

Homework 02 Data Handling, Graphics, More R

Due by 11:59pm, Friday, 2.2.24

S&DS 230/530/ENV 757

(1) **Obama Tweets: Retweets vs. Favorites** A .CSV file containing recent Tweets from former President Barack Obama can be downloaded [HERE](#). The data is sorted by date, most recent at the top.

The variables (columns) are:

- **text**: the body of the tweet
- **date**: when the tweet was sent, original format
- **date2**: when the tweet was sent, JUST the day (not the time of day)
- **retweet_count**: how many people retweeted this tweet
- **favorite_count**: how many people favorited this tweet
- **is_retweet**: whether or not this tweet is a retweet of someone else's tweet
- **source**: device used to send the tweet
- **is_quote**: is the tweet a quote of someone else
- **is_reply**: is the tweet a reply

There are two ways in which other Twitter users can indicate support for a tweet: *favoriting* and *retweeting*. For example, if a tweet has **favorite_count** = 5 and **retweet_count** = 10, then this suggests that 5 people favorited the tweet (saved it) and 10 people retweeted it (broadcasted it to their followers).

(1.1) Insert an R code chunk right below this that imports the data into a dataframe called **recent**. Note that the data is sorted in reverse time order. Get the header names of **recent** to confirm that the data imported correctly. Look at the first few rows of the data and the final few rows of the data. Also get the dimension of **recent**. What is the date range of the tweets? How many tweets does this dataset include?

```
recent <- read.csv("http://reuningscherer.net/S&DS230/data/ObamaTweets.csv")
colnames(recent)
```

```
## [1] "X"          "text"       "date"       "source"
## [5] "is_quote"   "is_retweet" "is_reply"   "favorite_count"
## [9] "retweet_count" "date2"
```

```
head(recent)
```

```
## X
## 1 1
## 2 2
## 3 3
## 4 4
## 5 5
## 6 6
##
## 1
## 2
## 3
## 4
## 5
```

This week, Illinois joined states across the country in passing a historic g

```

## 6 If you haven't already, I hope you'll take some time to watch Descendant on @Netflix. It's an impor
##          date
## 1 2023-01-13 13:30:43
## 2 2023-01-13 13:30:43
## 3 2023-01-12 08:30:25
## 4 2023-01-11 10:45:56
## 5 2023-01-11 09:31:33
## 6 2023-01-10 14:37:04
##
##                                     source
## 1 <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>
## 2 <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>
## 3 <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>
## 4 <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>
## 5 <a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>
## 6      <a href="https://mobile.twitter.com" rel="nofollow">Twitter Web App</a>
##  is_quote is_retweet is_reply favorite_count retweet_count      date2
## 1    FALSE      FALSE      TRUE          4045          847 2023-01-13
## 2    FALSE      FALSE      FALSE         15256         1563 2023-01-13
## 3    FALSE      FALSE      FALSE         28154         3760 2023-01-12
## 4    FALSE      FALSE      FALSE           0          347 2023-01-11
## 5    FALSE      FALSE      FALSE           0          3145 2023-01-11
## 6    FALSE      FALSE      TRUE          8404         1310 2023-01-10
tail(recent)
##          X
## 1995 1995
## 1996 1996
## 1997 1997
## 1998 1998
## 1999 1999
## 2000 2000
##
## 1995                                     It's time for the United States to #LeadOnLeave-show your support if
## 1996                                     Retweet if you believe it's time for the United States to #L
## 1997                                     Speak up for a fair hearing for Judge Merrick Garland:
## 1998                                     This is unprecedented.
## 1999 Add a comment if you agree: American workers shouldn't have to choose between their health and a
## 2000      Working families in America should have the basic security of paid sick leave. #
##          date
## 1995 2016-04-11 10:11:20
## 1996 2016-04-11 08:34:06
## 1997 2016-04-08 14:23:02
## 1998 2016-04-08 11:52:17
## 1999 2016-04-08 10:04:33
## 2000 2016-04-08 08:45:49
##
##                                     source
## 1995 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
## 1996 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
## 1997 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
## 1998 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
## 1999 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
## 2000 <a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>
##  is_quote is_retweet is_reply favorite_count retweet_count      date2
## 1995    FALSE      FALSE      FALSE          1544          378 2016-04-11

