Graph Analysis Summary

Daniel Qian, Nathan Vallapureddy, Jenny Wu

January 25, 2018

1 Science In The Cloud

1.1 Improvements

On one machine, we could not access scienceinthe.cloud docker at all. While it could've been a problem with docker rather than scienceinthe.cloud, it would've been useful to have some troubleshooting help on the website. A second problem is that we couldn't access the full container with the commands given on the website, and we had to debug on our own.

1.2 Things that went well

Once we had the docker demo running, it was very easy to use. In addition, we like the UI/UX, especially the simplicity of the demo. Finally, we like how it presents data and graphs that do a good job of summarizing the connectomes in a simple way.

2 Viewing Connectome

To view the connectome, we computed the mean connectome matrices for both males and females, as well as both classes together. To distinguish between males and females, we created an array for male matrices and an array for female matrices. We used the Matplotlib implot function to plot the connectome as a color-coded matrix, which allows us to see which areas of the brain are highly connected.

3 Data manipulation

In the middle of adding features and troubleshooting the program, we noticed that the processing of the data was incorrect because it was misaligned. This was an important lesson in data science: we needed to make sure our data was good before trying to manipulate it. The correction of the code added almost 0.10 to our initial accuracy, which was an incredibly significant change. When

we added more complex features, the change was magnified even more, since incorrect data couldn't possibly be highly accurate.

4 Features

4.1 Closeness Centrality

The first feature we decided to keep was the $nx.closeness_centrality$, which computes the reciprocal of the sum of the shortest path distances from one node to the others (higher values of closeness indicate a higher centrality). Centrality ended up being a strong distinguishing feature for us to use because it is designed specifically for graph analysis.

4.2 Betweenness Centrality

second feature, which functions similarly, is $nx.betweenness_centrality$, which factors in the shortest elements that pass through a node, or betweenness.

4.3 Edges

Our third feature factors is simply each entry in the matrix. This is our strongest covariate. For each matrix, this appends all 4900 connection strengths. This reflects which areas of the brain are connected to all of the other areas.

4.4 Matrix

Our fourth feature calculates the log of the sum of every row, which works as a function of connection strength. This tells us whether certain neurons are particularly strong in a certain class.

4.5 Clustering

Our fifth feature is $nx.average_clustering$, which computes the average clustering of nodes. This tells us whether certain classes are more clustered or more spread out.

4.6 Mean

Finally, our last feature calculates the mean of each value in the matrix; though a simple statistical method, it was undoubtedly effective as a classifier. This tells us if a certain class is more connected than the other.

5 Results

Using the Random Forest Classifier, and using 3 of the covariates, we reached 67.74% accuracy on the Desikan dataset. With Naive Bayes, we reached 66% accuracy on the same.