

Introduction to R

Ryan Donovan

2024-01-18

Contents

1	Introduction	5
1.1	Who is this resource for?	5
1.2	Should I learn R?	5
1.3	What will I learn to do in R?	6
1.4	What will I not learn to do in R?	6
1.5	Where and when will the workshops take place?	7
1.6	Are there any prerequisites for taking this course?	8
1.7	Do I need to bring a laptop to the class?	8
2	Getting Started with RStudio	9
2.1	Let's test this out	9
3	Prerequisites	11
4	Get your GitBook	13
5	Editing the book	19
5.1	Creating new chapters	19
5.2	Linking across chapters	20
5.3	Advanced editing	20
6	Figures and tables	21

7	Examples	23
7.1	Statistics with R (H. Quene)	23
7.2	Theory Construction and Statistical Modeling (C. J. van Lissa) .	23
7.3	Doing Meta-Analysis in R (C. J. van Lissa)	23
7.4	Métodos quantitativos em Psicologia com R (L. Anunciação) . .	24
8	Open Educational Resources	25
9	Compatibility with existing systems	27
9.1	Add a hyperlink	27
9.2	Embed the whole book	27
10	License your GitBook	29

Chapter 1

Introduction

This set of workshops describes how to use R to import, clean, and process psychological data. All materials, data, and information in these workshops are used for educational purposes only. This document should only be shared within the University of Galway's School of Psychology, and is not intended for widespread dissemination. The workshop's e-book is very much in its draft stages and will be updated and fine-tuned in the future. Several materials are adapted from several online resources on teaching R.

1.1 Who is this resource for?

These workshops are designed to help people who come from a Psychology or social science background learn the necessary programming skills to use R effectively in their research. These workshops are designed to help people who have no programming experience whatsoever to learn the necessary programming skills and ideas that you will need to conduct typically statistical techniques in Psychology (e.g., Power Analyses, Correlation, ANOVA, Regression, Mediation, Moderation).

These workshops are ***not*** for people interested in learning about statistical theory or the who, what, where's of any of the aforementioned statistical techniques. I want these workshops to focus entirely on how to do statistical analyses in R - I assume you know the rest or know how to access that information.

1.2 Should I learn R?

There are a lot of reasons to learn R.

Psychological research is increasingly moving towards open-science practices. One of the key principles of open-science is that all aspects of data handling - including data wrangling, pre-processing, processing, and output generation are openly accessible. This is not only an abstract want or desire, several top-tier journals require that you submit R scripts along with any manuscripts. If you don't know how to use R (or at least no one in your lab does), then this will put you at an disadvantage.

R enables you to import, clean, analysis and publish manuscripts from R itself. You do not have to switch between SPSS, Excel, and Word or any other software. You can conduct your statistical analysis directly in R and have that "uploaded" directly to your manuscript. In the long run, this will save you so much time and energy.

R is capable of more than statistical analysis. You can create websites, documents, and books in R. This workshop textbook was developed in R! While these initial workshops will not be discussing how to do this (although it is something that I would like to do in the future), I wanted to mention it as an example how powerful R can be.

1.3 What will I learn to do in R?

The following workshops will teach you on how to conduct statistical analysis in R.

R is a statistical programming language that enables you to wrangle, process, and analyse data. By the end of these workshops you should be able to import a data file into R, do some processing and cleaning, compute descriptive and inferential statistics, generate nice visualizations, and output your results.

The learning objectives of this course are:

- Learn how to import and create data sets in R.
- Learn and apply basic programming concepts such as data types, functions, and loops.
- Learn key techniques for data cleaning in R to enable statistical analysis
- Learn how to create APA-standard graphs in R
- Learn how to deal with errors or bugs with R code
- Learn how to export data.

1.4 What will I not learn to do in R?

This is not an exhaustive introduction to R. Similar to human languages, programming languages like R are vast and will take years to master. After this

course, you will still be considered a “newbie” in R. But the material covered here will at least provide you a solid foundation in R, enabling you to go ahead and pick up further skills if required as you go on.

This course will teach you data cleaning and wrangling skills that you will enable you to wrangle and clean a lot of data collected on Gorilla or Qualtrics. But you will not be able to easily handle all data cleaning problems you are likely to find out in the “wild” world of messy data. Such data sets can be uniquely messy, and even experience R programmers will need to bash their head against the wall a few times to figure out a way to clean that data set entirely in R. If you have a particularly messy data set, you might still need to use other programs (e.g., Excel) to clean it up first before importing it to R.