```

```
## 1996 FALSE FALSE FALSE 6015 3184 2016-04-11
## 1997 FALSE FALSE FALSE 2271 762 2016-04-08
## 1998 FALSE FALSE FALSE 4388 1890 2016-04-08
## 1999 FALSE FALSE FALSE 3141 724 2016-04-08
## 2000 FALSE FALSE FALSE 7082 1732 2016-04-08
```

```
dim(recent)
```

```
## [1] 2000 10
```

The date range is from 2016-04-08 to 2023-01-13. This dataset has 2000 tweets

(1.2) Create a table that shows how many of the Tweets were quotes (that is, President Obama retweeted someone else's tweet but added additional commentary), and call this object `table1`. Show the results of `table1`. Write a single line that calculates the percent of Tweets that were quotes, rounds this value to two decimal places, multiplies the results by 100, and pastes on a “%” symbol. There should be no space between the number and the “%” symbol. Finally, have the entire line read “?% of Obama’s tweets were quotes”, where ? is the calculated percentage.

```
table1 <- table(recent$is_quote)
print(table1)
```

```
##
## FALSE TRUE
## 1817 183
```

```
paste(round(table1[2]/(sum(table1)), 2)*100, "% of Obama's tweets were quotes")
```

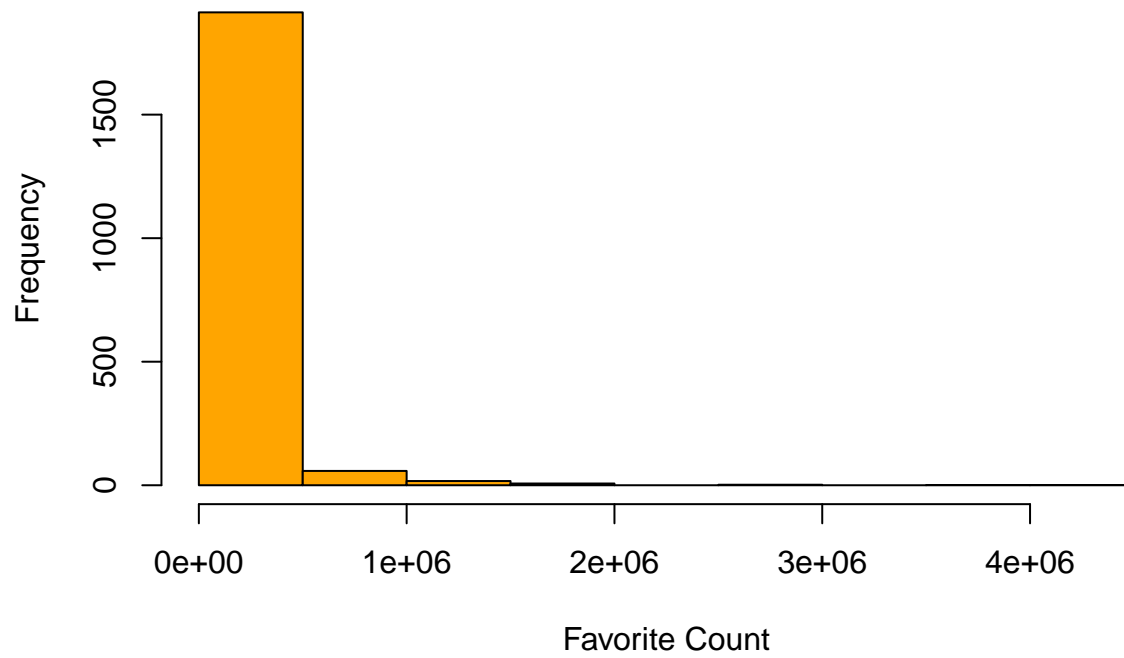
```
## [1] "9 % of Obama's tweets were quotes"
```

