R Lab - Day 2 (part 1)

Introduction to R and Rstudio

MF9130E V24

2024.04.09

Chi Zhang
Oslo Center for Biostatistics and Epidemiology
chi.zhang@medisin.uio.no

About

About me:
PhD in Biostatistics, UiO.
Researcher / R developer. R user since 2014

I also teach MF9130 (Intro to statistics to PhD students), ERN4110 (statistics to nutrition master students), MED3007 (statistics in genomics)

About the guided R lab sessions:

Purpose: help you get started with R, so that you can use it for your own analysis

Theory recap, examples, practice

Overview

Plan for this lecture

Introduction to R and Rstudio, with practice

- Create variables
- Data types
- Important data structures in R
- Import a dataset

Why do we use R

R has a few advantages over other softwares (e.g. STATA)

- It implements not only the **classical statistical models**, but also **latest developments** in methodology and computation
- Free, no need for license
- Powerful and flexible visualization tools
- Lots of users in different fields; active community: you can get help easily online
- In the future, your statistician is more likely to know R than other softwares

Rstudio interface

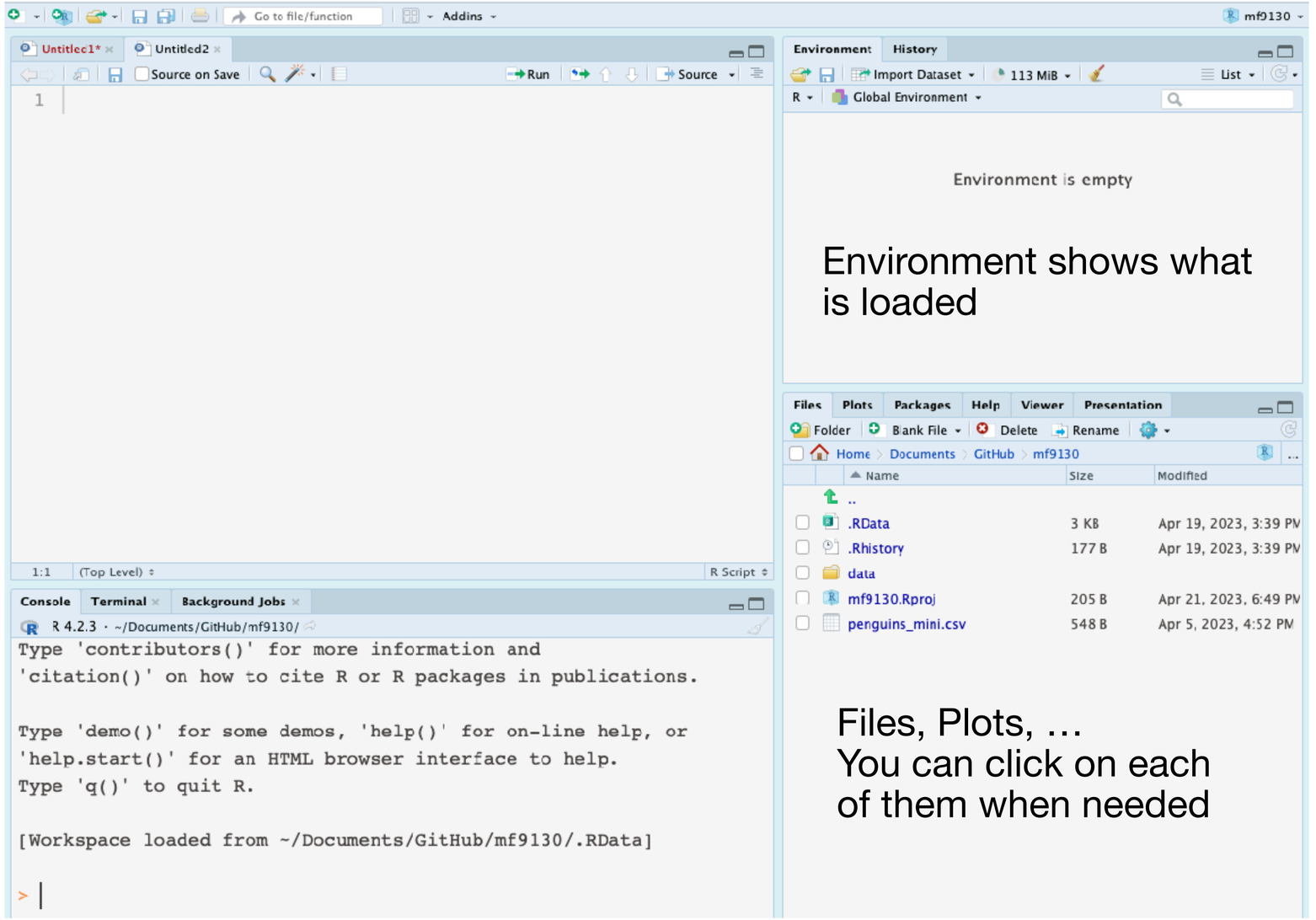
Script (source)

