Assignment 4

CS532-s16: Web Sciences

Spring 2016

John Berlin

Generated 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/

Answer

After downloading the GraphML file and visually inspecting it I thought to myself there has to be a library for this. As usual there was one for python called *Pygraphml* [1]. Using this library made parsing and extraction of the information easy. The python script to extract the information is found in listing 1. The process was so easy please as the library puts all data portions of a node inside of a dictionary and simply loop through the nodes of the graph for them.

As usual be sure to be in the directory containing the graphml file. To run the script execute it as such:

```
$ chmod +x parseGraph.py
$ ./parseGraph.py
```

After the Python script finishes running it will produce a file called *mlnfb-count.csv*. This file contains the number of friends Dr. Nelson's friends have as

Mean	358.987
Median	266.5
Std Dev	371.585

Table 1: Statistics from MLN Facebook friends

well as an entry of how many friends he has. His entry is not included in the calculations of the mean, median, and standard deviation. Those calculations can be found in table 1.

Dr. Nelson has 154 Facebook friends which means he has less friends than his Facebook friends. How can I be sure of that. For one he has less friends than the median. Secondly I used the R script found in listing 2 to generate the plot seen in figure 1 to calculate what percent of his friends have more or less friends than him. Those results are seen below.

```
mln has less fb friends than  72.26% of his friends
mln has more fb friends than  27.1% of his friends
```

Since the calculation done in R even say that Dr. Nelson has less friends than 72.26 percent of his own friends the paradox holds.

```python
#!/usr/bin/env python3
from pygraphml import GraphMLParser

if __name__ == "__main__":

    # well that was easy look what have here a parser!
    # create a new graph parser
    parser = GraphMLParser()
    # get the graph
    g = parser.parse("mln.graphml")

    # set up how we keep track of everything
    friendCounter = {}
    mlnFCount = 0

    # extract the data by simply looping through the data
    for node in g.nodes():
        try:
            print(node['name'], node['friend_count'])
            name = node['name']
            fcount = node['friend_count']
            friendCounter[name] = fcount
            # glorious leader has one more friend
            mlnFCount += 1
        except KeyError:
            print("bad key", node['name'])

    # add out glorious leader
    friendCounter["mln"] = str(mlnFCount)
    print(mlnFCount)

    # write out findings to a file
    with open("mlnfbcount.csv", "w+") as out:
        out.write("friend,fcount\n")
        for fc in friendCounter.items():
            out.write("%s,%s\n" % fc)
```

