BDSI R workshop – Reading data and producing quick summaries

Robert Cope & Terry Neeman

2022

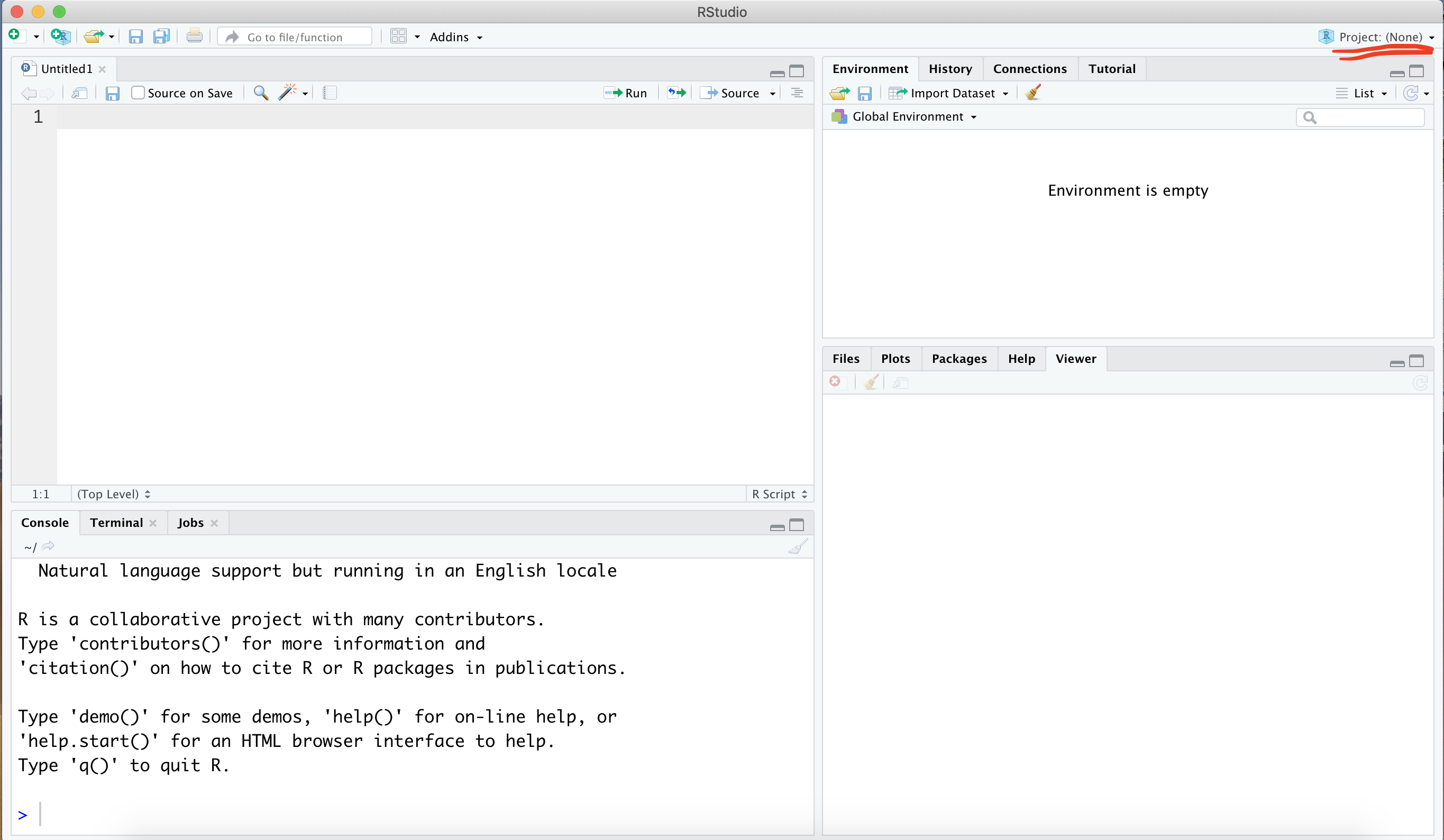
The goal of this workshop is to encourage you to be comfortable getting your data into R and using it for your basic data visualisations, summaries, etc.. Going forward, you will expand on these skills, learn some more complex techniques, and produce a statistical workflow for your data. But for the moment, you just need to feel ok about setting up a project, getting your data into R, and looking at it. This is the first and perhaps most important step of any statistical analysis. Further, having your work in R helps you keep a record of what you have done and helps us help you at the data analysis workshops and drop-in sessions.

