Health Similarity Graph Project Writeup

1. **Project overview:** My project constructs and analyzes an undirected graph from a large dataset, in CSV form, that contains health and lifestyle information for a group of about 108000 people. I got this dataset from Kaggle and it came from Canadian Community Health Survey. The rows corresponded to the people(108253) and the 50 columns were different variables. Due to the large size of the dataset I had to use a random sample of 10000 people instead of the whole dataset because it was taking way too long to run. Each node in my graph represents a person and the edges are added between the nodes when two individuals have similar specific attributes. The main questions I was trying to answer were: How does physical activity and weight perception relate to life satisfaction and health and does low income and food security cause for worse health outcomes? The variables(columns) I looked at were the Weight_state, Total_physical_act_time, Life_satisfaction, Gen_health_state, Total_income, Food_security, High_BP, High_cholestrol, and Diabetic. My goal was to analyze the connectivity of nodes corresponding to people and their variables in the graph to see if there is a correlation between the variables to answer my research questions and also to compute the overall metrics of the graph. I then tried to visualize the final output so that I could see how the results looked.
2. **Data processing:** The dataset was in a CSV file so I was easily able to load it into my project using the csv crate methods I learned in class. Each row of the dataset sample(10000 random rows) is parsed into a PersonNode struct using the load_people function in my parser.rs file. I put in the default fallbacks by adding question marks just in case there was missing or invalid data. I catagorized activity level into Low, Medium, or High, based on the amount of time a person does physical activity. I also did similarity checks and in these I filtered out the invalid code/variables.
3. **Code structure:**
    a. Modules: I added 5 modules into my project: graph.rs which defines the person representation and the graph data structure, similarity.rs which implements similarity logic for connecting nodes, parser.rs which loads and parses CSV file into structures PersonNode data(this module is probably the most important and trickiest to write), analysis.rs which preforms graph analytics for the overal metrics and specific things I was interested in analyzing, and tests.rs which is where I put my tests to make sure the similarity logic and operations on the graph was running smoothly.
    b. Key functions and types: The most important type is my PersonNode struct. It represents a singular person and contains the column data from the csv file on the id, weight state, activity level, life satisfaction general health state, income, food security, and booleans for high blood pressure, high cholesterol, and diabetes. I then have an enum called ActivityLevel which just categorizes the physical activity into 3 different levels: low, medium, and high. It also has an unknown level just in case there is not enough data or it is bad. I have a HealthGraph struct which keeps people(nodes) and their connections(edges)

based on similarity. It uses a few important functions: add_node() which inserts a PersonNode, add_edge() which connects two nodes in both directions, neighbors() which checks the neighbors, degree() which calculates the degree of the graph, and total_edges(). An extremely important function is my is_similar function which takes two people as PersonNode references and checks if they should be connected based on their similarity. It requires certain variables like income and weight to be matching in order for the nodes to be considered similar. I then have an analyze_health_by_income_and_food_security() function that groups and prints statistics based on the income and food security. I used it to look at the percentage of people with issues like high blood pressure based on the income/food security groups. I also have an average_shortest_path_length() function which preforms breadth-first search and computes the mean distance between all connected nodes. It outputs a float that represents the average path length across the graph.

c. Main workflow: The workflow starts with loading the CSV data into PersonNodes and then inserting the nodes into a HealthGraph. It then uses is_similar() to evaluate all pairs of nodes and adds edges. It then runs graph analysis where it uses compute_degrees(), average_degree(), and node_w_highest_degree() to give degree stats, average_shortest_path_length() to do path analysis, and analyze_health_by_income_and_food_security() to analyze the grouped health stats. Finally it outputs the analysis into the terminal.

d. **Additional final add in: To my final version I added a visual components. The bar chart summarizes the findings of the early analysis. I used plotly to do this which I had to do research on and it was a little tricky to work with. I made this a separate rust project in my project folder since I input an output from the graph analysis. I used the plotly crate because it allowed me to create interactive plots. I didn't have to define any functions or anything for this, I just had to learn how to implement plotly and get a desired output.