Similarly, do not expect to be fluent in the concepts you learn here after these workshops. It will take practice to become fluent. You might need to refer to these materials or look up help repeatedly when using R on real-life data sets. That’s normal.

This workshop is heavily focused on the tidyverse approach to R. The tidyverse is a particular philosophical approach on how to use R (more on that later). The other approach would be to use base R. This can incite violent debates in R communities on which approach is better. We will focus mainly on tidyverse and use some base R.

This workshop does not teach you on how to use R Markdown. R Markdown is a package in R that enables you to write reproducible and dynamic reports with R that can be converted into word documents, PDFs, websites, power point presentations, books, and much more. That will be covered in the intermediate workshop program.

1.5 Where and when will the workshops take place?

The sessions will take place in **AMB-G035** (Psychology PC Suite). The schedule for the sessions is as follows:

- Feb 7th: Introduction to R and RStudio
- Feb 14th: Basic Programming (Part I)
- Feb 21st: Basic Programming (Part II)
- Feb 28th: Data Cleaning in R (Part I)
- March 6th: Data Cleaning in R (Part II)
- March 13th: Data Visualization
- March 20th: Running Inferential Statistical Tests in R (Part I)
- March 27th: Running Inferential Statistical Tests in R (Part II)

Each session is on a Wednesday and will run between 11:00 - 13:00.

1.6 Are there any prerequisites for taking this course?

None at all. This course is beginners friendly. You also do not need to purchase anything (e.g., textbook, or software).

1.7 Do I need to bring a laptop to the class?

If you have a laptop that you work on, I strongly encourage you to bring it. That way we can get R and RStudio installed onto your laptop and you'll be able to run R outside of the classroom.

If you work with a desktop, don't worry. The lab space will have computers that you can sign-in and work on and use R.

Chapter 2

Getting Started with RStudio

2.1 Let's test this out

Chapter 3

Prerequisites

This GitBook is created in Rstudio, using the `bookdown` package. To get your system set up correctly, you have to install several software packages, and register on GitHub. You only have to perform these steps once, and the entire process should take approximately 1 hour if you start from scratch. In case some software is already installed on your system, you can skip related steps. Follow these steps in order:

1. Install R from cloud.r-project.org
2. Install Rstudio Desktop (Free) from rstudio.com
3. Install Git from git-scm.com. Use the default, recommended settings. It is especially important to leave these settings selected:
 - Git from the command line and also from third party software
 - Use the OpenSSL library
 - Checkout Windows-style, commit Unix-style line endings
 - Enable Git Credential Manager
 - If you run into any trouble, a more comprehensive tutorial on installing Git is available at happygitwithr.com.
4. Register on GitHub
 - Go to <https://github.com/> and click “Sign up”. Choose an “Individual”, “Free” plan.
 - Request a free academic upgrade. This allows you to create *private repositories*, which are only visible to you and selected collaborators, and can be made public when your work is published.

5. Connect Rstudio to Git and Github (for more support, see this Rstudio article, and this blog post)
 - a. Open Rstudio, open the Tools menu, click *Global Options*, and click *Git/SVN*
 - b. Verify that *Enable version control interface for RStudio projects* is selected
 - c. Verify that *Git executable:* shows the location of git.exe. If it is missing, manually fix the location of the file.
 - d. Click *Create RSA Key*. Do not enter a passphrase. Press *Create*. A window with some information will open, which you can close.
 - e. Click *View public key*, and copy the entire text to the clipboard.
 - f. Close Rstudio (it might offer to restart by itself; this is fine)
 - g. Go to <https://github.com>
 - h. Click your user icon, click *Settings*, and then select the *SSH and GPG keys* tab.
 - i. Click *New SSH key*. Give it an arbitrary name (e.g., your computer ID), and paste the public key from your clipboard into the box labeled “Key”.
 - j. Open Rstudio again (unless it restarted by itself)
6. Install all required packages by running the following code in the Rstudio console. Be prepared for three contingencies:
 - If you receive any error saying *There is no package called XYZ*, then run the code `install.packages("XYZ")`
 - If you are prompted to update packages, just press **3: None**. Updating packages this way in an interactive session sometimes leads to errors, if the packages are loaded.
 - If you see a pop-up dialog asking *Do you want to install from sources the package which needs compilation?*, click *No*. If this leads to errors, then please follow *Step 3* from this online guide, and run `install.packages("devtools")`. This will take a long time, but will allow you to install packages from source.

