ECE 232E (Spring 2018)

Project 2: Social Network Mining

Group Members:

Yunhao Ba (705032832)

Shuangyu Li (805035359)

Jingchi Ma (705027270)

Chenguang Yuan (005030313)

Abstract

This projected was developed on R to study the various properties of social networks. We performed the first undirected social network with the data of Facebook from http://snap.stanford.edu/data/egonets-Facebook.html. The second part is a directed social network with the data of Google+ from http://snap.stanford.edu/data/egonets-Gplus.html

1. Facebook Networks

1.1 Structural properties of the facebook network

After building the undirected network from "facebook_combined.txt" file, we focused on the *connectivity* and *degree distribution* of the graph.

Question 1:
This network is **connected.**

Question 2:
The diameter of this graph is 8.

Question 3:
The degree distribution of the social network is plot below.

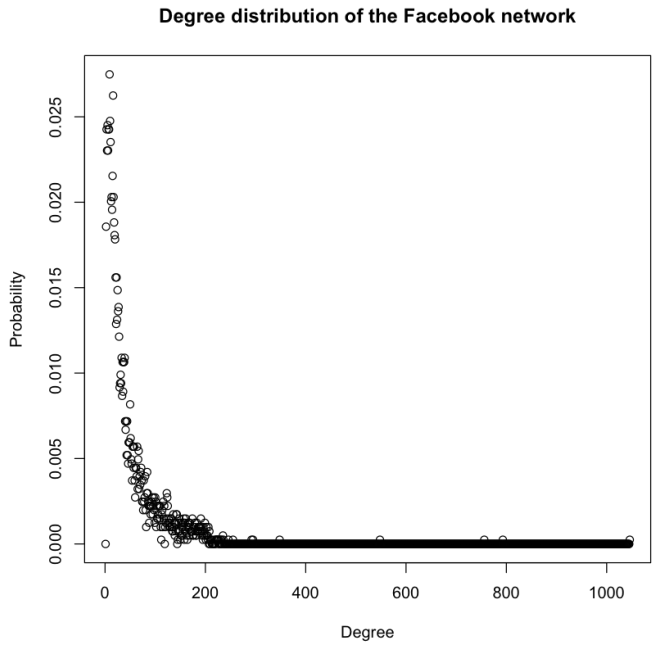

Fig : Degree distribution of the Facebook network

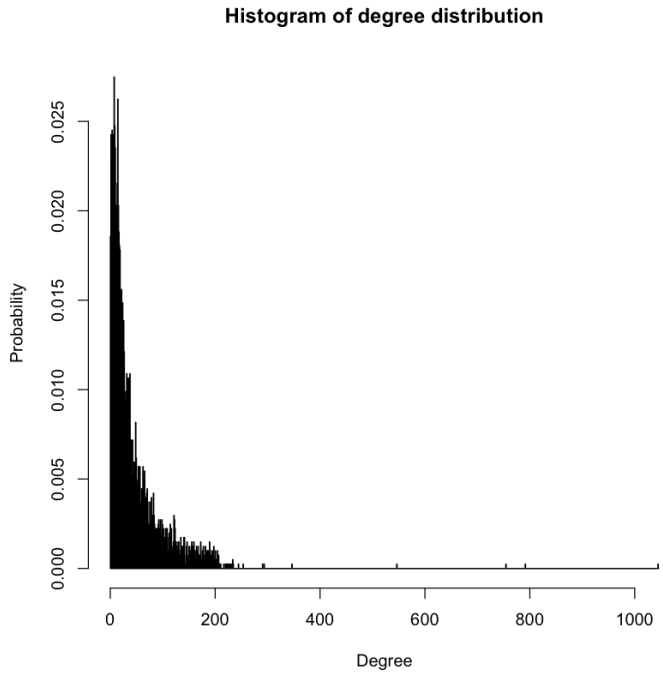

Fig: Histogram of the Facebook degree distribution

As observed, majority of the nodes are with low degrees, and the network follows the power law distribution.

Question 4:

The degree distribution is plotted in log-log scale and the slope of the line after log both axis is **-1.2648.** The plot is shown below. The left is the original fitting (slope: -1.2475) and the right one (slope: -1.2648) is the linear fitting after truncated points at beginning and end of the data, which gives us a better understanding of the power law for this network.

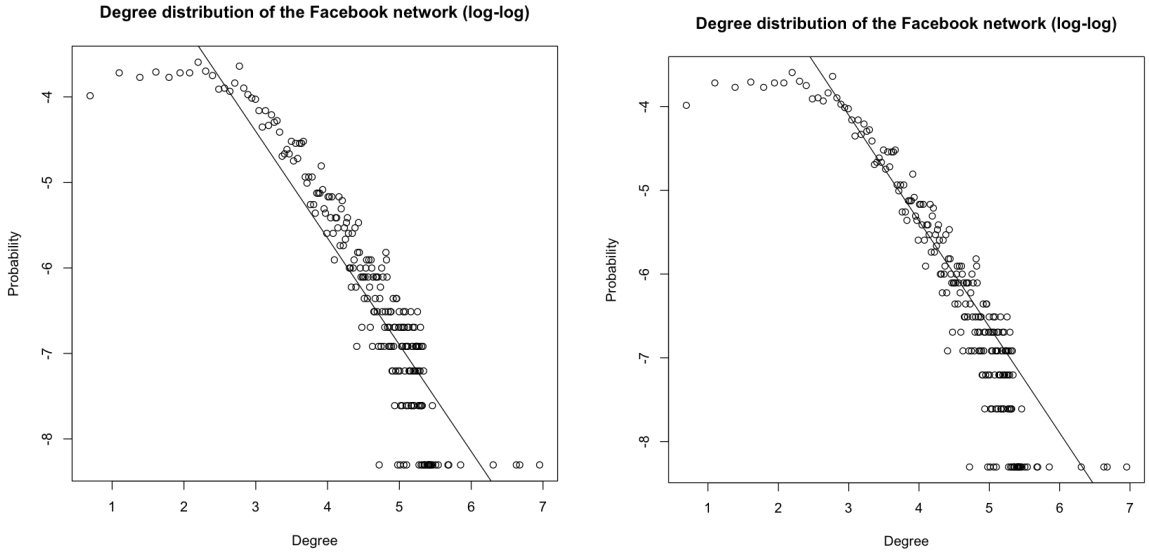

Fig: Degree distribution of the Facebook network(log-log)
Left: before truncation Right: after truncation

1.2 Personalized network

Question 5:

A personalized network is defined as a node with all 1 distance neighbors of it on the graph. Therefore for the user whose ID is 1, the number of nodes is 348, and the number of edges is 2866.

Question 6 & 7:

The diameter of the network is filled in the table below, and the upper bound of the diameter of the personalized network is 2, and the lower bound is 1. Here, we do not consider the case that the core node is just a single isolated node without any neighbor. We treat this situation as a single node, rather than a personalized network, i.e. the size of the personalized network is at least two. (The diameter of a single node is 0, if we consider the isolated core node cases.)

The lower bound for a personalized network is 1, which represents every node within this network connecting to another directly (fully connected network). This means that everyone within this network knows each other. The upper bound for a personalized network is 2, which represents a divergent network- some nodes connected to the core node but not all other nodes. In this case, the node can get to its foreigner nodes through the core node. The diameter is thus 2.

Table:personalized network of node #1

Number of nodes	348

Number of edges	2866
diameter	2

1.3 Core node's personalized network

A core node is defined as the nodes that have more than 200 neighbors. In this part, we will discuss various properties of the personalized net- work of the core nodes.

Question 8:
For this Facebook graph, there are **40** core nodes and their average degree is **279.375**

1.3.1 Community structure of core node's personalized network

Question 9:
In this question, we plot the community structure with two methods (mark the community regions or not). Besides, we also follows the fruchterman.reingold layout for consistency and better visualization.

Algorithm overview:
Fast greedy method assumes every vertex belongs to a separate community, and communities are merged iteratively such that each merge is locally optimal. The algorithm stops when it is impossible to increase the modularity any more. This is an unreliable algorithm comparatively, one shortest path among nodes will count them into same community and lead a lower modularity score.

Edge-Betweenness is defined as the number of shortest paths between pairs of vertices that run along them. The edge with high edge-betweenness has high potential to become a bridge between communities. After finding the edge-betweenness of every edge, the algorithm will delete the highest one persistently until it does not exist. Therefore, more mutual edges leads lower score here.

Infomap calculates modularity using flows. Higher probability of a node could go back itself under random walk results in a higher modularity score based on the equation. Therefore, more circles in a graph may result in a higher modularity score.

Modularity explanation:

For node#1, the modularity by fast greedy is slightly higher than the other 2 methods. The potential reason for this could be the sparse edges in the graphs. Except one obvious community, all other node are distribute around and loosely connected which bring difference into results.

For node #108, the fast greedy method cares more about the absolute connection between nodes instead of the comparative connection other algorithms care more about. Therefore, from the graph, we could observe that the fast greedy separate graph into two large communities due to their dense edges and therefore has a lower score here.

For node#349, infomap and edge-Betweenness get a very low modularity score because of the even distributed of the edges. In this scenario, both algorithms will end very early and consider the general group as a community.

For node#484, three algorithms achieve similar score here due to the simple structure of the graph. From the plots below, we could clearly observe that the ego network is separated into 3 large communities, dense connections within communities and sparse connections among communities in the other word. Algorithms' decision on the margin node result in the slight difference in the score.

For node #1087, the score is very low for each node no matter the algorithms because of its "double-core" structure. From the degree distribution plots, we could observe there are two nodes in the range of the highest degrees. With two famous nodes, algorithms may tend to consider them as a whole communities instead of a bridge node which happened in one core node senario. Therefore, the colored communities is more like a sub-communities in this situation.

Detailed features of the personalized networks and plots are shown below.

Table:personalized network of node #1

Number of nodes	348
Number of edges	2866
diameter	2
Modularity by fast greedy	0.4131014
Modularity by Edge-Betweenness	0.3533022
Modularity by Infomap	0.3891185

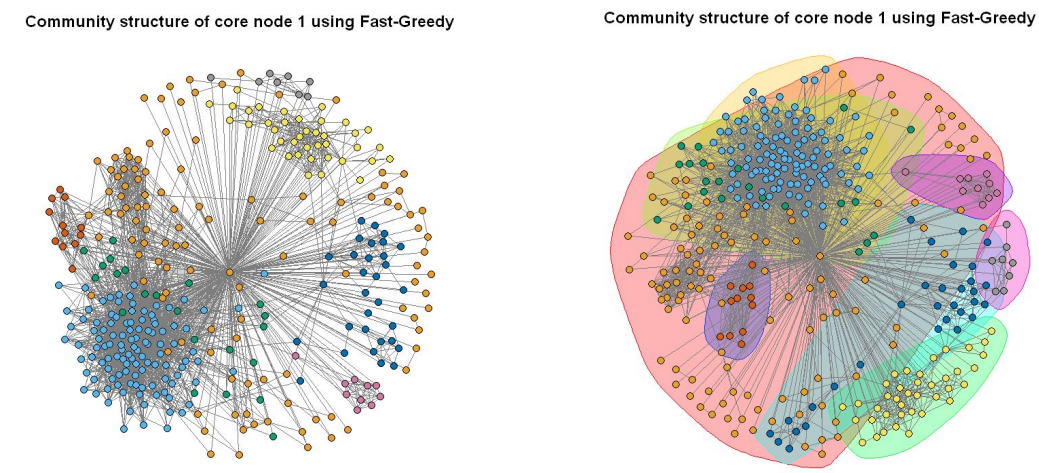

Fig : Community structure of Node 1 by fast greedy

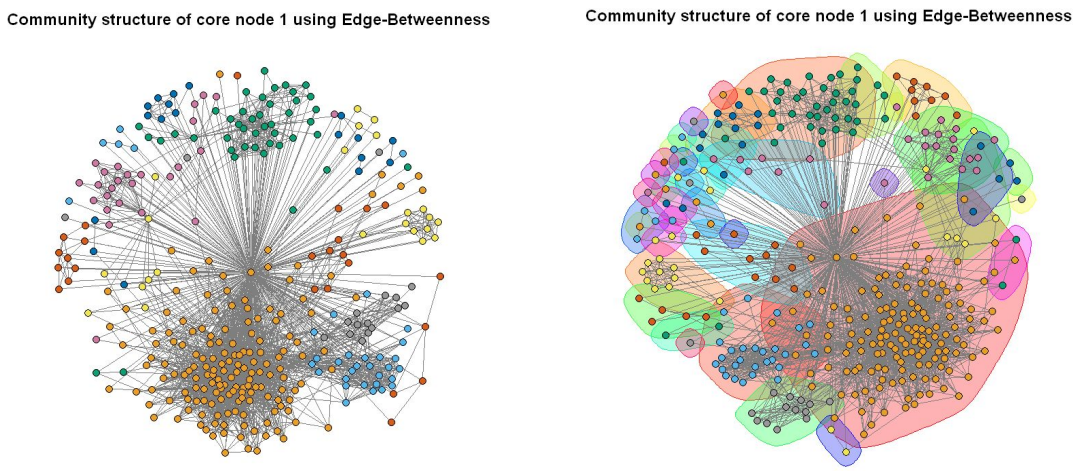

Fig : Community structure of Node 1 by Edge-Betweenness

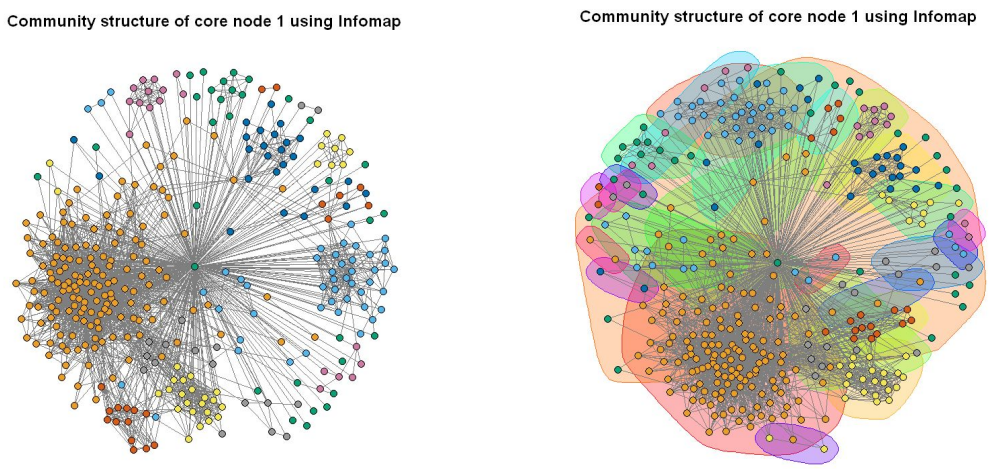

Fig : Community structure of Node 1 by Infomap

Table:personalized network of node #108

Number of nodes	1046
Number of edges	27795
diameter	2
Modularity by fast greedy	0.4359294
Modularity by Edge-Betweenness	0.5067549
Modularity by Infomap	0.5082492

