Katherine Walker

Software Design Document

Summary

The primary objective of this project is to quantify the connectedness of individual brains. This will include improving upon existing quantification algorithms that calculate measures of small-worldness. Small-worldness is a measure to determine whether a given graph is a small-world network. Small-world networks are networks in which most nodes are not neighbors, but paths exist between nodes such that graph traversal is possible in a small number of steps. This project will utilize the concept of small-worldness to determine the level of connection in an individual brain. Existing literature proposes that brains are similar to small-world networks. This project will support that claim or provide evidence to the contrary. Once the brains have been measured, those measurements will be correlated with existing demographic information to determine whether more connected brains have certain properties, possibly higher IQs. To supplement this analysis, a visualization of the connectome will be displayed with darker colors denoting stronger connections. All data for this project comes from the Human Connectome Project.

Users

This project is intended for neurological researchers who want to analyze the relationship between connectomes, which are maps of functional connections between brain regions, and behavioral data. The only user interface is command line using javac Brains.java to compile the project and java Brains to run the project.

Use Cases

This project has only one use case by design. A neurological researcher can input a set of 100 x 100 connectomes, which is the format used by the Human Connectome Project, and a set of behavioral variables in a CSV file. The project will then provide the small-worldness of each connectome, the correlation between small-worldness and each behavioral variable, and a 3D visualization of the connectome.

Software Components

Small-worldness Algorithm

Functional requirements:

- The algorithm must return the small-worldness of the given connectome.
- The algorithm must record each small-worldness in a text file.
- The algorithm must take connectome input from the netmats2.txt file.

Non-functional requirements:

- The algorithm should return the small-worldness within five seconds.

Time-line:

I will begin development in week 3 and finish development in week 13. I will begin testing in week 13 and finish testing in week 17. I will complete integration of this component in week 19.

Correlation Algorithm

Functional requirements:

The algorithm must return the correlation between connectomes and the given behavioral variable(s).

The algorithm must record each correlation in a text file.

The algorithm must take behavioral variable input from the behavioral.csv file.

Non-functional requirements:

The algorithm should return the correlation between connectomes and behavioral variables within fifteen minutes.

Time-line:

I will begin development in week 19 and complete development in week 21. I will begin testing in week 21 and complete testing in week 23. I will complete integration of this component in week 23.

Visualization

Functional requirements:

The visualization component must display a point for every pair of physical regions that have non-zero functional connections in the connectome.

The visualization component must also color the point must according to the magnitude of the functional connection between that pair of physical regions.

Non-functional requirements:

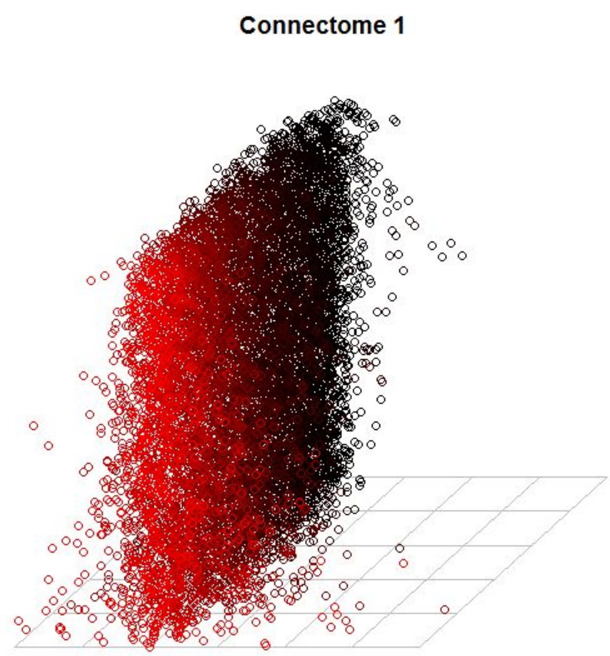
The visualization component must display a connectome within one minute of the user's request.

Time-line:

I will begin development in week 13 and finish development in week 17. I will begin testing in week 17 and finish testing in week 21. I will complete integration of this component in week 23.

Katherine Walker

Visualization Mockup



Information Flow