

The R Series

Reproducible Research with R and RStudio

Third Edition



Christopher Gandrud



CRC Press
Taylor & Francis Group

A CHAPMAN & HALL BOOK

Reproducible Research with R and RStudio

Third Edition

Chapman & Hall/CRC The R Series

Series Editors

John M. Chambers, Department of Statistics, Stanford University, California, USA

Torsten Hothorn, Division of Biostatistics, University of Zurich, Switzerland

Duncan Temple Lang, Department of Statistics, University of California, Davis, USA

Hadley Wickham, RStudio, Boston, Massachusetts, USA

Recently Published Titles

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

Displaying Time Series, Spatial, and Space-Time Data with R, Second Edition

Oscar Perpinan Lamigueiro

Reproducible Finance with R

Jonathan K. Regenstein, Jr

R Markdown

The Definitive Guide

Yihui Xie, J.J. Allaire, Garrett Grolmund

Practical R for Mass Communication and Journalism

Sharon Machlis

Analyzing Baseball Data with R, Second Edition

Max Marchi, Jim Albert, Benjamin S. Baumer

Spatio-Temporal Statistics with R

Christopher K. Wikle, Andrew Zammit-Mangion, and Noel Cressie

Statistical Computing with R, Second Edition

Maria L. Rizzo

Geocomputation with R

Robin Lovelace, Jakub Nowosad, Jannes Muenchow

Advanced R, Second Edition

Hadley Wickham

Dose Response Analysis Using R

Christian Ritz, Signe Marie Jensen, Daniel Gerhard, Jens Carl Streibig

Distributions for Modelling Location, Scale, and Shape

Using GAMLSS in R

Robert A. Rigby, Mikis D. Stasinopoulos, Gillian Z. Heller and Fernanda De Bastiani

Hands-On Machine Learning with R

Bradley Boehmke and Brandon Greenwell

Statistical Inference via Data Science

A ModernDive into R and the Tidyverse

Chester Ismay and Albert Y. Kim

Reproducible Research with R and RStudio, Third Edition

Christopher Gandrud

For more information about this series, please visit: <https://www.crcpress.com/go/the-r-series>

Reproducible Research with R and RStudio

Third Edition

Christopher Gandrud



CRC Press

Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

© 2020 by Taylor & Francis Group, LLC
CRC Press is an imprint of Taylor & Francis Group, an Informa business

No claim to original U.S. Government works

Printed on acid-free paper

International Standard Book Number-13: 978-0-367-14398-5 (Paperback)
International Standard Book Number-13: 978-0-367-14402-9 (Hardback)

This book contains information obtained from authentic and highly regarded sources. Reasonable efforts have been made to publish reliable data and information, but the author and publisher cannot assume responsibility for the validity of all materials or the consequences of their use. The authors and publishers have attempted to trace the copyright holders of all material reproduced in this publication and apologize to copyright holders if permission to publish in this form has not been obtained. If any copyright material has not been acknowledged please write and let us know so we may rectify in any future reprint.

Except as permitted under U.S. Copyright Law, no part of this book may be reprinted, reproduced, transmitted, or utilized in any form by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying, microfilming, and recording, or in any information storage or retrieval system, without written permission from the publishers.

For permission to photocopy or use material electronically from this work, please access www.copyright.com (<http://www.copyright.com/>) or contact the Copyright Clearance Center, Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923, 978-750-8400. CCC is a not-for-profit organization that provides licenses and registration for a variety of users. For organizations that have been granted a photocopy license by the CCC, a separate system of payment has been arranged.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks and are used only for identification and explanation without intent to infringe.

Library of Congress Cataloging-in-Publication Data

Names: Gandrud, Christopher, author.
Title: Reproducible research with R and RStudio / by Christopher Gandrud.
Description: Third edition. | Boca Raton, FL : CRC Press, [2020] | Series: The R series | Includes bibliographical references and index. | Summary: "Brings together the skills and tools needed for doing and presenting computational research. Using straightforward examples, the book takes you through an entire reproducible research workflow"-- Provided by publisher.
Identifiers: LCCN 2019046298 (print) | LCCN 2019046299 (ebook) | ISBN 9780367143985 (paperback) | ISBN 9780367144029 (hardback) | ISBN 9780429031854 (ebook)
Subjects: LCSH: Research--Statistical methods. | R (Computer program language)
Classification: LCC Q180.55.S7 G36 2020 (print) | LCC Q180.55.S7 (ebook) | DDC 001.4/2202855133--dc23
LC record available at <https://lcn.loc.gov/2019046298>
LC ebook record available at <https://lcn.loc.gov/2019046299>

