Project 2
Social Network Mining

Due July 22, 2024, by 11:59 pm

1. Facebook network

1. Structural properties of the Facebook network

Having created the Facebook network, we will study some of the structural properties of the network. To be specific, we will study.

- Connectivity
- Degree distribution

QUESTION 1: A first look at the network:
QUESTION 1.1: Report the number of nodes and number of edges of the Facebook network.

Answer:

Number of nodes in the Facebook network: **4039**
Number of edges in the Facebook network: **88234**

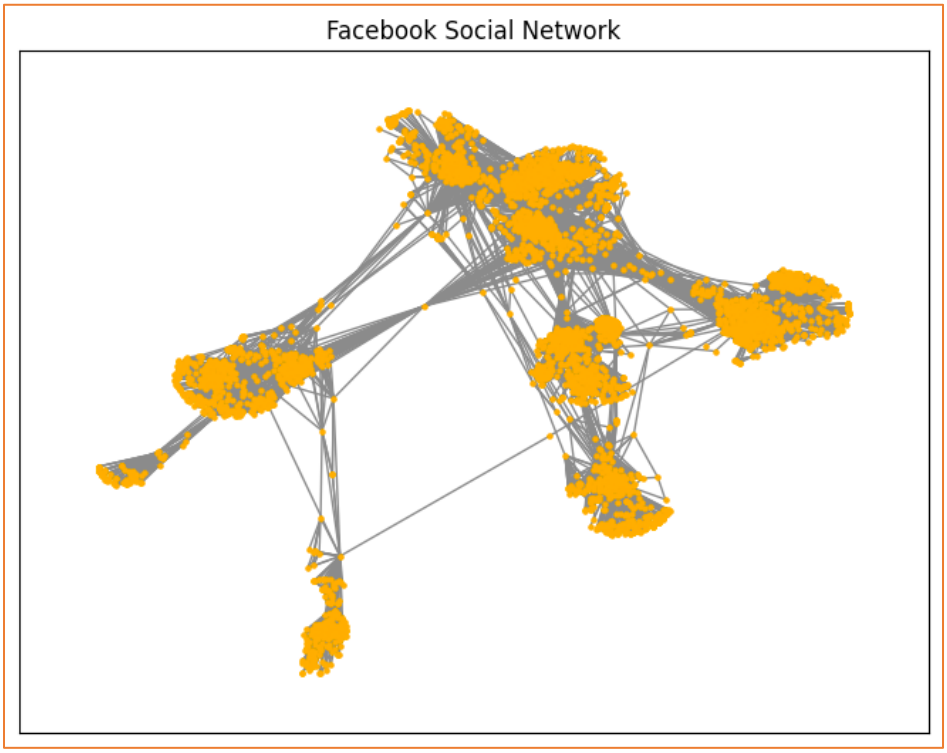


Figure 1

QUESTION 1.2: Is the Facebook network connected? If not, find the giant connected component (GCC) of the network and report the size of the GCC.
Answer:
The Facebook network is fully connected.

QUESTION 2: Find the diameter of the network. If the network is not connected, then find the diameter of the GCC.

Answer:
Diameter of the Facebook network: 8.

QUESTION 3: Plot the degree distribution of the facebook network and report the average degree.

Answer:

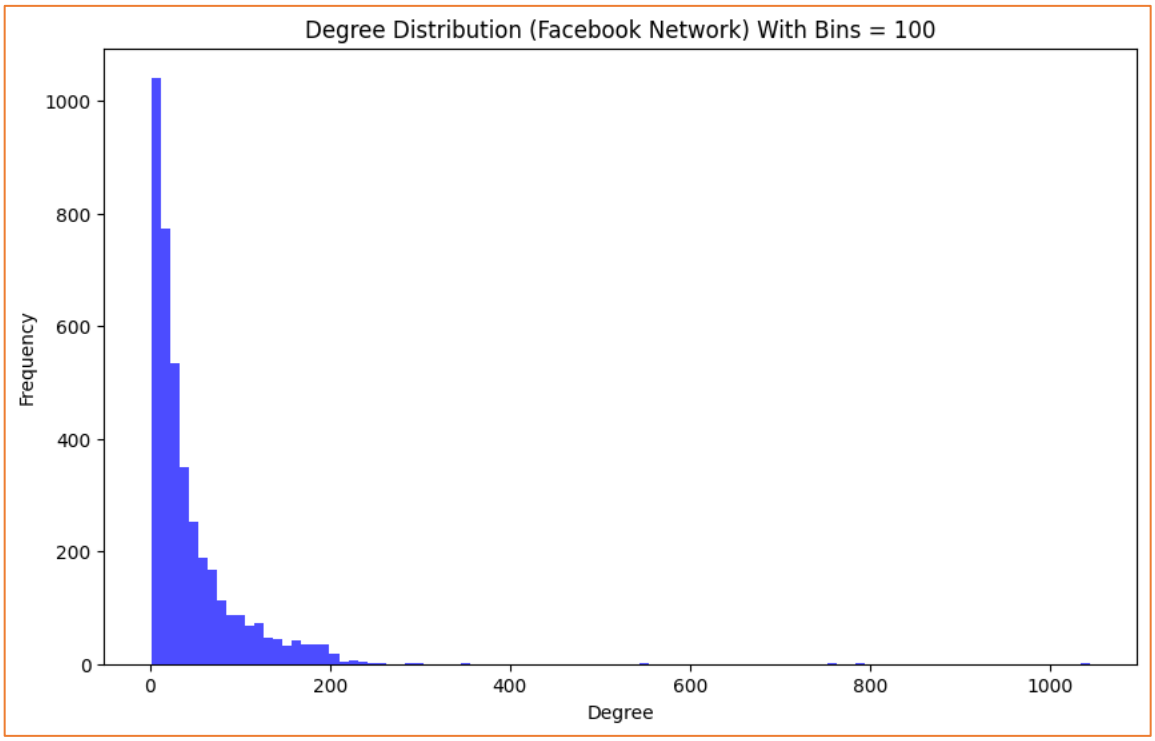


Figure 2

Average degree: 43.69

QUESTION 4: Plot the degree distribution of Question 3 in a log-log scale. Try to fit a line to the plot and estimate the slope of the line.

Answer:

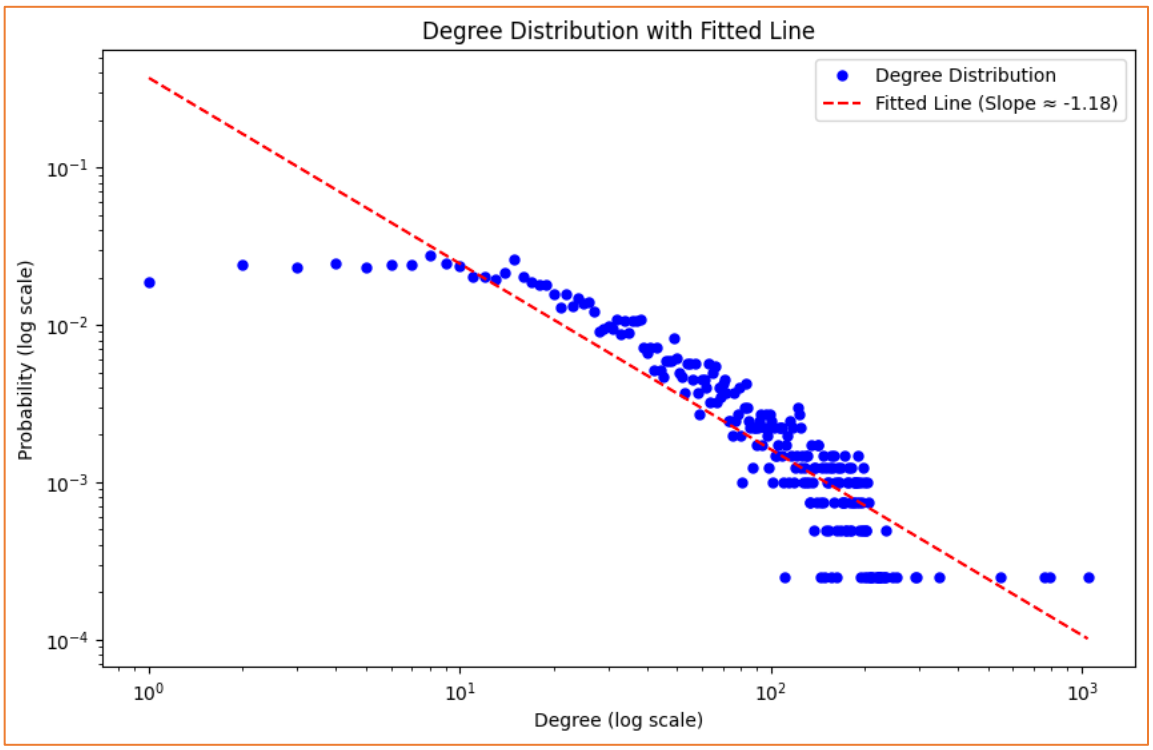


Figure 3

2. Personalized network

A personalized network of an user v_i is defined as the subgraph induced by v_i and it's neighbors. In this part, we will study some of the structural properties of the personalized network of the user whose graph node ID is 1 (node ID in edgelist is 0). From this point onwards, whenever we are refering to a node ID we mean the graph node ID which is 1 + node ID in edgelist.

QUESTION 5: Create a personalized network of the user whose ID is 1. How many nodes and edges does this personalized network have?

Hint Useful function(s): makeegograph

Answer:

In the problem statement, the term "node ID" refers to the graph node ID, which is 1 plus the node ID in the edgelist. So when we mention "user ID," we mean the adjusted node ID corresponding to the user.

User ID 1 (Node 0)

Number of nodes: 348
Number of edges: 2866

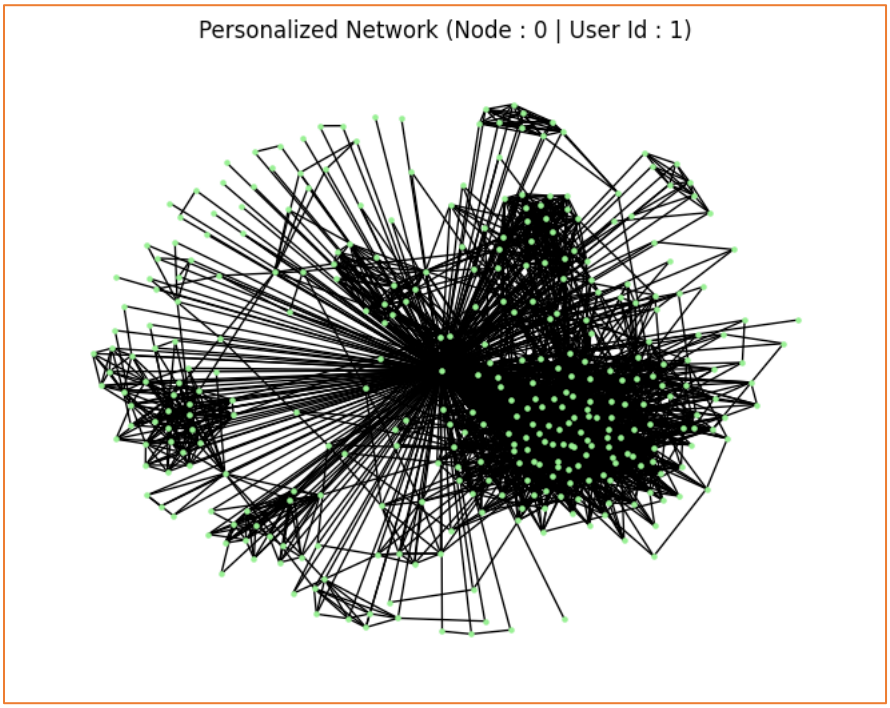


Figure 4

QUESTION 6: What is the diameter of the personalized network? Please state a trivial upper and lower bound for the diameter of the personalized network.

Answer:

Diameter of the entire graph: 8

1. **Trivial Lower Bound**:
    - The diameter of a graph is the maximum shortest path length between any pair of nodes.
    - In a personalized network (where we focus on a specific node), the shortest path from that node to any other node is at least 1 (direct edge).
    - Therefore, the trivial lower bound for the diameter of a personalized network is 1.
2. **Trivial Upper Bound**:
    - Consider a personalized network where the central node has direct edges to all other nodes.
    - In this case, the longest shortest path (diameter) is the path from the central node to the farthest node.
    - So, the trivial upper bound for the diameter of a personalized network is the maximum distance from the central node to any other node.

Based on the explanation in above,

- Trivial upper bound: 8
- Trivial lower bound: 1

QUESTION 7: In the context of the personalized network, what is the meaning of the diameter of the personalized network to be equal to the upper bound you derived in Question 6. What is the meaning of the diameter of the personalized network to be equal to the lower bound you derived in Question 6 (assuming there are more than 3 nodes in the personalized network)?

Answer:

1. Diameter Equal to the Upper Bound (Trivial Upper Bound):

   - When the diameter of the personalized network is equal to the upper bound (which is the diameter of the entire graph), it implies that the user's connections are spread out across the entire graph.
   - In other words, the user has connections that span the farthest distances within the graph. This could indicate a diverse set of connections or interactions with various parts of the network.
   - For example, if the user is a central figure in a large social network, their connections might reach distant clusters or communities.

2. Diameter Equal to the Lower Bound (Trivial Lower Bound):

   - When the diameter of the personalized network is equal to the lower bound (which is 1), it suggests that the user's connections are tightly clustered around them.
   - In this case, the user's neighbors are directly connected to them (within one hop). The personalized network is compact and focused.
   - For instance, if the user interacts primarily with a small group of close friends or colleagues, their personalized network will have a low diameter.

Now let's analyze based on the outcome from question 6.

1. **Upper Bound (Diameter = 8)**:

   - When the diameter of the personalized network equals the upper bound (8), it means that there exists a path (sequence of edges) connecting two nodes in the network that requires at most 8 steps (hops) to traverse.
   - In other words, the longest shortest path (diameter) between any pair of nodes within the personalized network is 8.
   - This upper bound represents the worst-case scenario for reaching any node from the central node (assuming the personalized network has more than 3 nodes).

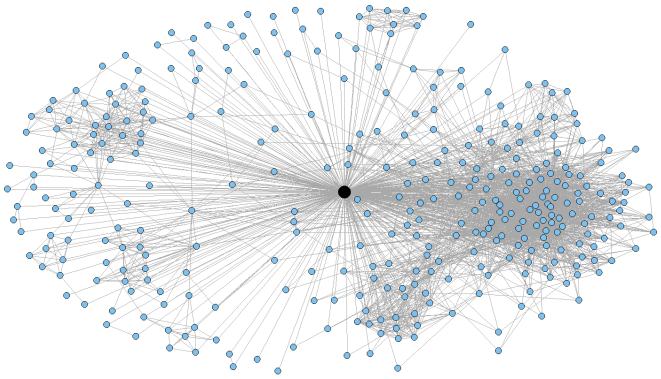
2. **Lower Bound (Diameter = 1)**:

- When the diameter of the personalized network equals the lower bound (1), it means that the shortest path from the central node to any other node is direct (i.e., a single edge connects them).
- In practical terms, this suggests that the central node has direct connections to all other nodes in the personalized network.
- The lower bound of 1 indicates that the network is highly interconnected around the central node.

Remember that these bounds provide insights into the connectivity and reachability within the personalized network.

3. Core node's personalized network

A core node is defined as the nodes that have more than 200 neighbors. For visualization purpose, we have displayed the personalized network of a core node below.



An example of a personal network. The core node is shown in black.

In this part, we will study various properties of the personalized network of the core nodes.

QUESTION 8: How many core nodes are there in the Facebook network. What is the average degree of the core nodes?

Answer:

Number of core nodes: 40
Average degree of core nodes: 279.38

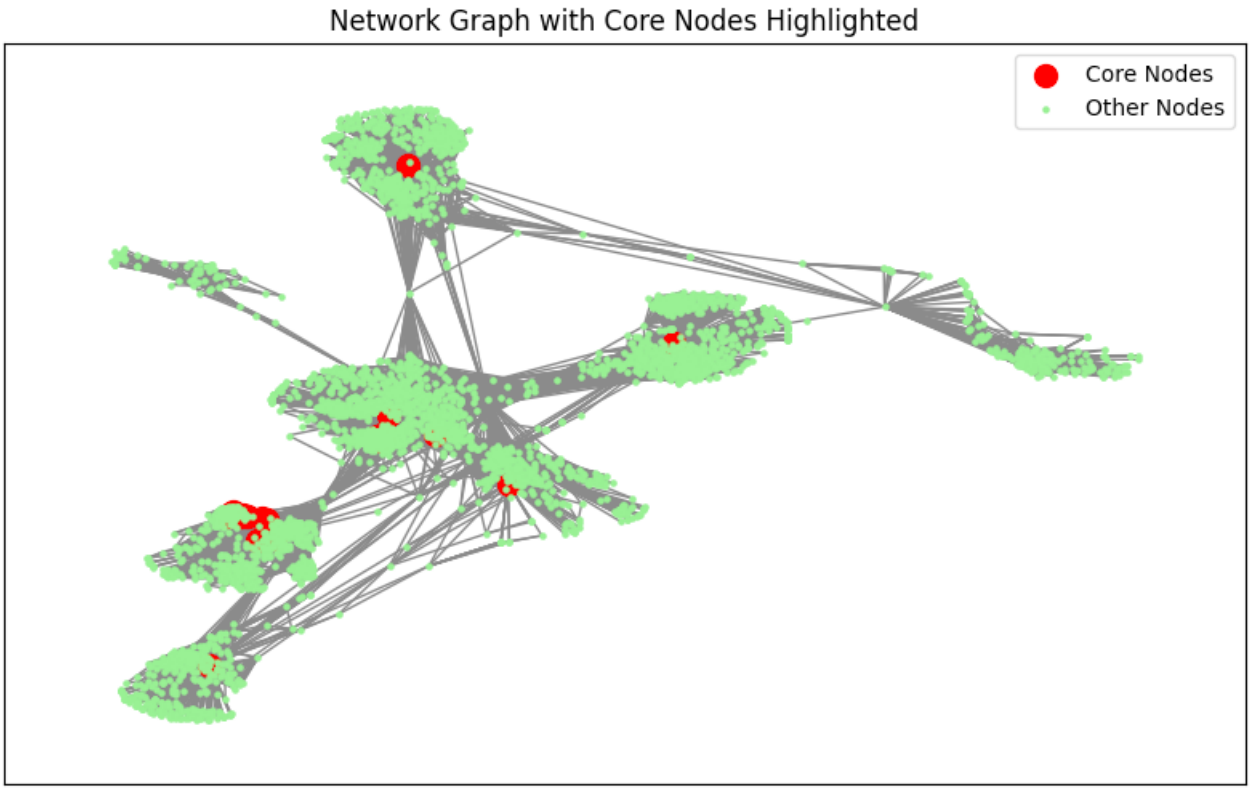


Figure 5

3.1. Community structure of core node's personalized network

In this part, we study the community structure of the core node's personalized network. To be specific, we will study the community structure of the personalized network of the following core nodes: • Node ID 1 • Node ID 108 • Node ID 349 • Node ID 484 • Node ID 1087

QUESTION 9: For each of the above core node's personalized network, find the community structure using Fast-Greedy, Edge-Betweenness, and Infomap community detection algorithms. Compare the modularity scores of the algorithms. For visualization purpose, display the community structure of the core node's personalized networks using colors. Nodes belonging to the same community should have the same color and nodes belonging to different communities should have different color. In this question, you should have 15 plots in total.

