EE 232E
Graphs and Network Flows
Homework 3
Winter 2016

Liqiang Yu, Rongjing Bai, Yunwen Zhu
904592975, 204587519, 104593417

05-01-2016

Contents

1 Problem 3

When transforming the undirected network into directed network, it's not trivial to choose the method. Here we have two options : (1) we can keep the number of edges unchanged and just remove the direction, however it will lead to the non-simple network. (2) or we can merge the two directed edges between two nodes and make the new weight the geometric mean of the original weight. The fast greedy method can only be applied to the second method and label propagation method can be applied to both.

1.1 Option 1

When using label propagation method to calculate the community structure for option 1, we got 5 communities, the sizes of each community are shown in table 1. From the table we can see that there is a dominant community with the size 10472, which means it cannot separate reasonable communities from the original giant connected component. The structure graph is shown in figure 1 and the modularity is 0.0001494257.

Table 1: The community sizes

Index	1	2	3	4	5
Size	10472	4	5	3	3

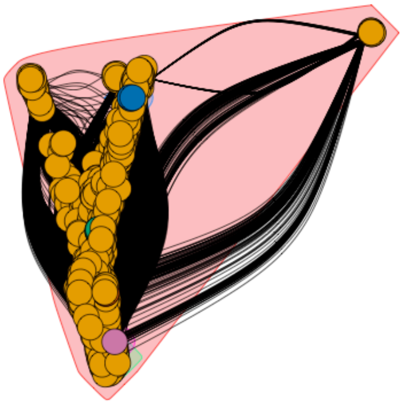


Figure 1: The structure of the community derived from option 1 with label propagation method

1.2 Option 2

We can use both label propagation method and fast greedy method to calculate the community for option 2. The respective community sizes are shown in table 2 and table 3. The community structures are shown in figure 2 and figure 3. The modularity for label propagation method is 0.0001698002 and is 0.3287988 for fast greedy method. We can see that compared with label propagation method, fast greedy method will produce communities with more uniform sizes and higher modularity, which means the fast greedy method is more effective in this case.

Table 2: The community sizes for label propagation method

Index	1	2	3	4	5	6
Size	10469	4	5	3	3	3

Table 3: The community sizes for fast greedy method

Index	1	2	3	4	5	6	7	8
Size	1856	1666	1022	2266	731	1236	633	1077

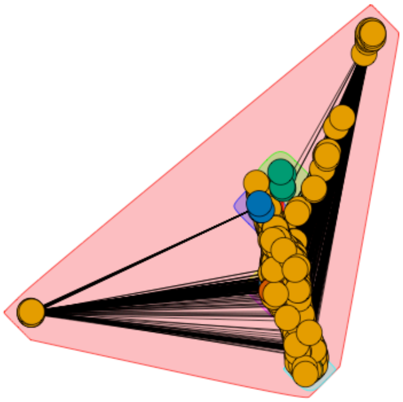


Figure 2: The structure of the community derived from option 2 with label propagation method

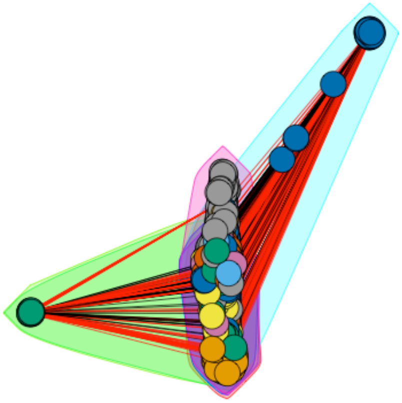


Figure 3: The structure of the community derived from option 2 with fast greedy method

2 Problem 4

We have calculated the communities with the fast greedy method in problem 3 and find that community 4 has the largest size of 2266. By deleting all the other nodes that don't belong to cummunity 4, we have another network. Then fast greedy method was applied on this network to calculate its structure. The sizes of community are shown in table 4 and the structure graph is shown in figure 4. The modularity is 0.3595153.

Table 4: The community sizes for fast greedy method

Index	1	2	3	4	5	6	7	8
Size	306	457	313	365	426	347	47	5

3 Problem 5

By deleting all the nodes which belong to the communities smaller than 100, we can calculate the structures of all sub communities which are larger than 100. From table 4 we can see that there are six qualified communities and their modularities are shown in table 5. The structure graphs of all sub communities are shown in figure 5.