Visit the Taylor & Francis Web site at
<http://www.taylorandfrancis.com>

and the CRC Press Web site at
<http://www.crcpress.com>

Contents

Preface	xi
About the Author	xv
Stylistic Conventions	xvii
Additional Resources	xix
I Getting Started	1
1 Introducing Reproducible Research	3
1.1 What Is Reproducible Research?	4
1.2 Why Should Research Be Reproducible?	5
1.2.1 For science	5
1.2.2 For you	6
1.3 Who Should Read This Book?	8
1.3.1 Academic researchers	9
1.3.2 Students	9
1.3.3 Instructors	9
1.3.4 Editors	10
1.3.5 Private sector researchers	10
1.4 The Tools of Reproducible Research	11
1.4.1 Why Use R, knitr/R Markdown, and RStudio for Reproducible Research?	12
1.5 Installing the main software	15
1.5.1 Installing markup languages	15
1.5.2 GNU Make	16
1.5.3 Other tools	16
1.6 Book Overview	17
1.6.1 How to read this book	18
1.6.2 Reproduce this book	19
1.6.3 Contents overview	19
2 Getting Started with Reproducible Research	23
2.1 The Big Picture: A Workflow for Reproducible Research	23
2.1.1 Reproducible theory	24
2.2 Practical Tips for Reproducible Research	25

2.2.1	Document everything!	26
2.2.2	Everything is a (text) file	27
2.2.3	All files should be human readable	28
2.2.4	Explicitly tie your files together	30
2.2.5	Have a plan to organize, store, and make your files available	32
3	Getting Started with R, RStudio, and knitr/R Markdown	33
3.1	Using R: The Basics	33
3.1.1	Objects	34
3.1.2	Functions	42
3.1.3	The workspace and history	45
3.1.4	R history	46
3.1.5	Global R options	46
3.1.6	Installing new packages and loading functions	47
3.2	Using RStudio	47
3.3	Using knitr and R Markdown: The Basics	50
3.3.1	What <i>knitr</i> does	50
3.3.2	What <i>rmarkdown</i> does	50
3.3.3	File extensions	53
3.3.4	Code chunks	53
3.3.5	Global chunk options	55
3.3.6	<i>knitr</i> package options	57
3.3.7	Hooks	57
3.3.8	knitr, R Markdown, and RStudio	57
3.3.9	knitr and R	61
3.3.10	R Markdown and R	63
	Appendix: Jupyter Interactive Notebooks	65
	Appendix: knitr and Lyx	67
4	Getting Started with File Management	69
4.1	File Paths and Naming Conventions	70
4.1.1	Root directories	70
4.1.2	Sub-directories and parent directories	70
4.1.3	Working directories	71
4.1.4	Absolute vs. relative paths	71
4.1.5	Spaces in directory and file names	73
4.2	Organizing Your Research Project	73
4.3	Organizing Research with RStudio Projects	74
4.4	R File Manipulation Functions	75
4.5	Unix-like Shell Commands for File Management	79
4.6	File Navigation in RStudio	83

II Data Gathering and Storage 85

5	Storing, Collaborating, Accessing Files, and Versioning	87
5.1	Saving Data in Reproducible Formats	88
5.2	Storing Your Files in the Cloud: Dropbox	89
5.2.1	Storage	90
5.2.2	Accessing data	91
5.2.3	Collaboration	92
5.2.4	Version control	92
5.3	Storing Your Files in the Cloud: GitHub	93
5.3.1	Setting up GitHub: Basic	95
5.3.2	Version control with Git	96
5.3.3	Remote storage on GitHub	104
5.3.4	Accessing on GitHub	106
5.3.5	Summing up the GitHub workflow	109
5.4	RStudio and GitHub	110
5.4.1	Setting up Git/GitHub with Projects	110
5.4.2	Using Git in RStudio Projects	111
6	Gathering Data with R	113
6.1	Organize Your Data Gathering: Makefiles	113
6.1.1	R Make-like files	114
6.1.2	GNU Make	115
6.2	Importing Locally Stored Data Sets	121
6.3	Importing Data Sets from the Internet	122
6.3.1	Data from non-secure (<i>http</i>) URLs	122
6.3.2	Data from secure (<i>https</i>) URLs	123
6.3.3	Compressed data stored online	123
6.3.4	Data APIs and feeds	124
6.4	Advanced Automatic Data Gathering: Web Scraping	126
7	Preparing Data for Analysis	129
7.1	Cleaning Data for Merging	129
7.1.1	Get a handle on your data	129
7.1.2	Reshaping data	132
7.1.3	Renaming variables	135
7.1.4	Ordering data	136
7.1.5	Subsetting data	137
7.1.6	Recoding string/numeric variables	139
7.1.7	Creating new variables from old	140
7.1.8	Changing variable types	143
7.2	Merging Data Sets	143
7.2.1	Binding	143
7.2.2	Merging data frames	144
7.2.3	Duplicate columns	147

