

Lab 1: R and RStudio

[Code](#)

AUTHOR

Maxim Dokukin

PUBLISHED

Due January 29, 2024

In this introductory lab, you will familiarize yourself with the R console and R Studio layout. Before you begin, you should install both R and RStudio using the instructions in the lecture slides. **Follow the instructions below and submit a pdf document with your answers to the following questions on Gradescope.**

Part A. Introduction to R

Open R (not RStudio). You should see a window that looks something like this (your version number may differ slightly):

At the bottom of this window, you should see a `>` symbol. This is called a prompt and indicates that R is waiting for instructions (in the form of code). Type `100 + 100` and then press return or enter. R should produce the following output: `[1] 200`. For now, ignore the `[1]`. This is R's way of letting us know that the answer is a **single** value: 200.

1. Compute 2^{1023} by typing the line: `2 ^ 1023`. What is the resulting value? Do you trust R's answer? 8.988466e+307. Good enough for certain applications.
2. Compute 2^{1024} by typing the line: `2 ^ 1024`. What is the resulting value? Do you trust R's answer? Inf. Trust issues.

We can save values as objects by giving them names. Try this out by typing `x <- 10` and then pressing return/enter.

3. What does the command `print(x)` produce? Try to predict what the answer will be before you run the code. `[1] 10`
4. What does the command `x + 10` produce? Try to predict what the answer will be before you run the code. `[1] 20`
5. Run the following two lines of code, one after the other. What is the result? Try to predict what the answer will be before you run the code.

```
x <- x + 1  
x * x
```

Since x was 10, then 11, $11*11=121$

[1] 121

6. Run the following lines of code, one after the other. What is the result? In your own words, explain what this code is doing.

```
sides <- c("H", "T")  
flip1 <- sample(sides, 1, replace = T)  
print(flip1)
```

It's flipping a coin object, with possible outputs being 'H' or 'T' 1 time.

[1] "T"

7. Run the following lines of code, one after the other. What is the result? In your own words, explain what this code is doing.

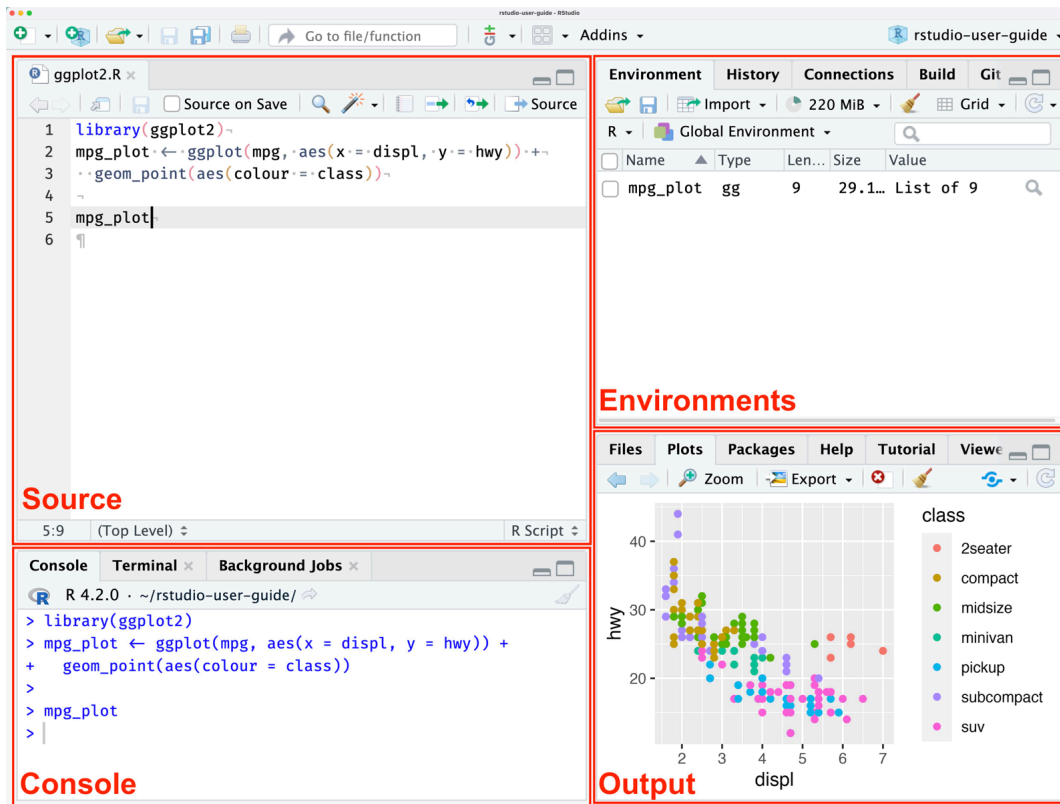
```
sides <- c("H", "T")  
flip10 <- sample(sides, 10, replace = T)  
print(flip10)
```

It's flipping a coin object, with possible outputs being 'H' or 'T' 10 times.

[1] "T" "T" "H" "H" "H" "T" "H" "H" "H" "T"

Part B. Introduction to RStudio

Now, open RStudio. It should look something like this image, from the official [RStudio Guide](#):



RStudio is like Microsoft Word/Google Docs for writing R code—it has useful tools (analogous to spell check) for programmers. In the lower left corner is your Console. This should look like the R window you were working in for Part A. The Console is meant for brief **interactive R** commands. The top right corner should be your **Environment** pane, which displays R objects that you have created. Right now your environment should be empty.

- In the **Console**, type the command `x <- 10`. What do you see in your environment pane?

Values:

x is 10

- The bottom right pane is the **Output** pane, which will often be used to display results/visualizations of your R code. There are many tabs here, but click on the **Files** tab. What do you see? In your own words, explain what you think the **Files** tab shows.

It's file manager window. Mine is set up in the organized way covered in the class.

- The top left pane is the **Source** pane. When you want to write more

than one line of code at a time, it's a good idea to organize your code in a single file. At the top of the RStudio window, click **File -> New File -> R Script** to open an untitled **.R** file. Copy and paste the following code into your R file. Then highlight the code and click **Run** (which will be in the top left corner of the Source pane). What do you see?

```
library(ggplot2)
mpg_plot <- ggplot(mpg, aes(x = displ, y = hwy)) +
  geom_point(aes(color = class))
mpg_plot
```

Error in library(ggplot2) : there is no package called 'ggplot2'

```
$install.packages("ggplot2")
```

After this command, there is a scatter plot with axes hwy and displ.

When you are finished, remember to write up your solutions, save them in a pdf file, and submit to Gradescope.