4. **Tests:** I have three tests: test_add_node_and_edge which that nodes are added correctly and that the edges are mutual, test_similarity_pos which tests a situation where to nodes would be considered similar and makes sure the code would recognize that, and test_similarity_neg which makes sure the code would recognize two nodes that are not similar. These tests confirm that my graph connectivity logic is correct and that the similarity function runs correctly and recognizes the necessary criteria. When I run cargo test, I get this output:

```
(base) willannis@Wills-MacBook-Air Project % cargo test
   Compiling Project v0.1.0 (/Users/willannis/Documents/Final_Project/Project)
    Finished `test` profile [unoptimized + debuginfo] target(s) in 1.24s
     Running unittests src/main.rs (target/debug/deps/Project-6b8a6df5dce048b5)

running 3 tests
test tests::tests::test_similarity_pos ... ok
test tests::tests::test_similarity_neg ... ok
test tests::tests::test_add_node_and_edge ... ok

test result: ok. 3 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out; finished in 0.00s

(base) willannis@Wills-MacBook-Air Project %
```

5. **Results:** I ran my program a few times with different sample sizes and ultimately decided to take 10,000 random people from the csv files. The output is never the same

since it is taking data samples so I ran it a few times and found that 10,000 random people is enough to get consistent outputs. Unfortunately the does take a little long time run(a few minutes) because it has a quadratic runtime so I used cargo run –release. Here are a few outputs:

```
(base) willannis@Wills-MacBook-Air Project % cargo run --release
   Compiling Project v0.1.0 (/Users/willannis/Documents/Final_Project/Project)
    Finished `release` profile [optimized] target(s) in 1.11s
     Running `target/release/Project`
108252 people were loaded.
10000 people were selected for the graph.
10000 nodes were added to the graph.
1274600 edges and 10000 nodes were added to the graph.

--- Graph Analysis ---
Average connections per person: 254.92
Person 88112 has the highest degree: 1357
Average shortest path length: 1.86

--- Health Conditions by Income and Food Security ---
Income: 3, Food Security: 2 | n = 59 | High BP: 27.1%, High Cholesterol: 11.9%, Diabetic: 5.1%
Income: 5, Food Security: 0 | n = 3780 | High BP: 21.4%, High Cholesterol: 14.9%, Diabetic: 6.9%
Income: 1, Food Security: 6 | n = 68 | High BP: 20.6%, High Cholesterol: 14.7%, Diabetic: 7.4%
Income: 2, Food Security: 1 | n = 53 | High BP: 37.7%, High Cholesterol: 17.0%, Diabetic: 17.0%
Income: 2, Food Security: 0 | n = 1248 | High BP: 38.1%, High Cholesterol: 22.1%, Diabetic: 14.7%
Income: 4, Food Security: 1 | n = 39 | High BP: 10.3%, High Cholesterol: 5.1%, Diabetic: 7.7%
Income: 5, Food Security: 2 | n = 108 | High BP: 10.2%, High Cholesterol: 8.3%, Diabetic: 6.5%
Income: 3, Food Security: 1 | n = 48 | High BP: 33.3%, High Cholesterol: 16.7%, Diabetic: 14.6%
Income: 4, Food Security: 2 | n = 36 | High BP: 19.4%, High Cholesterol: 11.1%, Diabetic: 5.6%
Income: 1, Food Security: 1 | n = 24 | High BP: 58.3%, High Cholesterol: 29.2%, Diabetic: 8.3%
Income: 5, Food Security: 6 | n = 553 | High BP: 14.6%, High Cholesterol: 11.8%, Diabetic: 3.1%
Income: 2, Food Security: 2 | n = 104 | High BP: 28.8%, High Cholesterol: 16.3%, Diabetic: 15.4%
Income: 2, Food Security: 6 | n = 165 | High BP: 29.1%, High Cholesterol: 10.3%, Diabetic: 5.5%
Income: 3, Food Security: 0 | n = 1241 | High BP: 36.5%, High Cholesterol: 23.1%, Diabetic: 12.2%
