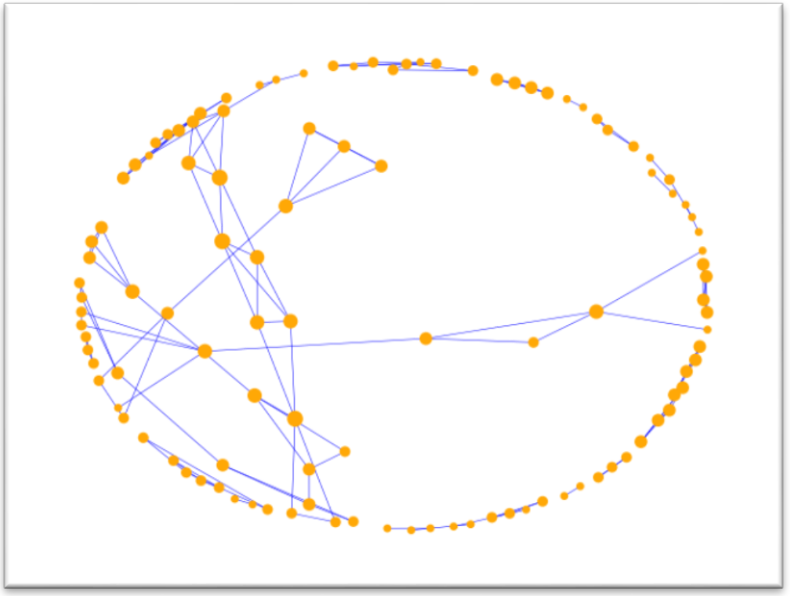
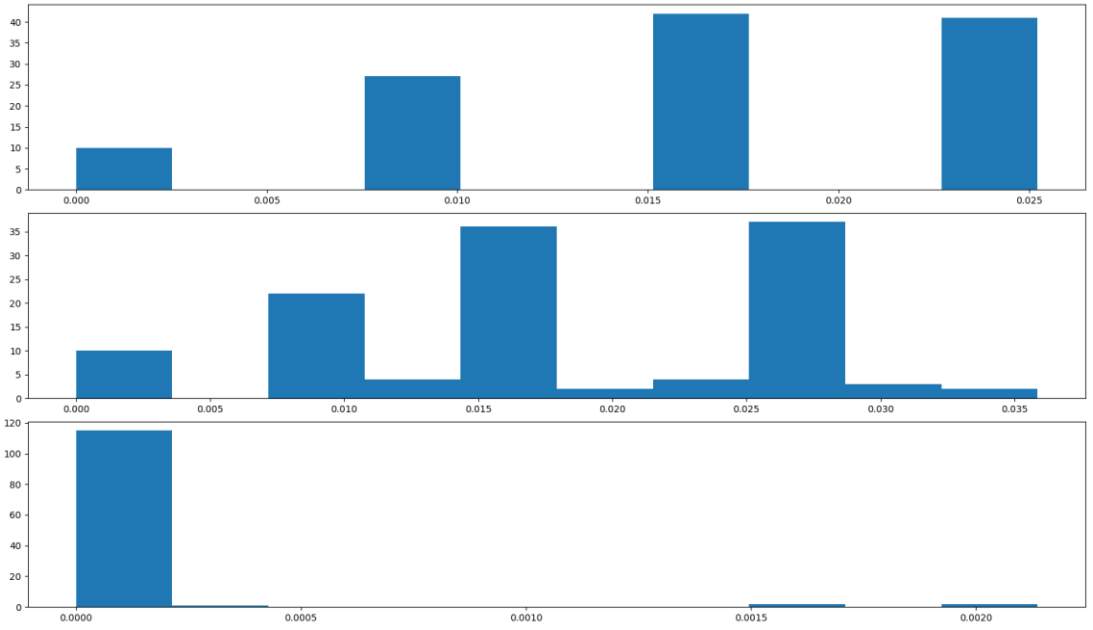
REPORT FILE OF ALGORITHM AND METHODS OF DATA MINING HOMEWORK 4 GROUP 17

In this file we will discuss about the results we got in the part 2 of the homework (Statistics and Visualizations) and in the part 3 (Generalized version of the Erdös Number).

In the first section, as requested by the task and based on the full_db, this is the network we obtain giving as input the id_conference=3052



After the network plot we have the plot of the centrality measure:



So, analyzing one row at time we have:

- Degree centrality (first row) = "Important row are involved in large number of interactions". Indeed the first plot show us the degree centrality ***normalized***, or in other words the frequency of the nodes that show a certain number of interaction/connection with other nodes of the subgraph.
  As shown by the plot only 10 nodes show a minimum value of interaction, we think maybe about 1-2 link(s), then about 25 nodes show a degree centrality value between 0.008 and 0.01 with which we can guess that those nodes have about 3-5 links per node. The other two columns present very high frequency and relatively high value of centrality degree: in fact we can say they represent, in particular the last column, the node that are more "involved" in the subgraph because of their high number of connection. According to this measure (not in generally) we can say that the last column represent ***the most important points*** of the subgraph.

$$Degree\ Centrality = \frac{degree(u)}{n-1}$$

- Closeness centrality (second row) = "An important row is generally close to the other and can communicate quickly with them". This is a measure of the proximity of a node to the rest of the network. Given that when distance grows up centrality goes down the function is

$$Closeness\ Centrality = \frac{n-1}{\sum d(u,v)}$$

  Looking at the value of the plot, we see 2/3 high frequency column that shows node with a mid-range value of closeness: we can say that generally, the graph is not so concentrated/dense but we are facing a quite sparse graph. Just very few nodes (5 more or less) shows the highest values of closeness (0.029-0.036). The first plot of the network is indeed the proof of our supposes: the nodes are all spreaded among an ellipsis curve with just few nodes lie in the middle of the ellipsis that can have an higher value of closeness. In this case, those inner nodes, according to closeness centrality, are the most important nodes.