Appendix	149
III Analysis and Results	151
8 Statistical Modeling and knitr/R Markdown	153
8.1 Incorporating Analyses into the Markup	154
8.1.1 Full code chunks	154
8.1.2 Showing code and results inline	157
8.1.3 Dynamically including non-R code in code chunks	159
8.2 Dynamically Including Modular Analysis Files	159
8.2.1 Source from a local file	160
8.2.2 Source from a URL	162
8.3 Reproducibly Random: <code>set.seed()</code>	163
8.4 Computationally Intensive Analyses	164
9 Showing Results with Tables	167
9.1 Basic <i>knitr</i> Syntax for Tables	168
9.2 Table Basics	168
9.2.1 Tables in LaTeX	169
9.2.2 Tables in Markdown/HTML	173
9.3 Creating Tables from Supported Class R Objects	177
9.3.1 <i>kable</i> for Markdown and LaTeX	177
9.3.2 <i>xtable</i> for LaTeX and HTML	178
9.3.3 <i>texreg</i> for LaTeX and HTML	181
9.3.4 Fitting large tables in LaTeX	184
9.3.5 <i>xtable</i> with non-supported class objects	185
9.3.6 Creating variable description documents with <i>xtable</i>	187
10 Showing Results with Figures	191
10.1 Including Non-knitted Graphics	192
10.1.1 Including graphics in LaTeX	192
10.1.2 Including graphics in Markdown/HTML	194
10.1.3 Non-knitted graphics with <i>knitr/rmarkdown</i>	195
10.2 Basic <i>knitr/rmarkdown</i> Figure Options	196
10.2.1 Chunk options	196
10.2.2 Global options	197
10.3 Knitting R's Default Graphics	198
10.4 Including <i>ggplot2</i> Graphics	202
10.4.1 Showing regression results with caterpillar plots	205
10.5 JavaScript Graphs with <i>googleVis</i>	208
10.5.1 Basic <i>googleVis</i> figures	209
10.5.2 Including <i>googleVis</i> in knitted documents	210
10.5.3 JavaScript Graphs with <i>htmlwidgets</i> -based packages	211

IV Presentation Documents 213

11 Presenting with LaTeX 215

11.1 The Basics	216
11.1.1 Getting started with LaTeX editors	216
11.1.2 Basic LaTeX command syntax	217
11.1.3 The LaTeX preamble and body	217
11.1.4 Headings	222
11.1.5 Paragraphs and spacing	222
11.1.6 Horizontal lines	222
11.1.7 Text formatting	223
11.1.8 Math	224
11.1.9 Lists	225
11.1.10 Footnotes	226
11.1.11 Cross-references	226
11.2 Bibliographies with BibTeX	226
11.2.1 The <i>.bib</i> file	227
11.2.2 Including citations in LaTeX documents	228
11.2.3 Generating a BibTeX file of R package citations	228
11.3 Presentations with LaTeX Beamer	231
11.3.1 Beamer basics	231
11.3.2 <i>knitr</i> with LaTeX slideshows	234

12 Presenting in a Variety of Formats with R Markdown 237

12.1 The Basics	237
12.1.1 Getting started with Markdown editors	238
12.1.2 Preamble and document structure	239
12.1.3 Headings	240
12.1.4 Horizontal lines	240
12.1.5 Paragraphs and new lines	240
12.1.6 Italics and bold	241
12.1.7 Links	241
12.1.8 Lists	241
12.1.9 Math with MathJax	242
12.2 Further Customizability with <i>rmarkdown</i>	243
12.2.1 CSS style files and Markdown	247
12.3 Slideshows with Markdown, R Markdown, and HTML	248
12.3.1 HTML slideshows with <i>rmarkdown</i>	249
12.3.2 LaTeX Beamer slideshows with <i>rmarkdown</i>	250
12.3.3 Slideshows with Markdown and RStudio's R Presentations	251
12.4 Publishing HTML Documents Created with R Markdown	254
12.4.1 Further information on R Markdown	256