You can add commands and save them;

RUN the commands line by line with **ctrl+enter** (**cmd+enter** in macOS)

Console

This is where R code is executed (run) by pressing **enter**, and returns results



Create a variable (in console)

Open RStudio

Locate **Console**

Type in `a <- 3` in Console (after >)

Click **enter**

Create a variable (use R script)

Create an empty R **script**, call it “first_script.r”

In the script, type in `b <- 5`

Execute this line with **ctrl+enter** (**cmd+enter**)

R as calculator

```
a <- 3
b <- 4
c <- 7

# calculate the average of a,b,c

(a+b+c)/3
```

Data types

Common data types:

Numeric - numbers, such as 1.2, -1

(‘double’, ‘integer’…)

Character - “hadley”, “female”

(‘string’)

Logical - true or false, 1/0

(‘binary’)

```
a <- 3.1
class(a)

student <- “hadley”
class(student)

true_or_false <- T
# or TRUE
class(true_or_false)
```

Data structure

Scalar

33

Vector

1	2	3	4

a
b
c
d
e

Matrix

1	2	3
4	5	6
7	8	9

Data frame

age	sex	Smoker
33	F	FALSE
44	M	TRUE
34	M	FALSE

The elements in vectors and matrices need to be of the same type: all numeric; all characters, etc

Each **column** of a data frame needs to be of the same type

Different columns can be of different types

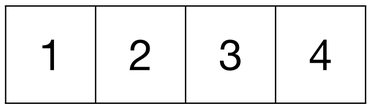
Your data most likely have mixed types of variables

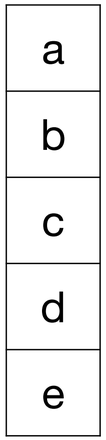
Vector

Create a vector

Use c() to combine elements (scalars)

Some shortcuts to create a sequence





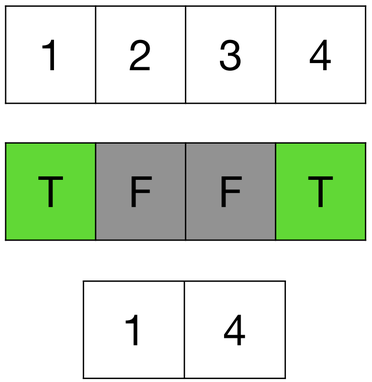

```
v1 <- c(1, 2, 3, 4)
v2 <- c("a", "b", "c", "d", "e")

# shortcuts

rep(0, 5) # repeat 0 for 5 times
v20 <- 1:20 # from 1 to 20

# combine with math operation

v20 * 2
```

Vector

Select elements of a vector

Select with indices (e.g. first, 3rd)

Select based on logical vector




```
# select the 3rd elements of v2
v1[3]


# select first 10 elements of v20
v20[1:10]

# select based on vector (T,F,F,T)
# prints 1 and 4
v1[c(T,F,F,T)]
```