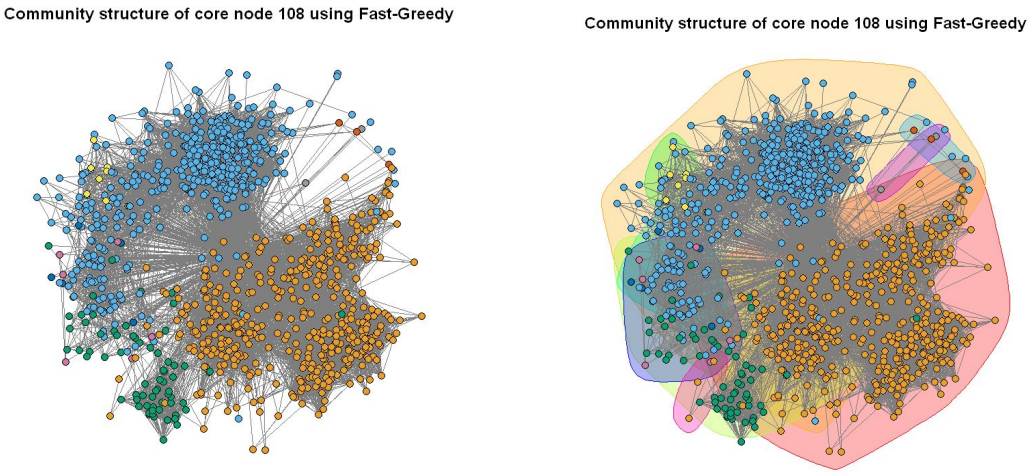

Fig : Community structure of Node 108 by fast greedy

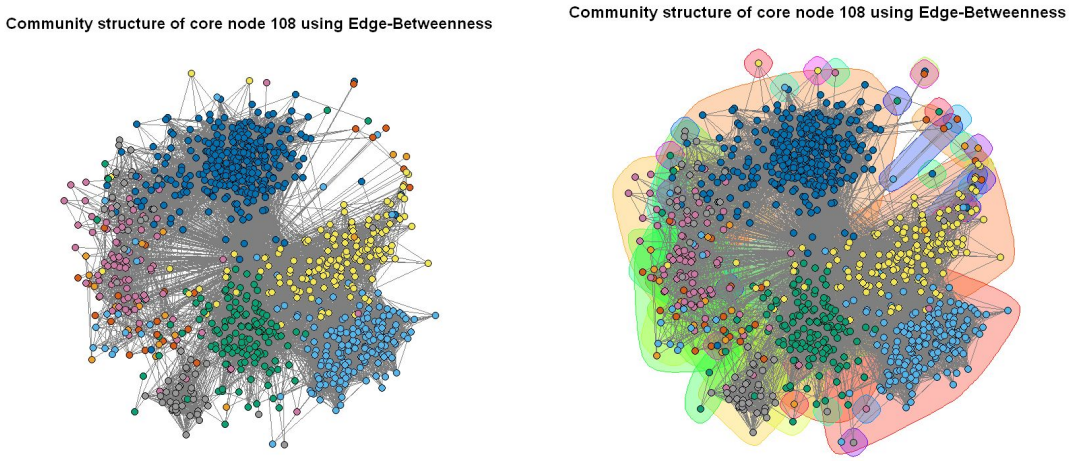

Fig : Community structure of Node 108 by Edge-Betweenness

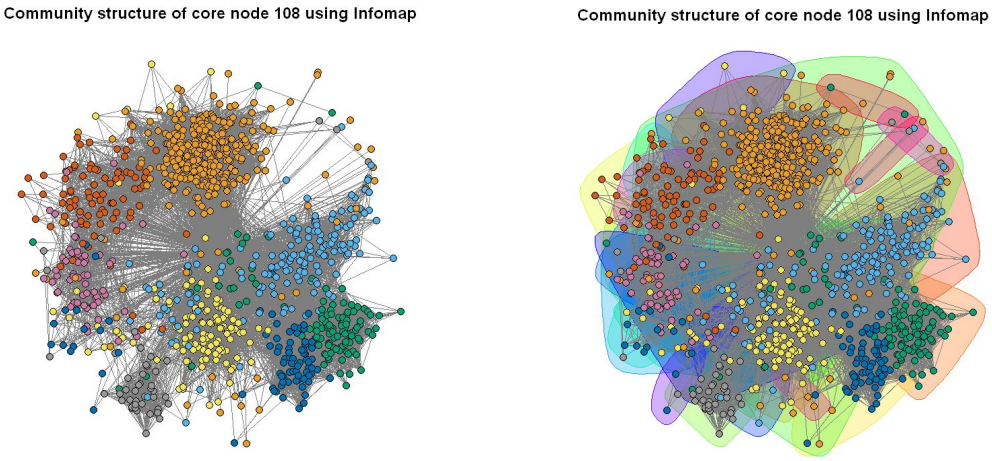

Fig : Community structure of Node 108 by Infomap

Table:personalized network of node #349

Number of nodes	230
Number of edges	3441
diameter	2

Modularity by fast greedy	0.2517149
Modularity by Edge-Betweenness	0.133528
Modularity by Infomap	0.0954642

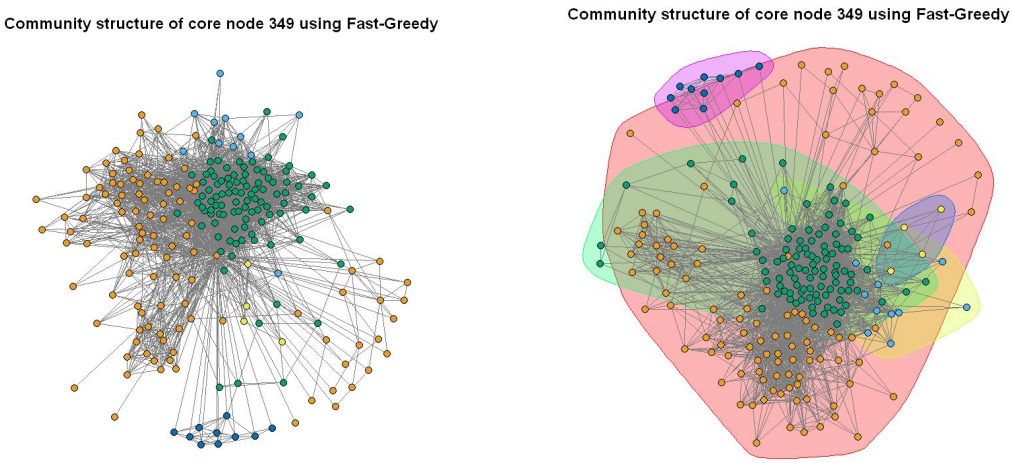

Fig : Community structure of Node 349 by fast greedy

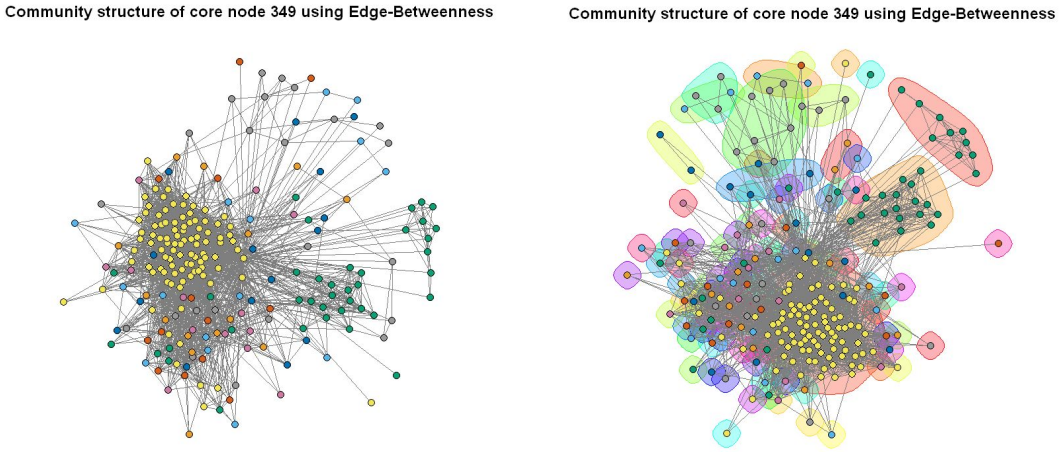

Fig : Community structure of Node 349 by Edge-Betweenness

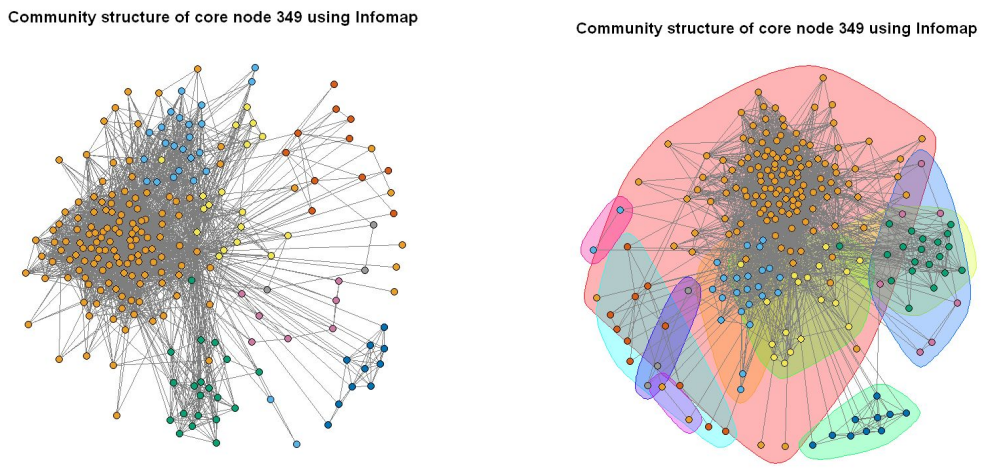

Fig : Community structure of Node 349 by Infomap

Table:personalized network of node #484

Number of nodes	232
Number of edges	4525
diameter	2
Modularity by fast greedy	0.5070016
Modularity by Edge-Betweenness	0.4890952
Modularity by Infomap	0.5152788

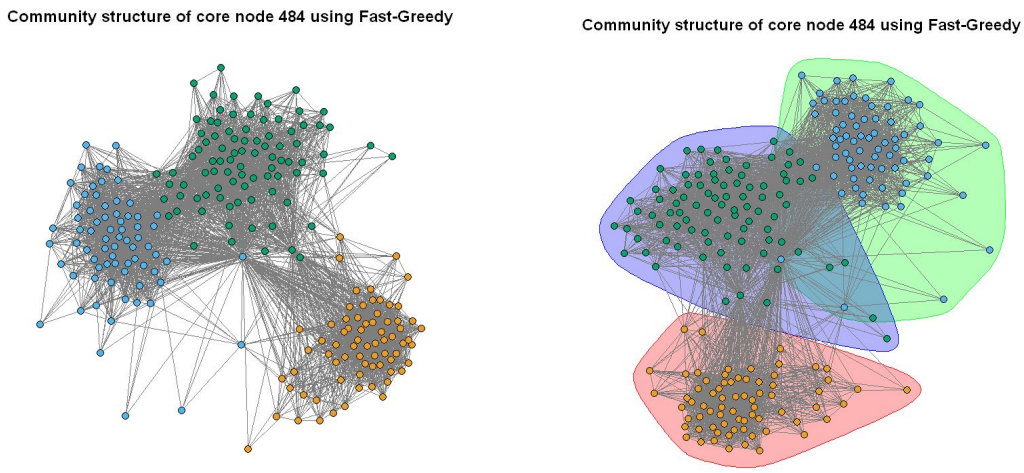

Fig : Community structure of Node 484 by fast greedy

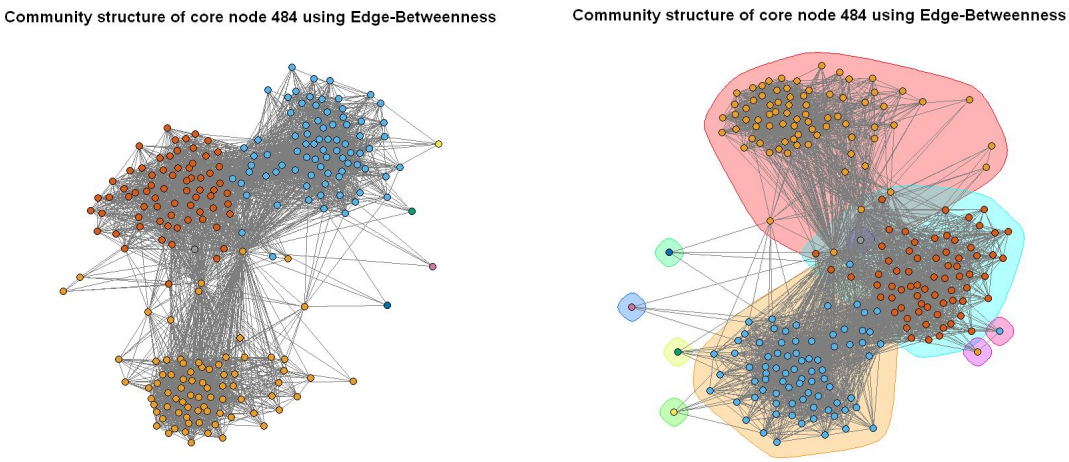

Fig : Community structure of Node 484 by Edge-Betweenness

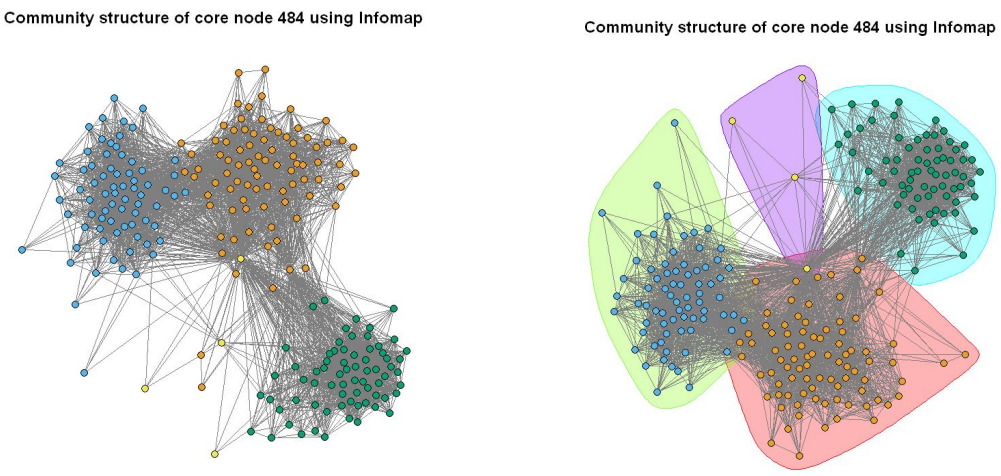

Fig : Community structure of Node 484 by Infomap

Table:personalized network of node #1087

Number of nodes	206
Number of edges	7409
diameter	2
Modularity by fast greedy	0.1455315
Modularity by Edge-Betweenness	0.02762377
Modularity by Infomap	0.02690662

