

1 Introduction

The goal is to perform an exploratory study of structural brain connectomes and identify useful insights that can help in understanding the relationship between connectomic data and cognitive diseases (e.g autism). In particular, I aim to visualize connectomic data as graph networks so that I can collect explainable metrics to formalize my insights. As a byproduct, I believe I will gain knowledge within the field of graph theory and some of its major relevant algorithms.

Before I could even begin to perform analyses or conduct experiments, I needed to obtain structural brain connectomes. As such, I referred to the BrainGraph.org connectomic dataset which was generated by Kerepesi et al [1]. Once I got hold of the structural connectomes, the next natural step was to visualize them. Matrices and graph networks are two popular visualization tools that are used to represent structural brain connectomes.

At the moment, I have been able to access and also visualize the connectomes available on BrainGraph.org in two different formats (as matrices and graph networks). Additionally, I was able to gather the basic centrality metrics and apply the Louvain algorithm to cluster the network.

2 Visualization

For the purposes of my analysis, I picked two connectomes to visualize: (1) an averaged connectome, (2) a single subject's connectome.

The connectome matrices were just visualized as heatmaps that correspond to their respective adjacency matrices. For the graph networks, I focused on visualizing them in an aesthetic way so that structural and relational properties are easily visible. In particular, the graphml data files available on BrainGraph.org included attributes for each node that pertained to their 3D spatial locations as well as attributes dictating the edge weights (FA mean) between nodes. Using the spatial attributes and NetworkX layout functionalities, I was able to represent the connectomic data as two and three dimensional networks (plotted via Matplotlib). In addition, using the FA mean, I was able to create a color map to encode the strength of relationships (or edges) between nodes.

To visualize the clustering performed by the Louvain algorithm, I simply colored the nodes in the network based on the cluster they belonged to. Nodes of the same color belong to the same cluster.

2.1 Connectome as a Matrix

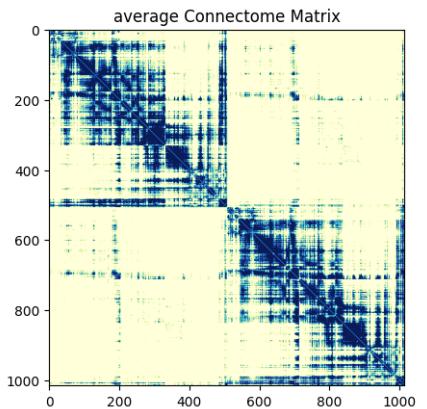


Figure 1: Averaged consensus connectome matrix across all subjects

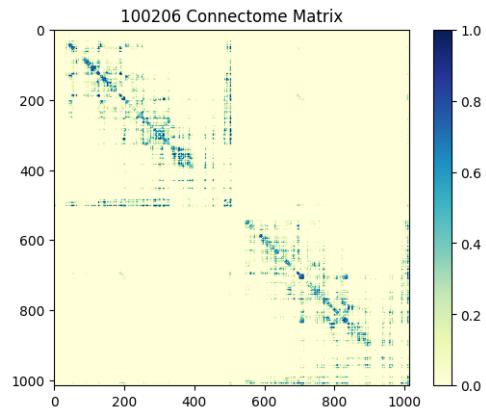


Figure 2: Connectome matrix corresponding to a single subject (ID = 100206)

The figure to the left is the averaged consensus connectome matrix across all subjects in the dataset. The figure to the right is the connectome matrix for a single subject with an id of 100206. If two nodes

in the connectome have an edge, the corresponding cell in the matrix will be colored blue.