# Working with data

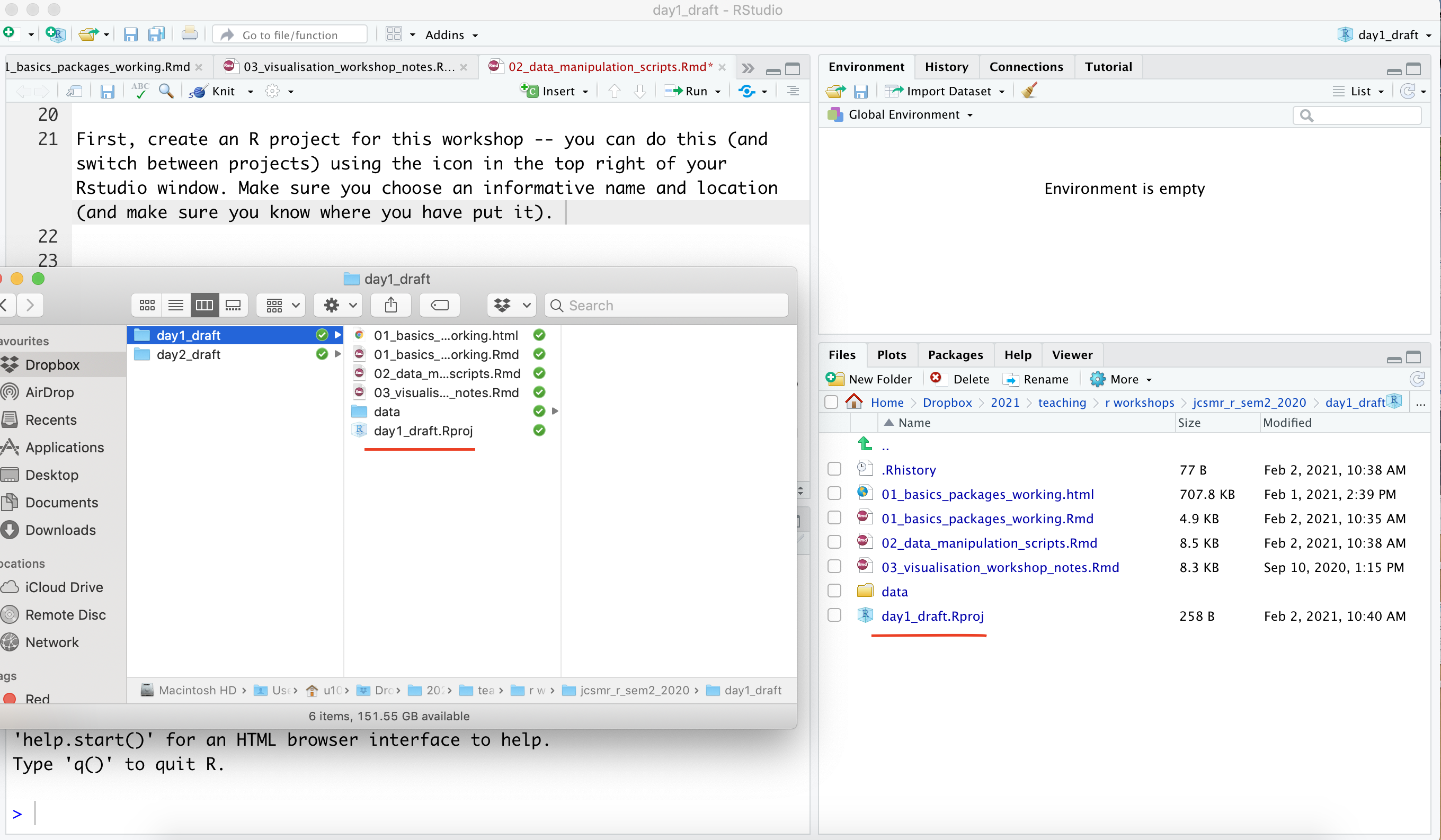
The strength of R over other languages is that it is built to handle data. We will start by looking at some data from the following paper:

Cuzick, J., Warwick, J., Pinney, E., Duffy, S. W., Cawthorn, S., Howell, A., … & Warren, R. M. (2011). Tamoxifen-induced reduction in mammographic density and breast cancer risk reduction: a nested case–control study. Journal of the National Cancer Institute, 103(9), 744-752.

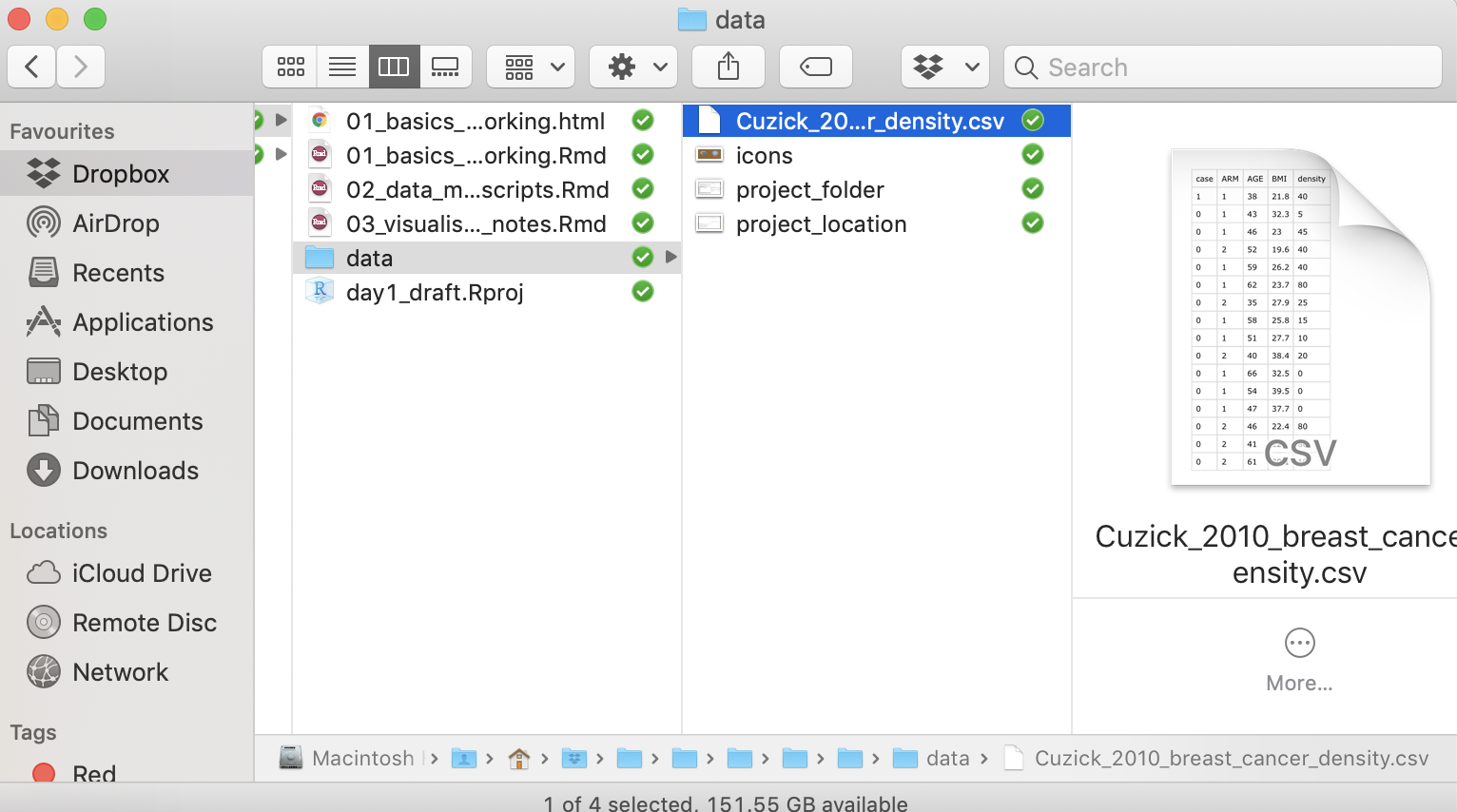
First, **create an Rstudio project for this workshop** – you can do this (and switch between projects) using the icon in the top right of your Rstudio window. Make sure you choose an informative name and location (and make sure you know where you have put it).



