Assignment 4

CS 532: Introduction to Web Science

Spring 2016

Manoj Chandra Kompalli

Finished on February 25, 2016

1

Question

1. Determine if the friendship paradox holds for my Facebook account.* Compute the mean, standard deviation, and median of the number of friends that my friends have. Create a graph of the number of friends (y-axis) and the friends themselves, sorted by number of friends (x-axis). (The friends don’t need to be labeled on the x-axis: just f1, f2, f3, ... fn.) Do include me in the graph and label me accordingly.

* = This used to be more interesting when you could more easily download your friend’s friends data from Facebook. Facebook now requires each friend to approve this operation, effectively making it impossible.

I will email to the list the XML file that contains my Facebook friendship graph ca. Oct, 2013. The interesting part of the file looks like this (for 1 friend):

```
<node id="Johan_Bollen_1448621116">
        <data key="Label">Johan Bollen</data>
        <data key="uid"><![CDATA[1448621116]]></data>
        <data key="name"><![CDATA[Johan Bollen]]></data>
        <data key="mutual_friend_count"><![CDATA[37]]></data>
        <data key="friend_count"><![CDATA[420]]></data>
</node>
```

It is in GraphML format: http://graphml.graphdrawing.org/

Mean	358.987
Median	266.500
Std Dev	371.585

Table 1: Statistics of Dr.Nelson's Friends' friends values taken from R

Answer

I was trying to figure out how to get my own graphml to fetch my friends data .I could not succeed due to facebooks privacy settings. I have downloaded Dr. Nelsons friend list in graphml format. Now the task ahead of me was to find a way to extract the graphml data and sort it.I researched and found a library called minidom which can extract the xml data. I have extracted the data tag and two key values name and friend count. The "name" got me all the friends names and friend_count shown in Listing 1 gave me the count of friends for that friend.I have iterated the loop for all the friends to get the complete list. I have sorted the list in increasing order using Excel and copied the values to a text file.I had computed Dr. Nelsons friends and added it to the list of his friends.

I used the text file as input to R.I found the mean, median and standard deviation of the sorted list in R.Code shown in Listing 2. I have plotted a bargraph shown in Figure 1 . I had found out that Dr. Nelson has less than half of the mean of his friends friends that is 154 .This proves the friendship paradox.

The R script generates these values:

```
Mean     :358.987
Median   :266.500
Std Dev:371.585
```

