**EDUC 641 Lab: Applied Statistics in Education and Human Services I**

**Lab 1:** 9/27 and 9/28

*Setting up R/RStudio and using R script*

**Goals**:

1. Install R and RStudio
2. Identify the four panes in RStudio
3. Write and run code through RStudio Console
4. Write and run code through R Script
5. Set-up an R project

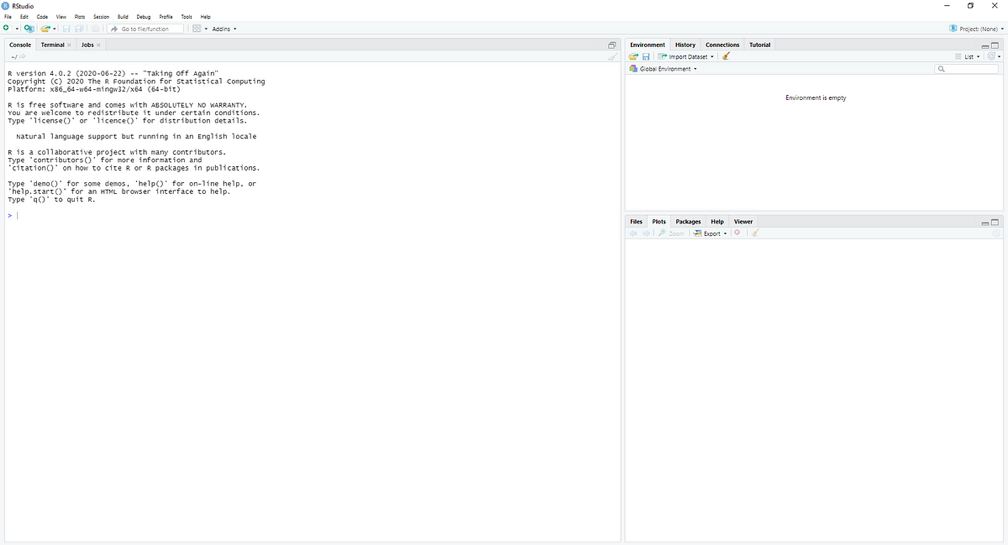
**Remember to:**

1. Work in groups and discuss with your peers as you go.
2. Write down your questions as and when they arise.
3. Call for assistance.
4. Sit near the GE.

**Materials aligned with the lab:** [EDUC R Workshop](https://community.uoregon.edu/courses/18778) - Modules 1 and 2

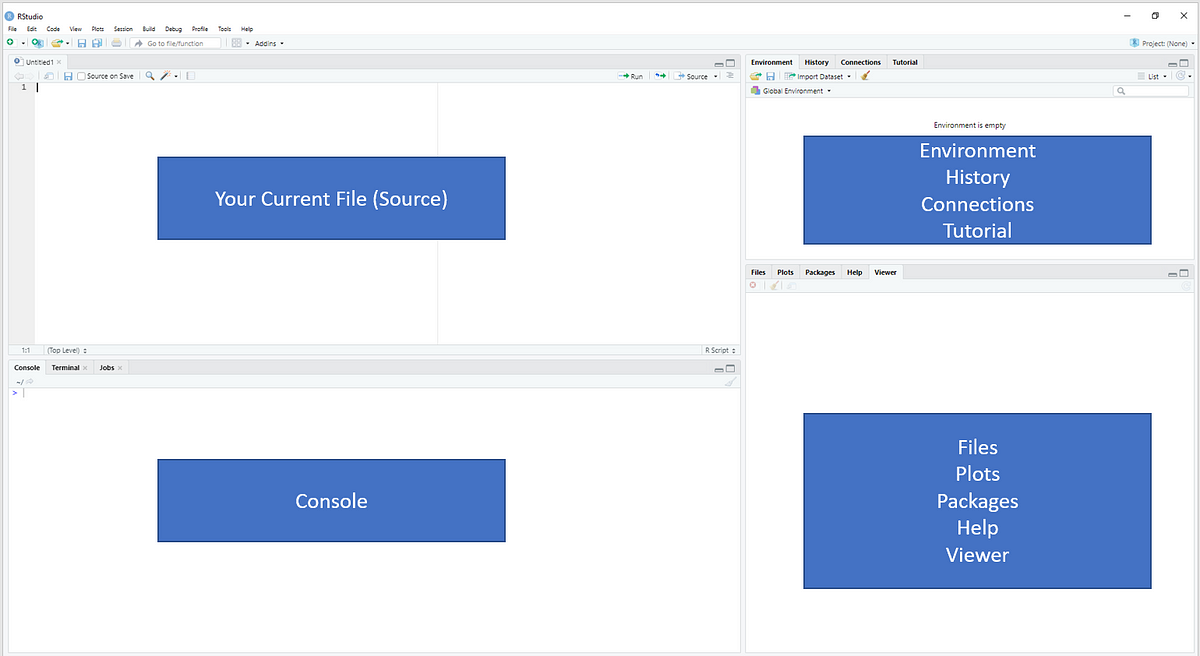
Tasks:

1. Install R and RStudio: R is a programming language we use in the class to conduct statistical analysis. RStudio is the application or graphical interface we use to access R. You need both for coding purposes.
   1. Download R from *here* based on your system: <https://mirror.las.iastate.edu/CRAN/>
   2. Once downloaded, open the downloaded file to install it on your device. Keep the default options.
   3. Download RStudio from here based on your system: <https://posit.co/download/rstudio-desktop/>.
   4. Again, after the download has been completed, open the downloaded file to install it on your device. Keep the default options.
   5. Once both are downloaded, open RStudio from the applications. If you see something like what is shown below, go on to task 2.



Note: R and RStudio keep frequent updates (on an annual basis). You should re-install them to have the most up-to-date version.

1. RStudio: RStudio is an interface to use the R programming language. You would see four panes once you open RStudio.



These are code editor (or Source; left-top), R console (left-bottom), environment and history (right-top), files and plots viewer (right-bottom). Today, we will focus on the code editor and the console. We will cover the others in the next few weeks. You can also customize the look of RStudio per your taste. For the next 5 minutes, explore the Appearance customization options available for RStudio and choose per your liking.

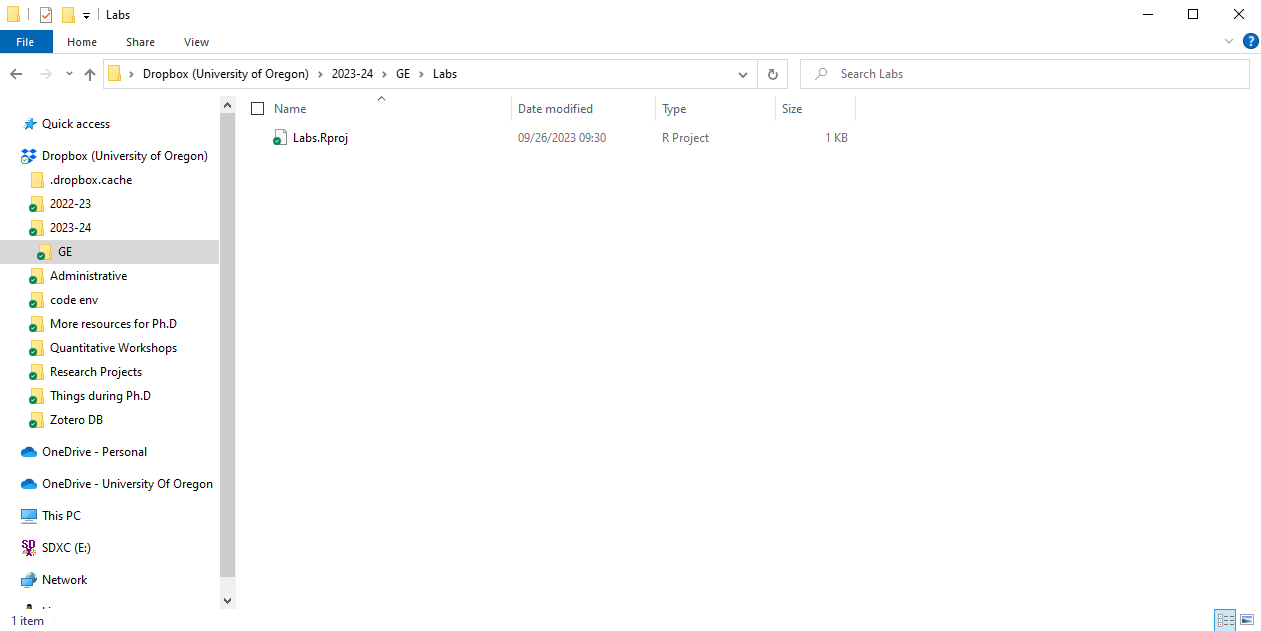
* 1. Select Tools on the menu bar. Then click, Global Options.
  2. A new pop-up window will come. From the menu on the left side, click on Appearance to change the background color, font, etc. of your RStudio interface.
  3. In the General tab, unselect the ‘Restore .RData into workspace at Startup’
  4. There are many more customizations available that you are encouraged to explore on your own time

