Assignment 6 for Large Scale Data Mining

Name: Zijian Zhang
Matrikelnr.: 3184680

June 7, 2016

1

1.1

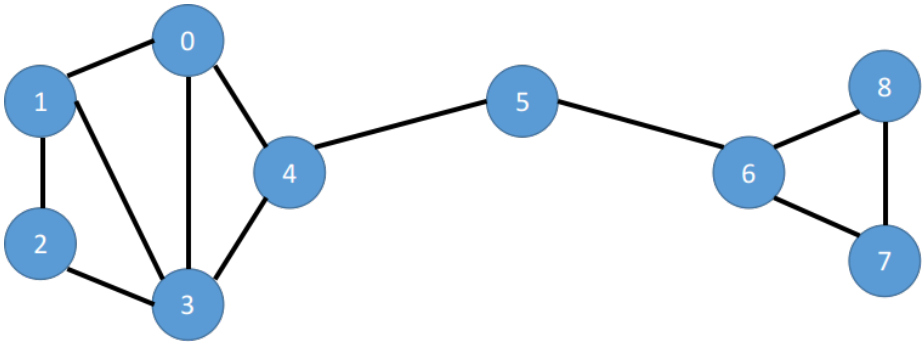
Degree Centrality = 2
Closeness Centrality = $\frac{1}{11}$
Betweenness Centrality = $\frac{5}{21}$

1.2

Edge Btweenness Centrality = $\frac{3}{7}$

1.3

As shown in Figure. The vertex with highest degree centrality is 3
with highest closeness centrality is 4
with highest betweenness centrality is 5



2

2.1

The algorithm could be constructed as such:

Input: Adjacency matrix A of the Graph G

Map: <Key, Value>= <Vertex index i, i-th line of A>

In the Map phase the degree of each vertex is summed up together.

Reduce: <Key, Value>= <Vertex index i, degree of A>

In Reduce phase is basically nothing to be done.

2.2

For the graph in the lecture, the flow runs level-wise from bottom to top. We could at first label those levels from bottom to top, from 0 to 4. So the Map and Reduce phase could be constructed iteratively over the number of levels. The Map and Reduce phase for round i could be constructed as:

Map: <K,V>= <Index of vertexes i within level l, Connection condition of vertex i (i-th row of adjacency matrix)>

Inside the Map phase, based on the flow pushed from level l-1 the new flow is generated by adding 1 and pushed to vertexes in "higher" level, which is defined as those vertexes that connected with i but are not yet been traversed.

Reduce: <K,V>= <Index of vertexes j within level l+1, Flows pushed by the vertexes in level l>

Here are the flows pushed from level l are collected.

2.3

If the graph is a tree, the closeness centrality of the root can be calculated by simply multiply the number of all the descendants of pairs selected from its sons. And the number of descendants of a vertex could be summed from bottom using the tree-like dynamic programming technique.

3

3.1

The main change is that we have to maintain two 'version' of labels. One of them is to record the shortest distance from current point to each point in the table, while the other version represents the opposite direction, from each other vertex to current vertex.

3.2

As to the query we have to consider about two kinds of situation. One from source point to the destination point via some vertex labeled, where the path

from the source vertex to the relay vertex could be found in first 'version' mentioned in the sub-question above, while the second part of path is in the contradict 'version'.

4

4.1

The labels for each vertex over all iteration are:

		round						
		1	2	3	4	5	6	7
vertex	1	0	100	2	1	1	0	100
	2	2	0	100	1	0	0	0
	3	2	2	0	100	3	0	100
	4	1	1	1	0	100	0	0
	5	1	0	100	2	0	100	0
	6	2	0	0	100	1	0	0
	7	1	100	1	100	2	0	0

