Week of May 23-27

1. Added tables of interactions for the first and second neighbours in the graph. This allows a user to explore the strength of the interactions occurring at each level. I also implemented the ability to download these tables as a csv file for the user’s convenience.
2. Created much more advanced animations for the locating of genes within the graph. It turns out that changing the color of a gene of interest and panning to its location isn’t always enough to be able to instantly see that gene. Therefore, I created an animation that grows and shrinks the gene in order to attract attention to it.
3. Created the FDR adjusted correlation matrices for the entire genome for P Values of: 0.001, 0.01. 0.05, 0.1. I used these matrices to test and see what the performance would be like and if any optimizations to the R scripts could be made. It turns out that I was using the significantly less efficient dput and dget, as opposed to saveRDS and readRDS. Switching to the more efficient functions greatly improved the performance of my code.
4. I adjusted the concentric layout in order to make it more dynamic. Prior to this, the radii were all hardcoded, so the clusters would overlap once the number of nodes exceed a certain threshold. Now the radii are based on the number of nodes that need to be displayed. First, I determine the size of the biggest cluster in the graph (i.e. its diameter). Then using this information and the total number of clusters that need to be drawn, I calculate the size of the main radius.
5. Added an edge inspector card that is used to display information about an edge when a user clicks that edge.
6. Overhauled the way that styles are applied to nodes on the server. A single class used to be responsible for setting both color, size, and label placement. Unfortunately, once a style is already applied via a class selector, cytoscape does not allow for the overriding of that style via !important. Additionally, it does not allow one to override that style by using a more specific selector i.e. node.class as opposed to .class. Therefore, classes must be toggled on and off. However, if a single class is responsible for the styling of more than just one property, we might end up disabling a property that we want to permanently remain. I decided to split up the classes and make them more modular by giving them single responsibilities.
7. Added a graph summary card that shows relevant graph information such a # of self loops and total interactions without having to switch to the table view.
8. Changed the random layout so that the bounding box that it uses has more dynamic dimensions based on the size of the nodes and the number of nodes. Also added better styling to the random layout so that labels have a background color, and the genes of interest that the user selected are larger and striped.