(1.3) Get summary statistics for both `favorite_count` and `retweet_count`. Make histograms for each of these two variables as well. Put a title on each histogram, label the horizontal axis, and make the bars orange. How would you describe the shape of these distributions (use words like ‘symmetric’ or ‘skewed’, or perhaps the name of some distribution that has a similar shape . . .)?

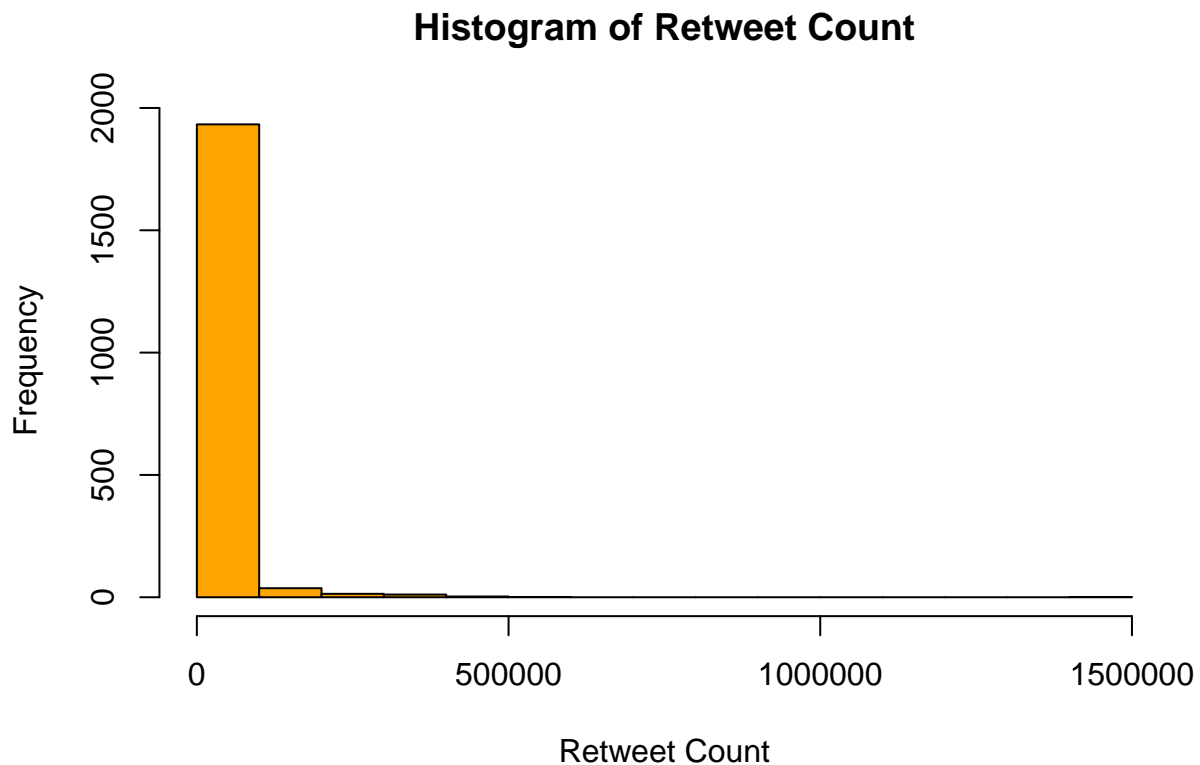
#check this one

```
favoite_summary <- summary(recent$favorite_count)
retweet_summary <- summary(recent$retweet_count)
hist(recent$favorite_count, main = "Histogram of Favorite Count", xlab = "Favorite Count", col = "orange")
```