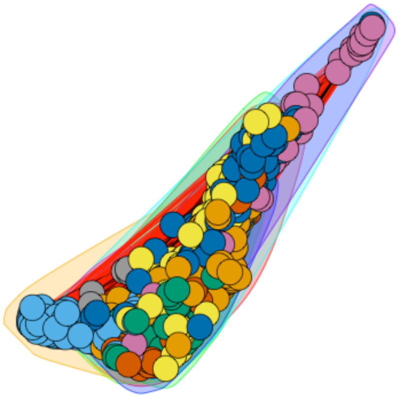


Figure 4: The structure of the largest sub community

Table 5: The modularity of all sub communities larger than 100

Index	1	2	3	4	5	6
Modularity	0.372	0.375	0.433	0.490	0.474	0.314

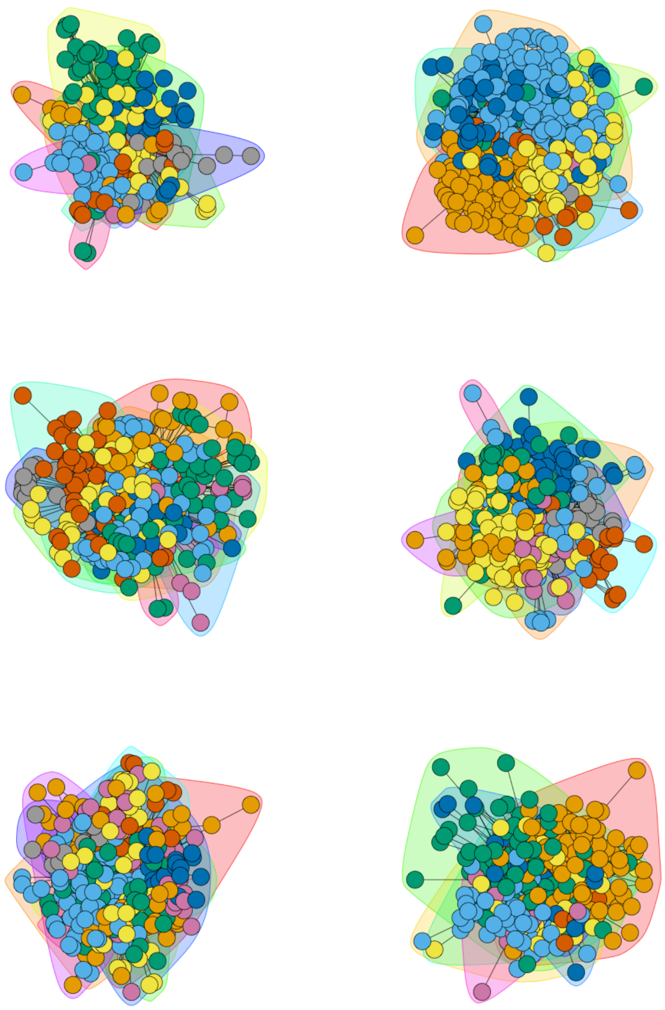


Figure 5: The structure graphs of all sub communities with the size larger than 100

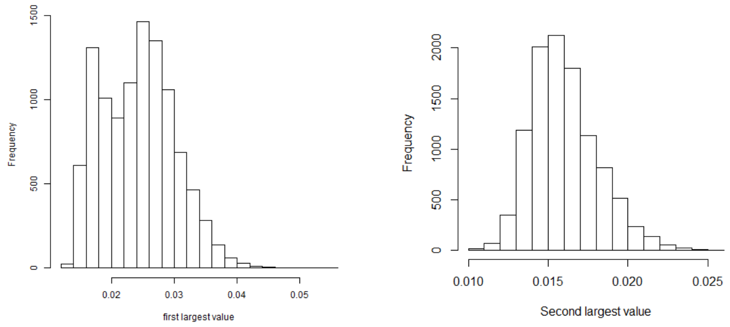
4 Problem 6

In order to study the strucutures of overlapped communities, we can use personalized pagerank. Every time we start a random walk from node i with the damping parameter 0.85 in the original directed network, we can get the visit probabilities reflecting the relation between the other nodes and node i. Therefore we can compute

$$\vec{M}_i = \sum_j v_j \vec{m}_j$$

where v_j is the visit probability of node j and $\vec{m}_j$ is its community membership computed in Problem 3. In order to avoid the large amount of computation, we choose the largest 30 v_j and replace values in $\vec{M}_i$ less than a specific threshold to zero.

It is tricky to find a proper threshold. Here we plot the distribution of the largest values and the second largest values of $\vec{M}_i$ in figure 6. From the histograms we can see that if we choose a small threshold there will be too many nodes belonging to multiple communities, however if we choose a large threshold there will be some nodes belonging to no community. So there is a tradeoff. Since the problem asked us to detect at least three nodes, we choose a relative large threshold 0.024 and hince found 11 nodes belonging to multiple communities, their indices are 375, 385, 389, 3629, 5808, 6631, 6634, 6635, 6650, 9608, 9942.



(a) The distribution of the largest values

(b) The distribution of the second largest values

Figure 6