Project 0: Computing Environment Setup

Over the course of the semester, you will work with a variety of software packages, including Python Pandas, Jupyter Notebook, Spark, and others. Installing those packages and getting started can often be a hassle because of software dependencies. You have two choices.

- Install the different software packages on your own machine (most of these packages should have tutorials to install them on different OSs). If you have a Linux box or a Mac, this should be possible; it may be more difficult with Windows. In any case, although we will try our best, we would likely not be able to help you with any problems.
- **(Preferred Option)** Use Docker (as discussed below). If you have a reasonably modern machine (within last 3-4 years), this should generally work fine, but with older laptops, the performance may not be as good. See below for more details on this.

Git & Github

Git is one of the most widely used version control management systems today, and invaluable when working in a team. GitHub is a web-based hosting service built around git – it supports hosting git repositories, user management, etc. There are other similar services, e.g., bitbucket.

We will use GitHub to post the final project. Our use of git/github for the class will be minimal; however, we encourage you to use it for collaboration for your class projects, or for other classes.

Setting up a GitHub Account

Repositories hosted on github for free accounts are public; however, you can easily sign up for an educational account which allows you to host 5 private repositories. More details at: `https://education.github.com/`

- Create an account on Github: `https://github.com`
- Generate and associate an SSH key with your account
  - Instructions to generate SSH Keys: `https://help.github.com/articles/generating-ssh-keys#platform-linux`
    - Make sure to remember the passphrase
  - Go to Profile: `https://github.com/settings/profile`, and SSH Keys (or directly: `https://github.com/settings/ssh`)
  - Add SSH Key
- To clone a repository
  - InTerminal: 'git clone git@github.com:<git directory>/<repository>.git'
  - The master branch should be checked out in a new directory.
- Familiarize yourself with the basic git commands
  - At a minimum, you would need to know: 'clone', 'add', 'commit', 'push', 'pull', 'status'
  - But you should also be familiar with how to use **branches**

Docker

Docker is a software technology providing 'containers', that provides an additional layer of abstraction and automation of operating-system-level 'virtualization' on Windows and Linux. Here is a nice introductory blog post that describes virtualization and containers: `https://medium.freecodecamp.org/a-beginner-friendly-introduction-to-containers-vms-and-docker-79a9e3e119b`. Briefly speaking, a virtual machine (VM) is an emulation of an ('guest') operating system on a computer (with potentially a different 'host' operating system). Linux is the most common guest OS that is used, especially since Apple and Microsoft make it difficult to emulate their OSs. 'Containers' look like a VM, but share the host kernel (if possible) to be more efficient, both in terms of the memory used and the slowdown.

Docker is perhaps the most popular container technology at this time, and is widely used to package and ship applications. There are many pre-existing 'images' that can be pulled from `https://hub.docker.com/`(or elsewhere) to quickly install and run different software (as you will see below).

- To get started, install Docker by following the instructions on the webpage for your machine: `https://docs.docker.com/engine/installation/`. We suggest the Stable Channel of the Community Edition.
- Follow the appropriate 'Getting Started' guide to make sure that Docker is working as expected.
  - Macs: `https://docs.docker.com/docker-for-mac/`
  - Windows: `https://docs.docker.com/docker-for-windows/`
- We will use the 'Jupyter Notebook Data Science Stack' for now. You can start it using the following command in the commandline – replace PWD with the path to the git directory. More detailed description of the image is available at: `https://hub.docker.com/r/jupyter/datascience-notebook/`
  - `docker run -it -v PWD/project0:/home/jovyan/notebooks --rm -p 8888:8888 jupyter/datascience-notebook`
    For example: assume your git directory on Mac is '/Users/yourname/Dropbox/CMSC320/project0'. Run the following code

    ```
    docker run -it -v /Users/yourname/Dropbox/CMSC320/project0:/home/jovyan/notebooks
    --rm -p 8888:8888 jupyter/datascience-notebook
    ```

    On Windows:

    ```
    docker run -it -v C://Users//yourname//CMSC320//project0:/home/jovyan/notebooks
    --rm -p 8888:8888 jupyter/datascience-notebook
    ```
- Quick explanation of the above command (don't worry if you don't follow this right now):
  - '-p 8888:8888' maps the 8888 port on the host OS to the 8888 port on the guest container. So if you were to go to http://localhost:8888, it will redirect to the 8888 port on the container - Jupyter Notebook starts a web server on that port on the guest.

  - '-v PWD/project0:/home/jovyan/notebooks' mounts the current project0 directory on the guest, so that everything in project0 directry will be available in 'notebooks' directory on the guest.
  - 'jupyter/datascience-notebook' tells docker which image to pull from the Docker Hub. The first time you do this, it will take a few minutes to download everything it needs.

- Once everything is initialized and the notebook starts, you can connect it to by opening your web browser and going to:
  http://localhost:8888/tree?token=279fb5e0fc0f240a90f913e7b9c9c068f36543a7d9544663 — the 'token' will be different for you. Look for it in the output of the command above.

Python and Jupyter/IPython

We will be using Python for most of the assignments. Python is easy to pick up, and we will also provide skeleton code for most of the assignments.

IPython is an enhanced command shell for Python, that offers enhanced introspection, rich media, additional shell syntax, tab completion, and rich history.

Jupyter/IPython Notebook started as a web browser-based interface to IPython, and proved especially popular with Data Scientists. A few years ago, the Notebook functionality was forked off as a separate project, called `http://jupyter.org/`. Jupyter provides support for many other languages in addition to Python. Several other projects have been started in the recent years, inspired by the idea of Notebooks, e.g., Zeppelin.

Use the command listed above to start a docker container with Jupyter. On your local browser, go to: `http://localhost:8888` (which requires you to enter the token).

- You should see the Notebooks in the notebooks/ directory (or any other directory where you save the code). Click to open the "Jupyter Getting Started" Notebook, and follow the instruction therein.

- You should play with the Notebook to try out different Python commands. You can try creating a new notebook.

Assignment

Complete the function in '*Project 0 Assignment*' Notebook, and upload the '*Project 0 Assignment.ipynb*' file to ELMS. In addition, export this notebook as a pdf file and upload it to Gradescope.