

This assignment is designed to give you the code to follow along with the recorded lecture, familiarize you with basic features of the Brain Connectivity Toolbox, and help you have an intuitive feel for graph theory and network analysis. You will need to have Matlab on your computer.

1. Download the Brain Connectivity Toolbox from <https://sites.google.com/site/bctnet/Home>.
2. Use Matlab to open GroupAverage_rsfcMRI.mat from the data folder and make sure the Brain Connectivity Toolbox is in your path.
3. Type: `imagesc(GroupAverage_rsfcMRI)` to see a visual representation of the functional connectivity matrix from Crossley et al., PNAS 2013, which has already been thresholded. Are there self connections in this matrix? Is the matrix directed or undirected?
4. Type: `histogram(reshape(GroupAverage_rsfcMRI,638*638,1))` to determine where the threshold has been set. What is it?
5. Type: `nnz(GroupAverage_rsfcMRI)` to find out how many nonzero entries there are in the matrix. What is the density?
6. Type: `d=degrees_und(GroupAverage_rsfcMRI)`; and then `plot(d)` to look at the distribution of degrees across nodes.
7. If you'd like to look at the strength of the nodes on the same plot, type `hold`. Then type `s=strengths_und(GroupAverage_rsfcMRI)`; to find the strength of the connections for each node and `plot(s)` to display them.
8. Now let's binarize the matrix and calculate some metrics (note that there are generalized versions of many metrics that can handle weighted matrices). Type `binMat=GroupAverage_rsfcMRI>0`; and then `imagesc(binMat)`. The matrix should look like the one in Step 3 but now all values are 0 or 1.
9. To calculate the clustering coefficient for each node, type `clust=clustering_coef_bu(binMat)`; and then `plot(clust)` to display the values. You can also type `mean(clust)` to obtain the average value for the clustering coefficient.
10. How do clustering coefficients compare to node degree or strength? You can find out using `plot`, e.g., `plot(clust, d, '*')`. Is the relationship between clustering coefficient and degree similar to the one between degree and strength?
11. Now let's change the threshold. Type `binMat2=GroupAverage_rsfcMRI>0.8`. If you look at binMat2 with `imagesc`, you can see that the number of nonzero nodes has decreased. Type `clust2=clustering_coef_bu(binMat2)`; and then `plot(clust2)` and `mean(clust2)`. How are the values different from before?
12. Let's look at the modularity of the network. There are a number of algorithms that can be used for this. Type `[M, Q]=community_louvain(binMat)`; M will be the module assignment for each parcel and Q the measure of overall modularity. You can obtain a visual representation of the module assignment with `plot(M)`. How many modules are there? Maybe you'd like to see how many parcels end up in each module. Type `hist(M)` for a display. For comparison, try the same commands using binMat2 instead of binMat, to see what happens when the threshold is increased.
13. Now we'll look at distance between nodes. Type `dist = distance_bin(binMat)`; and `imagesc(dist)` to display. If you'd like to see a histogram of the shortest distances between nodes, type `hist(reshape(dist, 638*638,1))`. Note that each node pair is currently counted twice because we have used the full symmetric matrix. Again, you can repeat the commands with binMat2 instead of binMat to see the effects of thresholding.

14. For global efficiency, type `E=efficiency_bin(binMat)` and `E=efficiency_bin(binMat2)`. How is efficiency affected by the higher threshold?
15. For betweenness centrality, type `bc=betweenness_bin(binMat)`; then `plot(bc)`. If you'd like to directly compare betweenness centrality at different thresholds, then type `hold` then `bc2=betweenness_bin(binMat2)`; then `plot(bc2)`.
16. If you'd like to visualize the nodes of a matrix, you can type `[x y z]=adjacency_plot_und(binMat2, Coord)`; then `plot3(x,y,z)`. If the matrix is too dense (like `binMat`), it will be hard to see the structure.

Bring your questions and comments to the live Q and A session!