Histogram of Favorite Count



```
hist(recent$retweet_count,main = "Histogram of Retweet Count",xlab = "Retweet Count",col = "orange")
```



these are both examples of extremely right skewed histograms, which are asymmetrical

(1.4) Get summary statistics for `favorite_count` FIRST for the observations for which `is_quote` is TRUE, then for the observations for which `is_quote` is FALSE. Compare the medians of these two distributions - what do you observe?

```
summary(recent$favorite_count[recent$is_quote == TRUE])
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##         0  14492   58453  114081  164038 1159695
```

```
summary(recent$favorite_count[recent$is_quote == FALSE])
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##         0   3474   13278   94539   69172 4010967
```

The median for the tweets that aren't quotes is 13,278 which is less than the median for the ones that are quotes at 58,453. The quartiles and maximum values indicate that quotes receive more favorites than nonquotes

(1.5) Create a new dataframe called `recent_NoQuote` that contains all data from `recent` for which `is_quote` is FALSE (essentially, we're removing quotes and only looking at strictly original texts). USE THIS NEW DATAFRAME for the remainder of this problem set. Get the dimension of this dataframe to make sure the remaining number of rows (and columns) are consistent with the results in part 1.2.

```
recent_NoQuote <- recent[recent$is_quote == FALSE,]
dim(recent_NoQuote)
```

```
## [1] 1817  10
```

(1.6) Make two new variables as a part of `recent_NoQuote` which will be the log base 10 transformations of `favorite_count` and `retweet_count`. Call these variables `log10favCnt` and `log10reCnt`, respec-

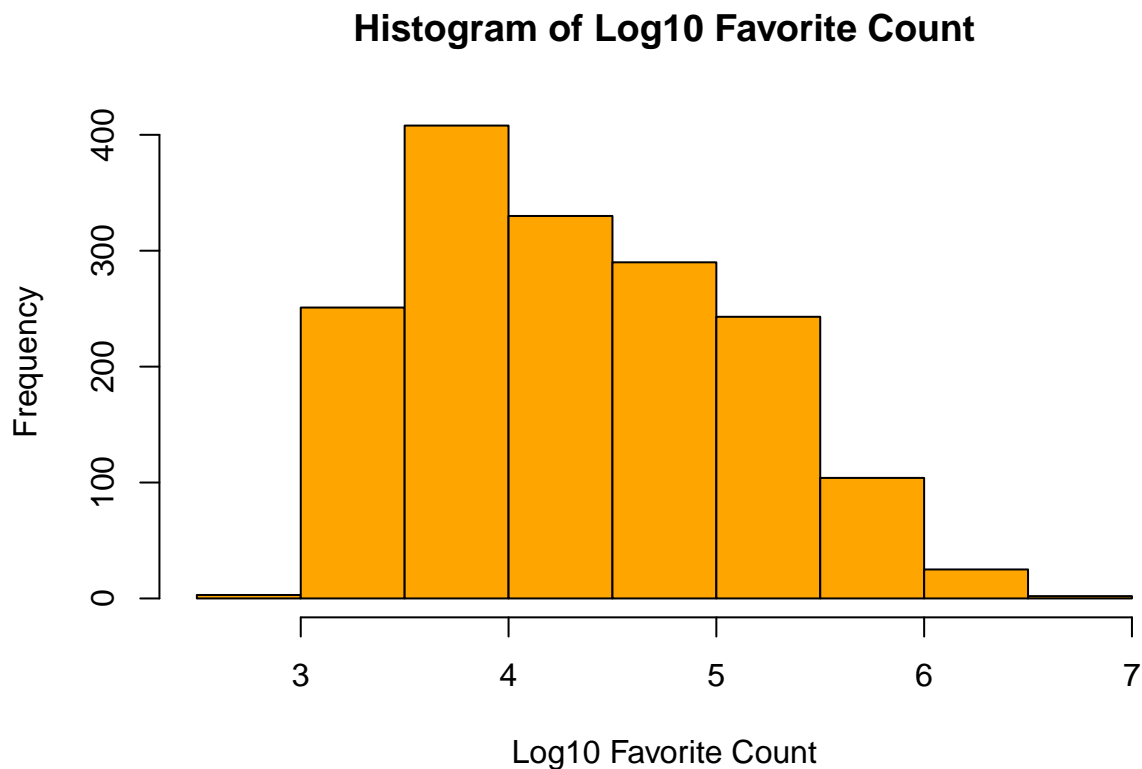
tively. The function you want to take log base 10 is called `log10()`. **Note** - you can add a variable to dataframe by simply creating a name using the `$` operator and then assigning it the desired value : e.g. `recent_NoQuote$log10favCnt <- (whatever you want to assign this)`

```
recent_NoQuote$log10favCnt <- log10(recent_NoQuote$favorite_count)
recent_NoQuote$log10reCnt <- log10(recent_NoQuote$retweet_count)
```