2.2 Connectome as Graph Network

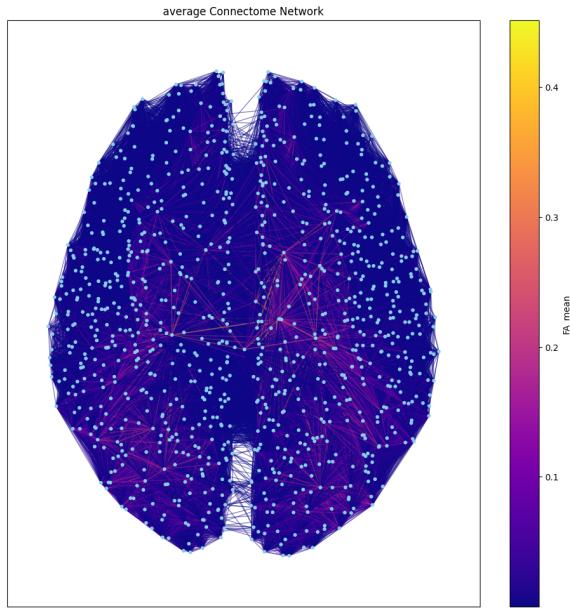


Figure 3: Averaged consensus connectome network across all subjects

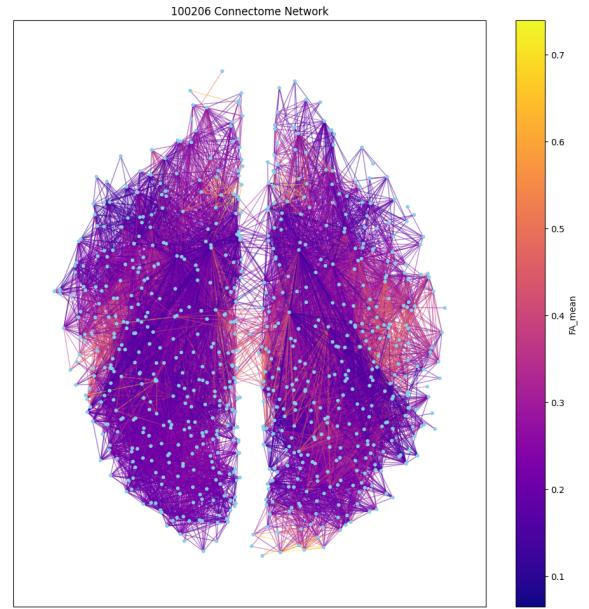


Figure 4: Connectome network corresponding to a single subject (ID = 100206)

The figure to the left is the averaged consensus connectome graph network across all subjects in the dataset. The figure to the right is the connectome graph network for a single subject with an id of 100206. Note that for the network representation, edge weights are analogous to the mean fractional anisotropy between two nodes (or the index of the anisotropy of a white matter tract between two brain regions). Dark blue indicates low fractional anisotropy whereas bright yellow (warmer colors) is indicative of high fractional anisotropy.

3 Network Analysis

In graph theory there is a wide assortment of metrics that can be obtained from a network. One such family of metrics are centrality metrics. In my evaluation, I mainly focused on degree centrality, closeness centrality, betweenness centrality, and eigenvector centrality.

Clustering and identifying communities is another useful analysis. I have used the Louvain algorithm to recognize and separate clusters within the network.

