Assignment 4

Tarek Fouda

2016-02-25

1 Introduction

This report explains how I managed to solve Assignment 4 in Web Science class which is due 02/25/2016. It is mainly consisted of Two Questions and two extra credit points questions. Will show my approaches and implementation in each of the two Questions.

2 Problem 1 Getting the Median/Standard Deviation/Mean

2.1 friendship paradox

Basically, there is something called the "friendship paradox" which checks if your friends have more friends than you have. First, I implemented a program that takes the xml file given in the assignment, and extract all the data tags that has a key equals to " Friends count" and put it in a text file named "friendscount.txt". We were looking to the following tag:

```
<data key="friend_count">
```

The following program is the code that performs this extraction from the xml file:

```
def readingxml():
        i = 0
        with open('xml.txt', 'r') as file:
                with open('freindscount.txt','w') as f:
                        for line in file:
                                if '"friend_count"' in line:
                                        i = i +1
                                        counter = str(int(i))
                                        f.write(line)
```

Listing 1: python code to get friend counts from XML file

After that we get the text file named "friendscount.txt" and we loop through all it's lines and get the number that exists in each line. That will be the number of friends. The following is the program that performs such a task:

```
def extractingnumber():
        with open('freindscount.txt','r') as f ,open('numbers_count
            .txt ','w') as n:
                        for line in f:
                                temp="".join(re.findall(r'\d+',line
                                    )) +"\n"
                                n.write(temp)
```

Listing 2: python code to get number in each line

The next step after getting the number of friends is to perform the calculations to get the mean, standard deviation, and median of the number of friends that the professor's friends have.

```
import re
import math
import statistics


def readingxml():
        i = 0
        with open('xml.txt', 'r') as file:
                with open('freindscount.txt','w') as f:
                        for line in file:
                                if '"friend_count"' in line:
                                        i = i +1
                                        counter = str(int(i))
                                        f.write(line)


def extractingnumber():
        with open('freindscount.txt','r') as f ,open('numbers_count
            .txt ','w') as n:
                        for line in f:
                                temp="".join(re.findall(r'\d+',line
                                    )) +"\n"
                                n.write(temp)


def calculateMean():
        lis=[]
        with open ('numbers_count.txt ','r') as nc:
                total = sum(int(x)
                for line in nc
                for x in line.split())
                #print ("Total = ",total)
        mean = total /154
        print ("Mean = ",mean)
        return mean



def calculateSD(var):
        ls=[]
        with open ('numbers_count.txt ','r') as nc:
                for line in nc:
                        no = int(line) - var
```

```
                        no2= no *no
                        ls.append(no2)
                sqTotal = sum(ls)
                Mean= sqTotal/ var
                sd= math.sqrt(sqTotal)
                print ("STD = ",sd)









def calculateMedian():
        ls=[]
        with open ('numbers_count.txt ','r') as n:
                for line in n:
                        ls.append(line.strip('\n'+''))
                ls =list(map(int, ls))
                median = statistics.median(ls)
                print ("Median = ",median)
readingxml()
extractingnumber()
mean = calculateMean()
calculateSD(mean)
calculateMedian()
```

Listing 3: python code to get number in each line

The following is the graph, the x-co-ordinates represents the friends number, and the y-co-ordinates represent the number of friends the professor's friends have.

The calculations which were used in the above graph are shown below upon executing the python code:

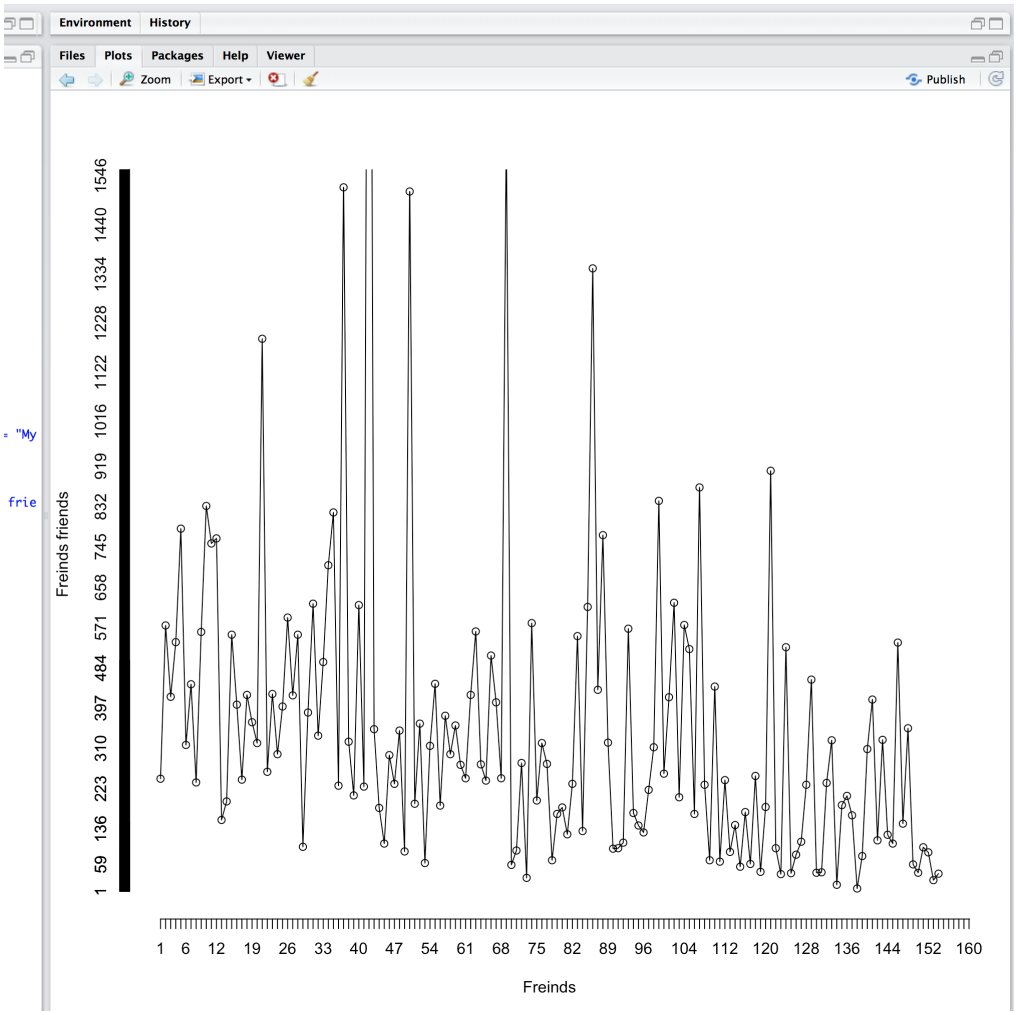

Figure 1: TheGraph

```
Mean =  358.987012987013
STD =  4596.256299862528
Median =  266.5
```

Figure 2: TheGraph

3 Problem number 2

In thisPart of the assignment we were required to get the number of followers my twitter account has and see if the "friendship paradox" will hold, or in other words if my followers have more followers than I have.

So we need to check the number of followers I have, then loop on each one of these followers, and check the number of followers they have and do the calculations the same as part 1.

I used Tweepy Api to get the number of followers.

```
import time
import tweepy

consumer_key = "KEtYeXDYJwgdtX0IHuwUf1Hsw"
consumer_secret = "xkErPgMDfParcniEkbhRLgf8T6EXGOfhNP1bAbKSbog5rqCKxi"
access_key = "308651543-bWKmgqe2AP3xTx85jyHPBUrovjdMtNej2SOqOjZd"
access_secret = "PgKsQJxjvqaocAZmNG2D5t2Q7ZkAcDoPTHvLfOEe2ghj9"

auth = tweepy.OAuthHandler(consumer_key,consumer_secret)
auth.set_access_token(access_key,access_secret)

api = tweepy.API(auth)

user = api.get_user('amrali70')

print "Number of Followers I have is = " + "{0}".format(user.followers_count)

ids = []
i = 0
for user in tweepy.Cursor(api.followers, screen_name="amrali70",count=1000).items():
    try:
      i = i+1
      name = api.get_user(user.screen_name)
      with open('detailsfollowers.txt','a') as f:
        line = "Number of Followers "+ user.screen_name + " have is = " + "{0}".format(name.followers_count) + '\n'
        f.write(line)
      print "Number of Followers "+ user.screen_name + " have is = " + "{0}".format(name.followers_count)
      if((i%50) == 0):
        print i
      if(i==299):
        time.sleep(60*15)
    except:
        print "waiting"
        time.sleep(10)
print "Number of Followers" + "{0}".format(i)
```