Run the following code to install the required packages:

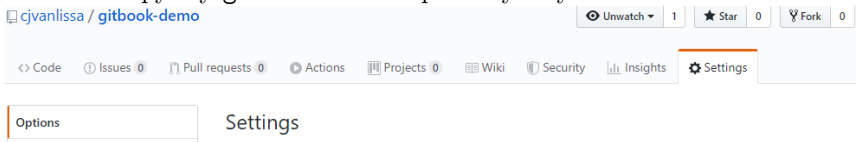
```
install.packages("bookdown")
install.packages("tinytex")
tinytex::install_tinytex()
git2r::config(global = TRUE, user.name = "your.name", user.email = "your.email")
```

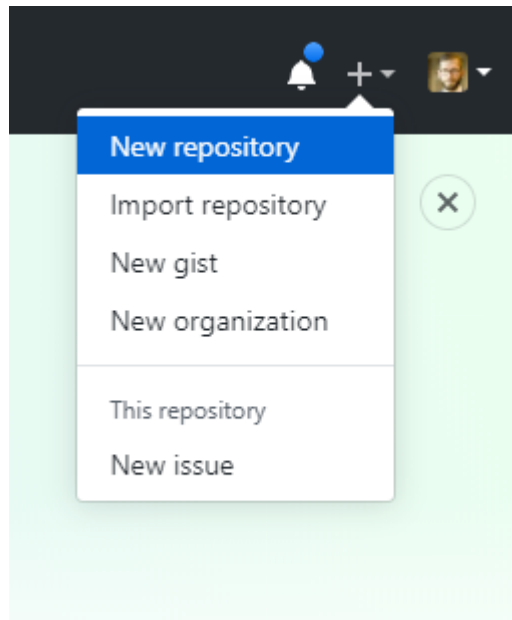
That's it! Everything should be installed and connected now.

Chapter 4

Get your GitBook

To get your GitBook, you should follow these steps:

1. Go to <https://github.com/cjvanlissa/gitbook-demo>
2. In the top right of the page, click **Fork**.
This will copy my `gitbook-demo` repository to your GitHub account.

3. My repository is now copied to your account. It is a template repository, which means that you can create a *new repository* based on this one.
4. Create a new repository for your own GitBook. Create one for a course you've been wanting to update. In the top-right corner of the GitHub website, click the + icon, and select "New repository":



5. In the dialog, select the `gitbook-demo` as “Repository template”, and give the repository an appropriate name for your course. Then, press **Create repository**:

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Repository template

Start your repository with a template repository's contents.

 `cjvanlissa/gitbook-demo` ▼

Owner

Repository name *

 `cjvanlissa` ▼

/ `your_course_name` ✓

Great repository names are short and memorable. Need inspiration? How about [jubilant-memory](#)?

Description (optional)



Public

Anyone can see this repository. You choose who can commit.

