Jinghan(Tracy) Cui DS210

Final Project Report

The final project is based on the data of the social circle in Facebook, which includes 4039 nodes as Facebook users and 88234 edges as their friendship circles. This dataset is a social net work dataset and it has a considerable size and complexity: https://snap.stanford.edu/data/ego-Facebook.html.

The overall project includes three modules and one main function. The three modules include the module of the data processing, the algorithms for computing the shortest path and the average shortest path length between all pairs of nodes in the graph by running Dijkstra's algorithm, and the visualization of the connected component size distribution. Moreover, the main function rust file includes the implemented algorithms of breadth-first search.

Starting from the order in main function, I printed out the possible paths of starting node for BFS. It visits all nodes at a given level of the graph. The output is in vector. After that, I used the Dijkstra's algorithm to compute the shortest path lengths from the start node and construct the shortest path by tracing back from the end node. After the function of shortest_path, I iterate over all pairs of nodes in the graph and runs Dijkstra's algorithm to compute the shortest path length between each pair of nodes, and dividing the total number of pairs of nodes to check the average shortest path length. The result is stable and interesting, which is 0.0019338911. The visualizing results function at last is to compute the size distribution of the connected components. The function uses the "plotters" crate to create a chart of the size distribution, and the result is shown by the output.svg.

The takeaways that I got from this project is that I have a better understanding of the use of Rust crate. I have a more thorough understand of how the algorithms and construction of the function in rust.