Listing 4: python code to get number of followers my twitter username and my follower's usernames have

And this is a sample of the output I get on running the above code:

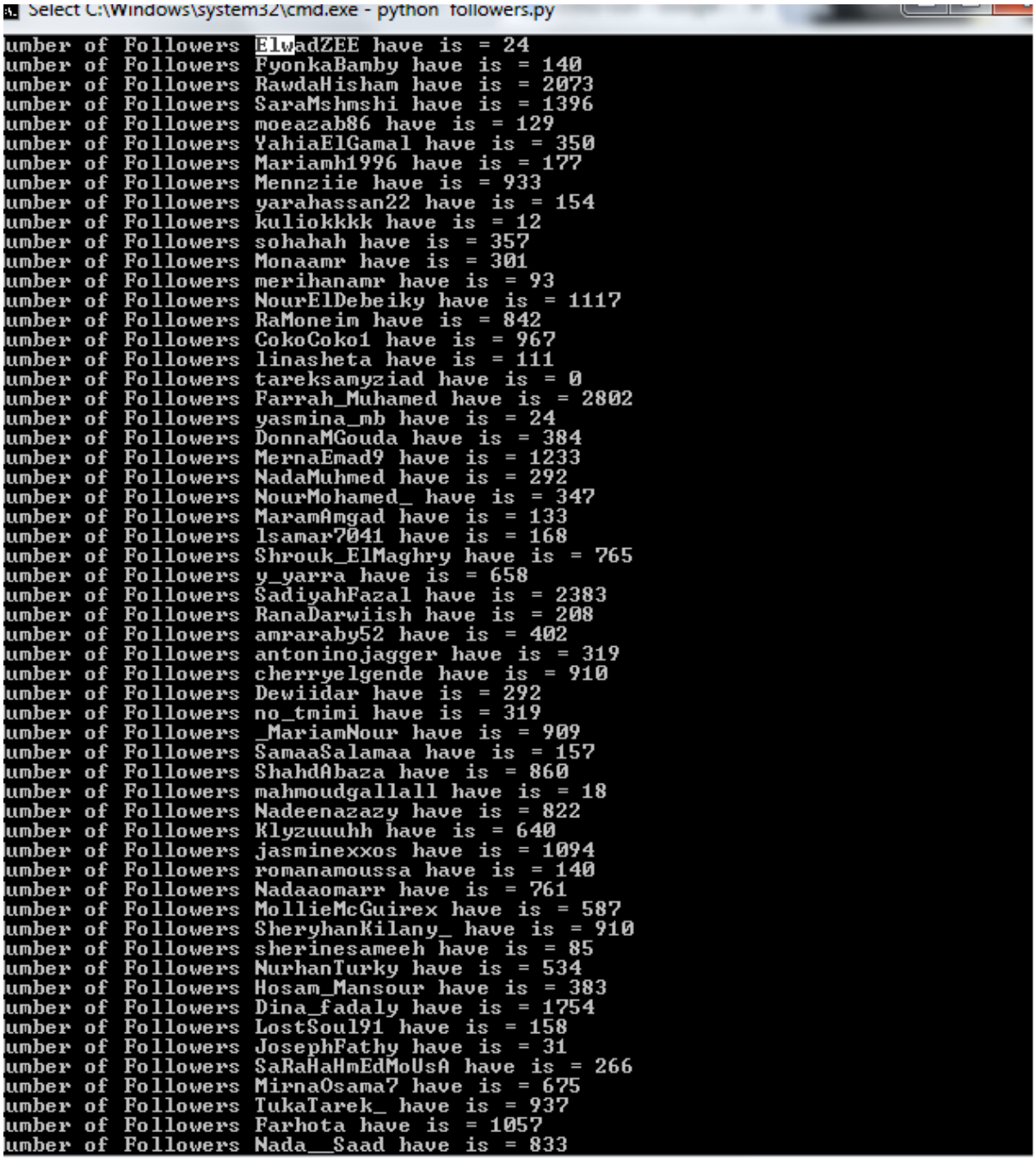

Figure 3: The followers and the number of followers they have

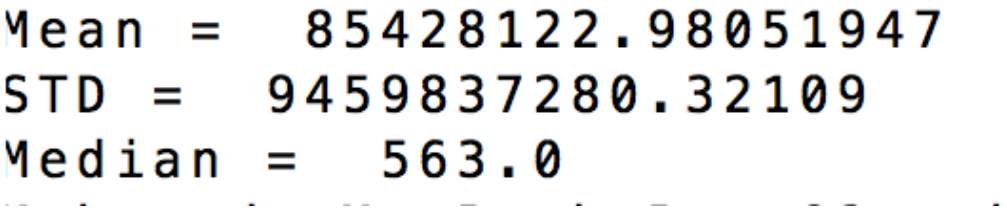

Figure 4: The followers and the number of followers they have