Income: 4, Food Security: 0 | n = 1064 | High BP: 29.8%, High Cholesterol: 21.1%, Diabetic: 10.3%
Income: 1, Food Security: 3 | n = 57 | High BP: 42.1%, High Cholesterol: 36.8%, Diabetic: 7.0%
Income: 3, Food Security: 3 | n = 31 | High BP: 16.1%, High Cholesterol: 3.2%, Diabetic: 3.2%
Income: 4, Food Security: 6 | n = 140 | High BP: 26.4%, High Cholesterol: 12.1%, Diabetic: 7.9%
Income: 3, Food Security: 6 | n = 180 | High BP: 26.7%, High Cholesterol: 17.2%, Diabetic: 7.2%
Income: 5, Food Security: 3 | n = 43 | High BP: 9.3%, High Cholesterol: 14.0%, Diabetic: 4.7%
Income: 5, Food Security: 1 | n = 95 | High BP: 20.0%, High Cholesterol: 11.6%, Diabetic: 3.2%
Income: 1, Food Security: 2 | n = 71 | High BP: 22.5%, High Cholesterol: 22.5%, Diabetic: 8.5%
Income: 4, Food Security: 3 | n = 14 | High BP: 42.9%, High Cholesterol: 14.3%, Diabetic: 21.4%
Income: 1, Food Security: 0 | n = 301 | High BP: 32.2%, High Cholesterol: 20.3%, Diabetic: 12.0%
Income: 2, Food Security: 3 | n = 63 | High BP: 31.7%, High Cholesterol: 17.5%, Diabetic: 14.3%
(base) willannis@Wills-MacBook-Air Project % cargo test
```

```
(base) willannis@Wills-MacBook-Air Project % cargo run --release
    Finished `release` profile [optimized] target(s) in 0.07s
     Running `target/release/Project`
108252 people were loaded.
10000 people were selected for the graph.
10000 nodes were added to the graph.
1283815 edges and 10000 nodes were added to the graph.

--- Graph Analysis ---
Average connections per person: 256.76
Person 31038 has the highest degree: 1368
Average shortest path length: 1.86

--- Health Conditions by Income and Food Security ---
Income: 2, Food Security: 0 | n = 1218 | High BP: 43.4%, High Cholesterol: 28.0%, Diabetic: 17.3%
Income: 3, Food Security: 0 | n = 1301 | High BP: 33.7%, High Cholesterol: 22.0%, Diabetic: 11.8%
Income: 5, Food Security: 1 | n = 106 | High BP: 13.2%, High Cholesterol: 10.4%, Diabetic: 8.5%
Income: 5, Food Security: 6 | n = 554 | High BP: 14.4%, High Cholesterol: 11.6%, Diabetic: 3.8%
Income: 3, Food Security: 6 | n = 171 | High BP: 24.0%, High Cholesterol: 11.7%, Diabetic: 8.8%
Income: 4, Food Security: 1 | n = 36 | High BP: 22.2%, High Cholesterol: 13.9%, Diabetic: 8.3%
Income: 4, Food Security: 2 | n = 37 | High BP: 24.3%, High Cholesterol: 13.5%, Diabetic: 13.5%
Income: 4, Food Security: 6 | n = 148 | High BP: 22.3%, High Cholesterol: 13.5%, Diabetic: 8.1%
Income: 1, Food Security: 0 | n = 346 | High BP: 29.5%, High Cholesterol: 20.8%, Diabetic: 10.7%
Income: 3, Food Security: 1 | n = 47 | High BP: 34.0%, High Cholesterol: 10.6%, Diabetic: 19.1%
Income: 1, Food Security: 2 | n = 65 | High BP: 20.0%, High Cholesterol: 12.3%, Diabetic: 7.7%
Income: 1, Food Security: 1 | n = 27 | High BP: 37.0%, High Cholesterol: 33.3%, Diabetic: 14.8%
Income: 4, Food Security: 0 | n = 1107 | High BP: 30.4%, High Cholesterol: 22.0%, Diabetic: 10.6%
Income: 5, Food Security: 2 | n = 104 | High BP: 13.5%, High Cholesterol: 8.7%, Diabetic: 5.8%
Income: 2, Food Security: 1 | n = 62 | High BP: 33.9%, High Cholesterol: 17.7%, Diabetic: 11.3%
Income: 3, Food Security: 3 | n = 31 | High BP: 22.6%, High Cholesterol: 9.7%, Diabetic: 3.2%
Income: 2, Food Security: 2 | n = 87 | High BP: 37.9%, High Cholesterol: 23.0%, Diabetic: 8.0%
Income: 4, Food Security: 3 | n = 11 | High BP: 27.3%, High Cholesterol: 0.0%, Diabetic: 0.0%
Income: 3, Food Security: 2 | n = 70 | High BP: 22.9%, High Cholesterol: 12.9%, Diabetic: 11.4%
Income: 1, Food Security: 3 | n = 59 | High BP: 33.9%, High Cholesterol: 32.2%, Diabetic: 6.8%