Hint Useful function(s): clusterfastgreedy , clusteredgebetweenness , clusterinfomap

Answer:

For each of the above core node's personalized network, we have found the community structure using Fast-Greedy, Edge-Betweenness, and Infomap community detection algorithms and provide the calculated modularity scores of the algorithms in below for comparison.

Community Detection Algorithm	Node	Modularity
Fast-Greedy	1	0.413101
	108	0.435929
	349	0.251715
	484	0.507002
	1087	0.145531
Edge-Betweenness	1	0.353302
	108	0.506755
	349	0.133528
	484	0.489095
	1087	0.027624
Infomap	1	0.389118
	108	0.508223
	349	0.096029
	484	0.515279
	1087	0.026907

Node: 1

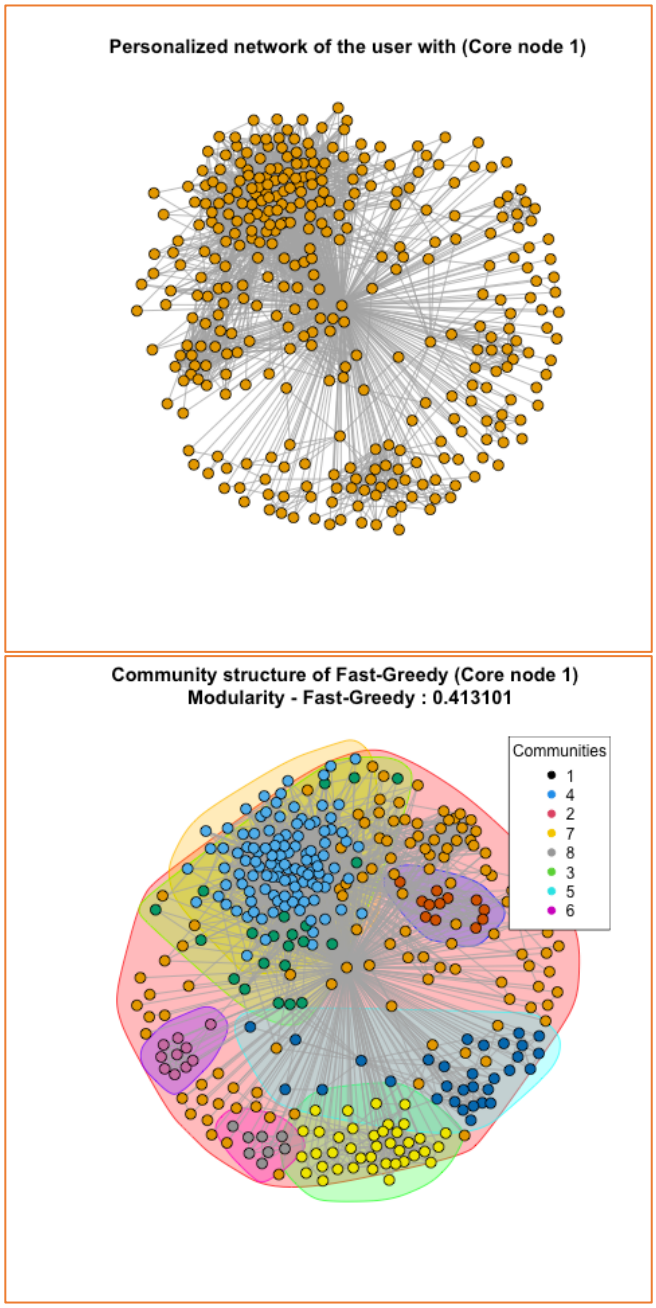
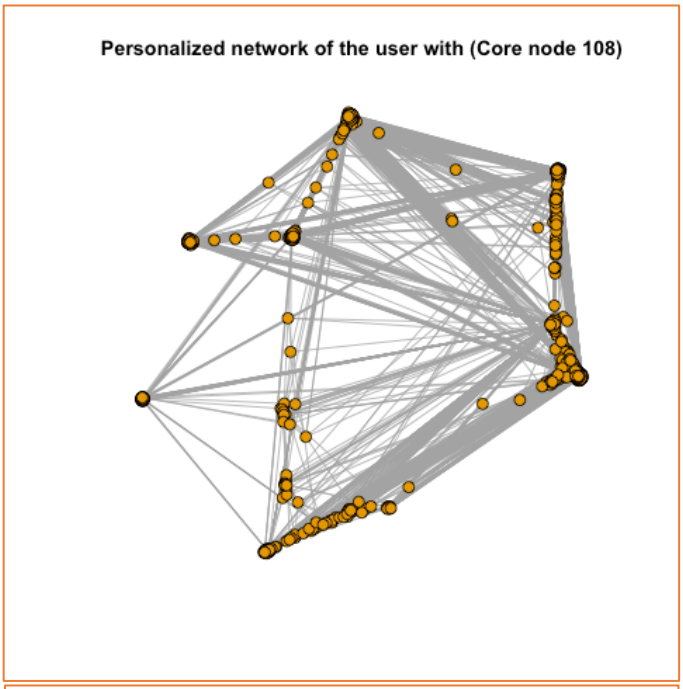