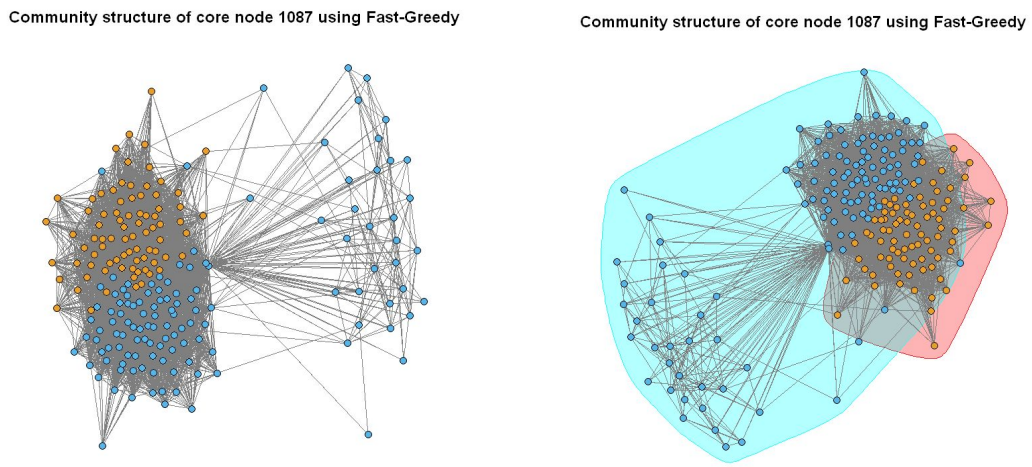

Fig : Community structure of Node 1087 by fast greedy

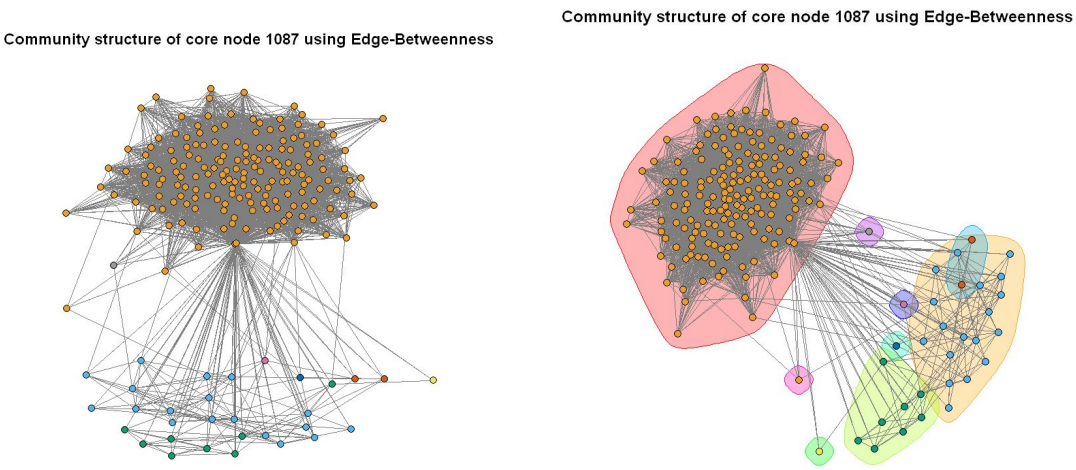

Fig : Community structure of Node 1087 by Edge-Betweenness

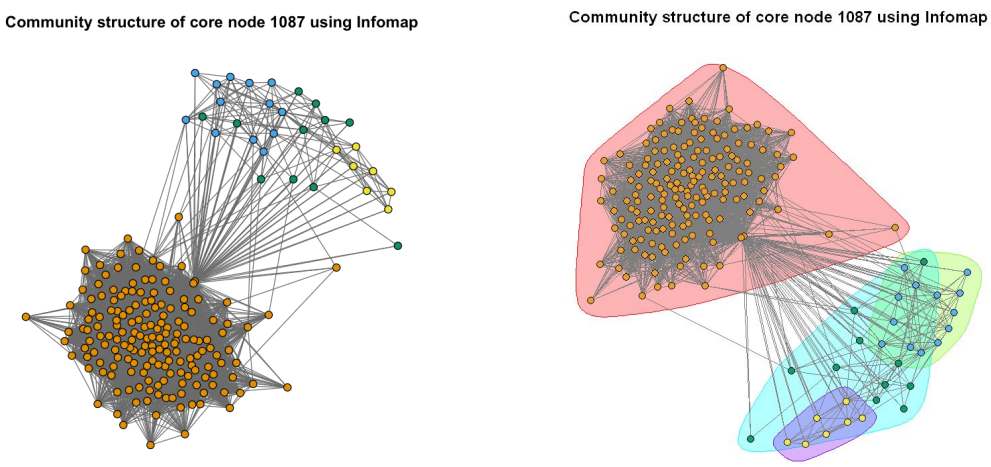

Fig : Community structure of Node 1087 by Infomap

1.3.2 Community structure with the core node removed

In this part, we will explore the effect on the community structure of a core node's personalized network when the core node itself is removed from the personalized network. We plot the community structure with two methods (mark the community regions or not). Besides, we also follows the fruchterman.reingold layout for consistency and better visualization.

Question 10:

The shaded section is the modularity from the question 9 where the center node is not removed. It is obvious that the modularity score increases for all algorithms. This result follows intuition. Because the "centered" node is removed, the connections between communities also decrease, and as the most important node, core nodes has the most edges that centralize the network. Modularity can be interrupted as the strength of dividing a network into communities. With the loss of these interconnections among communities, the network can be divided into separate communities more easily, which will lead to a higher modularity score.

Table:personalized network of node #1

Modularity by fast greedy	0.4131014
Modularity by Edge-Betweenness	0.3533022
Modularity by Infomap	0.3891185
Modularity by fast greedy	0.4418533
Modularity by Edge-Betweenness	0.4161416

Modularity by Infomap	0.4180077

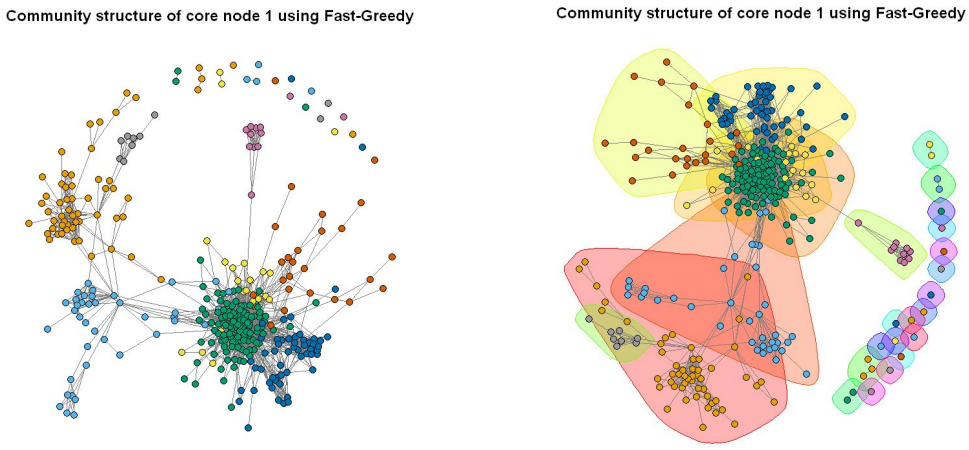

Fig : Community structure of Node 1 by fast greedy

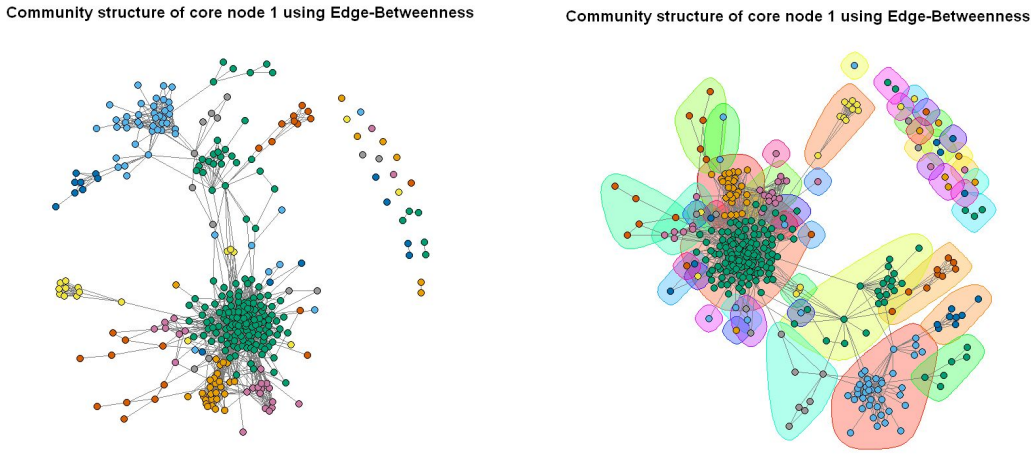

Fig : Community structure of Node 1 by Edge-Betweenness

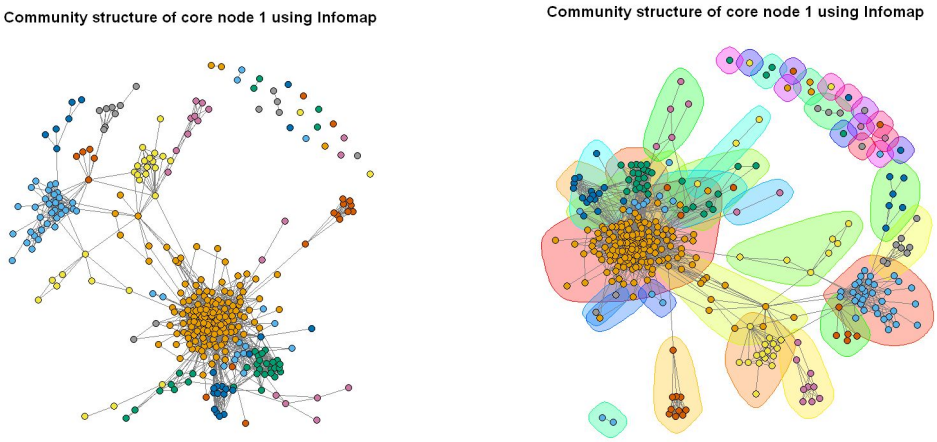

Fig : Community structure of Node 1 by Infomap

Table:personalized network of node #108

Modularity by fast greedy	0.4359294
Modularity by Edge-Betweenness	0.5067549
Modularity by Infomap	0.5082492
Modularity by fast greedy	0.4581271
Modularity by Edge-Betweenness	0.5213216
Modularity by Infomap	0.5205171

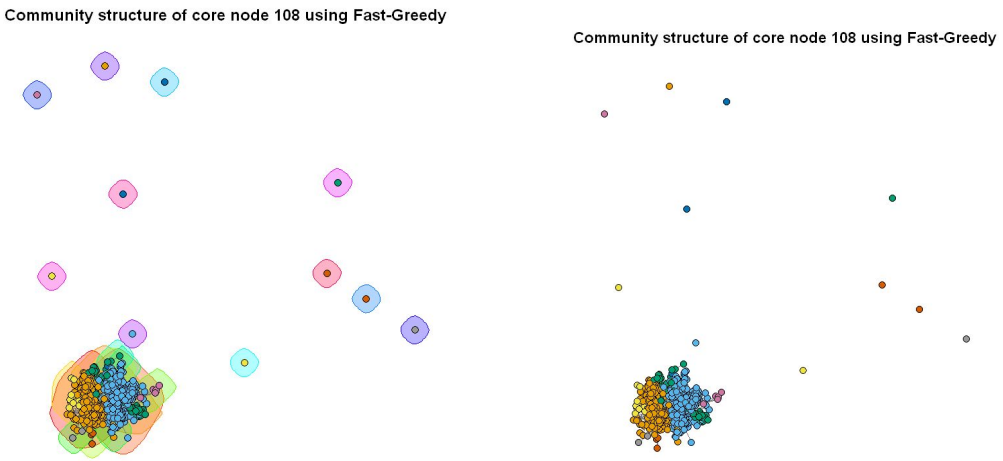

Fig : Community structure of Node 108 by fast greedy

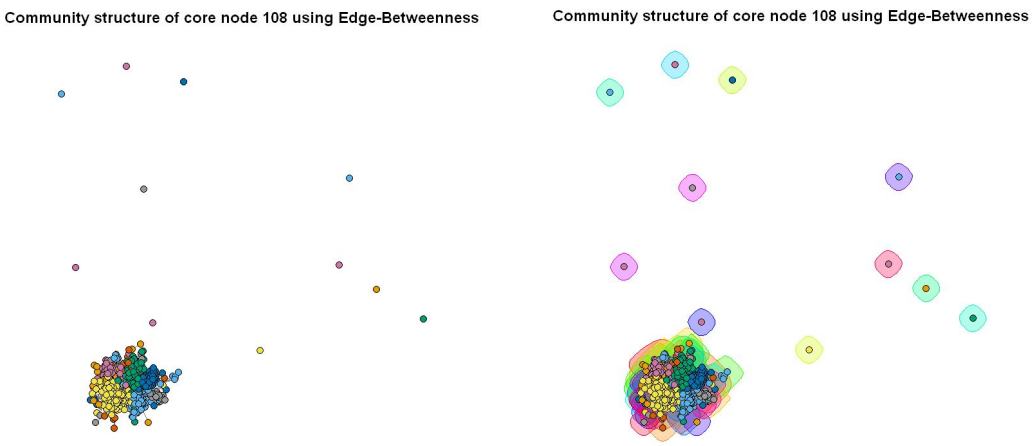

Fig : Community structure of Node 108 by Edge-Betweenness

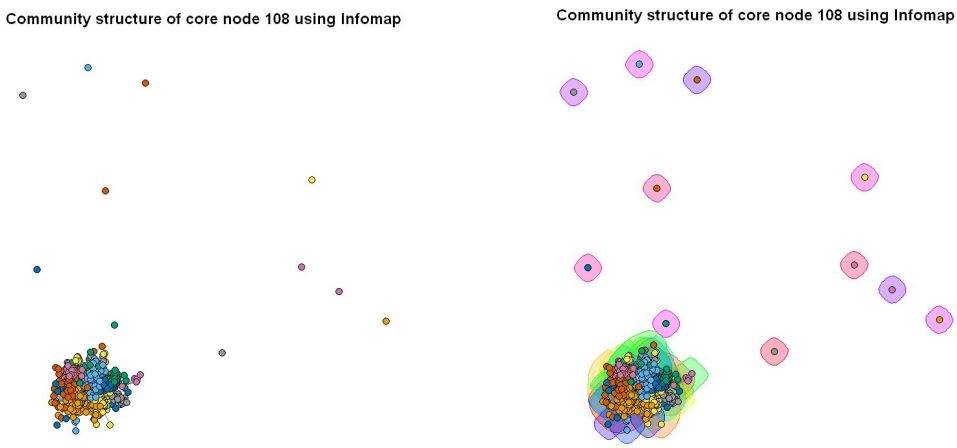

Fig : Community structure of Node 108 by Infomap

Table:personalized network of node #349

Modularity by fast greedy	0.2517149
Modularity by Edge-Betweenness	0.133528
Modularity by Infomap	0.0954642
Modularity by fast greedy	0.2456918

