Mohamed Aturban (maturban)
CS595 — FAll 2013 **Assignment 6** November 1, 2013

All files mentioned in this document should be uploaded into the *github* repository.

Problem 1

R script, *rcode.r*, has been written to work on Karate Club graph and try predicating same result. I have used the idea of Girvan-Newman Algorithm, explained in our class (week 7 slides – Graph Analysis), This algorithm is depending on calculating the edge betweennees of all edges in the graph and then remove the one that has the maximum value. Repeating this several time will cause the graph to split into different regions. The following is a brief description of how the algorithm works:

- Read the graph vertics and edges from `http://igraph.sourceforge.net/karate.net`.
- Compute the betweenness of all edges using:

  `edge.betweenness( graph )`

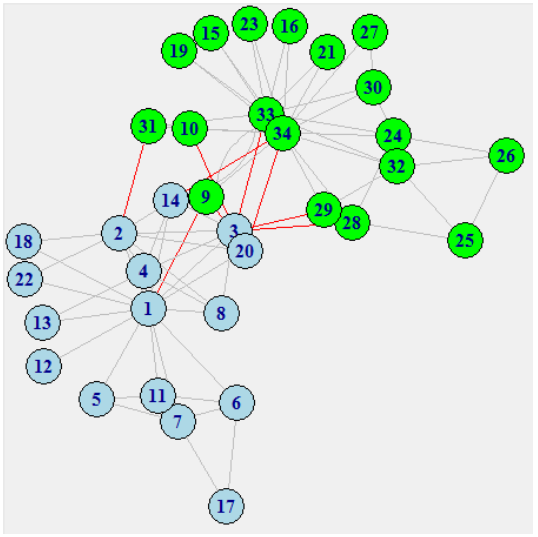
- Remove the edge that has the maximum value.
- Stop if the graph is split into two different clusters. This can be done by checking the value of *clusters(graph)['no']* .
- if *clusters(graph)['no']* is still equal to one, repeat again from step 2.

Let me show the algorithm output into two different format:

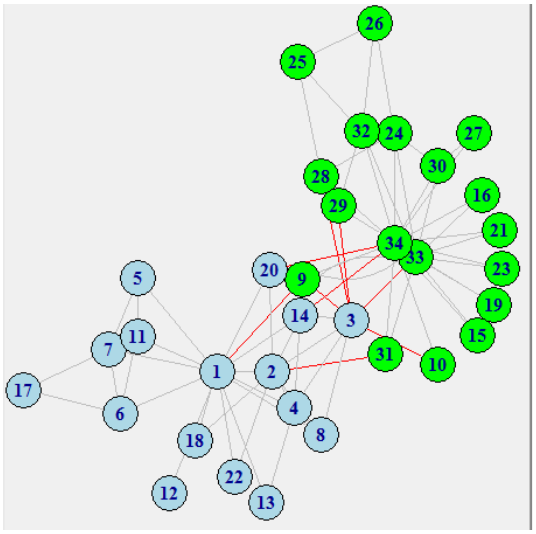
1. Text format: after each iteration, the edge, which has the maximum betweenness value, will be printed on the screen along with its betweenness value:

```
Edges will be deleted in the following order :
1  ->  32   -- Betweenness = 68.454304029304
1  ->  3    -- Betweenness = 62.757375957376
1  ->  9    -- Betweenness = 83.0110106432357
14 ->  34   -- Betweenness = 82.1695725842785
20 ->  34   -- Betweenness = 123.46625041625
3  ->  33   -- Betweenness = 100.372222222222
2  ->  31   -- Betweenness = 139.182683982684
2  ->  3    -- Betweenness = 109.25
3  ->  4    -- Betweenness = 107.666666666667
3  ->  14   -- Betweenness = 142.75
3  ->  8    -- Betweenness = 285
```

2. Graphs: a graph, in R, will be drawn on the screen after each iteration indicating which of the edges has been deleted. see the following figures:

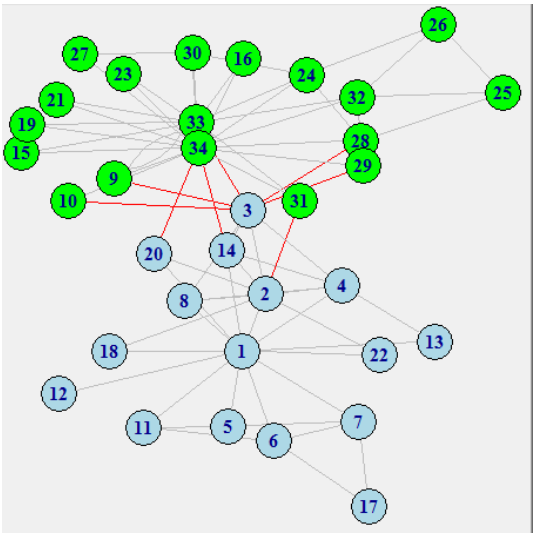


(a) After removing 1 – > 32

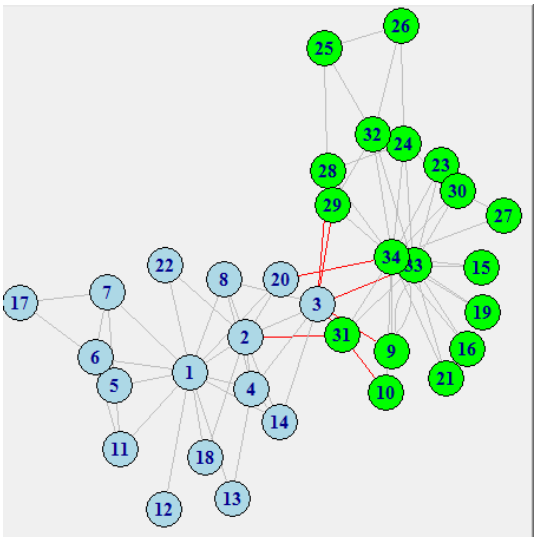


(b) After removing 1 – > 3

Figure 1: First two algorithm iterations

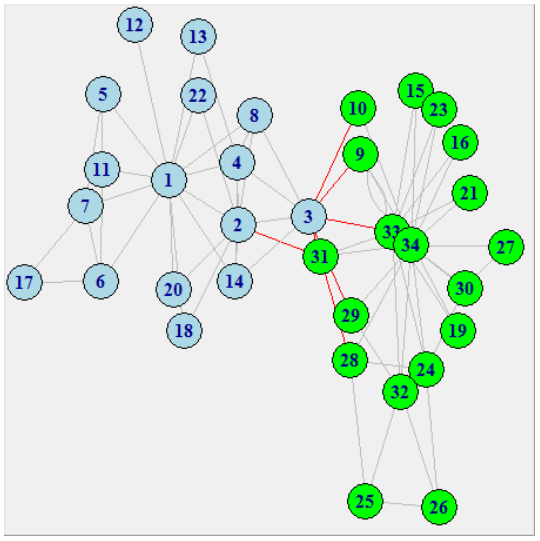


(a) After removing 1 – > 9

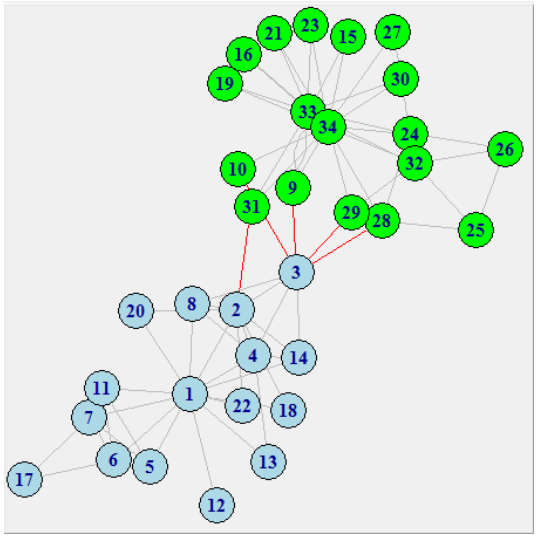


(b) After removing 14 – > 34

Figure 2: Third and forth iterations

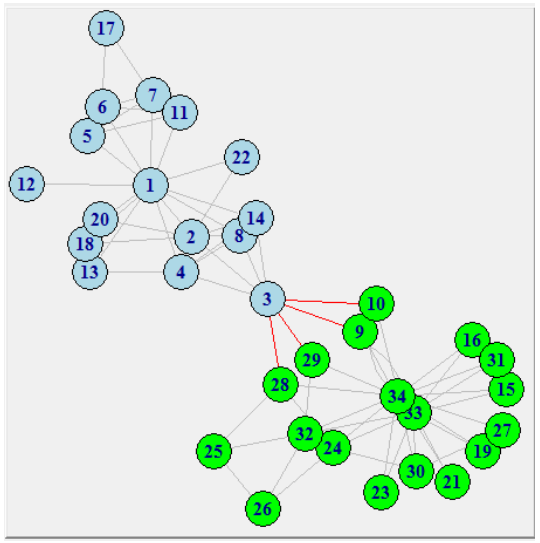


(a) After removing 20 – > 34

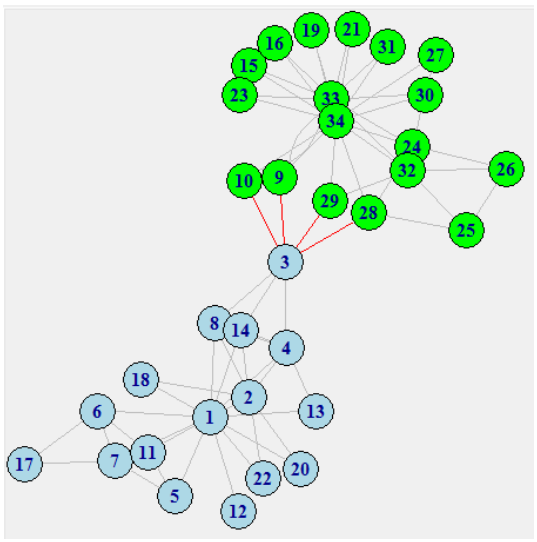


(b) After removing 3 – > 33

Figure 3: Fifth and sixth iterations

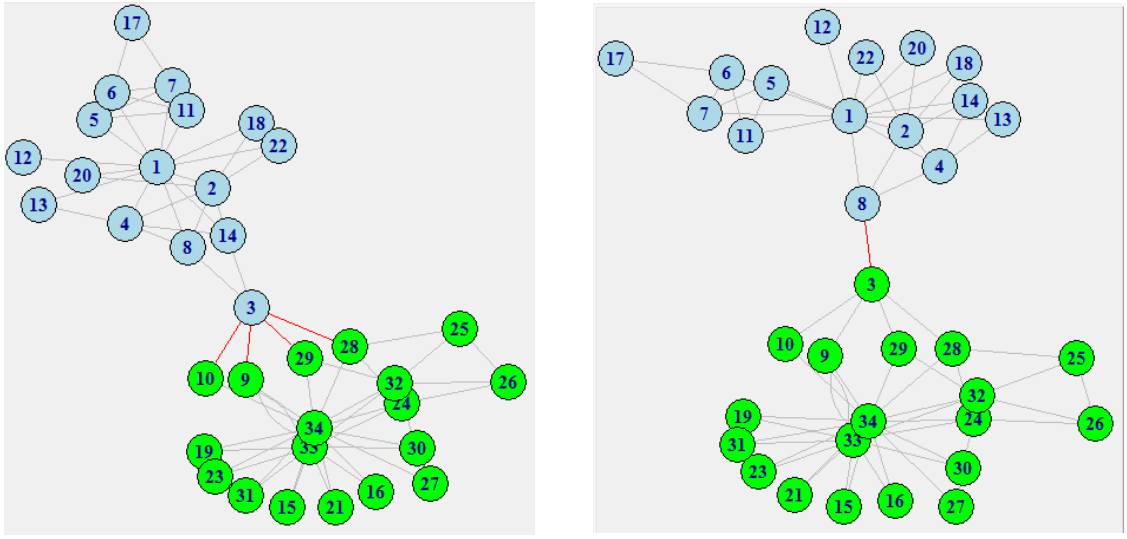


(a) After removing 2 – > 31



(b) After removing 2 – > 3

Figure 4: Iteration number seven and eight



(a) After removing 3 – > 4

(b) After removing 3 – > 14

