

Interactive Brain Connectivity Network in D3

Yan Jin

Abstract—The human brain is a complex system whose topological organization can be represented using connectomics. However, the existing brain connectivity network visualization toolboxes are difficult to use, and some toolboxes could not provide interactive operations. In this paper, we implement an interactive brain connectivity network based on D3.js, which could provide more animations and interactive operations so that it is much helpful to the brain connectivity research.



1 INTRODUCTION

The human brain is a complex system whose topological organization can be represented using connectomics. The human connectomes could be constructed using various neuroimaging technologies and further characterized using sophisticated analytic strategies, such as graph theory. On the other hand, in order to visualize brain networks in an easy, flexible and quick manner, there are some existing toolbox, such as BrainNet Viewer [5] in matlab. However, BrainNet Viewer has several drawbacks in below so that it encourage us to develop the new interactive brain connectivity network in D3:

- BrainNet Viewer is based on matlab, which means you have matlab installed before running this toolbox. Our toolbox would be based Javascript that is supported by most modern browsers.
- Although BrainNet Viewer could draw the brain surface, nodes and edges in sequence and displays brain networks in multiple views, it dose not support more insightful interaction, for example, user may want to zoom in and see the detailed interaction of one specific Region of Interest (ROI, served as one node in network).

• Yan Jin is with Industrial Engineering Department at University of Washington, Seattle. Email: yanjin@uw.edu.

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxx/TVCG.201x.xxxxxx

- BrainNet Viewer is powerful, but not easy to be extended. It supports many implemented functionality, but not easy to implement user developed functions. Our toolbox, however, is lightweight, which only focus on the visualization of brain network, instead of including all functions in one toolbox.

2 RELATED WORK

Here is a summary of neuroscience networks tools. Among these toolboxes, the most commonly used one is BrainNet Viewer that is developed on Matlab, with a Graphical User Interface (GUI) to provide a flexible and rapid visualization platform and generate figures for brain connectome studies in a user-friendly and intuitive manner [5]. In this toolbox, the brain surface, node, edge and volume files can be defined as input containing fundamental information about brain networks, and we have designed an easy-to-use optional panel to modify the details of the network display.

Connectome Viewer is another popular toolbox that is based on Python. It provides a GUI with extensive libraries, an interactive scripting shell and built-in script editors with code-highlighting and debugging functionality. Graph analysis libraries are required to unravel the complex brain network organization of structural and functional systems [3].

3 METHODS

3.1 Dataset

The data set is FDG-PET images that is downloaded from the ADNI website. Then we apply Automated Anatomical Labelling (AAL) to extract data

Table 1. Literature Review of Existing Brain Connectivity Network Visualization

Name	Feature	Website
BrainNet Viewer [5]	3D graph-based brain network demonstration with nodes and edges; 3D brain surface view	www.nitrc.org/projects/bnv/
Connectome Viewer [3]	3D graph-based brain network demonstration with nodes and edges; 3D brain surface view	cmtk.org/viewer/
Caret	3D surface view and nodes	About">brainvis.wustl.edu/wiki/index.php/Caret>About
NetworkX	Graph-based network analysis; 2D demonstration with dots and lines	networkx.github.io
Pajek	Graph-based network analysis; 2D demonstration with dots and lines	vlado.fmf.uni-lj.si/pub/networks/pajek/

from each of the 116 anatomical volumes of interest (AVOI), and derived average of each AVOI for each subject. The AVOIs represent different regions of the whole brain. 42 AVOIs are selected for brain connectivity modeling, as they are considered to be potentially related to AD. These regions distribute in the frontal, parietal, occipital, and temporal lobes.

3.2 Graphical Lasso

An inverse covariance matrix can be represented graphically [2]. If used to represent brain connectivity, the nodes are activated brain regions; existence of an arc between two nodes means that the two brain regions are closely related in the brain's functional process [4].

Let $\{X_1, \dots, X_p\}$ be all the brain regions under study. We assume that $\{X_1, \dots, X_p\}$ follows a multivariate Gaussian distribution with mean μ and covariance matrix Σ . Let $\Theta = \Sigma^{-1}$ be the inverse covariance matrix. Suppose we have samples (e.g., n subjects with AD) for these brain regions. Note that we will only illustrate here the SICE for AD, whereas the SICE for MCI and NC can be achieved in a similar way.

We can formulate the SICE into an optimization problem, i.e.,

$$\hat{\Theta} = \arg \max_{\Theta \succ 0} (\log(\det(\Theta)) - \text{tr}(S\Theta) - \lambda \|\text{vec}(\Theta)\|_1)$$

where $\Theta = \Sigma^{-1}$ is the inverse covariance matrix, which is to be estimated; S is the sample covariance matrix; \det , tr , $\|\text{vec}(\cdot)\|_1$ denote the determinant, trace and sum of the absolute values of all elements of the matrix, respectively.