Modularity by Edge-Betweenness	0.1505663
Modularity by Infomap	0.2337732

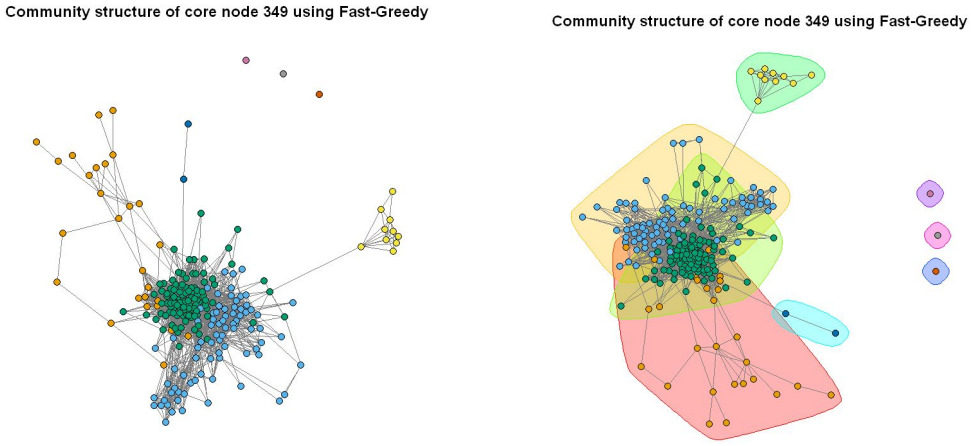

Fig : Community structure of Node 349 by fast greedy

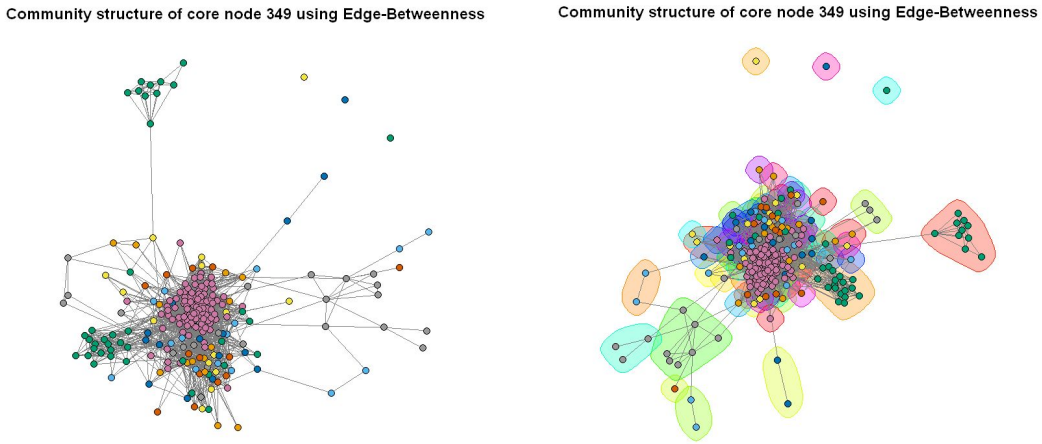

Fig : Community structure of Node 349 by Edge-Betweenness

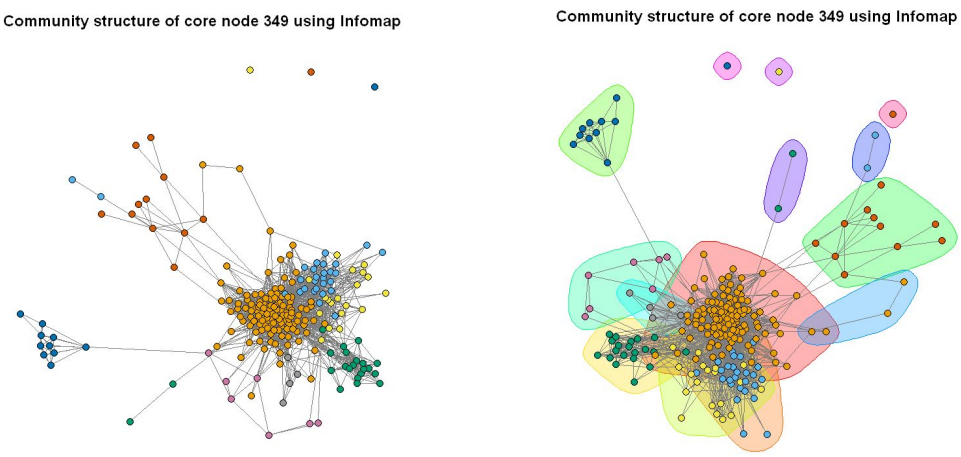

Fig : Community structure of Node 349 by Infomap

Table:personalized network of node #484

Modularity by fast greedy	0.5070016
Modularity by Edge-Betweenness	0.4890952
Modularity by Infomap	0.5152788
Modularity by fast greedy	0.5342142
Modularity by Edge-Betweenness	0.5154413
Modularity by Infomap	0.5434437

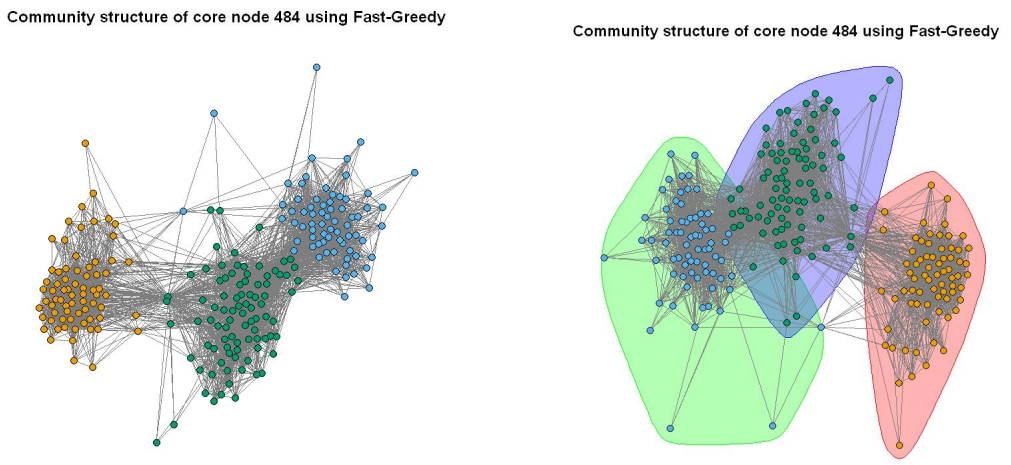

Fig : Community structure of Node 484 by fast greedy

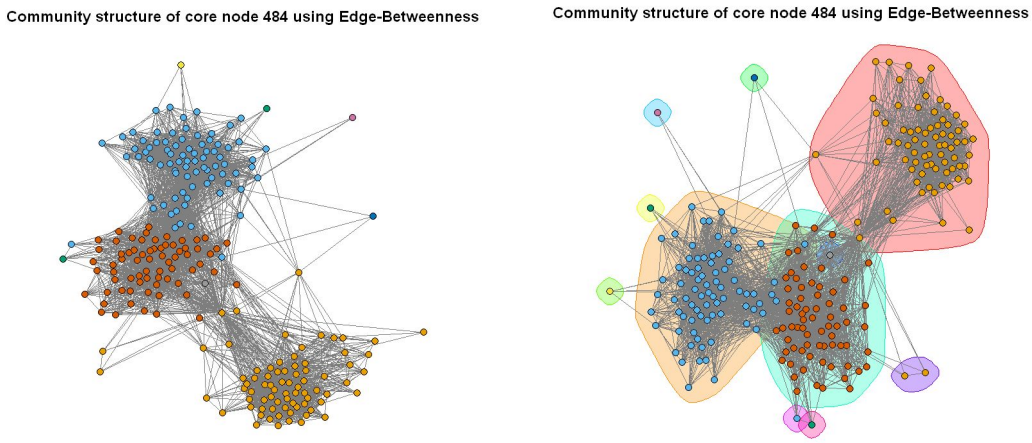

Fig : Community structure of Node 484 by Edge-Betweenness

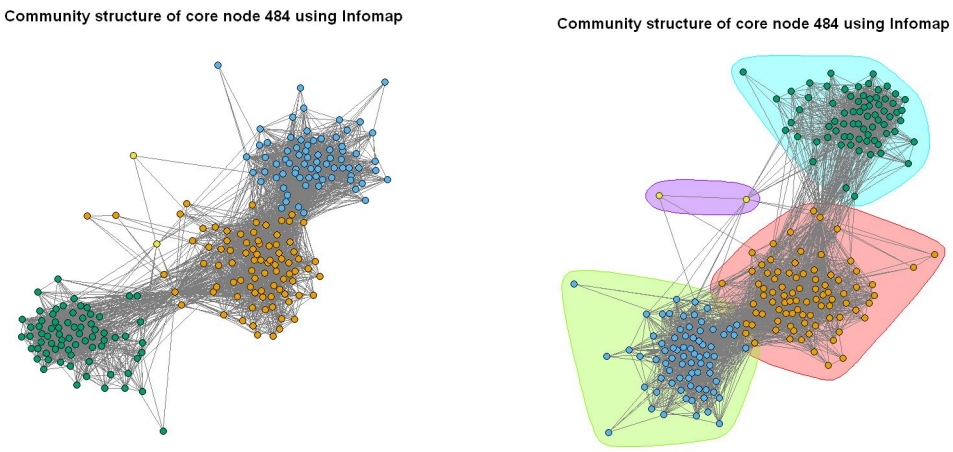

Fig : Community structure of Node 484 by Infomap

Table:personalized network of node #1087

Modularity by fast greedy	0.1455315
Modularity by Edge-Betweenness	0.02762377
Modularity by Infomap	0.02690662
Modularity by fast greedy	0.1481956
Modularity by Edge-Betweenness	0.0324953
Modularity by Infomap	0.02737159

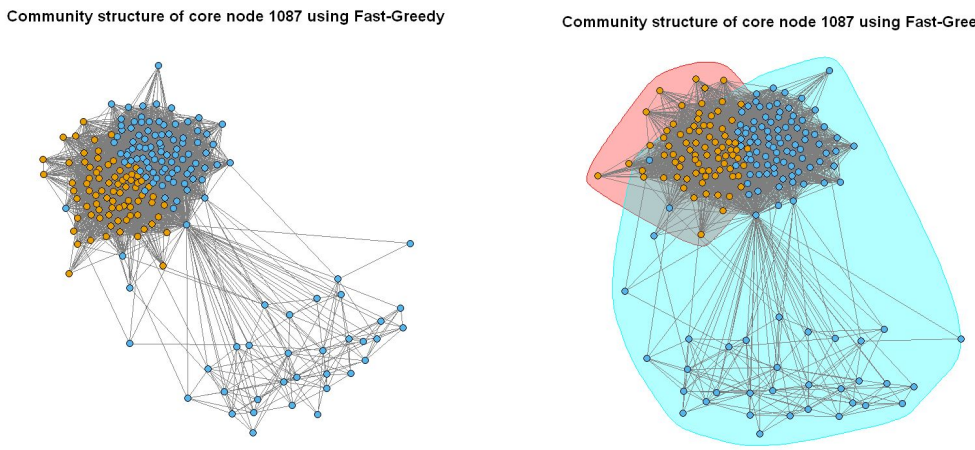

Fig : Community structure of Node 1087 by fast greedy

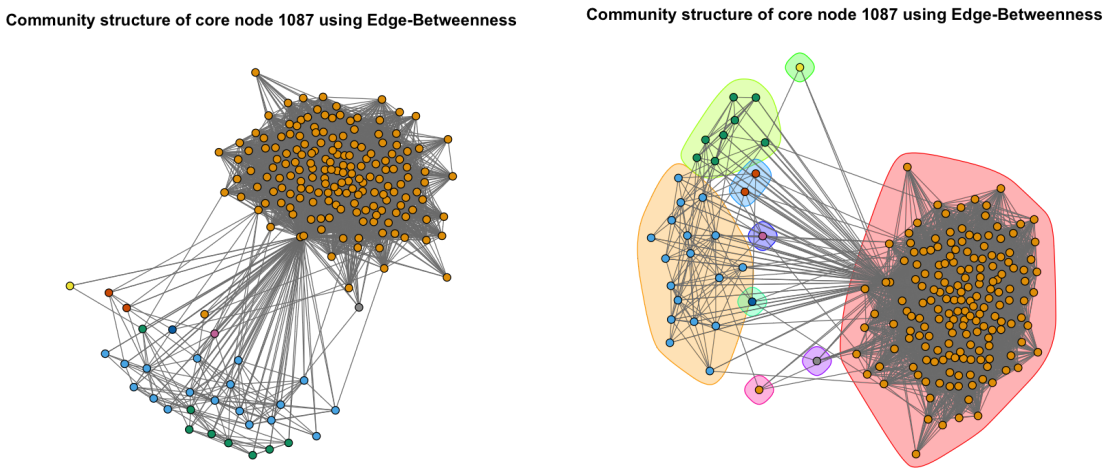

Fig : Community structure of Node 1087 by Edge-Betweenness

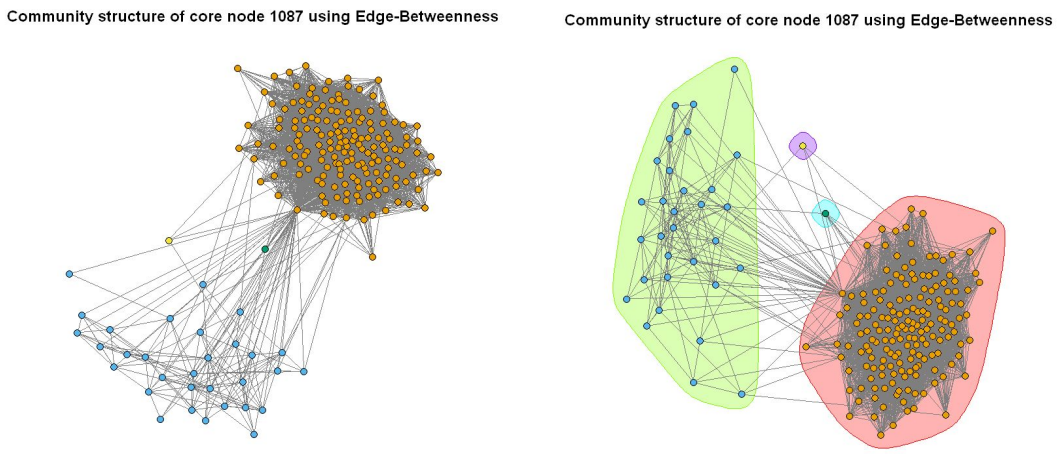

Fig : Community structure of Node 1087 by Infomap

1.3.3 Characteristic of nodes in the personalized network

In this part, we will explore characteristics of nodes in the personalized network using two measures. These two measures are stated and defined below:

Measure 1: **Embeddedness** of a node is defined as the number of mutual friends a node shares with the core node.