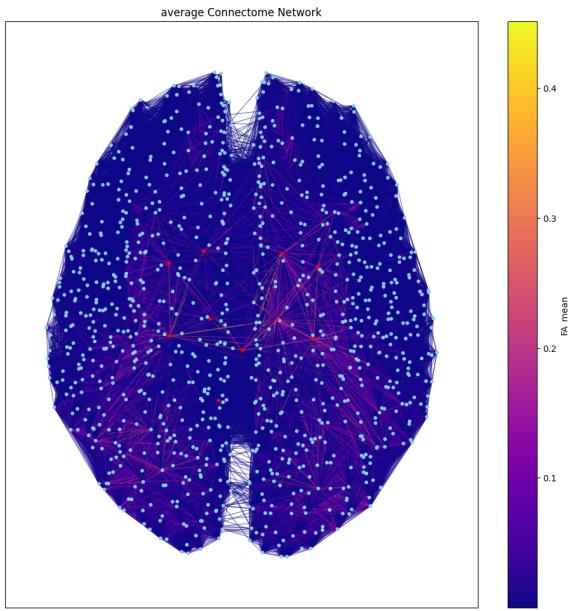


Figure 5: Averaged consensus connectome network across all subjects

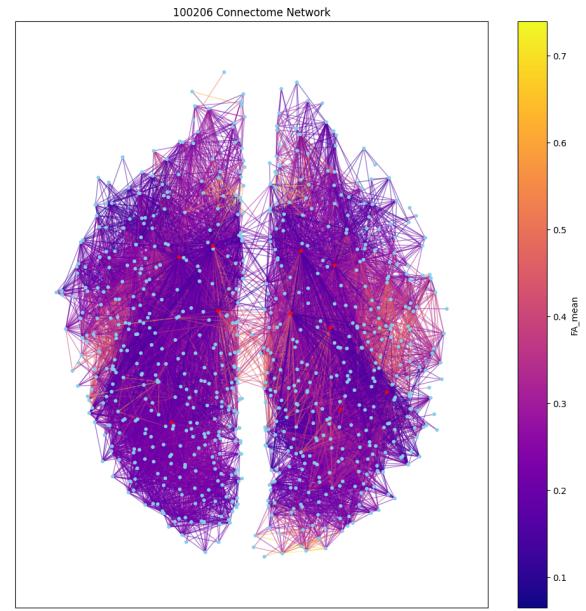


Figure 6: Connectome network corresponding to a single subject (ID = 100206)

Similar to before, the diagram on the left corresponds to the averaged consensus connectome while the one on the right corresponds to a single subject connectome with an id of 100206. While it may be hard to see, I have highlighted in red the top ten nodes that have the highest betweenness centrality. For conciseness, assume that the other mentioned centrality measures are visualized the same way.

3.1 Clustering

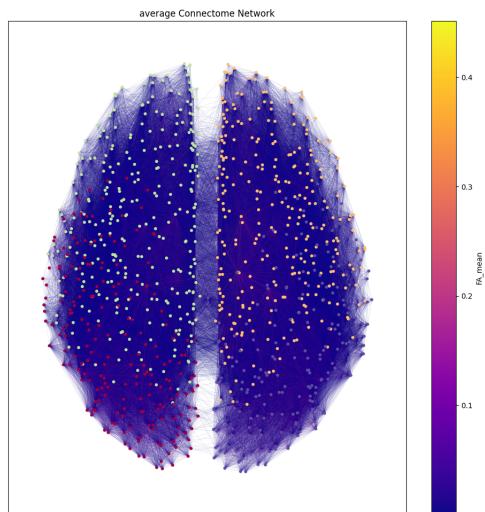


Figure 7: Averaged consensus connectome network across all subjects

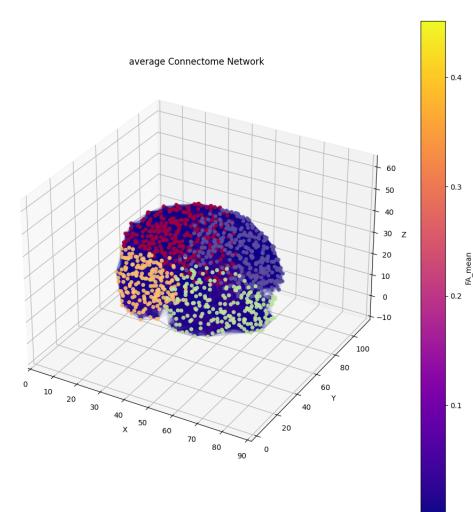


Figure 8: Connectome network corresponding to a single subject (ID = 100206)