Listing 1: Parse and Extract Dr. Nelson Facebook graph

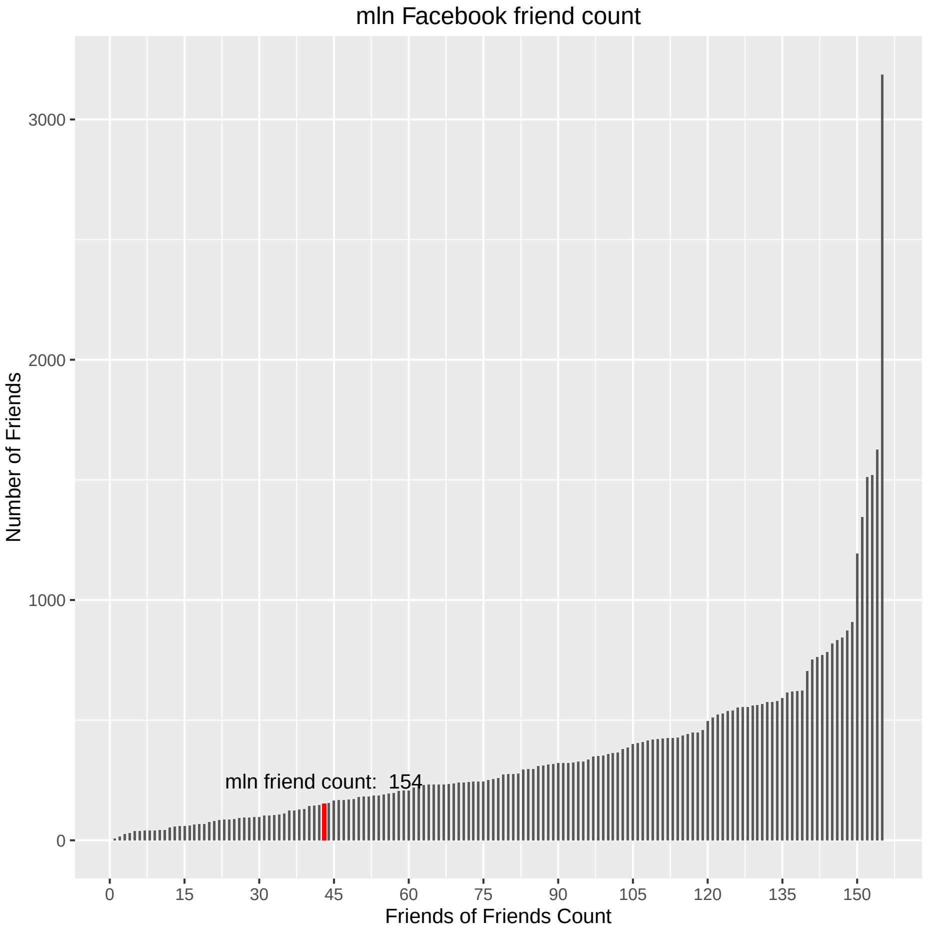

Figure 1: Bar plot showing the count of Dr. Nelson's Facebook Friends' Friends

```r
library(ggplot2)
options(scipen = 9999)
setwd(getwd())


#this function wonderfully borowed from
#http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_%28ggplot2%29/
multiplot <-
  function(..., plotlist = NULL, file, cols = 1, layout = NULL) {
    library(grid)
    # Make a list from the ... arguments and plotlist
    plots <- c(list(...), plotlist)
    numPlots = length(plots)
    # If layout is NULL, then use 'cols' to determine layout
    if (is.null(layout)) {
      # Make the panel
      # ncol: Number of columns of plots
      # nrow: Number of rows needed, calculated from # of cols
      layout <- matrix(seq(1, cols * ceiling(numPlots / cols)),
                       ncol = cols, nrow = ceiling(numPlots / cols))
    }
    if (numPlots == 1) {
      print(plots[[1]])

    } else {
      # Set up the page
      grid.newpage()
      pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

      # Make each plot, in the correct location
      for (i in 1:numPlots) {
        # Get the i,j matrix positions of the regions that contain this subplot
        matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

        print(plots[[i]], vp = viewport(
          layout.pos.row = matchidx$row,
          layout.pos.col = matchidx$col
        ))
      }
    }
  }


  # read the datfile
  data <- read.csv("mlnfbcount.csv")
  #get the friend count
  frinedCount <- sort(data$fcount)
  #find mln
  mln = data[which(data$friend == 'mln'),]$fcount
  numltmln <- with(data,sum(fcount < mln))
  numgtmln <- with(data,sum(fcount > mln))
  totalCount <- length(data$fcount)
```

```
print(paste("mln has less fb friends than ",as.character(round((numgtmln/totalCount)*100,digits = 2)),"% of his friends"))
print(paste("mln has more fb friends than ",as.character(round((numltmln/totalCount)*100,digits = 2)),"% of his friends"))

#create plot dataframe
dplot <-
  data.frame(seq(1, length(frinedCount), by = 1),frinedCount)
#change column names
names(dplot) <- c("fseq","fc")
#remove mln for stats
nomln <- subset(data,friend != "mln")

#find number of friends
numFriends <- length(data$friends)

#do stats
fbcmean <- round(mean(nomln$fcount),digits = 3)
fbcmedian <- round(median(nomln$fcount),digits = 3)
fbcstdev <- round(sd(nomln$fcount),digits = 3)

#inform user
print(paste("mln mean fb friends=",as.character(fbcmean)))
print(paste("mln media fb friends=",as.character(fbcmedian)))
print(paste("mln stdev fb friends=",as.character(fbcstdev)))
print("------------------------------")
# find position for the text annotations
xpos <- median(dplot$fseq)
ypos <- median(dplot$fc)


# do the plot
a <- ggplot(dplot,aes(fseq,fc)) +
  # scale x to see the number of friends mln has
  scale_x_continuous(breaks = seq(
    from = 0,to = max(dplot$fc),by = 15
  )) +
  # plot data first plot regular  data the plot and highlight mln
  geom_bar(
    data = subset(dplot,fc != mln),stat = "identity", width = 0.5, position =
      position_dodge(0.7)
  ) +
  geom_bar(
    data = subset(dplot,fc == mln),fill = "red",stat = "identity", width = 0.9, position =
      position_dodge(0.7)
  ) +
  # add annoations
  geom_text(aes(label = ifelse(fc == mln,paste('mln friend count: ',as.character(mln)),'')),vjust = -1) +
  labs(title = "mln Facebook friend count",x = "Friends of Friends Count",y =
         "Number of Friends")
# save plot to pdf
pdf("mlnFacebookParadox.pdf")
```

```
multiplot(a)
dev.off()
```