Figure 6

Node: 108



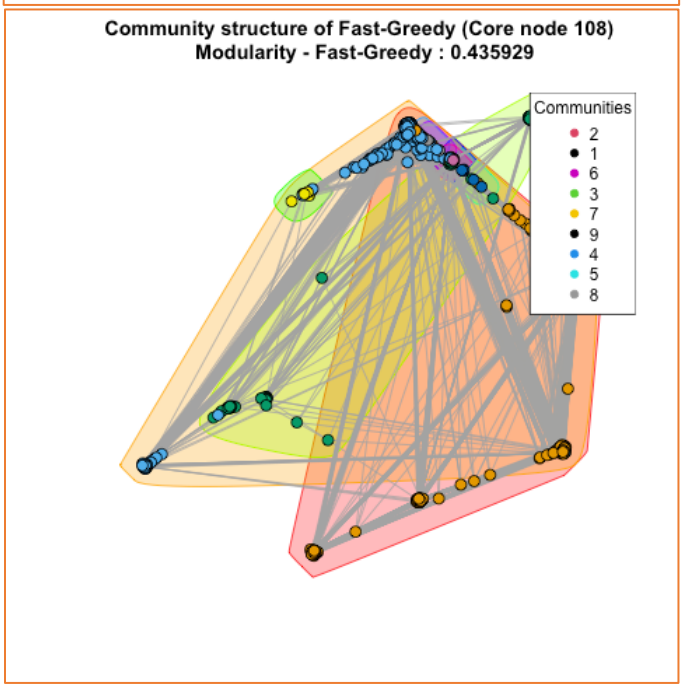


Figure 7

Node: 349

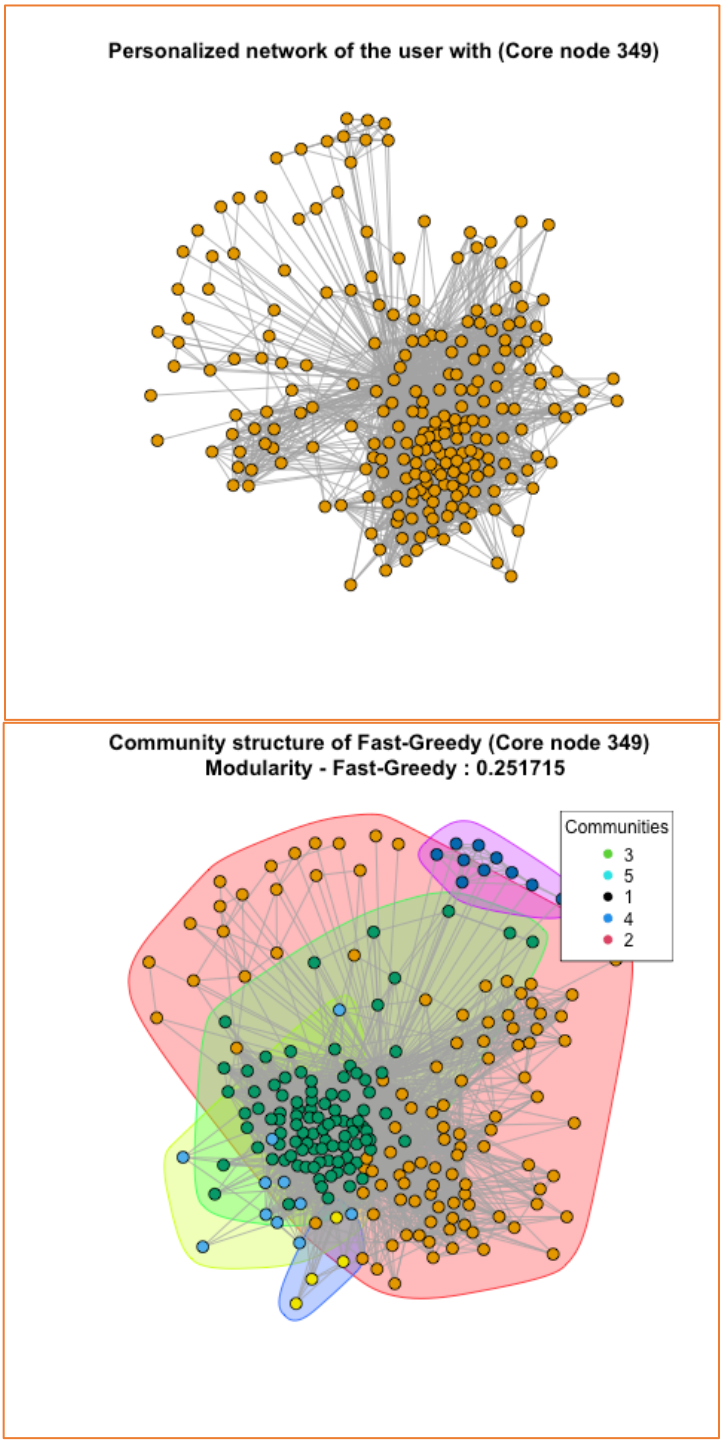


Figure 8

Node: 484

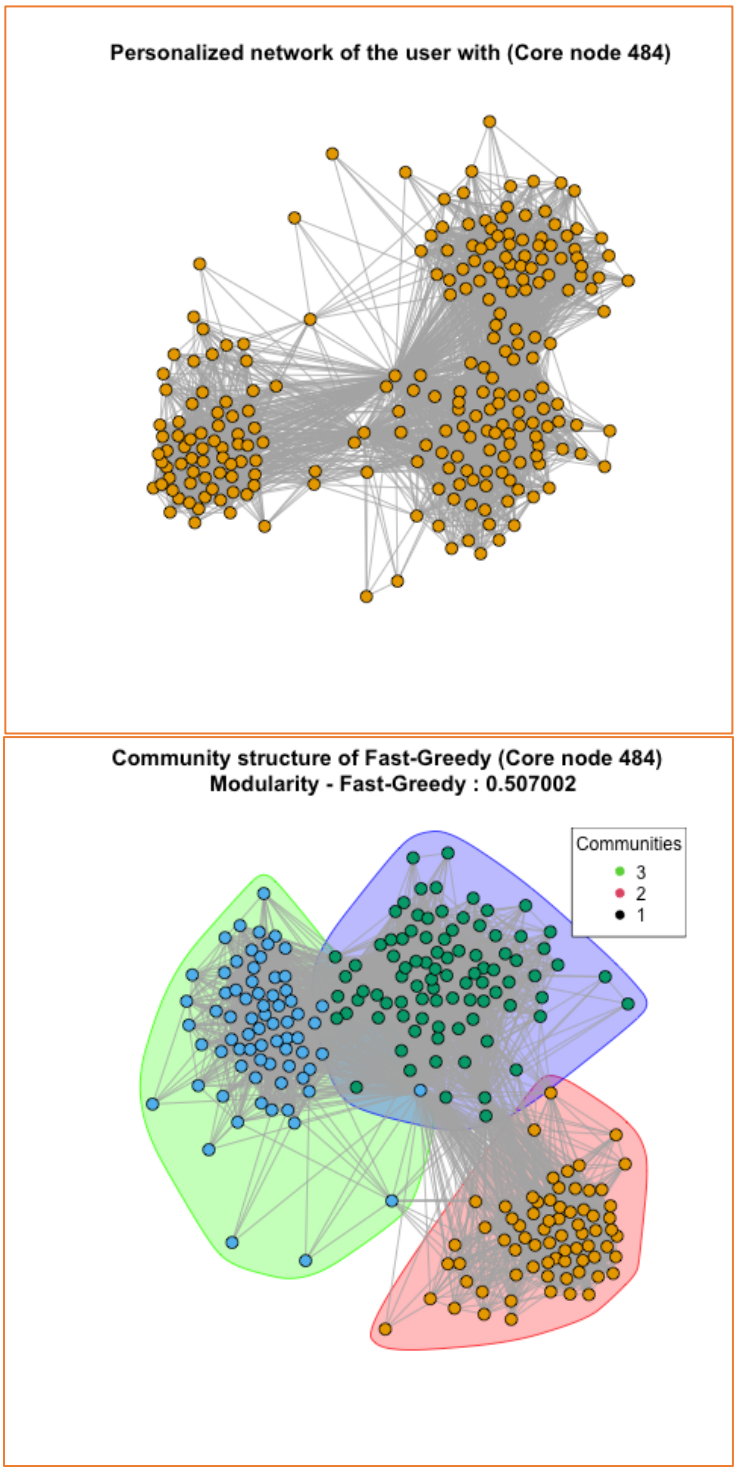


Figure 9

Node: 1087

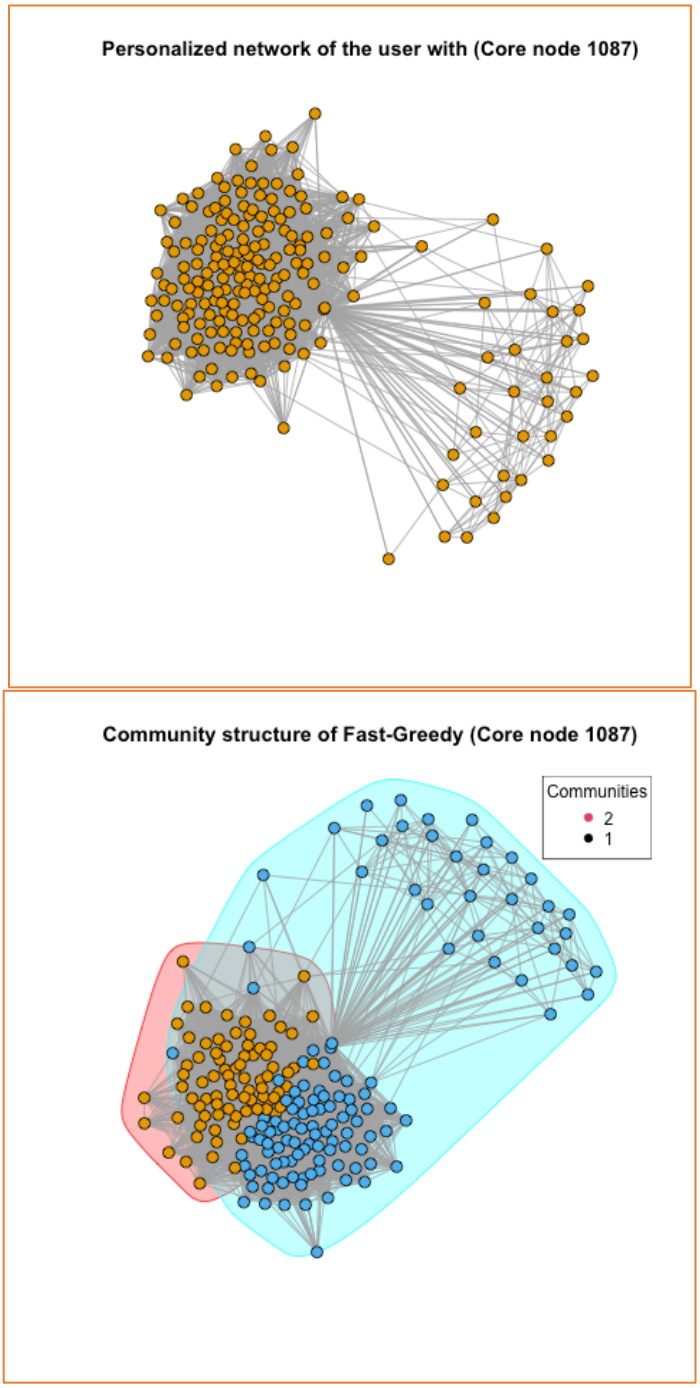


Figure 10

Node 1

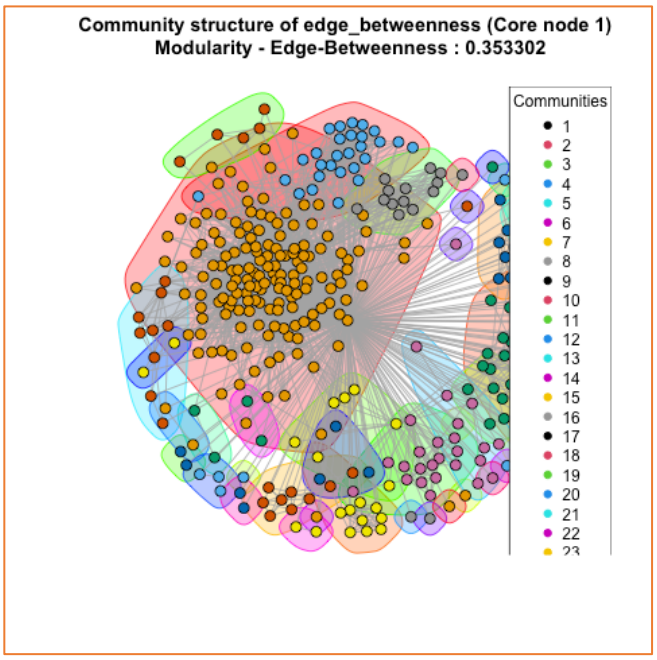


Figure 11

Node: 108

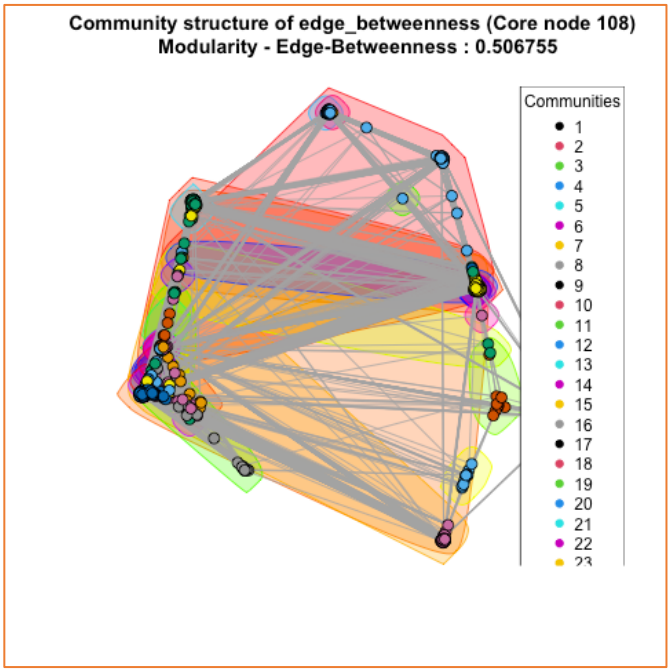


Figure 12

Node: 349

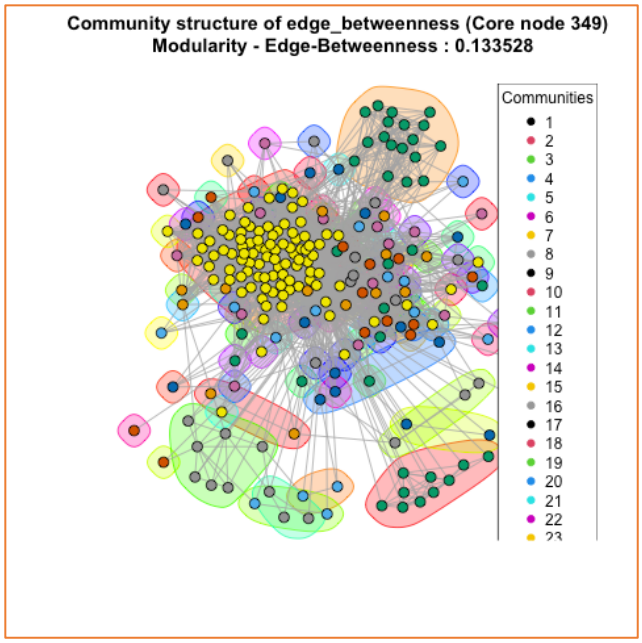


Figure 13

Node: 484

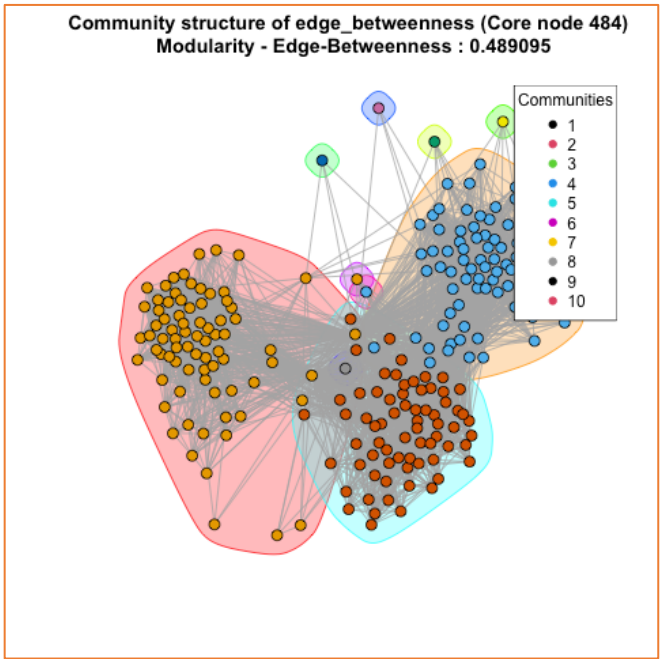


Figure 14

Node: 1087

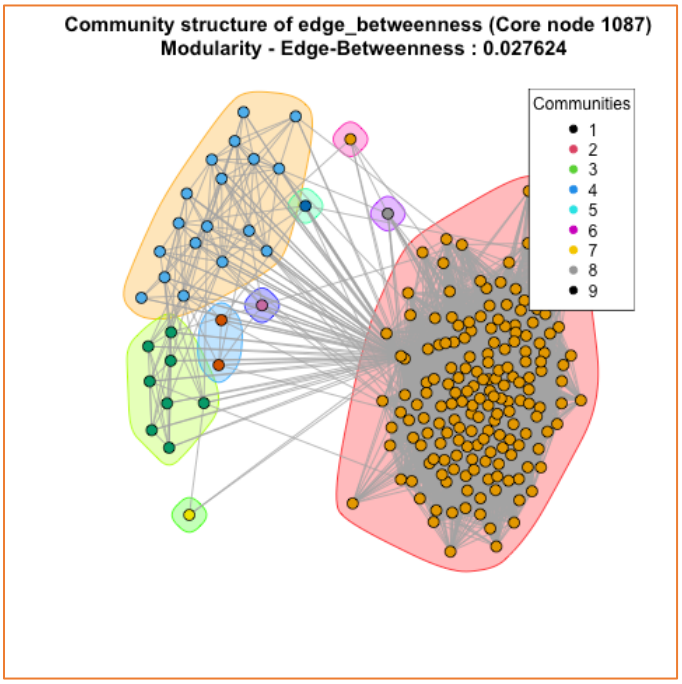


Figure 15

MapInfo (Node 1)

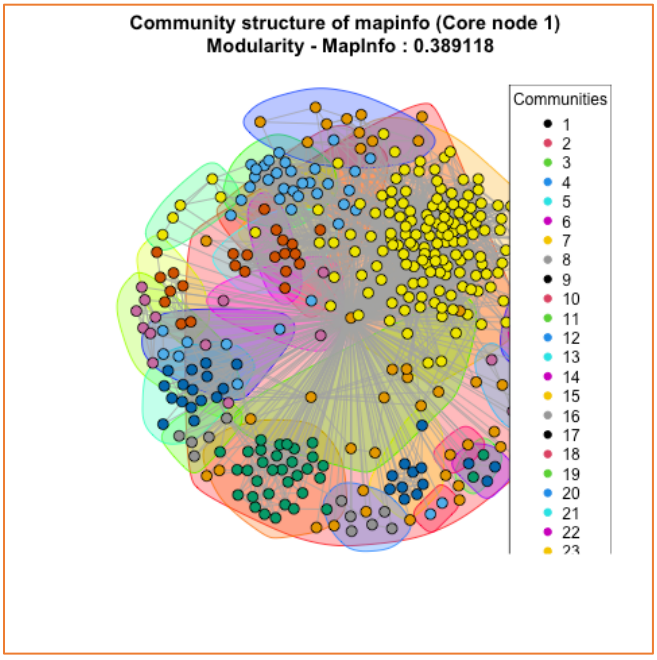


Figure 16

MapInfo (Node 108)

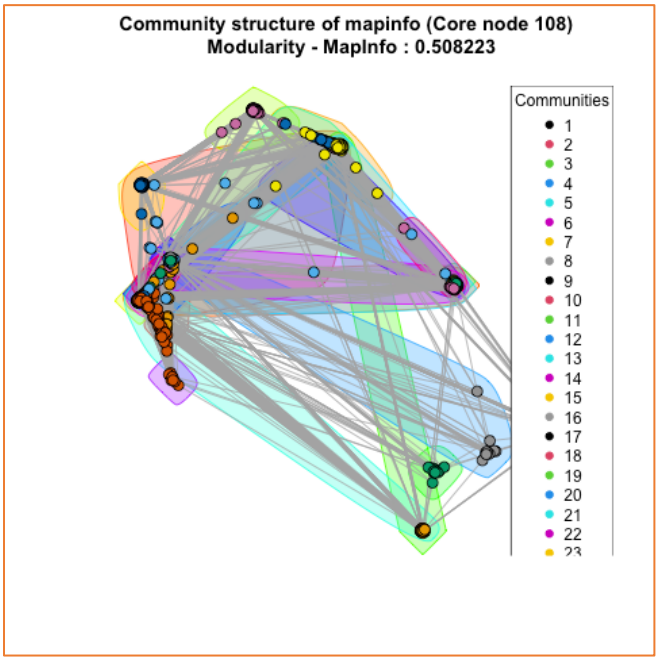


Figure 17

MapInfo (Node 349)

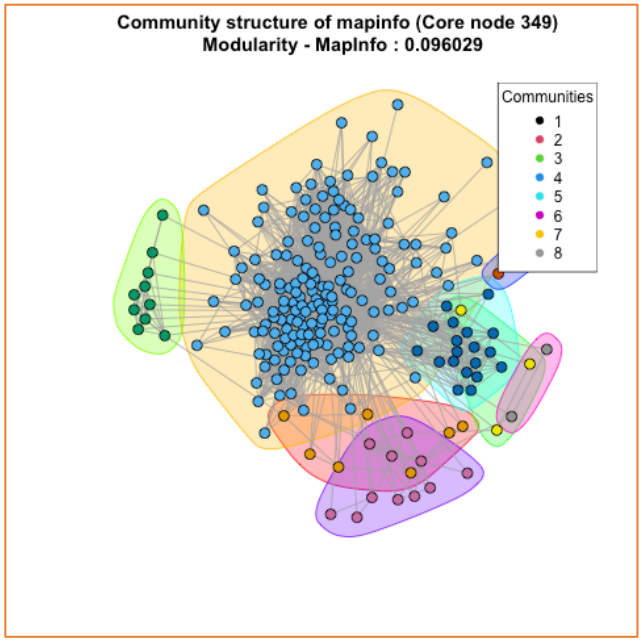


Figure 18

MapInfo (Node 484)

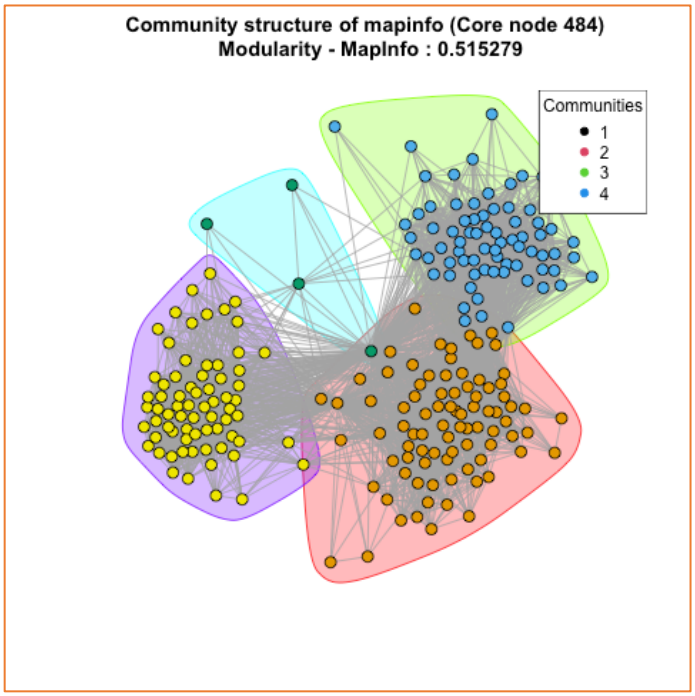


Figure 19

MapInfo (Node 1087)

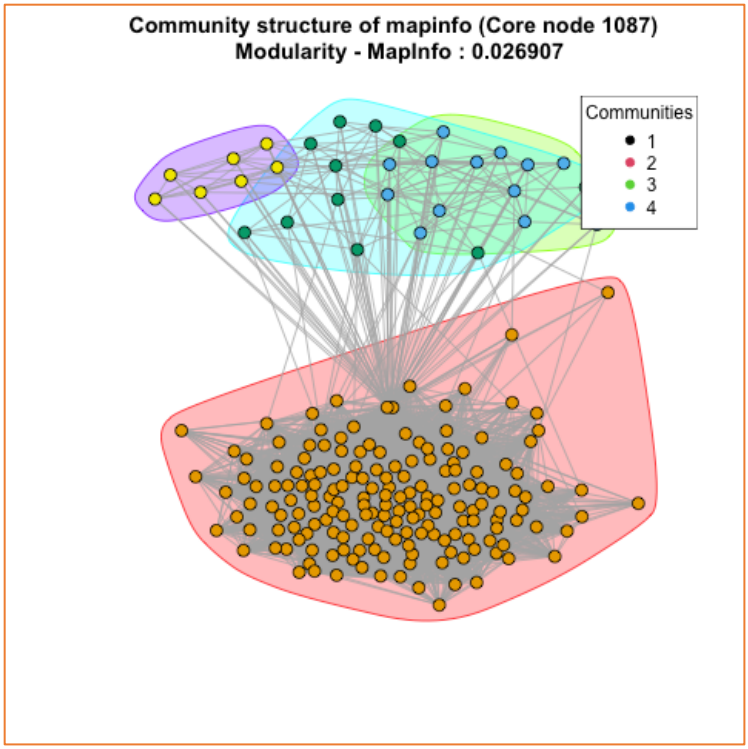


Figure 20

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.

QUESTION 10: For each of the core node's personalized network (use same core nodes as Question 9), remove the core node from the personalized network and find the community structure of the modified personalized network. Use the same community detection algorithm as Question 9. Compare the modularity score of the community structure of the modified personalized network with the modularity score of the community structure of the personalized network of Question 9. For visualization purpose, display the community structure of the modified personalized network using colors. In this question, you should have 15 plots in total.

Answer:

Community Detection Algorithm	Node	Modularity
Fast-Greedy	1	0.44185326886839066
	108	0.45812709371997556
	349	0.24569179594267446
	484	0.539356944374241
	1087	0.14819563195349847
Edge-Betweenness	1	**0.4161461420398303**
	108	**0.5213215763822159**
	349	**0.15056634018755946**
	484	**0.5154412771235044**
	1087	**0.03249529804991403**
Infomap	1	0.4180076594538908
	108	0.5209608294174162
	349	0.24657849262339693
	484	0.5434436792795224
	1087	0.027371594487114542

Community Detected by Fast-Greedy

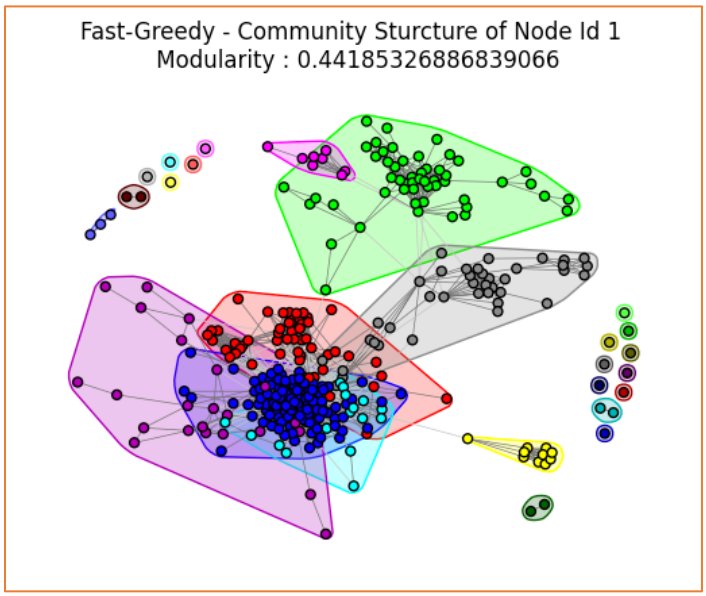


Figure 21

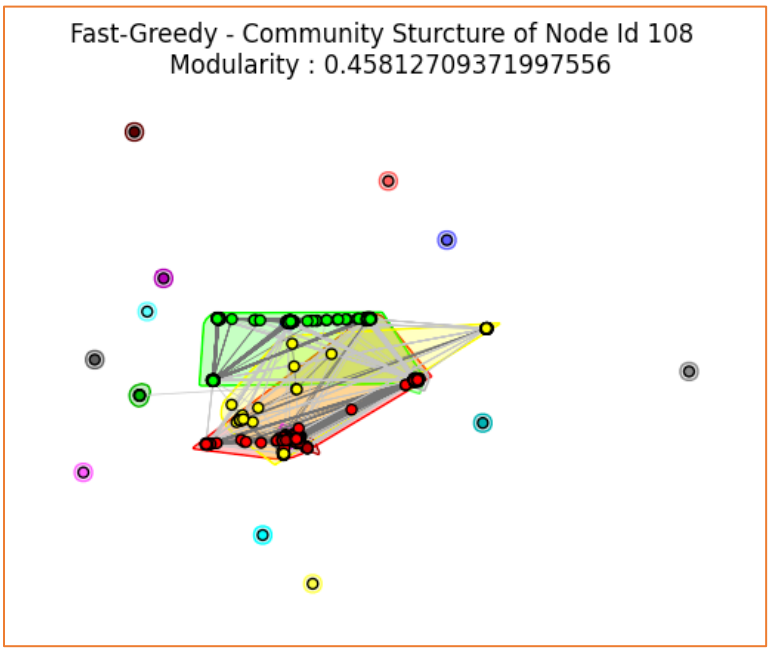


Figure 22

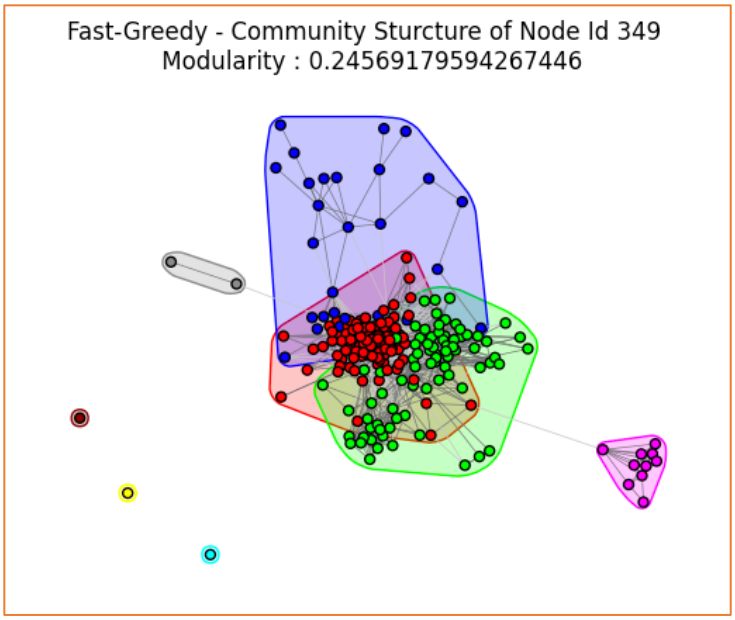


Figure 23

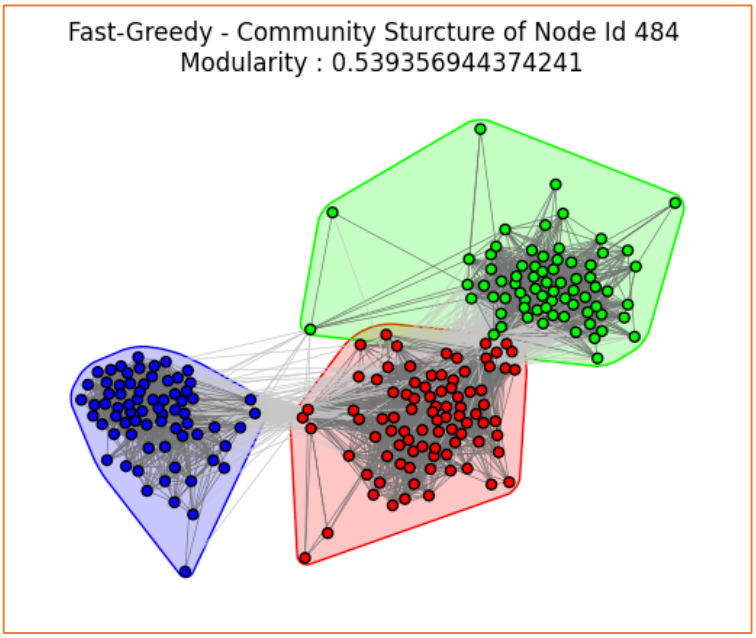


Figure 24

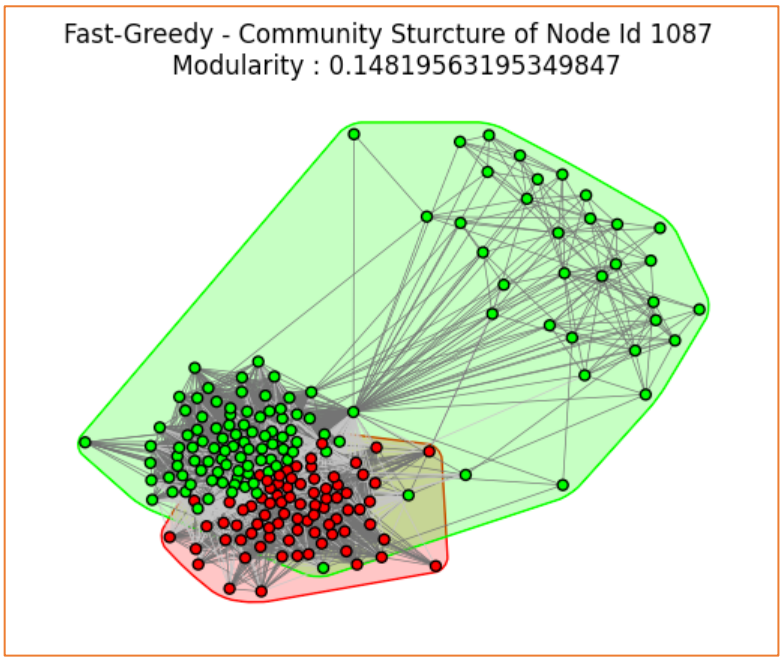

Fast-Greedy - Community Sturcture of Node Id 1087
Modularity : 0.14819563195349847