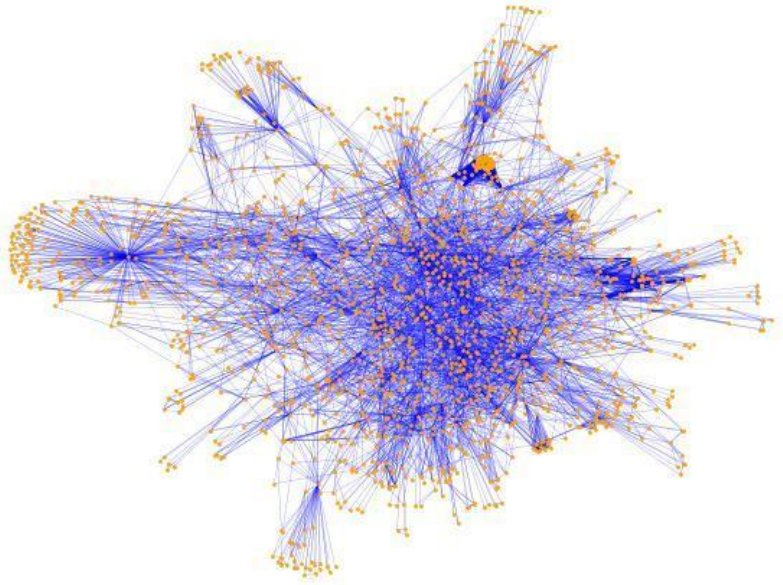
- Betweenness Centrality (third row) = "important nodes will lie on a high proportion of paths between other nodes. U, V are nodes of a graph, if they want to communicate they will ideally use the shortest path; any node J that is in that path has the ability to affect the communication. For this reason, any J node of the graph that belongs to such a lot paths is central in the sense that it can affect a lot the communications. In other words, the betweenness centrality represent the grade with which a node can control communications in the network."

$$Betweenness\ Centrality = \sum_{s \neq t \neq v} \frac{\sigma_{s,,t}(v)}{\sigma_{s,t}}$$

  The denominator represent all the possible shortest paths with which the nodes s, t can communicate, the numerator represent the number of shortest path that pass through the node v.
  Looking at the third plot, we notice how almost all the nodes are meaningless according to this measure. Just about 1-2 nodes, as shown by the last tiny column can be crucial for the inner communication.

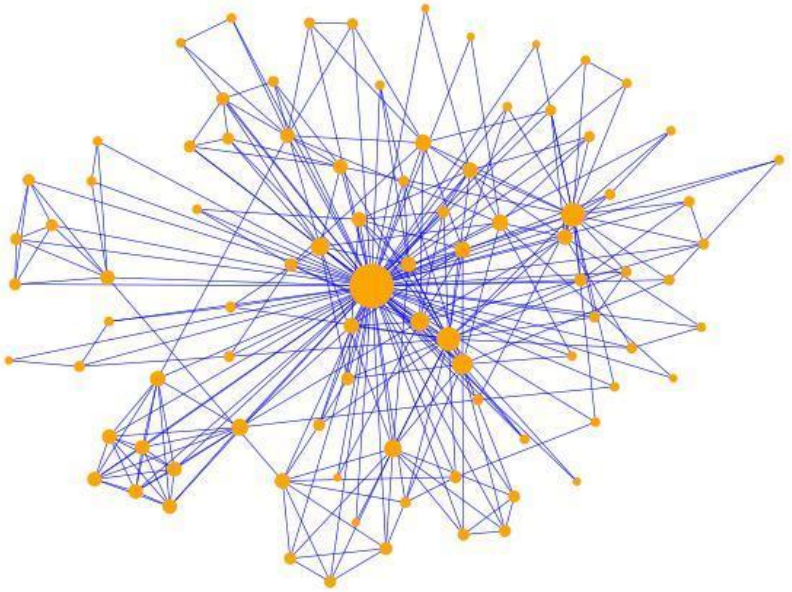
In the second section, given an ***author id*** and a ***distance d*** (as an integer) the function has to give back the subgraph induced by the nodes that have ***distance d.*** Given as input:

- Author id=18262
- d=2



Given the vast number of nodes and edges was impossible to maintain the original settings, we had to set a standard size for the nodes of 1 and width of edges set to 0.1. For a clear and simpler vision now we take a look to the same graph but with distance d=1.

- Author id=18262
- d=1



In the first section of part 3 it was asked us to find the path (given by the sum of all the weights of the steps necessary) between a given author and the target author ***Aris*** with ***author id = 256176.*** Taking the ***author id = 18262*** as test author and running the algorithm we get as result:

- Erdös Number: 1.8545454545454545
- Time: 0.6754329204559326 seconds

Second section of part 3 asked us to find the GroupNumber value between each node of the graph and a subset of nodes (with cardinality smaller than 21) typed by the user. Again, using

Input

- Subset I = 18262, 256176

Output

- Time = 25.884556770324707 seconds
- NOTICE The real output was the print of all the GroupNumber value of the nodes in the console; given the extremely large length of the print we have just reported the time taken by the algorithm to carry out the task.
- Example of output:

  Group number (1) = 5.361455205205205
  Group number (2) = 4.611455205205205
  Group number (3) = 5.111455205205205
  Group number (4) = 3.932923384791309
  Group number (5) = 4.511351807089653