DS 210 Final Project Report- Qingyuan Kong

For the DS 210 Final Project, I choose a dataset that shows friendship on Facebook. The data consists of 4039 nodes, and shows the interconnected social circle between the users with undirected edges.

It is commonly believed that many relationship in human behaviors follow the power distribution, when one variable varies as a power of another. The power distribution will show a graph where there is a long tail, and a few variables that dominate. This distribution is also known as the 20-80 rule, where roughly 20% of the data is has the most importance, while the rest 80% is less significance, usually in quantity. For instance, there will be a large amount of people who don't spend money, for spend very little money on video games, while only a few people will spend a lot on video games. Using the Facebook dataset provided on SNAP, the project aims at exploring whether social circles follows a similar distribution, where people with a small amount of friends are significantly higher in quantity in comparison with people with a large amount of friends on Facebook. In this report, I will explain my approach, justify the outcome and briefly describe what I learnt from the process.

Approach

The initial hypothesis of this research project is decided that the degree of nodes in this data follows a power distribution. The approach is designed to be able to support or disprove the hypothesis as the final result.

To start the project I read the file, parsed it and converted the obtained data into an adjacency list so the nodes each node is connected with is described. Since the graph of the dataset is undirected, the process of creating an adjacency list for a directed graph will be needed to done again with the reversed data. This step could be find in the read file.rs file.

Next, the count of degree for each node is collected via the .len( ) function on each node in the adjacency list. The getdegrees( ) function that performs this in my code will return a vector. Each number in the vector is the amount of edges the respective node is connected to. For example, if the first number of this vector is 2, it suggests Node 0 is connected with 2 other nodes, that is to say, the degree of Node 0 is 2.

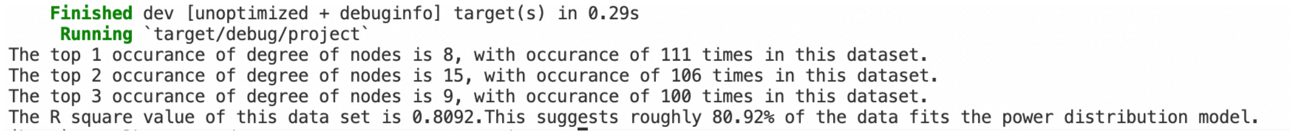
In order to continue the determination whether the data follows the power distribution, the occurrence of each degree is collected in the next function, count_occurance( ). The function returns a vector that contains tuples, which stores the following information in order: The value of the degree, and the number of times it occurs. The information is then sorted in reverse order for the value of degree. For example, if the degree 3 appeared 60 times, (3, 60) will be the third tuple in the returned vector. A test is written for this function in order to ensure the correctness. These steps could be found in the obtaindegrees.rs file.

Next, the x values and y values are extracted respectively from the vector. Since I have set our initial hypothesis to the data of node degrees does follow power distribution, I created two new vectors that stores ln x and ln y (natural log x and natural log y) values, since ln x and ln y will alter the relationship model for a power distribution to a linear one.

After the convergence, I write the final function rsquare( ) that calculates the R square value for ln x and ln y to see how well the data fits the model. The R square value is between 0 to 1. The higher the R square value indicates the better the data fits the model. Thus, if the function returns a high R square value, it would support my hypothesis that the distribution of degrees of nodes follows a power distribution. On the other hand, if the R square value is relatively low, it might suggest the data follows another distribution rather than power distribution.

The calculation for R square follows the formula of $[\ (n\Sigma xy - (\Sigma x)(\Sigma y)) / (\sqrt{n\Sigma x^2\text{-}(\Sigma x)^2} * \sqrt{n\Sigma y^2\text{-}(\Sigma y)^2})\ ]^2$. Vectors that store values for XY, X square and Y square is made respectively, and n is the length of the data. The final output of the value is shortened to four decimal places, since a higher precision in decimal places is not useful in this case. A test is made for this function, where the R square value for the testing data is calculator is calculated by an online calculator, and compared with the result calculated in this function. These steps could be found in the calculations.rs file. The code is executed in main.rs. The complexity is O(n^2).