Figure 25

Community Detected by Edge-Betweenness

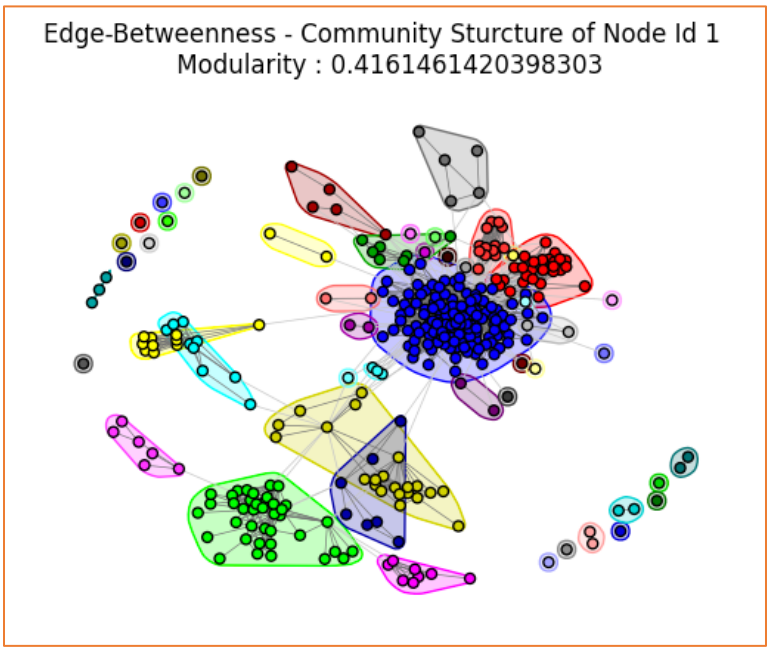


Figure 26

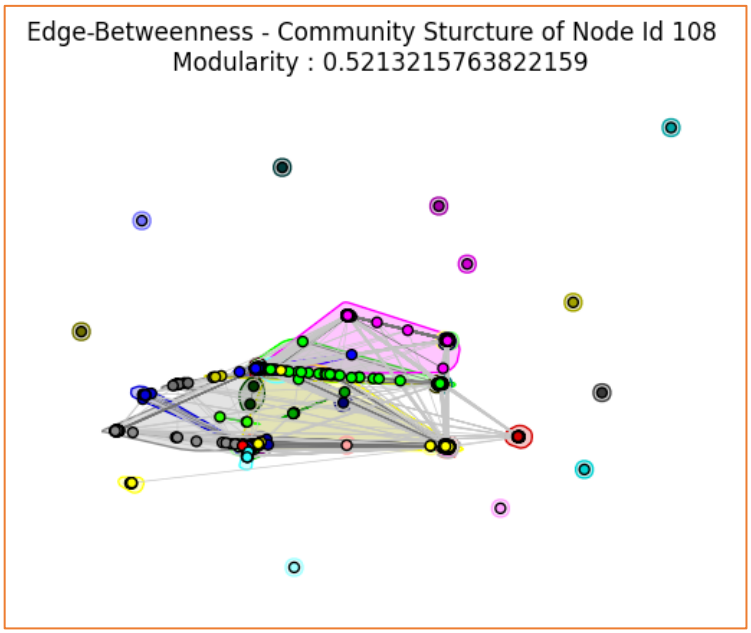


Figure 27

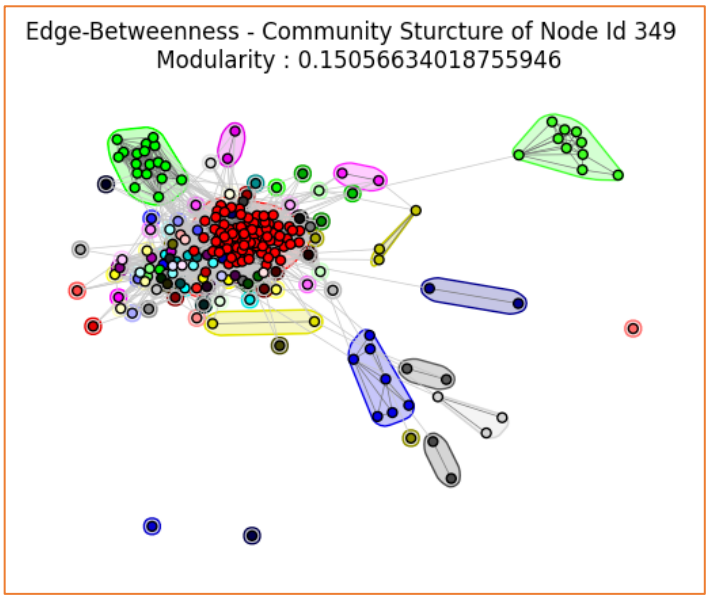


Figure 28

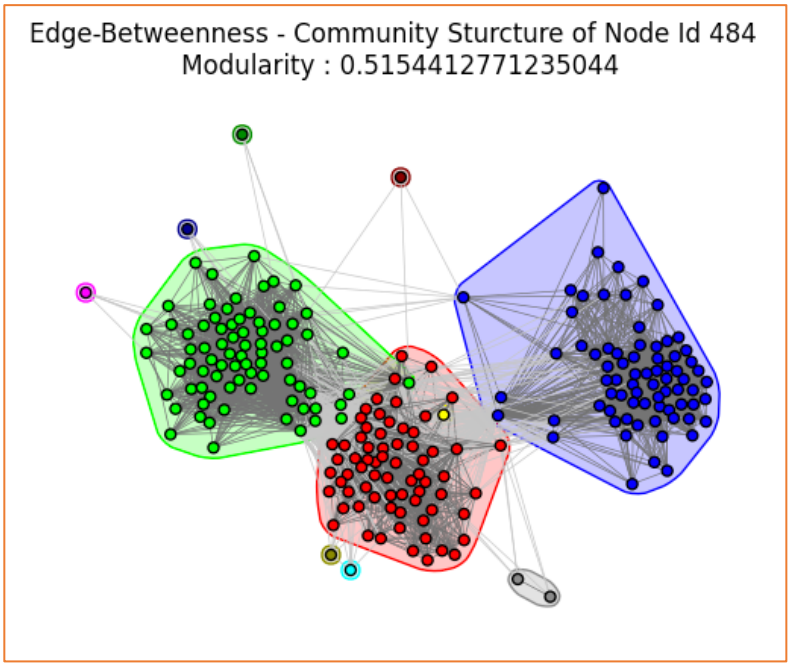


Figure 29

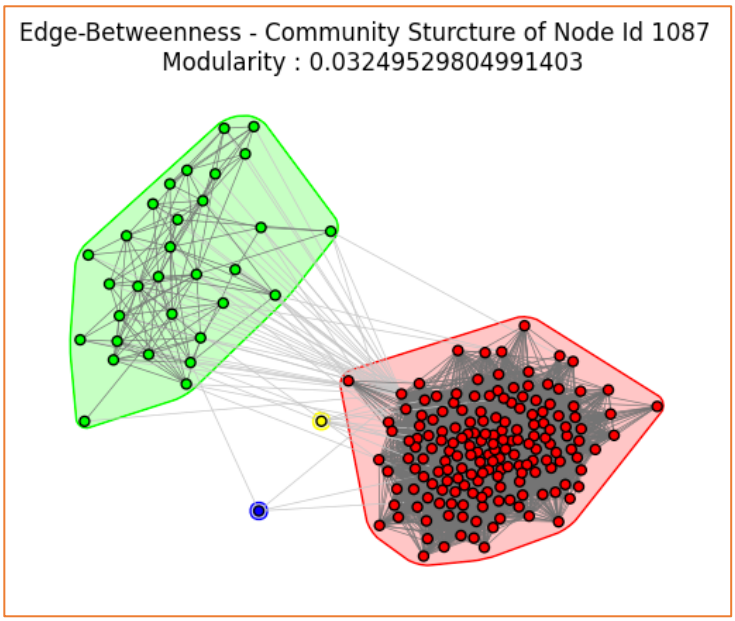


Figure 30

Community Detected by InfoMap

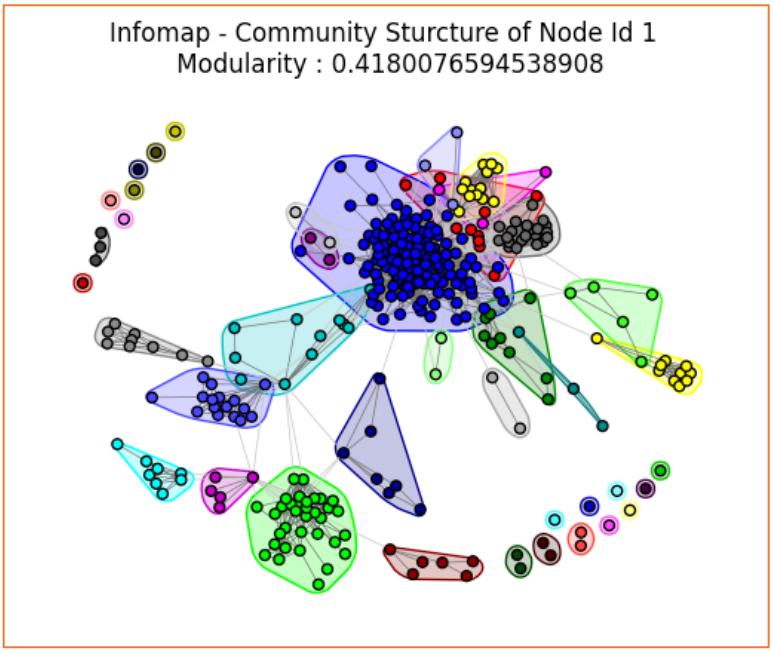


Figure 31

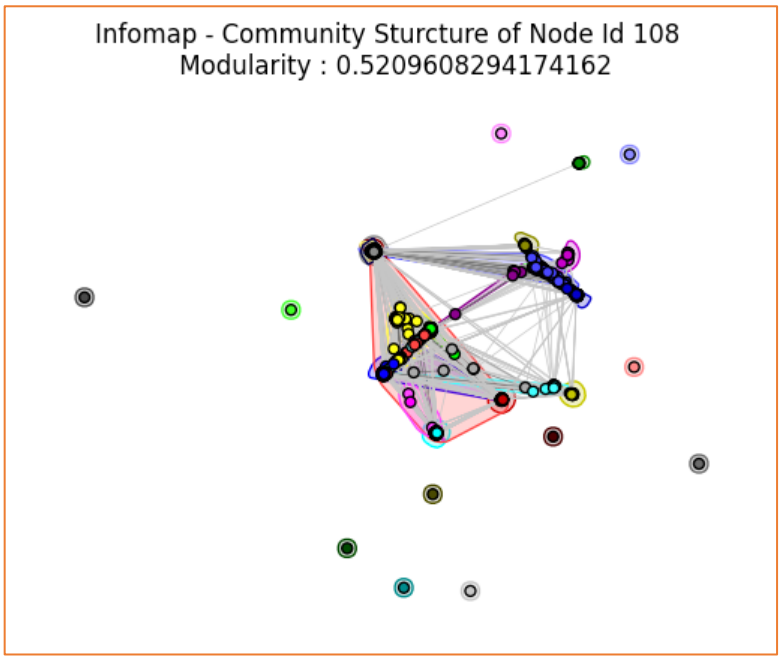


Figure 32

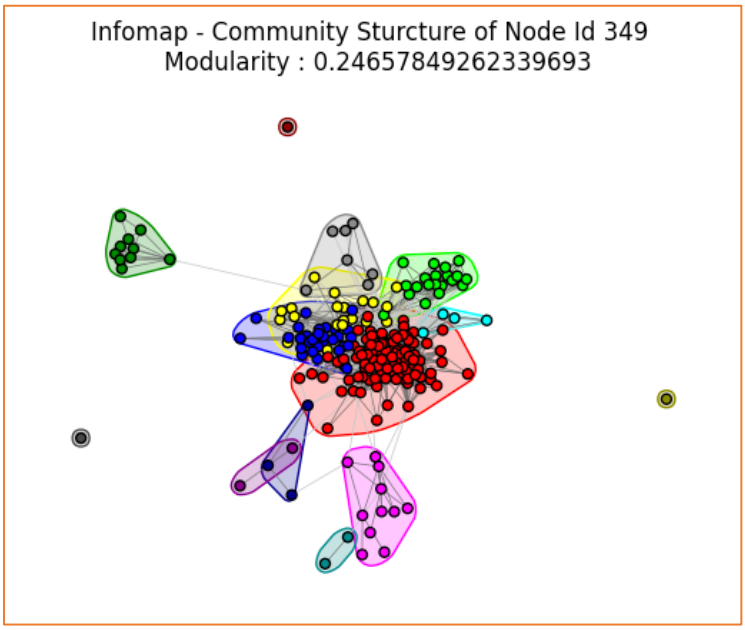


Figure 33

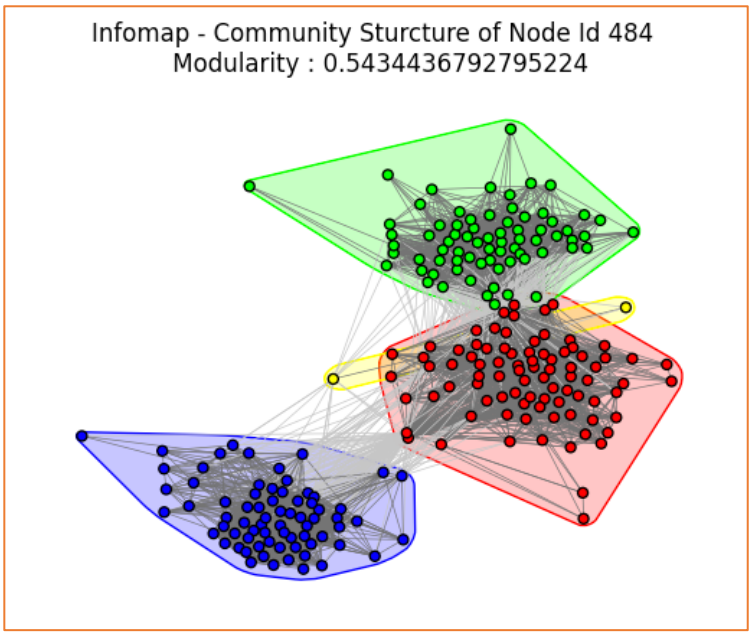


Figure 34

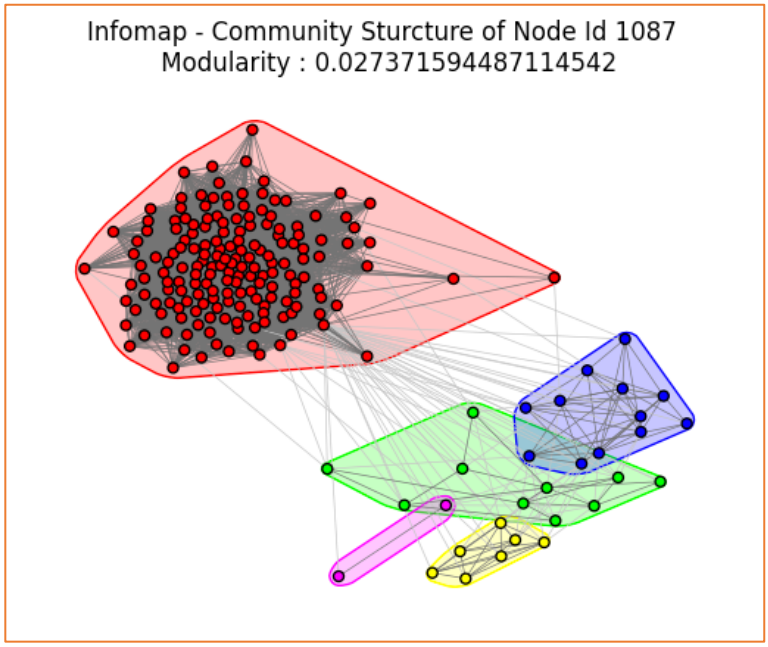


Figure 35

Comparison the modularity score of the community structure of the modified personalized network with the modularity score of the community structure of the personalized network of Question 9.

Community Detection Algorithm	Node	Modularity from Q9	Modularity from Q10
Fast-Greedy	1	0.413101	0.44185326886839066
	108	0.435929	0.45812709371997556
	349	0.251715	0.24569179594267446
	484	0.507002	0.539356944374241
	1087	0.145531	0.14819563195349847
Edge-Betweenness	1	0.353302	0.4161461420398303
	108	0.506755	0.5213215763822159
	349	0.133528	0.15056634018755946
	484	0.489095	0.5154412771235044
	1087	0.027624	0.03249529804991403
Infomap	1	0.389118	0.4180076594538908
	108	0.508223	0.5209608294174162
	349	0.096029	0.24657849262339693
	484	0.515279	0.5434436792795224
	1087	0.026907	0.027371594487114542

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:

- Embeddedness
- Dispersion

For further details on the above characteristics, you can read the paper below:

http://arxiv.org/abs/1310.6753

QUESTION 11: Write an expression relating the Embeddedness between the core node and a non-core node to the degree of the non-core node in the personalized network of the core node.

Answer:

The relationship between embeddedness and the degree of a non-core node in the personalized network of a core node is explained in below.

Embeddedness:

- Embeddedness measures the extent to which two nodes (a core node and a non-core node) share common neighbors within a network.
- It quantifies how well-connected these nodes are to each other through their mutual neighbors.

Degree of a Non-Core Node:

- The degree of a node refers to the number of edges (or neighbors) connected to that node.
- For a non-core node, its degree represents how many other nodes it is directly connected to.

Expression:

- We can express the embeddedness between a core node (denoted as C) and a non-core node (denoted as N) as follows:

  [\text{Embeddedness}(C, N) = \frac{\text{Common Neighbors}(C, N)}{\min(\text{Degree}©, \text{Degree}(N))}]

where:

- \text{Common Neighbors}(C, N) is the number of neighbors shared by both nodes C and N.
- \text{Degree}(C) is the degree of the core node C.
- \text{Degree}(N) is the degree of the non-core node N.

The denominator includes the minimum degree to account for cases where one of the nodes has a lower degree, preventing division by zero. This expression captures the embeddedness specifically between a core node and a non-core node in their personalized network.

QUESTION 12: For each of the core node's personalized network (use the same core nodes as Question 9), plot the distribution histogram of embeddedness and dispersion. In this question, you will have 10 plots.