This equation aims to achieve a tradeoff between the likelihood fit of the inverse covariance estimate and the sparsity. The tradeoff is controlled by λ , called the regularization parameter; larger λ will result in more sparse estimate for Θ . The formulation follows the same line of the L_1 -norm regularization, which has been introduced into the least squares formulation to achieve model sparsity and the resulting model is called Lasso. Next, we show that with λ going from small to large, the resulting brain connectivity models have a monotone property. Before introducing the monotone property, the following definitions are needed.

3.3 D3

D3 (or D3.js) is a JavaScript library for visualizing data using web standards. D3 helps you bring data to life using SVG, Canvas and HTML. D3 combines powerful visualization and interaction techniques with a data-driven approach to DOM manipulation, giving you the full capabilities of modern browsers and the freedom to design the right visual interface for your data [1].

D3.js is great for creating interactive network

graphs with JavaScript. The d3Network package makes it easy to create these network graphs from R. The main idea is that you should be able to take an R data frame with information about the relationships between members of a network and create full network graphs with one command. However, this R package cannot provide as many custom operations as the original D3.js could provide.

4 RESULTS

In Figure 1 and 2, the force directed graph structures are drawn to represent the brain connectivity network. It allows you to use individual link and node properties to change the distance between individual nodes and the colour of the nodes depending on their membership in specific groups.

A flexible force-directed graph layout implementation using position Verlet integration to allow simple constraints. For more on physical simulations, see Thomas Jakobsen. This implementation uses a quadtree to accelerate charge interaction using the BarnesHut approximation. In addition to the repulsive charge force, a pseudo-gravity force keeps nodes centered in the visible area and avoids expulsion of disconnected subgraphs, while links are fixed-distance geometric constraints. Additional custom forces and constraints may be applied on the "tick" event, simply by updating the x and y attributes of nodes.

The layout of brain connectivity network is 3D so that I plot three sub-figures to show the different view angles, i.e., x-y, x-z, y-z. In each sub-figure, each node represents one brain region, e.g., Frontal Sup L. Overall, there are 42 individual brain regions. The color encodes that those brain regions in same color are from the same part of brain, e.g., frontal lobe, temporal lobe, etc, and there are four different lobes in total. The pre-specified location of each node could be referred to the known 3 dimensional coordinates of the brain.

The links show the connection between every two nodes. The width of link indicates the strength of the connection, and it comes from the value of estimated sparse inverse covariance matrix.

More specifically, figure 1 shows that when the mouse moves over one of node, then this node and its connected nodes, and corresponding links would be highlighted to make it easier to check the details of the connections. Not only the sub-figure where

you refer to the node would be highlighted, but also other two sub-figures will highlight the relevant brain regions.

Besides that, figure 2 do one more step. When the user click the node, then not only it will highlight the corresponding nodes and links as same as figure 1 does, but also make other nodes and links transparent. Thus it is much easier than figure 1 to check the details of the connections.

5 FUTURE WORK

There are still a lot of space to improve in this brain connectivity visualization toolbox. Name a few:

- In the current version, it is not easy to look up the name of the interested node since the name is not shown near the label to reduce the complexity of the visualization. Ideally, when you move over or click the node, the names of corresponding nodes are expected to be highlighted in the right bottom table.
- Besides the name of node, the actual values would be better to show up when click the link.
- It would be better if it allows the zoom option that you can zoom in and out of with your mouse scroll-wheel. Then when the network is more dense (from 42 regions to 116 regions), users could zoom in to check the details within the sub-network.
- Without the brain shape as the background, it is hard to locate the brain regions even the layout could provide some information.

REFERENCES

- [1] M. Bostock, V. Ogievetsky, and J. Heer. D³ data-driven documents. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2301–2309, 2011.
- [2] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.
- [3] S. Gerhard, A. Daducci, A. Lemkaddem, R. Meuli, J.-P. Thiran, and P. Hagmann. The connectome viewer toolkit: an open source framework to manage, analyze, and visualize connectomes. *Frontiers in neuroinformatics*, 5:3, 2011.
- [4] S. Huang, J. Li, L. Sun, J. Ye, A. Fleisher, T. Wu, K. Chen, E. Reiman, A. D. N. Initiative, et al. Learning brain connectivity of alzheimer's disease by sparse inverse covariance estimation. *Neuroimage*, 50(3):935–949, 2010.
- [5] H. Y. Xia M, Wang J. Brainnet viewer: A network visualization tool for human brain connectomics. In *PLoS ONE* 8(7), 2013.