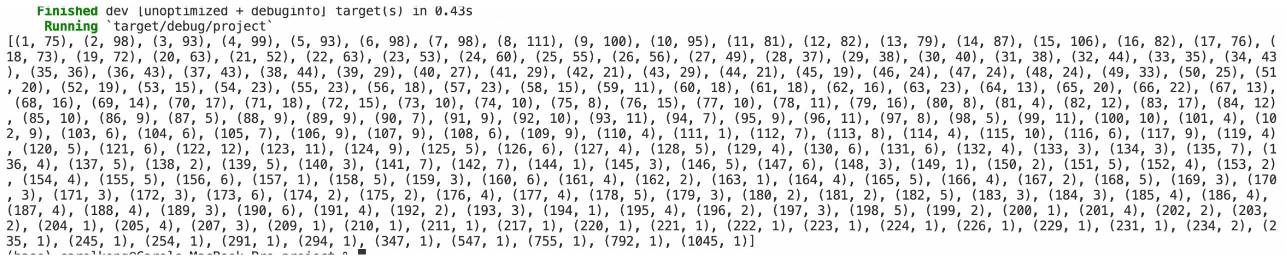
Results and Conclusions



```
     Finished dev [unoptimized + debuginfo] target(s) in 0.29s
      Running `target/debug/project`
The top 1 occurance of degree of nodes is 8, with occurance of 111 times in this dataset.
The top 2 occurance of degree of nodes is 15, with occurance of 106 times in this dataset.
The top 3 occurance of degree of nodes is 9, with occurance of 100 times in this dataset.
The R square value of this data set is 0.8092.This suggests roughly 80.92% of the data fits the power distribution model.
```

