

# Improve your workflow for reproducible science

Mine Çetinkaya-Rundel  
University of Edinburgh + Duke University + RStudio

 [bit.ly/repro-workflow](https://bit.ly/repro-workflow)

@minebocek   
mine-cetinkaya-rundel   
cetinkaya.mine@gmail.com 



The results in Table 1  
don't seem to correspond to  
those in Figure 2!

61

45

4

94

12

3

20

44



```
# set.seed  
set.seed(20190314)  
  
# generate 8 random numbers between 0 and 99  
runif(8, 0, 99) %>% round()
```

more than

percent

have tried and **failed** to reproduce  
**another** scientist's experiments

more than



percent

have tried and **failed** to reproduce  
their **own** experiments

Google Scholar yields

101,101

results containing the term **reproducibility crisis**  
just in **2020**



# setting the stage

# replicability

# same research question

# same results

# new data

# reproducibility

# same research question

# same results

# same data

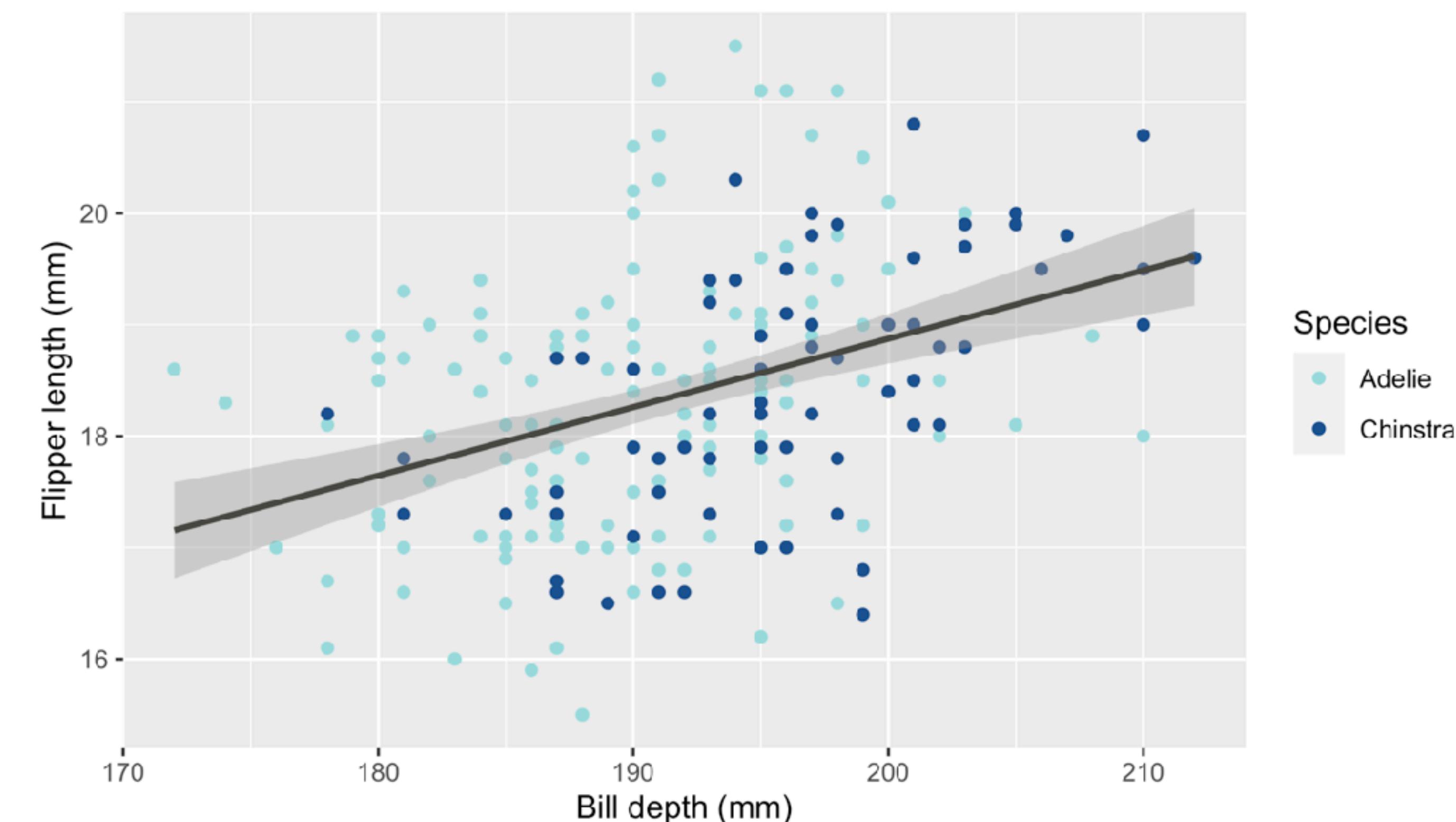


e.g.

**Table 1.** Regression output for predicting bill depth from flipper length.

| term              | estimate | std.error | statistic | p.value  |
|-------------------|----------|-----------|-----------|----------|
| (Intercept)       | 33.6     | 1.25      | 27.0      | 1.39e-86 |
| flipper_length_mm | -0.0820  | 0.00618   | -13.3     | 1.23e-32 |

**Figure 2.** Relationship between bill depth and flipper length.

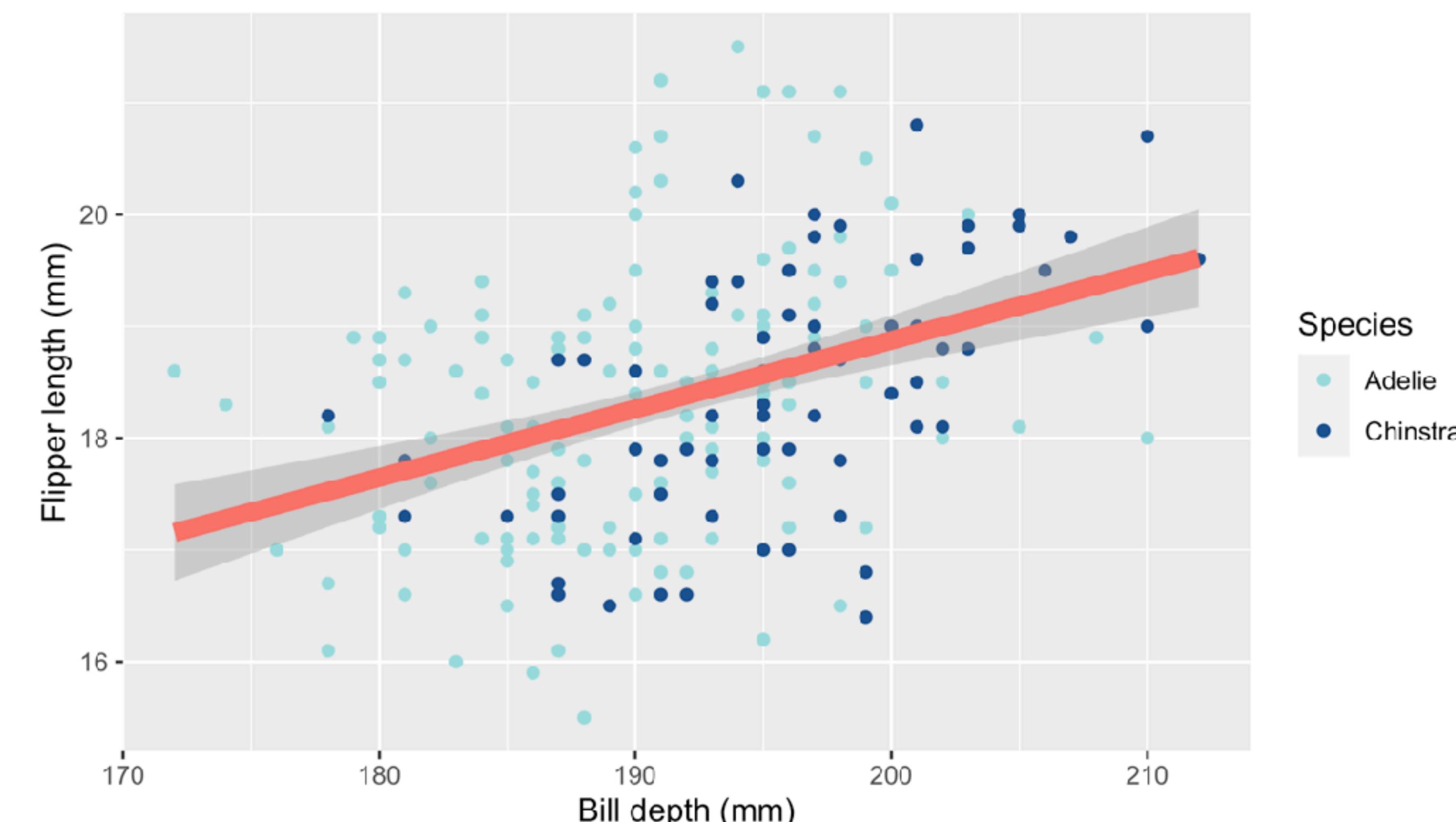


e.g.

**Table 1.** Regression output for predicting petal length from sepal width.

| term              | estimate       | std.error | statistic | p.value  |
|-------------------|----------------|-----------|-----------|----------|
| (Intercept)       | 33.6           | 1.25      | 27.0      | 1.39e-86 |
| flipper_length_mm | <b>-0.0820</b> | 0.00618   | -13.3     | 1.23e-32 |

**Figure 2.** Relationship between bill depth and flipper length.

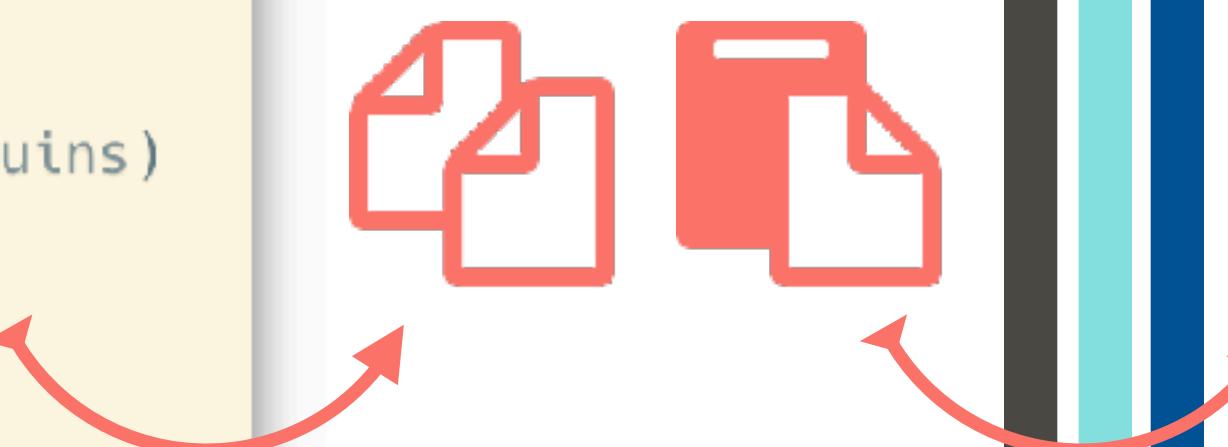


# analysis

# report



```
# fit model  
model <- lm(bill_depth_mm ~ flipper_length_mm, data = penguins)  
  
# print model summary  
tidy(model)
```



**Table 1.** Regression output for predicting bill depth from flipper length.

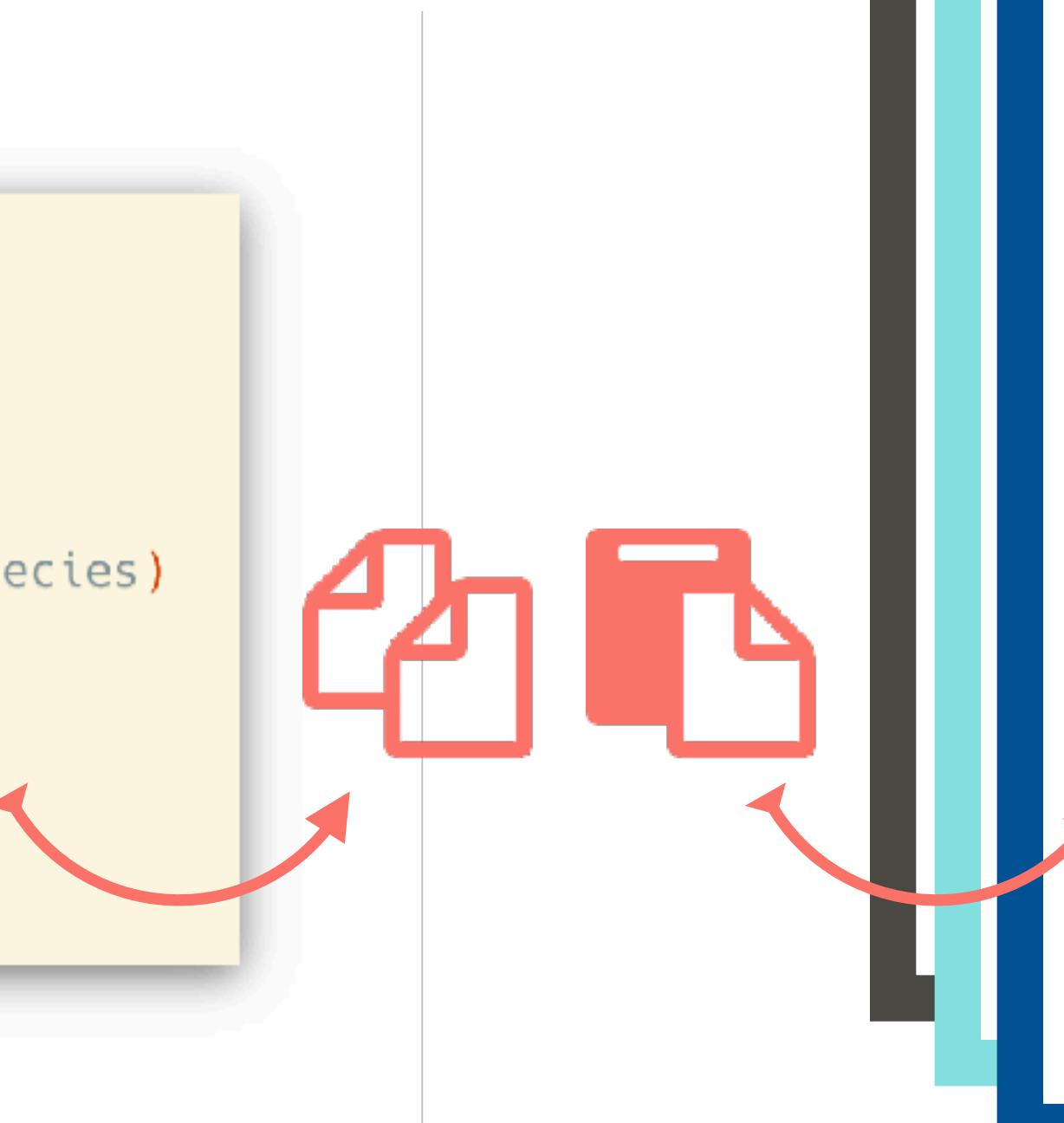
| term              | estimate | std.error | statistic | p.value  |
|-------------------|----------|-----------|-----------|----------|
| (Intercept)       | 33.6     | 1.25      | 27.0      | 1.39e-86 |
| flipper_length_mm | -0.0820  | 0.00618   | -13.3     | 1.23e-32 |