13 Conclusion	257
13.1 Citing Reproducible Research	258
13.2 Licensing Your Reproducible Research	259
13.3 Sharing Your Code in Packages	259
13.4 Project Development: Public or Private?	260
13.5 Is it Possible to Completely Future-Proof Your Research?	261
Bibliography	263
Index	271

Preface

Motivation

This book has its genesis in my PhD research at the London School of Economics. I started the degree with questions about the 2008/09 financial crisis and planned to spend most of my time researching capital adequacy requirements. But I quickly realized that I would actually spend a large proportion of my time learning the day-to-day tasks of data gathering, analysis, and results presentation. After plodding through for a while with Word, Excel, and Stata, my breaking point came while reentering results into a regression table after I had tweaked one of my statistical models, yet again. Surely there was a better way to *do* research that would allow me to spend more time answering my research questions. Making research reproducible for others also means making it better organized and efficient for yourself. My search for a better way led me straight to the tools for reproducible computational research.

The reproducible research community is very active, knowledgeable, and helpful. Nonetheless, I often encountered holes in this collective knowledge, or at least had no resource organizing it all together as a whole. That is my intention for this book: to bring together the skills I have picked up for actually doing and presenting computational research. Hopefully, the book, along with making reproducible research more widely used, will save researchers hours of googling, so they can spend more time addressing their research questions.

Changes to the Third Edition

- Spring cleaning: updated package recommendations, examples, and URLs. Removed technologies no longer in regular use.
- More advanced R Markdown and less LaTeX in discussions of markup languages and examples.
- Stronger focus on reproducible working directory tools.

- Updated discussion of cloud storage services and persistently citing reproducible material.
- Added discussion of Jupyter notebooks and reproducible practices in industry.
- Examples of data manipulation with Tidyverse tibbles (in addition to standard data frames) and `pivot_longer()` and `pivot_wider()` functions for pivoting data.
- Naming conventions are in current R-Tidyverse best practice.

A detailed list of changes for the third edition is available at <https://github.com/christophergandrud/Rep-Res-Book/issues/57#issuecomment-421739971>.

Changes to the Second Edition

The tools of reproducible research have developed rapidly since the first edition of this book was published just two years ago. The second edition has been updated to incorporate the most important of these advancements, including discussions of:

- The *rmarkdown* package, which allows you to create reproducible research documents in PDF, HTML, and Microsoft Word formats using the simple and intuitive Markdown syntax.
- Improvements and changes to RStudio's interface and capabilities, such as its new tools for handling R Markdown documents.
- Expanded *knitr* R code chunk capabilities.
- The `kable()` function in the *knitr* package and the *texreg* package for dynamically creating tables to present your data and statistical results.
- An improved discussion of file organization allowing you to take full advantage of relative file paths so that your documents are more easily reproducible across computers and systems.
- The *dplyr*, *magrittr*, and *tidyr* packages for fast data manipulation.
- Numerous changes to R syntax in user-created packages.
- Changes to GitHub's and Dropbox's interfaces.

Acknowledgments

I would not have been able to write this book without many people's advice and support. Foremost is John Kimmel, acquisitions editor at Chapman & Hall. He approached me in Spring 2012 with the general idea and opportunity for this book. Other editors at Chapman & Hall and Taylor & Francis have greatly contributed to this project, including Marcus Fontaine. I would also like to thank all of the book's reviewers whose helpful comments have greatly improved it. The first edition's reviewers include:

- Jeromy Anglim, Deakin University
- Karl Broman, University of Wisconsin, Madison
- Jake Bowers, University of Illinois, Urbana-Champaign
- Corey Chivers, McGill University
- Mark M. Fredrickson, University of Illinois, Urbana-Champaign
- Benjamin Lauderdale, London School of Economics
- Ramnath Vaidyanathan, McGill University

Many other anonymous reviewers also gave great feedback over the years.

The developer and blogging community has also been incredibly important for making this book possible. Foremost among these people is Yihui Xie. He is the main developer behind the *knitr* package, co-developer of *rmarkdown*, and also an avid blog writer and commenter. Without him, the ability to do reproducible research would be much harder and the blogging community that spreads knowledge about how to do these things would be poorer. Other great contributors to the reproducible research community include Carl Boettiger, Karl Broman, Markus Gesmann (who developed *googleVis*), Rob Hyndman, and Hadley Wickham (who has developed numerous very useful R packages). Thank you also to Victoria Stodden and Michael Malecki for helpful suggestions. And, of course, thank you to everyone at RStudio (especially JJ Allaire) for creating an increasingly useful program for reproducible research.

