

Case Studies in Reproducible Research: a spring seminar at UCSC

Eric C. Anderson, Kristen C. Ruegg, Tina Cheng, and the students of EEB 295

2017-03-24

Contents

Chapter 1

Course Overview

This is the home for the lecture notes for a proposed course in data analysis and reproducible research using R, Rstudio, and GitHub.

The seminar is called, “Case Studies in Reproducible Research,” but we utter that title with the caveat that, although the organizers have quite a few case studies that they could spin up for this course, the case studies we will be studying in this course are going to be actual research projects that *you*—the participants—are working on. You’re gonna bring ‘em, and we are going to collectively help you wrassle them into a reasonable and reproducible data analysis. In the process we will touch on a number of elements of data analysis with R.

For the interested, these materials were all prepared using RStudio’s bookdown package.

1.1 Meeting Times, Location, Requirements

Intended to be Friday afternoons, 1:00–2:40 PM, somewhere at Long Marine Lab.

Students must bring a laptop to do examples during the seminar, and all students are expected to have a data set that they are in the midst of analyzing (or upon which they hope to commence analysis soon!) for a research project. We will

1.2 The origin of this seminar

The idea for this course was floated by Tina Cheng who was planning to lead a seminar in spring 2017 based in part on Eric C. Anderson’s “Reproducible Research Course”, taught at the Southwest Fisheries Science Center in the fall of 2014. Although going over those notes might have been a reasonable exercise, it turns out that a lot has changed in the world of data analysis since fall 2014, and the notes from that course are, today, a little bit dated.

We have been particularly excited by the ascendancy of Hadley Wickham’s Tidyverse approach to data analysis, and the tremendous development of a variety of tools developed by RStudio for integrating report generation and data analysis into reproducible workflows. In fact, Eric has been saying for the last year that if he were to teach another course on data analysis it would be structured completely differently than his “Reproducible Research Course”. So, it was clearly time for him to stop talking and help put together an updated and different course.

At the same time, in working on our own projects and in helping others, we have consistently found that the most effective way for anyone to learn data analysis is to ensure that it is immediately relevant to whatever ongoing research project is currently consuming them. Therefore, in the current seminar, we are

hoping to spend at least half of our time “workshopping” the data sets that seminar participants are actually involved in analyzing. Together we will help students wrestle their data, analyses, and ideas into a single, well-organized RStudio project under version control with git. Therefore, every student should come to this course with a data set and an associated analysis project. This is also not a “first course in R”. Students coming to the course should have at least a little familiarity with using R.

1.3 Course Organizers

Kristen C. Ruegg We can put description here

Eric C. Anderson Eric is a statistician who specializes in genetic data. Since 2003 he has worked at the NMFS Southwest Fisheries Science Center in Santa Cruz. Although much of his statistical research involves the development of computationally intensive methods for specialized analyses of genetic data, he has been involved in a variety of data analysis projects at NMFS and with collaborators worldwide. Eric was an early adherent to reproducible research principles and continues, as such, performing most of his research and data analysis in the open and publicly available on GitHub (find his GitHub page here). In 2014, he taught the “Reproducible Research Course” at NMFS, and is excited to provide an updated version, focusing more, this time, on the recently developed “tidyverse”.

Tina Cheng something here

1.4 Course Goals

The goal of this course is for scientists, researchers, and students to learn to:

- properly store, manage, and distribute their data in a *tidy* format
- consolidate their digital research materials and analyses into well-organized RStudio projects.
- use the tools of the tidyverse to manipulate and analyze those data sets
- integrate data analysis with report generation and article preparation using the Rmarkdown format and using R Notebooks
- use git version control software and GitHub to effectively manage data and source code, collaborate efficiently with other researchers, and neatly package their research.

By the end of the course, the hope is that we will all have mastered strategies allowing us to use the above-listed, freely-available and open-source tools for conducting research in a reproducible fashion. The ideal we will be striving for is to be able to start from a raw data set and then write a computer program that conducts all the cleaning, manipulation, and analysis of the data, and presentation of the results, in an automated fashion. Carrying out analysis and report-generation in this way carries a number of advantages to the researcher:

1. Newly-collected data can be integrated easily into your analysis.
2. If a mistake is found in one section of your analysis, it is not terribly onerous to correct it and then re-run all the downstream analyses.
3. Revising a manuscript to address referee comments can be done quickly.
4. Years after publication, the exact steps taken to analyze the data will still be available should anyone ask you how, exactly, you did an analysis!
5. If you have to conduct similar analyses and produce similar reports on a regular basis with new data each time, you might be able to do this readily by merely updating your data and then automatically producing the entire report.
6. If someone finds an error in your work, they can fix it and then easily show you exactly what they did to fix it.