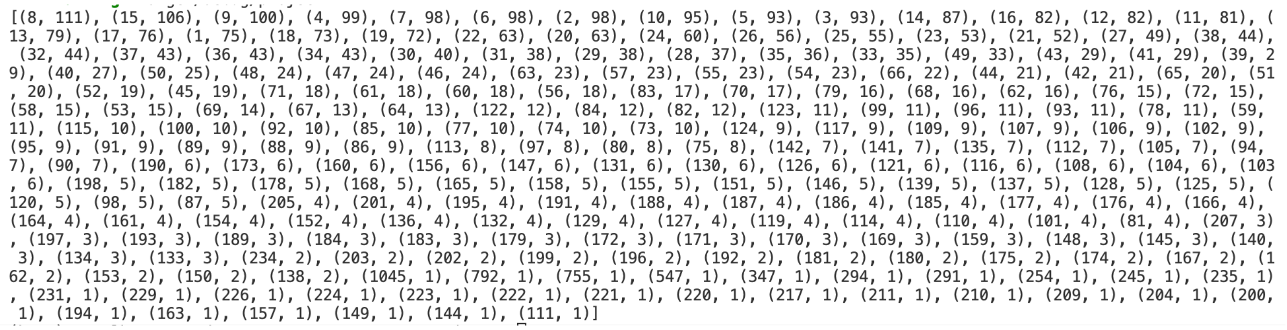
Figure 1: Final Output of the Project

The final R square value calculated for this data set is 0.8092. This suggests 80.92% of the data follows the power distribution model. While not an extremely high value, it is still high enough to support the hypothesis, that the degree of nodes follows a power distribution. That says, we could argue that social circles on Facebook roughly follows the 20-80 law, suggesting more people keep a small amount of friends in their social circles, while a few people have relatively large amount of friends in their social circles.



```
     Finished dev [unoptimized + debuginfo] target(s) in 0.43s
      Running `target/debug/project`
[(1, 75), (2, 98), (3, 93), (4, 99), (5, 93), (6, 98), (7, 98), (8, 111), (9, 100), (10, 95), (11, 81), (12, 82), (13, 79), (14, 87), (15, 106), (16, 82), (17, 76), (
18, 73), (19, 72), (20, 63), (21, 52), (22, 63), (23, 53), (24, 60), (25, 55), (26, 56), (27, 49), (28, 37), (29, 38), (30, 40), (31, 38), (32, 44), (33, 35), (34, 43
), (35, 36), (36, 43), (37, 43), (38, 44), (39, 29), (40, 27), (41, 29), (42, 21), (43, 29), (44, 21), (45, 19), (46, 24), (47, 24), (48, 24), (49, 33), (50, 25), (51
, 20), (52, 19), (53, 15), (54, 23), (55, 23), (56, 18), (57, 23), (58, 15), (59, 11), (60, 18), (61, 18), (62, 16), (63, 23), (64, 13), (65, 20), (66, 22), (67, 13),
 (68, 16), (69, 14), (70, 17), (71, 18), (72, 15), (73, 10), (74, 10), (75, 8), (76, 15), (77, 10), (78, 11), (79, 16), (80, 8), (81, 4), (82, 12), (83, 17), (84, 12)
, (85, 10), (86, 9), (87, 5), (88, 9), (89, 9), (90, 7), (91, 9), (92, 10), (93, 11), (94, 7), (95, 9), (96, 11), (97, 8), (98, 5), (99, 11), (100, 10), (101, 4), (10
2, 9), (103, 6), (104, 6), (105, 7), (106, 9), (107, 9), (108, 6), (109, 9), (110, 4), (111, 1), (112, 7), (113, 8), (114, 4), (115, 10), (116, 6), (117, 9), (119, 4)
, (120, 5), (121, 6), (122, 12), (123, 11), (124, 9), (125, 5), (126, 6), (127, 4), (128, 5), (129, 4), (130, 6), (131, 6), (132, 4), (133, 3), (134, 3), (135, 7), (1
36, 4), (137, 5), (138, 2), (139, 5), (140, 3), (141, 7), (142, 7), (144, 1), (145, 3), (146, 5), (147, 6), (148, 3), (149, 1), (150, 2), (151, 5), (152, 4), (153, 2)
, (154, 4), (155, 5), (156, 6), (157, 1), (158, 5), (159, 3), (160, 6), (161, 4), (162, 2), (163, 1), (164, 4), (165, 5), (166, 4), (167, 2), (168, 5), (169, 3), (170
, 3), (171, 3), (172, 3), (173, 6), (174, 2), (175, 2), (176, 4), (177, 4), (178, 5), (179, 3), (180, 2), (181, 2), (182, 5), (183, 3), (184, 3), (185, 4), (186, 4),
(187, 4), (188, 4), (189, 3), (190, 6), (191, 4), (192, 2), (193, 3), (194, 1), (195, 4), (196, 2), (197, 3), (198, 5), (199, 2), (200, 1), (201, 4), (202, 2), (203,
2), (204, 1), (205, 4), (207, 3), (209, 1), (210, 1), (211, 1), (217, 1), (220, 1), (221, 1), (222, 1), (223, 1), (224, 1), (226, 1), (229, 1), (231, 1), (234, 2), (2
35, 1), (245, 1), (254, 1), (291, 1), (294, 1), (347, 1), (547, 1), (755, 1), (792, 1), (1045, 1)]
```