# analysis

# report



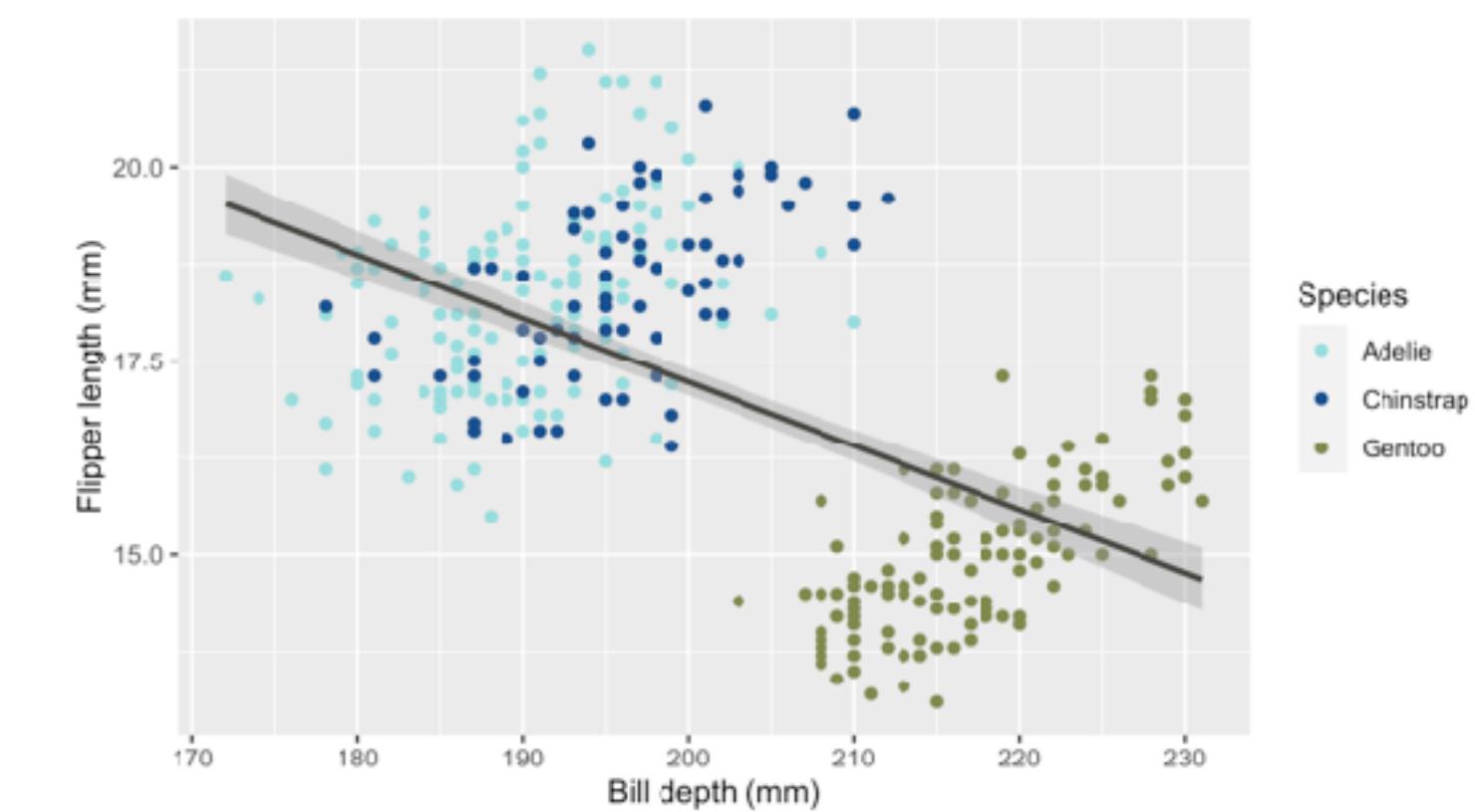
```
# visualize the relationship
ggplot(penguins) +
  geom_point(
    aes(x = bill_depth_mm, y = flipper_length_mm, color = species))
  +
  geom_smooth(
    aes(x = bill_depth_mm, y = flipper_length_mm),
    method = "lm")
```



**Table 1.** Regression output for predicting bill depth from flipper length.

| term              | estimate | std.error | statistic | p.value  |
|-------------------|----------|-----------|-----------|----------|
| (Intercept)       | 33.6     | 1.25      | 27.0      | 1.39e-86 |
| flipper_length_mm | -0.0820  | 0.00618   | -13.3     | 1.23e-32 |

**Figure 2.** Relationship between bill depth and flipper length.



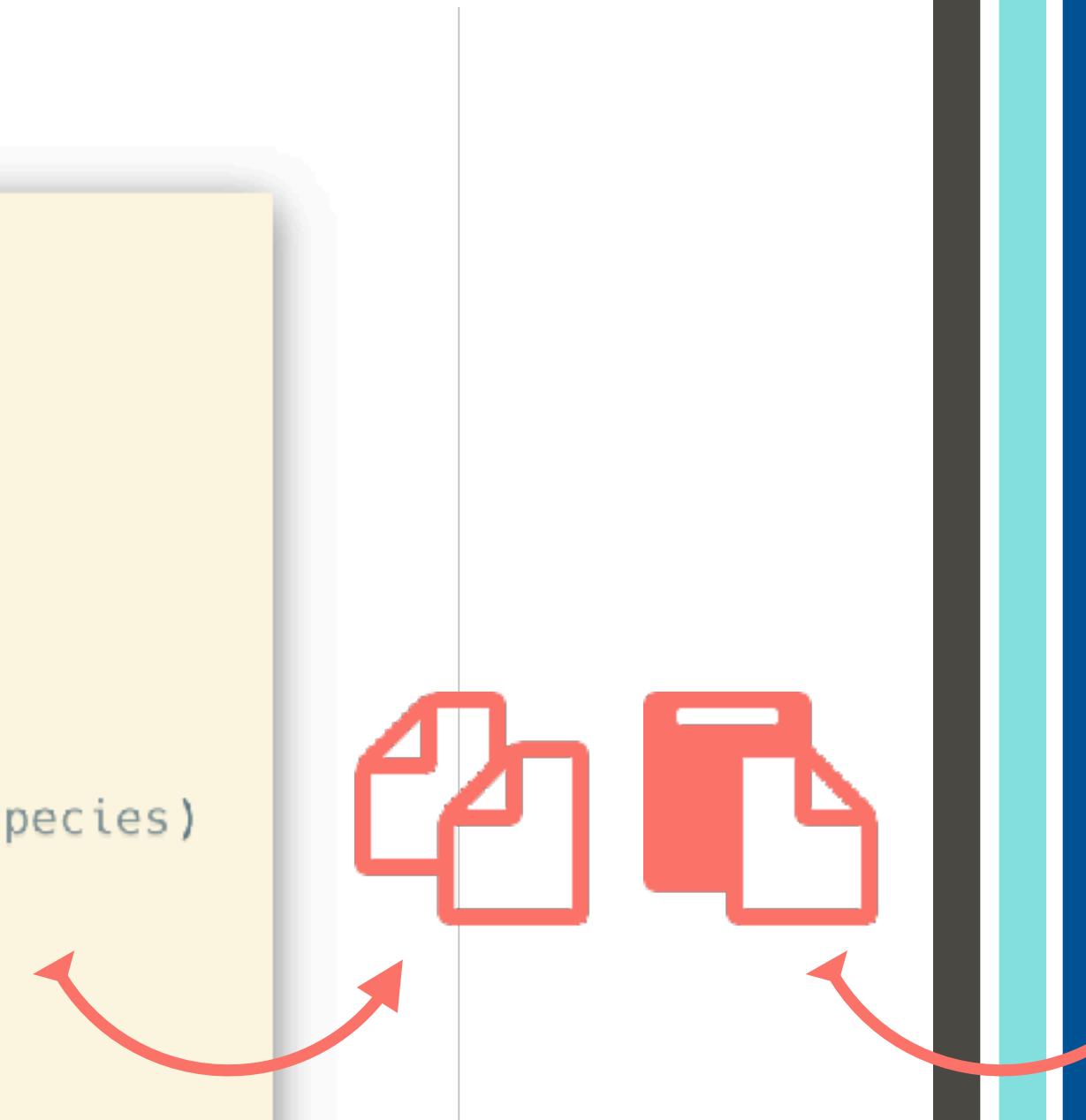
# analysis

# report



```
# filter out Gentoos
penguins_nongentoo <- penguins %>%
  filter(species != "Gentoo")

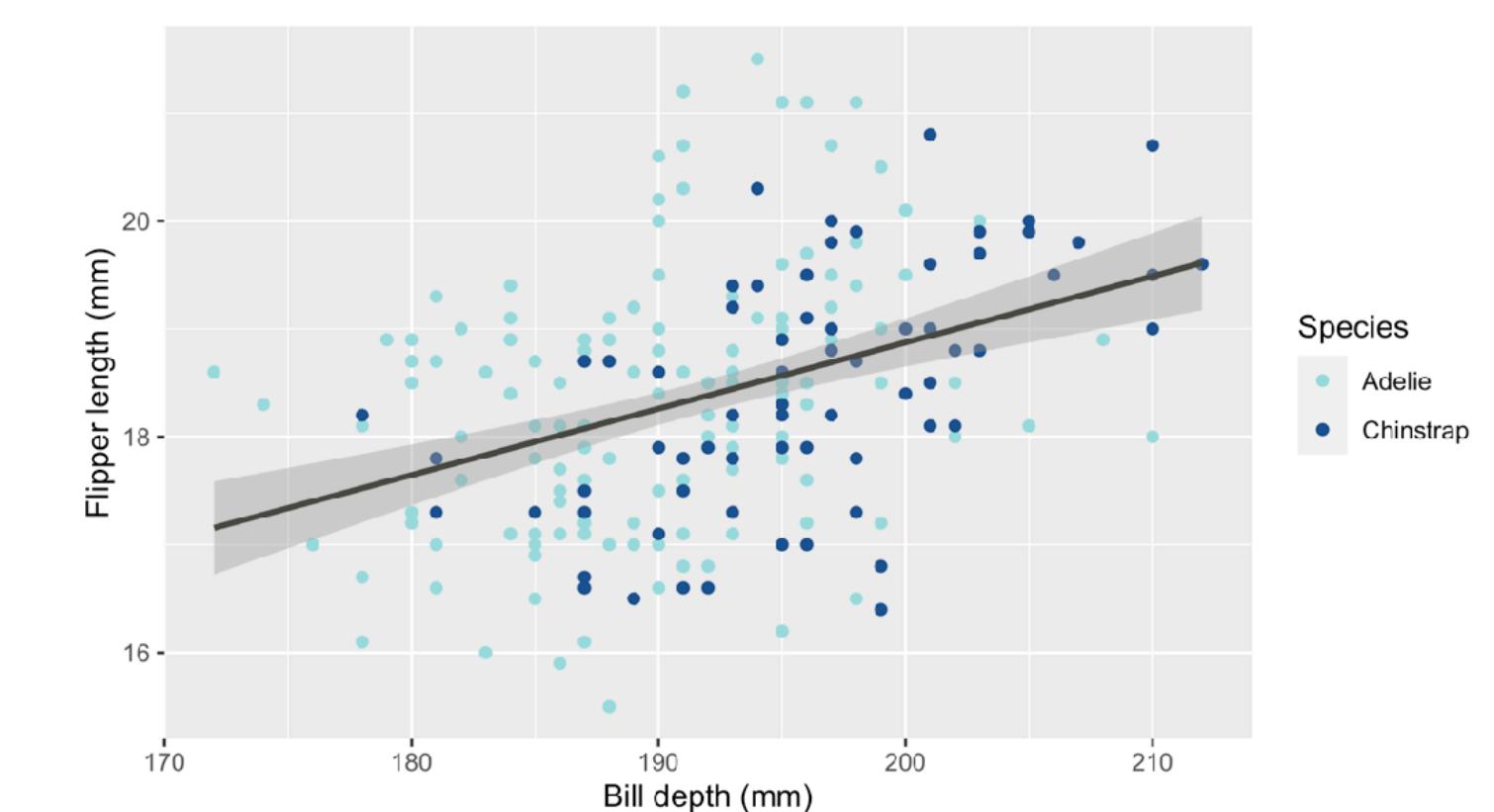
# visualize the relationship
ggplot(penguins_nongentoo) +
  geom_point(
    aes(x = bill_depth_mm, y = flipper_length_mm, color = species)
  ) +
  geom_smooth(
    aes(x = bill_depth_mm, y = flipper_length_mm),
    method = "lm"
  )
```