Hint Useful function(s): `neighbors` , `intersection` , `distances`

Answer:

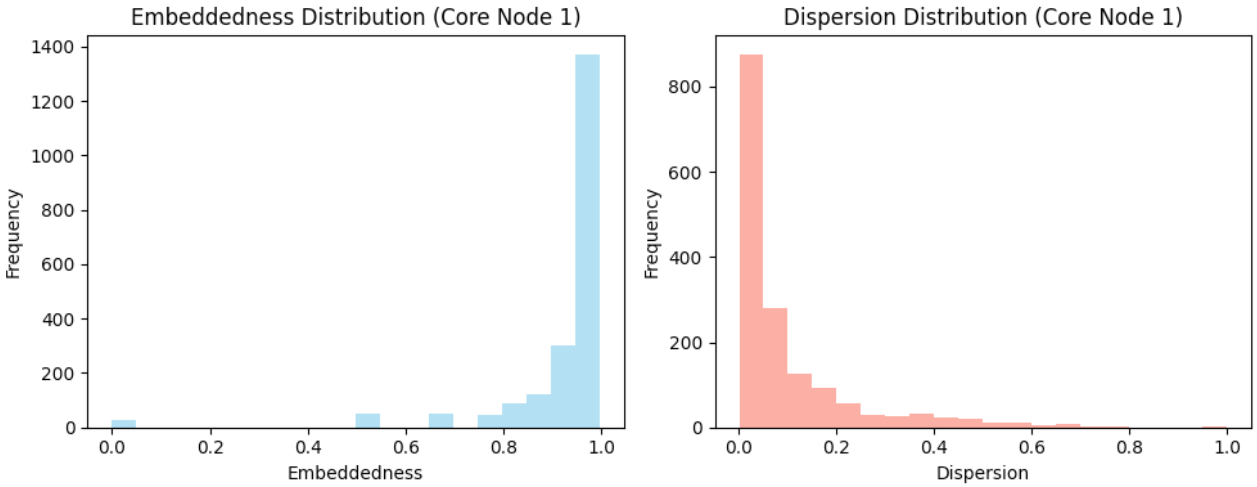


Figure 36

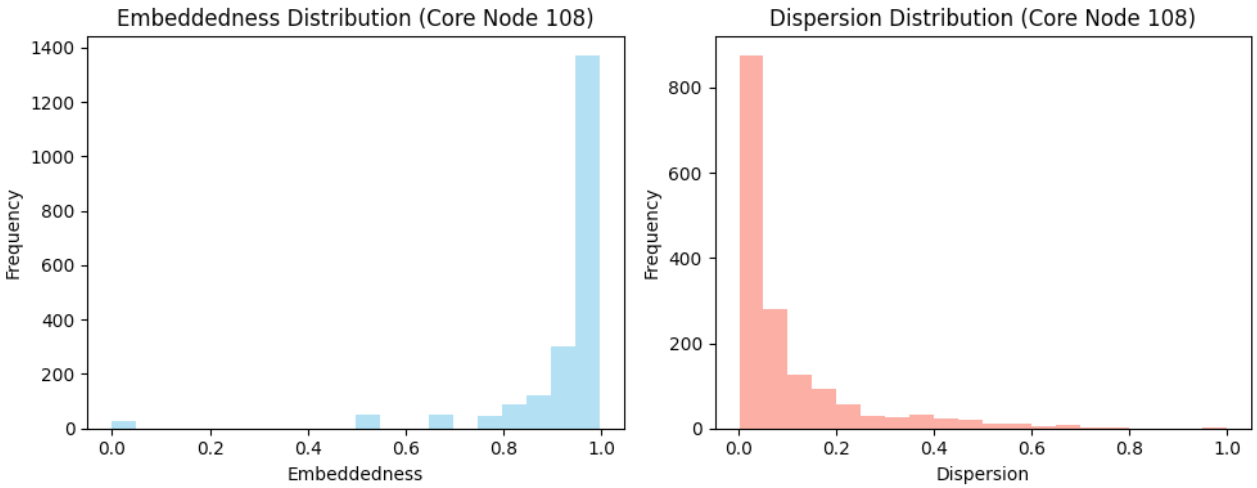


Figure 37

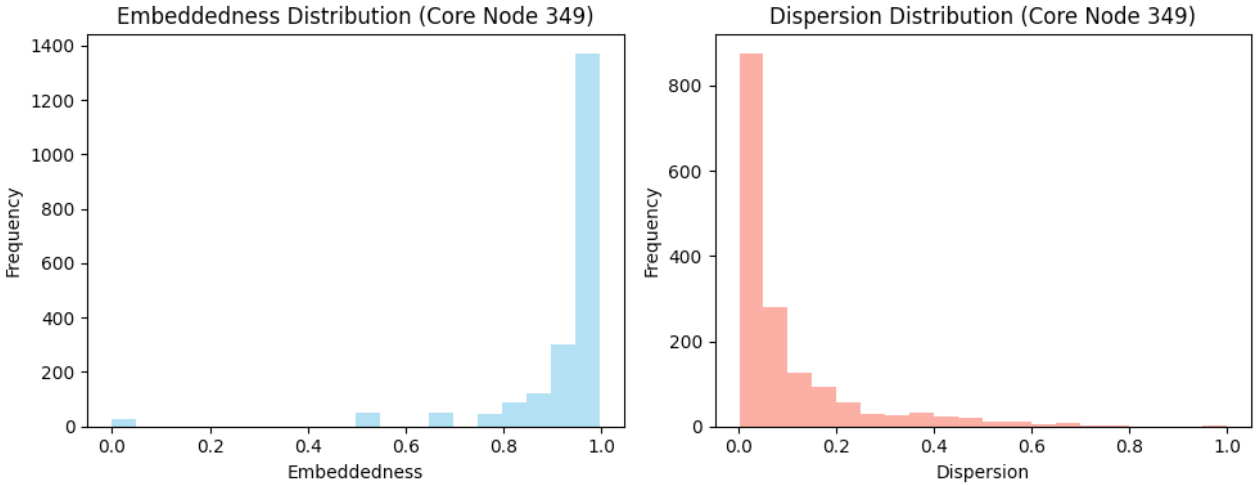


Figure 38

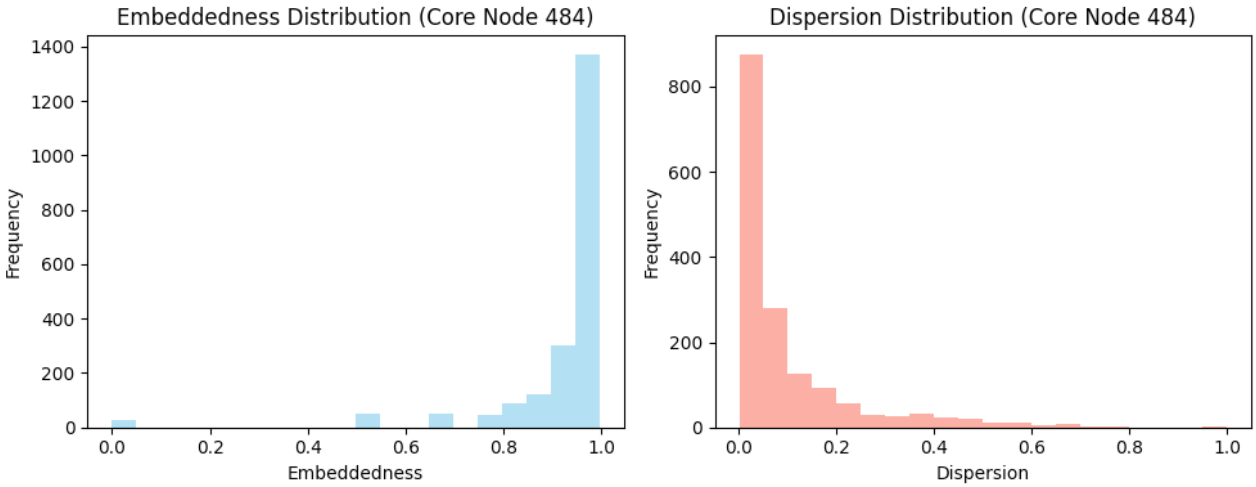


Figure 39

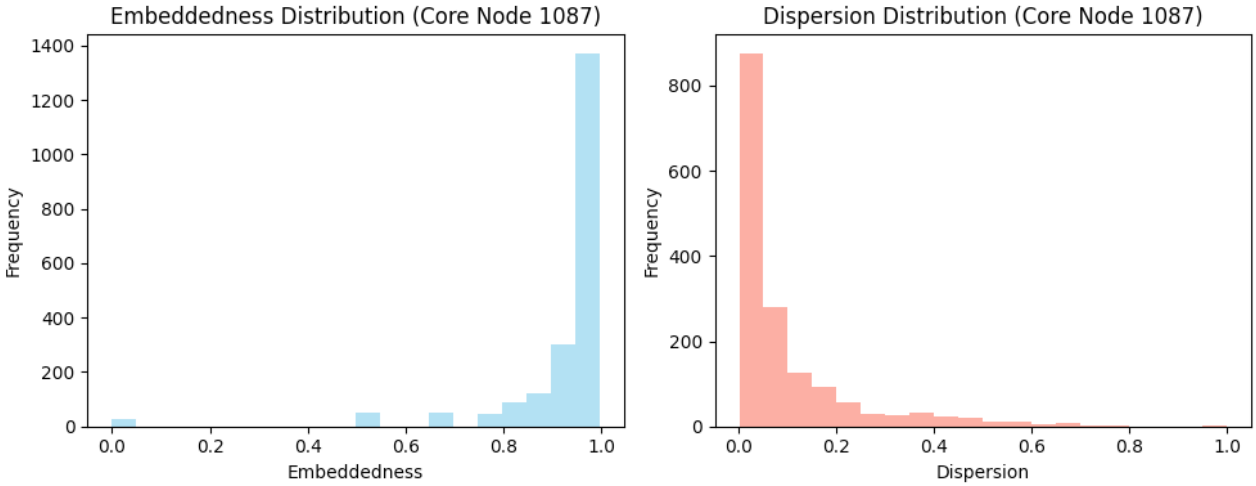


Figure 40

QUESTION 13: For each of the core node's personalized network, plot the community structure of the personalized network using colors and highlight the node with maximum dispersion. Also, highlight the edges incident to this node. To detect the community structure, use Fast-Greedy algorithm. In this question, you will have 5 plots.

Answer:

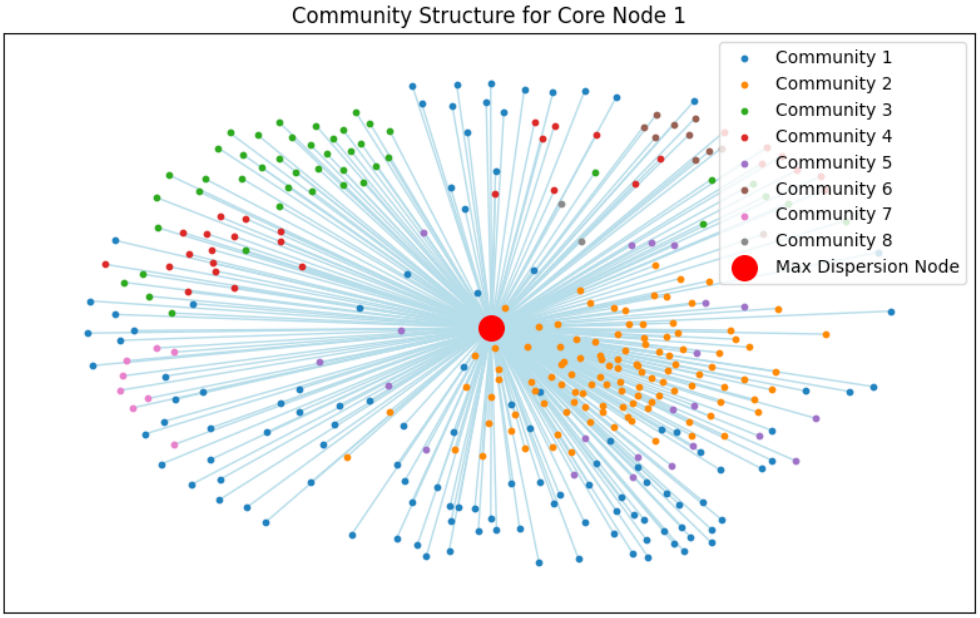


Figure 41

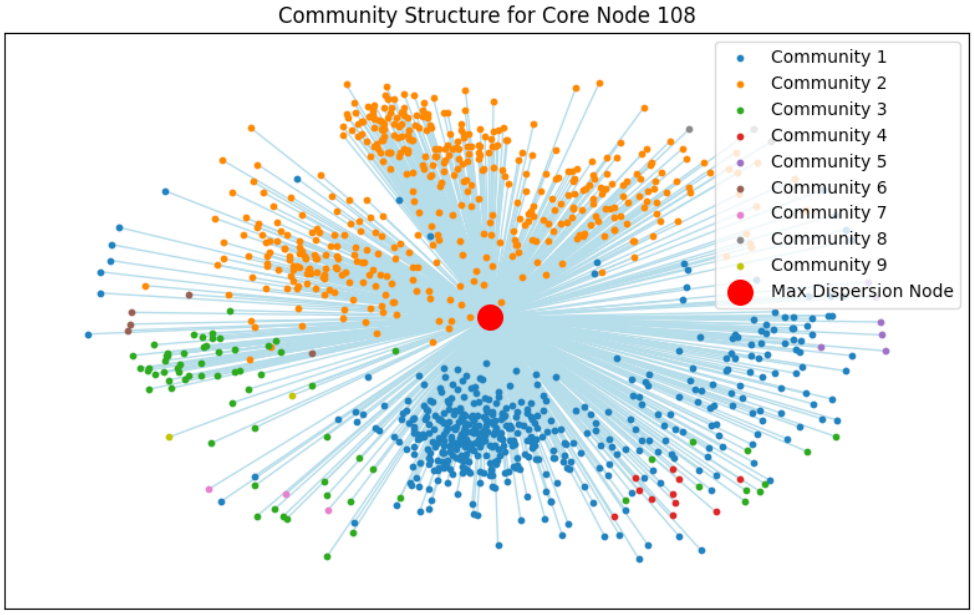


Figure 42

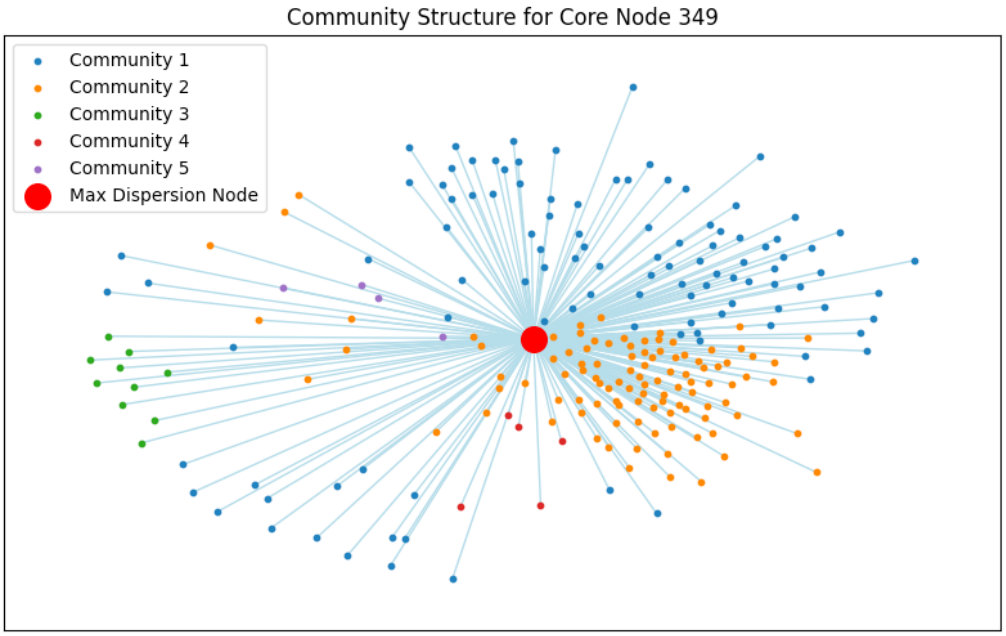


Figure 43

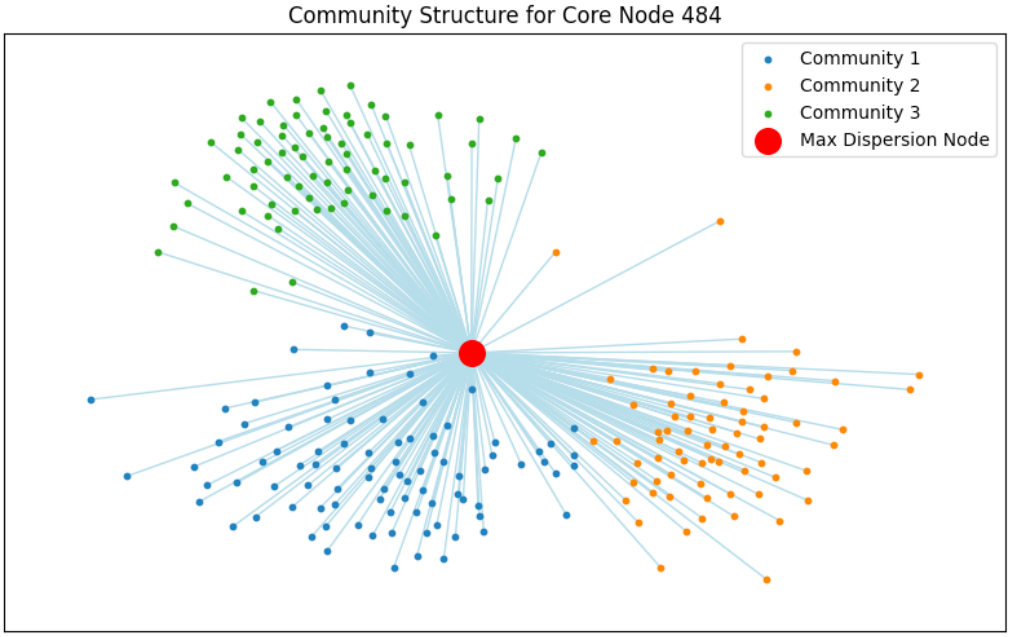


Figure 44

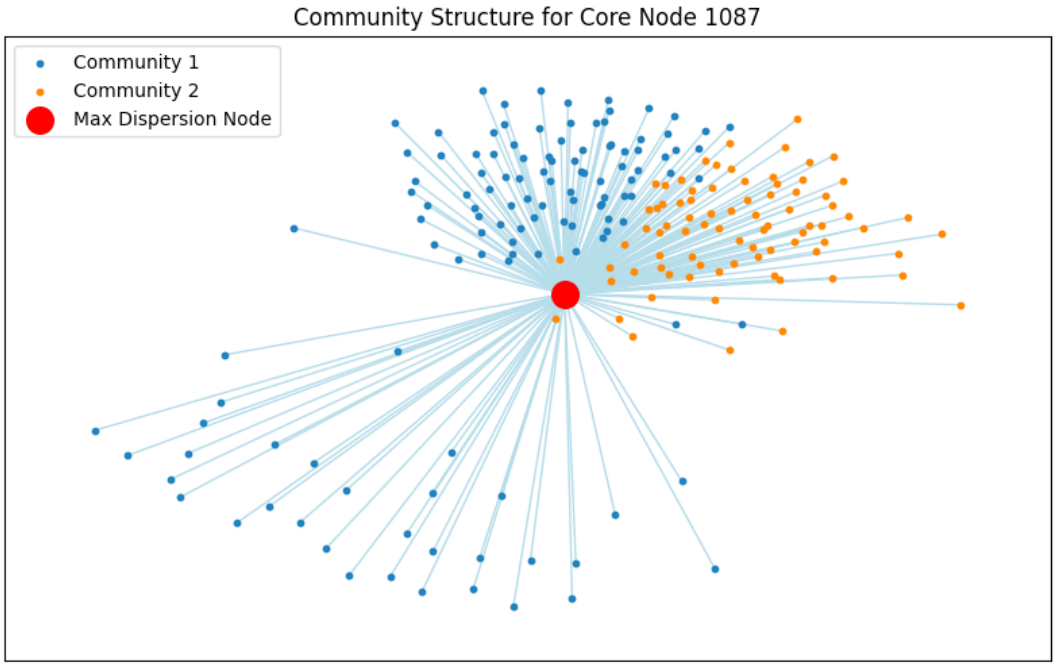

Community Structure for Core Node 1087
Community 1
Community 2
Max Dispersion Node

Figure 45

QUESTION 14: Repeat Question 13, but now highlight the node with maximum embeddedness and the node with maximum dispersion / embeddedness

(excluding the nodes having zero embeddedness if there are any). Also, highlight the edges incident to these nodes. Report the id of those nodes.