I printed the output in a text file and by executing the same methods and techniques in 2, I managed to calculate the mean, median and standard deviation. These were the calculations and the graph:

So the x-axis has the number of friends I have, and the Y-axis has the number of friends each follower have.

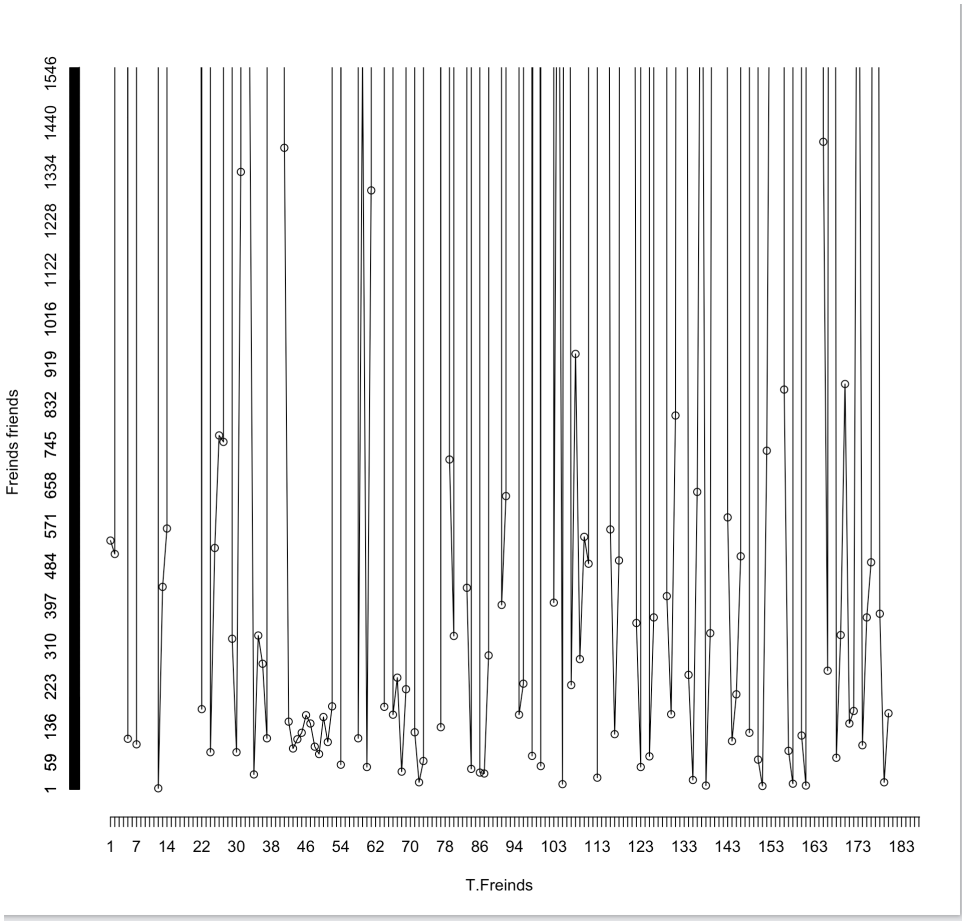

Figure 5: The followers and the number of followers they have