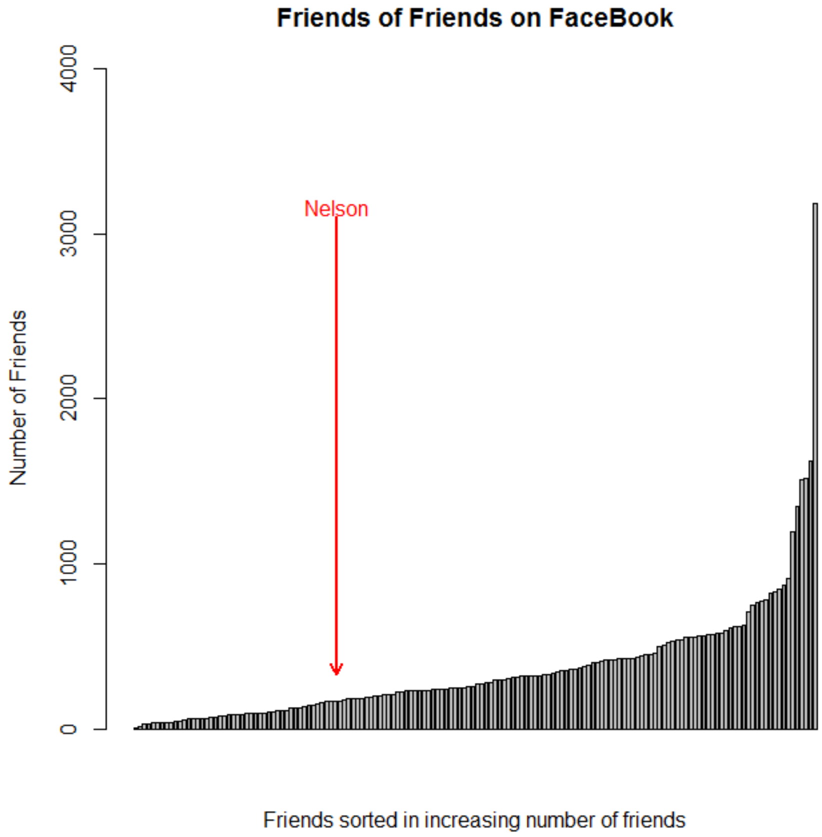

Figure 1: Bar plot showing the count of Dr.Nelson's Facebook Friends' Friends

```python
#!/usr/local/bin/python3
import csv
import sys
import numpy

from xml.dom.minidom import parseString


def friendData(xml):


    dom = parseString(xml)
    totalCount = {}

    for element in dom.getElementsByTagName("data"):
        if (element.attributes['key'].value == 'name'):
            name = element.childNodes[0].data

        if (element.attributes['key'].value == 'friend_count'):
            count = element.childNodes[0].data




            totalCount[name] = count



            name = ''
            count = ''

    return totalCount

def countFriends(xml):

    dom = parseString(xml)
    return len(dom.getElementsByTagName("node"))


if __name__ == "__main__":
        sum=0
        counter=0


        graphmlFile = sys.argv[1]
        f = open(graphmlFile)
        f2=open('output.csv','w')
        xml = f.read()
        f.close()
        countProfFriend = countFriends(xml)
        fbFriends = friendData(xml)
        print("Friend_Name,Friend_Count")
        print('Michael Nelson,' + str(countProfFriend))
        f2.write("Friend,")
        f2.write("Count")
        f2.write("\r\n")
        f2.write("Michael Nelson ,")
        f2.write("%s\n" %str(countProfFriend))

        # f2.write("\r\n")


        for friend in fbFriends:
                # print(friend + ',' + fbFriends[friend])
                f2.write("%s ,"%friend)

                friend1=int(fbFriends[friend])
                sum+=friend1
                counter+=1
                # print friend1




                f2.write("%s\n"%fbFriends[friend])
        print "Mean = ",sum/counter
```

Listing 1: Python program for processing GraphML file

```
data <- read.csv(”friendCount.txt”, stringsAsFactors = F, header = FALSE, sep = ” ”)

myData = data[,1]

meanData <- paste(”Mean: ”, mean(myData), collapse = ””)

medianData <- paste(”Median: ”, median(myData), collapse = ””)

sdData <- paste(”Std Dev: ”, sd(myData), collapse = ””)

write(meanData, stdout())
write(medianData, stdout())
write(sdData, stdout())

pos <- (myData == 155)

cols <- c(”white”, ”red”)


barplot(myData, main=”Friends of Friends on FaceBook”, xlab=”Friends sorted in increasing number of friends”, ylab=”Number of Friends”, col=cols[pos + 1], ylim=c(0, 4000))
text(x=match(c(155), myData)+12, y=(max(myData)-20), labels=”Nelson”, col=’red’)
arrows(x0=match(c(155), myData)+12, y0=(max(myData) - 80), x1=match(c(155), myData)+12, y1=325, length=0.1, lwd=2.5, col=’red’)
```

Listing 2: R program for bar plot shown in Figure 1

2

Question

2. Determine if the friendship paradox holds for your Twitter account. Since Twitter is a directed graph, use "followers" as value you measure (i.e., "do your followers have more followers than you?").

Generate the same graph as in question #1, and calcuate the same mean, standard deviation, and median values.

For the Twitter 1.1 API to help gather this data, see:

https://dev.twitter.com/docs/api/1.1/get/followers/list

If you do not have followers on Twitter (or don't have more than 50), then use my twitter account "phonedude_mln".

Extra credit, 3 points:

Mean	141518.205
Median	25
Std Dev	824927.542

Table 2: Statistics on the count of my Twitter followers' followers, values straight from R

Answer

Again, a similar question but a different approach. I had used Tweepy API to extract links in the previous assignments. Thankfully, this experience came in handy. It was a relatively simple task this time. I had to use tweepy.Cursor object and followers_count gave me the number of followers of my followers. screen_name gave me the user name .I have again looped through all users and wrote the values to a csv file. I had sorted the csv file,copied it to a text file and extracted the data. I used this list to generate the mean, median and standard deviation. This time around, the mean was even higher due to a single user having large number of followers. I have plotted a bar graph which again puts me nowhere in comparision to the mean. Friendship paradox is proved again.