(1.7) Make histograms of these two new log-scale variables. Put a title on each histogram, label the horizontal axis, and make the bars orange. How would you describe the shape of these transformed distributions (use words like 'symmetric' or 'skewed', or perhaps the name of some distribution that has a similar shape . . .)?

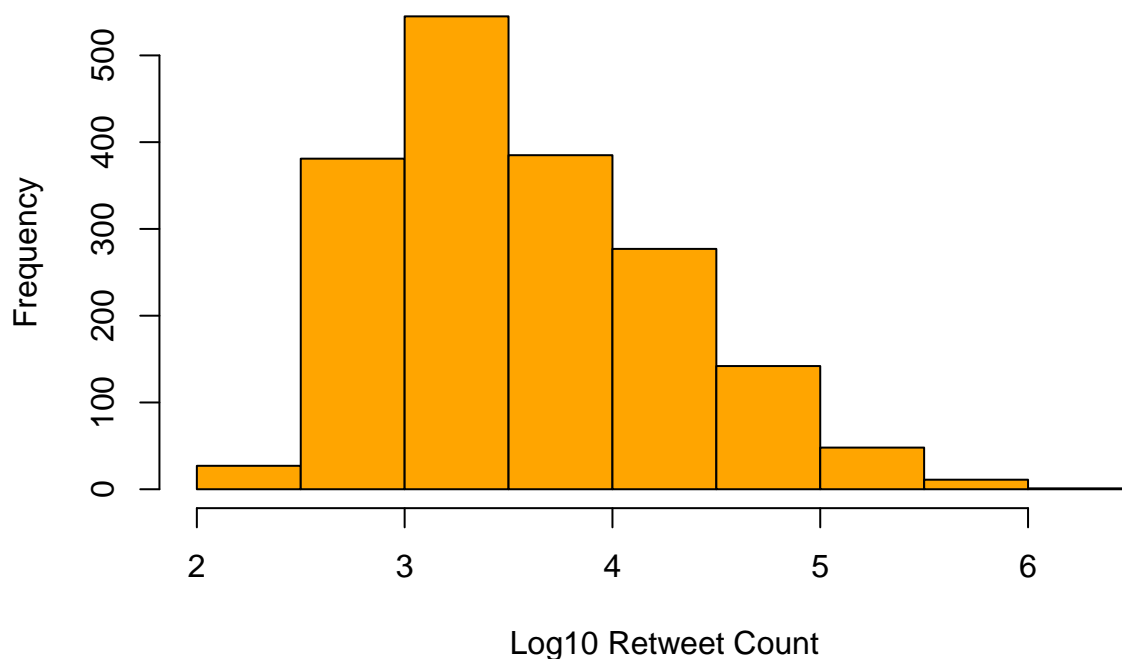
#check this one

```
hist(recent_NoQuote$log10favCnt,main = "Histogram of Log10 Favorite Count",xlab = "Log10 Favorite Count",col = "orange",border = "black",las = 1)
```



```
hist(recent_NoQuote$log10reCnt,main = "Histogram of Log10 Retweet Count",xlab = "Log10 Retweet Count",col = "orange",border = "black",las = 1)
```

