

INTRODUCTION TO DATA SCIENCE WITH R

Session 2: Data exploration basics

January 18, 2023

DATA EXPLORATION AND VISUALISATION: FIRST STEPS

REPRODUCIBILITY CHECKLIST

WHAT DOES IT MEAN FOR A DATA ANALYSIS TO BE “REPRODUCIBLE”?

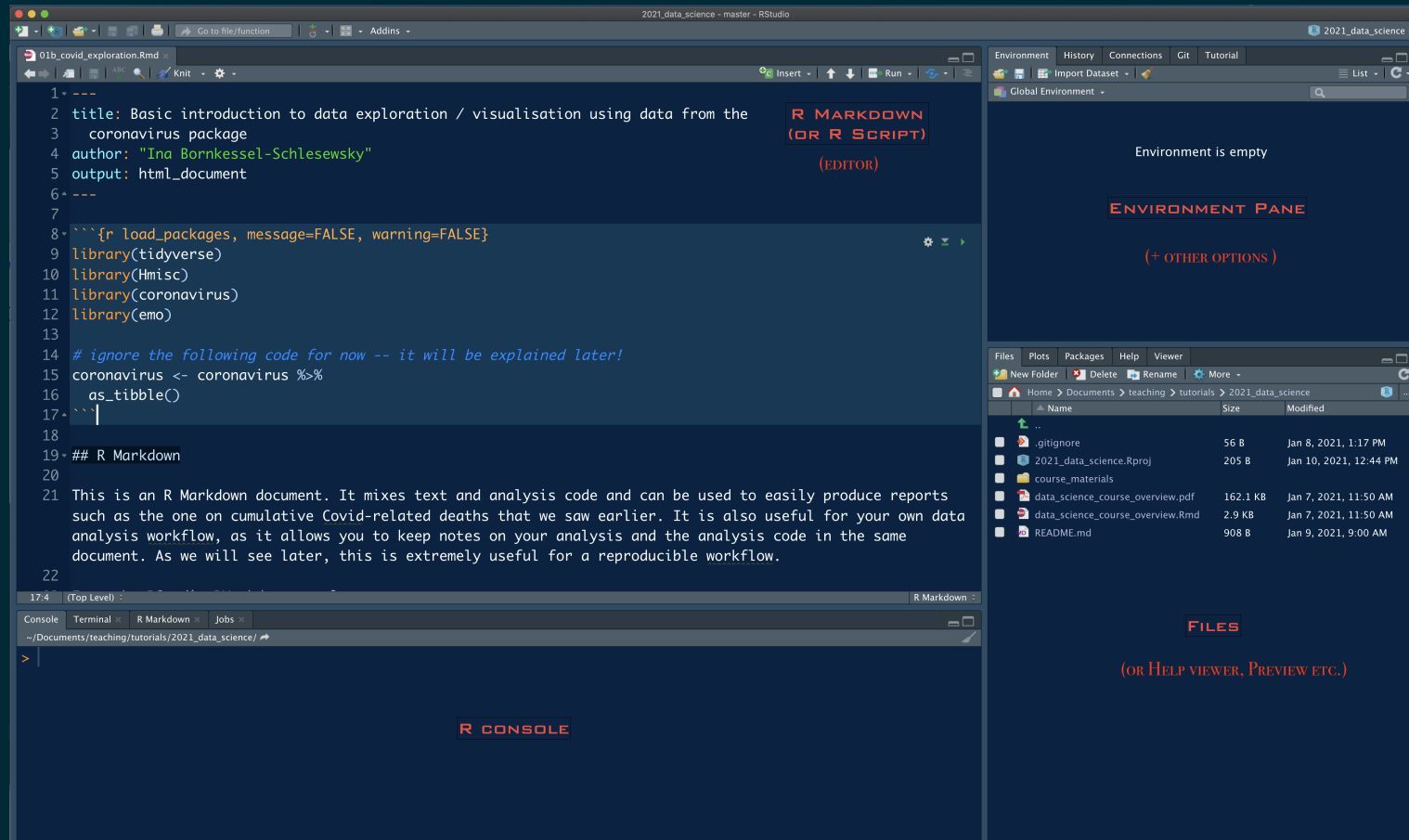
- Are the tables, figures (and any other results reported) reproducible from the code and data?
- Does the code actually do what you think it does?
- In addition to what was done, is it clear *why* it was done?