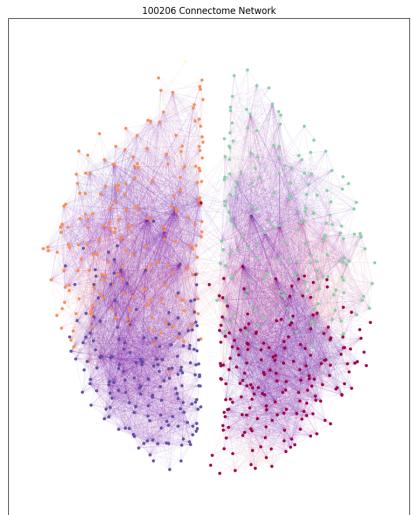


Figure 9: Averaged consensus connectome network across all subjects

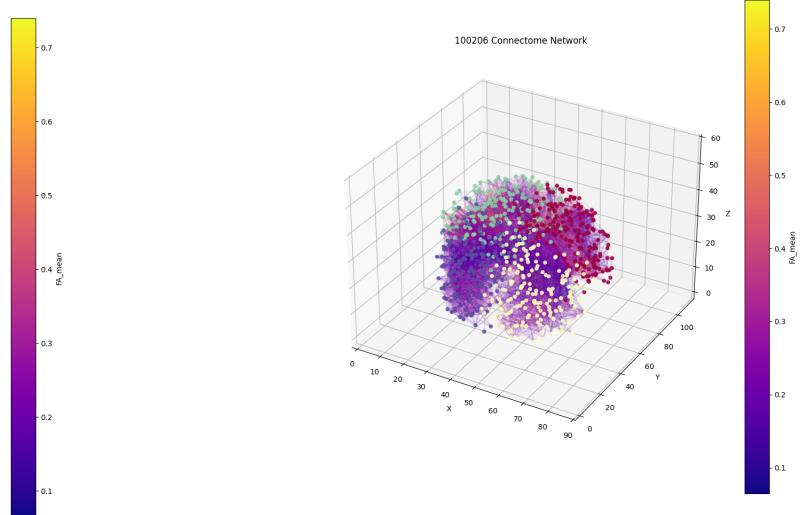


Figure 10: Connectome network corresponding to a single subject (ID = 100206)

The two figures on the top are correspondent to the 2D and 3D clustering visualization of the averaged consensus connectome. The two figures on the bottom are correspondent to the 2D and 3D clustering visualization of a single subject connectome with an id of 100206.

4 Concepts

4.1 Degree Centrality

This is the most basic centrality measure and the easiest to calculate. As the name suggests, for a given node, the degree centrality is simply the degree of that node. The degree of a node is the number of edges that are incident to that node. For a directed graph, this measure is typically split into in-degree centrality and out-degree centrality where in-degree is the number of edges that are incoming and out-degree is the number of edges that are outgoing.

So, for an undirected graph, the degree centrality for a node i is:

$$d_i = \sum_{j=1}^N a_{ij}$$

where A is the adjacency matrix with a_{ij} being an element in the matrix corresponding to the relation between the i th and j th node [3].

4.2 Closeness Centrality

The closeness centrality of a node i measures how "close" the node is to all other nodes in the graph. In particular, it is:

$$c_i = \frac{N - 1}{\sum_{j \neq i} d(i, j)}$$

where N is the number of nodes and j is an arbitrary node (that is not the same as i) in our graph. Intuitively, nodes with higher closeness centrality measures are closer to the "center" of the network.

4.3 Betweenness Centrality

Betweenness centrality is used as a measure to quantify how much of a control a node has during network flow. In other words, we want to check the number of times a node i falls in the path of an arbitrary node j and k :

$$b_i = \frac{2}{(N-1)(N-2)} \sum_{i \neq j, i \neq k, j \neq k} \frac{\sigma_{jk}(i)}{\sigma_{jk}}$$

where N is the number of nodes in the graph, σ_{jk} is the number of paths from j to k , and $\sigma_{jk}(i)$ is the number of times that i falls in their paths [3].

4.4 Eigenvector Centrality

Let v be the eigenvector corresponding to the largest eigenvalue of the adjacency matrix [3]:

$$v = \frac{1}{\lambda} Av$$

wherein which each value of v can be represented as:

$$v_i = \frac{1}{\lambda} \sum_{j=1}^N a_{ij} v_j$$

Influence propagation is a good way to gain some intuition behind why eigenvector centrality can be a powerful measure. A node that has a high eigenvector centrality is one that has influential neighbors. "Influence" refers to a node's own centrality measure (typically this is degree centrality). In other words, a node could have the lowest degree centrality but the highest eigenvector centrality if its neighbors have high degree centrality. You can imagine that this process of analyzing the influence of a node's neighbor can recurse or propagate to neighbors of neighbors and so on.