Histogram of Log10 Retweet Count

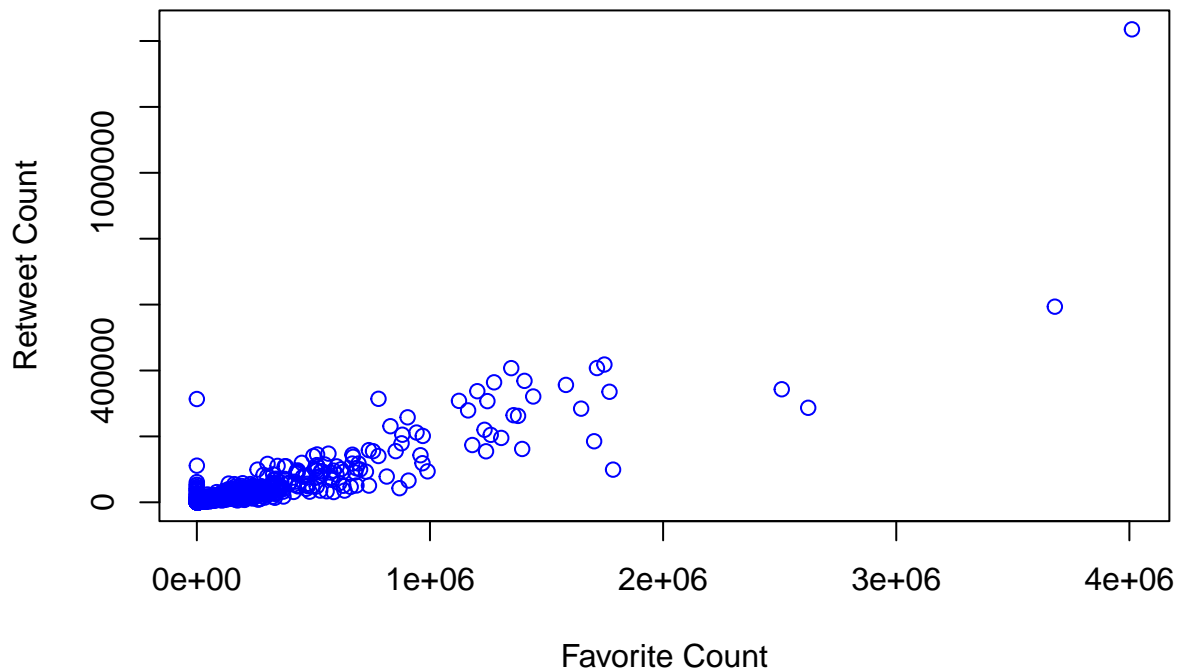


These histograms are both still slightly right skewed, and more normal looking distributions

(1.8) Make a plot of the number of times that each tweet was favorited vs. the number of times a tweet was retweeted. Put `favorite_count` on the x-axis and `retweet_count` on the y-axis. Label your axes, put on a main title, and make the plot characters blue.

```
plot(recent_NoQuote$favorite_count,recent_NoQuote$retweet_count,xlab = "Favorite Count",ylab = "Retweet
```