Measure 2: **Dispersion** of a node is defined as the sum of distances between every pair of the mutual friends the node shares with the core node. The distances should be calculated in a modified graph where the node (whose dispersion is being computed) and the core node are removed.

Question 11:

Because the core node has friendship with everyone within this personalized network, then the mutual friends of the a node with the core node is just the non-core neighbors of that node in the personalized network. Therefore, embeddedness could be expressed as below based on this definition of "the number of the mutual friends". "-1" means that we need to exclude the core node in the neighbors.

$$\text{Embeddedness}(\text{node}) = \text{degree}(\text{node}) - 1$$

Question 12:

For each of the core node's personalized network (use the same core nodes as question 9), we plot the distribution of embeddedness and dispersion. When calculating the dispersion, we might encounter the case where the number of mutual friends of a node is smaller than or equal to one. We define the dispersion to be 0 in this case. Besides, for some pairs in the mutual friend list, the

two nodes may not be connected in the modified network (deleting the core node and the target node). Their distance should be infinite in this case. To avoid this infinity dominating the dispersion, we set the upper bound of the distance to be "diameter + constant". The constant is added to distinguish the cases in which the actual distances are exactly the diameter. All the personalized networks used in this question are with a diameter of 2, and we set the constant to be 8, i.e. the distance when the two nodes are not connected is 10. This constant should not be too large, since too large constant will also dominate the dispersion. After testing, we found 8 is a reasonable number.

As expected, the embeddedness distribution is similar to the degree distribution, and the dispersion distributions are generally in a decreasing shape. Dispersion for the majority nodes is relatively small, and for some special nodes, their dispersion are extremely high. The dispersion distribution is affected by the degree distribution and the connectivity of a node's friends.

Embeddedness Distribution (with bin size = 1)

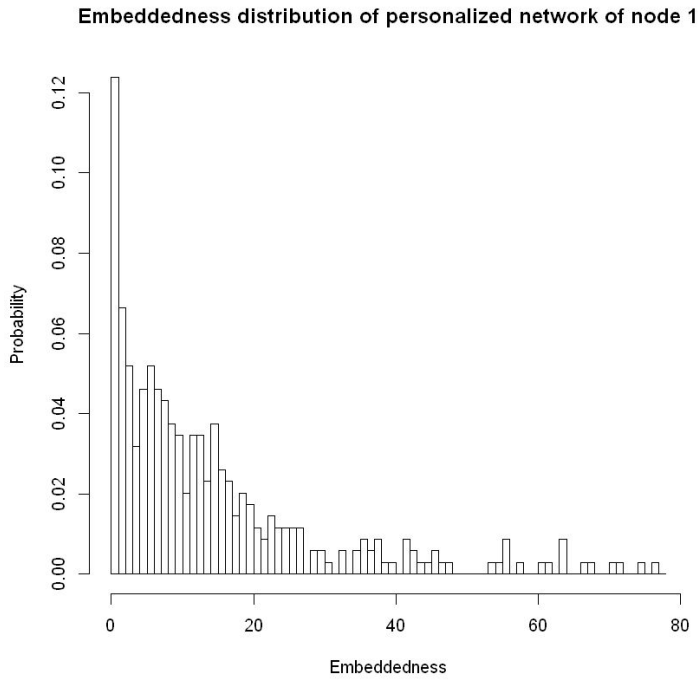

Fig: Embeddedness distribution for personalized network of node 1

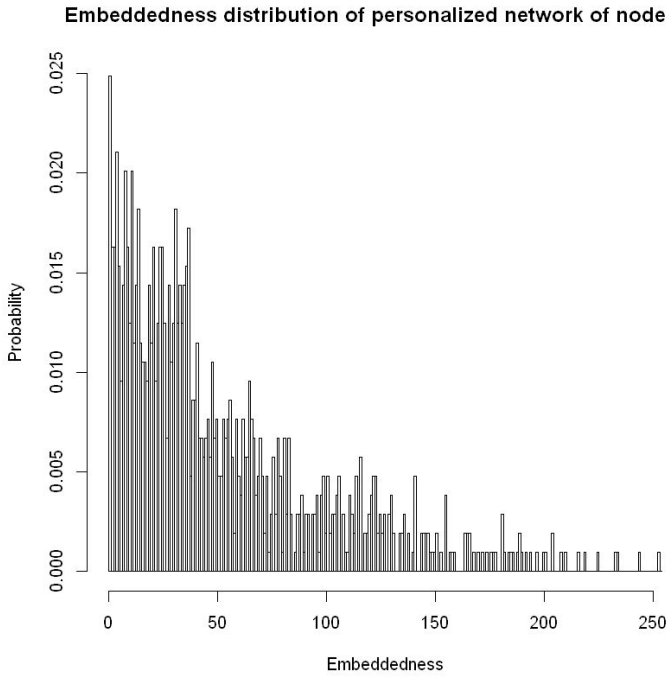

Fig: Embeddedness distribution for personalized network of node 108

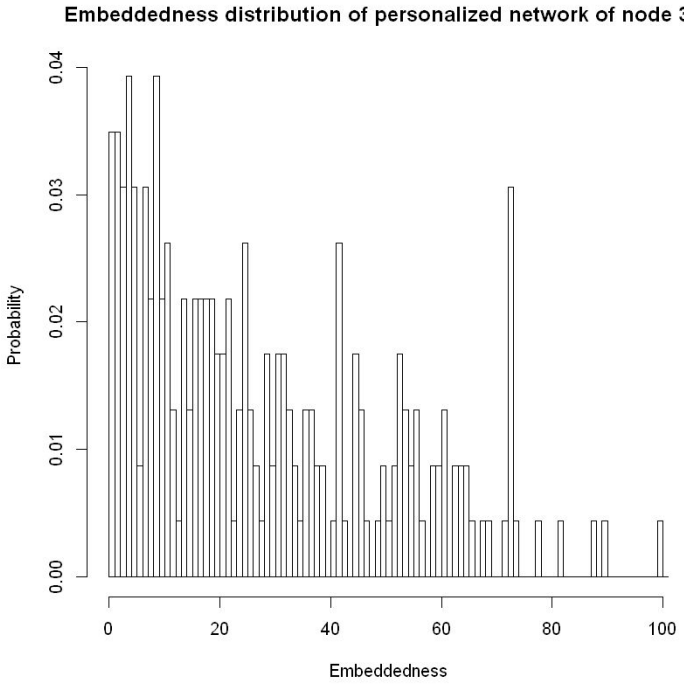

Fig: Embeddedness distribution for personalized network of node 349

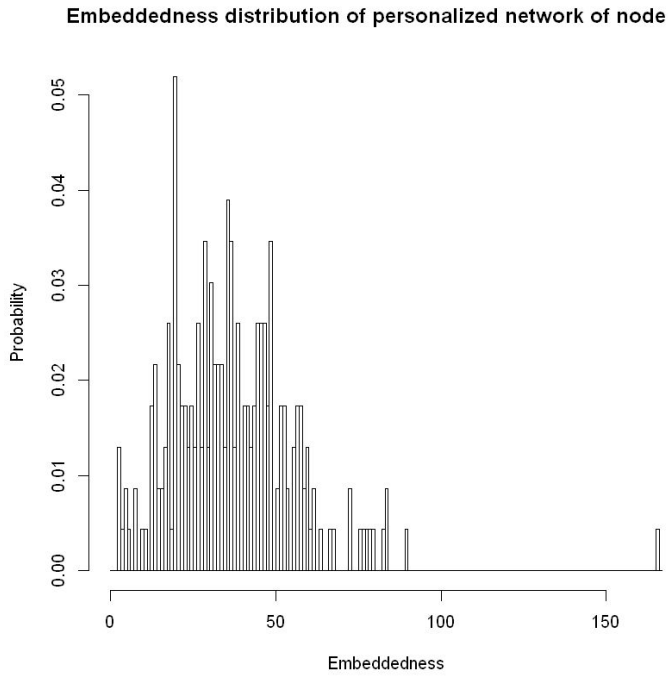

Fig: Embeddedness distribution for personalized network of node 484

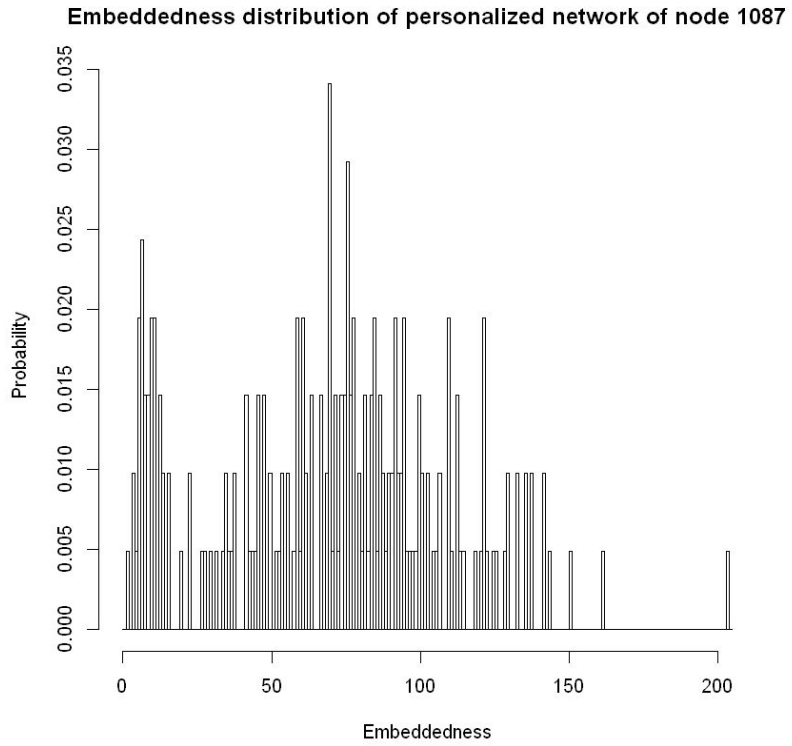

Fig: Embeddedness distribution for personalized network of node 1087

Dispersion Distribution (with bin size = 1 (left) and number of bins = 100 (right) respectively) :

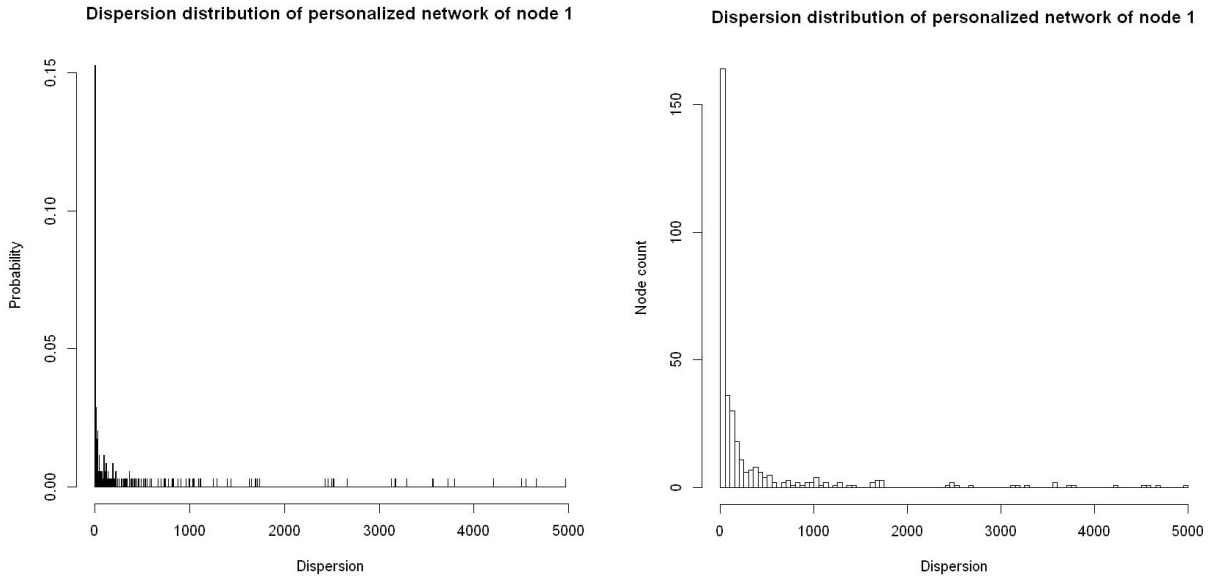

Fig: Dispersion distribution for personalized network of node 1
(with bin size = 1 and number of bins = 100 respectively)

