

# **Assignment 6**

**Fall 2014**

**CS595 Web Science**

**Dr. Michael Nelson**

Mathew Chaney

October 24, 2014

## Contents

<b>1</b>	<b>Question 1</b>	<b>3</b>
1.1	Question . . . . .	3
1.2	Answer . . . . .	3
<b>2</b>	<b>References</b>	<b>7</b>

## List of Figures

## Listings

1	Sort command . . . . .	5
---	------------------------	---

# 1 Question 1

## 1.1 Question

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.

Useful sources include:

- \* Original paper

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

- \* 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

[http://networkx.github.io/documentation/latest/examples/graph/karate\\_club.html](http://networkx.github.io/documentation/latest/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>

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

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

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

## 1.2 Answer

Using Dr. Michael Nelson's Twitter account and the Twitter API [1], specifically the GET friends/list [2] request, all of Dr. Nelson's Twitter friends were obtained and saved to the file called **friends**. This method also uses the API's paginating scheme: when there are a large number of results for a query, the API will send a cursor index to show that there are more results to process and that more requests are needed. The code to do this is in Listing ??.

To reduce the impact of high HTTP traffic, the Twitter API rate-limits most requests – the one needed to obtain a user’s friends list has a limit of fifteen message per fifteen minutes. Any requests received from a user or service that has reached the limit will be denied. To ensure no HTTP requests are sent after the limit has been reached the script will sleep until the limit resets. This is accomplished using Python’s `time` package [3] and the methods shown in Listing ??.

The `get_limit` method uses the API to find the number of available requests remaining for the GET friends/list method and also the time at which the limit will reset, received as seconds since the Unix epoch [4]. This method, combined with the `wait_for_reset` method, allowed the script to restart after an interruption and only require waiting for the appropriate amount of time. The sleep time was extended by 5 seconds to allow for a small buffer in case of mathematical errors.

The friends of Dr. Nelson’s friends were then obtained with the same `get_friends` method from Listing ?? and stored in a file called `friend_counts`, each on a single line preceded by their friend count. All of these operations were controlled by a main method, which is shown in Listing ??.

The `friend_counts` file was ordered in place with the Unix command in Listing 1.

```
1 [mchaney@mchaney-l a5]$ cat friend_counts | sort -g -o friend_counts
```

Listing 1: Sort command

This file was then processed by the R script shown in Listing ?? to produce the graph in Figure ??

The median, mean and standard deviation were all calculated, with the median, mean and median plus one standard deviation plotted as horizontal lines that intersect the data plot at their y-values. Only a single line was drawn for the standard deviation because the lower-end value was negative, and thus off the graph.

## 2 References

- [1] Twitter, Inc. Twitter API: Overview. <https://dev.twitter.com/overview/api/>, 2014.
- [2] Twitter Inc. Twitter API: GET friends/list. <https://dev.twitter.com/rest/reference/get/friends/list/>, 2014.
- [3] The Python Software Foundation. Python time module. <https://docs.python.org/2/library/time.html>, 2014.
- [4] Community Wiki, Stack Overflow. Why is 1/1/1970 the “epoch time”? <http://stackoverflow.com/questions/1090869/why-is-1-1-1970-the-epoch-time>, Last edited June 23, 2011.