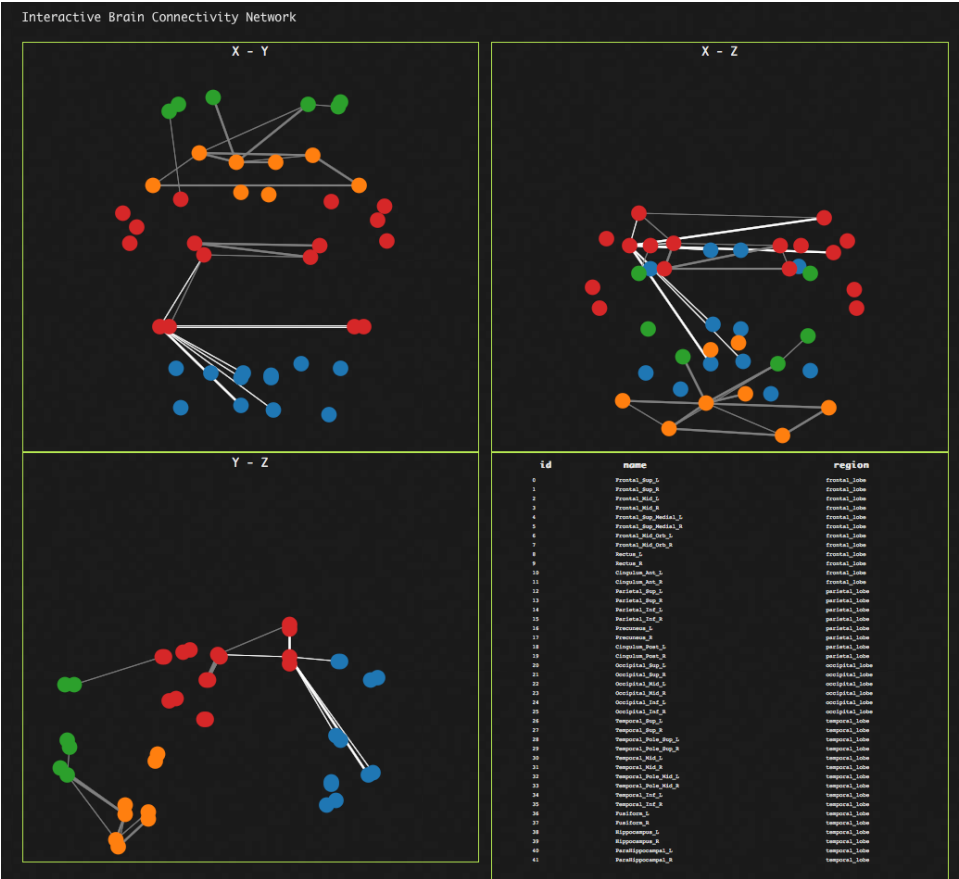


Fig. 1. when mouse is over one of nodes

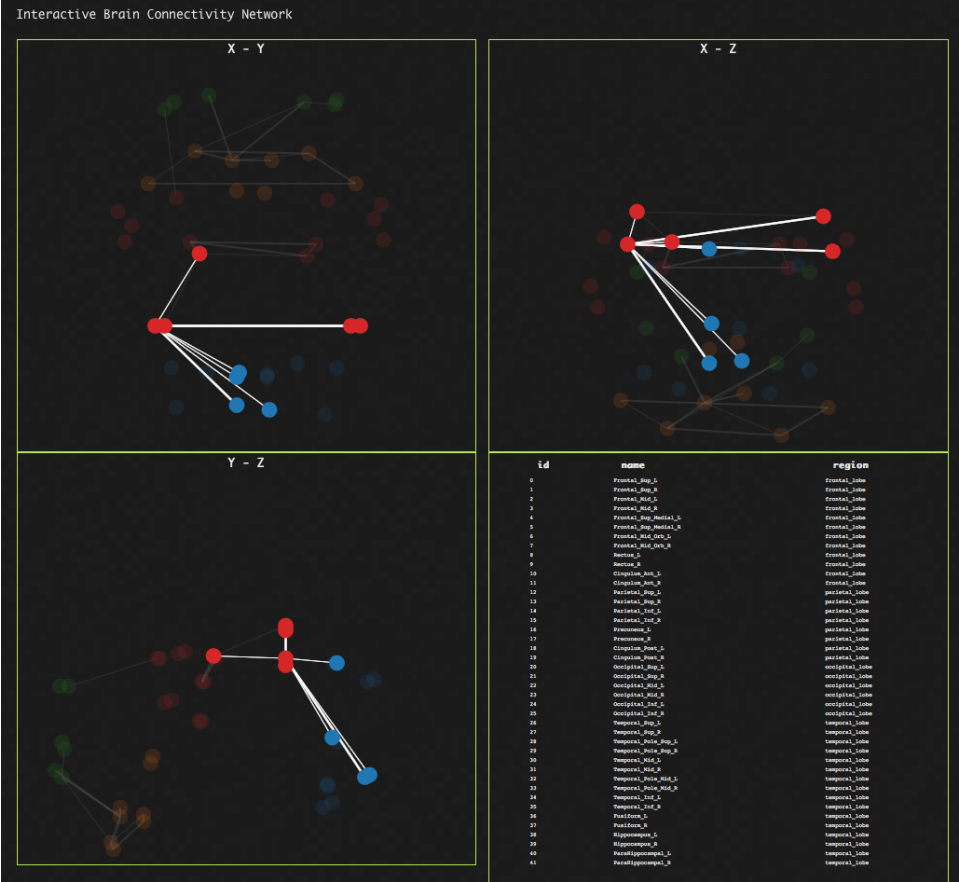


Fig. 2. when mouse click one of nodes