From the output, I can see that I have 34 followers.It is greater than the median.It means that I have a considerable rank amongst my followers. The R script shown in Listing 4 creates a similar bar plot to that shown in answer one and produces the statistics shown in listing 2.

```
Mean:    141518.205
Median:   25
Std Dev:    824927.542
```

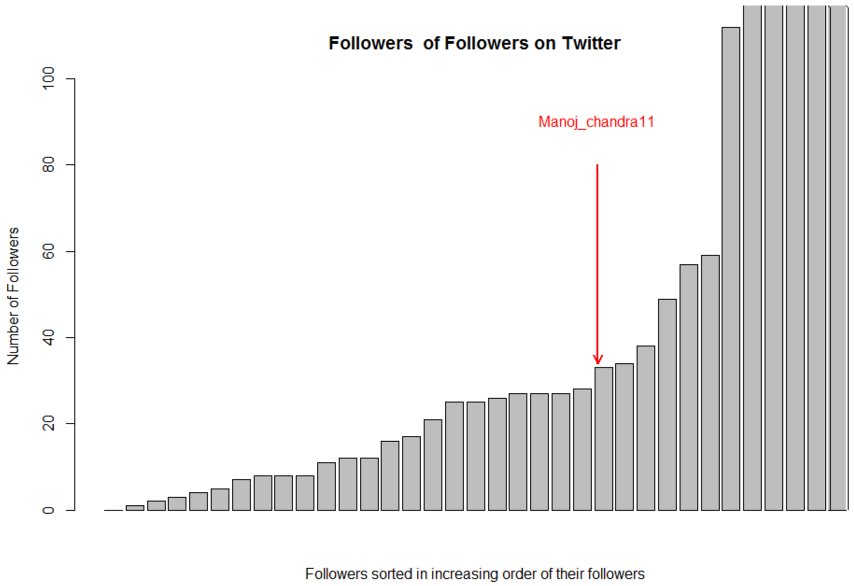

Figure 2: Bar plot showing the count of my Twitter followers' followers

```
import tweepy

import time
import sys
import re

# Authentication Keys to Connect to Twitter API
consumer_key=”vjemit5xYQdhgrEPa1FeFf5ZO”;
consumer_secret=”0
    PoNwIkHk29kUweChIIhGzVD3ZfXwRVqqwxYY3zPadY1BZeNq8”;
access_token=”3485785534-4FlFNlJtg1uNAlMummglVi9feR7fyvkUS0STp0G
    ”;
access_token_secret=”
    EpzAYEpf6tHdGFKj43HhnBeAhLNgkyXPdZyH72ec8Ew8d”;
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tweepy.API(auth) #Accessing tweepy API
counter=0

file1=open('follower.csv','w')


for user in tweepy.Cursor(api.followers, screen_name=”
    Manoj_Chandra11”).items():

        counter+=1

        file1.write(”%s ”%user.followers_count)
        file1.write(”,”)
        file1.write(”%s\n ”%user.screen_name)
```

Listing 3: Python program for acquiring Twitter followers for phonedude_mln

```
data <- read.csv("orderedfollowers.txt", stringsAsFactors = F,
    header = FALSE, sep = ",")
mydata = data[,1]
meanData <- paste("Mean: ", mean(mydata), collapse = "")
medianOut <- paste("Median: ", median(mydata), collapse = "")
sdData <- paste("Std Dev: ", sd(mydata), collapse = "")
write(meanData, stdout())
write(medianOut, stdout())
write(sdData, stdout())

barplot(mydata, main="Followers  of Followers on Twitter", xlab=
    "Followers sorted in increasing order of their followers",
    ylab="Number of Followers ",ylim=c(0,100))
 arrows(x0=match(c(34), mydata)+3, y0=80, x1=match(c(34), mydata
     )+3, y1=34, length=0.1, lwd=2.5, col='red')
 text(x=match(c(34), mydata)+3, y=90, labels="Manoj_chandra11",
     col='red')
```

Listing 4: R program for bar plot shown in Figure 2

3

Question

```
3.  Repeat question #1, but with your LinkedIn profile.
```

Answer

Not attempted.

4

Question

Extra credit, 1 point:

4. Repeat question #2, but change "followers" to "following"? In other words, are the people I am following following more people?

Mean	4938.863
Median	115
Std Dev	2881.577

Table 3: Statistics on the count of my Twitter friends' friends, values straight from R

Answer

This is again a very similar approach to what I did with the second question.I had to extract the friends friends data. It is the "following" data. Here, I used friends_count instead of followers_count.

The result appears to hold in the case of "following", as shown in Figure 3.I again lose terribly to the mean value.Hence,friendship paradox is proved again. The Python code for this case is in Listing 5 and the R code for this Figure is in Listing 6. The statistics are shown in Table 3.

Both scripts are executed as shown in the answer to Question 2.