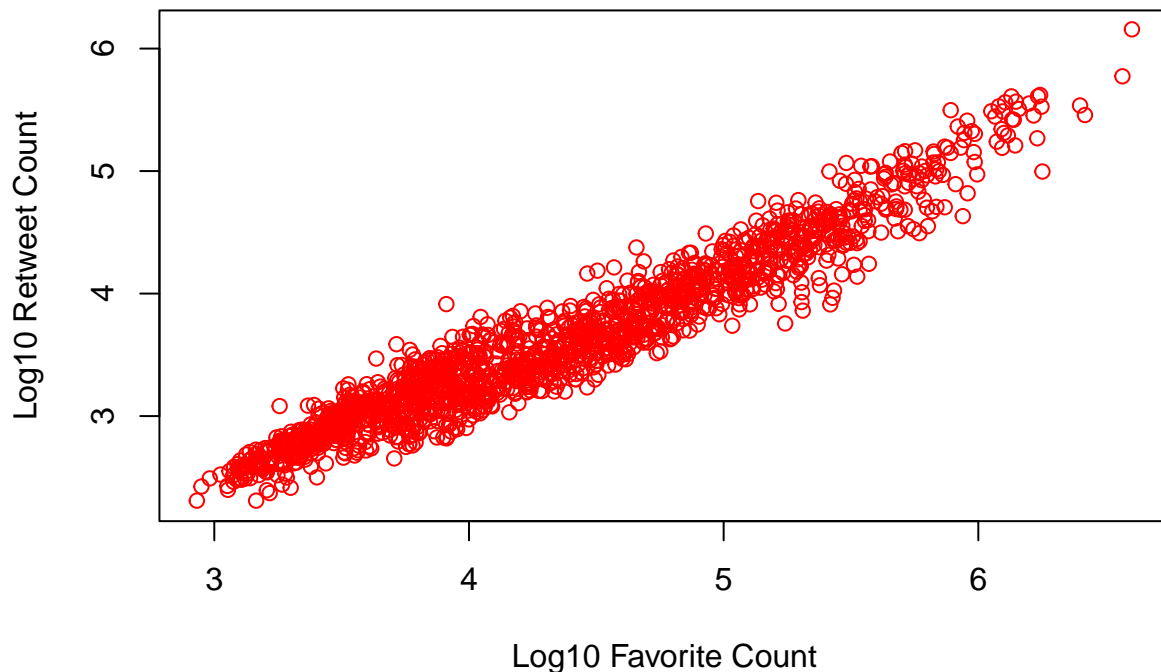
Retweet Count vs. Favorite Count



(1.9) Repeat part (1.8) but use the log-transformed variables. Label your axes, put on a main title, and make the plot characters red. How does the scatterplot on the log-scale compare to the scatterplot on the raw scale? Which one do you prefer?

```
plot(recent_NoQuote$log10favCnt, recent_NoQuote$log10reCnt, xlab = "Log10 Favorite Count", ylab = "Log10 Retweet Count", col = "red")
```


Log10 Retweet Count vs. Log10 Favorite Count



The plot that uses the log transformed variables shows a more linear correlation between the two variables, with a lot less outliers, so I would prefer the log10 plot (1.10) Create two new variables on the `recent_NoQuote` dataframe called `year` and `month` that will contain respectively the year and month the tweet was created. You'll need to look up how to use the function `substr()`. You'll also need to use the `as.numeric()` function to make sure that both new variables are numbers. Show the first 20 observations for each resulting variable.

```
recent_NoQuote$year <- as.numeric(substr(recent_NoQuote$date2,1,4))
head(recent_NoQuote$year, 20)
```

```
## [1] 2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 2023 2022 2022 2022 2022
## [16] 2022 2022 2022 2022 2022
```

