

Assignment-3

Mona Singh : MT21053

Nehal Chourasia : MT21056

Question 1 - [45 Points] Link Analysis

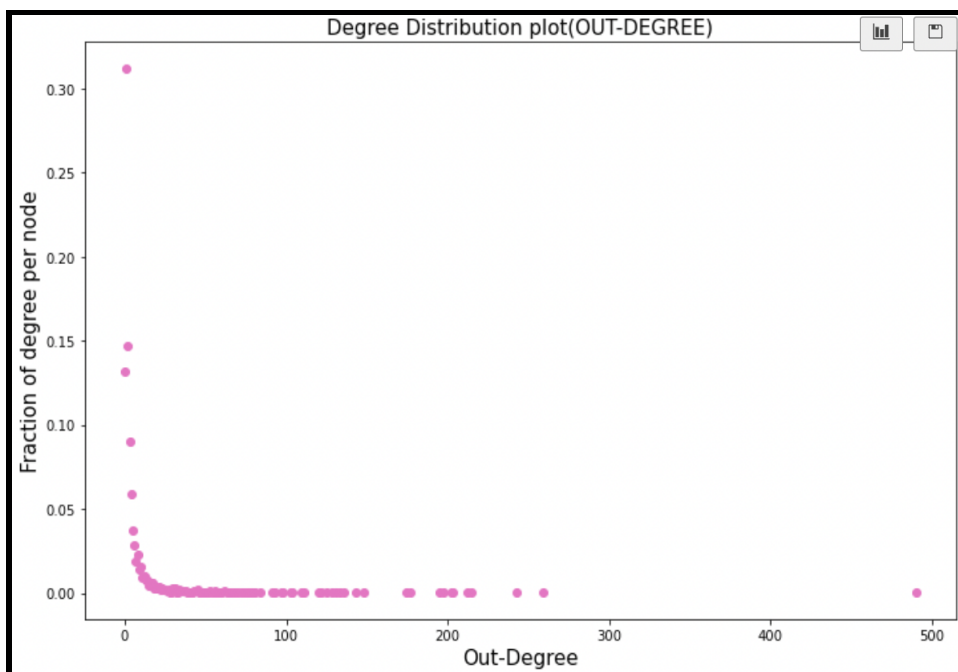
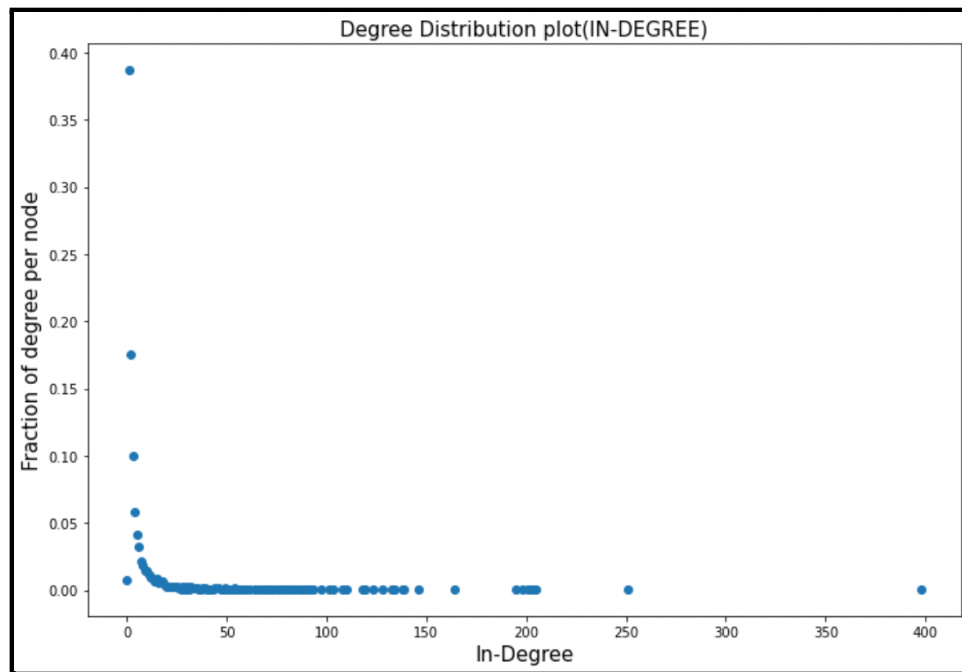
Dataset Description:

The dataset used is the *Bitcoin Alpha trust weighted signed network*. This is a dataset that describes the network of who trusts whom, in the trading of bitcoin using the application **Bitcoin Alpha**. Bitcoin trading is done anonymously, therefore there is a need to maintain the records of users to prevent frauds and lower risks.

Number of Nodes	: 3,783
Number of Edges	: 24,186
Avg In-degree	: 6.393338620142744
Avg. Out-Degree	: 6.393338620142744
Node with Max In-degree	: 0, degree 398
Node with Max Out-degree	: 0, degree 490
The density of the network	: 0.0012333558812539552

1. [5 points] Plot degree distribution of the network (in case of a directed graph, plot in-degree and out-degree separately).