Listing 2: R script to generate 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".

Answer

As I do not have the required amount of twitter followers, I resorted to using Dr. Nelson's Twitter followers to answer this question. The python script in listing 3 contains both the code used to generate the number of followers `@phonedude_mln` followers have as well as the followers the people he is following.

To run the script execute it as such:

```
$ chmod +x getTwitterFollowers.py
$ ./getTwitterFollowers.py
```

The python script uses the *Tweepy* library to abstract the communication with the *Twitter* api. The process of getting this information in brief is as such.

1. Set up OAuth. I left out my keys and can easily be replaced with your own by using a config.py file
2. Get an instance of the API
3. Execute methods mlnfollowing and mlnfollowers. Both methods execute as such
   (a) Open a cursor to query the Twitter api for friends or followers
   (b) Get the response and add it to a list
   (c) After all items have been gotten extract the friend or followers name and followers count
   (d) Write results to a file

Mean	1047.01
Median	258
Std Dev	4150.377

Table 2: Statistics for @phonedude_mln Twitter Followers

After consulting the output it was found `@phonedude_mln` has 489 followers. The R script seen in listing 4 was used to generate the graph in figure 2 and the stats seen in table 2. For more details on the process of the R script 4 please consult the comments.

Since `@phonedude_mln` has 489 and is shown in the plot in figure 2 he does not have more followers than his followers. As seen below in the output from running the R script seen in listing 4 Dr. Nelson has 63.67 percent more followers than his followers.

```
phonedude_mln has less twitter followers than  36.12 % of his followers
phonedude_mln has more twitter followers than  63.67 % of his followers
```

From this it is clear to see that the friendship paradox does not hold here.

Please note that the figure generated to answer this question has the y-axis in log10 scale.

```python
#!/usr/bin/env python3
import tweepy
import config


def mlnfollowers(api):
    fs = []
    it = {}
    # get the followers by using a cursor to query the twitter api
    for page in tweepy.Cursor(api.followers, screen_name="phonedude_mln", count=200).pages():
        print(page)
        fs.extend(page)

    # add the followers to out dic
    for pp in fs:
        it[pp.screen_name] = pp.followers_count
        print(pp.screen_name, pp.followers_count)

    # add our glorious leader
    it["phonedude_mln"] = str(len(fs))

    # write it out to a file
    with open("mlntwfollowers.csv", "w+") as out:
        out.write("following,count\n")
        for k, v in it.items():
            out.write("%s,%s\n" % (str(k), str(v)))


def mlnfollowing(api):
    fs = []
    it = {}
    # get the friends by using a cursor to query the twitter api
    for page in tweepy.Cursor(api.friends, screen_name="phonedude_mln", count=200).pages():
        print(page)
        fs.extend(page)

    # add the friends to out dic
    for pp in fs:
        it[pp.screen_name] = pp.followers_count
        print(pp.screen_name, pp.followers_count)

    # add our glorious leader
    it["phonedude_mln"] = str(len(fs))

    # write it out to a file
    with open("mlntwfollowing.csv", "w+") as out:
        out.write("following,count\n")
        for k, v in it.items():
            out.write("%s,%s\n" % (str(k), str(v)))


if __name__ == '__main__':
    # set up oauth
    auth = tweepy.OAuthHandler(config.consumer_key, config.consumer_secret)
```