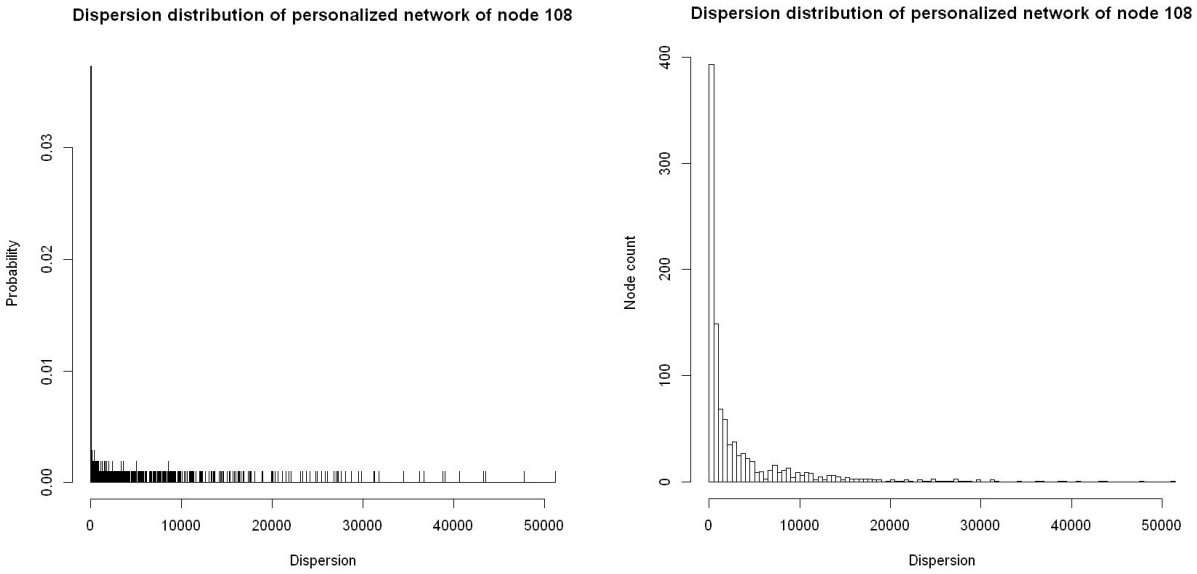

Fig: Dispersion distribution for personalized network of node 108
(with bin size = 1 and number of bins = 100 respectively)

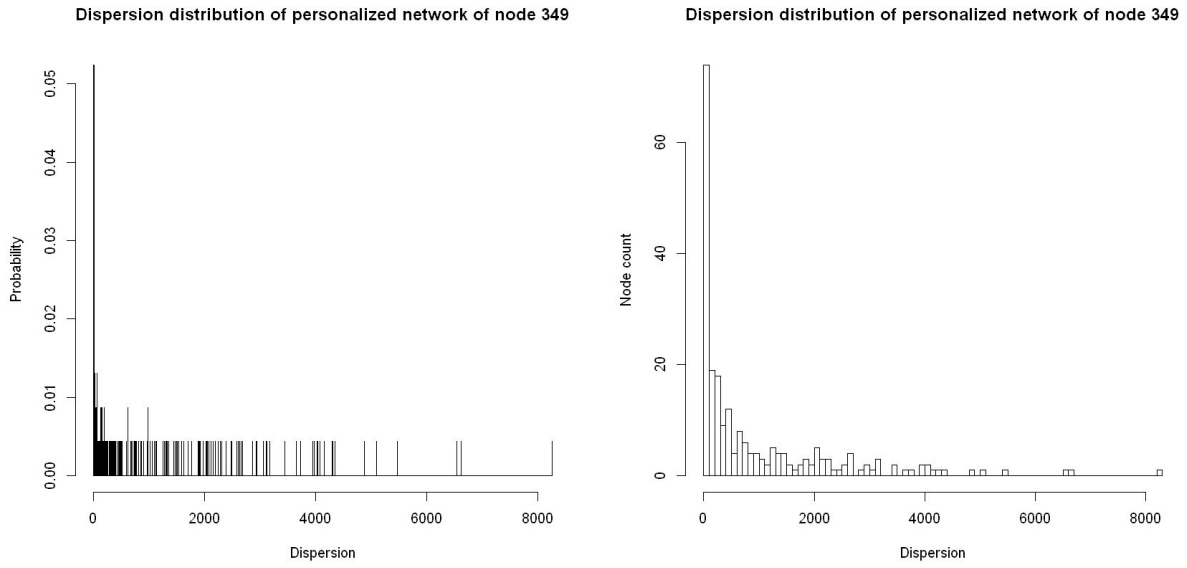

Fig: Dispersion distribution for personalized network of node 349
(with bin size = 1 and number of bins = 100 respectively)

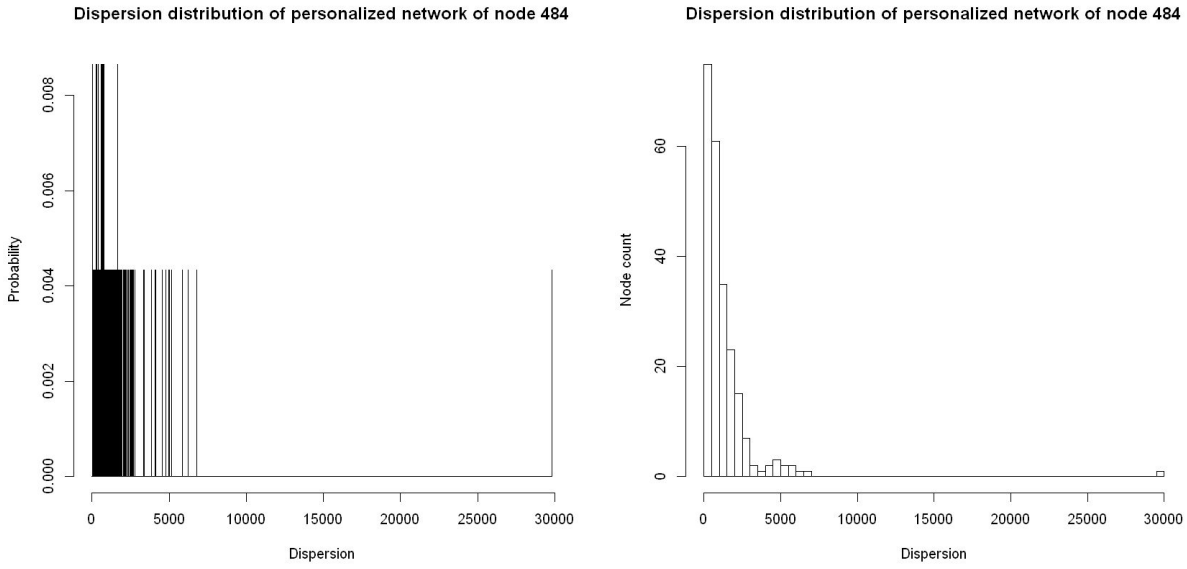

Fig: Dispersion distribution for personalized network of node 484
(with bin size = 1 and number of bins = 100 respectively)

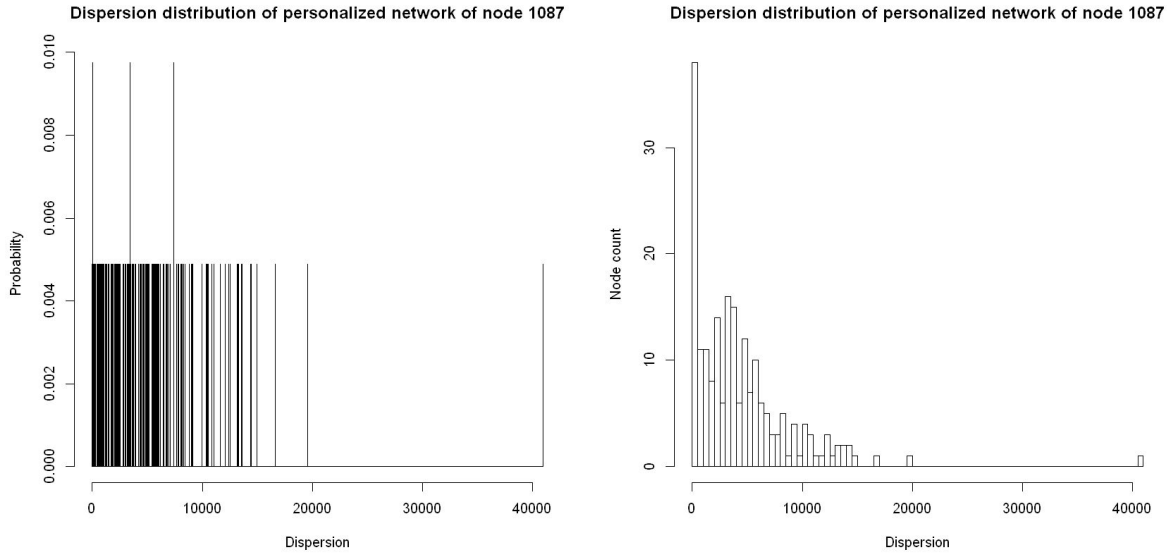

Fig: Dispersion distribution for personalized network of node 1087
(with bin size = 1 and number of bins = 100 respectively)

Question 13 & Question 14

For each of the core node's personalized network, we used fast greedy algorithm to plot the community structure of the personalized network using colors and highlight the node with maximum dispersion. We also highlight the edges incident to this node.

Then, we repeat question 13, but now highlight the node with maximum embeddedness and the node with maximum quotient (=dispersion/embeddedness). Also, we highlight the edges incident to these nodes. When calculating the quotient, it is possible that both the dispersion and the embeddedness are 0. We define this 0/0 as 1 in our implementation, since the dispersion and the embeddedness are the same in this case. Majority (except personalized network of node 1) of the max dispersion node, max embeddedness node, and max dispersion/embeddedness node are the same node in these personalized networks.

Note: we follows the fruchterman.reingold layout for consistency and better visualization.

Personalized Network of Node 1:
Node ID with max dispersion: 57
Node ID with max embeddedness: 57
Node ID with max quotient: 26

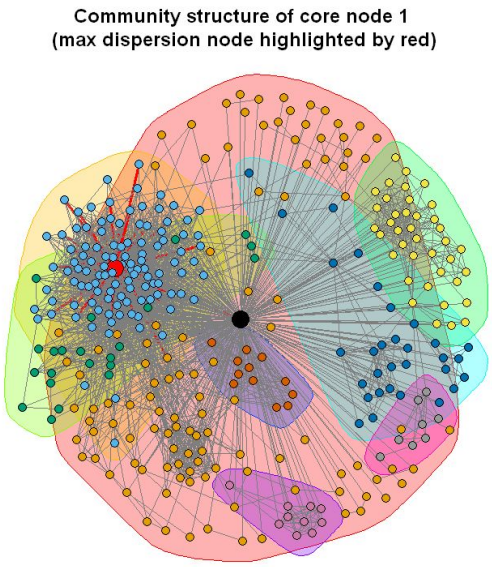

Fig: Community structure of core node 1
(max dispersion node and incident edges highlighted by red)

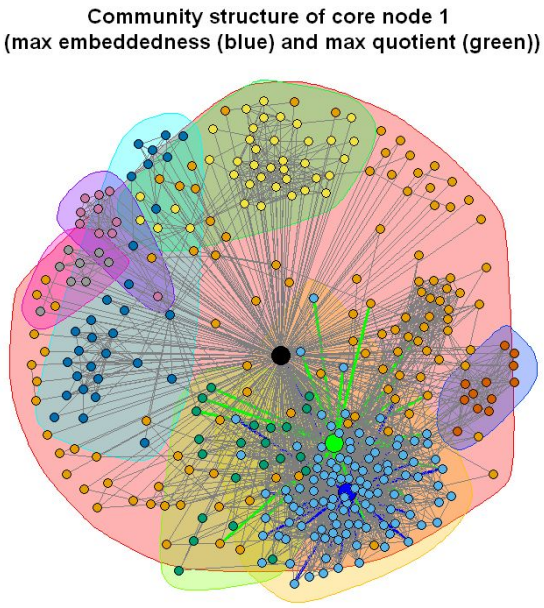

Fig: Community structure of core node 1
(max embeddedness node and incident edges highlighted by blue
max quotient node and incident edges highlighted by green)

Personalized Network of Node 108:

Node ID with max dispersion: 1889

Node ID with max embeddedness: 1889

Node ID with max quotient: 1889

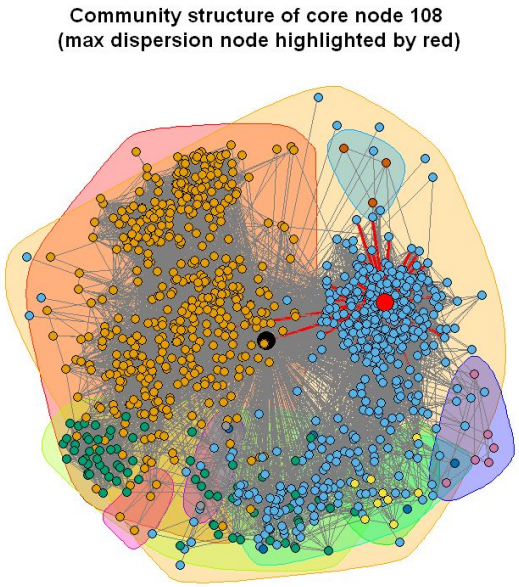

Fig: Community structure of core node 108
(max dispersion node and incident edges highlighted by red)

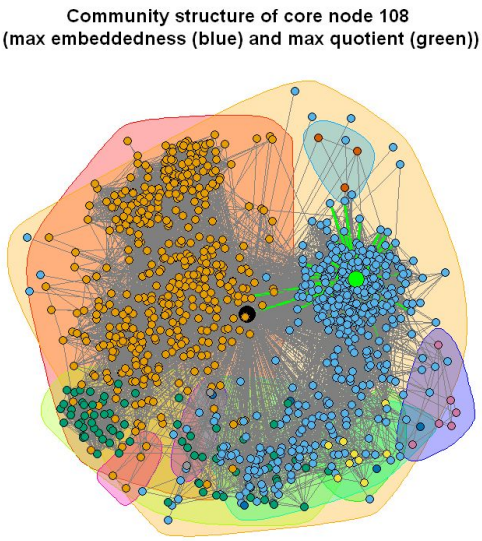

Fig: Community structure of core node 108
(max embeddedness node and incident edges highlighted by blue
max quotient node and incident edges highlighted by green)

Personalized Network of Node 349:
Node ID with max dispersion: 377
Node ID with max embeddedness: 377
Node ID with max quotient: 377

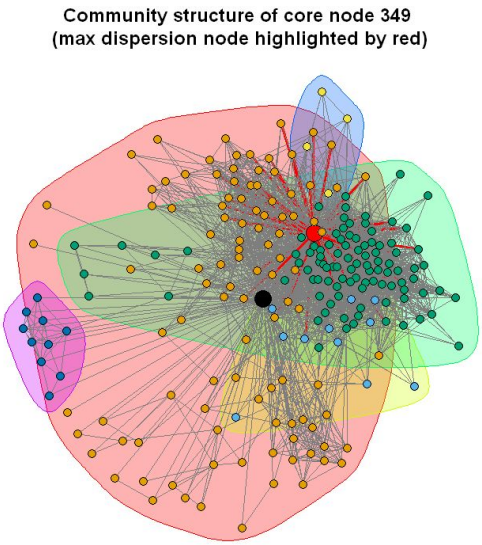

Fig: Community structure of core node 349
(max dispersion node and incident edges highlighted by red)