Answer:

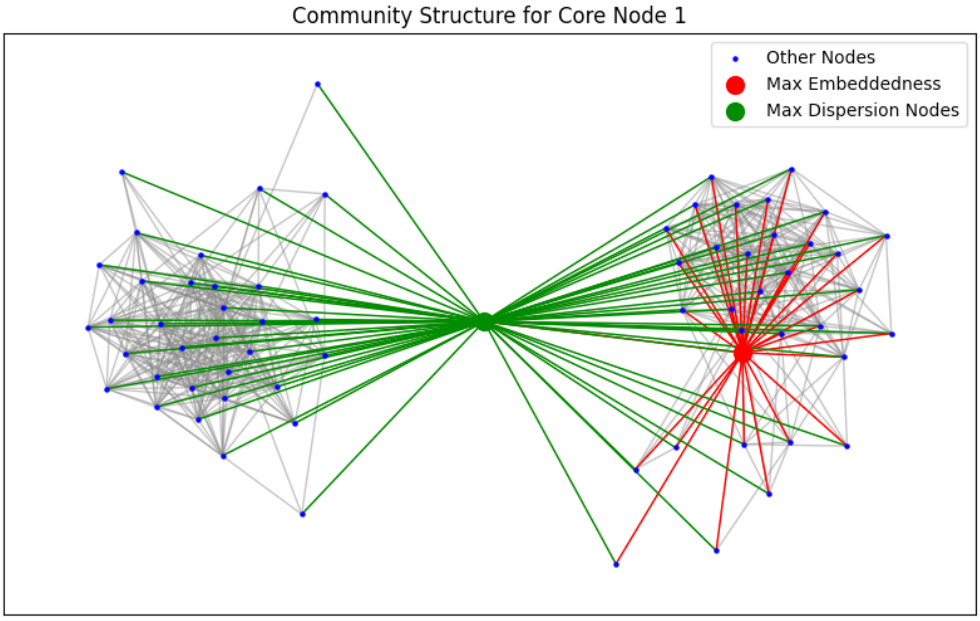


Figure 46

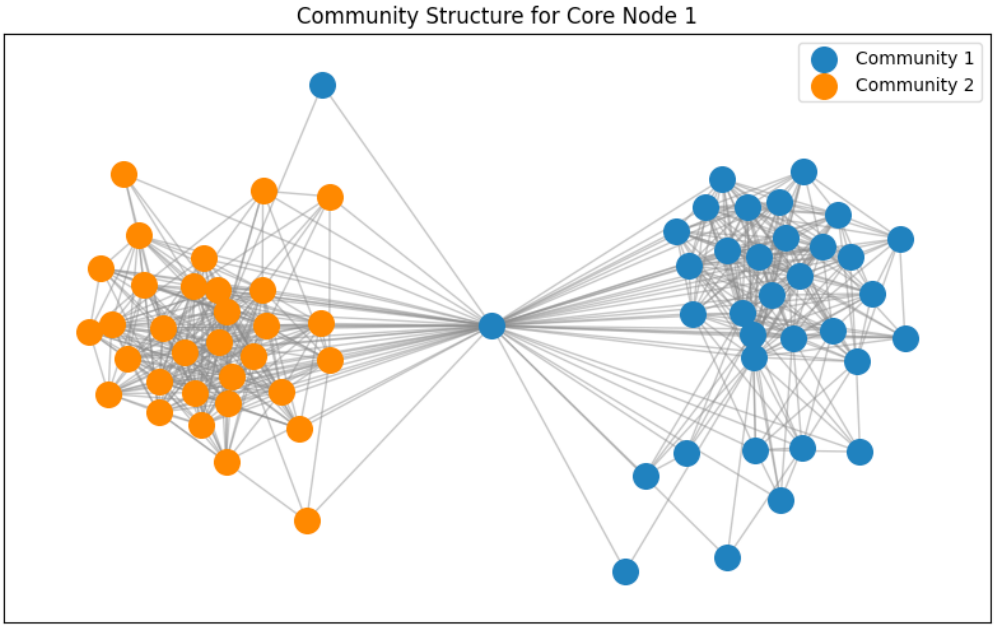


Figure 47

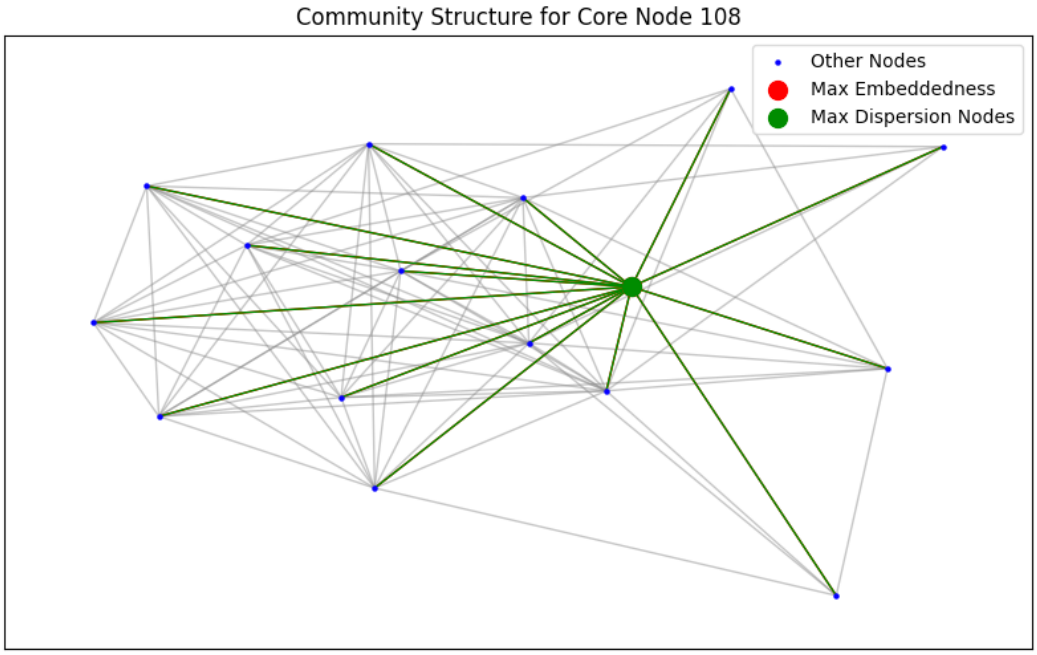


Figure 48

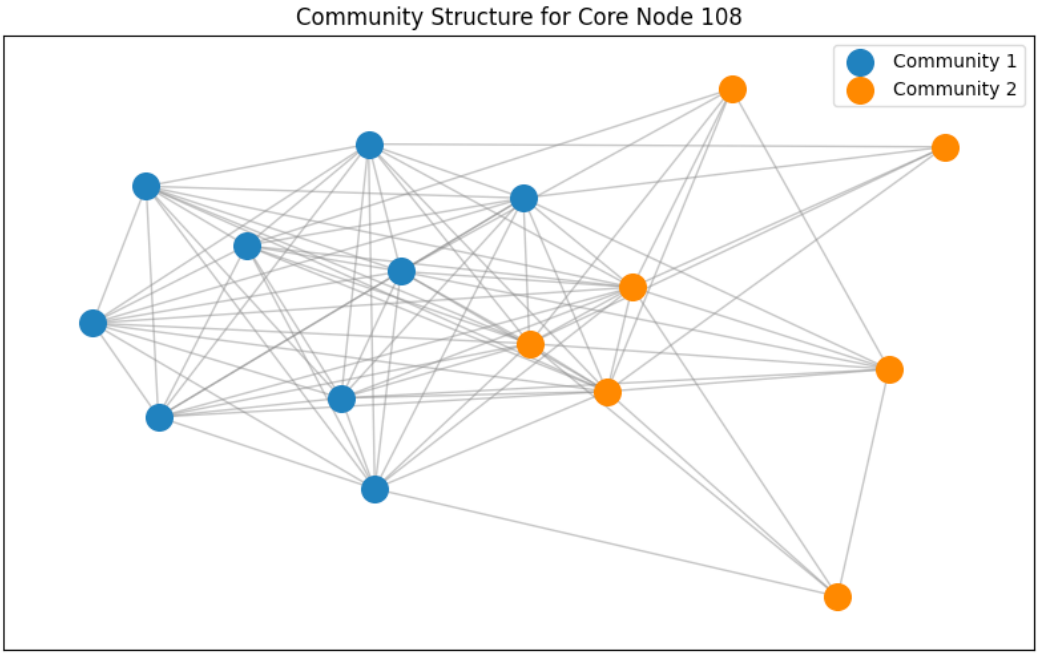


Figure 49

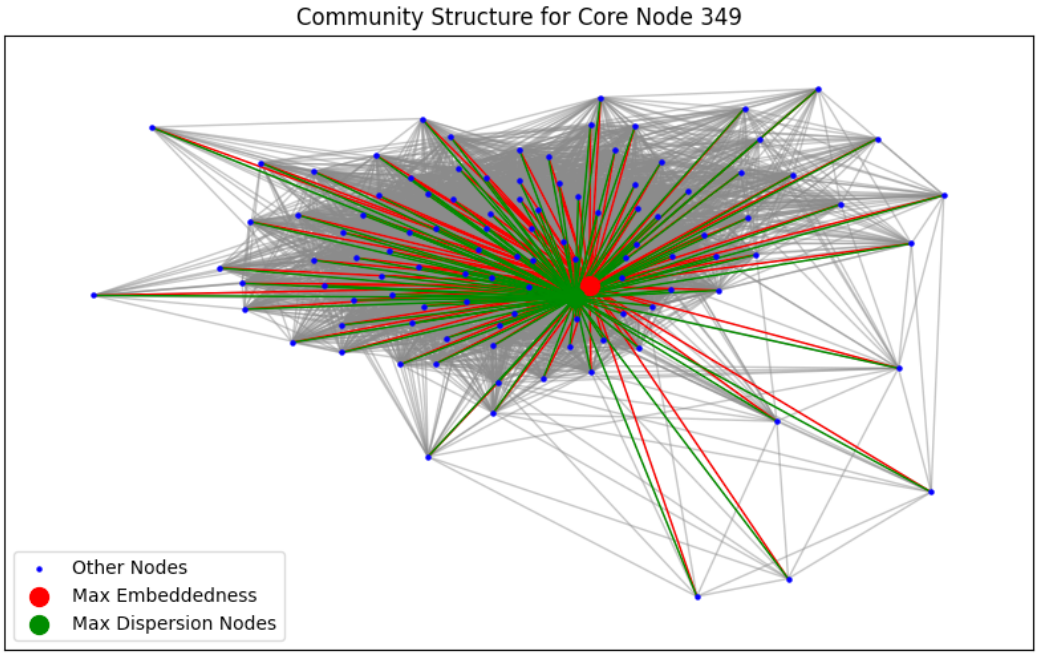


Figure 50

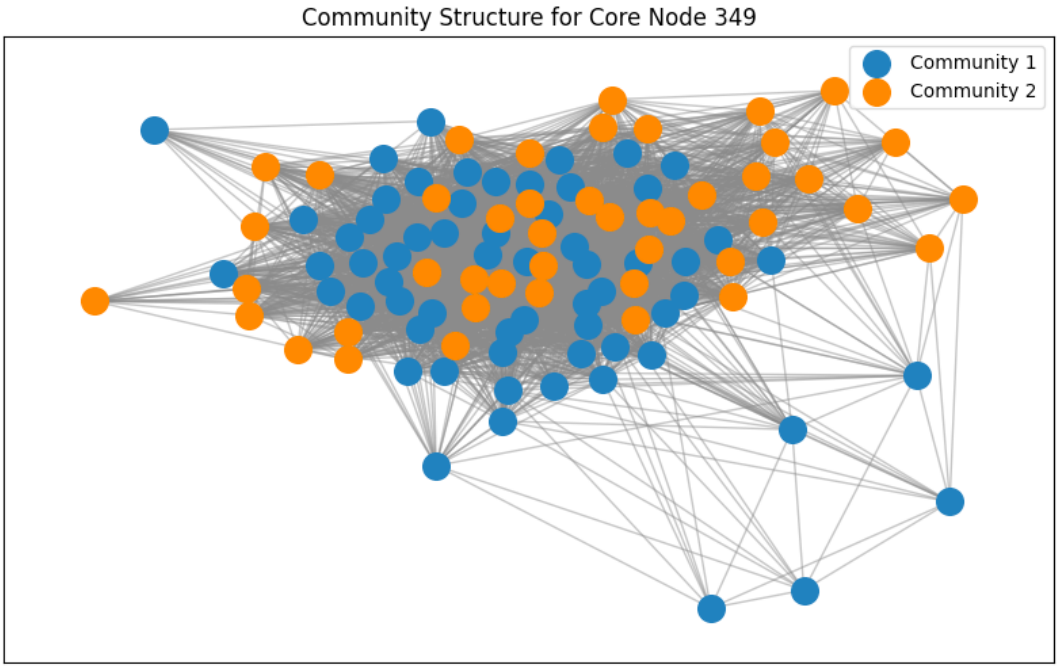


Figure 51

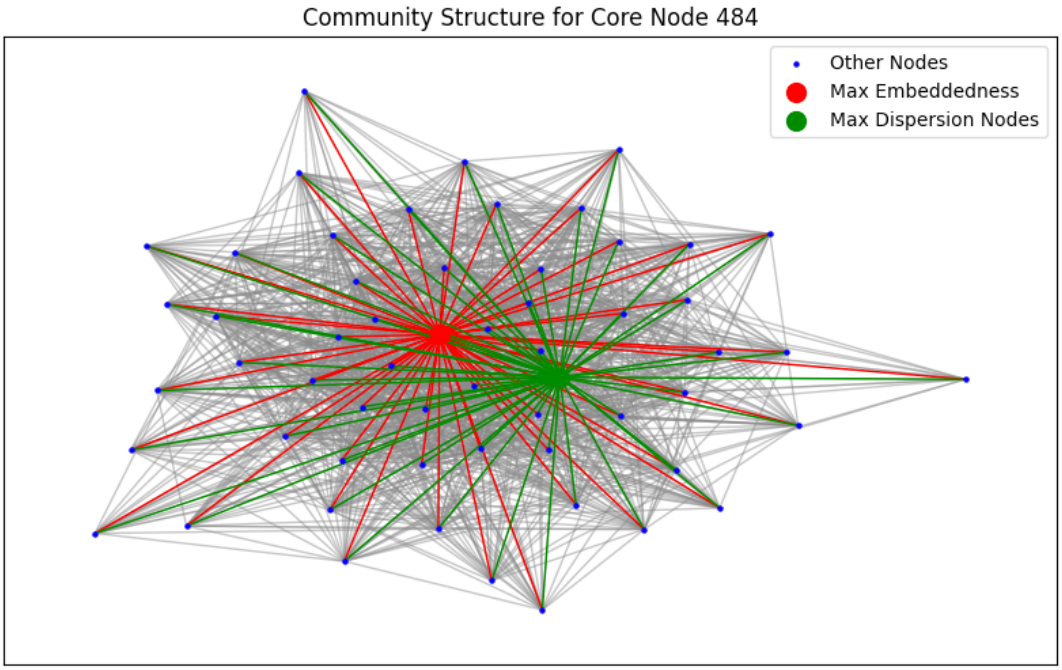


Figure 52

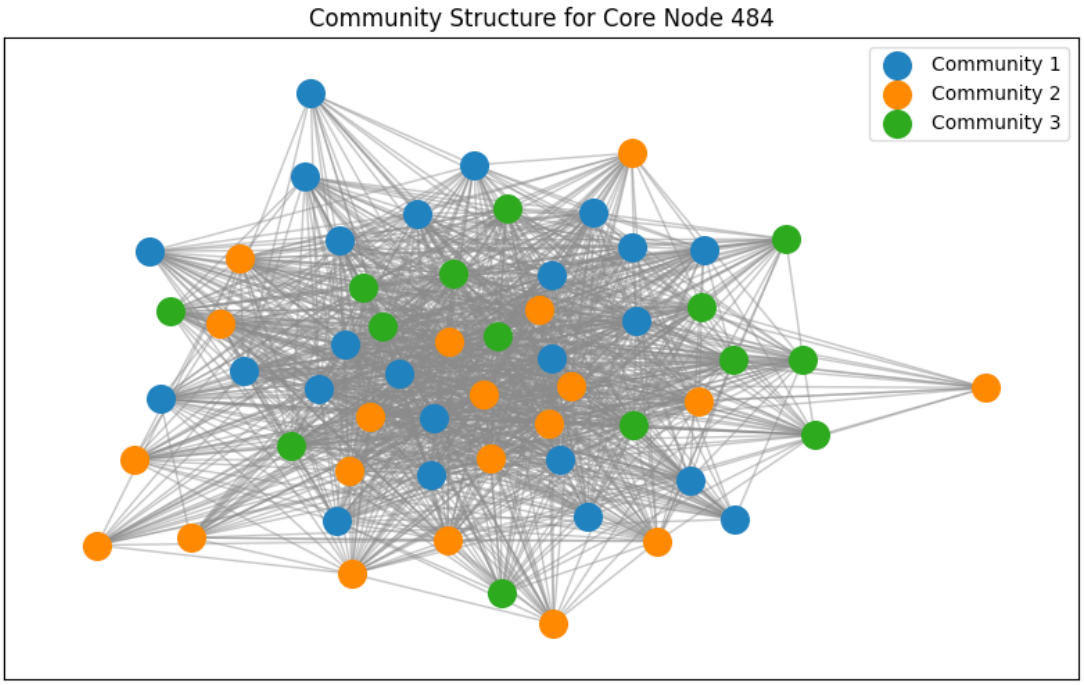


Figure 53

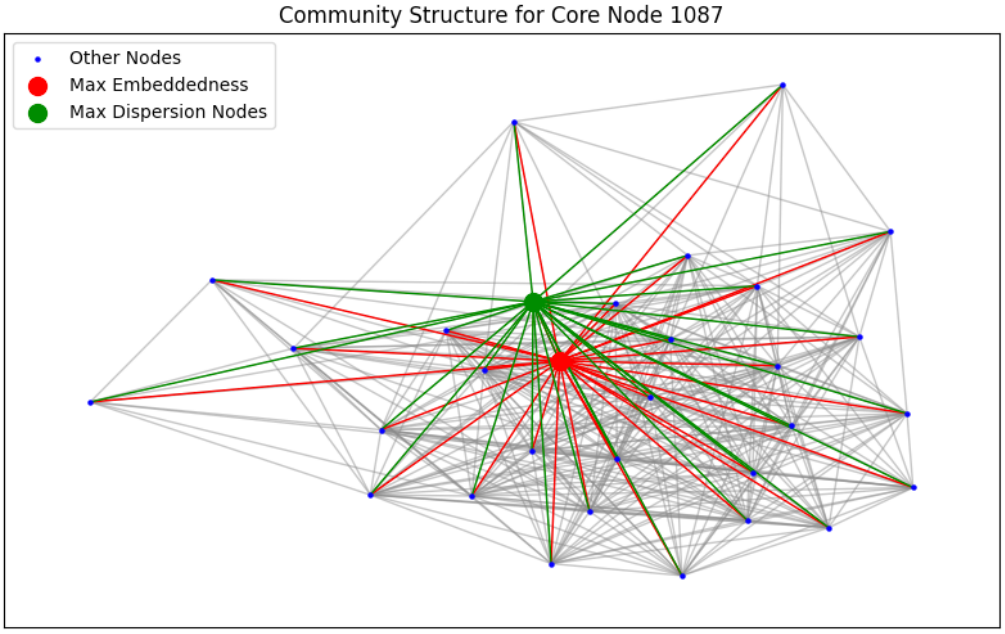


Figure 54

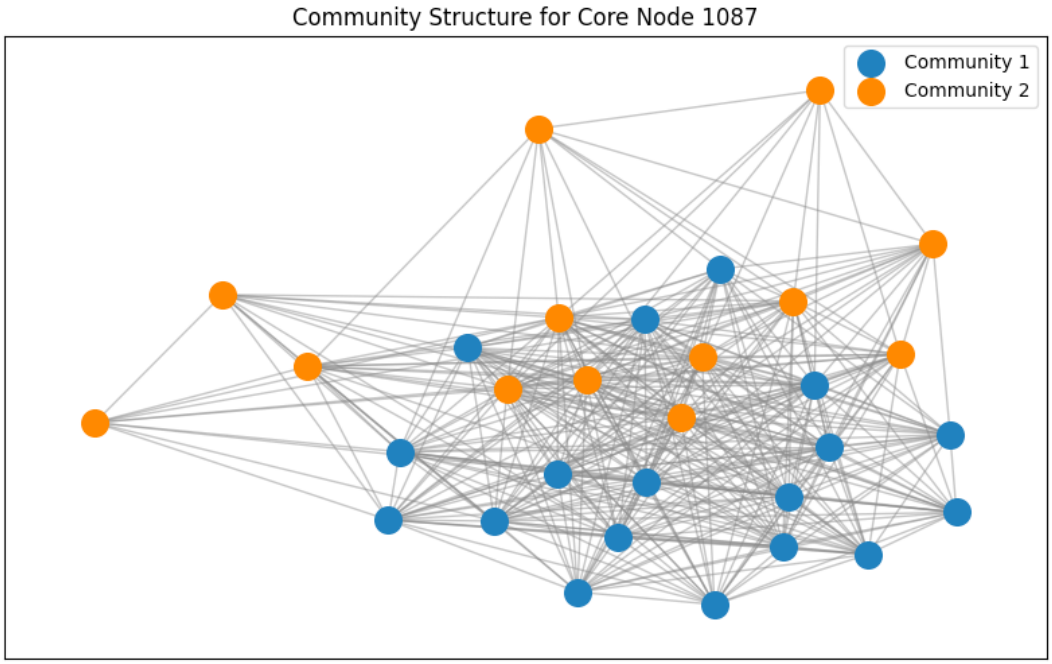

Community Structure for Core Node 1087
Community 1
Community 2

Figure 55

QUESTION 15: Use the plots from Question 13 and 14 to explain the characteristics of a node revealed by each of this measure.

Answer:

I have taken one graph from each of the section and explained in below.

Graph from Question 13 for Core Node 1:

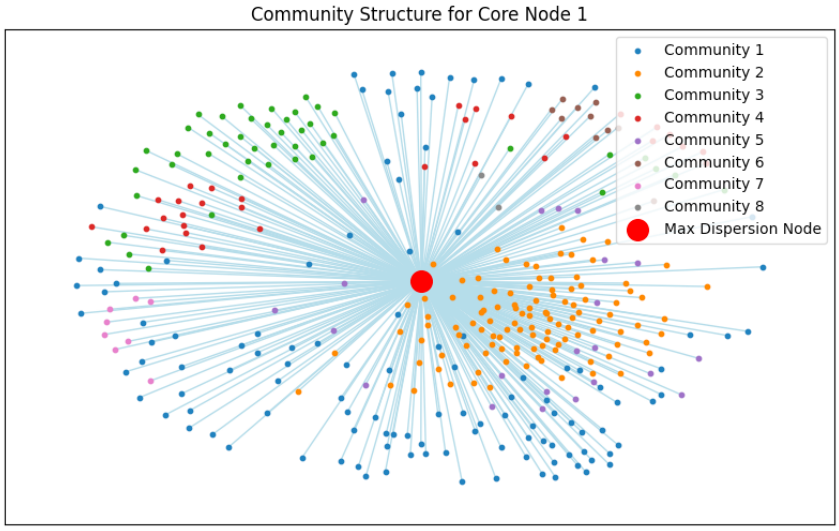


Figure 56

This graph represents the community structure for a core node (labeled as "Max Dispersion Node"). Here are the key points:

1. Node Representation: The central large red node represents the "Max Dispersion Node." Numerous lines radiate outward from this central node, connecting it to smaller nodes.
2. Community Structure: Each smaller node is assigned to a specific community. The legend on the right side of the graph lists communities 1 through 8, each represented by a different color. The colors help distinguish the communities.
3. Interpretation: The graph visualizes how the Max Dispersion Node is connected to its surrounding communities. It highlights the relationships and interconnections between the central node and other nodes within each community.

Additional Note: Community detection algorithms (such as Fastgreedy, Edge Betweenness, and Infomap) are often used to identify these community structures in networks.

Graph from Question 14 for Core Node 1:

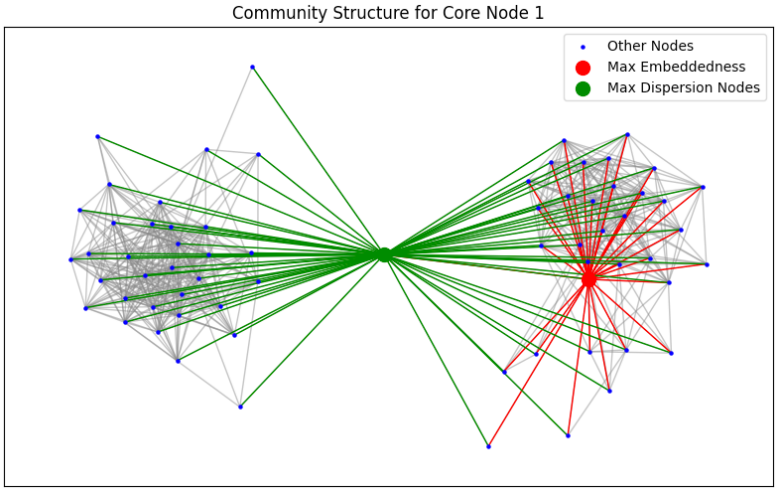


Figure 57

The graph titled “Community Structure for Core Node 1” has been analyzed and provided the details in below:

1. Node Types: The graph consists of nodes (points) connected by lines.There are three types of nodes, each represented by a different color: Blue Nodes (Other Nodes): These nodes are marked in blue.Green Nodes (Max Embeddedness): These nodes have the most connections radiating from them. They play a central role in the network. Red Nodes (Max Dispersion Nodes): These nodes appear to be on the periphery with fewer connections.

2. Community Structure: The green “Max Embeddedness” nodes likely form the core of the community. The red “Max Dispersion Nodes” are less connected and may represent outliers or less influential nodes.

3. Interpretation: The graph visually represents the structure of a community around the core node (Max Dispersion Node). It highlights the importance and roles of different nodes within that community.

Overall, this graph provides insights into the relationships and connectivity patterns within the network.

4. Friend recommendation in personalized networks

In many social networks, it is desirable to predict future links between pairs of nodes in the network. In the context of this Facebook network it is equivalent to recommending friends to users. In this part of the project, we will explore some neighborhood-based measures for friend recommendation. The network that we will be using for this part is the personalized network of node with ID 415.