(you will need it to filter two variables:
e.g. select heights for men and women (filter based on sex))

data.frame

Each row is a **subject**
(usually with unique IDs)

Each column is a **variable**
(feature, measurement, parameter)

age	sex	smoker
33	F	FALSE
44	M	TRUE
34	M	FALSE

Select elements from data frame
Index
$ operator
Variable name

```
# select the 3rd elements of v2
df <- data.frame(
  age = c(33, 44, 34),
  sex = c("F", "M", "M"),
  smoker = c(F, T, F)
)
df

# select first subject
df[1, ]

# select variable age
df$age
df[, "age"]
df[["age"]] # not df["age"]
```

Working with data

We are almost ready to practice the skills on a dataset!

Before we start the analysis, better keep organized!

Do you know where you are keeping the data, and where to tell R to look for it?

0. (Recommended step) **Create an R project.** This helps you stay organized with the workspace.

1. **Download data**

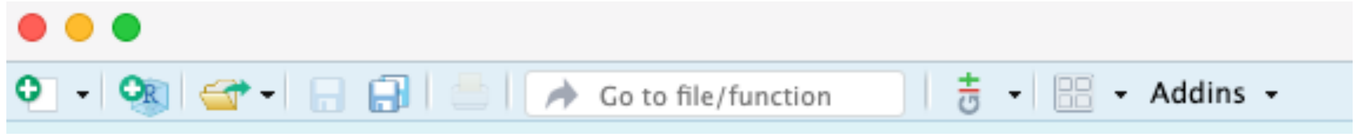
2. **Put it in the folder where you can find it**

3. **Load the data in Rstudio**

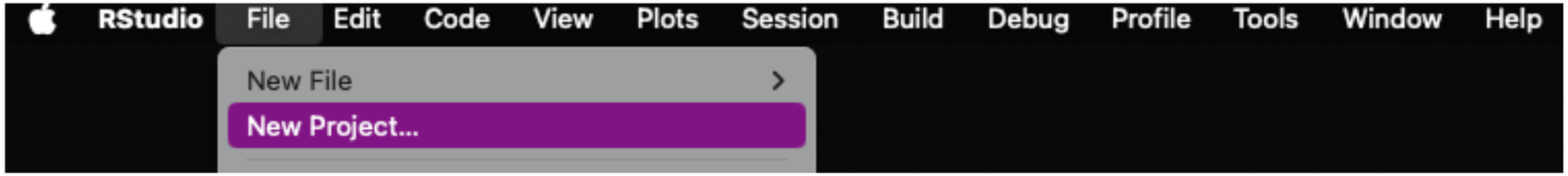
Working with data

0. Create an R project

In RStudio, locate the blue button;



Alternatively, File -> New Project



You can also follow the guide here.