The second edition has benefited immensely from first edition readers' comments and suggestions. For a list of their valuable contributions, please see the book's GitHub Issues page <https://github.com/christophergandrud/Rep-Res-Book/issues> and the first edition's Errata page <http://christophergandrud.github.io/RepResR-RStudio/errata.htm>.

My students at Yonsei University were an important part of making the first edition. One of the reasons that I got interested in using many of the tools covered in this book, like using *knitr* in slideshows, was to improve a course I taught there: Introduction to Social Science Data Analysis. I tested many of the explanations and examples in this book on my students. Their feedback has been very helpful for making the book clearer and more useful. Their

experience with using these tools on Microsoft Windows computers was also important for improving the book's Windows documentation. Similarly, my students at the Hertie School of Governance inspired and tested key sections of the second edition.

The vibrant community at Stack Overflow <http://stackoverflow.com/> and Stack Exchange <http://stackexchange.com/> are always very helpful for finding answers to problems that plague any computational researcher. Importantly, the sites make it easy for others to find the answers to questions that have already been asked.

The library at the University of California, San Francisco was a great home for writing the third edition.

Kristina Gandrud has been immensely supportive and patient with me throughout the writing of this book (and my entire career).

About the Author

Christopher Gandrud is Head of Economics and Experimentation at Zaldando SE. He leads teams of social data scientists and software engineers building and evaluating large-scale automated decision-making systems. He was previously a research fellow at the Institute for Quantitative Social Science, Harvard University developing statistical software for the social and physical sciences. He has held posts at City, University of London, the Hertie School of Governance, Yonsei University, and the London School of Economics where in 2012 he completed a PhD in quantitative political science.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Stylistic Conventions

I use the following conventions throughout the book:

- **Abstract variables:** Abstract variables, i.e. variables that do not reference specific objects, are in ALL CAPS TYPEWRITER TEXT.
- **Clickable buttons:** Clickable buttons are in typewriter text.
- **Code:** All code is in typewriter text.
- **File names and directories:** File names and directories more generally are printed in *italics*. Words are separated by em dashes—*kebab-case*.¹
- **File extensions:** Like file names, file extensions are *italicized*.
- **Individual variable values:** Individual variable values mentioned in the text are in *italics*.
- **Objects:** Objects are printed in *italics*. I use underscores (`_`) to separate words in object names.
- **Object columns:** Data frame object columns are printed in **bold**.
- **R Function names:** are followed by parentheses (e.g., `stats::lm()`)
- **Packages:** R packages are printed in *italics*. When a system, rather than the package that shares its name is referred to, it is not italicized, e.g. R Markdown (system) vs. *rmarkdown* (package).²
- **Windows and RStudio panes:** Open windows and RStudio panes are written in *italics*.
- **Variable names:** Variable names are printed in **bold**. Underscores (`_`) separate words in variable names.

¹See <https://stackoverflow.com/a/17820138>. Posted 23 July 2013.

²See Yihui Xie's comment at: <https://andrewgelman.com/2016/01/14/rstanarm-and-more/#comment-259425>. Posted 14 January 2016.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Additional Resources

You can freely download additional resources supplementing examples in this book. These resources include longer examples discussed in individual chapters and a complete short reproducible research project.

Chapter Examples

Longer examples discussed in individual chapters, including files to dynamically download data, code for creating figures, and markup files for creating presentation documents, can be accessed at: <https://github.com/christophergandrud/rep-res-book-v3-examples>. Please see [Chapter 5](#) for more information on downloading files from GitHub, where the examples are stored.

Short Example Project

To download a full (though very short) example of a reproducible research project created using the tools covered in this book, go to: <https://github.com/christophergandrud/rep-res-book-v3-examples>. Please follow the replication instructions in the main *README.md*. It is a good idea to hold off looking at this complete example in detail until after you have become acquainted with the individual tools it uses. Become acquainted with the tools by reading through this book and working with the chapter examples.

The following two figures give you a sense of how the example's files are organized. [Figure 1](#) shows how the files are organized in the file system. [Figure 2](#) illustrates how the main files are dynamically tied together. In the *data* directory, we have files to gather raw data from the World Bank (2018) on fertilizer consumption and from Pemstein et al. (2010) on countries' levels of democracy. They are tied to the data through the `WDI()` and `download.file()`

functions. A *Makefile* can run *gather-1* and *gather-2.R* to gather and clean the data. It runs *merge-data.R* to merge the data into one data file called *main-data.csv*. It also automatically generates a variable description file and a *README.md* recording the session info.