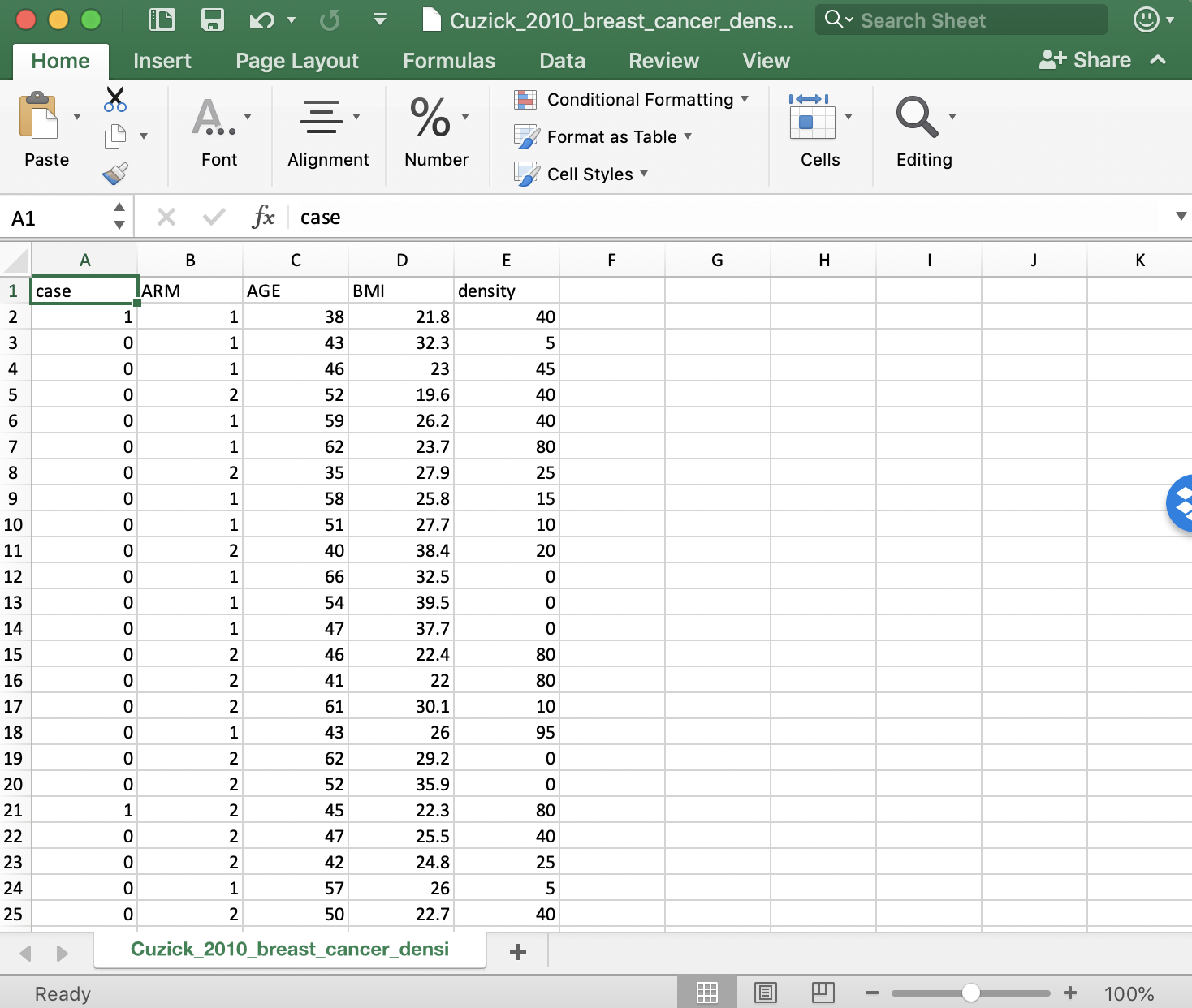
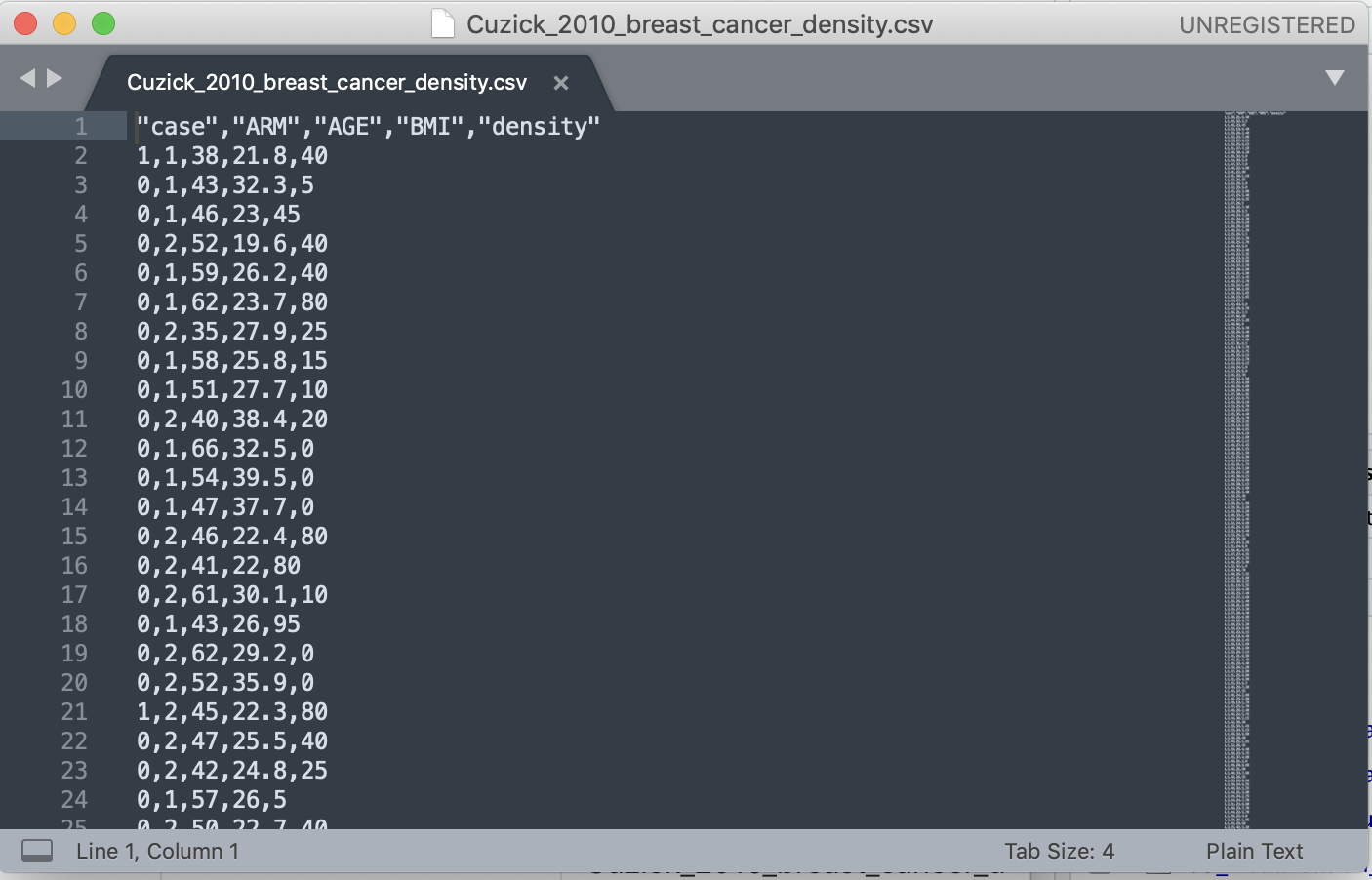
If you open that location using your finder / windows explorer you’ll see that the Rstudio project is just a folder with a .Rproj file inside – you can create subfolders (like ‘data’ shown here), copy and paste files, etc. as you normally would.