1. RStudio Console: The console is where you can type code that executes immediately. Just type and press Enter. Type the following code and write what you see below. The output will be in the console.
   1. print(“hello world!”)
   2. 12345\*54321
   3. 4^3
   4. 1:7
   5. obj <- seq(from = 1, to = 5, by = 0.5)
   6. obj
   7. mean(obj)
   8. ?mean
   9. length(obj)
   10. #length(obj)
   11. plot(obj)
   12. install.packages(“tidyverse”)

Using the above code, you have used objects, assignment operator, functions, help, comments, and packages in R. Here is a brief about these terminologies:

* Objects: A named value that holds data, for instance, the object ‘obj.’ Object names should not start with a number or have spaces.
* Assignment operator: The symbol <- (less than followed by minus sign) to save data to an object.
* Functions: A command that executes something, for instance, the ‘mean()’ function. Inside the function, you must pass an object (also called argument). Therefore, we wrote ‘mean(obj).’
* Help: Typing question mark (?) followed by function name opens the help page in the bottom-right pane. For instance, ?mean
* Comment: Adding hashtag (#) sign before a code is a comment and is not executed. For instance, #length(obj)
* Packages: Installing and using packages makes specialized functions available to use.

1. R Script: One downside of executing code in console is that you are unable to keep track of what you have done. To overcome this, we can use R Script (or R Markdown).
   1. To open an R Script, go to File. Then click on New File, and then on R Script. A code editor would have opened in the top-left panel (source). You can write code on the editor and save it on your device to be used later.
   2. To run a code, go anywhere on the line where you wrote the code and either click ‘Run’ using the mouse, or press Ctr/Command + Enter on your keyboard. The output will be printed on the console.
   3. Try the following on the R Script
      1. Make a new object ‘obj2’ and assign it the value ‘c(1,3,4,7)’. ‘c()’ is the combine function.
      2. Use the function ‘sum()’ on obj2.
      3. Look at the help page for the ‘sum’ function.
      4. Write a comment in the R Script about what the ‘sum()’ function does.
      5. Save it on your device by selecting File, and then Save.
2. R project: *We will do this if time allows*. R project is a way to organize your work on the device. It is a collection of documents, workspace, history that has, a) its own folder on your computer, and b) can be loaded to RStudio by opening the file with the ".Proj" extension. (Havi will demonstrate how to make an R project for the labs on her device. You should see here, and then try on your own.)
   1. First, decide where on your device you want to have a folder for the project (here, labs).
   2. Then, from the RStudio application, click on Files -> New Project -> New Directory -> New Project.
   3. Click on ‘browse’ and go to the location on your device where you want to make a new folder for the project. Note that a new folder will be made inside the location you choose.
   4. Then, give a name to the Directory under the ‘Directory name:’ and click ‘Create Project.’
   5. Now go to the location on your device and check if a new folder is created there. Also, check what is there inside the new folder. Do you see something like this? If yes, you have successfully made an R project.



* 1. Inside the Labs folder, now make the following new folders by left clicking on the mouse and selecting New Folder: "code", "data", "figure", "table". Now, you can save all the work related to the labs in this folder and in the respective folder.
  2. Try yourself: Make two more R projects for class code and assignments.

Note: If you realize that you created an R project in the wrong directory, it is easier to delete it and then just go through the process again for the correct directory. We will talk more about the benefits of making an R project next week when we import data and export results.