Open Science Skills in R

Christelinda Laureijs, Elizabeth Stregger, Dr. Julia Riley

February 04, 2025

Contents

Introduction

This book is part of a workshop series led by Christelinda Laureijs, Elizabeth Stregger, and Dr. Julia Riley. We are a group of enthusiastic R coders with a passion for open science and educational outreach! We created this book for three reasons:

- To provide curious learners with a useful resource beyond the workshops.
- To give a quick overview to anyone who wants to get started with setting up R, managing their projects, and writing papers with R.
- To gather tips, tricks and links to other resources all in one document.

0.1 Workshops

This book is divided into three chapters associated with each workshop:

1. Welcome and Introducing a Tidy Workflow

These skills will help in with organization for your research project, collaborations, and ensuring transparency and broad impact.

- How to set up R, RStudio, LaTeX, and required packages
- Tidy projects and data management strategies
- Tidy coding practices
- Organizing your research project

2. Git with it

After this section, you'll have an understanding of how you can track changes with Git, share your work with GitHub, reuse open code, and contribute to open projects.

- Setting up Git on your computer
- Creating a repository
- Tracking changes to your project

- Exploring the history of a project
- Collaborating in GitHub

3. Reproducible Scientific Writing with R

All the skills from the previous chapters come together in this last section. You will learn how to write a paper completely within R - no more copying and pasting between different platforms!

- Creating publication-ready figures and tables
- Managing citations
- Organizing a paper into chapters
- Customizing your paper
- Extract statistical values to insert directly into the paper

0.2 Files

Throughout this book, we will provide links to files in the GitHub repository for this book. To make it easier to work with files, we have sometimes attached files directly in the book. Here is an example of a `.csv` file that you can download:

What is Open Science?

You might be wondering: **"What is open science?"** It is the process of making the content and process of producing evidence and claims transparent and accessible to others. This involves making the entire research process - from project conceptualization to publication - open to all and transparent. There are a number of techniques that you can work on to improve transparency and reproducibility (the ability to repeat the same project and obtain the same results) of your work. In this workshop series, we will be covering a few skills that will help you advance science using open practices.

We don't have too much time to review the importance of open science practices and its philosophy in this workshop series, because we are focusing on skill building. **But, we encourage you to learn more through reading and engaging with additional resources on this topic. Here are a few we find helpful:**

- A Manifesto for Reproducible Science by Munafò et al. 2017
- We will not cover preregistration, which is simply specifying your research plan in advance of your study and submitting it to a registry. This is a important first step of a research project that adheres to open science practice. For more details, check out the Center for Open Science page on preregistration.
- The Center for Open Science also has a number of useful resources, and they also host their own repsoitory where data and code (even those tracked with version control) can be hosted and shared with others called the Open Science Framework.

Chapter 1

Welcome and Introducing a Tidy Workflow

Today we are going make sure everyone is set-up with the same software and R packages that will be used across this workshop series. Then, we will work through "tidy" principles for project organisation (including how to set-up an R Project), data management, and coding in R. These are skills that are critical to support open science and facilitate effective and productive collaborations.

A handout of the slides from the presentation for this workshop can be found in the attached PDF:

1.1 Let's Get Set Up!

This workshop relies on using software that requires a bit of preparation at the beginning to make sure we are all on the same page. You may have all or some of these installed, but the person next to you may not. Let's take some time to run through a checklist step-by-step of what needs to be installed. **Please be patient as we work to get everyone ready to start this workshop series.**

1. You will need software to work with spreadsheets. I'd recommend Excel, which MtA offers to everyone as part of Office365 available on the Mount Allison website. Sheets for MacOs or Google Sheets are probably fine too if you are more comfortable with these, but our team will have less capacity to help you.

2. Install the latest version of R for your operating system.

3. Install R Studio Desktop for your operating system. There are many ways to use R and you might have your own preference. But, RStudio is very user-friendly and will be using it for this workshop series.

4. Depending on your operating system (Windows), it is likely that you will also need to install RTools to compile new R packages. Please install it now.

5. The last software we need to install is LaTex, which facilitates the creation of PDF documents in R Markdown. There are a myriad of options, including MiKTeX, MacTeX, and TeX Live. I suggest that you use TinyTeX, which you can install with the R package **tinytex** (Xie 2024c) by running this code in the Console:

```r
install.packages('tinytex')
tinytex::install_tinytex()
```

You may run into issues, depending on how your operating system is set-up. The TinyTeX website has instructions for each operating system. It may be possible that you will need to manually install one of types of software that facilitates LaTex too. If that seems to be the case, you can access these for install at this website.