REPRODUCIBILITY TOOLKIT

- Scriptability → R
- Literate programming (code, narrative, output in one place) → Quarto
- Version control → Git / GitHub (more on this on Day 3)

Adapted from <https://github.com/rstudio-education/datascience-box/blob/master/course-materials/slides/u1-d02-toolkit-r/u1-d02-toolkit-r.Rmd>

A BIT MORE ON R AND RSTUDIO

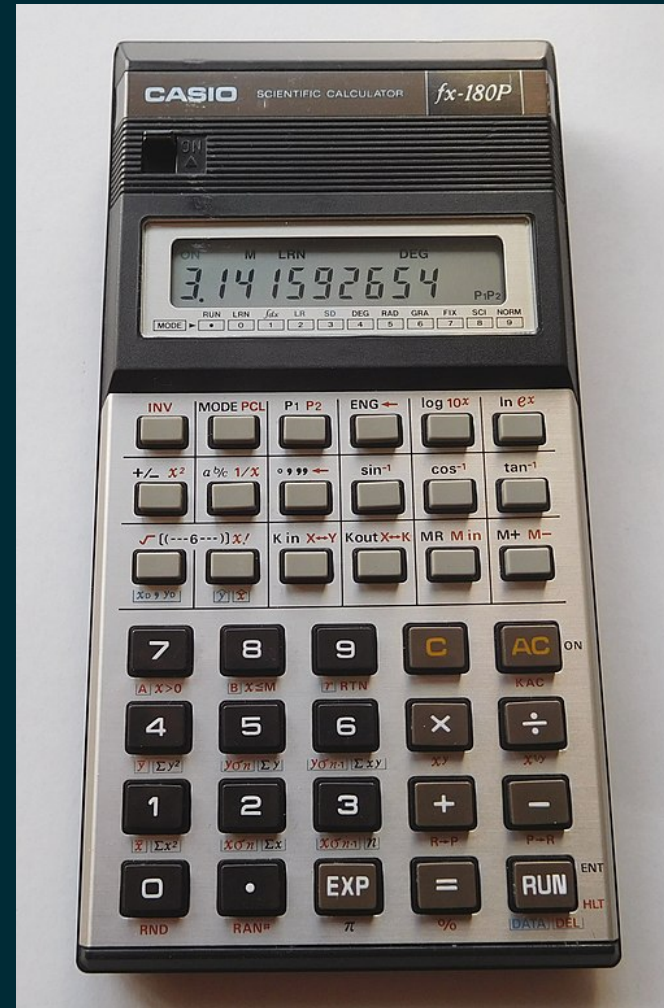


Note: this is the default layout - all the panes can be moved around to best suit your workflow. Note also the possible change of appearance (e.g. dark themes).

R CONSOLE

- this is where the “magic happens”, i.e. where the calculations take place
- think of a calculator on steroids 🤔
- we can interact with the console directly (see example)
- mostly, however, we will be working with scripted input