I will provide you with a data file over dropbox or similar. Download it, create a ‘data’ folder in your project, and put the file there.



This data is formatted as a ‘.csv’, which stands for ‘comma separated values’ – it is just a spreadsheet. Before looking at it in R, we can look at it in a text editor or in excel.



It is a single spreadsheet with no formatting – each line is a row, and columns are separated by commas. More specifically, each row represents a patient, and the columns are the relevant measurements/observations/variables. Note that the data starts in the top left, and has a single ‘header’ row with the names of the columns. This is the ideal way to set up your data for analysis. We will talk a little more about spreadsheets tomorrow.

## Reading in data

We will jump straight in to looking at the data in R. First, you need to load the ‘tidyverse’ library (you may need to install it if you haven’t).

#install.packages(tidyverse)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1   
## ✔ readr 2.1.2 ✔ forcats 0.5.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

Then, we read in the data using the read\_csv() function, and assign it to the variable with the name cancer\_df. Note: - when you create the file name (with ’‘) you can use tab to autocomplete it and avoid spelling mistakes etc. - the name of the file should appear green in rstudio, the other R code should stay black. -’<-’ means ‘is’, so you might read this line of code as ‘cancer\_df is the output of read\_csv() of/with the file “data/….csv”’ -

cancer\_df <- read\_csv('data/Cuzick\_2010\_breast\_cancer\_density.csv')

## Rows: 1065 Columns: 5  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (5): case, ARM, AGE, BMI, density  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

The first thing I would do after reading in a file is look at it.

cancer\_df

## # A tibble: 1,065 × 5  
## case ARM AGE BMI density  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 1 38 21.8 40  
## 2 0 1 43 32.3 5  
## 3 0 1 46 23 45  
## 4 0 2 52 19.6 40  
## 5 0 1 59 26.2 40  
## 6 0 1 62 23.7 80  
## 7 0 2 35 27.9 25  
## 8 0 1 58 25.8 15  
## 9 0 1 51 27.7 10  
## 10 0 2 40 38.4 20  
## # … with 1,055 more rows

A more informative/readable thing to look at is from the function str() or glimpse().

glimpse(cancer\_df)

## Rows: 1,065  
## Columns: 5  
## $ case <dbl> 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,…  
## $ ARM <dbl> 1, 1, 1, 2, 1, 1, 2, 1, 1, 2, 1, 1, 1, 2, 2, 2, 1, 2, 2, 2, 2,…  
## $ AGE <dbl> 38, 43, 46, 52, 59, 62, 35, 58, 51, 40, 66, 54, 47, 46, 41, 61…  
## $ BMI <dbl> 21.8, 32.3, 23.0, 19.6, 26.2, 23.7, 27.9, 25.8, 27.7, 38.4, 32…  
## $ density <dbl> 40, 5, 45, 40, 40, 80, 25, 15, 10, 20, 0, 0, 0, 80, 80, 10, 95…

It is important to check that the data are what you expect, i.e., columns that you expect to be numbers are represented that way. Read through each line of the str ouput (i.e., each column of the data) and think about what it is, what the values are, etc..

