Dataset: https://snap.stanford.edu/data/twitch_gamers.html
Twitch is a long running social media platform for streaming different types of self produced content such as video games and real life events which serves as one of the main hubs of this content online. This dataset shows the relations and overlap of viewers between twitch streamers, so by performing different types of algorithmic analysis on these relationships I will be able to make conclusions about the dataset as a whole, and Twitch as a platform.

The dataset above gives information about specific Twitch streamers, and has edges between each node(a streamer) which represent shared viewers between them. This is interesting to me because there is a large degree of separation between the different communities of streamers on Twitch. This is notable in that entire communities may be basically unaware of the existence of other large communities because of total lack of viewer overlap due to a high enough difference in the type of streamed content. Because of this, I thought that it would be enlightening to investigate the relationship between streamers on Twitch through overlapping viewer connections to see how closely connected they are to each other.

The algorithm which I chose to run on this dataset was a modified version of breadth first search to find the relationship between the number of nodes at different distances away from any particular node. Essentially it works like a normal breadth first search which keeps track of the distance of each node added to the queue, but it stops adding nodes to the queue when the distance of a current node meets or exceeds a certain threshold, in this case a distance of 2. However I have left it open in the code that a distance of 3 is also possible and is something that can be easily worked out.

Unfortunately anything past a distance of 2 is computationally expensive. (there is no way around this, there is just a lot of nodes and the average amount of nodes connected at distance 2 is in the tens of thousands) Nevertheless, using this algorithm the number of nodes at distance 1 seems to average to about 80, while the number of nodes at distance 2 seems to average to about 42000. This is interesting because it shows how tight knit most direct circles for streamers are, where there will only really be great turnover between a small group of streamers, however when a viewer goes outside of that circle, they have an immense amount of options just one more step away. This is also a good distribution for a social media network, since it fits a power law model. Also, I found that the variance of values for the amount of streamers distance 1 for any given node is much higher than that of distance 2 for any given node.