Sia Sharma
DS210 - Final Project Report

Partial Collaborators: Om Italiya and Nikita Salkar (we talked about main ideas on how to code the modules for our individual datasets and helped debug our tests together)

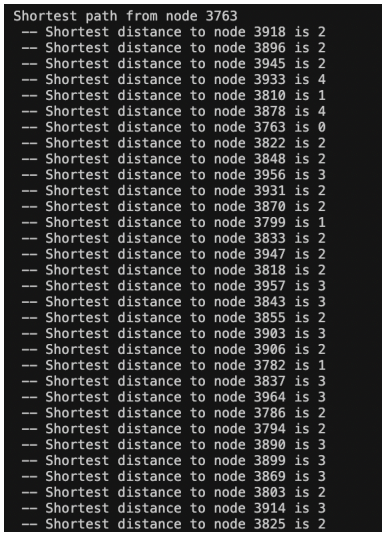
In the era of social media, relationships in social networks can represent important data analysis and trends. My project aims to address this question: "Are the friends of my friends also my friends?" By analyzing a dataset obtained from Facebook, I explore the interconnectedness of individuals within social networks. I do this by understanding how similar the sets of neighboring vertices for 2 nodes that are connected are.

Dataset Description:
The dataset comprises anonymized 'circles' or 'friends lists' from Facebook, collected via a survey app. Each node represents a user profile, and edges denote friendships between individuals.
Dataset link: https://snap.stanford.edu/data/ego-Facebook.html

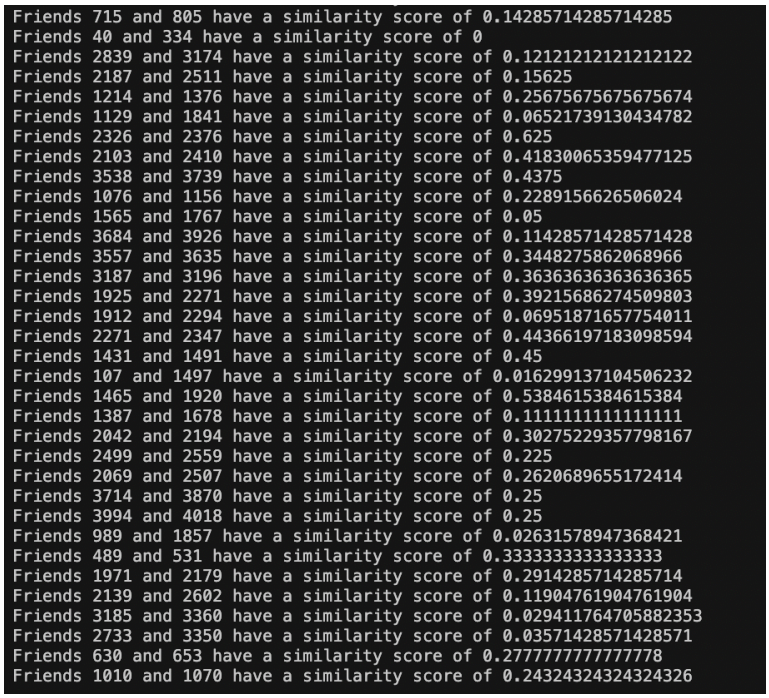
Code Output:

**** How to get output:** You can comment out certain lines in my main module to only see specific outputs. The overall output is my shortest paths, followed by similarity scores for each node pair, followed by the statistical analysis of mean, max, and percentages with a given BFS distance. You can change the BFS distance in the main function to be 1,2,3 etc and see how it affects the statistical analysis. I have long outputs for the shortest path and jaccard similarity, so I will only be including part of what is printed for each in this report.

Shortest path output: Prints for all the nodes the node at the top is connected to and how far the distance is. The shortest paths from each node to all other nodes in the network, and it indicates how many steps it takes to reach each node from a given starting node. It provides insights into the connectivity and structure of the social network.



Jaccard similarity score output (For a BFS distance of 1):



So what is a jaccard similarity and what does it measure?

Jaccard similarity is a measure of similarity between two sets. It is defined as the size of the intersection of the sets divided by the size of the union of the sets. In the context of social networks, Jaccard similarity can be used to quantify the similarity between the neighborhoods (sets of connections) of two individuals. In my project, I used it to find the similarity of neighboring vertices for 2 connected nodes. Jaccard similarity scores for pairs of nodes, showing the degree of overlap in their mutual connections. Higher scores imply greater similarity between the neighborhoods.