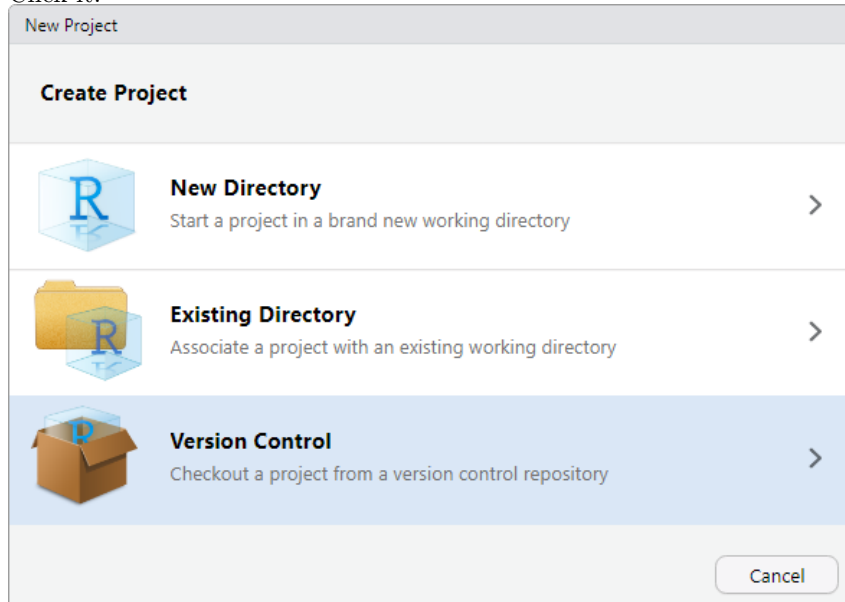


Private

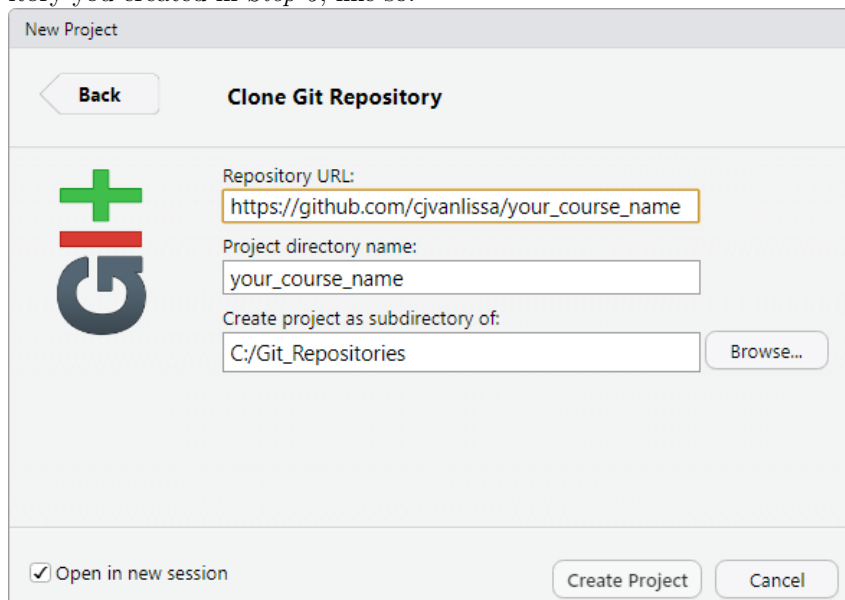
You choose who can see and commit to this repository.

Create repository

6. Now, go back to Rstudio on your computer. In Rstudio, click **File** > **New Project**. A dialog will open. If you set up Rstudio with Git correctly, the dialog should have an option to create a new project from Version control. Click it:

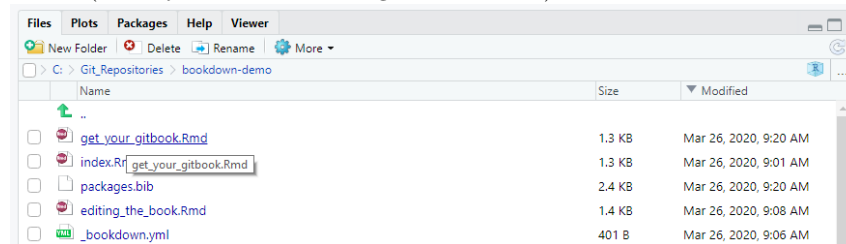


7. In the next dialog window, you should copy the URL of the GitHub repository you created in *Step 5*, like so:

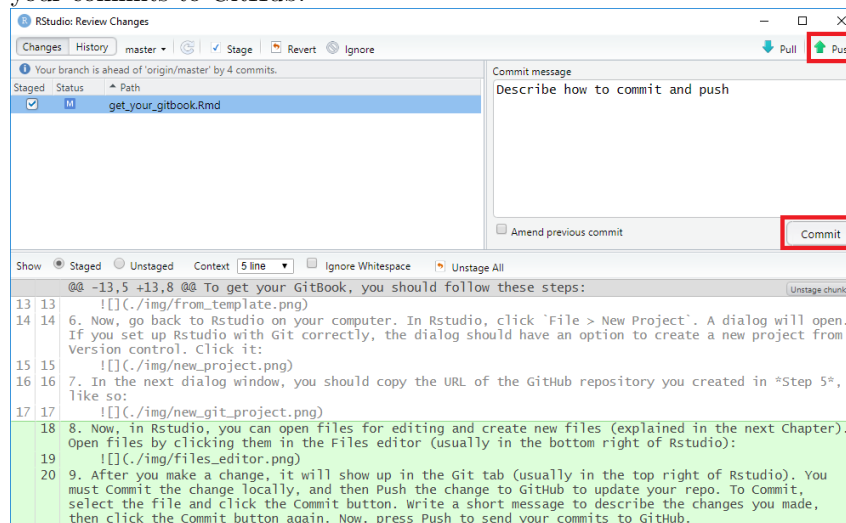


8. Now, in Rstudio, you can open files for editing and create new files (ex-

plained in the next Chapter). Open files by clicking them in the Files editor (usually in the bottom right of Rstudio):