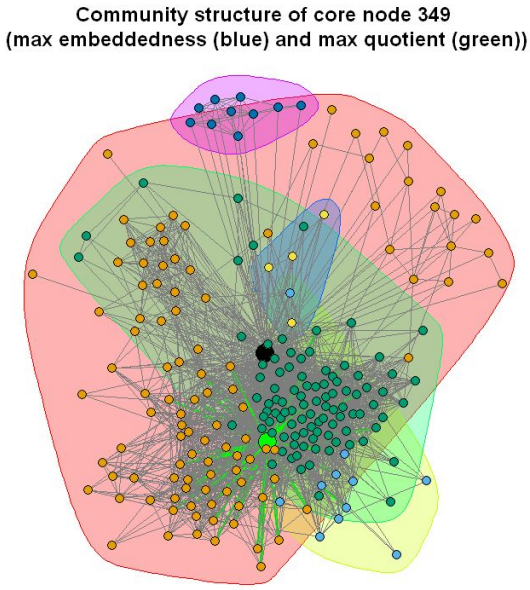

Fig: Community structure of core node 349
(max embeddedness node and incident edges highlighted by blue
max quotient node and incident edges highlighted by green)

Personalized Network of Node 484:

Node ID with max dispersion: 108
Node ID with max embeddedness: 108
Node ID with max quotient: 108

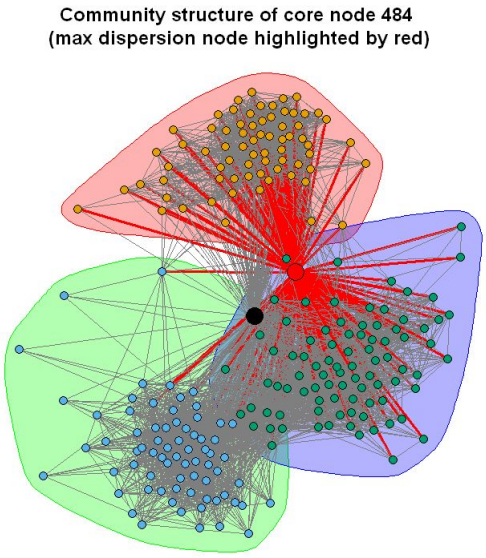

Fig: Community structure of core node 484
(max dispersion node and incident edges highlighted by red)

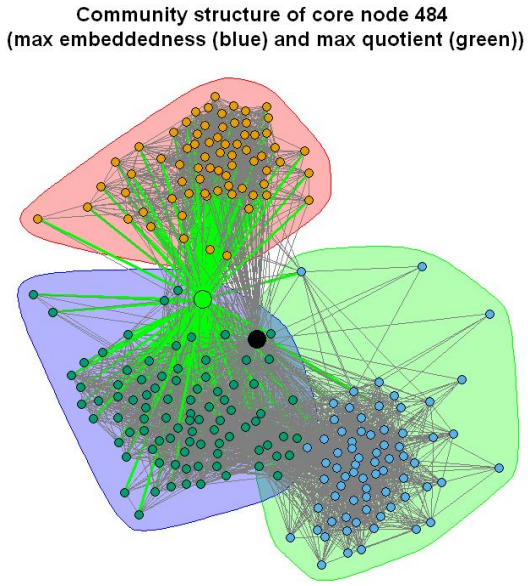

Fig: Community structure of core node 484
(max embeddedness node and incident edges highlighted by blue
max quotient node and incident edges highlighted by green)

Personalized network of node 1087 :

Node ID with max dispersion: 108

Node ID with max embeddedness: 108

Node ID with max quotient: 108

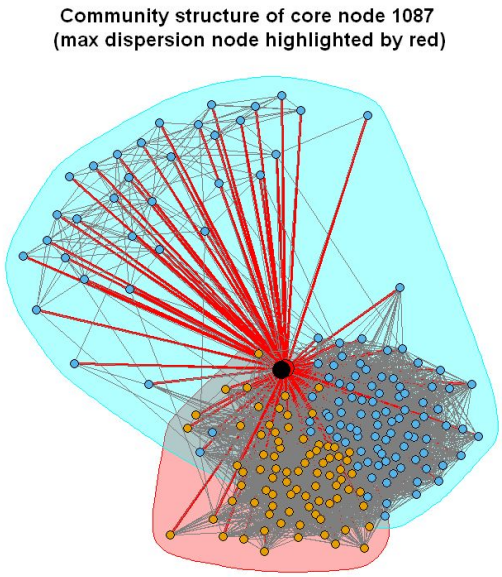

Fig: Community structure of core node 1087
(max dispersion node and incident edges highlighted by red)

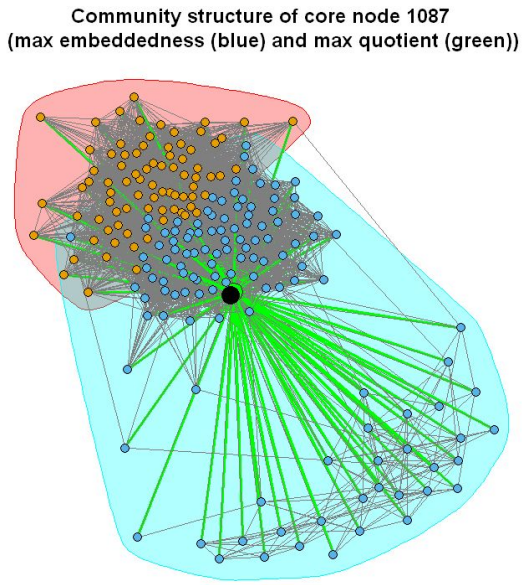

Fig: Community structure of core node 1087
(max embeddedness node and incident edges highlighted by blue

max quotient node and incident edges highlighted by green)

Question 15: Analysis of Results in Question 13&14

Embeddedness:
In the general cases, embeddedness has been so tightly associated with tie strength. However, more common friends reveal a better and stronger "friendship" or co-work relationship which usually has the highest embeddedness. In the personalized networks, the node with max embeddedness is the non-core node with max degree.

Dispersion：
Dispersion is a better indicator for "strong ties". It is a structural means of capturing the notion that a friend spans many contexts in one's social life — either because they were present through multiple life stages, or because they have been systematically introduced into multiple social circles. High dispersion of two nodes appear when the two nodes share many mutual friends, yet their mutual friends are not well connected to one another. So in real social networks, a higher dispersion suggests that people knows each other interactive communities which means a deeper relation like spouses and families [1]. In the personalized network models, a high dispersion means the degree of a node is high and connectivity among the neighbors of that node is not good.

Dispersion/Embeddedness:
In the general social networks, this ratio is a improved version to dispersion. It combines the two theories above. A higher quotient means people share communities instead of sharing friends and this is a better indicator to strong ties. It follows intuition that two people knows each other and dives in plenty of common groups individually. They have so many cultural background and could build a much deeper relation easily. However a higher embeddedness would suggest they are good "friend". Therefore, this is a better indicator to detect romantic relation [1]. In the personalized networks, this quotient can be viewed as a normalized version of the dispersion (normalized by the neighborhood sizes). The contribution of the degree to the score is alleviated due to the normalization. Therefore, compared with the standard dispersion, this ratio more emphasizes on the connectivity of the node's neighbors.

Explanation of the same highlighted nodes in the personalized networks:
It can be observed that majority of the max dispersion node, max embeddedness node, and max dispersion/embeddedness node are the same node in these personalized networks. This result is intuitive, since all the three measurements are related to the degree of a node to some extent. Denote N as the degree of a node, then embeddedness of this node is in O(N). Further, if we

assume the distances between the mutual friends in the modified networks of each node are with similar distributions. The dispersion of a node is thus in O(N^2), since the number of neighbor pair combinations is (N-1)*(N-2)/2. Therefore, the dispersion/embeddedness is in O(N). Then we observed that all the three measurements tend to select the node with max degree in the general cases, which is consistent with the results we observed in the personalized networks for node #108, #349, #484, #1087. As to the personalized network of node #1, the highlighted nodes for max dispersion and max quotient are different. Maybe this is because the distribution of mutual friend distances in the modified networks among each node may be different, and some nodes may achieve a larger quotient when their mutual friends are not well connected to one another. However, the general trend is that the node with max quotient is also with relatively high degree, e.g. for the personalized network of node 1, the degree of node with max dispersion and embeddedness is 78 (max degree of non-core nodes), and the degree of the node with max quotient is 69, which is close to the max degree.

1.4 Friend recommendation in personalized networks

1.4.1 Neighborhood based measure

In this project, we will be exploring three different neighborhood-based measures:

1: Common neighbor measure between node i and node j is defined as

$$CommonNeighbors(i,j) = |S_i \cap S_j|$$

2: Jaccard measure between node i and node j is defined as

$$Jaccard(i,j) = \frac{|S_i \cap S_j|}{|S_i \cup S_j|}$$

3: Adamic-Adar measure between node i and node j is defined as

$$AdamicAdar(i,j) = \sum_{k \in S_i \cap S_j} \frac{1}{log(|S_k|)}$$

1.4.2 Friend recommendation using neighborhood based measures

We can use the neighborhood based measures defined in the previous section to recommend new friends to users in the network.We follow the steps listed below:

1. For each node in the network that is not a neighbor of i, compute one of the three measures between the node i and the node not in the neighborhood of i.

2. Then pick t nodes that have the highest scores with node i and recommend these nodes as friends to node i.

1.4.3 Creating the list of users

Having defined the friend recommendation procedure, we can now apply it to the personalized network of node ID 415. Before we apply the algorithm, we need to create the list of users who we want to recommend new friends to. We create this list by picking all nodes with degree 24. We will denote this list as Nr.

Question 16:
Personalized network of Node 415 :
Number of nodes: 160
Number of edges: 1857
Diameter: 2
Number of target users: 11
User node ID: 497 579 601 616 619 628 644 659 660 662 663

Nr = {497 579 601 616 619 628 644 659 660 662 663} ; |Nr| = 11

1.4.4 Average accuracy of friend recommendation algorithm

Question 17:
Compute the average accuracy of the friend recommendation algorithm with 10 iterations that uses:
1.Common Neighbors measure
2.Jaccard measure
3.Adamic Adar measure

Final:
Average accuracy for Common Neighbors measure: **0.8273918**
Average accuracy for Jaccard measure: **0.8066158**
Average accuracy for Adamic Adar measure: **0.829412**

Based on the average accuracy values, **Adamic Adar** algorithm is the best.

Common neighbor algorithm calculates the number of mutual friend between two nodes. The algorithm performs poorly when there are multiple famous nodes in the network, since each famous node may have a lot of followers, and it is possible to recommend a completely unknown

user to another user only because they follow similar famous nodes. For network with few famous nodes, like the personalized networks, common neighbor may perform well. For the Jaccard algorithm, it takes the size of union neighbor set into consideration, which alleviates the effect of multiple famous nodes (neighbor sets of famous nodes are large and will lead to a lower score). However, for the personalized networks, it seems that the Jaccard algorithm does not perform better than the original common neighbor algorithm. Maybe that is because there are not so many famous nodes in the personalized networks, and penalizing the size of union neighbor sets are not effective. For Adamic Adar measure, it refines the simple common neighbor algorithm by weighting each common neighbor. Common neighbors with fewer degree will be associated with heavier weights (just as the formula 1/log(neighbor set size)). This weighting mechanism assumes the nodes with fewer degrees are more telling. It is intuitive, since two users who share one unfamous neighbor may know each other with higher chance, compared with the situation where two nodes share one famous neighbor. For example, two followers of a famous movie star may not know each other. However, if two people both know a specific UCLA student, there is high probability that they may know each other. In the personalized networks, this assumption also works, and that is the reason why Adamic Adar algorithm can achieve higher accuracy in the experiment.

2. Google+ Network

In this part, we will explore the structure of the Google+ network.
We want to create directed personal networks for users who have more than 2 circles.

Question 18:
Total Number of Ego Nodes is 132
Number of Nodes with more than 2 circles is 57
Thus, there are 57 personal networks.

Question 19:
For the 3 personal networks (node ID given below), we plot the in-degree and out-degree distribution of these personal networks. The in-degree distributions of each network are rather different (with different shapes and statistics). The out-degree distributions are similar in a way. Majority of the nodes are with low out-degrees, and the shapes of the distributions are similar.