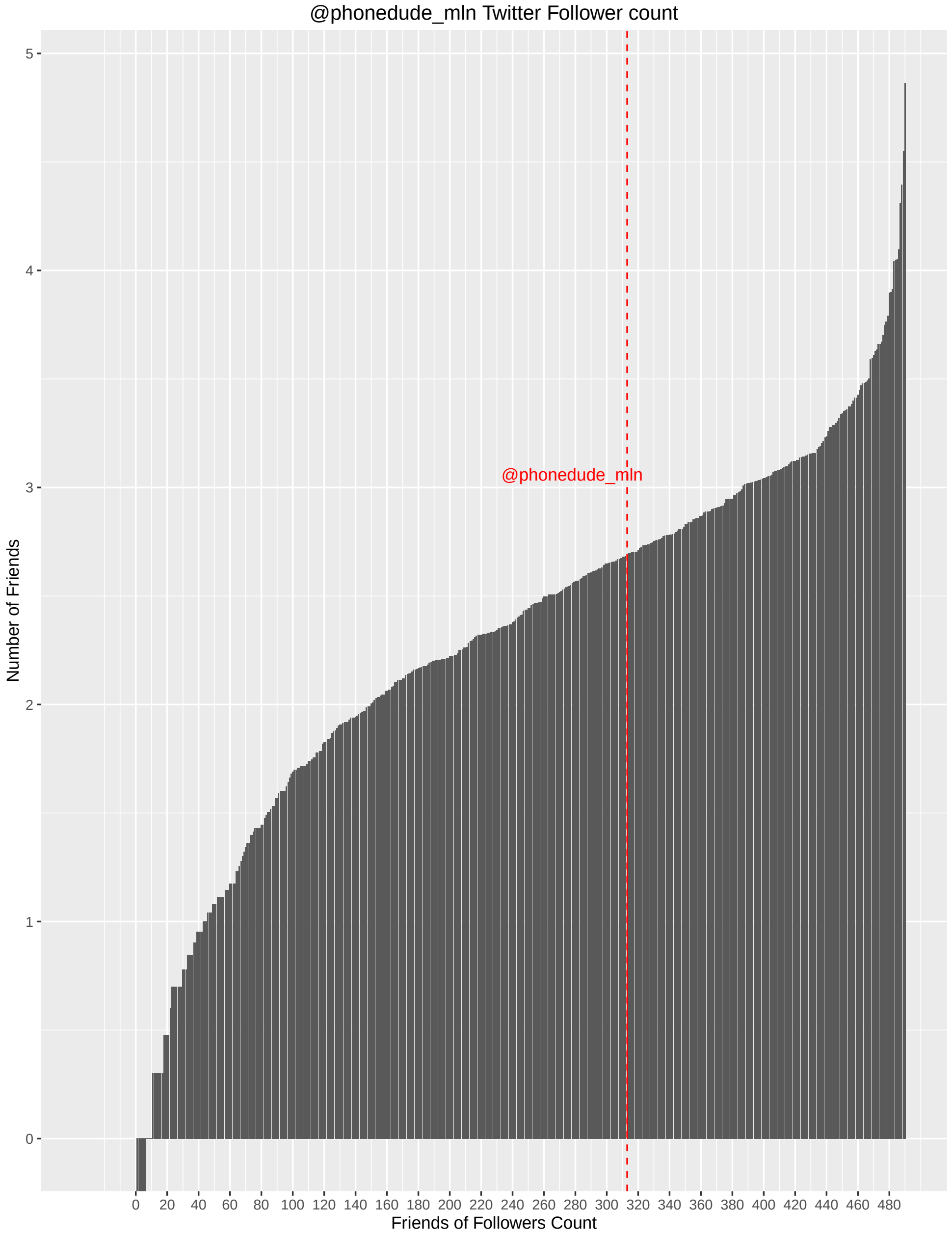

Figure 2: Bar plot showing the count of Dr. Nelson's Twitter Followers Friends

```
    auth.set_access_token(config.access_token, config.access_secret)
    # do not want twitter to slap a rate limit exceeded on me so explicitly wait after each request to avoid that
    api = tweepy.API(auth, wait_on_rate_limit=True, wait_on_rate_limit_notify=True)  # type: tweepy.API

    mlnfollowing(api)

    mlnfollowers(api)
```