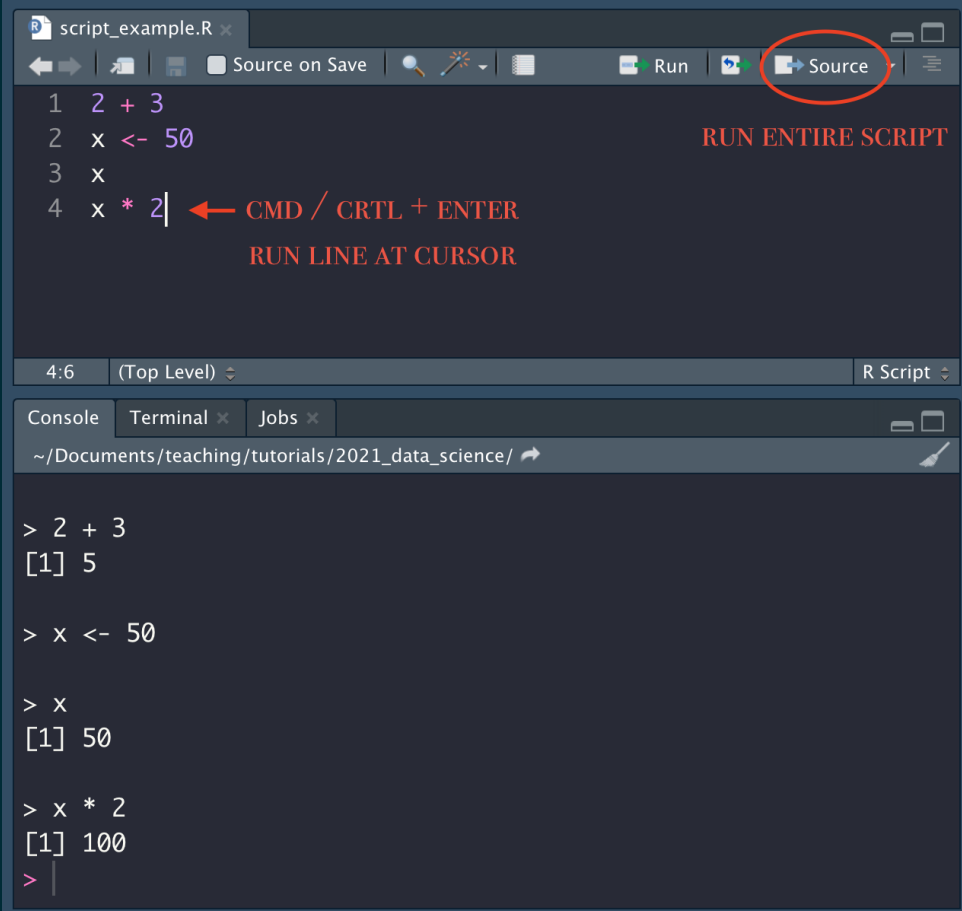
```
Console Terminal x R Markdown x Jobs x
~/Documents/teaching/tutorials/2021_data_science/ ➔
> 2 + 3
[1] 5
> x <- 50
> x
[1] 50
> x * 2
[1] 100
>
```



Cjp24 CC BY-SA 4.0, via Wikimedia Commons

SCRIPTED R CODE

- An R “script” is simply a text file with a collection of R commands
- Think of it as a set of instructions that you can feed into your “calculator”
- This is an important step towards reproducibility, as it means that you have a record of your analysis and you can recreate it at any time



The screenshot displays the RStudio interface. The top pane shows a script file named 'script_example.R' with the following R code:

```
1 2 + 3
2 x <- 50
3 x
4 x * 2
```

Annotations in red text are present: 'RUN ENTIRE SCRIPT' is next to the 'Source' button in the toolbar, and 'CMD / CTRL + ENTER' and 'RUN LINE AT CURSOR' are next to the fourth line of code. The 'Source' button is circled in red. The bottom pane shows the console output:

```
> 2 + 3
[1] 5

> x <- 50

> x
[1] 50

> x * 2
[1] 100
>
```

QUARTO

- Mixes text (formatted using Markdown) and R code
- Allows for documentation of analysis steps (the “why”): the next step towards reproducibility
- Easily generate reproducible reports in different formats (.html, .pdf, .docx)
- You can even use RMarkdown to create slides for presentations (these slides are written in RMarkdown!), interactive tutorials and interactive web applications

QUARTO

01b_basic_data_exploration_prep...

RENDER (i.e. format) INSERT CODE CHUNK

Render Run

Source Visual Outline

```
1 ---
2 title: "Data exploration basics"
3 author: "Ina Bornkessel-Schlesewsky"
4 date: "2023-01-18"
5 date-format: long
6 format: html
7 execute:
8   echo: TRUE
9   warning: FALSE
10  message: FALSE
11 ---
12
13 ## Load packages
14
15 We start by loading the packages that we will need using the library()
16 command.
17 ```{r}
18 library(tidyverse)
19 library(palmerpenguins)
20 ```
21
22 ## Inspect data
23
24 ```{r}
25 penguins
26
27 ```
28
```

HEADER WITH METADATA

TEXT

R CODE CHUNK

CHUNK OPERATIONS

QUARTO RENDERED DOCUMENT

The image shows a Quarto document in edit mode. The left pane displays the source code, and the right pane shows the rendered output.

