Week-5: Code-along

NM2207: Computational Media Literacy

2023-09-07

Welcome! Go through the steps described below, *carefully*. It is totally fine to get stuck - **ASK FOR HELP**; reach out to your friends, TAs, or the discussion forum on Canvas.

Here is what you have to do,

1. **Download** `Code-along.Rmd` file from Canvas and move it to the folder "Week-5" (see instructions for creating folder in Section I below)
2. **Open** the video lectures and start listening to them
3. **Every time** you come across a code chunk (inside shaded blocks) in the lecture video, **Pause the video**
4. **Edit** the `Code-along.Rmd` file with the codes explained in the lecture videos within appropriate R chunk/code-block/shaded area (environment enclosed within ```)
5. **Comments** inside the R chunk/code-block/shaded area indicates which command explained in the lecture should be typed in there
6. **Set** `eval=TRUE` to generate the output and verify it to the one shown in the lecture videos
7. **Knit** the file upon completion and submit the pdf document on Canvas **before** coming to the tutorial session

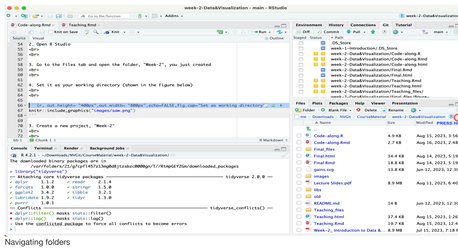
I. Preliminaries

1. Create a new folder, "Week-5", inside "NM2207" folder you created last week

2. Open R Studio

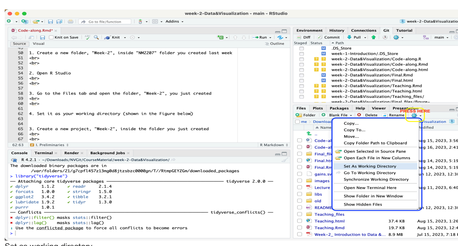
3. Go to the Files tab and open the folder, "Week-5", you just created

- Press the three horizontal dots highlighted in the Figure below
- Browse and select "Week-5" folder that you created in the previous step, inside "NM2207" folder



Navigating folders

4. Set it as your working directory (shown in the Figure below)



Set as working directory

5. Now, create a new project and name it "Week-5"

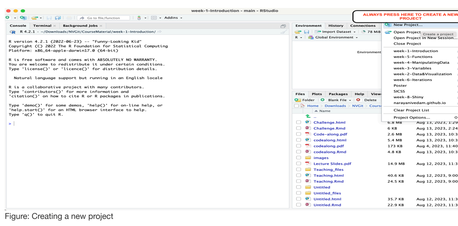


Figure: Creating a new project

6. Download the `Code-along.Rmd` file from Canvas and move it to the folder, "Week-5"

II. Code to edit and execute using the Code-along.Rmd file

A. Writing a function

1. Write a function to print a "Hello" message (Slide #14)

```
# Enter code here
```

2. Function call with different input names (Slide #15)

```
# Enter code here
```

3. typeof primitive functions (Slide #16)

```
# Enter code here
```

4. typeof user-defined functions (Slide #17)

```
# Enter code here
```

5. Function to calculate mean of a sample (Slide #19)

```
# Enter code here
```

6. Test your function (Slide #22)

```
# With one input
```

```
# With vector input
```

7. Customizing the function to suit input (Slide #23)

```
# Enter code here
```

8. Setting defaults (Slide #25)

```
# First define the function

# Call the function
```

9. Different input combinations (Slide #26)

```
# Enter code here
```

10. Different input combinations (Slide #27)

```
# set error=TRUE to see the error message in the output
# Enter code here
```

11. Some more examples (Slide #28)

```
# Enter code here
```

B. Scoping

12. Multiple assignment of z (Slide #36)

```
# Enter code here
```

13. Multiple assignment of z (Slide #37)

```
# Enter code here
```