***Report (Project One)***

Student Name: Xinqian Wang

Student ID: 45654897

Student Email: xinqian.wang@uq.net.au

Here is the first paragraph, you can briefly tell us what you have done in this report.

## ***Task 1***

Firstly, load the network data into the program by using libraries network. Then, use “nx.betweenness\_centrality” functions to check what exactly the final solution would bring. Also, finding that thse top-10 nodes would have a large degree value.

Secondly, try to use Brute-force Search, and realize the problem has a huge time complexity for my laptop. Finally, planned to implement the Brandes Algorithm and get the same solution with only around 500 seconds running time.

Thirdly, some visulizations are done to explore the network data. Figure 1. plots the relashionship between betweenness-centrality and degree value. Figure 2. uses a box plot and plots the whole distribution of degree value. Basically, we can see the one with a high degree or high betweenness-centrality is just minority.

Overall, the Top-10 betweenness\_centrality nodes are below:

[107 1684 3437 1912 1085 0 698 567 58 428]

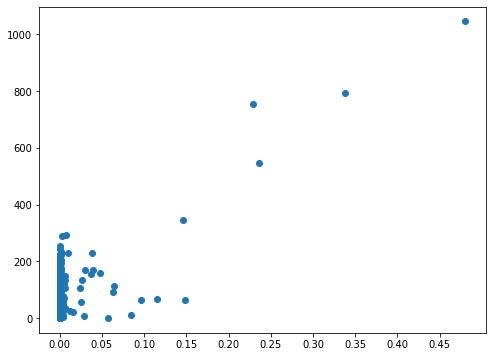
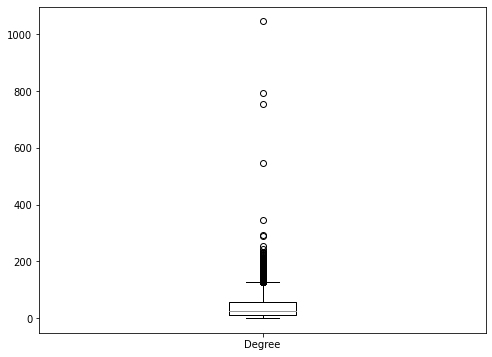
 

Figure 1. Figure 2.

## ***Task 2***

Firstly, using “nx.pagerank” to check what would the final out put going to be.

Secondly, constructing the degree matrix D and transfer it into a numpy array data structure to calculate it’s inverse.

Thirdly, constructing the adjacent matrix A by using a nested loop.

Fourthly, setting the hyperparameters, like alpha and beta.

Fifthly, using power iternation to do the calculation, my threshold is 0.001

The answer of top-10 pagerank nodes are below:

[3437 107 1684 0 1912 348 686 3980 414 483]

Additionally, there will be a slight different between the power iternation solution and “nx.pagerank” function, with a difference in the 10th nodes. Whereas if you output the top-11 highest page rank nodes, you would find the 11th of the power iternation solution is just the 10th node calculated by the “nx.pagerank” function, which is very interesting.

As for the reason of the differences happening, the further researches would be needed.

## ***Summary***

For Betweeness-centrality Calculation, the challenge would be implementing the Blandes Algorithm. It is not enough by only understanding how the algorithm is operating, you got to need to code by yourselg.

For Pagerank-centrality Calculation, the most important parts are the knowledge of matrix calculation in python and how the NetworkX load the data.