IB 120/201 - Lab 1
R & Python Basics

Due Date: January 29, 2021

University of California, Berkeley *GSI: Ksenia Arzumanova*

In this lab, we will go through how to install and run python and R environments on your computers. We will also write our first program by printing a message to the command prompt in order to verify a successful installation.

Background

R & RStudio

R is a programming language that is well-suited for statistical analysis and working with large datasets. The best way to use R is in an integrated development environment (IDE) called Rstudio, where one could simultaneously view and control the code, command prompt, variables, directory, and figures.

Python

Python is an all-purpose programming language that is suited for a variety of tasks and functionalities (e.g. machine learning to applications with graphical user interfaces). It is known for its intuitive syntax, simplicity, and open-source support. There are many environments and methods possible to run Python code, and we will first learn how to execute Python code by familiarizing oneself with the command line. A decent interactive development environment for Python is PyTorch, among many others. It provides an iPython interactive command line, meaning that one can execute code with Python syntax through the console, analogous to RStudio.

JupyterHub

Nowadays, most data scientists write python code in a notebook, meaning that there are chunks of code executed in a chronological order and output is presented right below the chunk of code. The advantage of notebooks is that debugging becomes much easier and it allows for easier dissection and understanding of one's workflow. Typically, these are called Jupyter notebooks because it was originally developed an organization called Project Jupyter and the code is run on a server that is either instantiated locally or located remotely. For an example, see this link.

Terminal

The terminal is an interface in which you can type and execute text based commands. It can be much faster to complete some tasks using a terminal than with graphical applications and menus. Another benefit is allowing access to many more commands and scripts. Recall some of the basic functions that were shown at the beginning of lab: 'cd', 'ls', and 'cat'. 'cd', short for 'change directory' allows you to navigate your local or remote computer. You can change the directory one folder at a time, or list several subdirectories, separating them with a '/'. 'ls', abbreviated for 'list', shows you all files, folders, and

applications that exist within your current directory. If you don't want to change your directory and simply look at the contents of a particular directory, you can use this command, followed by the name of the directory, to see what's inside. Finally, 'cat', short for 'concatenate', allows you to read text files without having to open them.

Assignment

Python

1. Open a new text file, write `print('Hello World')`, and save as `first_script.py`.
2. Open a terminal window.
3. Navigate your command prompt to the directory containing the file and write `python3 first_script.py`.
4. Consider the following lists: $a = [1, 2, 3, 4, 5], b = [5, 6, 7, 8, 9]$. How could you create the list $[1, 2, 3, 4, 5, 6, 7, 8, 9]$? (Notice the '5' is not repeated).
5. You have a list of gene names and their IDs:
   "FOS1", 92231
   "JUN", 2313
   "BERP", 5641
   Given a set of gene IDs, how could you easily pull out which gene names correspond to which IDs?

Please submit a screenshot of the terminal, printing "HelloWorld" through running the python script. Create a notebook for the rest of the questions, download the notebook in 'Notebook (.ipynb)' format, and submit the file to bCourses.