To calculate the degree distribution of the network, firstly have calculated the frequency of every unique in and out degrees in the graph and then normalize the values. Store them in a dictionary and plot the x axis as the in and out degrees and y axis as the fraction of degree per node.



2. [10 points] Calculate the local clustering coefficient of each node and plot the clustering-coefficient distribution of the network.

For the clustering coefficient, the heuristic used is the following:

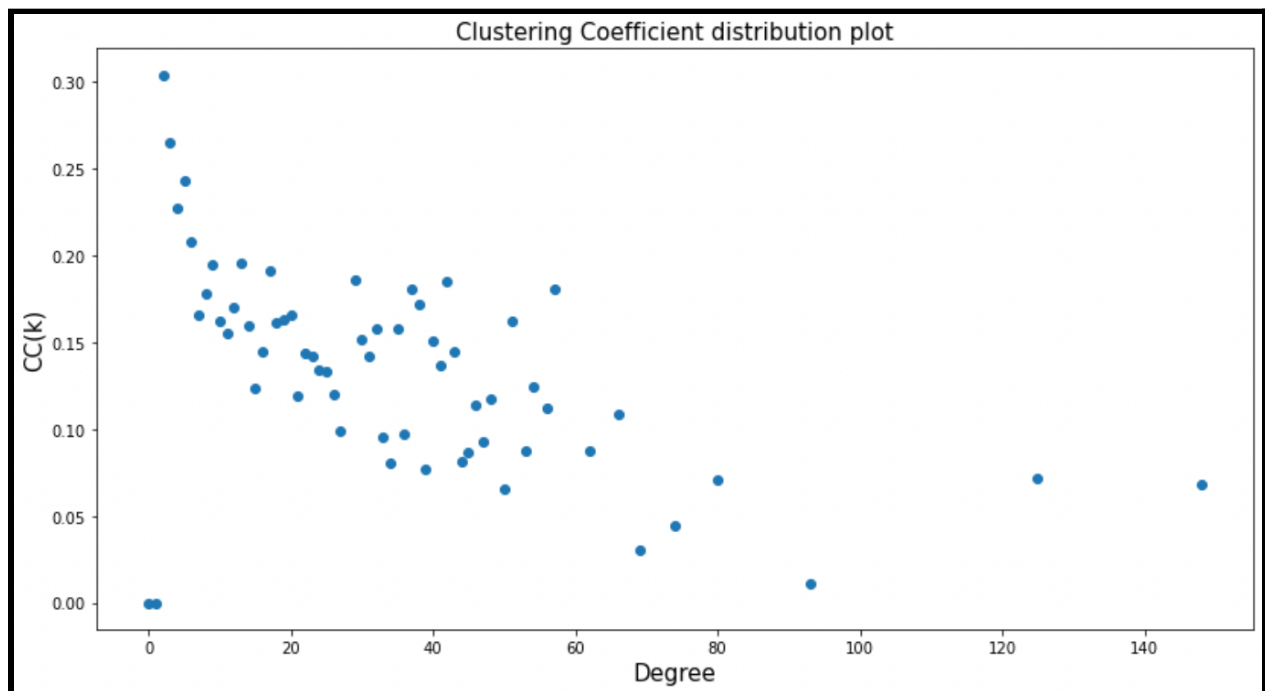
$$CC = \frac{2 * (\text{Number of links between neighboring nodes of } v)}{\text{Degree}(v) * (\text{Degree}(v) - 1)}$$

Degree(v) is the degree of node v.

To calculate the clustering coefficient, firstly, iterate over all the nodes and find the number of links of neighboring nodes using an adjacency list. Store this information in a dictionary.

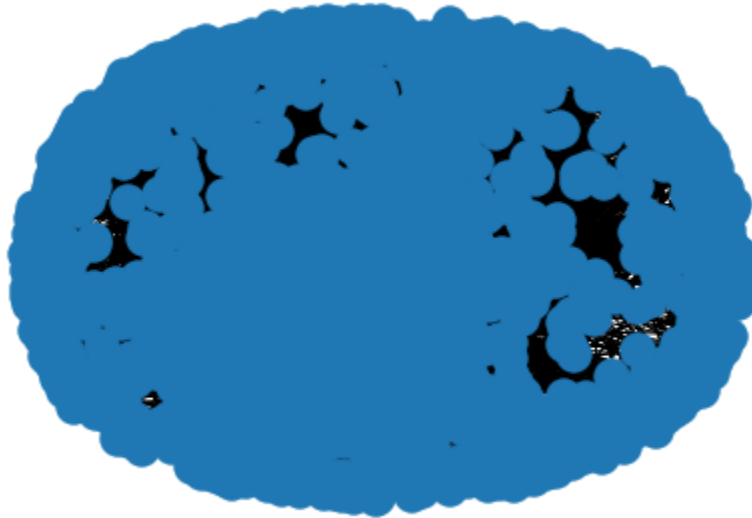
Now to plot the graph, calculate the average clustering coefficient value for every unique degree in the graph and plot the values.