QUESTION 16: What is $|N_r|$, i.e. the length of the list N_r?

Answer:

Personalized network of Node 414 is:==> Graph with 160 nodes and 1857 edges
The length of the list Nr is 11.

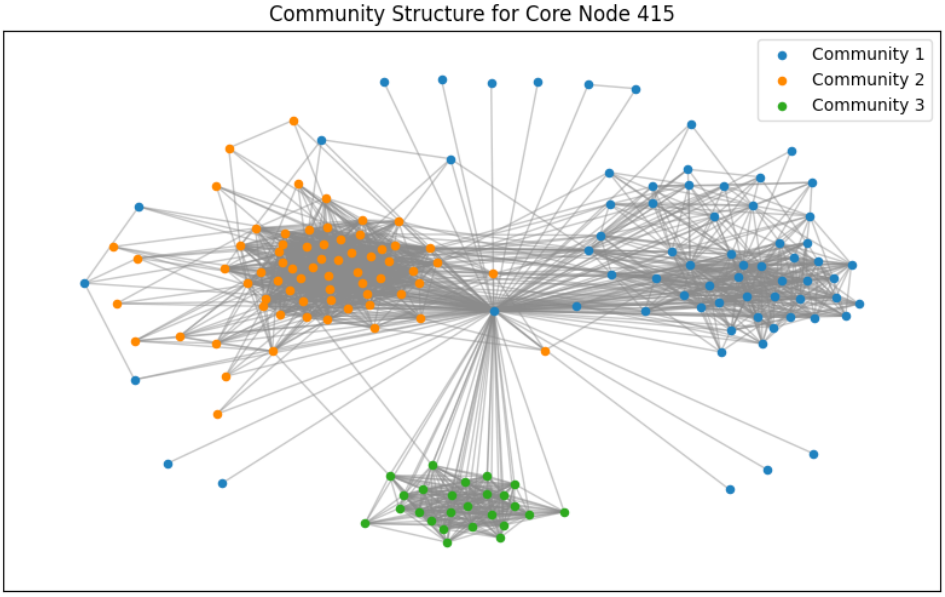


Figure 58

QUESTION 17: Compute the average accuracy of the friend recommendation algorithm that uses:

• Common Neighbors measure • Jaccard measure • Adamic Adar measure

Based on the average accuracy values, which friend recommendation algorithm is the best? Hint Useful function(s): similarity

Answer:

- Average Common Neighbors Accuracy: 54.8091
- Average Jaccard Accuracy: 0.3685
- Average Adamic-Adar Accuracy: 0.5584

2. Google+ network

In this part, we will explore the structure of the Google+ network. The dataset for creating the network can be found in the link below:

http://snap.stanford.edu/data/egonets-Gplus.html

Create directed personal networks for users who have more than 2 circles. The data required to create such personal networks can be found in the file named gplus.tar.gz.

QUESTION 18: How many personal networks are there?

Answer:

Total unique users in personal networks: 66754

QUESTION 19: For the 3 personal networks (node ID given below), plot the in-degree and out-degree distribution of these personal networks. Do the personal networks have a similar in and out degree distribution? In this question, you should have 6 plots.

• 109327480479767108490 • 115625564993990145546 • 101373961279443806744

Answer:

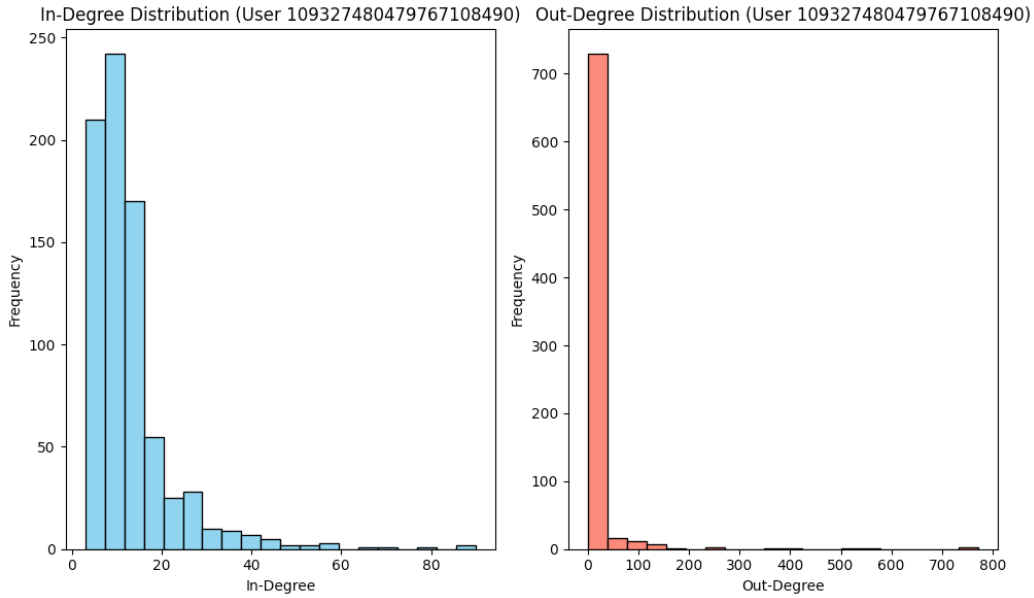


Figure 59

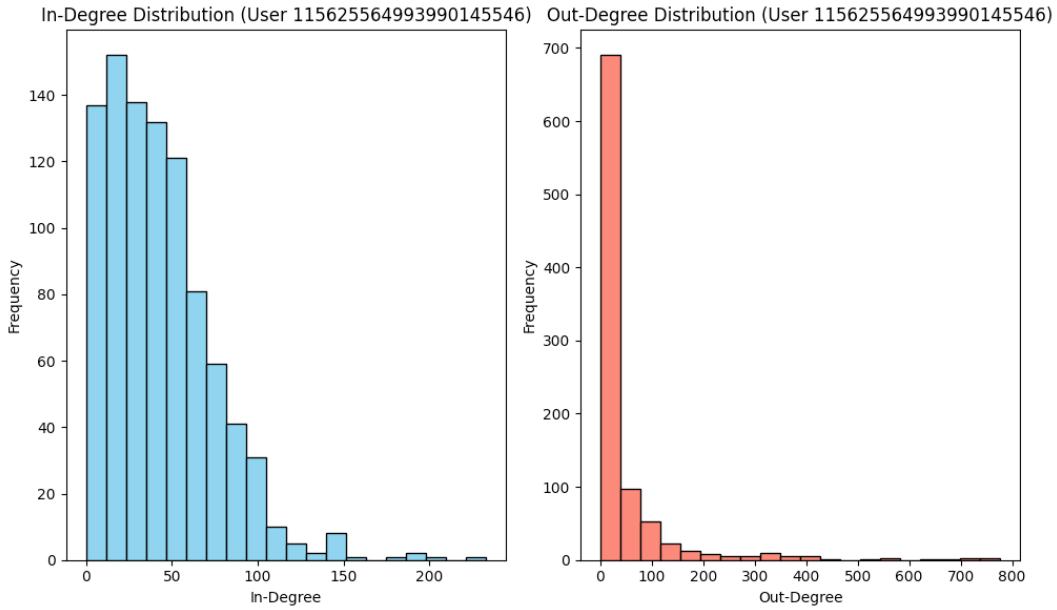


Figure 60

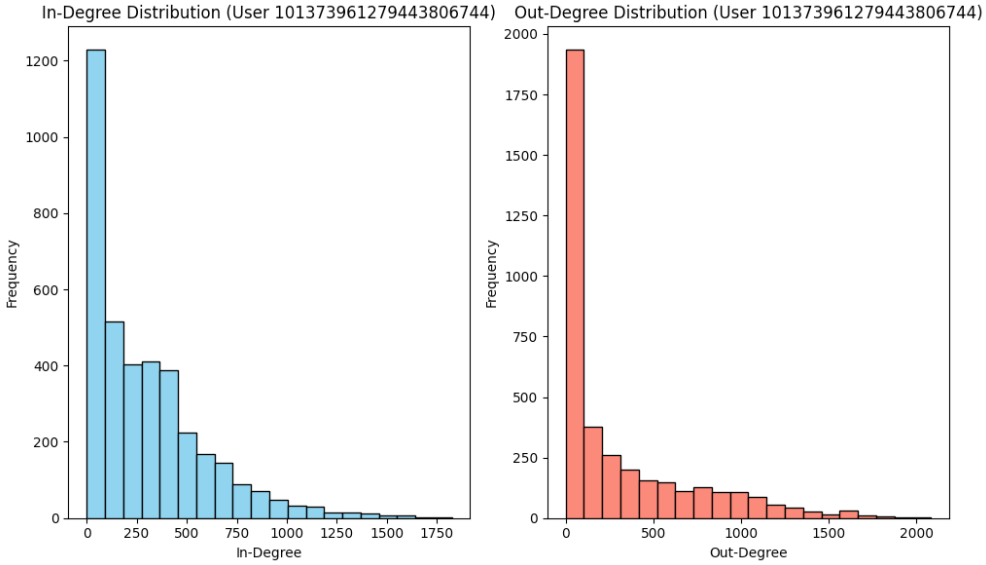


Figure 61

Here is the analysis on the two bar graphs that were generated for user 10932748047976108490.

The left graph represents the “In-Degree Distribution (User 10932748047976108490)” and the right graph represents the “Out-Degree Distribution (User 10932748047976108490).” Here are the key observations:

In-Degree Distribution:

- The bars are mostly tall and concentrated toward the lower end of the scale.
- Most bars have in-degrees under 10, and the frequency rapidly decreases as it moves toward an in-degree of 60.

Out-Degree Distribution:

- There is a single tall bar at an out-degree value close to zero.
- No other bars are present, indicating very few connections going out from nodes (out-degrees).

The in-degree distribution shows a range of values with a gradual decline, whereas the out-degree distribution indicates most values are concentrated at one point near zero with no spread. This suggests that for this particular user's network, there are many connections coming into nodes (in-degrees) with varying frequencies, but very few connections going out from nodes (out-degrees), which are almost non-existent beyond the initial point.

Based on these observations, we can conclude that personal networks do not have a similar in and out degree distribution.

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, extract the community structure of each personal network using Walktrap community detection algorithm. Report the modularity scores and plot the communities using colors. Are the modularity scores similar? In this question, you should have 3 plots.

