. Chapter 8 documents the `revealjs` package, which provides yet another appealing HTML5 presentation format based on the JavaScript library reveal.js. Chapter 9 introduces a few output formats created by the R community, such as the `prettydoc` package, which features lightweight HTML document formats. Chapter 10 teaches you how to build websites using either the `blogdown` package or `rmarkdown`’s built-in site generator. Chapter 11 explains the basics of the `pkgdown` package, which can be used to quickly build documentation websites for R packages. Chapter 12 introduces how to write and publish books with the `bookdown` package. Chapter 13 is an overview of the `rticles` package for authoring journal articles. Chapter 14 introduces how to build interactive tutorials with exercises and/or quiz questions.

Part IV covers other topics about R Markdown, and some of them are advanced (in particular, Chapter 16). Chapter 15 introduces how to generate different reports with the same R Markdown source document and different parameters. Chapter 16 teaches developers how to build their own HTML widgets for interactive visualization and applications with JavaScript libraries. Chapter 17 shows how to create custom R Markdown and Pandoc templates so that you can fully customize the appearance and style of your output document. Chapter 18 explains how to create your own output formats if the existing formats do not meet your need. Chapter 19 shows how to combine the Shiny framework with R Markdown, so that your readers can interact
with the reports by changing the values of certain input widgets and seeing updated results immediately.

Note that this book is intended to be a guide instead of the comprehensive documentation of all topics related to R Markdown. Some chapters are only overviews, and you may need to consult the full documentation elsewhere (often freely available online). Such examples include Chapters 5, 10, 11, 12, and 14.

Software information and conventions

The R session information when compiling this book is shown below:

```r
xfun::session_info(c('blogdown', 'bookdown', 'knitr', 'rmarkdown', 'htmltools', 'reticulate', 'rticles', 'flexdashboard', 'learnr', 'shiny', 'revealjs', 'pkgdown', 'tinytex', 'xaringan', 'tufte'), dependencies = FALSE)
```