Listing 3: Parse and Extract Dr. Nelson Facebook graph

```
library(ggplot2)
options(scipen = 9999)
setwd(getwd())


#this function wonderfully borowed from
#http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_%28ggplot2%29/
multiplot <-
  function(..., plotlist = NULL, file, cols = 1, layout = NULL) {
    library(grid)
    # Make a list from the ... arguments and plotlist
    plots <- c(list(...), plotlist)
    numPlots = length(plots)
    # If layout is NULL, then use 'cols' to determine layout
    if (is.null(layout)) {
      # Make the panel
      # ncol: Number of columns of plots
      # nrow: Number of rows needed, calculated from # of cols
      layout <- matrix(seq(1, cols * ceiling(numPlots / cols)),
                       ncol = cols, nrow = ceiling(numPlots / cols))
    }
    if (numPlots == 1) {
      print(plots[[1]])

    } else {
      # Set up the page
      grid.newpage()
      pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

      # Make each plot, in the correct location
      for (i in 1:numPlots) {
        # Get the i,j matrix positions of the regions that contain this subplot
        matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

        print(plots[[i]], vp = viewport(
          layout.pos.row = matchidx$row,
          layout.pos.col = matchidx$col
        ))
      }
    }
  }


# read the data
data <- read.csv("mlntwfollers.csv")

# i got smarter here

# order the data
data <- data[order(data$count),]

# add friend sequence numbers for x-axis
data$fseq <- seq(1, length(data$count), by = 1)

# change column names
names(data) <- c("follower","fc","fseq")

#find mln
mln = data[which(data$follower== 'phonedude_mln'),]$fc
numltmln <- with(data,sum(fc < mln))
numgtmln <- with(data,sum(fc > mln))
totalCount <- length(data$fc)
print(paste("phonedude_mln has less twitter followers than ",as.character(round((numgtmln/totalCount)*100,digits = 2)),"% of his followers"))
print(paste("phonedude_mln has more twitter followers than ",as.character(round((numltmln/totalCount)*100,digits = 2)),"% of his followers"))

# remove mln
nomln <- subset(data,follower != "phonedude_mln")
# get stats
twitmean <- round(mean(nomln$fc),digits = 3)
twitmedian <- round(median(nomln$fc),digits = 3)
twitstdev <- round(sd(nomln$fc),digits = 3)


data$fc <- log10(data$fc)




# get plot a mln is here and we are plotting less than or equal to mid
# get positon for stats annotation
xpos = median(data$fseq)
ypos = max(data$fc)
# where are you on the x-axis mln ?
mln = data[which(data$follower == 'phonedude_mln'),]$fseq
a<-ggplot(data,aes(fseq,fc)) +
  geom_bar(data = subset(data,follower != "phonedude_mln"),stat = "identity", width =
              0.7, position = position_dodge(0.7)
  ) +
  geom_bar(
    data = subset(data,follower == "phonedude_mln"),fill = "red", stat = "identity", width =
      0.7, position = position_dodge(0.7)
  ) +
  scale_x_continuous(breaks = seq(
    from = 0,to = max(data$fseq),by = 20
  )) +
  # add mln text marker since it is larger than simply mln add some sanity
  geom_text(
    aes(label =
          ifelse(
            follower == "phonedude_mln",'@phonedude_mln',
            ''
          )),vjust = -6,color = "red",nudge_x = -35
  ) +
  # explicitly add line to where mln is
  geom_vline(xintercept = mln,linetype = 2,color = "red") +
  labs(title = "@phonedude_mln Twitter Follower count",x = "Friends of Followers Count",y = "Number of Friends")

pdf("mlnTwitterParadox.pdf")
multiplot(a)
dev.off()


print(paste("mean followers=",as.character(twitmean)))
print(paste("median followers=",as.character(twitmedian)))
print(paste("stdev followers=",as.character(twitstdev)))
```

Listing 4: R script to generate 2

3

Question

```
Extra credit, 2 points:

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

Answer

Not attempted. As I was unable to nicely get the LinkedIn api to generate keys.

Mean	100257.974
Median	748
Std Dev	937488.014

Table 3: Statistics for @phonedude_mln Twitter F

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?
```