Source Code:

```
1 ----  
2 title: "Data exploration basics"  
3 author: "Ina Bornkessel-Schlesewsky"  
4 date: "2023-01-18"  
5 date-format: long  
6 format: html  
7 execute:   
8   echo: TRUE  
9   warning: FALSE  
10  message: FALSE  
11 ----  
12   
13 ## Load packages  
14   
15 We start by loading the packages that we will need using the library()  
   command.  
16   
17 ```{r}  
18 library(tidyverse)  
19 library(palmerpenguins)  
20 ```  
21   
22 ## Inspect data  
23   
24 ```{r}  
25 penguins  
26   
27 ```  
28   
29 ## Basic data exploration  
30
```

Rendered Output:

Data exploration basics

AUTHOR
Ina Bornkessel-Schlesewsky

PUBLISHED
January 18, 2023

Load packages

We start by loading the packages that we will need using the `library()` command.

```
library(tidyverse)  
library(palmerpenguins)
```

Inspect data

```
penguins
```

A tibble: 344 × 8

	species	island	bill_length_mm	bill_depth_mm	flipper_mm ¹	bc
	<fct>	<fct>	<dbl>	<dbl>	<int>	
1	Adelie	Torgersen	39.1	18.7	181	
2	Adelie	Torgersen	39.5	17.4	186	
3	Adelie	Torgersen	40.3	18	195	
4	Adelie	Torgersen	NA	NA	NA	
5	Adelie	Torgersen	36.7	19.3	193	

YOUR TURN: LET'S EXPLORE THE PENGUINS DATA WITH THE HELP OF A QUARTO DOCUMENT!

- For this, we will be using a technique called “live coding”.
- I will show you how to construct R code, talking you through the rationale for various coding choices.
- By copying my code in real time, you will get a feel for how the process works.
- Always type the code; don't copy and paste it, even if certain aspects keep repeating. This will ensure that you start to build up the “muscle memory” required to conduct data exploration and visualisation yourself in future.
- For this session, we will be working with the file `01b_basic_data_exploration.qmd`, which you can find under `exercises > 01b_basic_data_exploration`

KEY LEARNING GOALS

BY THE END OF THIS SESSION, YOU SHOULD

- understand the basics of how to work with a Quarto document for data exploration
- know how to inspect a data frame
- be able to undertake simple data manipulation steps (filtering, sorting, summarising)
- be able to create basic graphs of different types using the `ggplot()` function

RESOURCES

- [R for Data Science \(2nd ed.\), Chapter 2](#)
- [Introduction to dplyr](#)
- [Posit Cheatsheets](#); see the ones on data transformation and visualisation in particular

FOR YOUR REFERENCE

USING BASIC DATA EXPLORATION FUNCTIONS

- Basic data exploration functions are typically verbs to remind you that they “do stuff” with a dataset (e.g. *filter*, *arrange*, *summarise*)
- start with the data that you want to explore / manipulate (e.g. **penguins**)
- use the “pipe” operator **|>** to send the data to a function (N.B. keyboard shortcut for the pipe: CTRL/CMD + SHIFT + M)
- specify the function and any additional parameters in parentheses (these differ depending on the function)
- this basic pattern works for all the functions we’ve looked at and more!

```
1 penguins |>
2   filter(species == "Adelie")
```

```
1 penguins |>
2   arrange(body_mass_g)
```

```
1 penguins |>
2   summarise(m_mass = mean(body_mass_g,
3                     na.rm=TRUE))
```

COMBINING BASIC FUNCTIONS

- You can use pipes to chain together different functions
- This is a very powerful basis for data exploration and visualisation!

```
1 penguins |>
2   filter(species == "Adelie") |>
3   arrange(body_mass_g)
```

```
1 penguins |>
2   filter(island == "Dream") |>
3   summarise(m_mass = mean(body_mass_g,
4                           na.rm=TRUE))
```

(VERY) BASIC PLOTTING

ANATOMY OF A GGPLOT

```
```{r}
penguins |>
 ggplot(aes(x = bill_length_mm, y = body_mass_g))
 geom_point()
```
```

“Aesthetics” such as what to put on the x and y axis

plus to add a layer of graphics

type of plot to create, e.g. geom_point() for a scatterplot

```
```{r}
penguins |>
 ggplot(aes(x = flipper_length_mm, y = body_mass_g,
 colour = species))
 geom_point()
```
```

can add additional aesthetics such as colour

USEFUL RSTUDIO KEYBOARD SHORTCUTS

- insert code chunk: **CMD (Mac) / CTRL (Windows) + Option / ALT + i**
- insert pipe: **CMD / CTRL + M**
- run block of code: **CMD / CTRL + Return / Enter**
 - you can place the cursor anywhere in a connected block of code for this
 - if you want to run only a particular selection of code, say part of a block, you can select it using the cursor and then use this keyboard shortcut

EXERCISES

Complete the exercises in the `session2_exercises.qmd` document (in the *exercises > 01b_basic_data_exploration* directory)