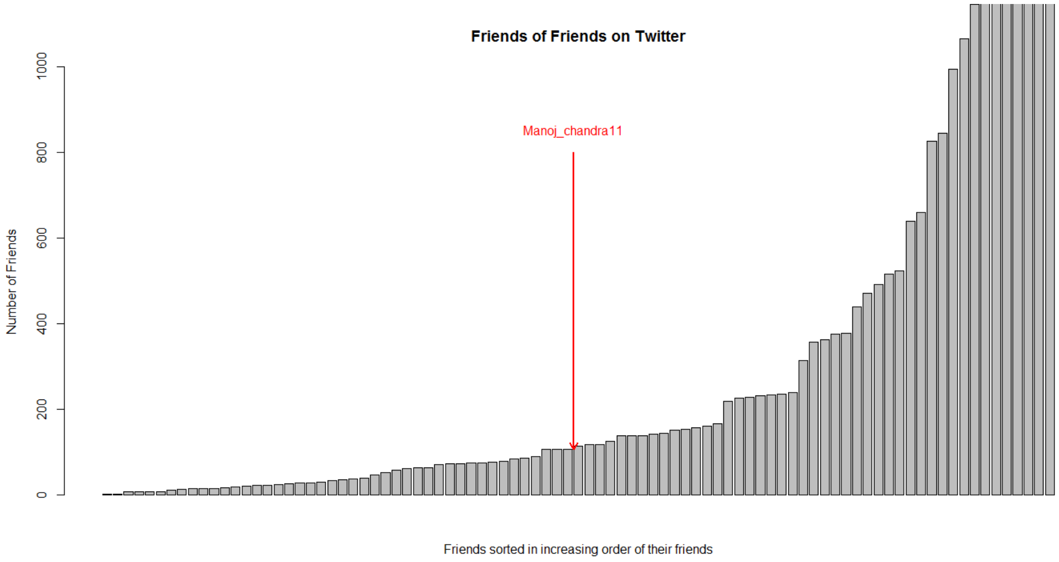

Figure 3: Bar plot showing the count of my Twitter friends' friends

```
import tweepy
import time
import sys
import re

# Authentication Keys to Connect to Twitter API
consumer_key=”vjemit5xYQdhgrEPa1FeFf5ZO”;
consumer_secret=”0
    PoNwIkHk29kUweChIIhGzVD3ZfXwRVqqwxYY3zPadY1BZeNq8”;
access_token=”3485785534−4FlFNlJtg1uNAlMummglVi9feR7fyvkUS0STp0G
    ”;
access_token_secret=”
    EpzAYEpf6tHdGFKj43HhnBeAhLNgkyXPdZyH72ec8Ew8d”;
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tweepy.API(auth) #Accessing tweepy API
counter=0

file1=open('following.csv','wb')

for user in tweepy.Cursor(api.friends, screen_name=”
    Manoj_Chandra11”,count=200).items#gets all my friends details
     on twitter
         print user.screen_name
         print user.friends_count
         file1.write(”%s”% user.screen_name)
         file1.write(”,%s\n”% user.friends_count)
```

Listing 5: Python program for acquiring my Twitter followers

```
data <- read.csv("followingorder.csv", stringsAsFactors = F, header = FALSE, sep = ",")
mydata = data[,2]
meanOut <- paste("Mean: ", mean(mydata), collapse = "")
medianOut <- paste("Median: ", median(mydata), collapse = "")
sdOut <- paste("Std Dev: ", sd(mydata), collapse = "")
write(meanOut, stdout())
write(medianOut, stdout())
write(sdOut, stdout())
pos <- (mydata == 89)
 barplot(mydata, main="Friends of Friends on Twitter", xlab="Friends sorted in increasing order of their friends", ylab="Number of Friends ",ylim=c(0,1000))
 text(x=match(c(89), mydata)+12, y=850, labels="Manoj_chandra11", col='red')
 arrows(x0=match(c(89), mydata)+12, y0=800, x1=match(c(89), mydata)+12, y1=105, length=0.1, lwd=2.5, col='red')
```

Listing 6: R program for bar plot shown in Figure 3

References

[1] Dr.Nelson's powerpoint slides:.

http://phonedude.github.io/cs532-s16/

[2] Minidom API.

https://docs.python.org/2/library/xml.dom.minidom.html.

[3] Getting the follower's data using Tweepy:.

http://stackoverflow.com/questions/17455107/the-best-way-to-get-a-list-of-followers-in-python-with-tweepy.

[4] Getting the friends' friend's count in Twitter:.

http://stackoverflow.com/questions/32322519/tweepy-iterating-over-tweepy-cursorapi-friends-items.