4.5 Louvain Method for Community Detection [2]

The Louvain method is a popular choice when clustering nodes in large networks because of its speed, convergence rate, and ability to output strong communities with high modularities. The algorithm works in two phases to greedily maximize modularity and has a runtime complexity of $O(n \log n)$. The first phase involves an optimization of the modularity but with the constraint that only local changes can be made where nodes can change community memberships. Note that at the start of the algorithm, each node is its own cluster. The second phase is an aggregation of clusters to super nodes. We can then recursively follow these two phases until the modularities of the communities cannot be increased (e.g reached convergence).

I have used "modularity" a few times and it is worth formalizing; the modularity of a community C can be represented as:

$$\Delta Q(C) = \frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2$$

where \sum_{in} is the sum of edge weights between nodes in C and \sum_{tot} is the sum of edge weights of all nodes in C . Note that the difference is that \sum_{tot} also allows for external edges to be counted.

Using the modularity for a given community C , we can derive the modularity gain when we insert a node i into C :

$$\Delta Q(i \rightarrow C) = \left[\frac{\sum_{in} + k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2 \right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2 \right]$$

where k_i is the sum of all edge weights incident to a node i and $k_{i,in}$ is the sum of all edge weights between i and C .

5 Data and Code

The dataset I used can be found on BrainGraph.org where Kerepesi et al. have collected data of 426 human subjects from the Human Connectome Project (HCP) [1]. They then use this data to construct structural connectomes to finally encode them as graphML files. There are various datafiles that are present, but the ones used in my analysis can be found by downloading the files associated with 1015 nodes (un-augmented) and the single averaged consensus connectome.

The code for this analysis can be found on my GitHub repo: <https://tinyurl.com/4f4ndy92>

6 Future Work

At the moment, I have visualized and gathered metrics for the connectomic data. Through the rest of the study I hope to explore the following research directions:

1. Clustering is great, but how well can identified clusters map to structural brain regions? If they can map well, would there be any advantage in identifying brain regions using clustering methods over typical coordinate to region mapping?
2. Within each cluster, is there some predictable distribution of edge weights? If so, can they be used identify anomalous edge weight distributions (and thereby indicate some form of cognitive issue)?
3. It would be interesting to see if typical social network graph analysis methods could be applied to structural brain connectome networks (e.g nodes in the net correspond to social beings). Consequently, it may be that brain networks closely resemble the qualities of social networks.
4. There are a myriad of generative models, and I would like to train a model so that it can generate a reasonable connectome network given some prior information about the structure. In other words, given sets of nodes and edges pertaining to clusters in a connectome network, can a generative model generate the rest of the network?

7 Relevant Definitions

- **NetworkX**: A popular Python package used for network analysis and visualization (<https://networkx.org/>)
- **Fractional Anisotropy (FA) mean**: Measure of diffusivity along an orientation; derived from Diffusion Tensor Imaging (DTI)
- **Generative models**: A family of machine learning models that find patterns in data and are able to generate new data using these identified patterns.

8 Works Cited

References

- [1] Csaba Kerepesi et al. “The Braingraph.Org Database of High Resolution Structural Connectomes and the Brain Graph Tools”. In: *Cognitive Neurodynamics* 11.5 (Oct. 2017), pp. 483–486. ISSN: 1871-4099. DOI: [10.1007/s11571-017-9445-1](https://doi.org/10.1007/s11571-017-9445-1). (Visited on 10/25/2024).
- [2] Jure Leskovec. *CS 224 Lecture 13: Community Structure in Networks*. <https://snap.stanford.edu/class/cs224w-2020/slides/13-communities.pdf>. Accessed: 2024-10-25. 2021.

- [3] Salma Salhi et al. “Network Analysis of the Human Structural Connectome Including the Brain-stem”. In: *PLOS ONE* 18.4 (Apr. 2023), e0272688. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0272688. (Visited on 10/09/2024).