Socio Informatics 348
Practical 0

Purpose

This is a non-compulsory practical designed to help students familiarise themselves with RStudio, which will be used throughout the course.

Instructions

1. Download and install:
   - R: `https://cran.r-project.org`
   - RStudio: `https://posit.co/download/rstudio-desktop/`
   - If you are using a university computer, this should already be done.
2. Open RStudio and create a new script (Name it appropriately – stick to naming conventions). Complete the following steps in the script and run the lines of code you have written by making sure the cursor is on the line you want to run and pressing `Ctrl + Enter` (or `Cmd + Enter` on Mac). You can also select multiple lines and run them together.
3. Install the tidyverse package and its dependencies. This package is essential for data manipulation and visualisation in R. You can do this by running the following:

   ```
   install.packages("tidyverse")
   ```

   You only need to install a package once, but you can run this command again if you want to ensure you have the latest version.
4. Load the tidyverse package into your R session by running:

   ```
   library(tidyverse)
   ```

   This will make all the functions and datasets in the tidyverse package available for use. You will need to do this at the start of every new R session.

5. Identify loaded packages by refering to the `Packages` tab in the bottom right pane. You can also check the loaded packages by running:

```
search()
```

6. Get help on a specific package. Tidyverse includes several packages such as `ggplot2`, `dplyr`, and `tidyr`. You can get help on a specific package by running:

```
help(package = "tidyr")
```

   This will open the documentation for the `tidyr` package in the Help pane of RStudio.

7. Explore the documentation for the `tidyr` package. You can explore specific functions within the package by clicking on the function names in the documentation or by entering "?" followed by the function name in the console.
   For example, to get help on the `drop_na` function, you can run:

```
?drop_na
```

   This will open the documentation for the `drop_na` function in the Help pane of RStudio.

8. Check and set your working directory. The working directory is the folder where R will read and save files by default. You can check your current working directory by running:

```
getwd()
```

   To set a new working directory, you can use the following command:

```
setwd("path/to/your/directory")
```

   Replace "path/to/your/directory" with the actual path to your desired directory.

9. A string variable is a variable that stores a sequence of characters, such as words or sentences. In R, string variables are typically represented as text enclosed in quotation marks (e.g., `"Socio Informatics"`). Create a string variable and count the number of characters in it. For example:

```
my_string <- "Socio Informatics"
nchar(my_string)
```

This will return the number of characters in `my_string`.

10. Explore different number types in R. Try assigning integer, and numeric (double) values to variables and check their types using the `typeof()` function:

```
x <- 42L      # Integer
y <- 3.14     # Numeric (double)
typeof(x)
typeof(y)
```

11. Create a logical variable. Logical variables can hold values of `TRUE` or `FALSE`. For example:

```
is_true <- TRUE
is_false <- FALSE
typeof(is_true)
```

12. Check for missing values using `is.na()`. For example:

```
values <- c(1, NA, 3)
is.na(values)
```

This will return a logical vector indicating which elements are missing (`NA`).

Additional Resources

Below are some additional resources that can help you get started with R and RStudio:

- 23 RStudio Tips, Tricks, and Shortcuts (dataquest.io)
- Cheatsheets for easy reference
- Kaggle: A platform for data science competitions and datasets. This is a great place to practice/test your R skills.