The *analysis* folder contains two files that create figures presenting this data. They are tied to *main-data.csv* with the `import()` function. These files are run by the presentation documents when they are knitted. The presentation documents tie to the analysis documents with *knitr* and the `source()` function.

Though a simple example, hopefully these files will give you a complete sense of how a reproducible research project can be organized. Please feel free to experiment with different ways of organizing the files and tying them together to make your research really reproducible.

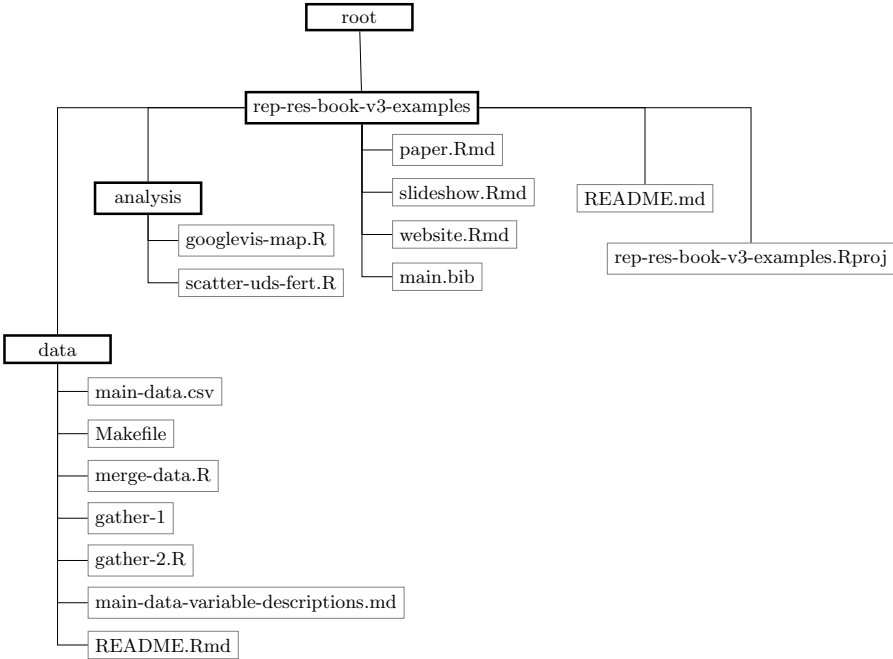


FIGURE 1: Short Example Project File Tree

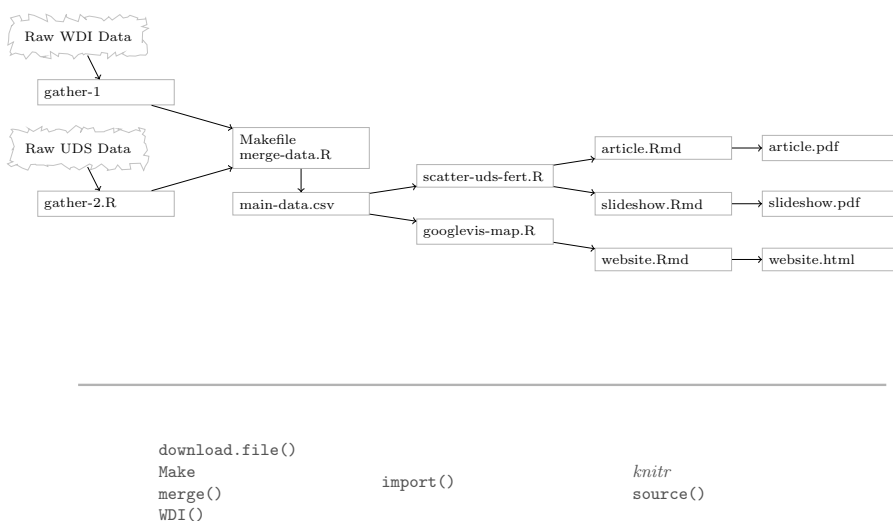


FIGURE 2: Short Example Main File Ties

Updates

Many of the reproducible research tools discussed in this book are improving rapidly. Because of this, I will regularly post updates to the content covered in the book at: <https://github.com/christophergandrud/Rep-Res-Book>.

