Course: IDS 564 Social Media Network Analysis
Dr. Tafti Ali
Fall 2019

Submitted by
Vaidehi Gosawi
662635635

Advanced Lab 2 Submission

Implications of Network Structure of the SAP Online Knowledge Community Platform

The CSV file provided to us has the particulars about the Network Structure of the SAP Online Knowledge Community Platform. The CSV file has two columns, in which the right-hand column signifies the ID of a user who posts a question that starts a new thread in a SAP community user forum and the left-hand column represents the ID of a user who provides an answer to the posted question. The data set provided to us is a sub-network based on 10% of nodes drawn from the dataset generously provided by Prof. Peng Huang, of the University of Maryland. When we plot the network graph we obtain Figure 1 and this graph contains multiple duplicate values and does not qualify as a "simple graph". The edge count and the vertex count for Figure 1 are 6090 and 3415 respectively.

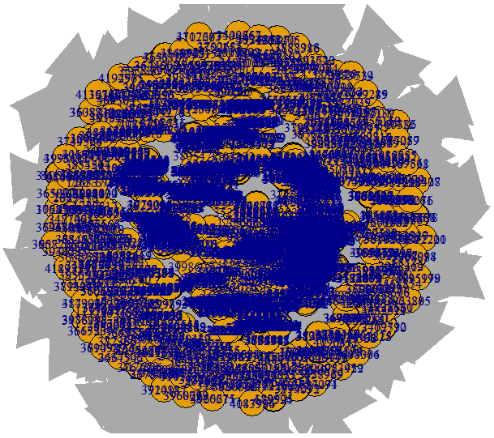


Figure 1

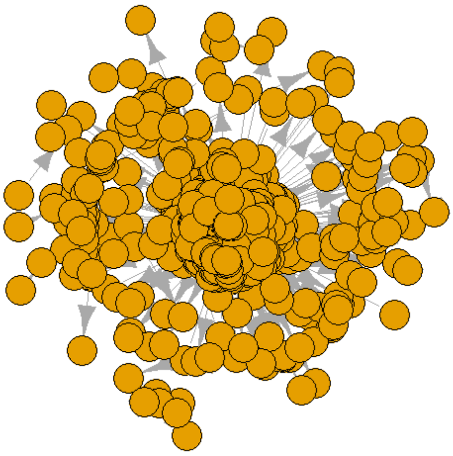
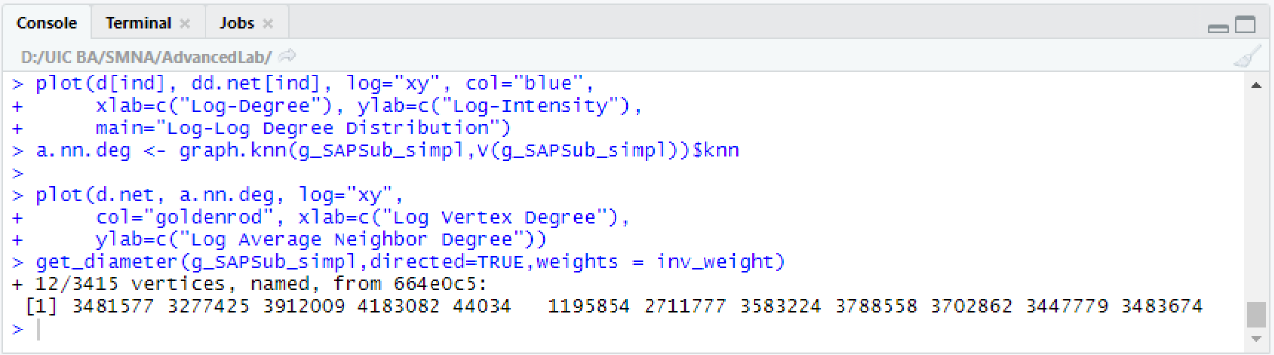


Figure 2

Next, we attempted to reduce the multiple duplicate values and produce a graph that contains a lot less edges than before. We used the R command simplify and set the values of the parameters *remove.multiple* and *remove.loops* to True. We obtain the graph shown in Figure 2 which displays the vertices and the directed edges clearly. We see a reduction in the edge count to 4120 with no change in vertex count. The simplified network graph provides us with a lot of important information. It is necessary to realize the different features of the network so that it helps us identify relevant information about the communities present in the network.

Some of the network properties that I have explored on *g_SAPSub_simpl* are its diameter, which gives the longest path traversed in the network, *get_diameter* which produces the path, *mean_density* which gives returns the average number of edges between any two nodes in the network, *edge_density* which is the proportion of edges in the network over all possible edges that could exist, reciprocity and transitivity.

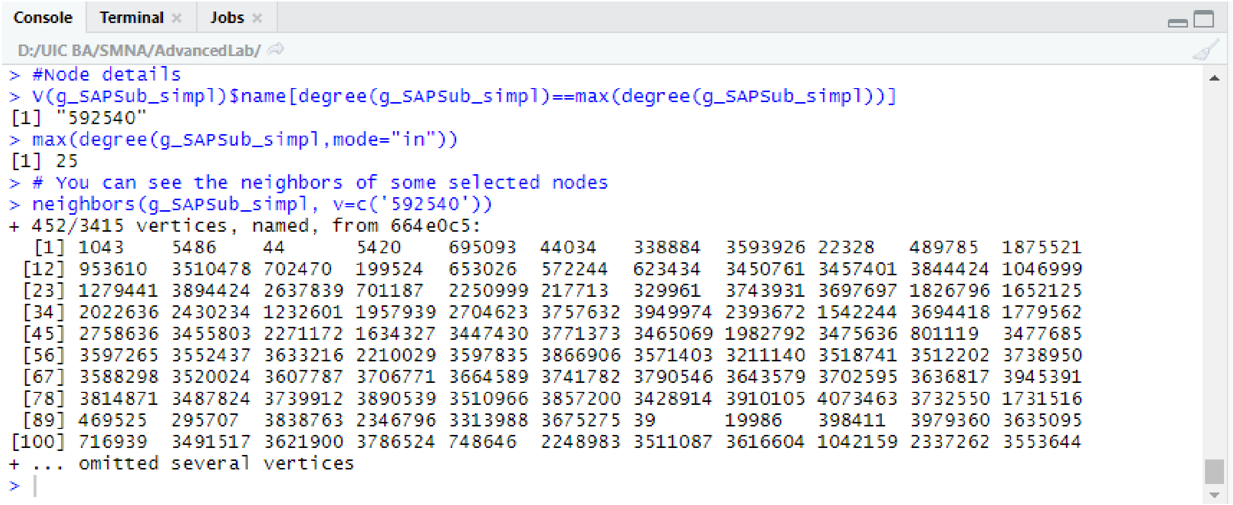


```
> plot(d[ind], dd.net[ind], log="xy", col="blue",
+      xlab=c("Log-Degree"), ylab=c("Log-Intensity"),
+      main="Log-Log Degree Distribution")
> a.nn.deg <- graph.knn(g_SAPSub_simpl,V(g_SAPSub_simpl))$knn
>
> plot(d.net, a.nn.deg, log="xy",
+      col="goldenrod", xlab=c("Log Vertex Degree"),
+      ylab=c("Log Average Neighbor Degree"))
> get_diameter(g_SAPSub_simpl,directed=TRUE,weights = inv_weight)
+ 12/3415 vertices, named, from 664e0c5:
 [1] 3481577 3277425 3912009 4183082 44034   1195854 2711777 3583224 3788558 3702862 3447779 3483674
> 
```

Figure 3

Figure 3 displays the specifics of the network and we see the longest path traversed is in the path 3481577-3277425-3912009-4183082-44034 -1195854-2711777-3583224-3788558-3702862-3447779-3483674. As the network is complex, reciprocity =0.059 tells us that there is likelihood of occurring double links with opposite directions between vertex pairs. The transitivity of the SAP Network is 0.0099, which is very less which means that there is very little probability that the adjacent vertices of the vertex in SAP network are connected.

Now, we proceed on to study the nodes of the network. A fascinating item to study in this network is which node has supreme number of edges which gives us an idea that the resulting node will be the most interactive one. This feature is known as the degree of the node and we list out the neighbors of the node that has the highest degree. See Figure 4 for more details.



```
> #Node details
> V(g_SAPSub_simpl)$name[degree(g_SAPSub_simpl)==max(degree(g_SAPSub_simpl))]
[1] "592540"
> max(degree(g_SAPSub_simpl,mode="in"))
[1] 25
> # You can see the neighbors of some selected nodes
> neighbors(g_SAPSub_simpl, v=c('592540'))
+ 452/3415 vertices, named, from 664e0c5:
  [1] 1043    5486    44      5420    695093  44034   338884  3593926 22328   489785  1875521
 [12] 953610  3510478 702470  199524  653026  572244  623434  3450761 3457401 3844424 1046999
 [23] 1279441 3894424 2637839 701187  2250999 217713  329961  3743931 3697697 1826796 1652125
 [34] 2022636 2430234 1232601 1957939 2704623 3757632 3949974 2393672 1542244 3694418 1779562
 [45] 2758636 3455803 2271172 1634327 3447430 3771373 3465069 1982792 3475636 801119  3477685
 [56] 3597265 3552437 3633216 2210029 3597835 3866906 3571403 3211140 3518741 3512202 3738950
 [67] 3588298 3520024 3607787 3706771 3664589 3741782 3790546 3643579 3702595 3636817 3945391
 [78] 3814871 3487824 3739912 3890539 3510966 3857200 3428914 3910105 4073463 3732550 1731516
 [89] 469525  295707  3838763 2346796 3313988 3675275 39      19986   398411  3979360 3635095
[100] 716939  3491517 3621900 3786524 748646  2248983 3511087 3616604 1042159 2337262 3553644
+ ... omitted several vertices
> 
```

Figure 4

There can be different subsets of the network that are more connected to each other than to the rest of the network. We can use the community detection algorithms to detect these subsets. One of the best algorithms for community detection is fast greedy but it can be used only when the graph is undirected.

Hence, we try to implement walktrap algorithm (Figure 5), which finds communities through a series of short random walks and infomap method (Figure 6) which attempts to map the flow of information in a network, and the different clusters in which information may be retained for longer periods. We have omitted the node labels for readability of the graphs.

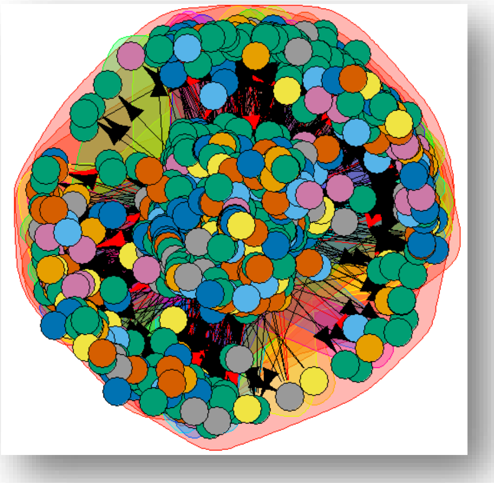


Figure 5

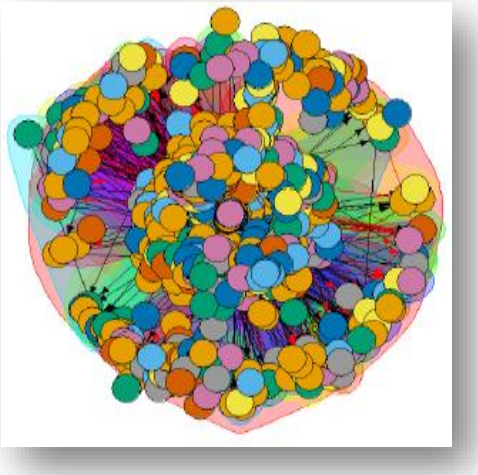


Figure 6

A clique in a graph G, refers to a complete subgraph. In the given network we have 335 cliques of length 3 nodes which corresponds to 335 triadic closure in the network. We also have 39 square and 5 pentagonal graphs in our network.

Betweenness of nodes measures brokerage or gatekeeping potential. It is (approximately) the number of shortest paths between nodes that pass through a particular node. A high betweenness indicates that it is a very important node and lot of information pass through that node in the network. As we can see from Figure 7, the Average Betweenness of edges has a positive linear relationship to Weight of the edges and a negative linear relationship with the clustering of the edges.

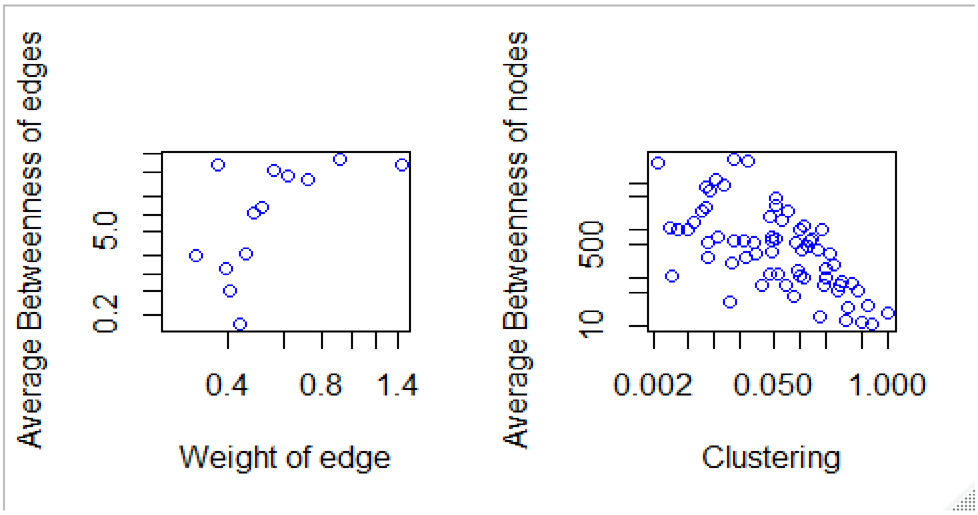


Figure 7

A node with a high degree will have high betweenness because that node will be the most collaborating one which will be communicating with a lot of other nodes. Hence, we observe a positive linear relationship amongst them. If a cluster consists of a lot of nodes, it will have a high value of degree. It might also happen that a small cluster is interacting with another very large cluster and can have a high degree. Embeddedness of a network is how deeply the node is placed in the network. The relationship of the various measures is as below.

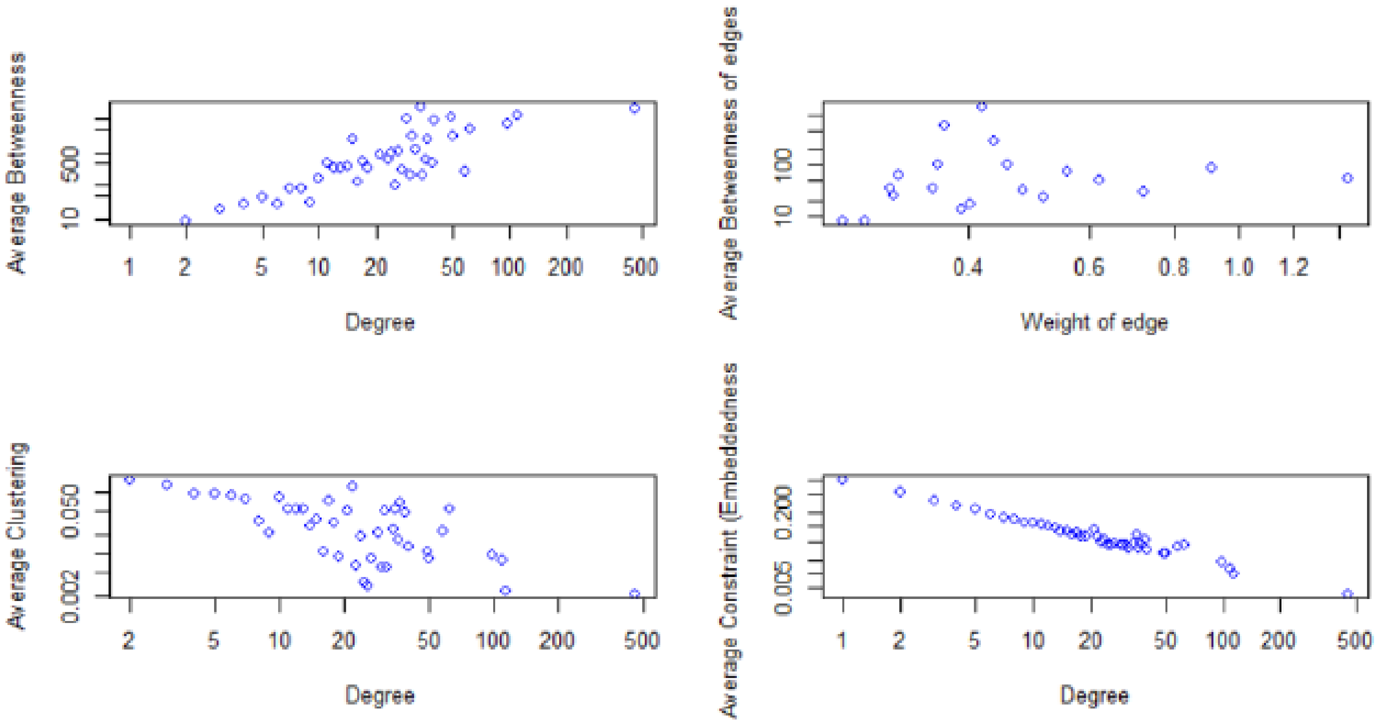


Figure 8