Answer

The same python script seen in listing 3 was used to generate this data. At the time when I got this data `@phonedude_mln` had 227 Friends or people he was personally following. The R script in listing 5 is used to produce the stats seen in table 3 and the resulting plot which is also in log10 scale seen in figure 3

Dr. Nelson has a mean number of followers for friends of 748 which is clearly greater than his own number friend at 227. The output from the R script stats:

```
@phonedude_mln has less twitter friends than  70.61 % of his friends
@phonedude_mln has more twitter friends than  28.95 % of his friends
```

So `@phonedude_mln` has less twitter friends than 70.61 percent of his friends thus the paradox holds.

```
library(ggplot2)
options(scipen = 9999)
setwd(getwd())

#this function wonderfully borowed from
#http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_%28ggplot2%29/
multiplot <-
  function(..., plotlist = NULL, file, cols = 1, layout = NULL) {
    library(grid)
    # Make a list from the ... arguments and plotlist
    plots <- c(list(...), plotlist)
    numPlots = length(plots)
    # If layout is NULL, then use 'cols' to determine layout
    if (is.null(layout)) {
      # Make the panel
      # ncol: Number of columns of plots
      # nrow: Number of rows needed, calculated from # of cols
      layout <- matrix(seq(1, cols * ceiling(numPlots / cols)),
                       ncol = cols, nrow = ceiling(numPlots / cols))
    }
    if (numPlots == 1) {
      print(plots[[1]])

    } else {
      # Set up the page
      grid.newpage()
      pushViewport(viewport(layout = grid.layout(nrow(layout), ncol(layout))))

      # Make each plot, in the correct location
      for (i in 1:numPlots) {
        # Get the i,j matrix positions of the regions that contain this subplot
        matchidx <- as.data.frame(which(layout == i, arr.ind = TRUE))

        print(plots[[i]], vp = viewport(
          layout.pos.row = matchidx$row,
          layout.pos.col = matchidx$col
        ))
      }
    }
  }


# same as twitter except for one small change ;)
data <- read.csv("mlntwfollowing.csv")
data <- data[order(data$count),]
data$fseq <- seq(1, length(data$count), by = 1)
names(data) <- c("follower","fc","fseq")

#find mln
mln = data[which(data$follower == 'phonedude_mln'),]$fc
numltmln <- with(data,sum(fc < mln))
numgtmln <- with(data,sum(fc > mln))
```

```r
totalCount <- length(data$fc)

print(paste("@phonedude_mln has less twitter friends than ",as.character(round((numgtmln/totalCount)*100,digits = 2)),"% of his friends"))
print(paste("@phonedude_mln has more twitter friends than ",as.character(round((numltmln/totalCount)*100,digits = 2)),"% of his friends"))

nomln <- subset(data,follower != "phonedude_mln")
twitmean <- round(mean(nomln$fc),digits = 3)
twitmedian <- round(median(nomln$fc),digits = 3)
twitstdev <- round(sd(nomln$fc),digits = 3)
#add log scale so that we can see all the data nicely
data$fc <- log10(data$fc)
xpos = median(data$fseq)
ypos = max(data$fc)
mln = data[which(data$follower == 'phonedude_mln'),]$fseq
ggplot(data,aes(fseq,fc)) +
  geom_bar(
    data = subset(data,follower != "phonedude_mln"),stat = "identity", width =
      0.7, position = position_dodge(0.7)
  ) +
  geom_bar(
    data = subset(data,follower == "phonedude_mln"),fill = "red", stat = "identity", width =
      0.7, position = position_dodge(0.7)
  ) +
  scale_x_continuous(breaks = seq(
    from = 0,to = max(data$fseq),by = 5
  )) +
  geom_text(
    aes(label =
          ifelse(
            follower == "phonedude_mln",'@phonedude_mln',
            ''
          )),vjust = -6,color = "red",nudge_x = -3.5
  ) +
  geom_vline(xintercept = mln,linetype = 2,color = "red") +
  labs(title = "@phonedude_mln Twitter Friends",x = "Followers of Friends Count",y =
         "Number of Friends")
multiplot(a)
print(paste("mean twitter friend followers=",as.character(twitmean)))
print(paste("median twitter friend followers=",as.character(twitmedian)))
print(paste("stdev twitter friend followers=",as.character(twitstdev)))
```