Corrections

If you notice any corrections that should be made to fix typos, broken URLs, and so on, you can report them at: <https://github.com/christophergandrud/Rep-Res-Book/issues>. I'll post notifications of changes to an Errata page at: <http://christophergandrud.github.io/RepResR-RStudio/errata.htm>.



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Part I

Getting Started



Taylor & Francis

Taylor & Francis Group

<http://taylorandfrancis.com>

Introducing Reproducible Research

Research is typically presented in very selective containers: slideshows, journal articles, books, or websites. These presentation documents announce a project's findings and try to convince us that the results are correct (Mesirov, 2010). It's important to remember that these documents are not the research. Especially in the computational and statistical sciences, these documents are the “advertising”. The research is the “full software environment, code, and data that produced the results” (Buckheit and Donoho, 1995; Donoho, 2010, 385). When we separate the research from its advertisement, we are making it difficult for others to verify the findings by reproducing them.

This book gives you the tools to dynamically combine your research with the presentation of your findings. The first tool is a workflow for reproducible research that weaves the principles of reproducibility throughout your entire research project, from data gathering to the statistical analysis, and the presentation of results. You will also learn how to use a number of computer tools that make this workflow easier and more robust. These tools include:

- the **R** statistical language that will allow you to gather data and analyze it;
- the **LaTeX** and **Markdown** markup languages that you can use to create documents—slideshows, articles, books, and webpages—for presenting your findings;
- the *knitr* and *rmarkdown* **packages** for R and other tools, including **command-line programs** like GNU Make and Git version control, for dynamically tying your data gathering, analysis, and presentation documents together so that they can be easily reproduced;
- **RStudio**, a program that brings all of these tools together.

1.1 What Is Reproducible Research?

Though there is some debate over the necessary and sufficient conditions for a full replication (Makel and Plucker, 2014, 2), research results are generally considered¹ *replicable* if there is sufficient information available for independent researchers to make the same findings using the same procedures with new data.² For research that relies on experiments, this can mean a researcher not involved in the original research being able to rerun the experiment, including sampling, and validate that the new results are comparable to the original results. In computational and quantitative empirical sciences, results are replicable if independent researchers can recreate findings by following the procedures originally used to gather the data and run the computer code. Of course, it is sometimes difficult to replicate the original data set because of issues such as limited resources to gather new data or because the original study already sampled the full universe of cases. So as a next-best standard, we can aim for “*really reproducible research*” (Peng, 2011, 1226).³ In computational sciences⁴ this means:

the data and code used to make a finding are available and they are sufficient for an independent researcher to recreate the finding.

In practice, research needs to be *easy* for independent researchers to reproduce (Ball and Medeiros, 2011). If a study is difficult to reproduce, it’s more likely that no one will reproduce it. If someone does attempt to reproduce this research, it will be difficult for them to tell if any errors they find were in the

¹Rokem et al. (2018, 3-4) note that some disciplines, e.g. computing machinery and meteorology, give “replicable” and “reproducible” the exact opposite meanings from the way they are used in this book and many other disciplines such as biology, economics, and epidemiology.

²This is close to what Lykken (1968) calls “operational replication”.

³The really reproducible computational research originates in the 1980s and early 1990s with Jon Claerbout and the Stanford Exploration Project (Fomel and Claerbout, 2009; Donoho et al., 2009). Further seminal advances were made by Jonathan B. Buckheit and David L. Donoho who created the Wavelab library of MATLAB routines for their research on wavelets in the mid-1990s (Buckheit and Donoho, 1995).

⁴Reproducibility is important for both quantitative and qualitative research (King et al., 1994). Nonetheless, we will focus mainly on methods for reproducibility in quantitative computational research.

original research or problems they introduced during the reproduction. In this book, you will learn how to avoid these problems.

In particular, you will learn tools for dynamically “*knitting*”⁵ the data and the source code together with your presentation documents. Combined with well-organized source files and clearly and completely commented code, independent researchers will be able to understand how you obtained your results. This will make your computational research easily reproducible.

1.2 Why Should Research Be Reproducible?

Reproducible research is one of the main components of science. If that’s not enough reason for you to make your research reproducible, consider that the tools of reproducible research also have direct benefits for you as a researcher.

1.2.1 For science

Replicability has been a key part of scientific inquiry from perhaps the 1200s (Bacon, 1859; Nosek et al., 2012). It has even been called the “demarcation between science and non-science” (Braude, 1979, 2). Why is replication so important for scientific inquiry?

