Week 1 Tutorial

Maria Prokofieva

15/01/2020

Week 1:

This week we are getting into basics: how to R

A screencast of the tutorial is available here:

Things to cover:

1. Setting up RStudio Cloud account here
2. Work through your `RStudio.cloud` account
3. Work with 2 types of documents:

- `R script`
- `RMarkdown` document.
- how to load libraries
- `R` syntax and basic functions: BUT

Remember: **This is not a programming unit! We focus on data and working with data**

What I need to learn:

- how to get to RStudio Cloud and navigate there
- how to create and work with `R script` files (basic)
- how to create and work with `RMarkdown` files (basic)
- how to create variables in R
- how to load datasets to variables
- how to load `pre-built` datasets
- how to `have a look` at values in variables

Resources

Chapter 1 Getting Started with Data in R https://moderndive.netlify.app/1-getting-started.html Basic Basics https://rladiessydney.org/courses/ryouwithme/01-basicbasics-0/ Workflow: basics https://r4ds.had.co.nz/workflow-basics.html

What is R?

https://www.computerworld.com/article/2497143/business-intelligence/business-intelligence-beginner-s-guide-to-r-introduction.html

R is a language and environment for statistical computing and graphics.

The R language is widely used for data science and stats to do data exploration, data analysis, data viz as well as developing statistical software.

R is free to use and the number of users are growing fast (well, it's free and bubbly community!) with lots of helpful packages at `CRAN` = `Comprehensive R Archive Network`.

`CRAN` is a network of ftp and web servers around the world that store identical, up-to-date, versions of code and documentation for R.

RStudio and RStudio Cloud

`RStudio` comes as a desktop application (*IDE* = Integrated development environment) and `RStudio Cloud` (which we are using in this unit).

You do not need to install `RStudio Cloud`, you just create your account there when you go to https://rstudio.cloud/

Task: go to https://rstudio.cloud/ and create your account there. You can use ANY email address to set this up. Please note the password you are using there

Optional: you can install a desktop version of RStudio from here. Make sure that you install R first from here and then RStudio. This will allow you to run a copy of RStudio on your local computer.

The RStudio interface has the following components including: 1. `Source editor` window`where you write and edit your files: - Docking station for multiple files - each file is tab - Useful shortcuts, such as`Run`for R code or`Knit‘ for RMarkdown documents
- Code-checking: give you "suggestions" as you write (e.g. hints about syntax of functions) - Debugging features

2. `Console` window where you can run your code directly as well as it shows execution of your scripts

- Highlighting/Tab-completion, e.g. Red spot shows you when your code is executed
- "History"= Recent commands

3. Other tabs/windows:

- Files: to access your files and navigate folders
- Plots: to show your graphics and data viz
- Help: very useful as you will get all the R knowledge there and R documentation
- Other tools, such as package development, git, etc

There's a cheatsheet in the "Help" menu, on tips for using this interface.

Task:

- Set up your free RStudio Cloud account and create your workspace "BCO6007"

`Workspace` is your current R working environment. It includes all your objects, such as data and files. Once you finish your work, it saves a copy of the current workspace and reloads the next time R is started.

Projects

Your workspace may have 1+ **projects**

`Project` are folders where you keep your work organized: it has your data, your code, your results all located in one place.

Task:

- Create a ptoject called `BCO6007` and locate your files under the `Files` tab. Make sure that you always work in the correct project for this class,

R Script and RMarkdown documents

we will work with two types of documents in this unit: - R script: R code to do data analysis - RMarkDown document: which is a document that combines R script and normal text. In the rmarkdown document you can present your data analysis (R code) and talk about data and results of analysis as well as show dataviz.

Task:

- Create an empty R script:
- File -> New file -> R script
- Save the file as `practice1.R`
- Upload a rmarkdown document to your project `week1.Rmd` You need to download the file from VU-Collaborate to your local computer and then upload it.
- Open the file in RStudio Cloud.

Notice the difference in extension `.R` vs `.Rmd`

You can read more about RMarkdown here. We will also have a dedicated tutorial for it.