Additionally, packaging one’s research in a reproducible fashion is beneficial to the research community. Others that would like to confirm your results can do so easily. If someone has concerns about exactly how a particular analysis was carried out, they can find the precise details in the code that you wrote to

do it. Someone wanting to apply your methods to their own data can easily do so, and, finally, if we are all transparent and open about the methods that we use, then everyone can learn more quickly from their colleagues.

In many fields today, publication of research requires the submission of the original data to a publicly-available data repository. Currently, several journals require that all analyses be packaged in a clear and transparent fashion for easy reproduction of the results, and I predict that trend will continue until most, if not all, journals will require that data analyses be available in easily reproduced formats. This course will help scientists prepare themselves for this eventuality. In the process, you will probably find that conducting your research in a reproducible fashion helps you work more efficiently (and perhaps even more enjoyably!)

1.5 Weekly Syllabus

1.5.1 Week 1 — Introduction and Getting Your Workspace Set Up

- At the end of this session we want to make sure that everyone has R, RStudio, and Git installed on their systems, and that they are working as expected.
- Additionally, everyone should have a free account on GitHub.
- And finally we need everyone's email address.

1.5.2 Week 2 — RStudio project organization; using git and GitHub; Quick RMarkdown

After this, students are going to have to put their own data into their own repositories and write a README.Rmd and make a README.md out of it.

1.5.3 Week 3 — Tibbles. Reading data in. Data rectangling

- Reading data into the data frames.
- `read.table` and `read.csv`
- tibbles
- The `readr` package
- Data types in the different columns and quick data sanity checks.
- A few different gotcha's
- Saving and reading data in R formats. `saveRDS` and `readRDS`.

1.5.4 Week 4 —

Chapter 2

Applications

Some *significant* applications are demonstrated in this chapter.

2.1 Example one

2.2 Example two

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter `??`. If you do not manually label them, there will be automatic labels anyway, e.g., Chapter `??`.

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)
```

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

```
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 (?) in this sample book, which was built on top of R Markdown and **knitr** (?).

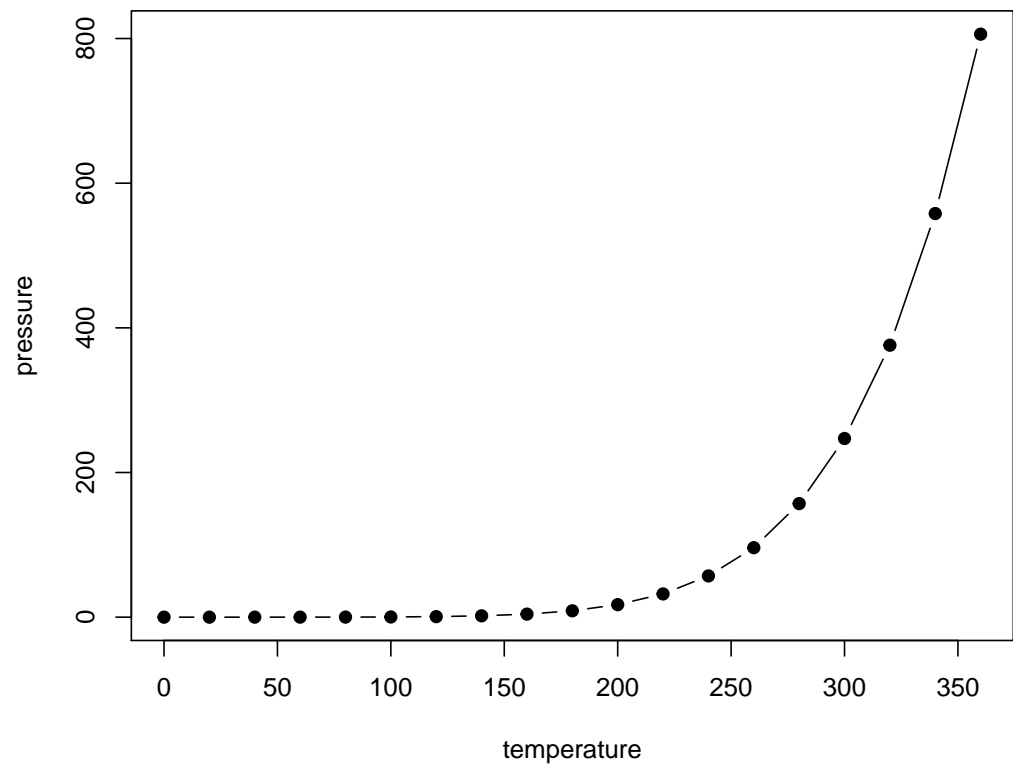


Figure 2.1: Here is a nice figure!

Table 2.1: 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