NOTE: Below is the (incomplete) log of progress notes used at the beginning of the project to record early thoughts at the beginning of the project, note questions that come up and their answers, summarize progress, and plan for future goals

Saturday, 11/16/2024

Today we briefly read through the paper once again and began looking through the dataset's .txt files and README file to import the data.

It took awhile to understand exactly what the data they included contained and how we could import it.

Essentially, they use a gyroscope and accelerometer of a smartphone for 30 participants. These two sensors generate time signal data for angular velocity and linear acceleration in the x, y, and z directions. Deriving both gives the angular acceleration and the jerk in the x, y, and z directions. Furthermore, taking the magnitudes and also converting the signals to the frequency domain ultimately results in 33 signals in the time domain and frequency domain which could be analyzed.

The researchers derived 17 statistics for each of the 33 signals results in 561 features (which is the total number of features for determining the multi-class output response of HAR (walking-1, walking upstairs-2, walking downstairs-3, sitting-4, standing-5, laying down-6).

QUESTION: how does the windowing relate to the number of datapoints created for each test subject? We see that there are 128 windows created for each test subject (windows of time which overlap with one another). There appear to be around 347 data points per subject. Where are the other data points coming from? How is the frequency domain being divided up to generate statistics?

Regardless, we have been able to understand how the data is distributed across the text files, import them into R, and then process them such that we have our output variable "Activity", and our Xs (subject, and the other 561 features via the statistics described in the paper).

So,

MISSION 0: Read papers, choose topic [COMPLETED]
MISSION 1: Import Data, Understand How ML is employed on it [COMPLETED]

After this, began running the svm() on the data according to how the paper describes their process; a support vector machine on a multi-class variable using a One-vs-All method. This method achieved them 96% accuracy and we're seeking to replicate that.

We were able to run the svm, BUT they mention in the paper that they used 10 fold cross validation to find the best cost and gamma parameters for the SVM (they do not mention the value of these parameters 🙁).

In trying to do this tuning process (using the tune() (or tune.svm()) command in R) to do the cross validation), it's taking WAY too long. Perhaps our code is sub-optimal? Perhaps we're supposed to wait a very long time for it to execute? Perhaps there's a way of dedicating more processing power to the execution given our Task Manager's show that the computer's processing power isn't being maxxed out? Finally, perhaps there's another computer that's more powerful which we could use?

Emailed the TA and maybe go to OH on Tuesday at 10am…

MISSION 2: Apply Paper's ML Methods to data and replicate results [IN PROGRESS]

Future goal:
MISSION 3: Apply our own ML Methods to data, explore, possibly improve upon results

- More greatly folded cross validation?
- Go into detail on what statistics reliably predict the behavior (not mentioned in paper)? They just mentioned the accuracy. Can you investigate that for an SVM? TBD
- Apply other methodologies (foresting?)

MISSION 4: Write Paper
MISSION 5: Put together PPT and Present (either the early date for E.C. or final date)

Monday, 11/18/2024

Met around 7:30→worked till 12am. Stefan figured out parallel processing (awesome), and converged on some gamma and cost values (takes hours but done).

However, these processes were with a default svm setting of one vs one and we're trying to replicate one vs all (one vs one will still be important for comparison's sake.

So.. **MISSION 2** is still in progress but nearly done, and **MISSION 3** is started as we've already began trying out different processes on the data.

He and I are both working on one vs all SVM, I'm doing it repetitively (not in a for loop, just repeating a command), he's doing it in a for loop. For loop would be better but I didn't readily see (for myself) how to do it.

Regardless of all this, good progress has been made. (NOTE: when "duplicate" error comes up, ensure that you're using colnames with features$V1.

Useful link:
https://stackoverflow.com/questions/27125381/implement-multi-class-classification-using-svm-in-r

Friday, 12/07/2024

Finished implementing the svm and replicating the paper's results. Got the confusion matrix and the paper's same accuracy.

Saturday, 12/07/2024

Now that we've replicated the paper's findings, time to make improvements/work of our own. Some ideas:

1) Implement a tree based search → trees are less computationally taxing I believe compared to svm and usually more interpretable
2) Previously to determine the ideal cost and gamma values for the svm we used a combination of grid and random search with the 10-CV w/ gaussian kernel (NOTE: the paper doesn’t specify what kind of search they used). Today we’ll use bayesian search which ‘learns’ where to go from previously found values.
3) Split data randomly by 80-20 instead of 70-30
    a) Change which participants are doing this

Q: their dataset is split by people, but then CV on the training I imagine doesn’t split along the lines of what person did what→ do we need to split on the basis of people?