Figure 2: List of Occurrence Count, in order of number of edges for node



```
[(8, 111), (15, 106), (9, 100), (4, 99), (7, 98), (6, 98), (2, 98), (10, 95), (5, 93), (3, 93), (14, 87), (16, 82), (12, 82), (11, 81), (
13, 79), (17, 76), (1, 75), (18, 73), (19, 72), (22, 63), (20, 63), (24, 60), (26, 56), (25, 55), (23, 53), (21, 52), (27, 49), (38, 44),
 (32, 44), (37, 43), (36, 43), (34, 43), (30, 40), (31, 38), (29, 38), (28, 37), (35, 36), (33, 35), (49, 33), (43, 29), (41, 29), (39, 2
9), (40, 27), (50, 25), (48, 24), (47, 24), (46, 24), (63, 23), (57, 23), (55, 23), (54, 23), (66, 22), (44, 21), (42, 21), (65, 20), (51
, 20), (52, 19), (45, 19), (71, 18), (61, 18), (60, 18), (56, 18), (83, 17), (70, 17), (79, 16), (68, 16), (62, 16), (76, 15), (72, 15),
(58, 15), (53, 15), (69, 14), (67, 13), (64, 13), (122, 12), (84, 12), (82, 12), (123, 11), (99, 11), (96, 11), (93, 11), (78, 11), (59,
11), (115, 10), (100, 10), (92, 10), (85, 10), (77, 10), (74, 10), (73, 10), (124, 9), (117, 9), (109, 9), (107, 9), (106, 9), (102, 9),
(95, 9), (91, 9), (89, 9), (88, 9), (86, 9), (113, 8), (97, 8), (80, 8), (75, 8), (142, 7), (141, 7), (135, 7), (112, 7), (105, 7), (94,
7), (90, 7), (190, 6), (173, 6), (160, 6), (156, 6), (147, 6), (131, 6), (130, 6), (126, 6), (121, 6), (116, 6), (108, 6), (104, 6), (103
, 6), (198, 5), (182, 5), (178, 5), (168, 5), (165, 5), (158, 5), (155, 5), (151, 5), (146, 5), (139, 5), (137, 5), (128, 5), (125, 5), (
120, 5), (98, 5), (87, 5), (205, 4), (201, 4), (195, 4), (191, 4), (188, 4), (187, 4), (186, 4), (185, 4), (177, 4), (176, 4), (166, 4),
(164, 4), (161, 4), (154, 4), (152, 4), (136, 4), (132, 4), (129, 4), (127, 4), (119, 4), (114, 4), (110, 4), (101, 4), (81, 4), (207, 3)
, (197, 3), (193, 3), (189, 3), (184, 3), (183, 3), (179, 3), (172, 3), (171, 3), (170, 3), (169, 3), (159, 3), (148, 3), (145, 3), (140,
 3), (134, 3), (133, 3), (234, 2), (203, 2), (202, 2), (199, 2), (196, 2), (192, 2), (181, 2), (180, 2), (175, 2), (174, 2), (167, 2), (1
62, 2), (153, 2), (150, 2), (138, 2), (1045, 1), (792, 1), (755, 1), (547, 1), (347, 1), (294, 1), (291, 1), (254, 1), (245, 1), (235, 1)
, (231, 1), (229, 1), (226, 1), (224, 1), (223, 1), (222, 1), (221, 1), (220, 1), (217, 1), (211, 1), (210, 1), (209, 1), (204, 1), (200,
 1), (194, 1), (163, 1), (157, 1), (149, 1), (144, 1), (111, 1)]
```

Figure 3: List of Occurrence Count, in order of number of occurrence

In addition, if looked at the initial degree occurrence count vector as shown above, it is true that relatively high occurrences is found in smaller numbers in the range of 1 to 20. After 20, the amount of friends/edges each node have starts dropping significantly. The highest number of occurrence of degree of each node(as shown in Figure 3) is 8. All degree of nodes with occurrences number lower or equal to 5 are above 85, the lowest being 87 degrees, suggesting that most people in this dataset don't have over 85 friends on Facebook. Therefore, it could be suggested that the distribution of degree of nodes follows a power distribution.

Takeaways and Future Changes

I explored how to analyze graphs in Rust with this project. I think my greatest challenge while the process is to construct the approach that could explore my research question and provide insightful outcomes. I finally decided to use R square value as the way to justify my output, since it is a direct number that could show to what extent is the data fitting the initial model. I am glad that decision process in the approach stage helped me made the later stages of coding easier, since I already have a clear vision in mind of what I intended to do with this project.

There are also some reflections I made on changes I could have done to make the project better. For example, I could have created a graph that shows the degree of the nodes in addition the the R square value, so the final output would be visual. While the R square value is direct support of the hypothesis, a visual output would be clearer and easier to understand if I have included this as well.