Income: 5, Food Security: 0 | n = 3715 | High BP: 21.8%, High Cholesterol: 15.0%, Diabetic: 6.9%
Income: 2, Food Security: 6 | n = 155 | High BP: 28.4%, High Cholesterol: 11.0%, Diabetic: 6.5%
Income: 5, Food Security: 3 | n = 31 | High BP: 12.9%, High Cholesterol: 6.5%, Diabetic: 3.2%
Income: 2, Food Security: 3 | n = 55 | High BP: 27.3%, High Cholesterol: 14.5%, Diabetic: 14.5%
Income: 1, Food Security: 6 | n = 63 | High BP: 19.0%, High Cholesterol: 12.7%, Diabetic: 0.0%
(base) willannis@Wills-MacBook-Air Project %
```

```
(base) willannis@Wills-MacBook-Air Project % cargo run --release
   Compiling Project v0.1.0 (/Users/willannis/Documents/Final_Project/Project)
    Finished `release` profile [optimized] target(s) in 1.82s
     Running `target/release/Project`
108252 people were loaded.
10000 people were selected for the graph.
10000 nodes were added to the graph.
1288441 edges and 10000 nodes were added to the graph.

--- Graph Analysis ---
Average connections per person: 257.69
Person 7509 has the highest degree: 1324
Average shortest path length: 1.85

--- Health Conditions by Income and Food Security ---
Income: 2, Food Security: 0 | n = 1183 | High BP: 40.2%, High Cholesterol: 22.9%, Diabetic: 16.0%
Income: 3, Food Security: 1 | n = 56 | High BP: 33.9%, High Cholesterol: 14.3%, Diabetic: 14.3%
Income: 2, Food Security: 3 | n = 68 | High BP: 23.5%, High Cholesterol: 11.8%, Diabetic: 4.4%
Income: 1, Food Security: 0 | n = 359 | High BP: 31.2%, High Cholesterol: 17.8%, Diabetic: 12.8%
Income: 5, Food Security: 0 | n = 3752 | High BP: 22.0%, High Cholesterol: 16.4%, Diabetic: 6.2%
Income: 1, Food Security: 6 | n = 61 | High BP: 16.4%, High Cholesterol: 16.4%, Diabetic: 3.3%
Income: 2, Food Security: 6 | n = 153 | High BP: 30.1%, High Cholesterol: 15.7%, Diabetic: 7.2%
Income: 3, Food Security: 0 | n = 1300 | High BP: 34.5%, High Cholesterol: 21.7%, Diabetic: 13.1%
Income: 1, Food Security: 2 | n = 69 | High BP: 24.6%, High Cholesterol: 21.7%, Diabetic: 14.5%
Income: 4, Food Security: 6 | n = 159 | High BP: 25.8%, High Cholesterol: 14.5%, Diabetic: 5.0%
Income: 5, Food Security: 6 | n = 564 | High BP: 18.6%, High Cholesterol: 11.9%, Diabetic: 4.8%
Income: 1, Food Security: 3 | n = 66 | High BP: 36.4%, High Cholesterol: 36.4%, Diabetic: 7.6%
Income: 2, Food Security: 2 | n = 109 | High BP: 31.2%, High Cholesterol: 22.9%, Diabetic: 15.6%
Income: 3, Food Security: 3 | n = 35 | High BP: 14.3%, High Cholesterol: 8.6%, Diabetic: 14.3%
Income: 1, Food Security: 1 | n = 29 | High BP: 31.0%, High Cholesterol: 17.2%, Diabetic: 3.4%
Income: 2, Food Security: 1 | n = 70 | High BP: 28.6%, High Cholesterol: 17.1%, Diabetic: 12.9%
Income: 3, Food Security: 2 | n = 64 | High BP: 21.9%, High Cholesterol: 14.1%, Diabetic: 9.4%
Income: 5, Food Security: 2 | n = 97 | High BP: 12.4%, High Cholesterol: 12.4%, Diabetic: 3.1%
Income: 4, Food Security: 1 | n = 37 | High BP: 24.3%, High Cholesterol: 16.2%, Diabetic: 2.7%
Income: 4, Food Security: 2 | n = 40 | High BP: 15.0%, High Cholesterol: 7.5%, Diabetic: 10.0%
Income: 4, Food Security: 3 | n = 20 | High BP: 40.0%, High Cholesterol: 25.0%, Diabetic: 5.0%
Income: 3, Food Security: 6 | n = 158 | High BP: 22.2%, High Cholesterol: 16.5%, Diabetic: 7.6%
Income: 4, Food Security: 0 | n = 1047 | High BP: 29.7%, High Cholesterol: 21.2%, Diabetic: 11.8%
Income: 5, Food Security: 1 | n = 95 | High BP: 20.0%, High Cholesterol: 12.6%, Diabetic: 5.3%
Income: 5, Food Security: 3 | n = 27 | High BP: 14.8%, High Cholesterol: 7.4%, Diabetic: 0.0%
(base) willannis@Wills-MacBook-Air Project %
```

