Coding Practice 1 Introduction to R

PLPA-5820 (Spring 2025)

Gabriel A. A. Silva

2025-02-06

Introduction to R – 25 pts

This assignment is intended to introduce you to R, RStudio, and integration with GitHub, as well as help you practice some R coding. You may work in pairs. When submitting the assignment on Canvas, please put both names on the assignment.

5 pts. Explain the following things about R and Rstudio:

- Where do you type code?

  > You type code in the console (bottom left) or in the script editor (top left), which can then be run.

- Where is the output of the code?

  > In the console or if it's a plot, it will show up in the plot window (bottom right).

- Where do plots show up?

  > In the plot window (bottom right).

- What is a global environment?

  > It's where the objects created by the code are stored. They are listed in th top right window.

- How do you get help for a function through R studio?

  > Through the command `?function_name` in the console or in the help tab (bottom right).

- What is an R package?

  An R packages are extentions to R that can contein code, data and documentation for easy distribution and use.

- What is a function?

  A function is a block of code that performs a specific task.

- How do you find the installed and loaded packages?

  Using `installed.packages()` and `search()`.

- What is a working directory, and how do you find it?

  The working directory is the current directory loaded in the enviroment, and can be found with `getwd()`.

- What is a relative file path and how is it different than an absolute file path?

  A relative path is a path that is relative to other directories, usually the current one. An absolute path is the full path to the file from the root of the system, and independs on the current location.

2 pts. Explain the steps to start a new R project connected to a GitHub repository.

With RStudio open, you click in “Create a project” button in the top right corner, then choose “Version Control”, then “Git”, and paste the URL of the repository. Then you choose the directory where the project will be saved and click “Create Project”.

3 pts. Please explain the different data classes and how they are similar and different from each other.

- Vector

  A vector is a collection of elements of the same class, and can be thought of as a list.

- Dataframe

  A dataframe is a collection of vectors of the same length, and can be thought of as a table, where each column is a vector of some class.

- Matrix

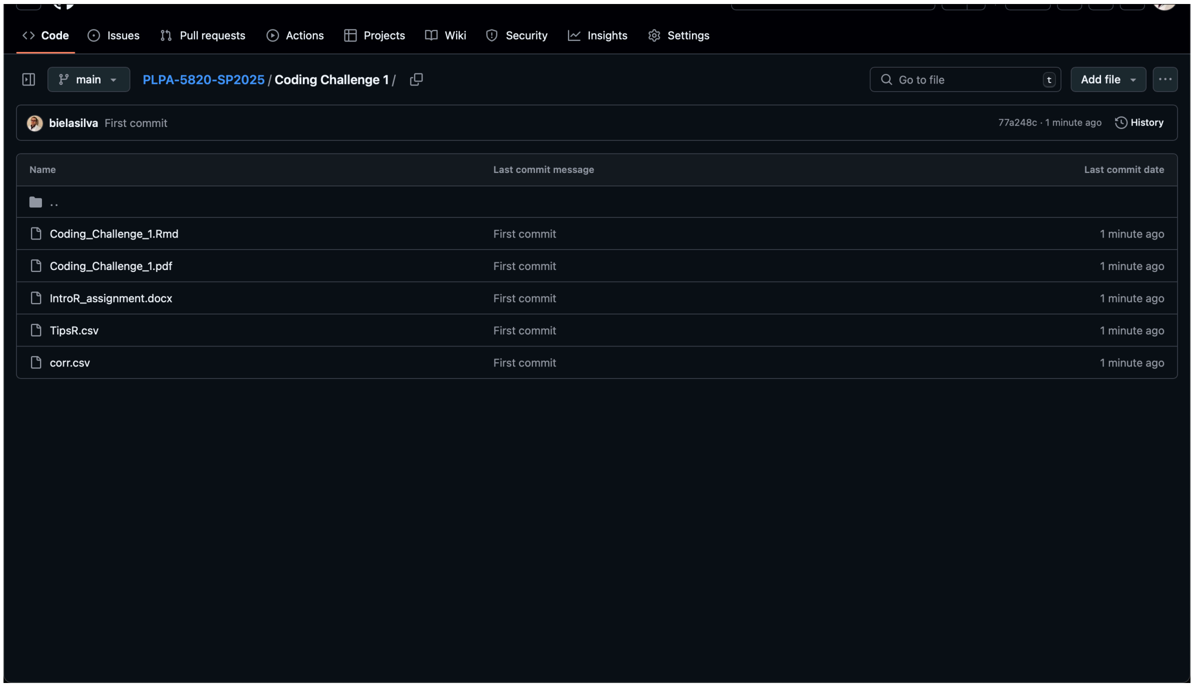
A matrix is also collection of elements in 2 dimentions. But unlike dataframes, all elements are of the same class.