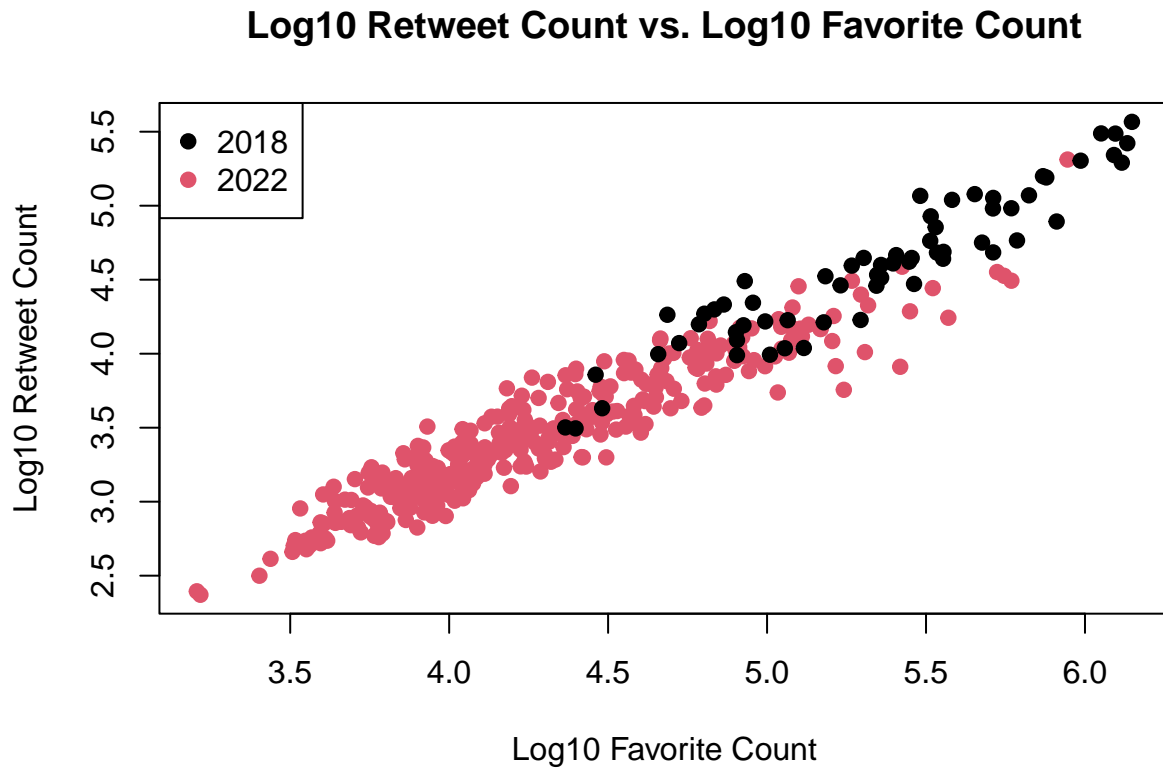
```
recent_NoQuote$month <- as.numeric(substr(recent_NoQuote$date2,6,7))
head(recent_NoQuote$month, 20)
```

```
## [1] 1 1 1 1 1 1 1 1 1 1 1 12 12 12 12 12 12 12 12
```

(1.11) Repeat part (1.9) BUT only for 2018 and 2022 First, create a dataframe called `recent_3` that only has observations from the specified years. You might want to use the `%in%` operator on your newly created variable `year`. Use this new dataframe to make your plot. Use the graphics option `pch = 19` to get solid round points, and use the graphics option `col = as.factor(year)` to make different colors for 2018 and 2022 The final line of code below will add a legend to the top left of the plot.

```
#put R code here to make dataframe and plot
recent_3 <- recent_NoQuote[recent_NoQuote$year %in% c(2018,2022),]
plot(recent_3$log10favCnt, recent_3$log10reCnt, xlab = "Log10 Favorite Count", ylab = "Log10 Retweet Count",
     legend("topleft",
           legend = c("2018", "2022"),
           col = c(1,2),
```

```
pch = 19)
```



(1.12) Write no more than three sentences that describe what you see. Does the pattern appear any different between 2018 and 2022?

The 2018 and 2022 data both have moderately strong positive linear association between favorites and retweets on the log scale. In 2018, there were many more likes and retweets per tweet, and the black dots representing them had a slightly higher linear correlation