Standard to judge scientific claims

Replication opens claims to scrutiny, allowing us to keep what works and discard what doesn’t. Science, according to the American Physical Society, “is the systematic enterprise of gathering knowledge . . . organizing and condensing that knowledge into testable laws and theories”. The “ultimate standard” for evaluating scientific claims is whether or not the claims can be replicated (Peng, 2011; Kelly, 2006). Research findings cannot even really be considered “genuine contributions to human knowledge” until they have been verified through replication (Stodden, 2009b, 38). Replication “requires the complete and open exchange of data, procedures, and materials”. Scientific conclusions

⁵Much of the reproducible computational research and literate programming literatures have traditionally used the term “weave” to describe the process of combining source code and presentation documents (see Knuth, 1992, 101). In the R community, the term “weave” is usually used to describe the combination of source code and LaTeX documents. The term “knit” reflects the vocabulary of the *knitr* R package (knit + R). It is used more generally to describe weaving with a variety of markup languages. The term is used by RStudio if you are using the *rmarkdown* package, which is similar to *knitr*. We also cover the *rmarkdown* package in this book. Because of this, I use the term knit rather than weave in this book.

that are not replicable should be abandoned or modified “when confronted with more complete or reliable . . . evidence”.⁶

Reproducibility enhances replicability. If other researchers are able to clearly understand how a finding was originally made, then they will be better able to conduct comparable research in meaningful attempts to replicate the original findings. Sometimes strict replicability is not feasible, for example, when it is only possible to gather one data set on a population of interest. In these cases reproducibility is a “minimum standard” for judging scientific claims (Peng, 2011).

It is important to note that though reproducibility is a minimum standard for judging scientific claims, “a study can be reproducible and still be wrong” (Peng, 2014). For example, a statistically significant finding in one study may remain statistically significant when reproduced using the original data/code, but when researchers try to replicate it using new data and even methods, they are unable to find a similar result. The original finding could have been noise, even though it is fully reproducible.

Avoiding effort duplication and encouraging cumulative knowledge development

Not only is reproducibility important for evaluating scientific claims, it can also contribute to the cumulative growth of scientific knowledge (Kelly, 2006; King, 1995). Reproducible research cuts down on the amount of time scientists have to spend gathering data or developing procedures that have already been collected or figured out. Because researchers do not have to discover on their own things that have already been done, they can more quickly build on established findings and develop new knowledge.

1.2.2 For you

Working to make your research reproducible does require extra upfront effort. For example, you need to put effort into learning the tools of reproducible research by doing things such as reading this book. But beyond the clear benefits for science, why should you make this effort? Using reproducible research tools can make your research process more effective and (hopefully) ultimately easier.

⁶See the American Physical Society’s website at http://www.aps.org/policy/statements/99_6.cfm. See also Fomel and Claerbout (2009).

Better work habits

Making a project reproducible from the start encourages you to use better work habits. It can spur you to more effectively plan and organize your research. It should push you to bring your data and source code up to a higher level of quality than you might if you “thought ‘no one was looking’” (Donoho, 2010, 386). This forces you to root out errors—a ubiquitous part of computational research—earlier in the research process (Donoho, 2010, 385). Clear documentation also makes it easier to find errors.⁷

Reproducible research needs to be stored so that other researchers can actually access the data and source code. By taking steps to make your research accessible for others, you are also making it easier for yourself to find your data and methods when you revise your work or begin a new project. You are avoiding personal effort duplication, allowing you to cumulatively build on your own work more effectively.

Better teamwork

The steps you take to make sure an independent researcher can figure out what you have done also make it easier for your collaborators to understand your work and build on it. This applies not only to current collaborators, but also to future collaborators. Bringing new members of a research team up to speed on a cumulatively growing research project is faster if they can easily understand what has been done already (Donoho, 2010, 386).

Changes are easier

A third person may or may not actually reproduce your research even if you make it easy for them to do so. But, *you will almost certainly reproduce parts or even all of your own research*. No actual research process is completely linear. You almost never gather data, run analyses, and present your results without going backwards to add variables, make changes to your statistical models, create new graphs, alter results tables in light of new findings, and so on. You will probably try to make these changes long after you last worked on the project and long since you remembered the details of how you did it. Whether your changes are because of journal reviewers’ and conference participants’ comments or you discover that new and better data has been made available since beginning the project, designing your research to be reproducible from the start makes it much easier to change things later on.

Dynamic reproducible documents make changes much easier. Changes made to

⁷Of course, it’s important to keep in mind that reproducibility is “neither necessary nor sufficient to prevent mistakes” (Stodden, 2009a).