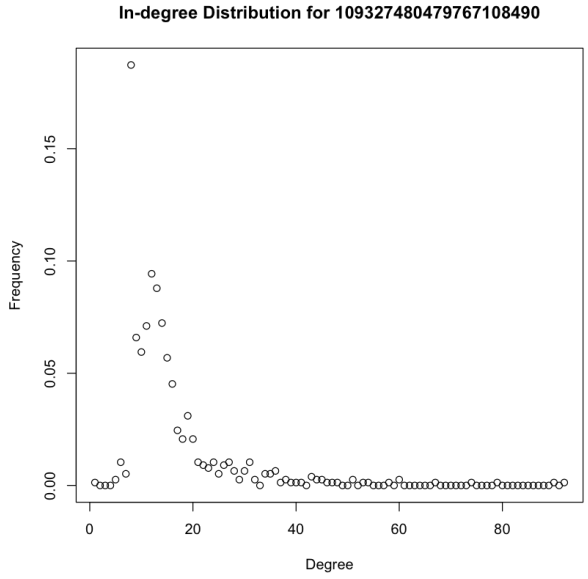

Fig: In-degree distribution of the personalized network with node ID 109327480479767108490

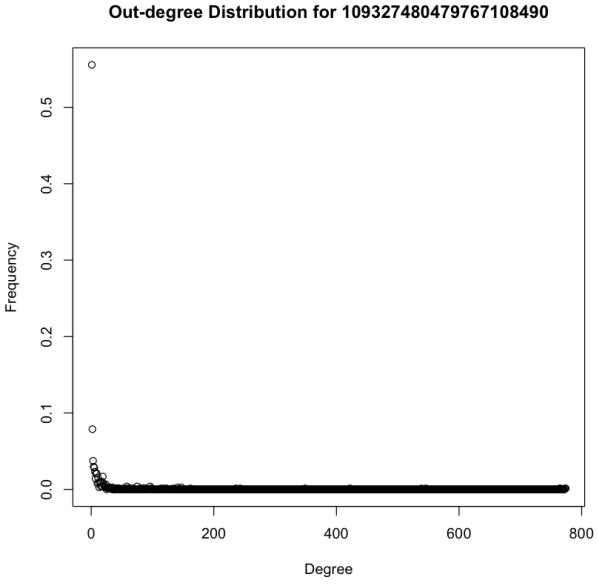

Fig: Out-degree distribution of the personalized network with node ID 109327480479767108490

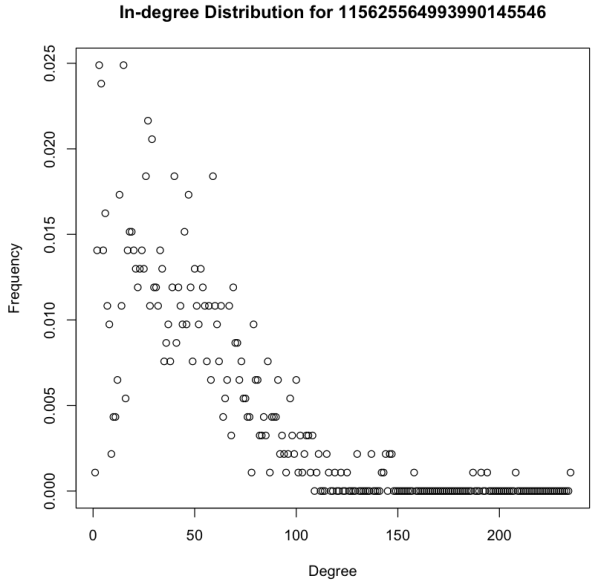

Fig: In-degree distribution of the personalized network with node ID 11562556493990145546

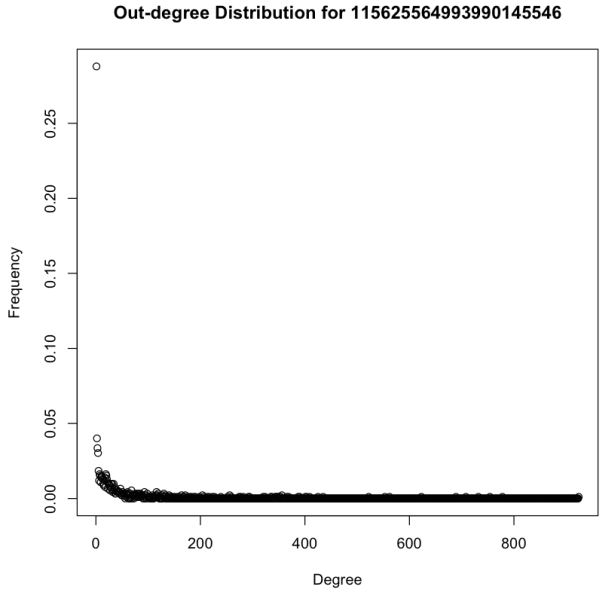

Fig: Out-degree distribution of the personalized network with node ID 11562556493990145546

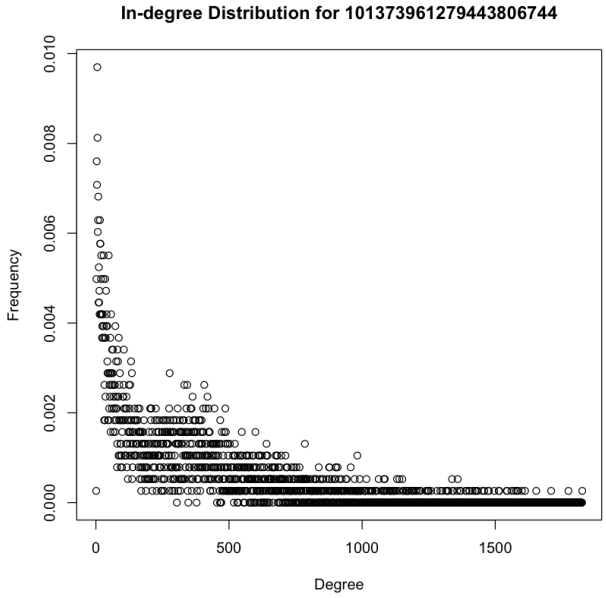

Fig: In-degree distribution of the personalized network with node ID 10137396127944380674

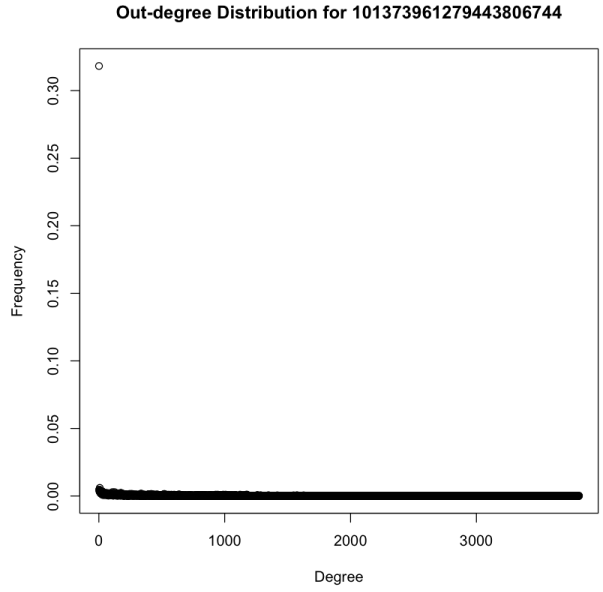

Fig: Out-degree distribution of the personalized network with node ID 10137396127944380674

2.1 Community structure of personal networks

In this part of the project, we will explore the community structure of the personal networks that we created and explore the connections between communities and user circles.

Question 20:

For the 3 personal networks picked in question 19, we extract the community structure of each personal network using Walktrap community detection algorithm. When plotting, we follows the fruchterman.reingold layout for consistency and better visualization.

We think the modularity for the 1st and 2nd networks are similar, and the modularity for the 3rd network is different from the first two networks. As is shown in the tables below, the 2nd network has greatest modularity, and modularity of the 1st network is relatively smaller because the yellow portion heavily overlaps with the blue portion, as show in the figure. Nevertheless, it can be seen that the first two networks have similar modularity scores because the communities of both networks don't overlap much (intra-connections within the communities are denser than the interconnections among different communities). But for the 3rd network, the modularity is clearly lower because there are many overlaps among different communities (the connectivity among communities is high), so modularity score is not very similar to the first two. Besides that, the walktrap is a algorithm based on the random walk, which count the nodes in same communities as long as they fall in the same area after a certain steps of random walk. The first and the second graph has comparatively diverse in-degree distribution compared to third one which has some very high in-degree node. Therefore, most nodes will have high probability falling in the different nodes, which results in a high modularity score.

The modularity scores for all the modularized network are as follows:

Table: Modularity of Selected Nodes

Node ID	Modularity
10932748047976710849 0	0.252765387296677
11562556499399014554 6	0.319472551345825
10137396127944380674 4	0.191090270876884

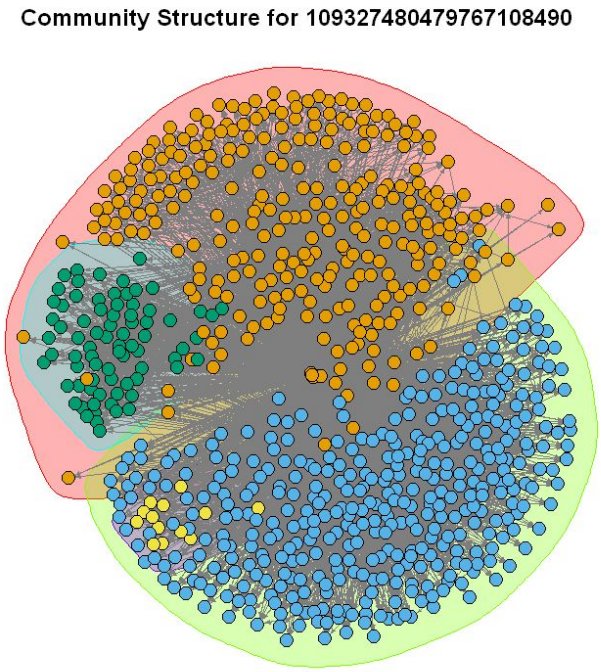

Fig: Community structure for personalized network with node ID 109327480479767108490

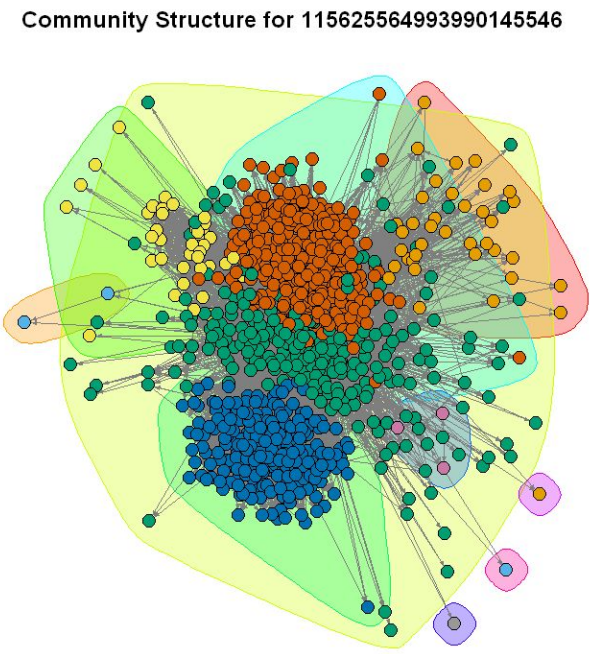

Fig: Community structure for personalized network with node ID 11562556499399014554 6

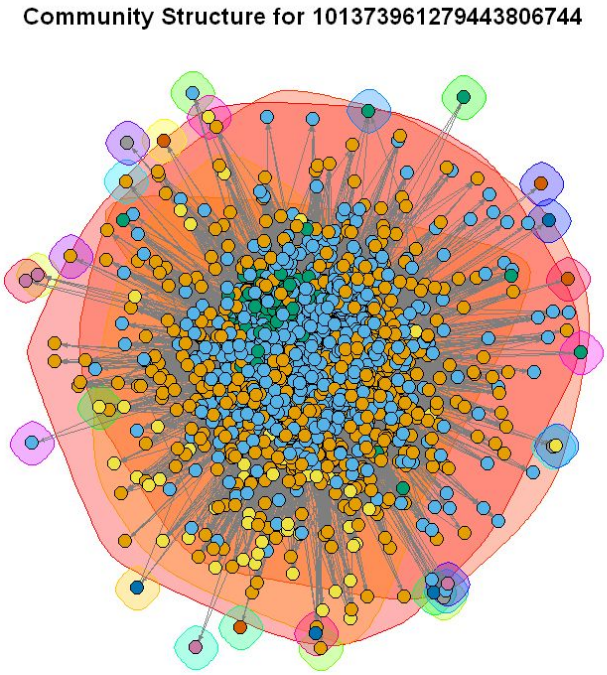

Fig: Community structure for personalized network with node ID 101373961279443806744

Question 21:

Homogeneity:

Homogeneity measures the homogeneity within clusters of the network. According to the equation h = 1 - H(C|K)/H(C), H(C|K) = H(C) => h = 0. That means that circles and communities have no relations, and the homogeneity is smallest(i.e. 0). And when H(C|K) = 0, h = 1. This is the situation where community information is known and thus no unknown information about circles. In other words, given a community, the labels (or circles) can be fully determined (i.e there's only single label in the community) and thus the homogeneity is highest (h = 1). For other circumstances where h not equal to 0 or 1, we can explain "homogeneity" as follows. Consider the case where there are n labels and m communities in the network. If in each community there only exists a few types of labels, then we call the network clusters are of good homogeneity. And if there are multiple different labels in a certain community, then we say the network lacks "homogeneity".

Completeness:

Completeness measures the extent that the nodes with certain labels can be found in one circle. According the the expression c = 1 - H(K|C)/H(K). Firstly, H(K|C) = 0 => c = 1. This means given a label Ci (or circle), we can fully determine the corresponding community Kj, and thus all vertices with label Ci are located in the community Kj. Secondly, H(K|C) = H(K) => c = 0. This shows the situation where given label, we still know nothing about the corresponding community so and c is lowest. In other words, we don't know how the vertices with given labels are distributed across the network. For the cases c neither equals to 1 or 0, we can explain completeness as follows. Imagine that we have n labels {L1, L2, … Ln} and m communities {K1, K2, … Km}. If in each community Kj, we can find almost all vertices labelled as Li, then we say the network is of good completeness. On the contrary, if in order to find all vertices with label Lj, we need to search many different communities, then we say the network lacks "Completeness".

Question 22:

The h and c values for communities structures of the selected networks are shown in the table below. When calculating N (the number of nodes with circle information), we only count the number of unique nodes, i.e. if a node belongs to multiple circles, we only count it once.

Table: Homogeneity and Completeness of Selected Networks

Node ID	Homogeneity	Completeness
109327480479767108490	0.851885115440867	0.329873913536689
115625564993990145546	0.451890303032235	-3.4239623491117
101373961279443806744	0.0038667069813052	-1.5042383879479

Explanations:

Node 109327480479767108490 has the highest homogeneity (0.85, close to 1), which shows that in each community of node 109327480479767108490, there's almost only one single label within the cluster. As for node 115625564993990145546, the homogeneity(0.45) is lower, which means within the clusters there are multiple labels (nodes belonging to several different circles). And node 101373961279443806744 has the lowest homogeneity(0.003, very close to 0). This basically means in a certain cluster, there are many different labels (near the total number of labels). In other words, the information about which cluster the vertex is in will do almost no help for identifying the label of it.

As for Completeness, it's shown that node 109327480479767108490 has small completeness (0.33). This means a community Ki of the network doesn't contain all vertices with label Cj.

Instead, vertices with label Cj spread multiple communities. Another observation is that networks of node 115625564993990145546 and 101373961279443806744 have negative completeness. This is because some vertices belong to many different circles. Therefore the total number of vertices in circles will be greater than N (number of nodes with circle information) due to repeated counts for those nodes. Besides, it can be seen that the completeness of node 115625564993990145546 is even greater than that of node 101373961279443806744 in terms of absolute value. This is because the compared to node 101373961279443806744, node 115625564993990145546 has much more vertices showing in multiple circles, which leads to much higher H(K|C). As a result, the coefficient c will be more negative. It can be concluded that when the computed completeness is negative, the greater the absolute value is, the more overlap there will be among different circles.

References

[1] L. Backstrom, J. Kleinberg, "Romantic Partnerships and the Dispersion of Social Ties: A Network Analysis of Relationship Status on Facebook", arXiv:1310.6753