Another useful thing to look at is the summary of a data frame, or the head (i.e., the first few rows).

summary(cancer\_df)

## case ARM AGE BMI   
## Min. :0.0000 Min. :1.000 Min. :35.00 Min. :17.60   
## 1st Qu.:0.0000 1st Qu.:1.000 1st Qu.:46.00 1st Qu.:23.20   
## Median :0.0000 Median :1.000 Median :49.00 Median :25.70   
## Mean :0.1155 Mean :1.476 Mean :50.17 Mean :26.72   
## 3rd Qu.:0.0000 3rd Qu.:2.000 3rd Qu.:54.00 3rd Qu.:29.40   
## Max. :1.0000 Max. :2.000 Max. :70.00 Max. :50.40   
## NA's :16   
## density   
## Min. : 0.00   
## 1st Qu.: 15.00   
## Median : 40.00   
## Mean : 44.45   
## 3rd Qu.: 70.00   
## Max. :100.00   
##

head(cancer\_df)

## # A tibble: 6 × 5  
## case ARM AGE BMI density  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 1 38 21.8 40  
## 2 0 1 43 32.3 5  
## 3 0 1 46 23 45  
## 4 0 2 52 19.6 40  
## 5 0 1 59 26.2 40  
## 6 0 1 62 23.7 80

## The gtsummary package

The gtsummary package offers formatted tables to get a quick summaries of the data. The default summary statistics are median (IQR) for numerical data and n(%) for categorical data. You can change these defaults as you wish. Here are a few examples:

library(gtsummary)  
  
cancer\_df %>%  
 tbl\_summary()

## Table printed with {flextable}, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| Characteristic | N = 1,0651 |
| --- | --- |
| case | 123 (12%) |
| ARM |  |
| 1 | 558 (52%) |
| 2 | 507 (48%) |
| AGE | 49 (46, 54) |
| BMI | 25.7 (23.2, 29.4) |
| Unknown | 16 |
| density | 40 (15, 70) |
| 1n (%); Median (IQR) | |

In this example, there are two treatment groups. We can summarise characteristics by treatment group.

cancer\_df %>%  
 tbl\_summary(by = ARM) %>%  
 add\_overall(last = TRUE)

## Table printed with {flextable}, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| Characteristic | 1, N = 5581 | 2, N = 5071 | Overall, N = 1,0651 |
| --- | --- | --- | --- |
| case | 72 (13%) | 51 (10%) | 123 (12%) |
| AGE | 49 (46, 54) | 49 (46, 54) | 49 (46, 54) |
| BMI | 25.8 (23.4, 29.1) | 25.5 (23.0, 29.7) | 25.7 (23.2, 29.4) |
| Unknown | 6 | 10 | 16 |
| density | 45 (15, 70) | 40 (20, 70) | 40 (15, 70) |
| 1n (%); Median (IQR) | | | |

Often our data has many columns and we only wish to summarise a few of them. The select() function in dplyr package (part of tidyverse) allows us to select the columns we want to summarise.

cancer\_df %>%  
 select(ARM, AGE, BMI) %>%  
 tbl\_summary(by = ARM) %>%  
 add\_overall(last = TRUE)

## Table printed with {flextable}, not {gt}. Learn why at  
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html  
## To suppress this message, include `message = FALSE` in the code chunk header.

| Characteristic | 1, N = 5581 | 2, N = 5071 | Overall, N = 1,0651 |
| --- | --- | --- | --- |
| AGE | 49 (46, 54) | 49 (46, 54) | 49 (46, 54) |
| BMI | 25.8 (23.4, 29.1) | 25.5 (23.0, 29.7) | 25.7 (23.2, 29.4) |
| Unknown | 6 | 10 | 16 |
| 1Median (IQR) | | | |

## Getting help with R

You aren’t alone in your R journey. You should always expect to make use of the experience of those around you, this includes:

* the other participants in this workshop or people around you: often getting a second opinion from a friend can help solve a problem.
* local experts – e.g., you could attend the drop in sessions which run each Tuesday at 10am (see <https://bdsi.anu.edu.au/training-courses/bdsi-bioinformatics-and-statistics-drop-ssssions>), or email me and I’m happy to try to help.
* internet resources – there are formal resources like the cheat sheet, free online textbooks (a great starting point is R for Data Science <https://r4ds.had.co.nz/>), or you can almost always search google for your problem or an error message you have seen

The most useful thing you can do to make your life easier is to practice. If you don’t use R for a few months, you will likely forget and you will have to refresh, whereas the more often you practice the less likely to are to forget and the easier life will be.