Accessing the Server

Please do the following:

(1) Log into the server with the additional port forwarding option.
(I am assuming this is 8888, you can try 8889 if it does not work and you see 8889 to be used by default)
ssh username@IP_address -L 8888:localhost:8888

(2) Key in your password and you are logged into the server

(3) **module load anaconda** **(or) module load miniconda/3**

(4) **conda-activate**

(5) **ipython notebook --no-browser**

(6) Copy the URL on your browser on your browser (or) press command key and click on the URL to open in browser
It should be something like :
http://localhost:8888/token=vd4322523gdghdhdfhhdedfbvdbd

(7) You will see the usual Jupyter interface. Go to File > New > Python 3 Notebook.

There you go.

Using the data

You should even be able to browse the data from this page as it will list **data** and **submission** directories.

If you do not have an explicit **data** directory, you should have at least have a **data.zip** file that you can unzip and extract the data.

You can **cd** or **ls** the data directory to explore what is in there and how it is structured.

The training data is inside **Train** and **Train/Profile** is relevant to this assignment. You can try shell commands to peek into the data.

You could use the path of the data to load it as a Pandas dataframe or a numpy array.

Pandas : Tutorial Documentation
NumPy : Query Tutorial Documentation

You could go ahead and setup the pipeline for submission using your own linux command.

Submitting the code

To submit your code on the server, you need to submit it as a shell script called "ift6758".

The script should get the same arguments as your python code. Assume your software is a python file called: **myPredictor.py** which get two arguments "-i" for the path to the test folder, and "-o" which specifies the path to the results.

If you are not familiar with command line arguments in python, take a look at: https://www.tutorialspoint.com/python/python_command_line_arguments.htm

You shell script should call your python code, in addition to all the packages that you load to run your software. If you don't know how to write a shell script, this blog post may help: https://humairahmed.com/blog/?p=7579