https://ocbe-uio.github.io/teaching_mf9130e/lab/lab_intro_rstudio.html

Working with data

1. Download data

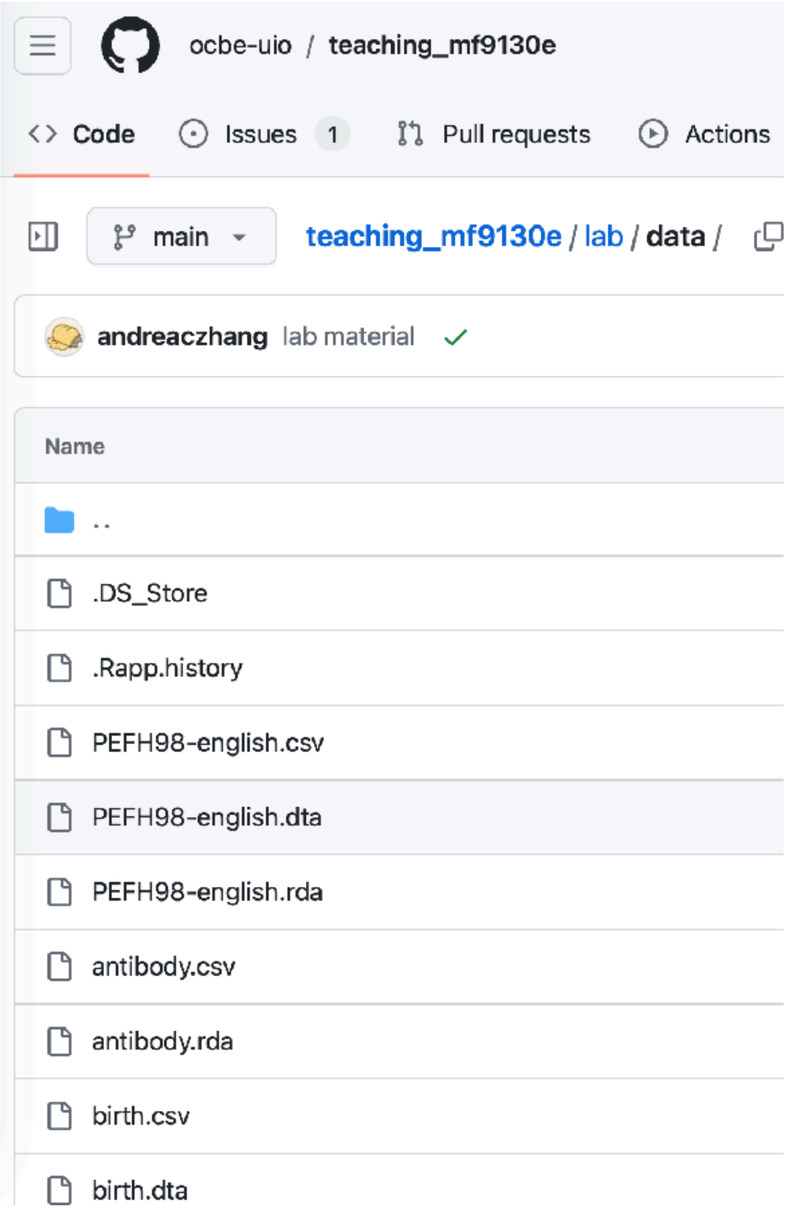
You can either download it from GitHub, or from Canvas.

Link on GitHub: https://github.com/ocbe-uio/teaching_mf9130e/tree/main/lab/data

Find the downloaded data
Sometimes it's in your **"Download" folder**

Common data formats:

.csv
.xlsx (excel sheet)
.RData, .rda (data format specific to R)
.txt



Working with data

2. Put data where you can find it

File system on your computer

~/folder/sub_folder/…/file

```
getwd() # know where you are
```

Find the folder where your new **R project** is

Make a new folder, name it ‘data’

Drag your data file inside the data folder

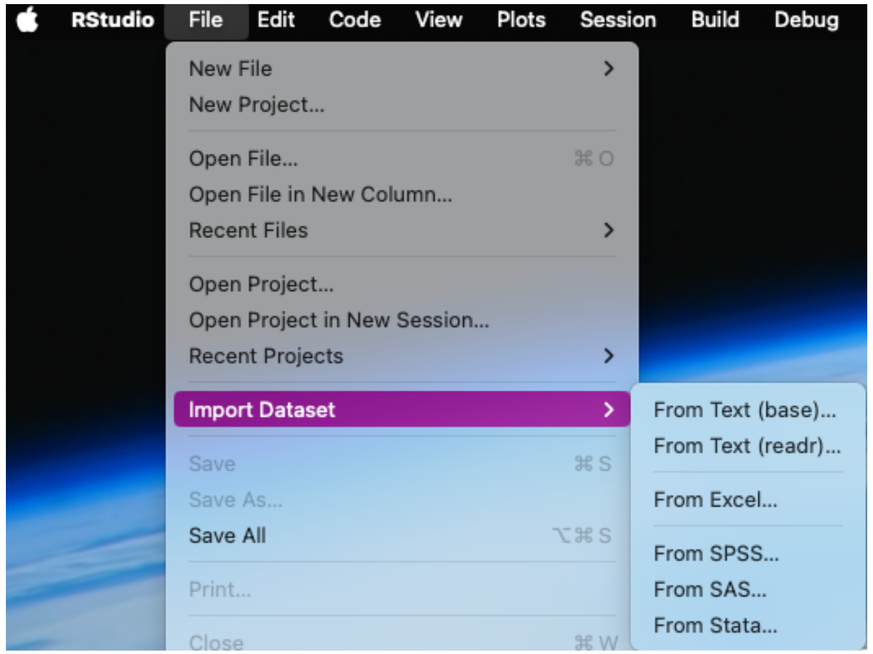
Alternatively, if you have experience to navigate the file system on your computer, feel free to put the data anywhere you can find it!