Answer:

Modularity score for User 109327480479767108490: 0.2577

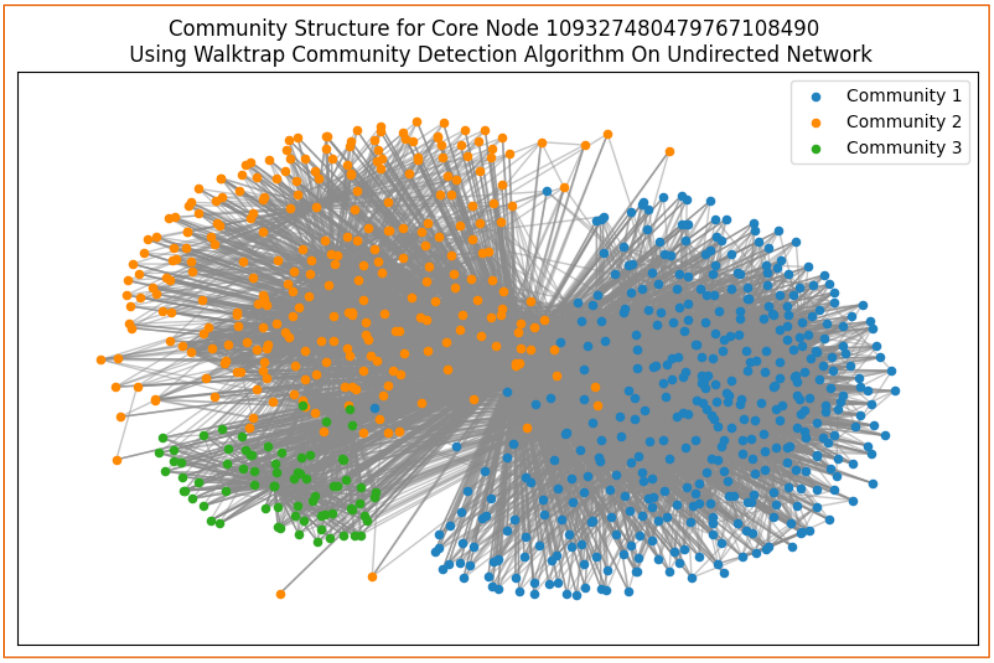


Figure 62

Modularity score for User 115625564993990145546: 0.3087

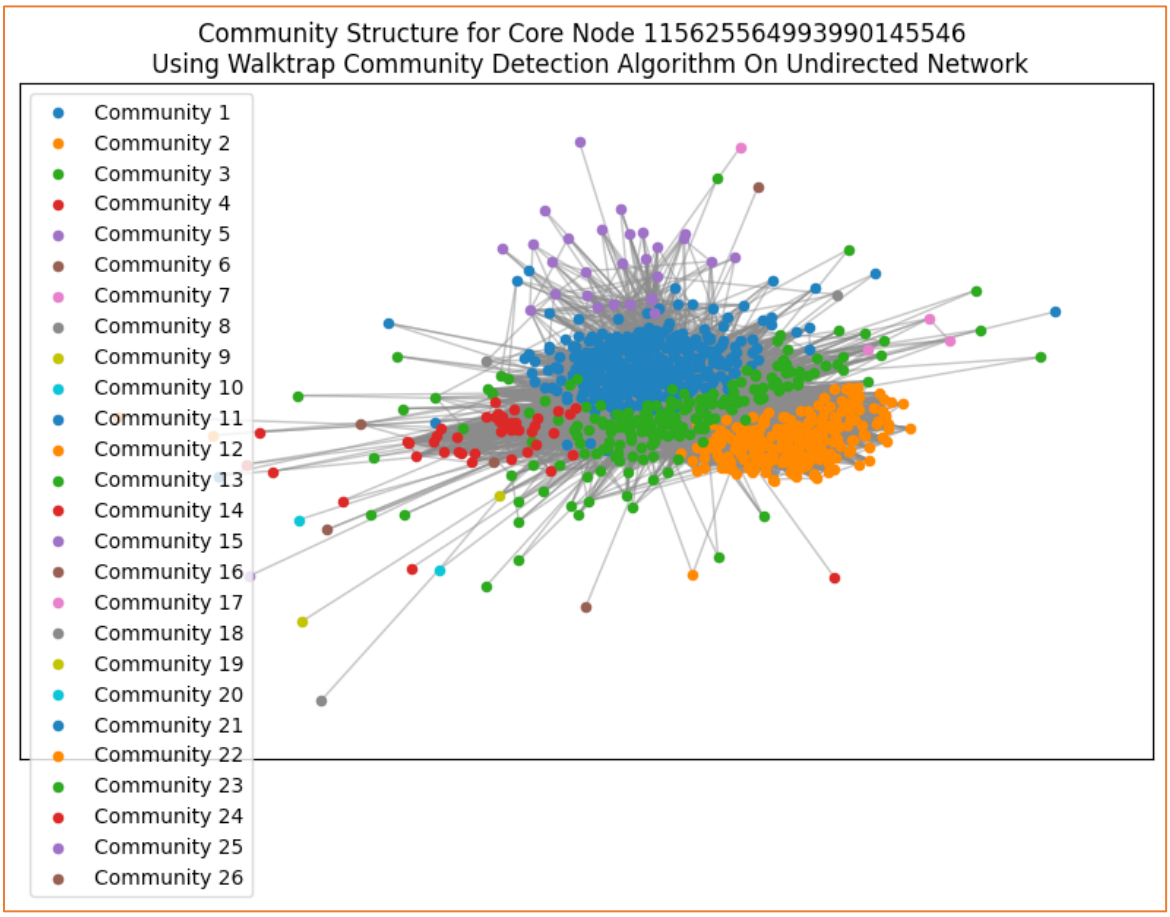


Figure 63

Modularity score for User 101373961279443806744: 0.1734

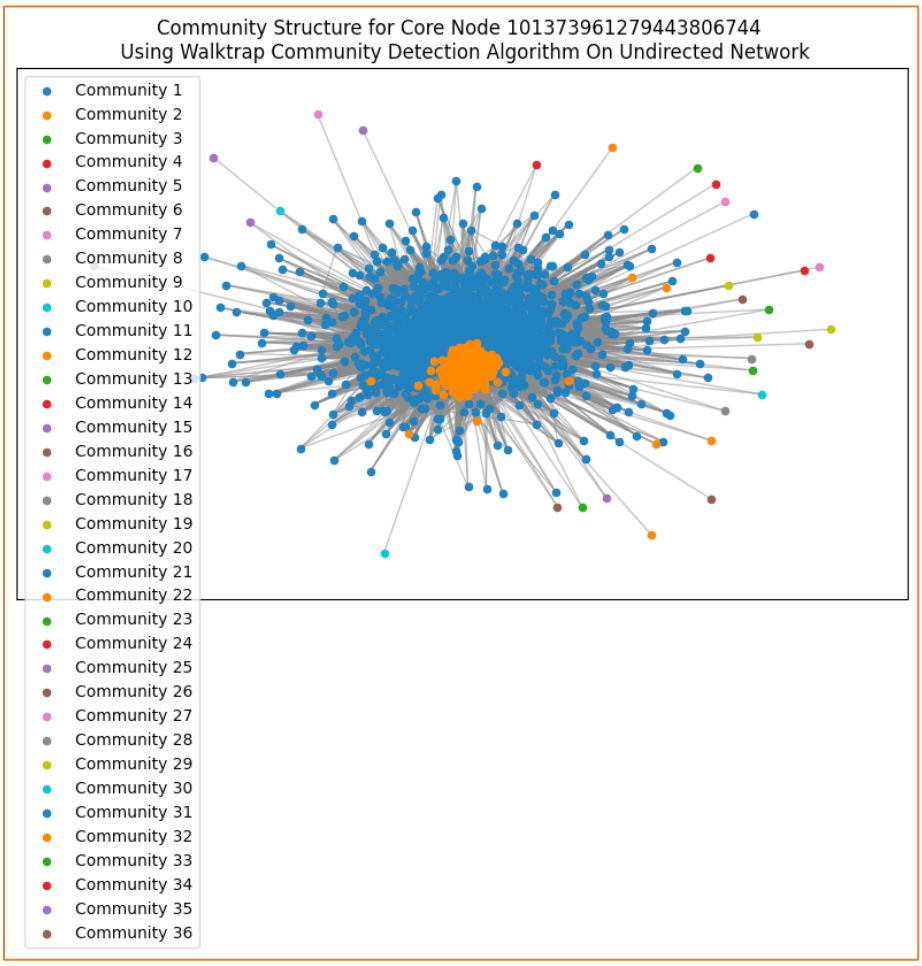


Figure 64

The modularity score is a measure used in community detection algorithms to evaluate the quality of the partitioning of nodes into communities within a network. It quantifies how well the network can be divided into distinct groups based on the connections between nodes.

In our case, the modularity scores are as follows:

1. User 109327480479767108490: 0.2577
2. User 115625564993990145546: 0.3087
3. User 101373961279443806744: 0.1734

Higher modularity scores indicate better community structure. However, whether these scores are considered similar depends on the context and the specific network that we are analyzing. Comparing the scores directly, we can observe that User 115625564993990145546 has the

highest modularity score, suggesting a stronger community structure in their network compared to the other two users.

QUESTION 21: Based on the expression for h and c, explain the meaning of homogeneity and completeness in words.

Answer:

Homogeneity (h):

- **Definition**: Homogeneity measures how well the circles (ground truth) are contained within the communities (predicted clusters).
- **Interpretation**: A high homogeneity score means that each community contains only members of a single circle, indicating that the clustering algorithm has successfully grouped together nodes that belong to the same circle.

Completeness (c):

- **Definition**: Completeness measures how well the members of a circle are assigned to the same community.
- **Interpretation**: A high completeness score means that all members of a circle are assigned to the same community, indicating that the clustering algorithm has successfully captured the entire circle within a single community.

QUESTION 22: Compute the h and c values for the community structures of the 3 personal network (same nodes as Question 19). Interpret the values and provide a detailed explanation. Are there negative values? Why?

Answer:

Homogeneity (h):

- Homogeneity measures how well the communities (detected by the algorithm) match the original circle labels. It indicates how pure the communities are in terms of circle membership.
- The formula for homogeneity is: [h = 1 - \frac{H(C|K)}{H©}] where:
  - (H (c)) represents the entropy of the circles (original ground truth labels).
  - (H(C|K)) represents the conditional entropy of the circles given the communities.
- A higher homogeneity value (closer to 1) indicates that the communities align well with the original circles.

Completeness (c):

- Completeness measures how well the original circle labels match the detected communities. It indicates how well the communities cover the circle memberships.
- The formula for completeness is: [c = 1 - \frac{H(K|C)}{H(K)}] where:
  - (H(K)) represents the entropy of the communities.
  - (H(K|C)) represents the conditional entropy of the communities given the circles.
- A higher completeness value (closer to 1) indicates that the communities cover the original circles well.

The calculated values from the given dataset are provided in below table.

User Id	Completeness	Homogeneity
109327480479767108490	-6.6252	-4.3531
115625564993990145546	-0.1539	-3.7453
101373961279443806744	-0.4331	-1.0068

A detailed explanation the values that are provided in above table, given in below.

- **User 109327480479767108490**:
  - Completeness: -6.6252
  - Homogeneity: -4.3531
  - Explanation:
    - The negative values suggest that the communities detected by the algorithm do not align well with the original circle labels.
    - The communities are highly fragmented and do not capture the circle memberships effectively.

- **User 115625564993990145546**:
  - Completeness: -0.1539
  - Homogeneity: -3.7453
  - Explanation:
    - The negative homogeneity value indicates that the communities are not pure; they mix different circle memberships.
    - The low completeness value suggests that the communities do not fully cover the original circles.

- **User 101373961279443806744**:
  - Completeness: -0.4331
  - Homogeneity: -1.0068
  - Explanation:
    - Similar to the previous cases, the negative values indicate poor alignment between communities and circles.
    - The communities are not representative of the original circle memberships.

Negative values for both homogeneity and completeness indicate that the community detection algorithm did not perform well in capturing the ground truth circle memberships. It's essential to consider these limitations when interpreting the results. Factors such as noise, overlapping communities, and algorithmic biases can contribute to these discrepancies.

3. Cora dataset

One of the well-known categories of machine learning problems is "supervised learning". To solve this problem for Cora dataset, we pursue three parallel ideas. Implement each idea and compare.

QUESTION 23: Idea 1
Use Graph Convolutional Networks [1]. What hyperparameters do you choose to get the optimal performance? How many layers did you choose?

Answer:

Observation on dataset

cora.content file:

- The **first** column indicates the **node name.**
- The **second** until the last second columns indicate the **node features.**
- The **last** column indicates the **label of that node/document.**

cora.cites file:

- Each line indicates the tuple of connected **nodes/documents**.

Dataset Statistics:

Number of nodes (documents): 2708
Number of edges (citations): 5278
Number of classes: 7

Number of features (F) of each node: 1433

Class Counts:

Neural_Networks	818
Probabilistic_Methods	426
Genetic_Algorithms	418
Theory	351
Case_Based	298
Reinforcement_Learning	217
Rule_Learning	180

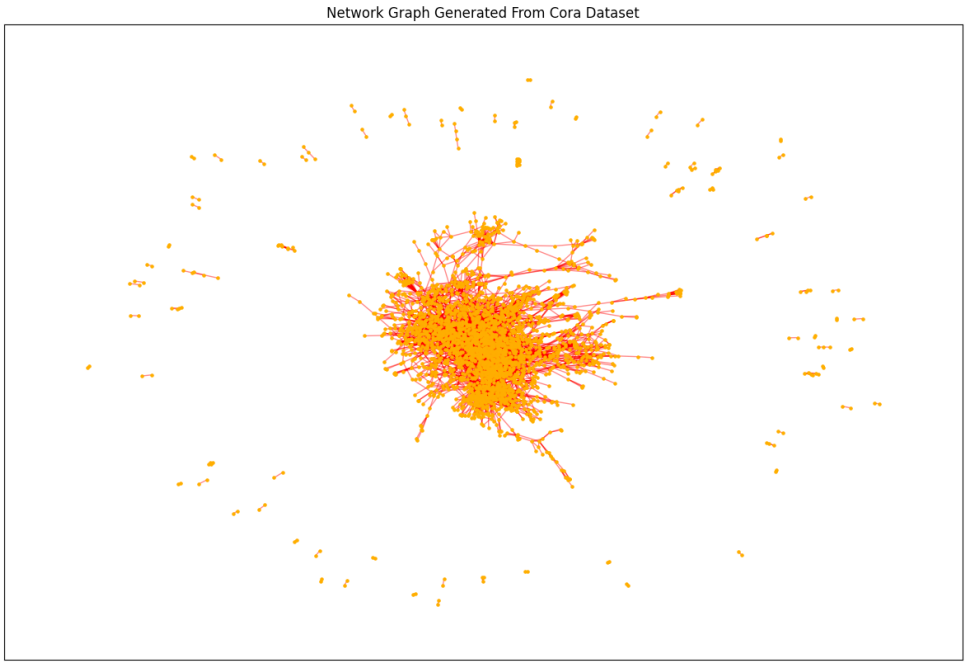


Figure 65

QUESTION 24: Idea 2

Extract structure-based node features using Node2Vec [2]. Briefly describe how Node2Vec finds node features. Choose your desired classifier (one of SVM, Neural Network, or Random Forest) and classify the documents using only Node2Vec (graph structure) features. Now classify the documents using only the 1433-dimensional text features. Which one outperforms? Why do you think this is the case? Combine the Node2Vec and text features and train your classifier on the combined features. What is the best classification accuracy you get (in terms of the percentage of test documents correctly classified)?

Answer:

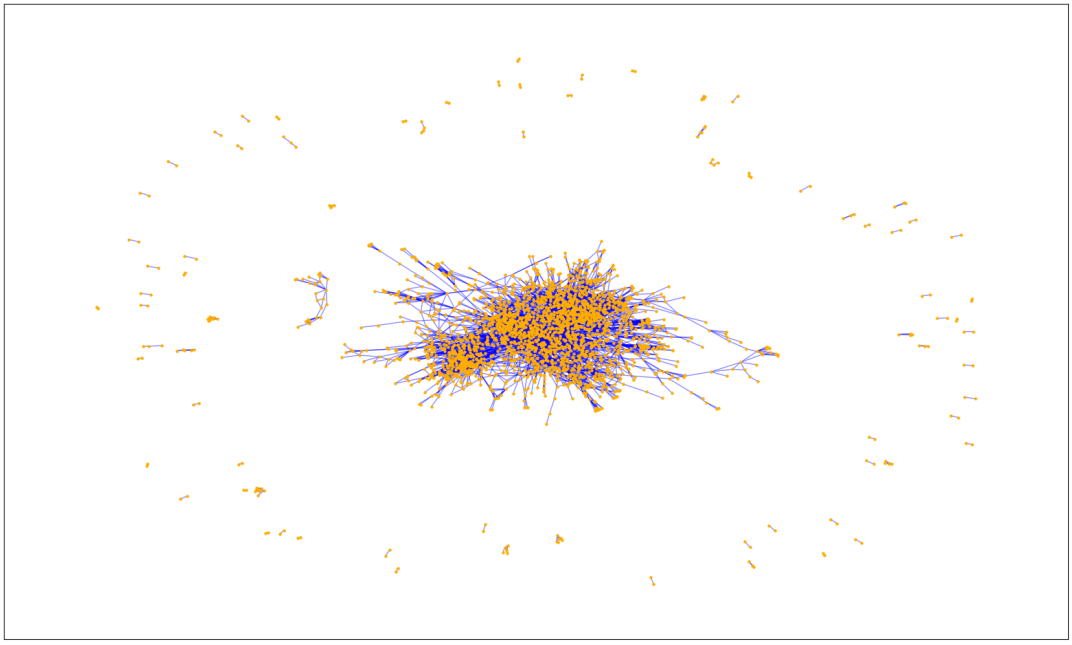


Figure 66

Results and Analysis

- **Node2Vec Features Accuracy**: The accuracy using Node2Vec features alone.
- **Text Features Accuracy**: The accuracy using text features alone.
- **Combined Features Accuracy**: The accuracy using both Node2Vec and text features.

Results:

- Accuracy using Node2Vec features: 0.8598
- Accuracy using text features: 0.2743
- Accuracy using combined features: 0.8413

Node2Vec Features:

- Node2Vec is an algorithm for learning node embeddings in a graph.
- It generates feature vectors for nodes based on their structural context (neighborhood) in the graph.
- The algorithm uses random walks to explore the graph and capture both local and global information.
- These learned features can be used for downstream tasks like classification.
- In our case, the accuracy using Node2Vec features alone is 0.8598.

Text Features:

- Text features refer to the 1433-dimensional features extracted from the document text.
- These features could be TF-IDF vectors, word embeddings, or any other representation.
- The accuracy using text features alone is 0.2743.
- This low accuracy suggests that the text features alone are not sufficient for effective classification.

Combined Features:

- Combining Node2Vec and text features leverages both graph structure and textual content.
- By doing so, we capture complementary information.
- The accuracy using combined features is 0.8413.
- Although slightly lower than Node2Vec alone, it's still a good compromise between the two.

Why Node2Vec Outperforms Text Features:

- Node2Vec captures the graph topology, which is crucial for community detection and classification.
- Graph structure encodes relationships between nodes, such as community memberships and influence.
- Text features alone lack this structural context, making them less effective for community detection.
- However, combining both features allows us to benefit from both worlds.

Best Classification Accuracy:

- The combined features achieve the highest accuracy (0.8413).
- This indicates that integrating graph structure and text features improves classification performance.

QUESTION 25: Idea 3

We can find the personalized PageRank of each document in seven different runs, one per class. In each run, select one of the classes and take the 20 seed documents of that class. Then, perform a random walk with the following customized properties: (a) teleportation takes the random walker to one of the seed documents of that class (with a uniform probability of 1/20 per seed document). Vary the teleportation probability in {0, 0.1, 0.2}. (b) the probability of transitioning to neighbors is not uniform among the neighbors. Rather, it is proportional to the cosine similarity between the text features of the current node and the next neighboring node. Particularly, assume we are currently visiting a document x_0 which has neighbors x_1,x_2,x_3.

Then the probability of transitioning to each neighbor is:

$$p_i = \frac{\exp(x_0 \cdot x_i)}{\exp(x_0 \cdot x_1) + \exp(x_0 \cdot x_2) + \exp(x_0 \cdot x_3)}; \text{ for } i = 1, 2, 3. \quad (7)$$

Repeat part b for every teleportation probability in part a.

Run the PageRank only on the GCC. for each seed node, do 1000 random walks. Maintain a class-wise visited frequency count for every unlabeled node. The predicted class for that unlabeled node is the class which lead to maximum visits to that node. Report accuracy and f1 scores.

For example if node 'n' was visited by 7 random walks from class A, 6 random walks from class B... 1 random walk from class G, then the predicted label of node of 'n' is class A.

Answer: