CS 432/532 Web Science: Assignment #5

Alexander C. Nwala

Bethany DeMerchant

24 March 2019

Contents

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

Problem 1– Methods:

Graph operations were done using the `NetworkX` library for Python[1]. The `karate_club_graph()` function provided by the library generates the social network graph for Zachary's Karate Club. The library also provides a `girvan_newman()` function to split the graph; however, this function returns only the nodes in each resulting community and no information on the links between nodes or the iterations between the initial graph and the final communities. As this information was necessary for the problem at hand, the Girvan-Newman algorithm was instead implemented by hand using the built-in `edge_betweenness_centrality()` function. The hand-written `girvanNewman()` function executes only one iteration rather than continuing until the graph is split. The function is then repeated in a loop until the graph is split and a picture of the graph at that time is saved after each iteration.

Problem 1– Results:

The original (pre-split) graph, as generated by `networkx.karate_club_graph()`, is shown in Figure 1. The red nodes correspond to members of Mr. Hi's faction and the green nodes correspond to members of John A's faction.

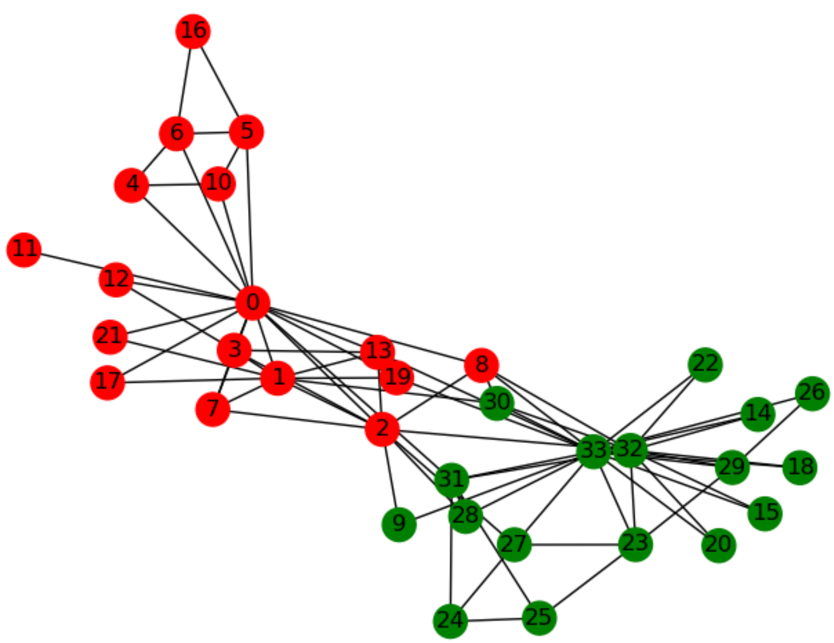

Figure 1: Zachary's Karate Club (NetworkX).

---

[1] https://networkx.github.io/documentation/stable/

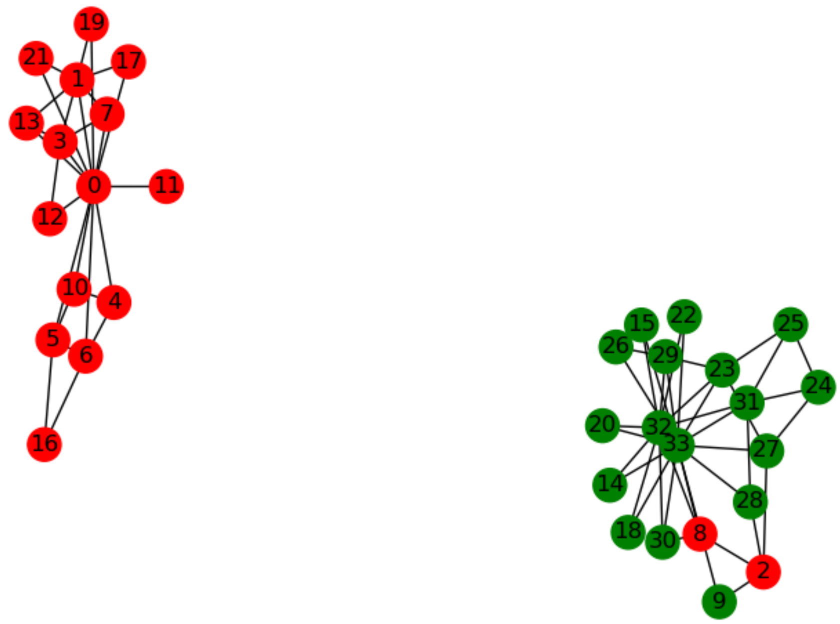

Figure 2: Zachary's Karate Club (experimental split).

Figure 2 shows the graph as split by the Girvan-Newman algorithm[2]. The graphs are quite similar; only two nodes have been miscategorized. The algorithm has assigned nodes 2 and 8 to the Officers' club rather than correctly placing them in Mr. Hi's. No nodes have been incorrectly assigned to Mr. Hi's club. 32 of 34 nodes have been assigned correctly, corresponding to an accuracy rate of roughly 94%. It is also worth noting that node 8 was incorrectly modeled in the same way in the original paper[3].

Problem 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.

Problem 2—Methods:

`D3.js` is a JavaScript library intended for manipulating and displaying data on the web[4]. At the time the initial graph is generated in `karate_club.py`, the data is written to `karate_club.json`. This data is then loaded and used by the `D3.js` library to generate the initial graph. When the button is clicked, the links connecting the two subgroups are removed and the figure is redrawn.

Problem 2—Results:

Figure 3 shows selected screenshots of the graph before and after clicking the button. The graph can be accessed at https://www.cs.odu.edu/~bdemerch/karate_club.html. The `karate_club.html` file and the related `a5.js` file are also available in the `A5` directory on GitHub.

---

[2] The graphs of the intermediate steps have been omitted for brevity. They are available under `A5/graphs/nx`.

[3] http://aris.ss.uci.edu/~lin/76.pdf: PDF page 17/22, p.468. The same node is labeled as 9 because the paper counts from 1, not 0.

[4] https://d3js.org/

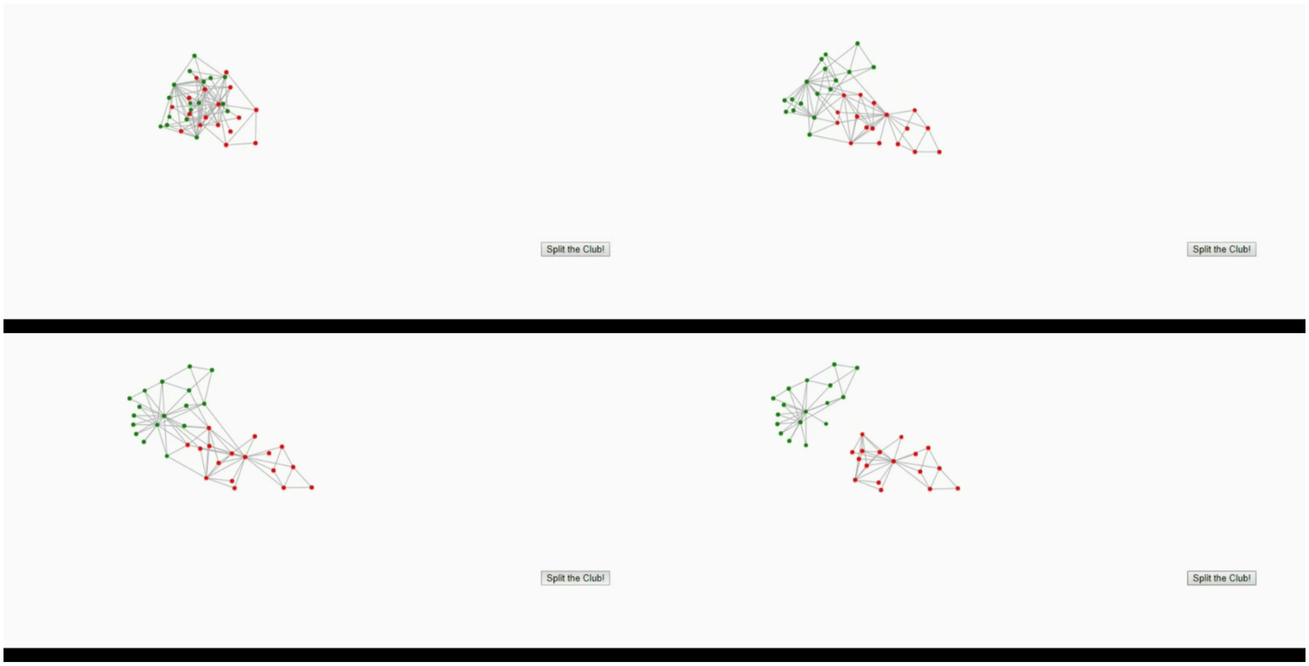

Figure 3: Zachary's Karate Club (D3.js)

Problem 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?

Problem 3—Methods:

As the handwritten `girvanNewman()` function used for Problem 1 removes only one edge at a time, this function can be called repeatedly to further divide the graph into smaller communities. If an integer argument is provided at the command line (e.g., `python karate_club.py 3`), the script will split the graph into that many communities and provide graphs for each step (the default argument is 2).

In addition, the `python-igraph` library provides a `community_edge_betweenness()` function that splits the graph into the smallest communities and returns a dendrogram[5]. At the time the initial graph is generated in `karate_club.py`, the data `community_edge_betweenness()` function is called and the resulting dendrogram is saved to `tree.svg`. Both the original `tree.svg` and the modified version `tree.png` are available in the `A5/graphs/ig` folder.

Problem 3—Results:

Figure 4 shows the dendrogram linked to the `community_edge_betweenness()` function as applied to Zachary's Karate Club graph. The dendrogram can be used to examine the results of breaking the karate club into an arbitrary number of communities. Potential communities of more than one node have been highlighted. Interestingly, the graph can only be split once beyond the initial split before most remaining separations remove one member at a time rather than separating a group into major subgroups. This can also be seen in Figures 5-7, which show the results of attempting to split the graph into 3, 4, and 5 subgroups using the `girvanNewman()` function.

[5] https://igraph.org/python/

Bethany DeMerchant Web Science Assignment 4

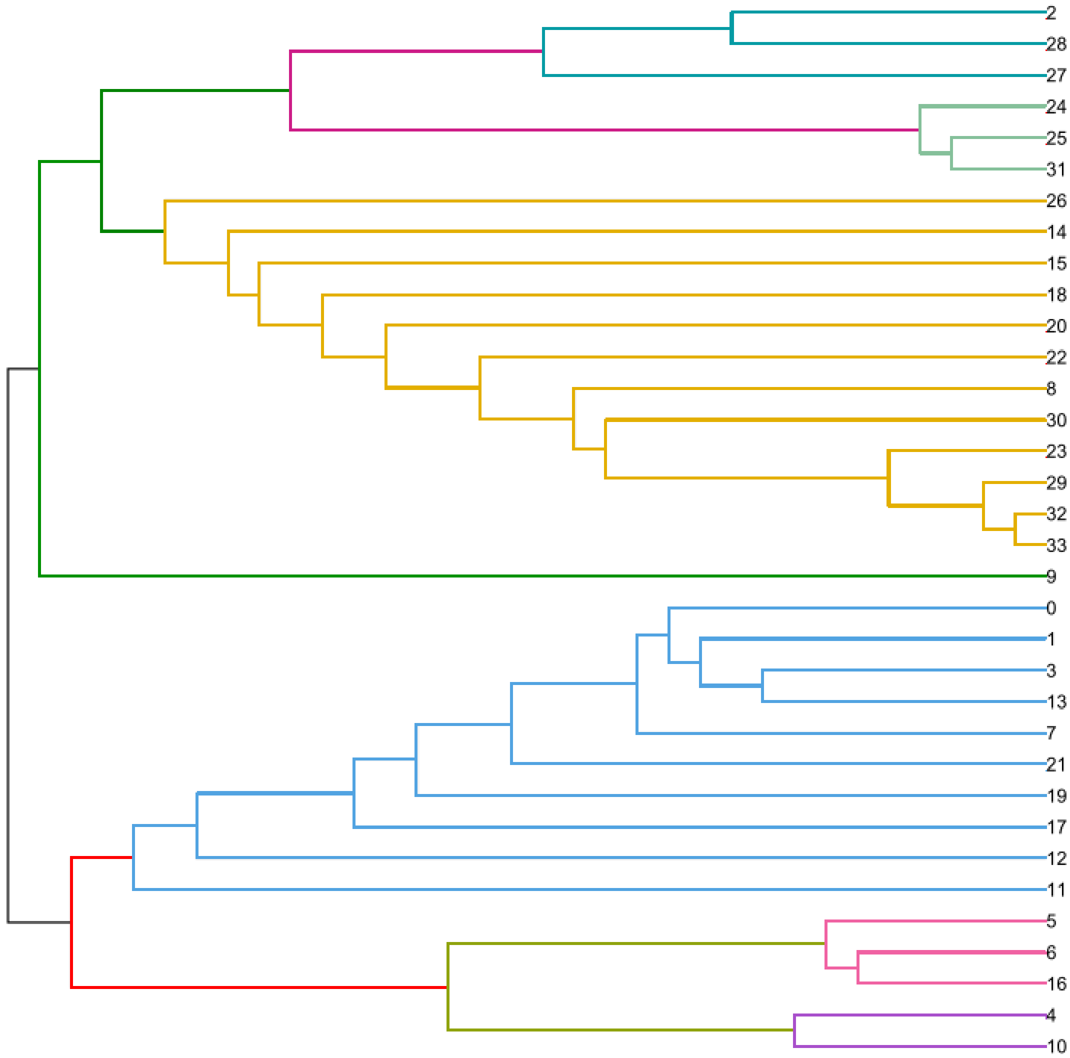

Figure 4: Zachary's Karate Club (arbitrary communities)

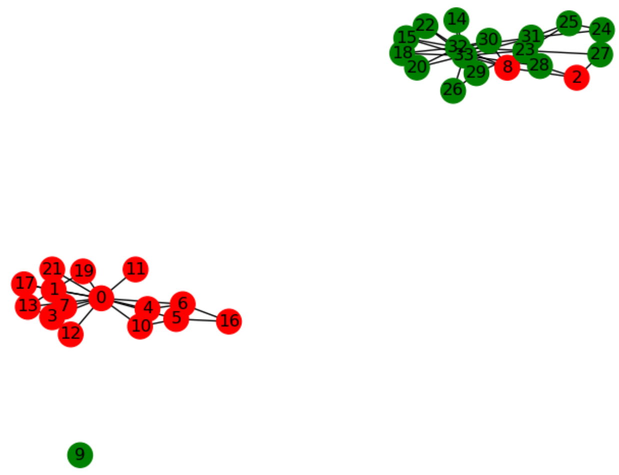

Figure 5: Node 9 has split off instead of a third group forming.

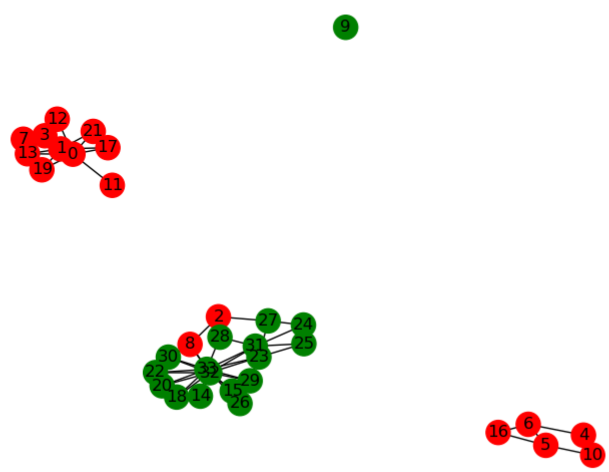

Figure 6: A third group has formed.

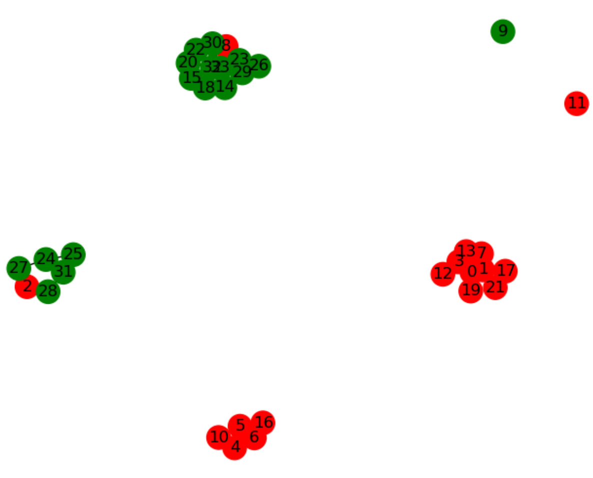

Figure 7: Node 11 has split off before another group formed.