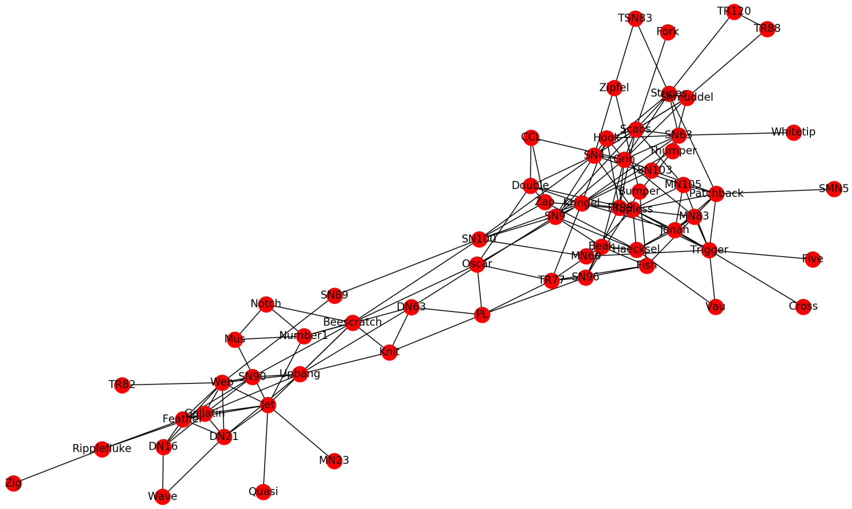
Data Analytics - Assignment-III

Anirban Biswas (Sr.No. - 14382)

September 21, 2018

1 Community Detection using Fiedler Vector

Given the graph corresponding to the dolphin-interaction network, the task is to find two communities by obtaining fiedler vector. This is how the network looks like:



Approach

In this approach, firstly I have used *networkx* package to read the graph and then converted it into an adjacency matrix. The dolphin network has a total of 62 nodes(i.e. dolphins) and 159 edges between them. Here the given graph is assumed to be undirected, hence the adjacency matrix will be symmetric. Let us consider the adjacency matrix is A.

Next step is to calculate the *Laplacian* of the graph and it is calculated using the following equation: $L = D - A$, where L is Laplacian matrix and D is the diagonal matrix obtained by $D_{ii} = \sum_{j=1}^{62} A_{ij}$. Note that, the laplacian matrix is also a symmetric matrix.

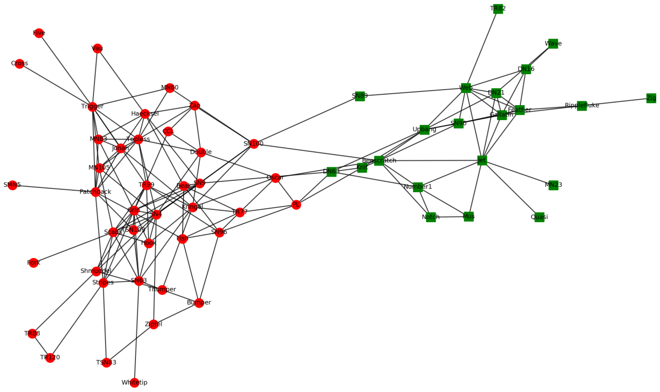
Our goal is to find out the eigen vector corresponding to the second smallest eigen value of L. This eigen vector has a special name - **Fiedler Vector**. I have used **fiedler_vector** API of *networkx* package directly to find out the fiedler vector. The vector is 62-dimensional, each element corresponds to one dolphin.

How to obtain two communities

Once I have got the fiedler vector, I have used K-Means algorithm to cluster them into two different communities. The reason behind the use of K-means is it basically tries to reduce within-cluster variance between the data points, hence most similar datapoints (read dolphins with highest interactions mong themselves) will belong to same cluster, which is essentially what we want.

Result

It turns out that the two clusters are of unequal sizes, **21** and **41** respectively. The following figure shows the two communities:



2 Community Detection using Louvain Method

Given the graph corresponding to the dolphin-interaction network, the task is to find two communities by obtaining Louvain method.

Approach

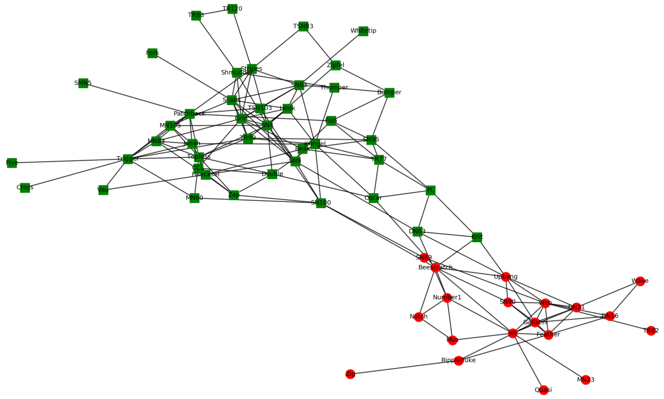
Similar to previous problem, in this approach also, firstly I have used *networkx* package to read the graph. To obtain communities based on Louvain method, I made use of the standard library available for python - *python-louvain*. It takes the graph as input and gives out the best partitions in the graph (it uses modularity scores internally). In particular for bottlenose dolphin network, it partitions into 5 (sometimes 4 also) clusters.

How to obtain two communities

Although the resulting best partition has 5(or 4 maybe) clusters, our goal is to obtain two communities finally. For that, what I have done is I took out two largest clusters as my base clusters. Then I kept on merging smaller clusters with one of these big clusters greedily, based on which merger gives maximum modularity. I kept on doing it in a greedy manner, till the point when number of clusters reduces to two.

Result

It turns out that the two clusters are of unequal sizes, **18** and **44** respectively. The following figure shows the two communities:


TR88
TR120
TSN83
Whitetip
Fork
Shmuddel
Stripes
Zipfel
SMN5
SN63
Thumper
Bumper
Scabs
TSN103
Hook
Patchback
Grin
Fish
MN105
SN4
TR99
SN96
MN83
Jonah
Kringel
Beak
Five
Trigger
Topless
SN9
TR77
CCL
Haecksel
Double
Cross
Vau
Zap
PL
MN60
SN100
Oscar
DN63
Knit
SN89
Beescratch
Upbang
Wave
Number1
SN90
Web
DN21
Notch
DN16
Gallatin
Jet
Feather
TR82
Mus
Ripplefluke
Zig
MN23
Quasi