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS High Sierra 10.13.5
##
##
## Package version:
## blogdown_0.6.12 bookdown_0.7.11
## flexdashboard_0.5.1 htmltools_0.3.6
## knitr_1.20.5 learnr_0.9.2
## pkgdown_1.1.0 reticulate_1.8
## revealjs_0.9 rmarkdown_1.10.2
## rticles_0.4.2.9000 shiny_1.1.0
## tinytex_0.5.8 tufte_0.3
## xaringan_0.6.7
##
## Pandoc version: 2.2.1
```
We do not add prompts (> and +) to R source code in this book, and we comment out the text output with two hashes ## by default, as you can see from the R session information above. This is for your convenience when you want to copy and run the code (the text output will be ignored since it is commented out). Package names are in bold text (e.g., rmarkdown), and inline code and filenames are formatted in a typewriter font (e.g., knitr::knit('foo.Rmd')). Function names are followed by parentheses (e.g., blogdown::serve_site()). The double-colon operator :: means accessing an object from a package.

“Rmd” is the filename extension of R Markdown files, and also an abbreviation of R Markdown in this book.

Acknowledgments

I started writing this book after I came back from the 2018 RStudio Conference in early February, and finished the first draft in early May. This may sound fast for a 300-page book. The main reason I was able to finish it quickly was that I worked full-time on this book for three months. My employer, RStudio, has always respected my personal interests and allowed me to focus on projects that I choose by myself. More importantly, I have been taught several lessons on how to become a professional software engineer since I joined RStudio as a fresh PhD, although the initial journey turned out to be painful. It is a great blessing for me to work in this company.

The other reason for my speed was that JJ and Garrett had already prepared a lot of materials that I could adapt for this book. They had also been offering suggestions as I worked on the manuscript. In addition, Michael Harper contributed the initial drafts of Chapters 12, 13, 15, 17, and 18. I would definitely not be able to finish this book so quickly without their help.

The most challenging thing to do when writing a book is to find large blocks of uninterrupted time. This is just so hard. Both others and myself could interrupt me. I do not consider my willpower to be strong: I read random articles, click on the endless links on Wikipedia, look at random Twitter mes-

\footnote{https://yihui.name/en/2018/02/career-crisis/}

\footnote{http://mikeyharper.uk}
sages, watch people fight on meaningless topics online, reply to emails all the time as if I were able to reach “Inbox Zero”, and write random blog posts from time to time. The two most important people in terms of helping keep me on track are Tareef Kawaf (President of RStudio), to whom I report my progress on the weekly basis, and Xu Qin\(^6\), from whom I really learned\(^7\) the importance of making plans on a daily basis (although I still fail to do so sometimes). For interruptions from other people, it is impossible to isolate myself from the outside world, so I’d like to thank those who did not email me or ask me questions in the past few months and used public channels instead as I suggested\(^8\). I also thank those who did not get mad at me when my responses were extremely slow or even none. I appreciate all your understanding and patience. Besides, several users have started helping me answer GitHub and Stack Overflow questions related to R packages that I maintain, which is even better! These users include Marcel Schilling\(^9\), Xiying Tan\(^10\), Christophe Dervieux\(^11\), and Garrick Aden-Buie\(^12\), just to name a few. As someone who works from home, apparently I would not even have ten minutes of uninterrupted time if I do not send the little ones to daycare, so I want to thank all teachers at Small Miracles for freeing my daytime.

There have been a large number of contributors to the R Markdown ecosystem. More than 60 people\(^13\) have contributed to the core package, rmarkdown. Several authors have created their own R Markdown extensions, as introduced in Part III of this book. Contributing ideas is no less helpful than contributing code. We have gotten numerous inspirations and ideas from the R community via various channels (GitHub issues, Stack Overflow questions, and private conversations, etc.). As a small example, Jared Lander, author of the book R for Everyone, does not meet me often, but every time he chats with me, I will get something valuable to work on. “How about writing books with R Markdown?” he asked me at the 2014 Strata conference in New York. Then we invented bookdown in 2016. “I really need fullscreen background images in ioslides. Look, Yihui, here are my ugly JavaScript hacks,\(^14\)” he showed me

\(^{6}\)http://home.uchicago.edu/~xuqin/
\(^{7}\)https://d.cosx.org/d/419325
\(^{8}\)https://yihui.name/en/2017/08/so-gh-email/
\(^{9}\)https://yihui.name/en/2018/01/thanks-marcel-schilling/
\(^{10}\)https://shrektan.com
\(^{11}\)https://github.com/cderv
\(^{12}\)https://www.garrickadenbuie.com
\(^{13}\)https://github.com/rstudio/rmarkdown/graphs/contributors
on the shuttle to dinner at the 2017 RStudio Conference. A year later, background images were officially supported in ioslides presentations.

As I mentioned previously, R Markdown is standing on the shoulders of the giant, Pandoc. I’m always amazed by how fast John MacFarlane, the main author of Pandoc, responds to my GitHub issues. It is hard to imagine a person dealing with 5000 GitHub issues over the years while maintaining the excellent open-source package and driving the Markdown standards forward. We should all be grateful to John and contributors of Pandoc.

As I was working on the draft of this book, I received a lot of helpful reviews from these reviewers: John Gillett (University of Wisconsin), Rose Hartman (UnderstandingData), Amelia McNamara (Smith College), Ariel Muldoon (Oregon State University), Yixuan Qiu (Purdue University), Benjamin Soltoff (University of Chicago), David Whitney (University of Washington), and Jon Katz (independent data analyst). Tareef Kawaf (RStudio) also volunteered to read the manuscript and provided many helpful comments. Aaron Simumba, Peter Baumgartner, and Daijiang Li volunteered to carefully correct many of my typos. In particular, Aaron has been such a big helper with my writing (not limited to only this book) and sometimes I have to compete with him in correcting my typos!