Figure 5: Ninth and tenth Iterations

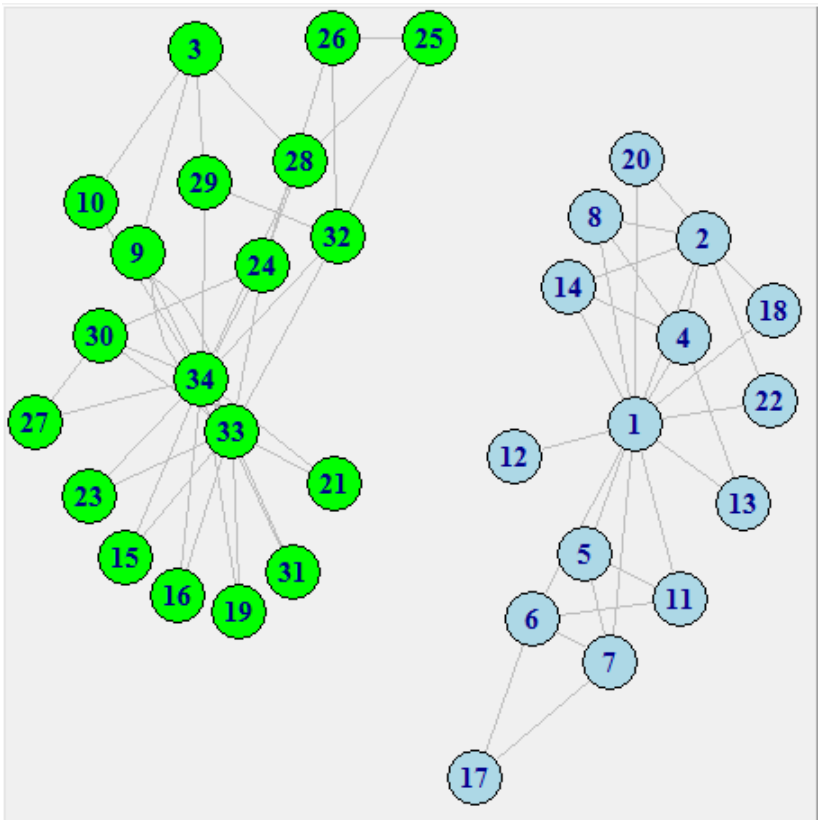


Figure 6: After removing 3 – > 8: Final Graph is split into two clusters

In conclusion, the result obtained from our algorithm is really close to the Karate Club graph. The only difference here is that node no. 3 has decided to change her mind and move to the other party.

Problem 2

This depends on which nodes will join new groups or move between that existing groups, but, in general, I believe that even with adding 2 or 3 different new group, node no. 1 and 34 will remain strong and keep their community alive since they are connecting to most of the other nodes. If more group are established, then, I have no idea how the new graph will look like.