When looking at these outputs I first noticed that the average number of connections is between 250 and 260 which shows that there is definitely clustering going on between groups of people, allowing me to conclude that general health state is most likely linked with a person's overall life satisfaction. If this number was smaller then I would not think this but since a typical node has around 260 connections, I do think that life satisfaction and general health go hand-in-hand. Also, the highest number of connections is similar for all 3 run times and it is around 1350 which shows that this large group is consistently being formed. I went back and looked at the nodes that had the highest degree and for each time I ran the project it was a node that has a low weight state(1), a high general health score(4/5) and a very high life satisfaction score(8/9) which shows me that typically when someone has better physical health, they are more satisfied with their life. When thinking about my other research question, in regards to income being related to health, I was looking at health conditions and seeing if they were linked with a persons income and food security which I grouped together, under the health conditions by income and food security line in my output. For this specific question I did definitely notice a trend in the results. I noticed that the groups with low food security or low income had much higher rates of high cholesterol and diabetes. The groups that had high food security and a higher income had much lower rates of diabetes and low cholesterol levels. I thought this was interesting as it shows that lower income and food security may be linked to worse overall health. I wasn't too sure if high blood pressure was significantly linked to the results, it seemed the vary a little more and not follow a strong trend. I also found that a lot of people have low food security levels as the largest groups had food security levels of 0 and higher diabetes levels. I mostly ignored the smaller groups as there was more room for outliers to scew the validity of the results. The visual charts also helped as they allowed me to actually see the data. My scatter plot came out odd and I struggled to fix it but it did show me that actually a lot of people with higher incomes has low food security which surprised me a lot.

6. **Usage instructions:** There is not user interaction after the program is run. All the user has to do is cd to the project in the terminal, specifically the Project subfolder of the main

project folder and then run **cargo build** in the terminal. After this, run **cargo run –release** and view the results! There is another folder called Projectdata in the main folder that has the keys for the data set so if the user wants to go look at the data set they can use the key to understand what it means. Using cargo run –release has caused my project to typically run in about a minute, maybe even less time. It runs most of the code pretty quickly, one of the last outputs just takes a little extra time since this has quadratic run time. If you want it to run faster, you can edit the code so that instead of data 10,000 random people, it would take less. Then you can either run the visualization to see both the scatter plot and the bar chart or you can click on the photo of the bar chart that I put in the visualization folder.

7. **Citations:** I didn't want to use AI because I have yet to find one that is good at Rust(it just confuses me more) but I used the Rust website because I find it really helpful whenever I am confused on how to write certain code. I used it to help me with iterations and some other random code questions. I used a Github repo to help me with plotly, it was really difficult for me but there are examples on github that were really helpful.