6. After installation of all these programs, etc. it is very important to ***restart your computer.*** A life tip - a new program won't be initiated to work on your computer if you don't restart it after it is installed! This is a critical part of the process.

7. After your computer restarts, now please open R Studio. We are going to be making use of some R packages repeatedly. So, let's just install them now, using the code below:

```r
install.packages(c('tidyverse', 'rmarkdown', 'ggplot2',
                   'scales', 'lazyWeave'))
```

Please note, that we will do our best to help with 'debugging' if things are not working correctly on your computer. But, you might have issues you need to work on during your own time. Here are a few steps that you can take to help:

- Read the documentation. If it is for a package or piece of software be sure to read and follow the instructions carefully.

- Google it. Google is your best friend. Google the package or piece of software, google the specific error message, and find those message board threads where people go to complain and find answers to their problems (likely GitHub and Stack Overflow). Go and find a solution for yourself!

We will also be making extensive use of **R markdown** in these workshops. Here are some resources specifically about R markdown, that are always helpful to keep on hand while coding:

- Rmarkdown Website that has explanations, tutorials, videos, etc.
- Rmarkdown Cheatsheet
- About how to make use of R markdown files as a beginner
- Rmarkdown the Definitive Guide is a textbook that has clear examples and information about how best to work in R markdown

---

1.2 Using *TIDY* Practices in your Open Science Projects

Now that we are all set-up, we are going to focus on why organisation is important at multiple scales of your project. Although it is likely not the most exciting or complex topic, ensuring your research project files, code, and data are in a consistent, expected format facilitates effective use of your time and productive collaborations. **Tidy, organised projects are the foundation of reproducible science.** This is why we are starting with this topic because it is the basis on which we will build the rest of our skills in this workshop series.

There are also **many online resources that are helpful** on these topics, and here is a list of a few of them:

- For **Tidy Projects**, some guidance on basic project management.
- For **Tidy Code**, some guidance on good practice for coding
- The Tidyverse Style Guide for help with writing **tidy code** in this particular coding style
- The research article by Hadley Wickham describing **Tidy Data** and why it is important
- A blog by Julia Lowndes and Allison Horst depicting and describing the importance of **Tidy Data**
- A research article by Karl Broman and Kara Woo describing important considerations for data organisation in spreadsheets for **Tidy Data**

1.2.1 Plan for the Workshop

Today's workshop is going to be a mix of lecture and activities. You have already downloaded the slides for the lecture, and after I introduce the concept of "tidy projects" we will start working on some activities. These are outlined below (on this website), and also on the presentation slides and within an editable Rmarkdown document. We will set-up that R markdown document together and I encourage you to use that file to take notes and work on the activities.

1.2.2 Tidy Projects: Activities and Discussion

First, **let's make our own R project** to work on the activities that will be happening during this workshop series. So, together, let's open R Studio. Then, we will create new R project and save it somewhere safe on our computers. Last, let's download the Rmarkdown file below and save it within the R project folder we just created:

Awesome! Let's open up that R markdown document together within our R project. You can see there are notes from today's workshop in this document. I will walk you through how to knit the document, and view it in a nicely formatted version. You will see there are places for you to fill in your own notes on the activities are going to do today. Feel free to fill it in as we go!

Now, let's **discuss as a group the concept of tidy projects** a little.

1. *Have a think and list all stakeholders that may conceivably have an interest in the outcomes of your research. That is, who will possibly benefit from your robust, reproducible science, in what way, and why?*
2. *Have a think & jot down down one potential positive consequences and one potential negative consequences of conducting open, reproducible research. Then we will chat about them as a group.*

Next, please take a bit of time to **create a tidy project template for a real or hypothetical research project** you are working on. Take some time to think about the common 'ingredients' of your project because that will help you create a flexible, generic project structure. Make sure to include a description for your files and let's make use of R markdown's list format to show any nesting of folders or files.

1.2.3 Tidy Code: Activity about Using the R package `stylr`

Install the R package `styler`. Refer to the code chunk above where we installed R packages. Please add a new code chunk below, and install `styler` and load it into the R environment by adding a line of code that says `library(styler)`. If you need help, please put your RED POST-IT on top of your laptop, and try to Google solutions until one of us arrives to help.

Run the chunk of R code below. It works! But it is hard to read and doesn't follow our 'tidy code' ideals.

```
x <- rnorm(15);mean(x);hist(x)
```

The R package `styler`'s default style transformation is the `tidyverse_style()`, which is what we will use in this workshop. Use it to fix the below R code chunk. To do this, select the text you want to fix. Then, click on the button titled "Addins" within R Studio. In the dropdown menu, click on "Style selection". Run the chunk of R code again.

The same thing happens in R Studio's Console! It is just much easier to read and understand.

1.2.4 Tidy Data: Activity about Making Messy Data Tidy

We will be working with a very messy dataset that is data from part of a long-term project on the effects of rodents and ants on the plant community, and has been running for almost 40 years and used in over 100 publications. The rodents are sampled on a series of 24 plots, with different experimental manipulations controlling which rodents are allowed to access which plots. We'll be working with a subset of the data which has been 'messed up' a bit for the purposes of this workshop. The 'mess', though artificial in the context of this dataset, is of the sort that I regularly come into contact (and you will to), so it's very much real in that sense.

Please download your own copy of the dataset before making any changes:

The data file contains:

- Three tabs: containing data from samples collected in 2013, 2014, and 2015, respectively
- Date Collected: date of collection of the sample
- Species: species identifier
- Plot: replicate plot identifier
- Weight: weight of the captured specimen in grams
- Sex: sex of the captured specimen

The mission here is simple, but will also be a bit challenging. Focus all of your tidy skills on the catastrophe that is `workshop-1-messy-survey.xlsx` to parse it into its cleanest, tidiest, and most useful self.

You can complete this by a mix of adjustments by hand and also in R. How you do it is up to you! But, because as we are making direct changes to the raw data, be sure to note down every change you make (especially if you are doing this by hand rather than using code in R). You can do this below (with a combination of R code within an R chunk, if you choose that option) or in a separate text file. It is up to you!

We will chat about what you have done as a group before we end the workshop. Thanks for your attention today!

In your R markdown file, document all the changes you made to the data file. This can be a bullet-point list, description, chunks of annotated R code, or a mixture of all three.

Chapter 2

Welcome Back! Let's Git With It.

Today we are going to add Git and GitHub to the toolkit we started building in the first workshop. In the last workshop, we saw that R and RStudio can be used to create reproducible projects. This workshop is all about adding collaboration and sharing to these projects. In other words, we're making the science open!

A handout of the slides from the presentation for this workshop can be found in the attached PDF:

2.1 Get Git and GitHub

We will need to spend some time getting software installed at the beginning of this workshop. Use your red sticky note to let us know that you are stuck or have a question. For those of you who already have this software installed, please help your neighbours if you can!

2.1.1 Register a GitHub account.

We recommend that you incorporate your name in your username so that future collaborators can find you. Use lowercase letters and dashes to prevent coding challenges. Do not include your current university or employer, since those may change.

Examples of good usernames: julia-riley, estregger, christelinda-laurejis

2.1.2 Install Git

1. Windows: Install Git for Windows

This version, from gitforwindows.org, is also known as "Git Bash" Modify one of the default installation prompts: select "Git from the command line and also 3rd-party software"

2. macOS: Install Git

Another option is to install the Xcode command line tools. This includes Git. In terminal, enter this code:

xcode-select –install

2.1.3 Configure Git from inside RStudio

Install the package "usethis" in the console. We'll use this package to set our user name and email. The user name here could include details about which computer you are using, if you use multiple computers for your R projects. Example: Stregger Desktop. The email address must be the email address you used to set up your GitHub account.

```
library(usethis)
use_git_config(user.name = "Participant Name", user.email = "participant@mta.ca")
usethis::git_default_branch_configure()
```

2.1.4 Connect Git to GitHub

Now that you have Git running on your computer, you need credentials to communicate with your GitHub account.

The easiest way to do this with the usethis package in R.

```
usethis::create_github_token()
```

This should open GitHub. Give the token a name. Again, if you use multiple computers, it is helpful to include the name or location of the computer so you can keep track of your tokens. You will need one for each computer.

Copy the personal access token and leave the browser window open until your credentials are set. We'll do that next.

```
gitcreds::gitcreds_set()
```

If this does not work for you, there are other types of keys, called SSH keys. There are instructions for setting up SSH keys in the Happy Git and GitHub for the useR guide.