9. After you make a change, it will show up in the Git tab (usually in the top right of Rstudio). You must Commit the change locally, and then Push the change to GitHub to update your repo. To Commit, select the file and click the Commit button. Write a short message to describe the changes you made, then click the Commit button again. Now, press Push to send your commits to GitHub.



10. To render your book as a GitBook, you must “Build” it. In the top-right panel of Rstudio, you see a “Build” tab. In this tab, simply click the “Build Book” button to build your book. You should see a lot of rendering messages, until a window pops up with your brand new GitBook. If you get errors at this stage, you probably made a mistake in preparing your system (see the previous Chapter).


```

" C:/Program Files/RStudio/bin/pandoc/pandoc" +RTS -K512m -RTS bookdown-demo.utf8.md --to latex
--from markdown+autolink_bare_uris+tex_math_single_backslash --output bookdown-demo.tex --self-contained
--table-of-contents --toc-depth 2 --number-sections --highlight-style tango --pdf-engine pdflatex
--natbib --include-in-header preamble.tex --variable graphics --lua-filter "C:/Program Files/R/R-3.6.2/library/rmarkdown/rmd/lua/pagebreak.lua"
--lua-filter "C:/Program Files/R/R-3.6.2/library/rmarkdown/rmd/lua/latex-div.lua" --wrap preserve --variable tables=y
s --standalone -Mhas-frontmatter=false

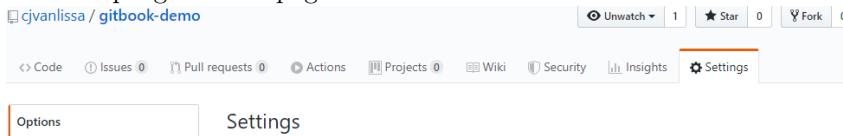
Output created: docs/index.html

processing file: bookdown-demo.Rmd
output file: bookdown-demo.knit.md

Output created: docs/bookdown-demo.pdf

```

11. Building the book generated a lot of new files in the `./docs` directory. This directory contains the website files for your GitBook. Open the Git tab again, verify that the `./docs` directory is listed, and Commit and Push all of these new files as described in *Step 9*.
12. There is only one last remaining task: To publish your GitBook on GitHub pages. Once you do this, any change to the `./docs` folder that you push to GitHub will lead to an immediate update of your GitBook website. Go back to the GitHub page for your Repository. Click on the **Settings** tab on the top right of the page:



13. On the Settings page, scroll all the way down until you reach a section called **GitHub Pages**. There, under the "Source" heading, click the word **None**, and select **master branch /docs folder**. When you select it, the page will update, and if you scroll back down to the **GitHub Pages** section, you will see the URL where your GitBook is published. The first time, it will take a few minutes for your GitBook to come online. When you publish updates to the GitBook however (simply by following *Step 11* again), the update will be near-instantaneous. The Pages section should now look like this (and that is hopefully the link where you found this book):

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

Your site is ready to be published at <https://civanlissa.github.io/gitbook-demo/>.

Source

Your GitHub Pages site is currently being built from the `/docs` folder in the `master` branch. [Learn more.](#)

master branch /docs folder ▾

Chapter 5

Editing the book

The contents of the book are written in **RMarkdown**. You can use any formatting code that Pandoc's Markdown supports, e.g., a math equation $a^2 + b^2 = c^2$. Moreover, you can include chunks of R-code, like this:

The results of these chunks can be rendered to the GitBook:

```
## [1] "This is an R-command!"
```

To edit the book, you can change the text in the `.Rmd` files. Each Rmd file should contain **one and only one** chapter. A chapter is defined by the first-level heading `#`, e.g., `# Editing the book`.

Any sub-headings within the chapter are indicated with several `#` signs, e.g., `##` (level 2) and `###` (level 3).