The plot :



Question 2 - [35 points] PageRank, Hubs and Authority

For the dataset chosen in the above question, calculate the following:



1. [15 points] PageRank score for each node

To calculate the pagerank for each node, the page rank of each node is initialized as 1 and then updated as the below formula.

$$PR(P_i) = \frac{(d)}{n} + (1 - d) \times \sum_{I_{j,i} \in E} PR(P_j) / \text{Outdegree}(P_j)$$

D(damping factor)=0.1~0.15
n=|page set|

Also calculated the page rank from networkx.pagerank algorithm.

```
[('3480', 3.965107057890563e-05), ('3999', 3.965107057890563e-05),  
('4888', 3.965107057890563e-05), ('5029', 3.965107057890563e-05), ('5415',
```

```
3.965107057890563e-05), ('5660', 3.965107057890563e-05), ('5837',
3.965107057890563e-05), ('6014', 3.965107057890563e-05), ('6123',
3.965107057890563e-05), ('6131', 3.965107057890563e-05), ('6157',
3.965107057890563e-05), ('6166', 3.965107057890563e-05), ('6257',
3.965107057890563e-05), ('6290', 3.965107057890563e-05), ('6317',
3.965107057890563e-05), ('6336', 3.965107057890563e-05), ('6434',
3.965107057890563e-05), ('6644', 3.965107057890563e-05), ('6667',
3.965107057890563e-05), ('6736', 3.965107057890563e-05), ('6786',
3.965107057890563e-05), ('6958', 3.965107057890563e-05), ('7063',
3.965107057890563e-05),
```

The pagerank for all the nodes are coming to be relative same.

```
[('3480', 7.225589930263372e-06), ('3999', 7.225589930263372e-06),
('4888', 7.225589930263372e-06), ('5029', 7.225589930263372e-06), ('5415',
7.225589930263372e-06), ('5660', 7.225589930263372e-06), ('5837',
7.225589930263372e-06), ('6014', 7.225589930263372e-06), ('6123',
7.225589930263372e-06), ('6131', 7.225589930263372e-06), ('6157',
7.225589930263372e-06), ('6166', 7.225589930263372e-06), ('6257',
7.225589930263372e-06), ('6290', 7.225589930263372e-06), ('6317',
7.225589930263372e-06), ('6336', 7.225589930263372e-06), ('6434',
7.225589930263372e-06), ('6644', 7.225589930263372e-06), ('6667',
7.225589930263372e-06), ('6736', 7.225589930263372e-06), ('6786',
7.225589930263372e-06), ('6958', 7.225589930263372e-06), ('7063',
7.225589930263372e-06)
```

2. [15 points] Authority and Hub score for each node

To calculate the authority and hub for each node, both value for each node is initialized as 1 and then updated as the below formula.

- New auth = the sum of the hub of all of its parents
- New hub = the sum of the auth of all of its children

Authority Score

```
[('1389', 0.0), ('1870', 0.0), ('3228', 0.0), ('3271', 0.0),  
('3388', 0.0), ('3480', 0.0), ('3999', 0.0), ('4888', 0.0), ('5029',  
0.0), ('5415', 0.0),
```

Hub Score

```
[('1019', 0.0), ('1028', 0.0), ('1070', 0.0), ('1075', 0.0),  
('1092', 0.0), ('1097', 0.0), ('1110', 0.0), ('1138', 0.0), ('1155', 0.0),  
('1176', 0.0),
```

Also found out authority and hub using the **networkx.hits** algorithm.

Authority Score

```
[('1591', -2.162719811416757e-21), ('978', -1.6166656365784312e-21),  
('1976', -7.283381273336779e-22), ('3480', 0.0), ('3999', 0.0), ('4888',  
0.0), ('5029', 0.0), ('5415', 0.0), ('5660', 0.0), ('5837', 0.0),
```

Hub Score

```
[('1929', -1.1640879563709423e-22), ('2578',  
-1.1640879563709423e-22), ('127', -1.0206433313834807e-22), ('5837',  
-8.230009842225695e-23), ('3388', -7.798609585752349e-23), ('294',  
-6.786180013201605e-23), ('760', -5.035154824663014e-23), ('1135',  
-1.0254975460402137e-23), ('1389', -6.66089439398859e-24), ('1870',  
-4.6156942799635946e-24),
```

[5 points] Compare the results obtained from both the algorithms in parts 1 and 2 based on the node scores.

Pagerank

```
[('3480', '3999', '4888', '5029', '5415', '5660', '5837', '6014', '6123',  
'6131')
```

Authority Score

```
[ ('1591', '978', '1976', '3480', '3999', '4888', '5029', '5415', '5660',  
'5837',
```

Hub Score

```
[ ('1929', '2578', '127', '5837', '3388', '294', '760', '1135',  
'1389', '1870',
```

From the sorted 10 values from each , we can see that pagerank and authority score have various same nodes.

Also it was seen that every time hits algorithm is run , it returns different score for authority and hub, whereas pagerank returned the same result everytime.

References:

<https://towardsdatascience.com/pagerank-3c568a7d2332>

<https://towardsdatascience.com/hits-algorithm-link-analysis-explanation-and-python-implementation-61f0762fd7cf>