Final Project - Algorithm Submission

To aid readability in these guidelines, all filenames are written in Red, and all code or variables are written in Blue.

The leaderboard competition ends 4/22, at 10:00pm. By this deadline, you need to submit your final, trained algorithm onto Canvas. We will be running this algorithm on a hidden test set. Your performance on the hidden test set will be used to determine the top 3 groups. Remember that you also have to submit your final report by 4/27 at 10:00pm. For instructions on the final report, look at BE21_Final_Proj_Part_2.pdf, which you should have already read through to do the leaderboard competition.

Here are some guidelines on how and what to submit to Canvas. PLEASE READ THESE INSTRUCTIONS CAREFULLY:

1. Submit your algorithm as a file. Choose the most applicable file type (pickle, hdf5, etc…). For example, if you are using the linear R matrix approach, then you would pickle your final matrix. If you are using a model from sklearn, then pickle the trained/fitted model. If you're using a neural network, then whatever package you're using probably has a function for saving the model.
    a. If your algorithm requires multiple files, submit them all as a .zip file.
    b. If your algorithm is very large, upload it to google drive and include a link in the accompanying code (see below) where we can download it.
    c. Your algorithm submission should also include whatever preprocessing data/parameters you need..
        i. For example, if you normalize your features by mean and standard deviation, include the mean and standard deviation, either as an additional file, or hard code it into the code (see below).
    d. Tell us what model you're using. That way, if something goes wrong with loading it, we have a chance to debug it.
2. Submit accompanying code (.ipynb) to load and apply the model on the hidden test set. We will be running your submitted code on Google Colab. YOUR CODE SHOULD NOT PERFORM ANY TRAINING.
    a. Your code should output a .mat file formatted exactly like your leaderboard submissions. CALL YOUR PREDICTION VARIABLE predicted_dg AS BEFORE.
    b. Your code must load or install all necessary packages on its own.
        i. We will be using the default Colab python distribution. If you are using a package that is not normally on that, your code must be able to install all necessary packages on its own (you can use !pip install statements in colab cells).
    c. You can assume that the hidden test set will be a .mat file called truetest_data.mat. Thus, include a cell that loads this file and the data within.
        i. truetest_data.mat has a single variable inside called truetest_data.
        ii. truetest_data is formatted nearly-identically to leaderboard_ecog from your leaderboard_data.mat file used in the leaderboard competition. Thus, it is a 3x1 cell array, where each element is a patient. The patients and their orders are the

same as before. Thus, patient 1 will be a Nx62 array, patient 2 will be Nx48, and patient 3 will be Nx64. **N from truetest_data will not be the same as N from leaderboard_ecog**.

    iii. You likely will be able to copy/modify your existing code for loading leaderboard_ecog for this purpose.

d. Your code should load the hidden test set, and load and apply your algorithm, including all steps necessary for pre-processing, filtering, generating predictions, and post-processing. You likely will be able to copy and paste whatever code you're using to generate the leaderboard predictions into your submitted code. We should be able to generate predictions from your code by just running it (after uploading all necessary files).

    i. If your algorithm has multiple files and is in a zip file, your code should be able to unzip it for us.

    ii. Do not include the code you wrote to train your algorithm, or to save it to a file.

e. Include instructions at the very top of the notebook (as a text cell, or a commented-out code cell; just make it readable) on how to use it. We will be following those instructions on Google Colab. You can assume that we already have uploaded truetest_data.mat. Here's an example set of instructions for someone using the R matrix as their algorithm.

    i. Upload our final matrix file (algorithm.pkl) to google colab.

    ii. Press the run button. The notebook should do the rest.

    iii. The code outputs predictions.mat, which contains a variable predicted_dg, which is our predictions on the hidden test set.