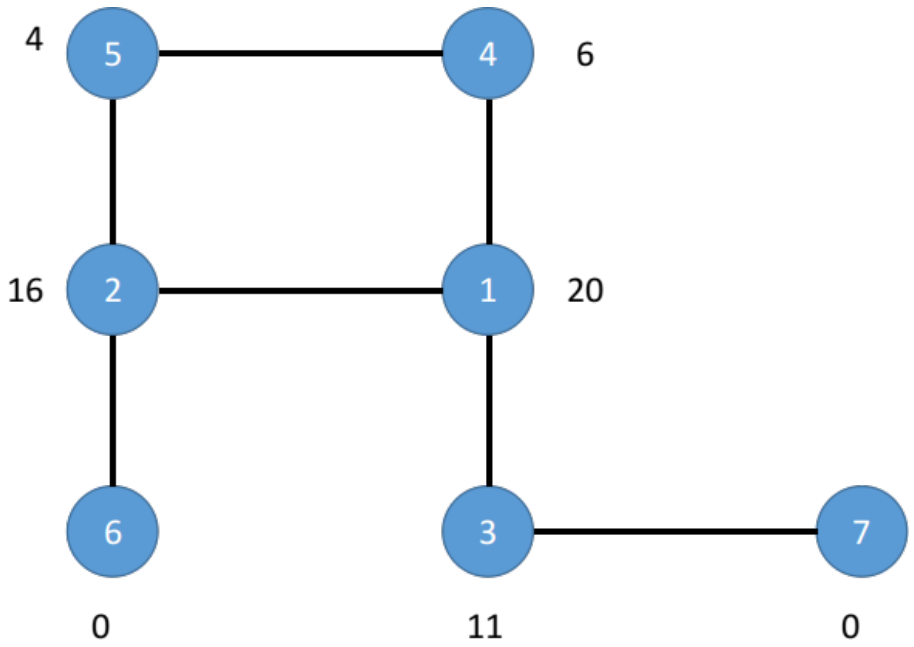
where label number of 100 means that this vertex is pruned.

4.2

The labels are now:

		round						
		1	2	3	4	5	6	7
vertex	1	0	100	4	3	1	0	0
	2	4	0	100	1	0	100	0
	3	4	2	0	100	3	2	1
	4	3	1	1	0	100	1	100
	5	1	3	3	2	0	100	0
	6	2	2	2	1	1	0	100
	7	3	3	1	100	2	1	0

which is much larger than before.
The optimal assignment of the number of vertexes should be firstly of reversed order of betweenness centrality and secondly of normal order of closeness centrality, which means the new arrangement should be illustrated as such:



4.3

Query $d(1,3)$ has the response of:
for first scenario: $d(1,3) = 2$
for second scenario: $d(1,3) = 4$

5

5.1

5.1.1

(a)Fractional flows begin with 3:
(4 , 5): 1
(6 , 5): 1
(7 , 8): 1
(9 , 8): 1
(5 , 2): 3
(8 , 1): 3
(1 , 3): 4
(2 , 3): 4

5.1.2

(b)Fractional flows begin with 4:

(7 , 6): $\frac{1}{2}$
(7 , 8): $\frac{1}{2}$
(9 , 8): 1
(4 , 5): 1
(6 , 5): $\frac{3}{2}$
(8 , 1): $\frac{3}{2}$
(1 , 2): $\frac{7}{2}$
(3 , 2): 1
(5 , 2): $\frac{7}{2}$

5.2

Because of the symmetry of this graph, points 3,4,9 are equivalent considering the betweenness centrality of edges begin with them, meanwhile, points 1,2,5,6,7 and 8 are equivalent.
Also according to symmetry, there are in total 3 types of edges considering betweenness centrality:

For example, at the BFS beginning with 5, the edge (2,3) is as (4,5) to the BFS beginning with 2, which means that the edge(2,3) has flow of 1.

Edge (1,3), (2,3), (4,5), (4,6), (8,9) and (7,9) have the BS = $\frac{1}{9}$
Edge (2,1), (5,6) and (7,8) have the BS = $\frac{11}{72}$
Edge (2,5), (6,7) and (1,8) have the BS = $\frac{19}{72}$

5.3

We could set the threshold as some $\frac{11}{72}<\upsilon<\frac{19}{72}$ and thus divide the whole graph into 3 community, which looks as the three triangles within the graph, thus (1,2,3), (4,5,6) and (7,8,9).