10 pts. Complete all tasks in an R script and push it to GitHub.

- Create a vector named 'z' with the values 1 to 200
- Print the mean and standard deviation of z on the console
- Create a logical vector named zlog that is 'TRUE' for z values greater than 30 and 'FALSE' otherwise.
- Make a dataframe with z and zlog as columns. Name the dataframe zdf
- Change the column names in your new dataframe to equal "zvec" and "zlogic"
- Make a new column in your dataframe equal to zvec squared (i.e., z2). Call the new column zsquared.
- Subset the dataframe with and without the subset() function to only include values of zsquared greater than 10 and less than 100
- Subset the zdf dataframe to only include the values on row 26
- Subset the zdf dataframe to only include the values in the column zsquared in the 180th row.
- Annotate your code, commit the changes and push it to your GitHub

```r
# Create a vector named 'z' with the values 1 to 200
z <- 1:200

# Print the mean and standard deviation of z on the console
print(paste("Mean:", mean(z)))
print(paste("Standard Deviation:", sd(z)))

# Create a logical vector named zlog that is 'TRUE' for z values greater than
# 30 and 'FALSE' otherwise.
zlog <- z > 30

# Make a dataframe with z and zlog as columns. Name the dataframe zdf
zdf <- data.frame(z, zlog)

# Change the column names in your new dataframe to equal "zvec" and "zlogic"
colnames(zdf) <- c("zvec", "zlogic")

# Make a new column in your dataframe equal to zvec squared (i.e., z2).
# Call the new column zsquared.
zdf$zsquared <- zdf$zvec^2

# Subset the dataframe with and without the subset() function to only include
# values of zsquared greater than 10 and less than 100
zdf_sub <- zdf[zdf$zsquared > 10 & zdf$zsquared < 100, ]
zdf_sub2 <- subset(zdf, zsquared > 10 & zsquared < 100)

# Subset the zdf dataframe to only include the values on row 26
zdf_row26 <- zdf[26, ]

# Subset the zdf dataframe to only include the values in the column zsquared
# in the 180th row.
zdf_row180 <- zdf[180, "zsquared"]
```

```
[1] "Mean: 100.5"
[1] "Standard Deviation: 57.8791845139511"
```



PLPA-5820-SP2025 / Coding Challenge 1 /

bielasilva First commit 77a248c · 1 minute ago

Name	Last commit message	Last commit date
..		
Coding_Challenge_1.Rmd	First commit	1 minute ago
Coding_Challenge_1.pdf	First commit	1 minute ago
IntroR_assignment.docx	First commit	1 minute ago
TipsR.csv	First commit	1 minute ago
corr.csv	First commit	1 minute ago

Figure 1: GitHub Screenshot

5 pts. Download the Tips.csv file from Canvas. Use the read.csv() function to read the data into R so that the missing values are properly coded. **Note the missing values are reported in the data as a period (i.e., "."). How do you know the data were read correctly?

```
# Load the data
tips <- read.csv("TipsR.csv", na.strings = ".")

# You can visually inspect the data using the View() function
View(tips)

# Or you can use is.na() to check for missing values, which returns a bollean
# matrix where TRUE is a missing value, and then use any() on the result to
# see if there are any true values. Which will return a single TRUE if there
# is at least 1 missing value.
any(is.na(tips))
```

```
[1] TRUE
```