Formula for Jaccard's:

$J(A,B) = (|A \cap B|$ **divided by** $|A \cup B|)$ where A and B is the size of the intersection of sets A and B (number of common elements), and A or B is the size of the union of sets A and B (total number of distinct elements).

Statistical analysis output (for a BFS distance of 1):

```
Performing Statistical Analysis:
Mean Similarity Value: 0.2069140776277656
The most similar set of friends are (3523, 3562) with a similarity of 0.9090909090909091

Percentage of Friends with Certain Similarities:
Percentage of friends that have a similarity of 0.1 are: 67.12594%
Percentage of friends that have a similarity of 0.2 are: 43.55020%
Percentage of friends that have a similarity of 0.3 are: 25.87016%
Percentage of friends that have a similarity of 0.4 are: 13.44837%
Percentage of friends that have a similarity of 0.5 are: 6.35814%
Percentage of friends that have a similarity of 0.6 are: 2.81599%
Percentage of friends that have a similarity of 0.7 are: 0.94497%
Percentage of friends that have a similarity of 0.8 are: 0.14902%
Percentage of friends that have a similarity of 0.9 are: 0.00355%
Percentage of friends that have a similarity of 1.0 are: 0.00000%
```

Statistical analysis output (for a BFS distance of 2):

```
Performing Statistical Analysis:
Mean Similarity Value: 0.04694559469702773
The most similar set of friends are (3481, 3523) with a similarity of 0.9166666666666666

Percentage of Friends with Certain Similarities:
Percentage of friends that have a similarity of 0.1 are: 16.76500%
Percentage of friends that have a similarity of 0.2 are: 5.73763%
Percentage of friends that have a similarity of 0.3 are: 1.92282%
Percentage of friends that have a similarity of 0.4 are: 0.61164%
Percentage of friends that have a similarity of 0.5 are: 0.19541%
Percentage of friends that have a similarity of 0.6 are: 0.04708%
Percentage of friends that have a similarity of 0.7 are: 0.00417%
Percentage of friends that have a similarity of 0.8 are: 0.00083%
Percentage of friends that have a similarity of 0.9 are: 0.00042%
Percentage of friends that have a similarity of 1.0 are: 0.00000%
```

Statistical analysis output (for a BFS distance of 3):

```
Performing Statistical Analysis:
Mean Similarity Value: 0.0030324132932944975
The most similar set of friends are (3243, 3364) with a similarity of 0.6666666666666666

Percentage of Friends with Certain Similarities:
Percentage of friends that have a similarity of 0.1 are: 0.62759%
Percentage of friends that have a similarity of 0.2 are: 0.10390%
Percentage of friends that have a similarity of 0.3 are: 0.02348%
Percentage of friends that have a similarity of 0.4 are: 0.00421%
Percentage of friends that have a similarity of 0.5 are: 0.00133%
Percentage of friends that have a similarity of 0.6 are: 0.00022%
Percentage of friends that have a similarity of 0.7 are: 0.00000%
Percentage of friends that have a similarity of 0.8 are: 0.00000%
Percentage of friends that have a similarity of 0.9 are: 0.00000%
Percentage of friends that have a similarity of 1.0 are: 0.00000%
```

The values of mean and max similarity are really important as the BFS distance changes, and it shows how similarities can change when there are more 'steps' between two nodes whose neighboring vertices are analyzed. The percentage of nodes with similarity scores above various thresholds is used to understand the strength and distribution of connections within the network. For each BFS we can see the

percentage of friends in the whole dataset who are at least that threshold similar from 0.1-1.0. The similarity value 1.0 is identical. An interesting finding is that even with a BFS distance of 1 (the closest connections between nodes), the percent of friends with a similarity of 1.0 is 0%. This means that in my Facebook dataset, no two people have the exact same friends. When we look at a similarity of 0.6 for the same distance (which is more than 50% similar), 2.81% of people are that similar, and while this number is still low, it proves that it is true that for some people, their friends can also be their friends friends to a pretty similar degree.

Further analysis on my program:

Code Explanations:
Graph Representation: I have used a graph data structure to model the social network, with nodes representing user profiles and edges indicating friendships. The Graph module facilitates the creation of the graph from a file and computation of shortest paths between nodes.
Shortest Path Calculation: Using the Breadth-First Search (BFS) algorithm, the Shortest Path module calculates the shortest paths from all nodes in the graph. This information is useful for understanding the closeness of nodes, and more broadly, the connections between individuals in the social network. BFS systematically visits all the neighboring nodes of the current node before moving on to the next level of neighbors. It operates by maintaining a queue data structure to keep track of nodes to be visited, making sure that nodes are visited in the order of their distance from the source node.
Similarity Score Computation: The Similarity Scores module computes the Jaccard similarity between the neighborhoods of pairs of nodes in the graph. This metric measures the similarity of mutual connections and is the main way to understand if friends of friends are friends themselves too or not, scored from a range of 0.0 to 1.0. Higher similarity scores suggest a greater overlap in mutual connections, indicating that friends of friends are likely to be friends themselves.
Statistical Analysis: My Statistical Analysis module conducts statistical analyses on the computed similarity scores. These include determining the mean similarity between pairs of nodes, identifying the maximum similarity, and analyzing the percentage of nodes with similarity scores above specified thresholds, like the percent of friends that are over 0.5% similar (meaning their connections are 50% similar too).
Testing: I used a test.txt with just 4 nodes to test all my functions (at the bottom of the main module). I did a cargo test, and all 4 tests passed.

Analysis of project output:

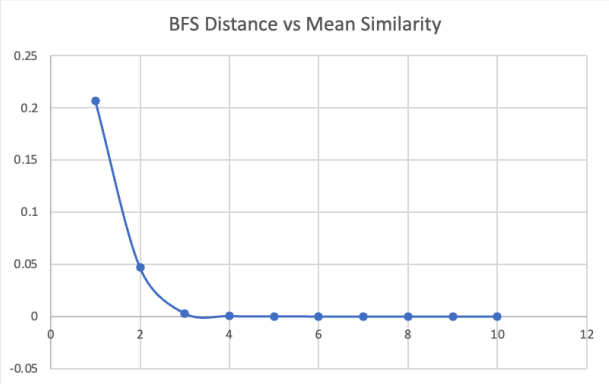
While analyzing the change in mean and max similarity values for different BFS neighbors (i.e. 1, 2, 3), if we make a table and graph the similarity outputs for each BFS distance, this is the result:

Table:

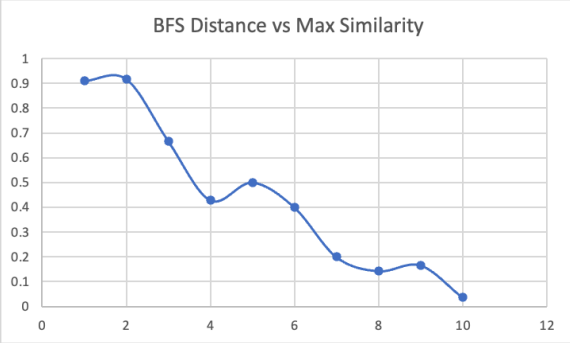
BFS Distance	Mean Similarity
1	0.20691408
2	0.04694559
3	0.00303241
4	0.00043522
5	0.00017112
6	4.9978E-05
7	3.2964E-05
8	1.6237E-05
9	1.4943E-05
10	3.7491E-06

BFS Distance	Max Similarity
1	0.90909091
2	0.91666667
3	0.66666667
4	0.42857143
5	0.5
6	0.4
7	0.2
8	0.14285714
9	0.16666667
10	0.03703704

Plot for Mean Similarity:



Plot for Max Similarity:



These plots show that as the Jaccard similarity is computed on more distances of BFS (like 2 nodes between a connection, 3, 4 or more), the similarity tends to decrease. This, for my project, allows me to infer that individuals who are less close to other individuals tend to have less similarities and different friends too. Therefore, the closer you are to someone, (3 or less degrees away), the more likely you are to have similar friend groups as them and as a result it is more likely that the friends of your friend are also your friends.

The maximum similarly found in my dataset for a BFS distance of 1 was around 0.909, which means the "individuals" 577 and 578 are almost identical when it comes to their connections. These neighboring vertices were 0.909 the same, making the nodes they were connected to (577 and 578) very similar as well. This shows that when friends are connected by a short degree (i.e. 1), they have a lot of mutual friends (almost 91%). However, when you go further away, like with a BFS distance of 3 separating the

friends, the maximum is just 0.66, which shows that they do not have that many mutual connections anymore. This data can be utilized for real work applications in terms of Facebook, like perhaps if someone is interested in an advertisement/something displayed on Facebook Marketplace, it would be useful for the algorithm to also present the advertisement to their close connections and vertices (mutuals) of the close connections (BFS distance of 3 or less) to gain as much revenue as possible.
Overall, this project demonstrates the utility of graph algorithms and similarity scores in analyzing social network connectivity. By computing similarity scores and conducting statistical analysis, I'm providing insights into the structure and dynamics of social networks, confirming the hypothesis that friends of friends tend to share mutual connections, but only if they are pretty well connected themselves.