There are many colleagues at RStudio whom I want to thank for making it so convenient and even enjoyable to author R Markdown documents, especially the RStudio IDE team including J.J. Allaire, Kevin Ushey, Jonathan McPherson, and many others.

Personally I often feel motivated by members of the R community. My own willpower is weak, but I can gain a lot of power from this amazing community. Overall the community is very encouraging, and sometimes even fun, which makes me enjoy my job. For example, I do not think you can often use the picture of a professor for fun in your software, but the “desiccated baseR-er” Karl Broman is an exception (see Section 7.3.6), as he allowed me to use a mysteriously happy picture of him.

Lastly, I want to thank my editor, John Kimmel, for his continued help with my fourth book. I think I have said enough about him and his team at Chap-
man & Hall in my previous books. The publishing experience has always been so smooth. I just wonder if it would be possible someday that our meticulous copy-editor, Suzanne Lassandro, would fail to identify more than 30 issues for me to correct in my first draft. Probably not. Let’s see.

Yihui Xie
Elkhorn, Nebraska
About the Authors

This book is primarily put together by me (Yihui Xie), making use of the existing R documentation of the `markdown` package and the `rmarkdown` website, which were mainly contributed by J.J. Allaire and Garrett Grolemund.

Yihui Xie

Yihui Xie (https://yihui.name) is a software engineer at RStudio (https://www.rstudio.com). He earned his PhD from the Department of Statistics, Iowa State University. He is interested in interactive statistical graphics and statistical computing. As an active R user, he has authored several R packages, such as `knitr`, `bookdown`, `blogdown`, `xaringan`, `tinytex`, `animation`, `DT`, `tufte`, `formatR`, `fun`, `xfun`, `mime`, `highr`, `servr`, and `Rd2roxygen`, among which the `animation` package won the 2009 John M. Chambers Statistical Software Award (ASA). He also co-authored a few other R packages, including `shiny`, `rmarkdown`, and `leaflet`.

He has authored two books, *Dynamic Documents with knitr* (Xie, 2015), and *bookdown: Authoring Books and Technical Documents with R Markdown* (Xie, 2016), and co-authored the book, *blogdown: Creating Websites with R Markdown* (Xie et al., 2017).

In 2006, he founded the Capital of Statistics (https://cosx.org), which has grown into a large online community on statistics in China. He initiated the Chinese R conference in 2008, and has been involved in organizing R conferences in China since then. During his PhD training at Iowa State University, he won the Vince Sposito Statistical Computing Award (2011) and the Snedecor Award (2012) in the Department of Statistics.

He occasionally rants on Twitter (https://twitter.com/xieyihui), and most of the time you can find him on GitHub (https://github.com/yihui).
He enjoys spicy food as much as classical Chinese literature.

J.J. Allaire

J.J. Allaire is the founder of RStudio and the creator of the RStudio IDE. J.J. is an author of several packages in the R Markdown ecosystem including rmarkdown, flexdashboard, learnr, and radix.

Garrett Grolemund

Garrett Grolemund is the co-author of R for Data Science and author of Hands-On Programming with R. He wrote the lubridate R package and works for RStudio as an advocate who trains engineers to do data science with R and the Tidyverse. If you use R yourself, you may recognize Garrett from his video courses on Datacamp.com and O’Reilly media, or for his series of popular R cheatsheets distributed by RStudio.

Garrett earned his PhD in Statistics from Rice University in 2012 under the guidance of Hadley Wickham. Before that, he earned a Bachelor’s degree in Psychology from Harvard University and briefly attended law school. Garrett has been one of the foremost promoters of Shiny, R Markdown, and the Tidyverse, documenting and explaining each in detail.
Part I

Get Started
1

**Installation**

We assume you have already installed R ([https://www.r-project.org](https://www.r-project.org)) (R Core Team, 2018) and the RStudio IDE ([https://www.rstudio.com](https://www.rstudio.com)). RStudio is not required but recommended, because it makes it easier for an average user to work with R Markdown. If you do not have RStudio IDE installed, you will have to install Pandoc ([http://pandoc.org](http://pandoc.org)), otherwise there is no need to install Pandoc separately because RStudio has bundled it. Next you can install the `rmarkdown` package in R:

```r
# Install from CRAN
install.packages('rmarkdown')

# Or if you want to test the development version,
# install from GitHub
if (!requireNamespace("devtools"))
  install.packages('devtools')
devtools::install_github('rstudio/rmarkdown')
```

If you want to generate PDF output, you will need to install LaTeX. For R Markdown users who have not installed LaTeX before, we recommend that you install TinyTeX ([https://yihui.name/tinytex/](https://yihui.name/tinytex/)):

```r
install.packages("tinytex")
tinytex::install_tinytex()  # install TinyTeX
```

TinyTeX is a lightweight, portable, cross-platform, and easy-to-maintain LaTeX distribution. The R companion package `tinytex` (Xie, 2018f) can help you automatically install missing LaTeX packages when compiling LaTeX or R Markdown documents to PDF, and also ensures a LaTeX document is compiled for the correct number of times to resolve all cross-references. If you do not understand what these two things mean, you should probably follow
our recommendation to install TinyTeX, because these details are often not worth your time or attention.

With the `rmarkdown` package, RStudio/Pandoc, and LaTeX, you should be able to compile most R Markdown documents. In some cases, you may need other software packages, and we will mention them when necessary.