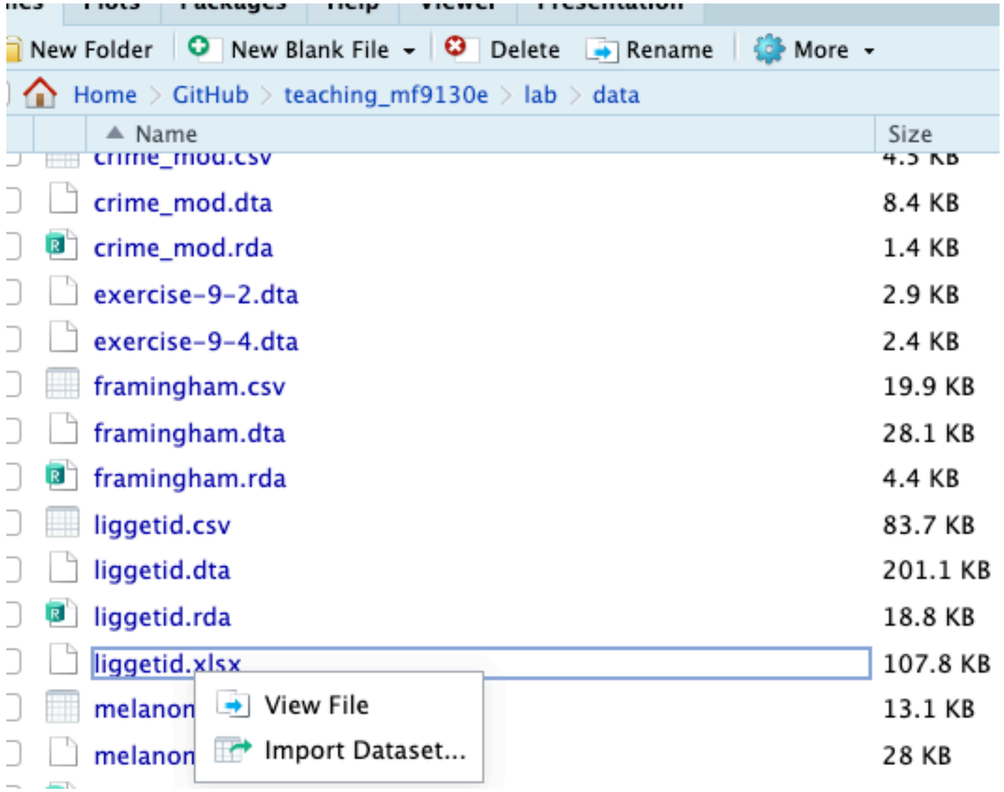
Working with data

3. Load the data (with Rstudio interface)

Option 1: click on the data file (*only if it's **.rda, .RData** format*)

Option 2: import dataset from Rstudio menu





Working with data

3. Load the data (with R command)

Option 3: use R command

e.g. load birth.rda data

The command depends on the data format. We will not be able to cover everything in this course, please search online to find the ones that you need!

```
getwd() # know where you are

# load a RData file (absolute path)
load("~/Documents/your_user_name/
data_folder/another_folder/birth.rda")


# load from R project (absolute)
load("~/Documents/this_project/data/
birth.rda")

# load from R project (relative)
load("data/birth.rda")
```