5.1 Creating new chapters

To create a new chapter, you must follow two steps: 1) Create the file, and 2) Include it in the list of chapters.

First, to create the file for a new chapter in Rstudio, click **File > New File > Text file**. At the top of the file, write your chapter heading, as explained above. Then, click **File > Save**. Save the file as `.Rmd`, without spaces in the file name, e.g.: `editing_the_book.Rmd`.

Second, to include it in the list of chapters, open the file `_bookdown.yml` (click it in the Files explorer in the bottom right of Rstudio). This file has a list of `.Rmd` files to be included in the book. In this example, the list looks like this:

```
tmp <- readLines("_bookdown.yml")
cat(tmp[grep("^rmd_files", tmp):grep("references\\.Rmd", tmp)], sep = "\n")
```

```
rmd_files: ["index.Rmd", "02-rstudio.Rmd", "prerequisites.Rmd", "get_your_gitbook.Rmd",
"editing_the_book.Rmd", "figures_tables.Rmd", "examples.Rmd", "open_educational.Rmd",
"use_in_course.Rmd", "licenses.Rmd", "references.Rmd"]
```

Insert the file name of your new chapter in the desired position in this list.

5.2 Linking across chapters

You can label chapter and section titles using `{#label}` after them. The labels can be used as cross-references. For example, we can link to Chapter 6. If you do not manually label chapters, there will be automatic labels anyway, e.g., Chapter 7.

5.3 Advanced editing

The convenient Rmarkdown Cheat Sheet by Rstudio covers most of the knowledge required for advanced Rmarkdown editing. You can print it out and stick it to your wall!

Chapter 6

Figures and tables

Figures and tables with captions will be placed in `figure` and `table` environments, respectively.

```
par(mar = c(4, 4, .1, .1))  
plot(pressure, type = 'b', pch = 19)
```

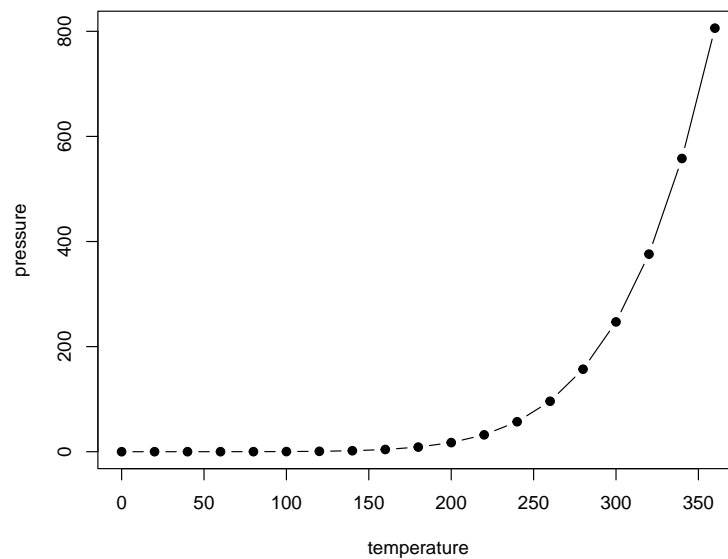


Figure 6.1: Here is a nice figure!

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure 6.1. Similarly, you can reference tables generated from `knitr::kable()`, e.g., see Table 6.1.

Table 6.1: Here is a nice table!

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa
4.6	3.4	1.4	0.3	setosa
5.0	3.4	1.5	0.2	setosa
4.4	2.9	1.4	0.2	setosa
4.9	3.1	1.5	0.1	setosa
5.4	3.7	1.5	0.2	setosa
4.8	3.4	1.6	0.2	setosa
4.8	3.0	1.4	0.1	setosa
4.3	3.0	1.1	0.1	setosa
5.8	4.0	1.2	0.2	setosa
5.7	4.4	1.5	0.4	setosa
5.4	3.9	1.3	0.4	setosa
5.1	3.5	1.4	0.3	setosa
5.7	3.8	1.7	0.3	setosa
5.1	3.8	1.5	0.3	setosa

```
knitr::kable(
  head(iris, 20), caption = 'Here is a nice table!',
  booktabs = TRUE
)
```

You can write citations, too. For example, we are using the **bookdown** package (Xie, 2023) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

Chapter 7

Examples

Here are some examples of other GitBooks for courses; if you want to have your GitBook added to the list, please send a Pull Request (here's how to send a pull request).

