# Assignment 5

*Dr. Nwala*

Nathaniel Everett

March 7, 2018

Contents:

1. We know the result of the Karate Club (Zachary, 1977) split.
Prove or disprove that the result of split could have been predicted by the weighted graph of social interactions. How well does the mathematical model represent reality?

Generously document your answer with all supporting equations, code, graphs, arguments, etc.

Clues:
1. Draw original Karate club graph (two connected components) after split (Week 6 lecture, slide 98).
2. Run multiple iterations of graph partioning algorithm (e.g., Girvan-Newman Algorithm) on experimental Karate club graph until the graph splits into two connected components.
3. Compare the connected components of the experimental graph (in 2.) with the original connected components of the split Karate club graph (in 1.). Are they similar?

Useful sources include:

* Original paper

http://aris.ss.uci.edu/~lin/76.pdf

* Week 6 Slides:

https://docs.google.com/presentation/d/1ihf6N8bHgzM5VLAyHkmF_i5JGUBVpCSdsvYpk8XgHwo/edit?usp=sharing

* Slides

http://www-personal.umich.edu/~ladamic/courses/networks/si614w06/ppt/lecture18.ppt

http://clair.si.umich.edu/si767/papers/Week03/Community/CommunityDetection.pptx

* Code and data

https://networkx.github.io/documentation/networkx-1.10/reference/generated/networkx.generators.social.karate_club_graph.html

https://networkx.github.io/documentation/networkx-1.9/examples/graph/karate_club.html

http://nbviewer.ipython.org/url/courses.cit.cornell.edu/info6010/resources/11notes.ipynb

http://stackoverflow.com/questions/9471906/what-are-the-differences-between-community-detection-algorithms-in-igraph/9478989#9478989

http://stackoverflow.com/questions/5822265/are-there-implementations-of-algorithms-for-community-detection-in-graphs

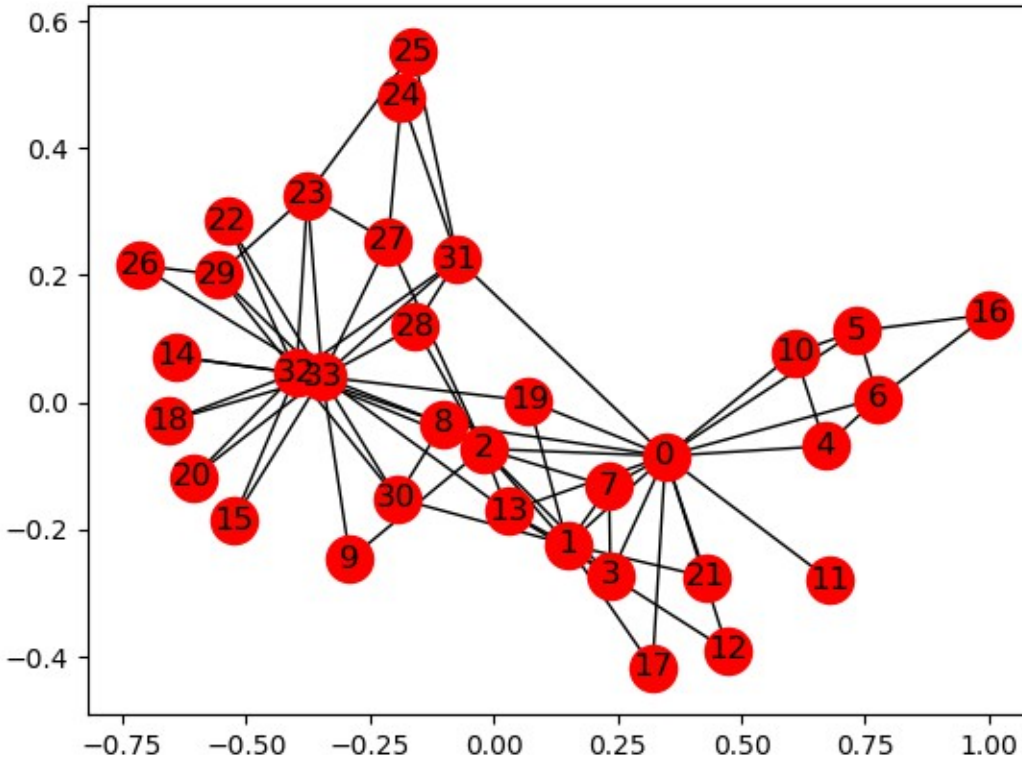
http://konect.uni-koblenz.de/networks/ucidata-zachary

http://vlado.fmf.uni-lj.si/pub/networks/data/ucinet/ucidata.htm#zachary

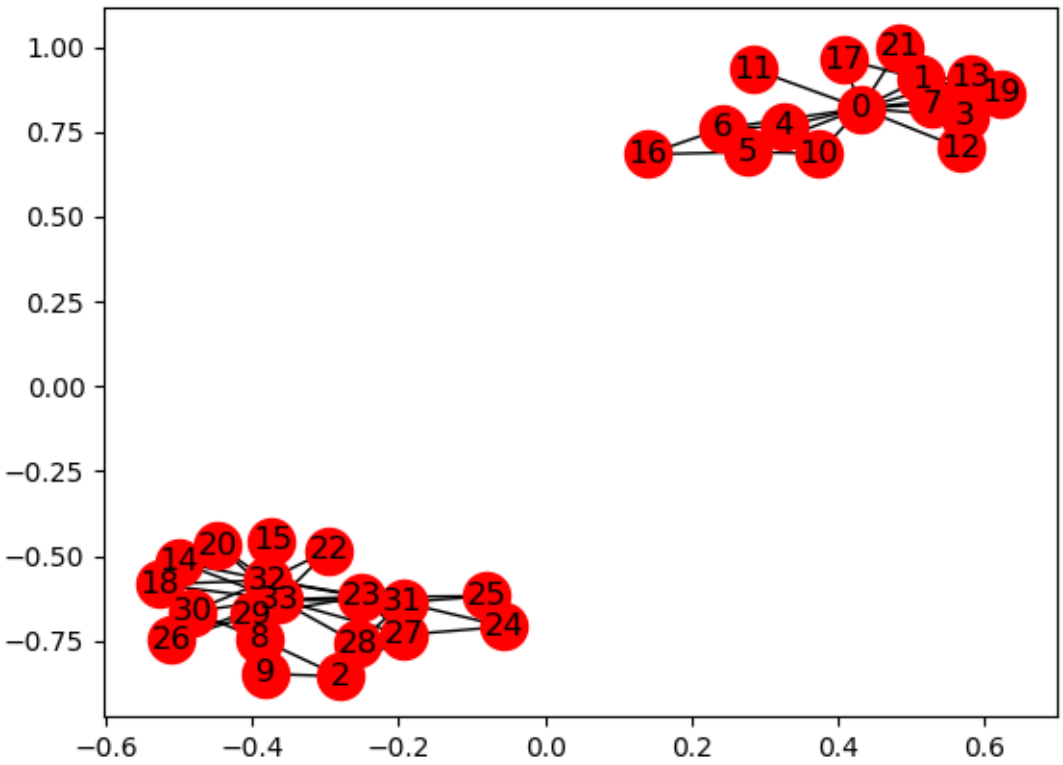
https://snap.stanford.edu/snappy/doc/reference/CommunityGirvanNewman.html

http://igraph.org/python/doc/igraph-pysrc.html#Graph.community_edge_betweenness

Finally I was able to install matplotlib using pip so I can utilize a python program, karateclubsplit.py. This program also runs networkx, and in conjunction with matplotlib, produces the necessary output, which are the network graphs of the Karate Club split. A dictionary is needed for the item lists, and they need to be saved into a new list. This dictionary takes in the edge_betweenness_centrality of networkx. The Girvan-Newman algorithm is presented as a function, and several letter variables are used for getting the very first positioning of the graph at the start.




A total of 12 iterations were needed to get this as the final result:




Other than being separated into two connected components, the graphs do not seem all that different as for where their tuples are located. Notice that the higher-numbered and the lower-numbered tuples tend to stick closely together. Successive iterations (run in problem 3) reveals quite a few differences that the graphs illustrate

---

2. Use D3.js's force-directed graph layout to draw the Karate Club Graph before split. Color the nodes according to the factions they belong to (John A or Mr. Hi). After a button is clicked, split the graph based on the original graph split. Include a link to the HTML/JavaScript files in your report and all necessary screenshots.

See: https://bl.ocks.org/mbostock/4062045

https://d3js.org/

---

This one required some lookups on D3's code, and how to integrate that into HTML coding. I had looked up several examples of how this was done, and even checked out the output, which is color coded before and after the split. My HTML file was done in a similar way, although from what I have noticed the output seems to not show anything. There is also a JSON file included here, which was copied from the sites I found, where it is mostly the same.

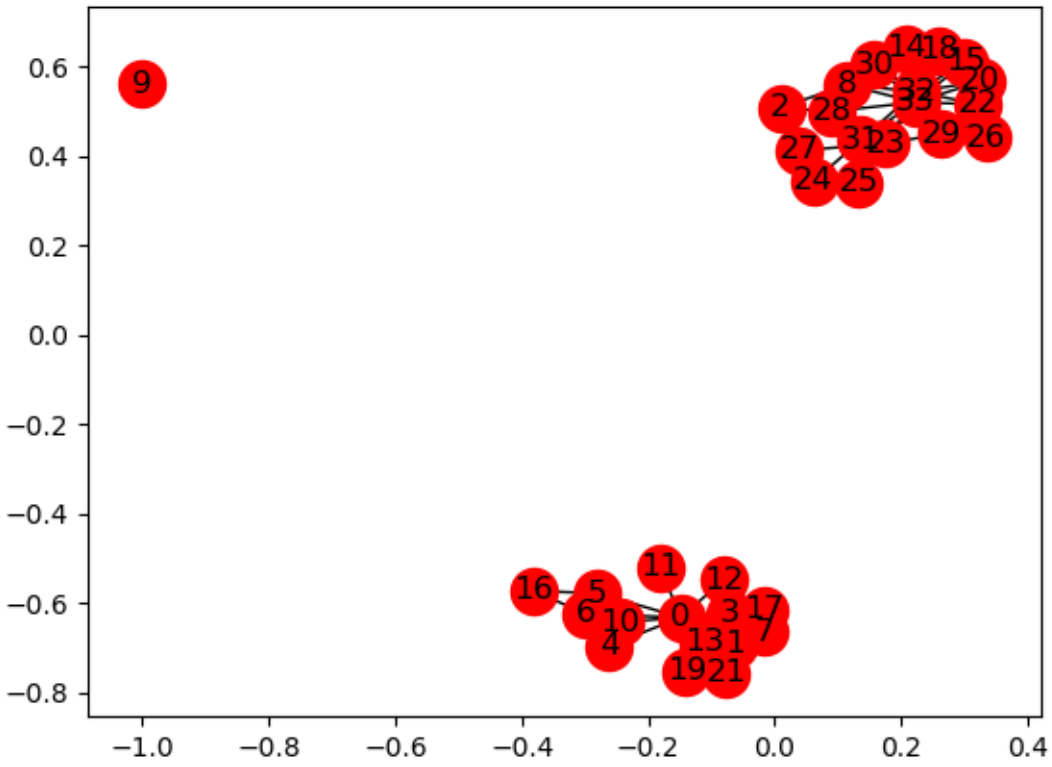
---

3. We know the group split in two different groups. Suppose the disagreements in the group were more nuanced -- what would the clubs look like if they split into groups of 3, 4, and 5?

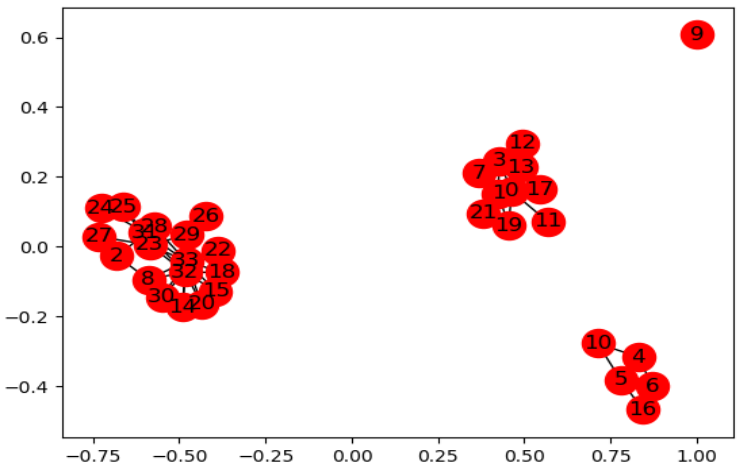
---

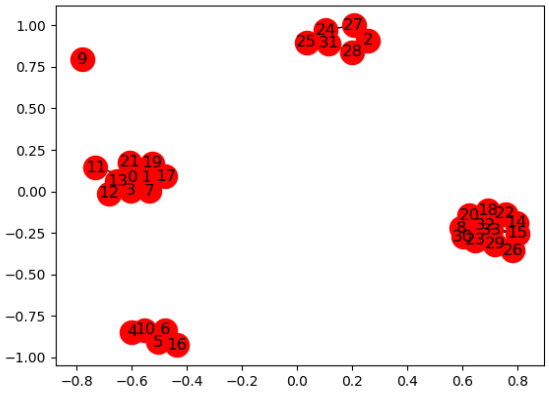
Split3 is a directory for this assignment which uses the same karateclubsplit.py file as before, except

with this code snippet changed: **while(l < 3):**
The changing of the loop makes it so that the program stops after three connected components are developed. 15 iterations were run and the graph ended up like this (note that the split produced one tuple well out of the range of the others, which is odd):




Split4 does the same thing as Split3, except karateclubsplit.py will stop at four connected components. Split5 will do the same except with five connected components. The final graphs for Split4 and Split5 are below. It took 19 iterations and 25 iterations, respectively, to get the final graphs for both.





Files included:
karate00.png – karate11.png – graphs which show the karate club split run through the Girvan-Newman algorithm until the split occurs.
Karateclubsplit.py – Python file to generate the split and the graphs above.
karate_club.json – JSON file for the second part of the assignment
karateclub.html – written HTML code file for the second part of the assignment.
Split3 – a directory redoing the split with a different karateclubsplit.py, except with the program stopping after there are three different components.
Split4 – same as above, except the karateclubsplit.py stops when there are four connected components
Split5 – same as above, except the karateclubsplit.py stops when there are five connected components