**Table 1.** Regression output for predicting bill depth from flipper length.

| term              | estimate | std.error | statistic | p.value  |
|-------------------|----------|-----------|-----------|----------|
| (Intercept)       | 33.6     | 1.25      | 27.0      | 1.39e-86 |
| flipper_length_mm | -0.0820  | 0.00618   | -13.3     | 1.23e-32 |

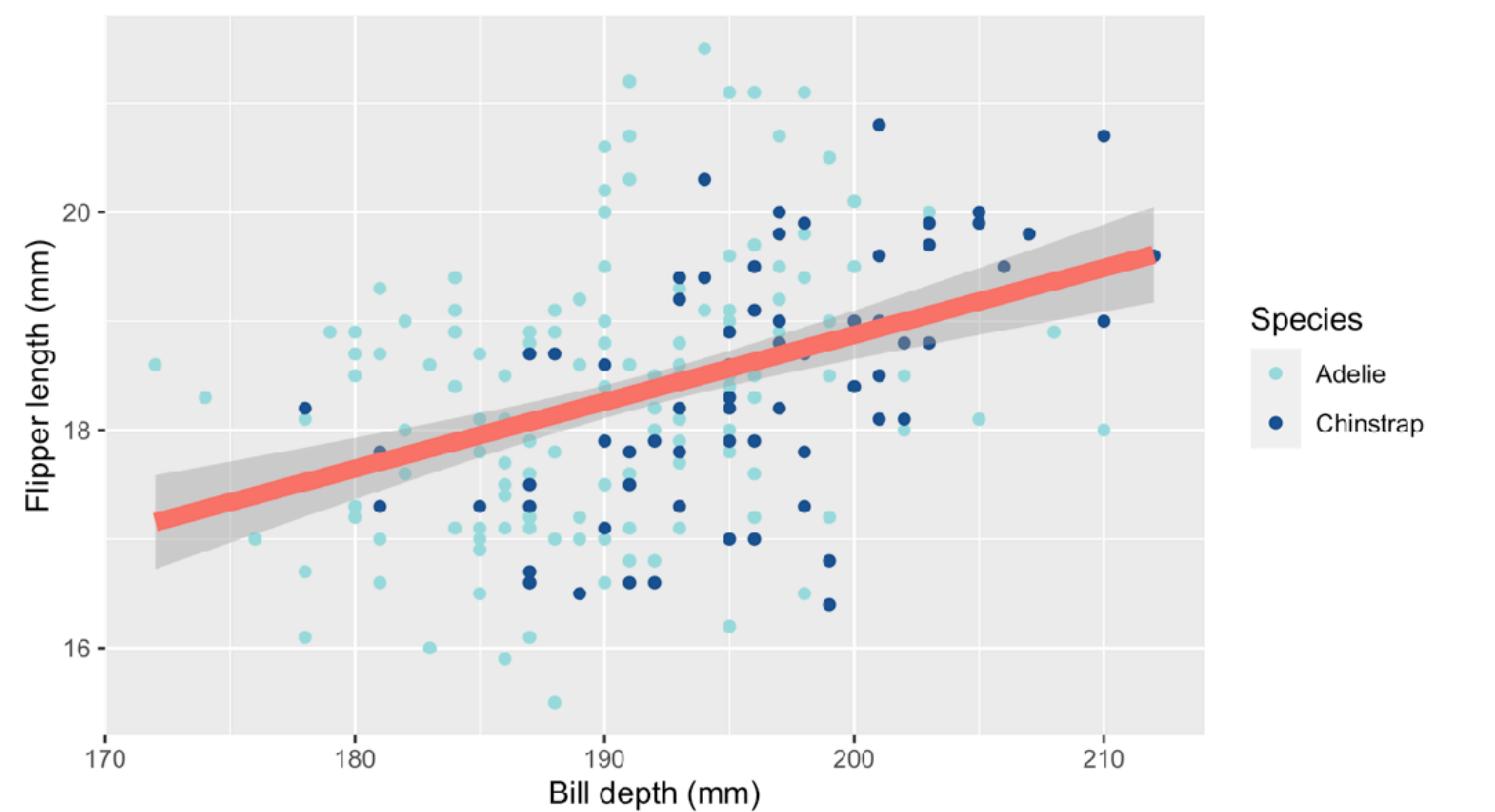
**Figure 2.** Relationship between bill depth and flipper length.

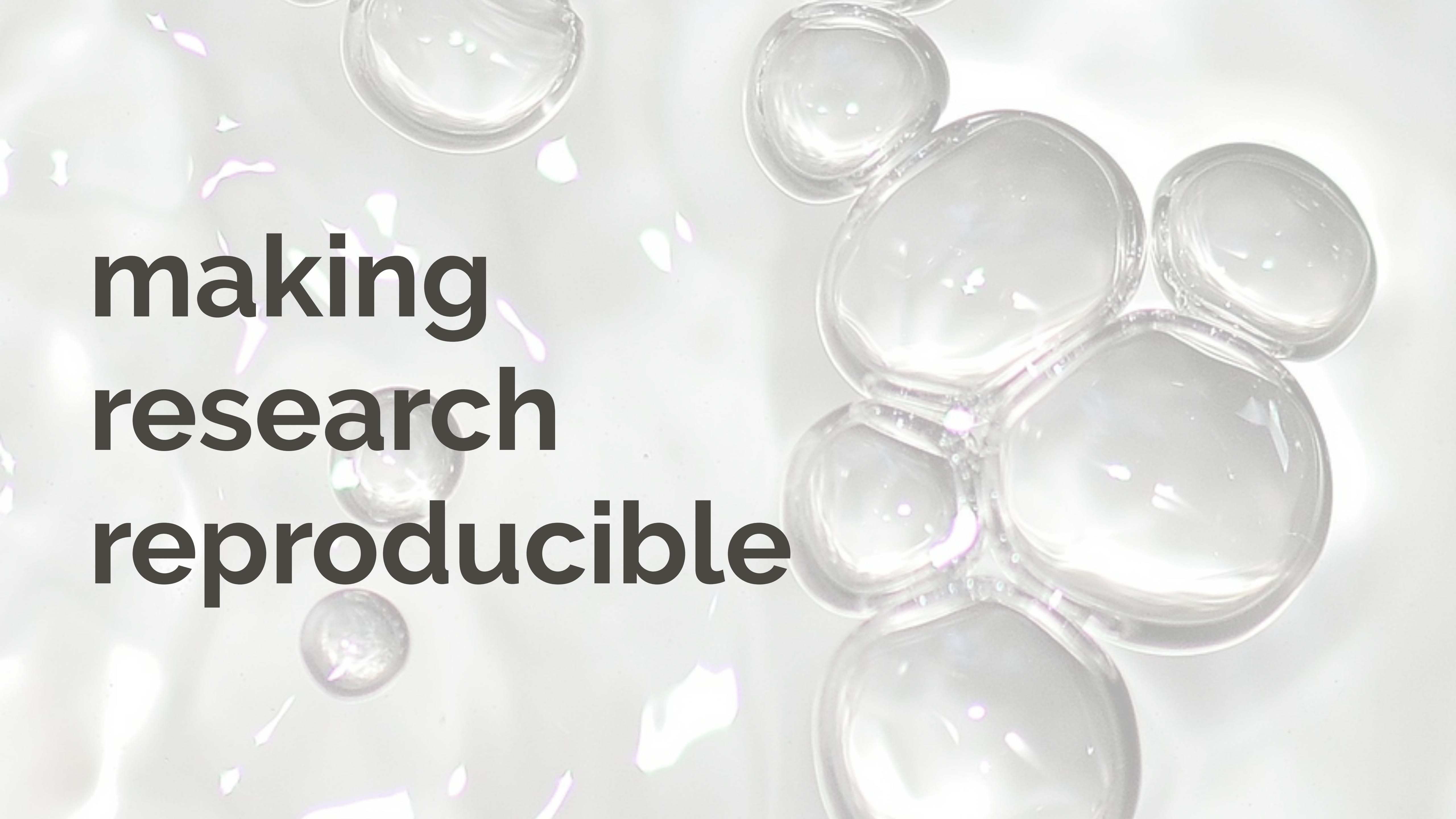


**Table 1.** Regression output for predicting bill depth from flipper length.

| term              | estimate       | std.error | statistic | p.value  |
|-------------------|----------------|-----------|-----------|----------|
| (Intercept)       | 33.6           | 1.25      | 27.0      | 1.39e-86 |
| flipper_length_mm | <b>-0.0820</b> | 0.00618   | -13.3     | 1.23e-32 |

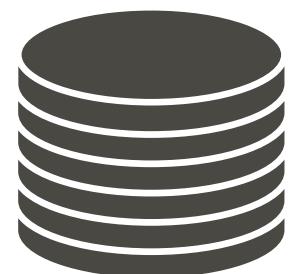
**Figure 2.** Relationship between bill depth and flipper length.





**making  
research  
reproducible**

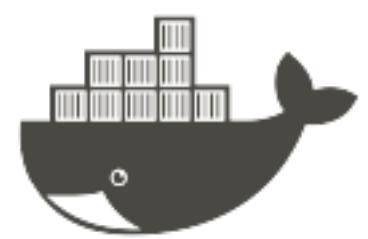
# make



raw data



code & documentation to reproduce the analysis



specifications of your computational environment

## available and accessible

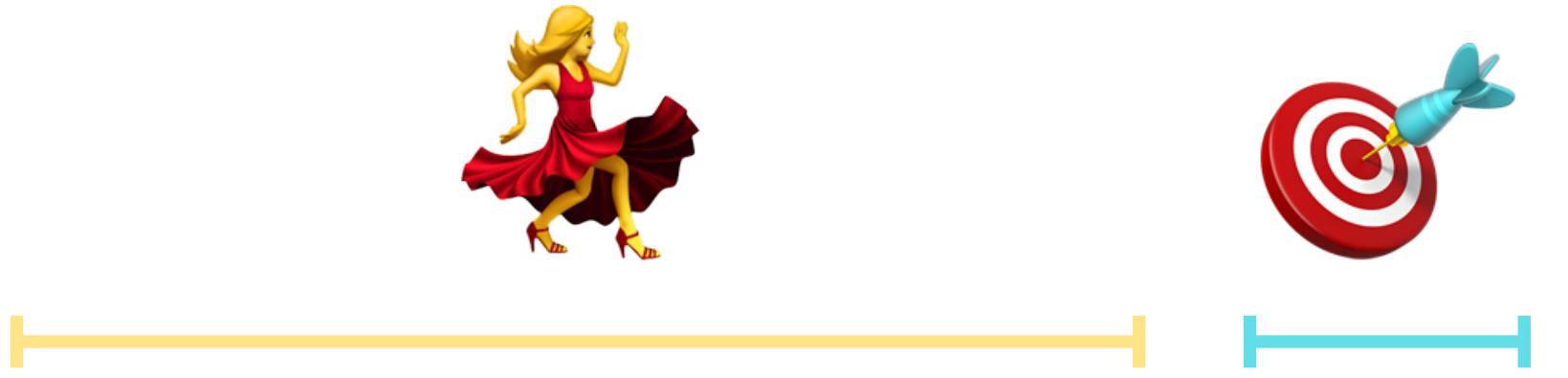
Peng, Roger. "The reproducibility crisis in science: A statistical counterattack." *Significance* 12.3 (2015): 30-32.

Gentleman, Robert, and Duncan Temple Lang. "Statistical analyses and reproducible research." *Journal of Computational and Graphical Statistics* 16.1 (2007): 1-23.

“The most important tool is  
the **mindset**, when starting,  
that the end product will be  
reproducible.”

– Keith Baggerly

nobody,  
not even yourself,  
can recreate any part  
of your analysis



push button  
reproducibility  
in published work

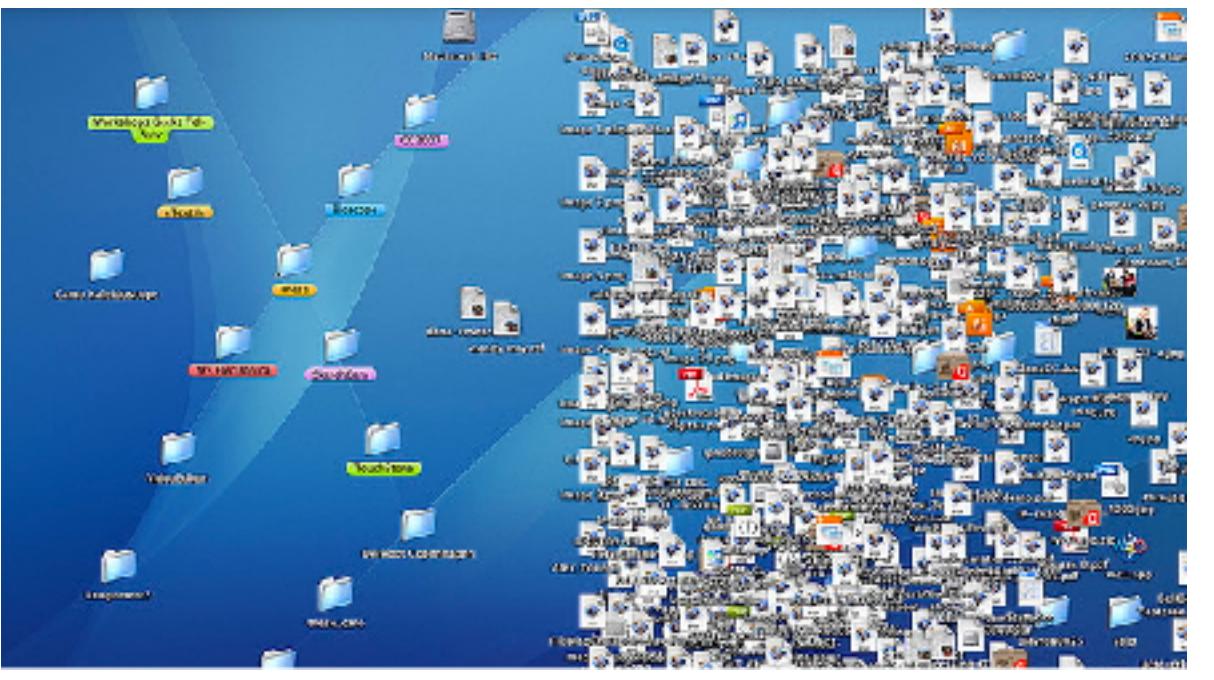
“There’s **no one-size-fits-all solution**  
for computational reproducibility.”

8 principles  
but the following<sup>^</sup> might help...

1

organize  
your  
project

level of organization



## simpler analysis

-  raw-data
-  processed-data
-  manuscript
  - └ manuscript.Rmd

## more complex analysis

-  raw-data
-  processed-data
-  scripts
-  figures
-  manuscript
  - └ manuscript.Rmd

stick with the conventions of your peers



write  
**READMEs**  
liberally



## raw-data

- └ README.md
- └ airlines.csv
- └ airports.csv
- └ flights.csv
- └ planes.csv
- └ weather.csv



## processed-data



## scripts



## figures



## manuscript



### # README

This folder contains the raw data for the project.

All datasets were downloaded from [openflights.org/data.html](http://openflights.org/data.html) on 2019-04-01.

- airlines: Airline names
- airports: Airports metadata
- flights: Flight data
- planes: Plane metadata
- weather: Hourly weather data



keep data  
tidy &  
machine readable

| Student          | Exam Grade  |      |                                  |
|------------------|-------------|------|----------------------------------|
| Name             | 1           | 2    | Major                            |
| Barney Donaldson | 89          | 76   | Data Science,<br>Public Policy   |
| Clay Whelan      | 67          | 83   | Public Policy                    |
| Simran Bass      | 82          | 90   | Statistics                       |
| Chante Munro     | 45          | 72   | Political Science,<br>Statistics |
| Gabrielle Cherry | 32          | 79   | .                                |
| Kush Piper       | 98          | sick | Statistics                       |
| Faizan Ratliff   | 82          | 75   | Data Science                     |
| Torin Ruiz       | 70          | 80   | Sociology,<br>Statistics         |
| Reiss Richardson | missed exam | 34   | Neuroscience                     |
| Ajwa Cochran     | 50          | 65   | Data Science                     |

→ record  
code +  
document  
non-code  
steps +  
write  
tests

| name             | exam_1 | exam_2 | first_major       | second_major  | participation |
|------------------|--------|--------|-------------------|---------------|---------------|
| Barney Donaldson | 89     | 76     | Data Science      | Public Policy | ok            |
| Clay Whelan      | 67     | 83     | Public Policy     | NA            | ok            |
| Simran Bass      | 82     | 90     | Statistics        | NA            | ok            |
| Chante Munro     | 45     | 72     | Political Science | Statistics    | Low           |
| Gabrielle Cherry | 32     | 79     | NA                | NA            | ok            |
| Kush Piper       | 98     | NA     | Statistics        | NA            | ok            |
| Faizan Ratliff   | 82     | 75     | Data Science      | NA            | ok            |
| Torin Ruiz       | 70     | 80     | Sociology         | Statistics    | ok            |
| Reiss Richardson | NA     | 34     | Neuroscience      | NA            | low           |
| Ajwa Cochran     | 50     | 65     | Data Science      | NA            | low           |

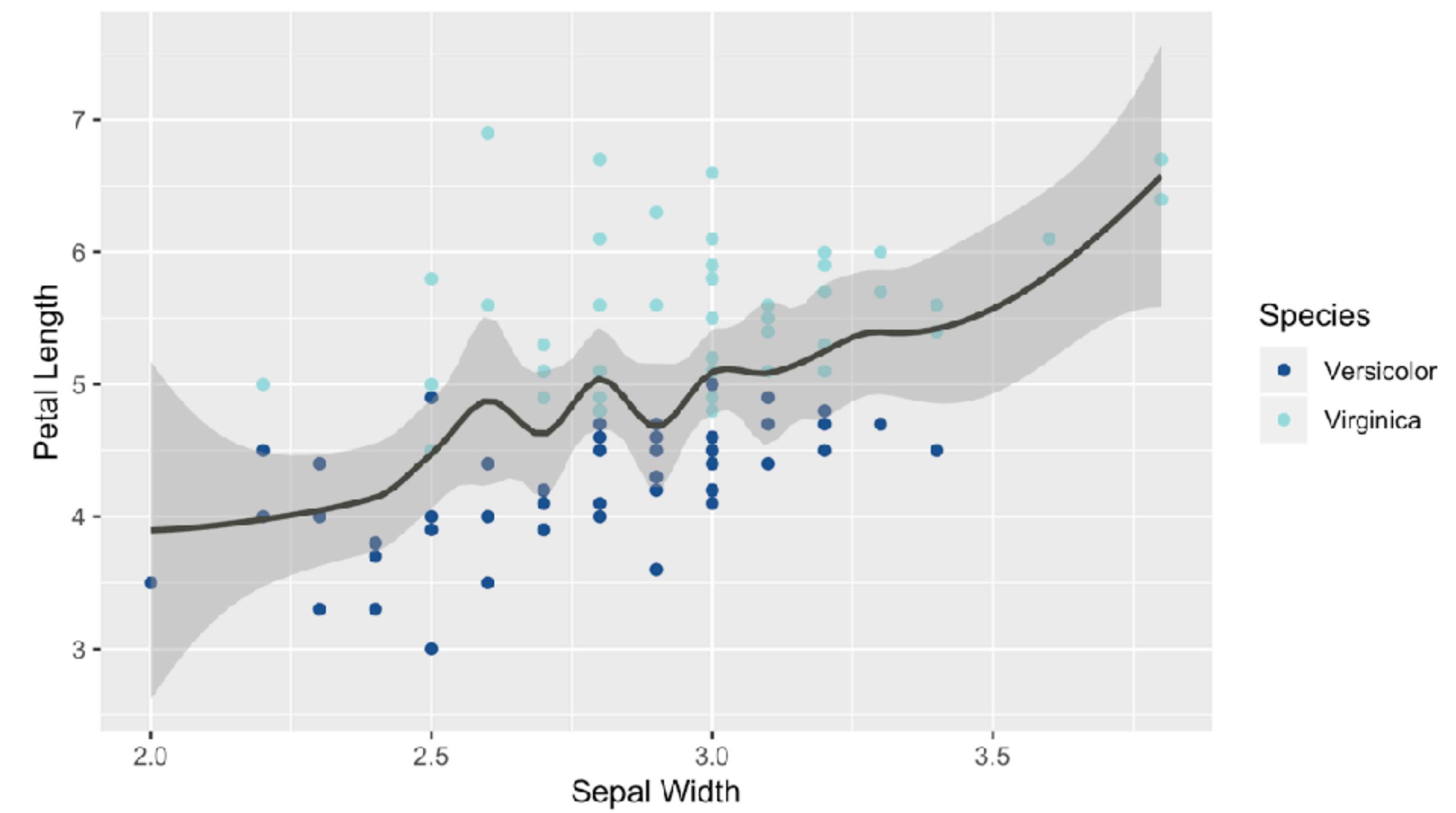
Low participation

1

comment  
your  
code



```
# use loess smoothing
ggplot(penguins_nongentoo) +
  geom_point(
    aes(x = flipper_length_mm, y = bill_depth_mm, color = species))
  ) +
  geom_smooth(
    aes(x = flipper_length_mm, y = bill_depth_mm),
    method = "loess", span = 0.375
  )
```





use  
literate  
programming

tab1-fig2.Rmd

Normal | B I U </> | X | : 1= , | @ | Format | Insert | Table

```
---
```

```
title: "Table 1 matches Figure 2!"  
author: "Mine Çetinkaya-Rundel"  
date: "`r Sys.Date()`"  
output:  
  bookdown::html_document2:  
    fig_caption: yes  
bibliography: references.bib  
---
```

```
{r chunk-options, include=FALSE}  
knitr::opts_chunk$set(echo = FALSE, message = FALSE)
```

```
{r setup}  
library(broom)          # for tidy model output  
library(glue)           # for gluing strings to data  
library(knitr)           # for kable  
library(palmerpenguins)  # for data  
library(tidyverse)        # for data wrangling and visualisation
```

In this report we evaluate the relationship between relationship between bill depth and flipper length of penguins. The data come from [@gorman2014](#).

The diagram shows a side view of a penguin's head. A horizontal dashed line extends from the tip of the beak to the back of the head. A vertical dashed line extends from the base of the beak down to the chin. A double-headed arrow along the horizontal line is labeled "Bill length". A double-headed arrow along the vertical line is labeled "Bill depth".

# Exploratory data analysis | R Markdown

improve-repro-workflow-reproducibilitea-2020 - RStudio

tab1-fig2.Rmd x Addins

Environment History Connections Tutorial

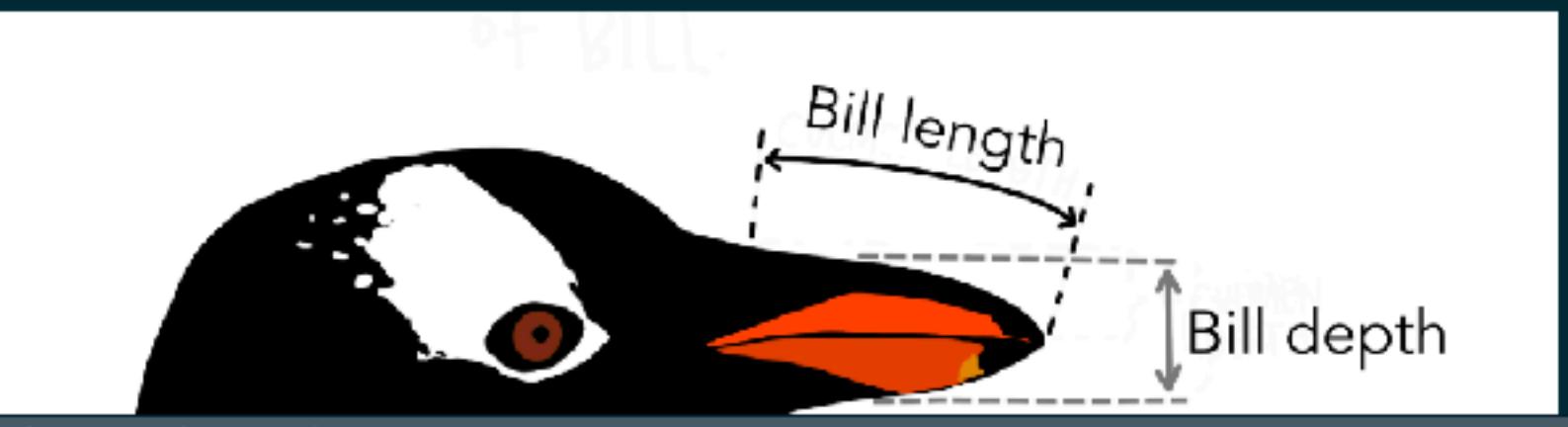
Files Plots Packages Help Viewer

Publish

Normal B I U Insert Table

```
---  
title: "Table 1 matches Figure 2!"  
author: "Mine Çetinkaya-Rundel"  
date: "r Sys.Date()"  
output:  
  bookdown::html_document2:  
    fig_caption: yes  
bibliography: references.bib  
---  
  
{r chunk-options, include=FALSE}  
knitr::opts_chunk$set(echo = FALSE, message = FALSE)  
  
{r setup}  
library(broom)      # for tidy model output  
library(glue)        # for gluing strings to data  
library(knitr)       # for kable  
library(palmerpenguins) # for data  
library(tidyverse)    # for data wrangling and visualisation
```

In this report we evaluate the relationship between bill depth and flipper length of penguins. The data come from @gorman2014.



# Exploratory data analysis R Markdown

Console

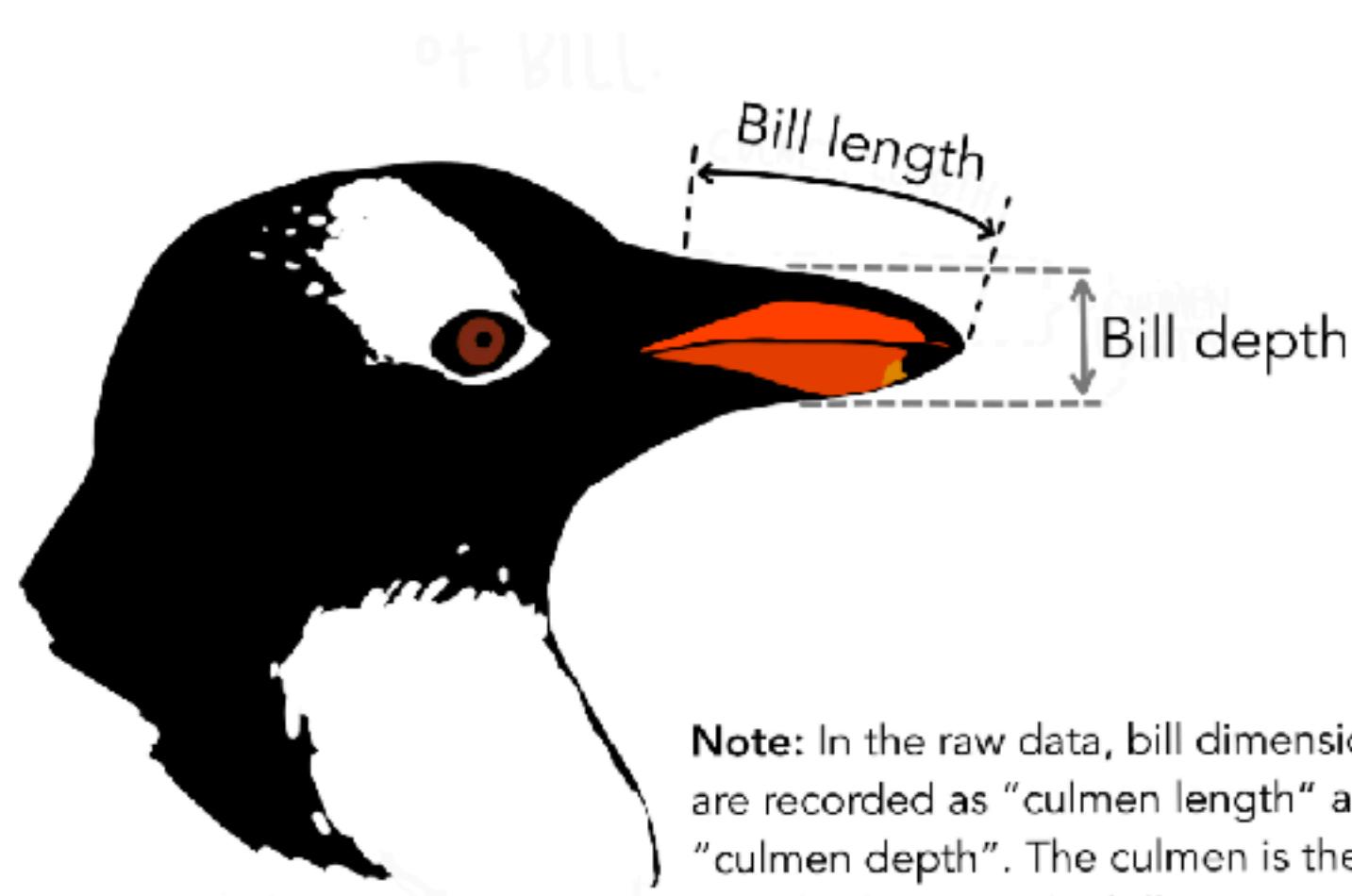
Table 1.1 shows some summary statistics.

Table 1.1: Summary statistics

# Table 1 matches Figure 2!

Mine Çetinkaya-Rundel  
2020-11-10

In this report we evaluate the relationship between bill depth and flipper length of penguins. The data come from Gorman, Williams, and Fraser (2014).



Note: In the raw data, bill dimensions are recorded as "culmen length" and "culmen depth". The culmen is the dorsal ridge atop the bill.

The original dataset has 3 species (Adelie, Gentoo, and Chinstrap), but we will only work with Adelie and Chinstrap species.

## 1 Exploratory data analysis

*demo*

**rmarkdown**

# more resources . . .

- ▶ Learn more about **R Markdown**:
  - ▶ Documentation: [rmarkdown.rstudio.com](https://rmarkdown.rstudio.com)
  - ▶ Book: [bookdown.org/yihui/rmarkdown](https://bookdown.org/yihui/rmarkdown)
  - ▶ Book: [bookdown.org/yihui/rmarkdown-cookbook](https://bookdown.org/yihui/rmarkdown-cookbook)
- ▶ Learn more about the **visual editor**:
  - ▶ Documentation: [rstudio.github.io/visual-markdown-editing](https://rstudio.github.io/visual-markdown-editing)
  - ▶ Blog post: [blog.rstudio.com/2020/09/30/rstudio-v1-4-preview-visual-markdown-editing](https://blog.rstudio.com/2020/09/30/rstudio-v1-4-preview-visual-markdown-editing)
  - ▶ Blog post: [blog.rstudio.com/2020/11/09/rstudio-1-4-preview-citations](https://blog.rstudio.com/2020/11/09/rstudio-1-4-preview-citations)



use  
version  
control

The screenshot shows the RStudio interface with an R Markdown document open. The code chunk contains YAML front matter and R code for setting up packages and options. A figure of a penguin with measurement annotations is included.

```
---
```

```
title: "Table 1 matches Figure 2!"  
author: "Mine Çetinkaya-Rundel"  
date: "^r Sys.Date()"  
output:  
  bookdown::html_document2:  
    fig_caption: yes  
bibliography: references.bib
```

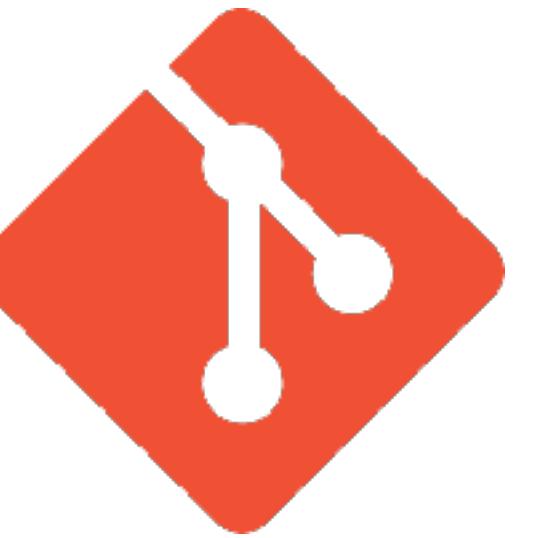
```
{r chunk-options, include=FALSE}  
knitr::opts_chunk$set(echo = FALSE, message = FALSE)
```

```
{r setup}  
library(broom)      # for tidy model output  
library(glue)       # for gluing strings to data  
library(knitr)      # for kable  
library(palmerpenguins) # for data  
library(tidyverse)   # for data wrangling and visualisation
```

In this report we evaluate the relationship between relationship between bill depth and flipper length of penguins. The data come from [@gorman2014](#).

A small image of a penguin's head and upper body. Two dashed arrows point from text labels to specific parts of the penguin's beak: one arrow points horizontally to the right and is labeled 'Bill length', and another arrow points vertically downwards and is labeled 'Bill depth'.

changes  
tracked by



hosted  
on

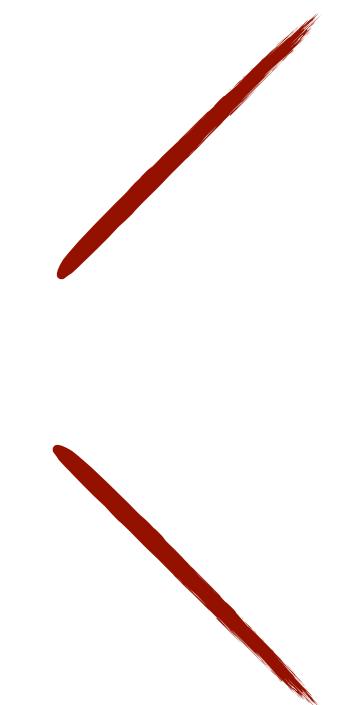


2

# Git workflows

GitHub first

Local first

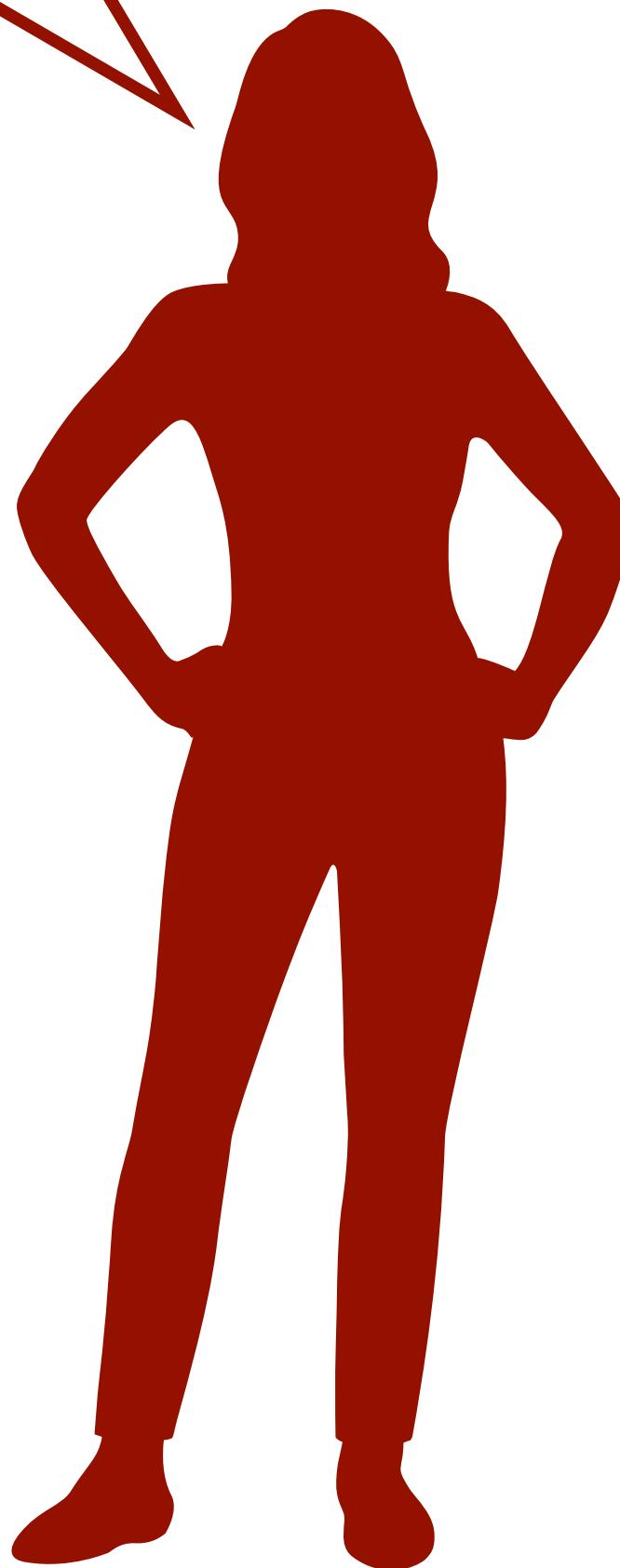


# GitHub first



- ▶ Step 1: Create a new repo on GitHub
- ▶ Step 2: Copy the repo URL
- ▶ Step 3: Clone it using RStudio
- ▶ Step 4: Make changes locally
- ▶ Step 6: Commit and push to GitHub
- ▶ Step 7: Confirm your changes have propagated to GitHub

# Local first

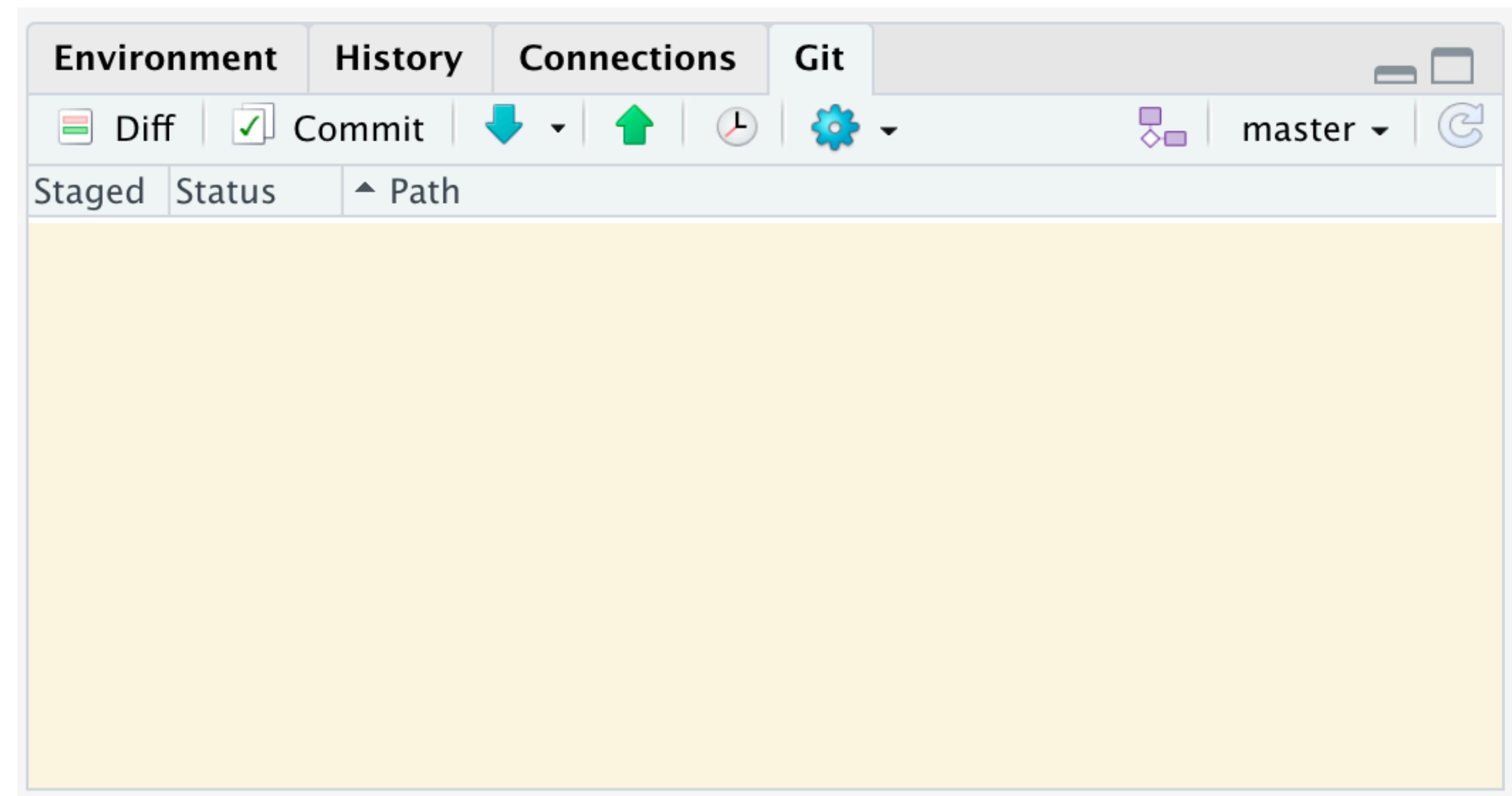


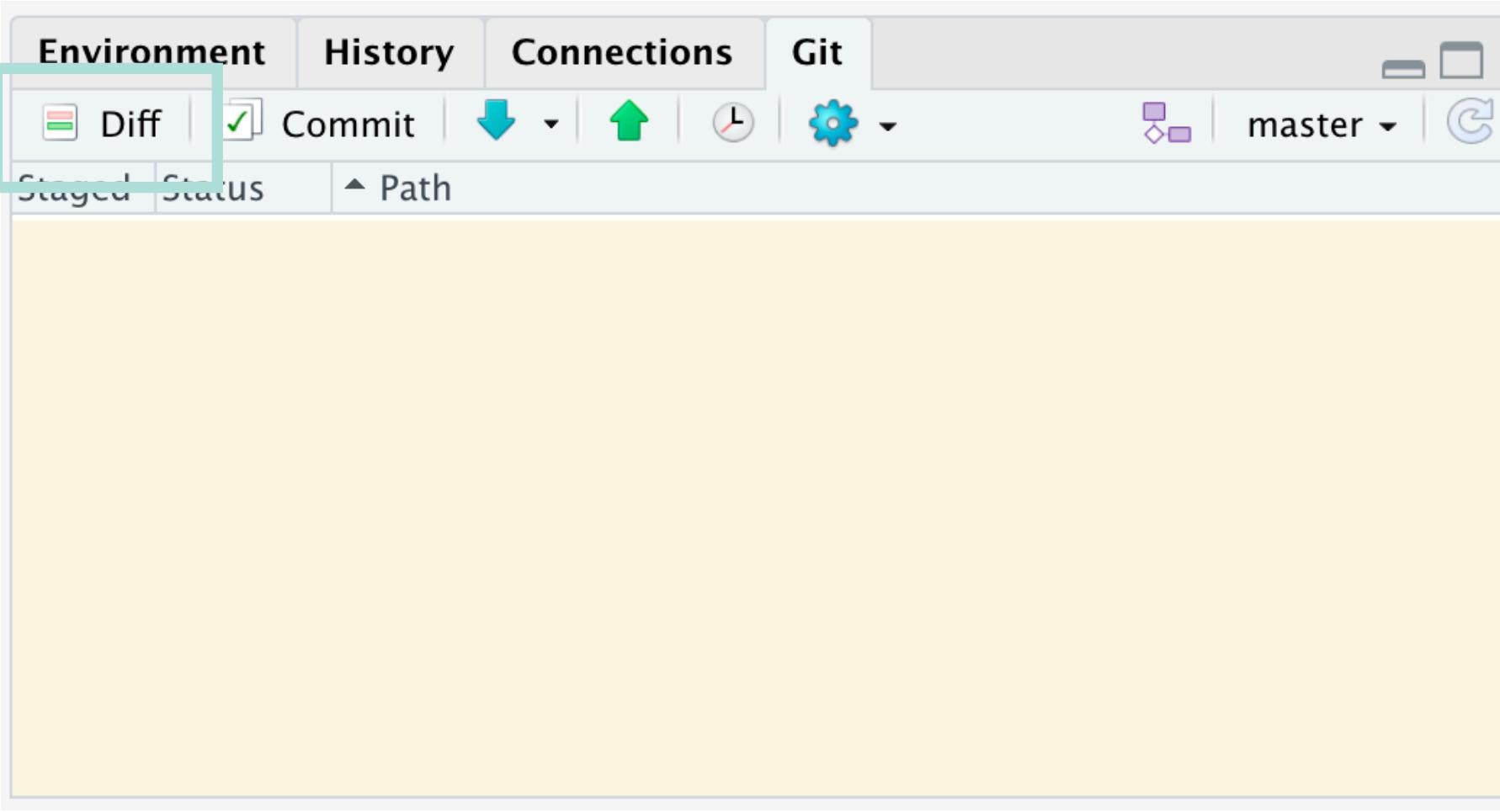
I have been working on  
a project for a while, and now  
I'm realising I should have  
been tracking it with git.

- ▶ Step 1: Create an RStudio Project from existing directory (if an `.Rproj` file doesn't already exist)
- ▶ Step 2: `usethis::use_git()` and follow instructions
- ▶ Step 3: `usethis::use_github()` and follow instructions

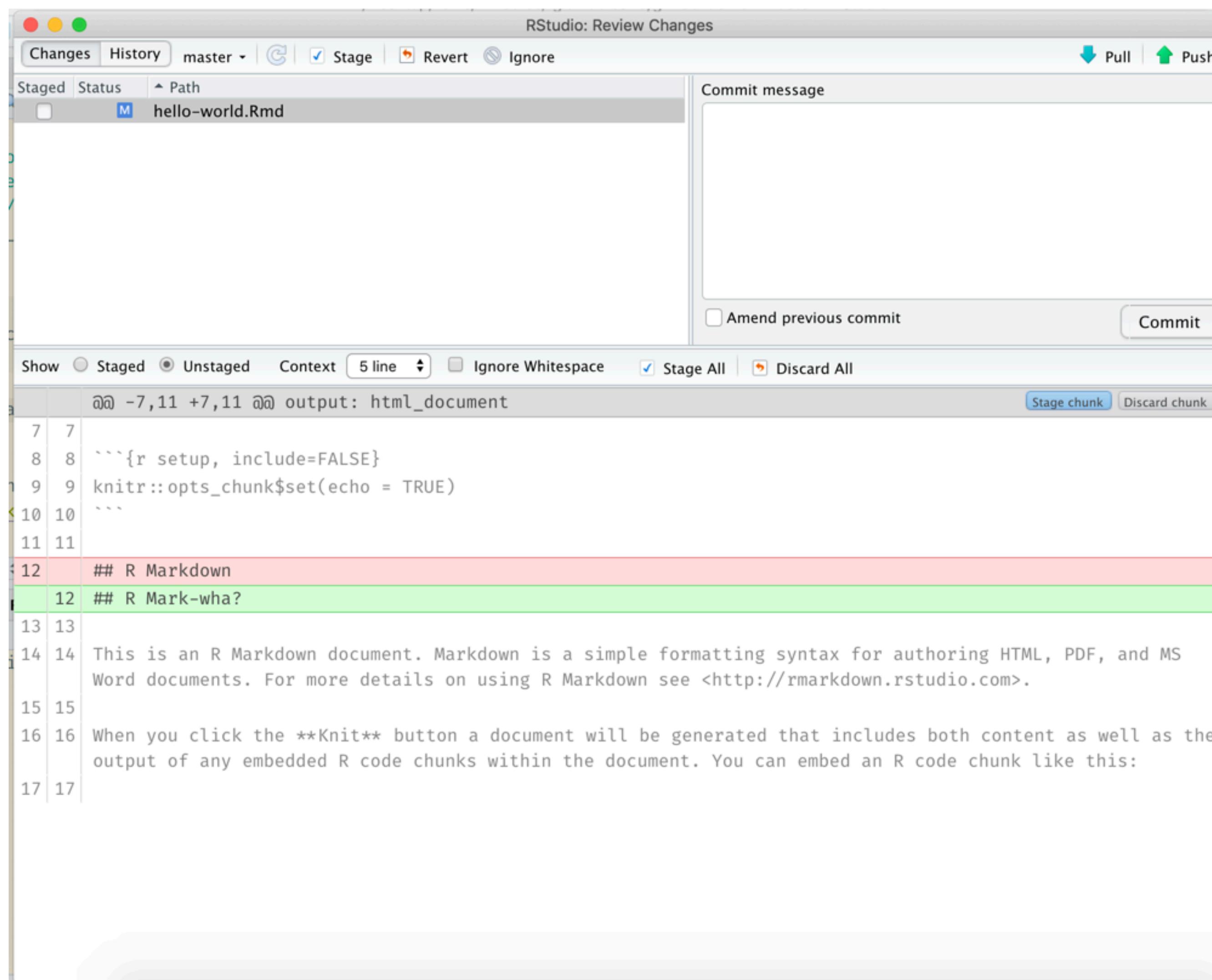
*demo*

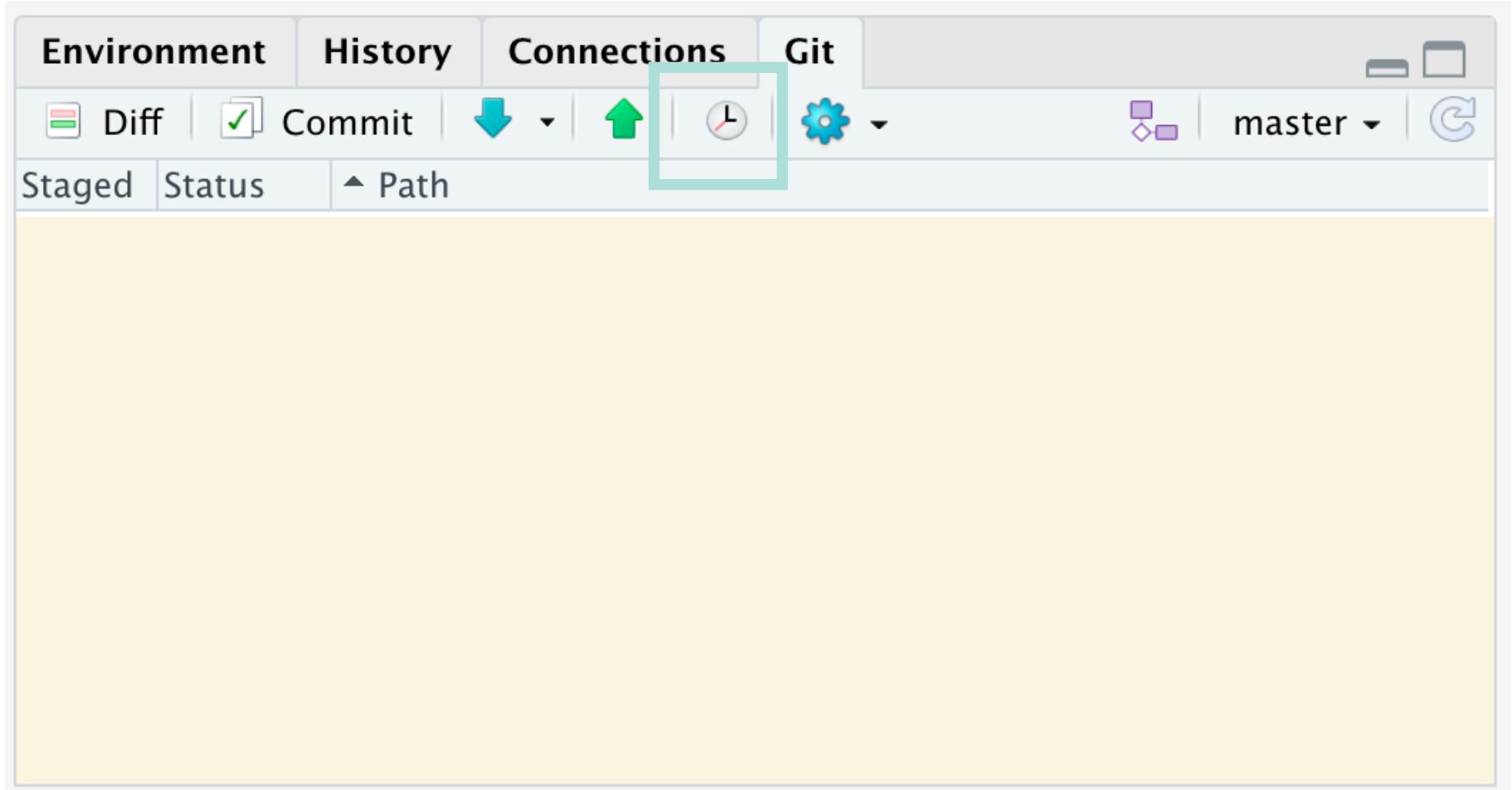
**git & github**





- ▶ View options
- ▶ Staging and committing all changes in a document at once
- ▶ Staging and committing various changes within a document one by one
- ▶ Commit messages
- ▶ Amending a previous commit
- ▶ Pushing





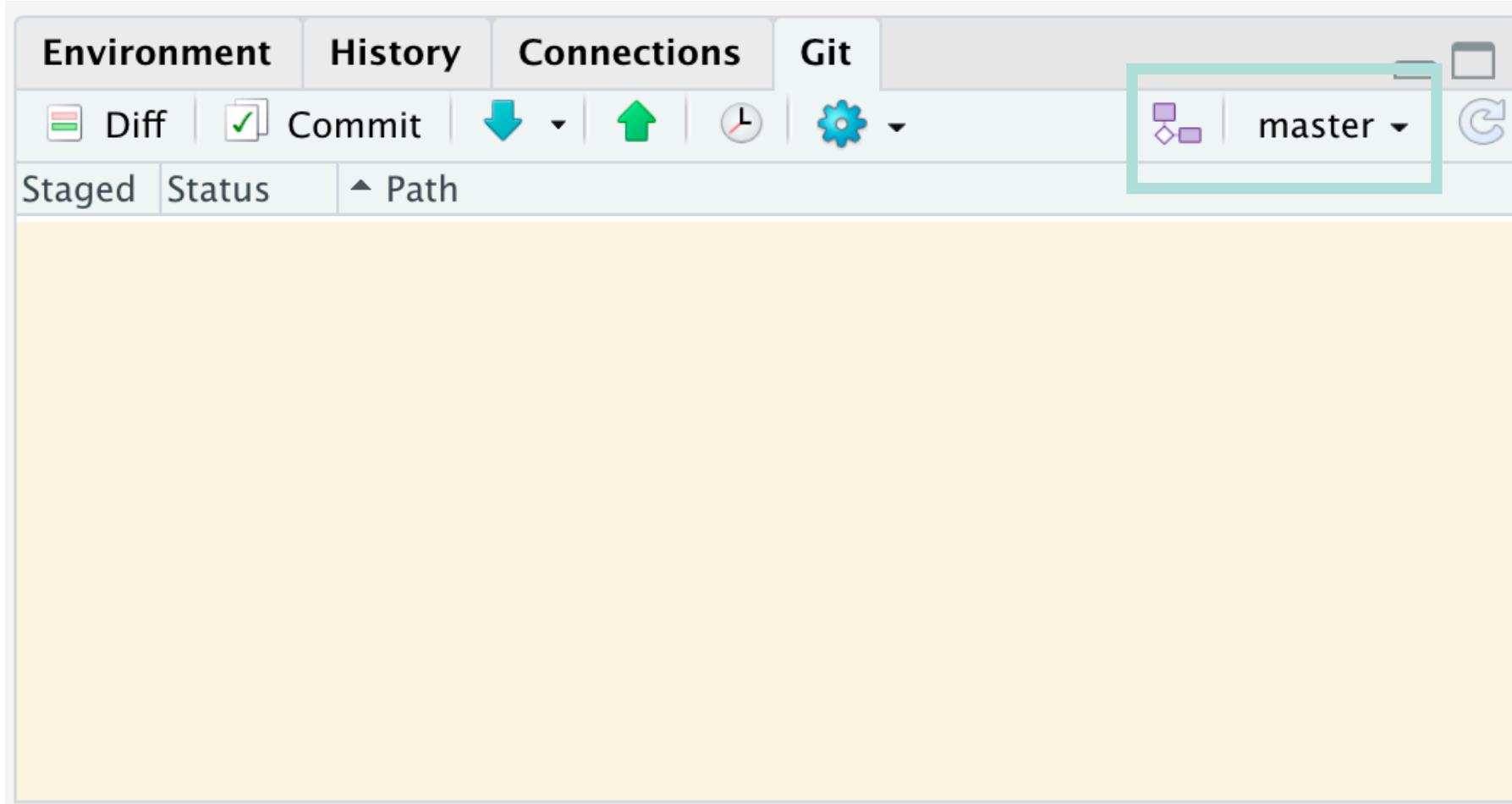
- ▶ History of commits
- ▶ What is HEAD?
- ▶ Filtering history of commits by File or Directory

RStudio: Review Changes

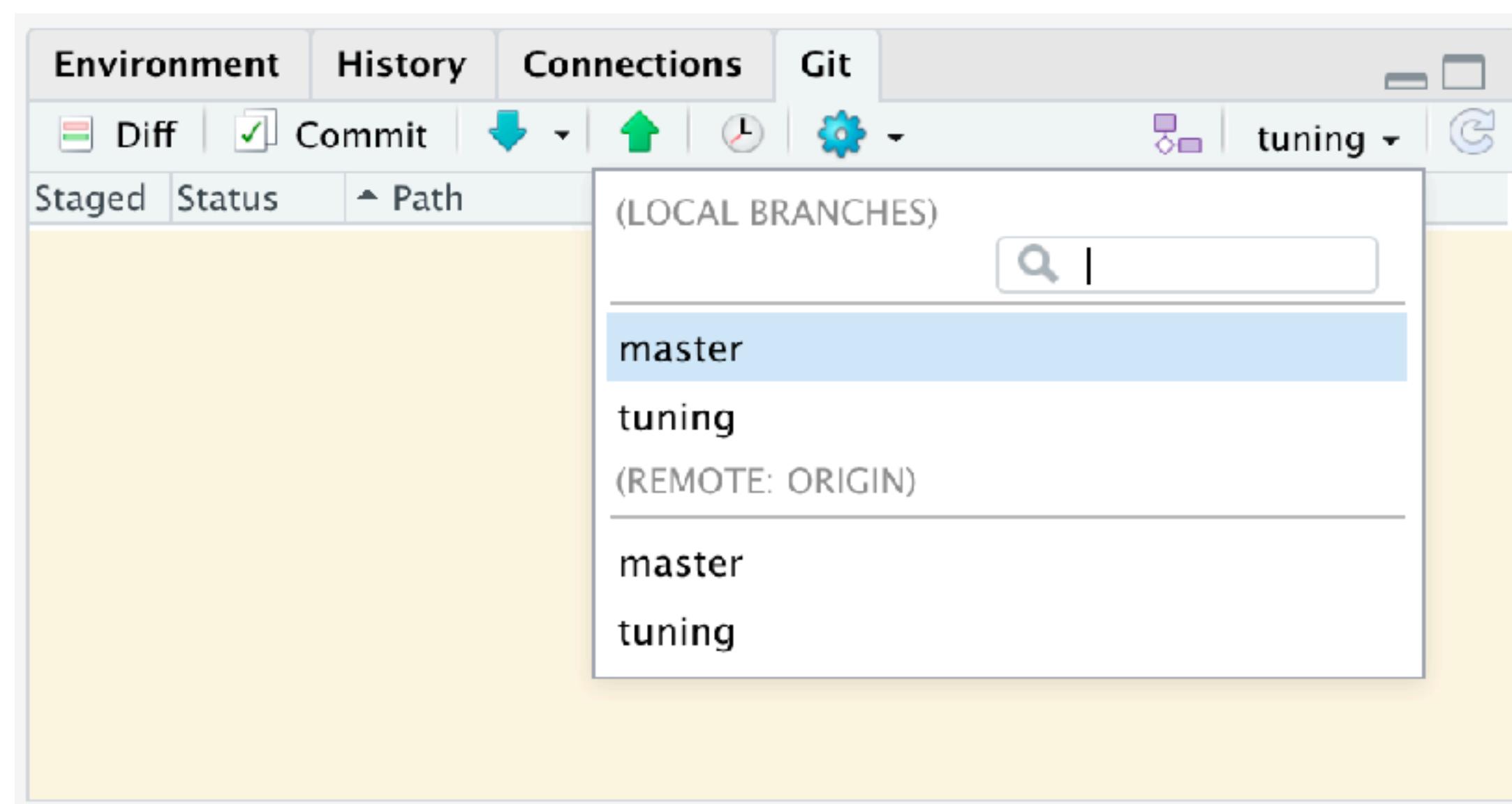
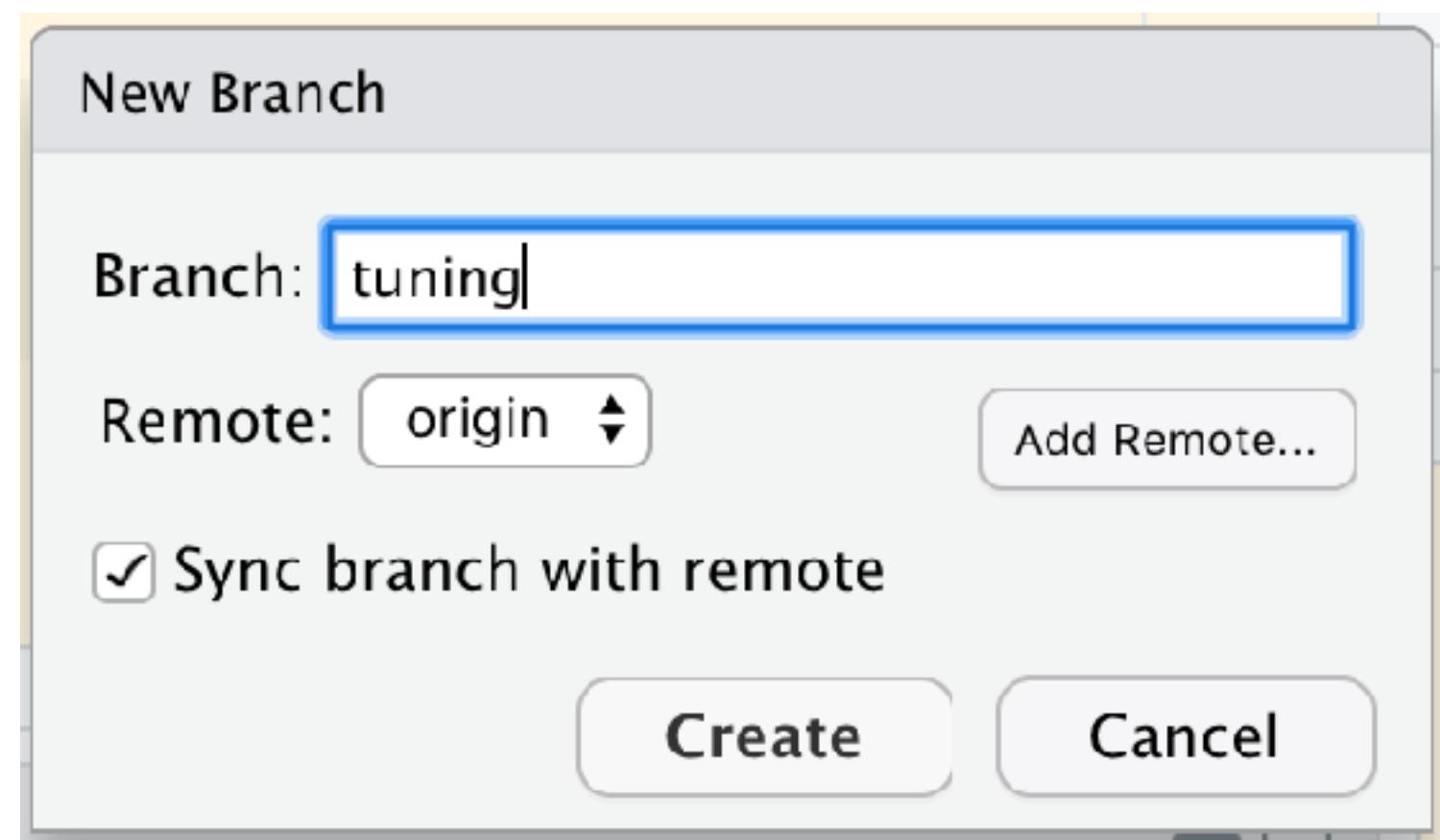
| Subject                    | Author                               | Date       | SHA      |
|----------------------------|--------------------------------------|------------|----------|
| HEAD -> refs/heads/master  | Mine Cetinkaya-Rundel <mine@stat.d   | 2018-12-18 | 07e3c74b |
| origin/master              |                                      |            |          |
| origin/HEAD                |                                      |            |          |
| huhu                       | Mine Cetinkaya-Rundel <mine@stat.d   | 2018-12-18 | 800655bf |
| hu                         | Mine Cetinkaya-Rundel <mine@stat.d   | 2018-12-18 | 70b51473 |
| Start data analysis report | Mine Cetinkaya-Rundel <mine@stat.d   | 2018-12-17 | b8f352fe |
| A new proj is born! 🎉      | Mine Cetinkaya-Rundel <mine@stat.d   | 2018-12-17 | 6ef76e86 |
| Initial commit             | Mine's testing agent <rundel.mine@gr | 2018-12-17 | 0c550d4f |

Changes History master - (all commits) Search Pull

Commits 1-6 of 6



- ▶ Branching
- ▶ Switching between branches



demo

pull requests

*more resources . . .*

- ▶ Learn more about using **Git and GitHub with R**:
  - ▶ Book: [happygitwithr.com](http://happygitwithr.com)
- ▶ Learn more about **Git setup**:
  - ▶ Documentation: [usethis.r-lib.org/articles/articles/usethis-setup.html](https://usethis.r-lib.org/articles/articles/usethis-setup.html)

?

automate  
your  
process



raw-data



processed-data



scripts

- └ 00-analyse.R
- └ 01-load-packages.R
- └ 02-load-data.R
- └ 03-clean-data.R
- └ 04-explore.R
- └ 05-model.R
- └ 06-summarise.R



```
00-analyse.R x
Source on Save | Run | Source | R Script

1 # run all -----
2
3 source("01-load-packages.R")
4 source("02-load-data.R")
5 source("03-clean-data.R")
6 source("04-explore.R")
7 source("05-model.R")
8 source("06-summarise.R")

1:1 # run all
```



figures



manuscript

# minimal make

A minimal tutorial on make

---

I would argue that the most important tool for reproducible research is not [Sweave](#) or [knitr](#) but [GNU make](#).