**To run an .R script you need to press `Run` or `Ctrl+Enter` (on Mac `Command+Return`) the line with your cursor.

To run your RMarkdown document you need to press `Knit` from the top panel.

Run `week1.Rmd` by pressing Knit and compare the page that is generated (Web page) with the code in your RStudio Cloud.

Tasks Work out how to run - a chunk of code in Rstudio - one line of R code? - Loate the line that starts with "“`{r}.` `What happens when you`Run‘ it

Some R Basics

Switch to your `practice1.R` and Type and run the following commands:

```
150/3
a<-5*10^2
b<-"Cool"
```

Notice what is `<-` It assigns value to variables that appear in `Environment`

- R has good help support for documentation. Find the help page for the `mean` command, either from the help menu, or by typing one of these: `help(mean)` and `?mean`. Most help pages have examples at the bottom.
- The `summary` command can be applied to almost anything to get a summary of the object. Try `summary(c(3,4, 6, 4))`

What do you think is `c(x, y, z)` - ti is a **vector* with data.

But this is to "booooring", let's move on to more exciting things~

Please read this section from "R for Data Science"

Working with packages (=libraries)

Package is a collection of some functions (=functionalities) and some datasets to try them. It is also called `library`

Most of the time you work with packages, though some functions are pre-built in R. The ones we will be using in this unit most are `ggplot2`, `tidyverse` and `tidymodels`.

To use a package you need to **install* it (you do it only ONCE) and then *load* any time in your file when you want to use it.

To install a package you need to use `install.packages(NameOfThePackage)`

```
install.packages("ggplot2")
# by the way use # to include NOTES in your code
```

To load your package (=can call them library!) you need to do `library(NameOfThePackage)`

```
library(ggplot2)
# by the way use # to include NOTES in your code
```

Tasks Install and load the following packages: `tidyverse` and `tidymodels`

Questions

- What's an R `package`?
- How do you install a package?
- Why do you use the `library()` function ?
- How often do you load a `package`?

Getting data

Data can be found in R packages

To view the data we can load it and use `head` function to have a look at first 6 rows of our data.

You can also use `glimpse` function to get an idea of the variables in the data.

```
data(mtcars)
head(mtcars) #let's have a look at mtcars dataset

library(tidyverse)
data(economics, package = "ggplot2")
# data frames are essentially a list of vectors
glimpse(economics)
```

Use `Help` window to locate docs on `glimpse` and `head`

Some datasets are very very popular

```
library(gapminder)
glimpse(gapminder)
```

Most of the time you work with datasets in files, e.g. `csv` or `excel`

To load your files to us we are using one of the `tidyverse` functions (to be exact it is in `readr` library which is part of the very very very big library `tidyverse`)

```
#let load some data from TidyTuesday
boston_cocktails <- read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/
glimpse(candy)
```

`TidyTuesday` is a weekly data project aimed at the R ecosystem. You can have a look at it here

To load an excel file you need to use `readxl` library (remember to install it first and then load)

You can load this `us_avg_tuition.xlsx` file from `VU Collaborate ->Datasets` to your local computer and then upload it to your RStudio project. Let's assume that you upload it to the root folder

```
#install.packages("readxl")
library(readxl)
tuition <- read_excel('us_avg_tuition.xlsx')

glimpse(tuition)
```

To save the file you need to use `write_csv()` function. Read the help on how to use it!

RMarkdown document

`RMarkdown` is a document that contains some `R code` (=R chunks) and normal text and graphics.

You can create `RMarkdown` document through File->New file->R Markdown. Today your tutorial is in .rmd.

Task

- Download your `week1.rmd` from `VU Collaborate` and upload it to your Rstudio.
- Open the file and have a look

We will work more closely with .rmd files later, but at this stage have a very brief look at syntax and how to *generate* the output for this file.

Please `Knit` at the top and see what is going to happen! ~

Task

- Read the help file about `gapminder` data. Have a closer look at the data and think about possible questions you can answer using it.
- Locate this line in your .rmd document in RStudio and add your notes there!

Homework

Set up your *free* DataCamp account and have a look at the free tutorial Introduction to R they offer. This provides some good insights on the data types you will commonly use in R.