Listing 5: R script to calculate the Friendship Paradox for Twitter Friends

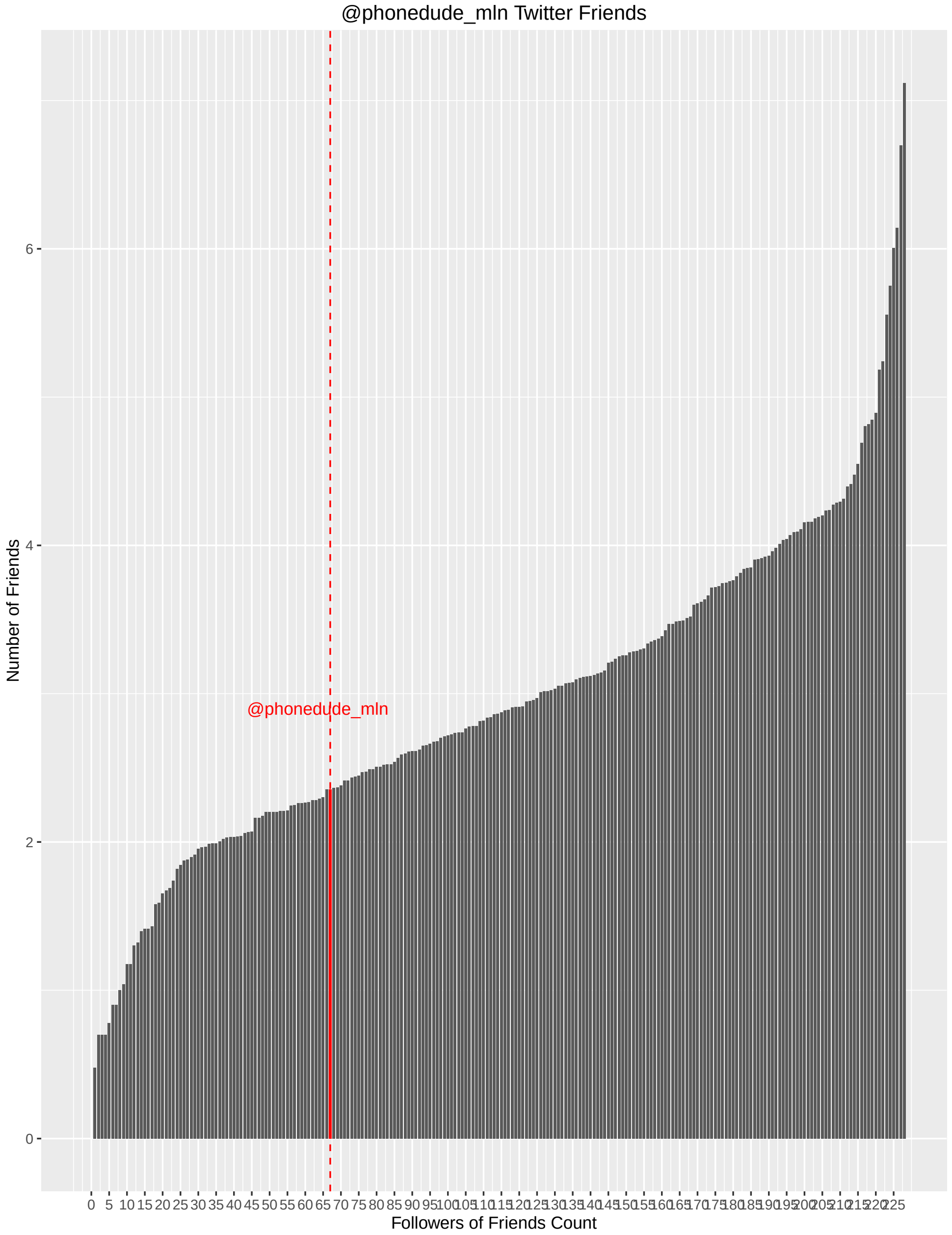

Figure 3: Bar plot showing the count of Dr. Nelson's Twitter Friends Friends

References

[1] Mary, H. Pygraphml. https://github.com/lowks/pygraphml.