Consider, for example, all of the files associated with a manuscript. In the simplest case, I would have an [R](#) script for each figure plus a [LaTeX](#) file for the main text. And then a [BibTeX](#) file for the references.

Compiling the final PDF is a bit of work:

- Run each R script through R to produce the relevant figure.
- Run latex and then bibtex and then latex a couple of more times.

And the R scripts need to be run before latex is, and only if they've changed.

## A simple example

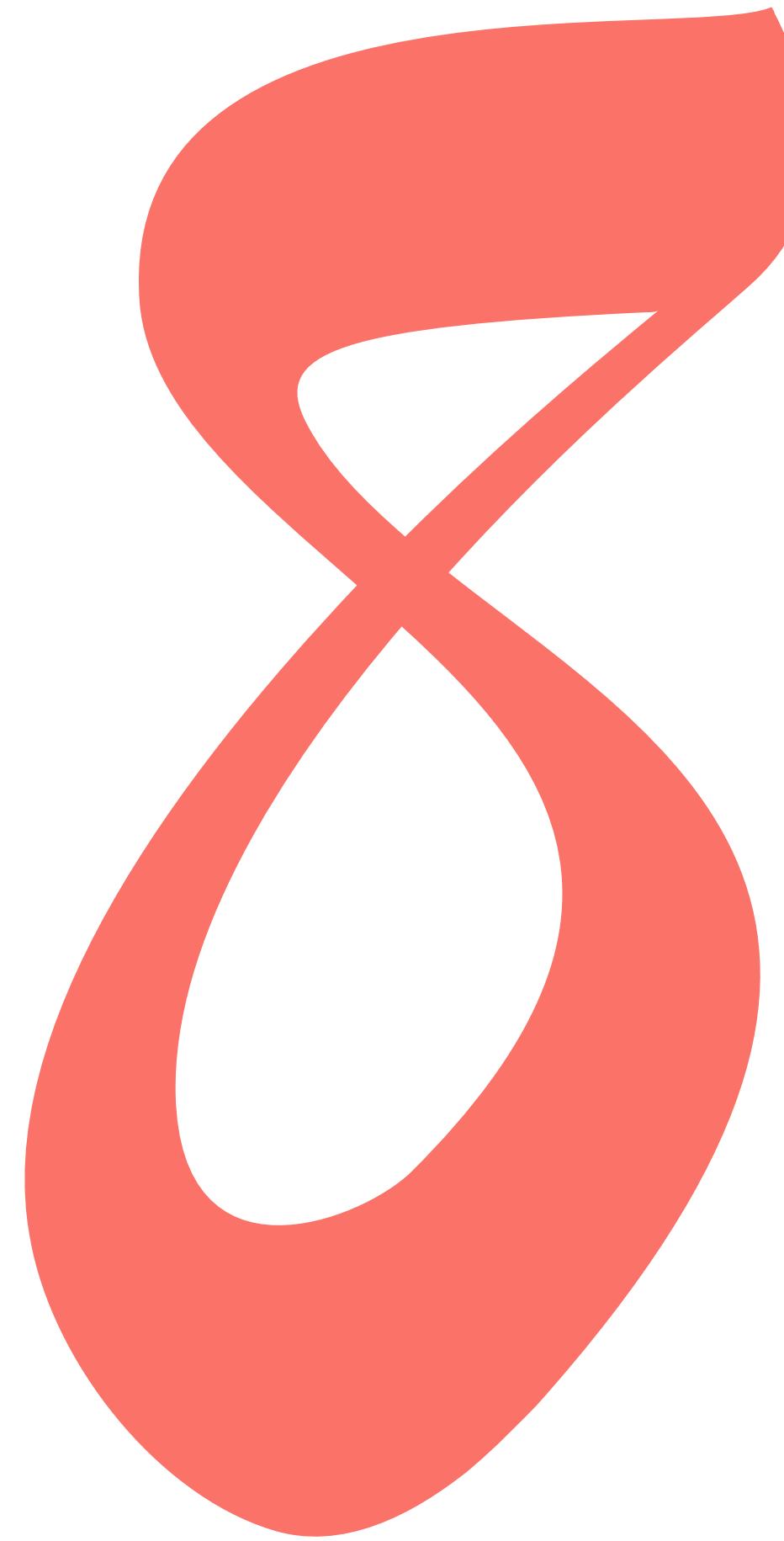
[GNU make](#) makes this easy. In your directory for the manuscript, you create a text file called `Makefile` that looks something like [the following](#) (here using [pdflatex](#)).

```
mypaper.pdf: mypaper.bib mypaper.tex Figs/fig1.pdf Figs/fig2.pdf
    pdflatex mypaper
    bibtex mypaper
    pdflatex mypaper
    pdflatex mypaper

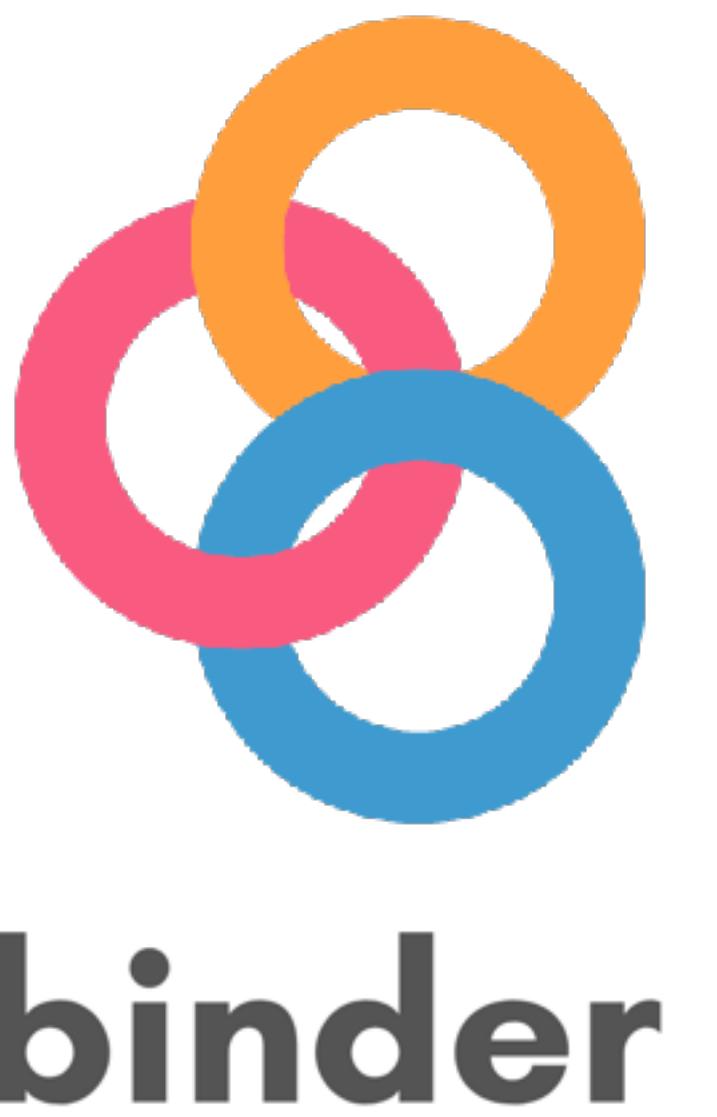
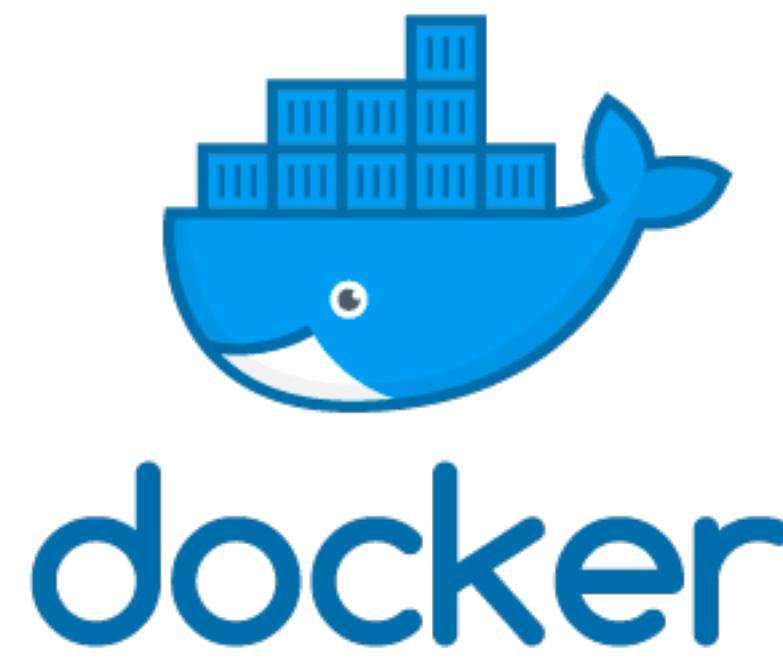
Figs/fig1.pdf: R/fig1.R
    cd R;R CMD BATCH fig1.R

Figs/fig2.pdf: R/fig2.R
    cd R;R CMD BATCH fig2.R
```

Each batch of lines indicates a file to be created (the *target*), the files it depends on (the *prerequisites*), and then a set of commands needed to construct the target from the dependent files. Note that the lines with the commands *must* start with a **tab** character (**not spaces**).



share  
computing  
environment



- 1 organize your project
- 2 write **READMEs** liberally
- 3 keep data **tidy & machine readable**
- 4 comment your code
- 5 use **literate programming**
- 6 use **version control**
- 7 automate your process
- 8 share computing **environment**

## PERSPECTIVE

## Good enough practices in scientific computing

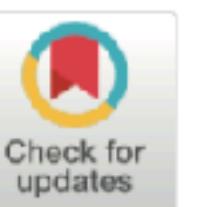
Greg Wilson<sup>1\*</sup>, Jennifer Bryan<sup>2</sup>, Karen Cranston<sup>3</sup>, Justin Kitzes<sup>4</sup>, Lex Nederbragt<sup>5</sup>, Tracy K. Teal<sup>6</sup>

**1** Software Carpentry Foundation, Austin, Texas, United States of America, **2** RStudio and Department of Statistics, University of British Columbia, Vancouver, British Columbia, Canada, **3** Department of Biology, Duke University, Durham, North Carolina, United States of America, **4** Energy and Resources Group, University of California, Berkeley, Berkeley, California, United States of America, **5** Centre for Ecological and Evolutionary Synthesis, University of Oslo, Oslo, Norway, **6** Data Carpentry, Davis, California, United States of America

\* These authors contributed equally to this work.

\* [gwilson@software-carpentry.org](mailto:gwilson@software-carpentry.org)

Greg Wilson, Jennifer Bryan, Karen Cranston,  
Justin Kitzes, Lex Nederbragt, Tracy K. Teal  
“Good enough practices in scientific computing.”  
PLoS computational biology 13.6 (2017): e1005510.



## OPEN ACCESS

**Citation:** Wilson G, Bryan J, Cranston K, Kitzes J, Nederbragt L, Teal TK (2017) Good enough practices in scientific computing. PLoS Comput Biol 13(6): e1005510. <https://doi.org/10.1371/journal.pcbi.1005510>

**Editor:** Francis Ouellette, Ontario Institute for Cancer Research, CANADA

**Published:** June 22, 2017

**Copyright:** © 2017 Wilson et al. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Funding:** The authors received no specific funding for this work.

**Competing interests:** The authors have declared that no competing interests exist.

## Author summary

Computers are now essential in all branches of science, but most researchers are never taught the equivalent of basic lab skills for research computing. As a result, data can get lost, analyses can take much longer than necessary, and researchers are limited in how effectively they can work with software and data. Computing workflows need to follow the same practices as lab projects and notebooks, with organized data, documented steps, and the project structured for reproducibility, but researchers new to computing often don't know where to start. This paper presents a set of good computing practices that every researcher can adopt, regardless of their current level of computational skill. These practices, which encompass data management, programming, collaborating with colleagues, organizing projects, tracking work, and writing manuscripts, are drawn from a wide variety of published sources from our daily lives and from our work with volunteer organizations that have delivered workshops to over 11,000 people since 2010.

## Overview

We present a set of computing tools and techniques that every researcher can and should consider adopting. These recommendations synthesize inspiration from our own work, from the experiences of the thousands of people who have taken part in Software Carpentry and Data Carpentry workshops over the past 6 years, and from a variety of other guides. Our recommendations are aimed specifically at people who are new to research computing.

## Introduction

Three years ago, a group of researchers involved in Software Carpentry and Data Carpentry wrote a paper called “Best Practices for Scientific Computing” [1]. That paper provided recommendations for people who were already doing significant amounts of computation in their research. However, as computing has become an essential part of science for all researchers, there is a larger group of people new to scientific computing, and the question then becomes, “where to start?”

# Improve your workflow for reproducible science



[bit.ly/repro-workflow](https://bit.ly/repro-workflow)

@minebocek

mine-cetinkaya-rundel

cetinkaya.mine@gmail.com