7.1 Statistics with R (H. Quene)

<https://hugoquene.github.io/emlar2020>

A GitBook for a tutorial on *Statistics with R (Basics)*, held as part of the workshop on Experimental Methods in Language Acquisition Research (EM-LAR, <https://emlar.wp.hum.uu.nl/>), Utrecht, on 17 April 2020. This compact introduction helps you with your first steps into R.

7.2 Theory Construction and Statistical Modeling (C. J. van Lissa)

<http://cjvanlissa.github.io/TCSM>

A GitBook for the course “*Theory Construction and Statistical Modeling*”, with some interesting code, for example: Blocks of answers to the tutorial questions that can be collapsed and expanded.

7.3 Doing Meta-Analysis in R (C. J. van Lissa)

<http://cjvanlissa.github.io/Doing-Meta-Analysis-in-R>

A GitBook on doing meta-analysis in R, based on the book ‘Doing Meta-Analysis in R’, by Mathias Harrer, Pim Cuijpers, & David Ebert, and adapted to focus on the metafor package, and exploring heterogeneity using metaforest. The original can be found here: https://bookdown.org/MathiasHarrer/Doing_Meta_Analysis_in_R/

7.4 Métodos quantitativos em Psicologia com R (L. Anunciação)

<https://anovabr.github.io/mqt/>

This book provides a short and to-the-point exposition on the essentials of statistics, and was written for undergraduate students at the Pontifical Catholic University of Rio de Janeiro (PUC-Rio). To a lesser degree, the mathematical modeling of statistical questions will be addressed. This book might be relevant for Portuguese-speaking students who enroll for laboratory-based statistics and anyone who wants to learn R.

Chapter 8

Open Educational Resources

UNESCO defines Open Educational Resources as *teaching, learning and research materials in any medium – digital or otherwise – that reside in the public domain or have been released under an open license that permits no-cost access, use, adaptation and redistribution by others with no or limited restrictions.*

Open Educational resources can help lighten the workload on individual teachers, who can collaborate with the development of high-quality open access resources, instead of having to develop their own proprietary materials from scratch. Moreover, Open Educational resources are inclusive, lowering the barrier to knowledge acquisition for learners around the world, and enabling lifelong learning for those outside academia.

Many universities support Open Educational Resources. Here are just a few (feel free to send a pull request with other relevant resources).

- **OER Commons:** A freely accessible online library of open educational resources.
- **Utrecht University Figshare:** Open learning objects from Utrecht University.
- **Johns Hopkins University OCW:** Open public health courses and materials.
- **University of Pittsburgh OER:** Big List of Open Educational Resources.
- **MERLOT:** Online learning and support materials and content creation tools, led by an international community of educators, learners and researchers.

Chapter 9

Compatibility with existing systems

Many universities offer digital platforms for learning. You might wish to embed your GitBook within these existing systems. Here are two ways in which you might do that. Currently, this section only discusses BlackBoard, but the same principles should apply to other platforms.

9.1 Add a hyperlink

You can add a link to your GitBook in the BlackBoard course menu by following this tutorial.

9.2 Embed the whole book

You can add a Blank Page to your BlackBoard course menu, and fill that page with a full-size “iframe” - a web page within the web page. This tutorial explains how to do it. It is possible that your university is blocking this feature, however.

Chapter 10

License your GitBook

In the spirit of Open Science, it is good to think about making your course materials Open Source. That means that other people can use them. In principle, if you publish materials online without license information, you hold the copyright to those materials. If you want them to be Open Source, you must include a license. It is not always obvious what license to choose.

The Creative Commons licenses are typically suitable for course materials. This GitBook, for example, is licensed under CC-BY 4.0. That means you can use and remix it as you like, but you must credit the original source.

If your project is more focused on software or source code, consider using the GNU GPL v3 license instead.

You can find more information about the Creative Commons Licenses [here](#). Specific licenses that might be useful are:

- CC0 (“No Rights Reserved”), everybody can do what they want with your work.
- CC-BY 4.0 (“Attribution”), everybody can do what they want with your work, but they must credit you. Note that this license may not be suitable for software or source code!

For compatibility between CC and GNU licenses, see [this FAQ](#).

Bibliography

Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.

Xie, Y. (2023). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.37.