Adam Hendel - DS 710
Homework 11
R assignment

1. In this problem, you will use R to do further analysis on the Amazon reviews data.

   a. In the Python assignment for homework 11, you created a .csv file with information about Amazon reviews. **Use scan()** to read this data set into R.

```
d<-scan('/Users/ahendel1/Documents/Academics/ds710fall2017assignment11/finefoods_cleaned.csv',
     sep = ',', what='list', skip = 1)
```

   b. Convert the data into a matrix or data frame in which each row represents one review. Read the header row into R and use it to create column names for the matrix or data frame.

```
# convert to matrix
# known rows based on analysis in python
d <- matrix(d, nrow=568454, ncol = 7, byrow = TRUE)

# assign column names
cols <- scan('/Users/ahendel1/Documents/Academics/ds710fall2017assignment11/finefoods_cleaned.csv',
             sep = ',', what='list', n=7)
colnames(d) <- cols
```

   c. Check whether the columns for total votes, review length, number of exclamation points, and helpful fraction are being treated as numeric vectors. If not, create new variables by converting them into numeric vectors.

   All are character.

```
> # check data types of certain columns
> for(x in c('NumHelp', 'review_num_chars', 'exclams', 'frac_helpful')){
+   print(paste(x, typeof(d[,x]), sep = ': '))
+ }
[1] "NumHelp: character"
[1] "review_num_chars: character"
[1] "exclams: character"
[1] "frac_helpful: character"
>
```

```
votes <- as.numeric(d[, 'NumHelp'])
review.length <- as.numeric(d[, 'review_num_chars'])
num.exclams <- as.numeric(d[, 'exclams'])
frac.helpful <- as.numeric(d[, 'frac_helpful'])
```

d. Examine the helpful fraction vector for unrealistic values. If you find any, set them to missing. Also set to missing the corresponding value of the total votes vector.

Values should be 0 <= x <= 1. Clearly, we see there is at least 1 value greater than one since the max is 3, but no values less than zero.

```
# examine fraction of helpful votes
summary(frac.helpful)
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
 0.00    0.60    1.00    0.78    1.00    3.00  270052
```

Indices 44737 and 64422 are both unrealistic values.

```
> which(frac.helpful>1)
[1] 44737 64422
> frac.helpful[which(frac.helpful>1)]
[1] 1.5 3.0
```

Assign these indices to NA.

```
d[which(frac.helpful>1), c('NumHelp', 'NumReviews', 'frac_helpful')] <- NA
```

Reassign variables and check our work.

```
> votes <- as.numeric(d[, 'NumHelp'])
> review.length <- as.numeric(d[, 'review_num_chars'])
> num.exclams <- as.numeric(d[, 'exclams'])
> frac.helpful <- as.numeric(d[, 'frac_helpful'])
> # examine fraction of helpful votes
> summary(frac.helpful)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.60    1.00    0.78    1.00    1.00  270054
> which(frac.helpful>1)
integer(0)
> frac.helpful[which(frac.helpful>1)]
numeric(0)
```

e. Write 1-2 sentences to document how many unrealistic values you found, what made them unrealistic, and the fact that you set those values to missing.
There were two unrealistic values. Unrealistic values in the fraction would be anything that's greater than 1 or less than 0. These values were set to NA in part D, above.

f. Create a new variable that describes whether more than 50% of people who voted considered it helpful. We will call these helpful reviews.

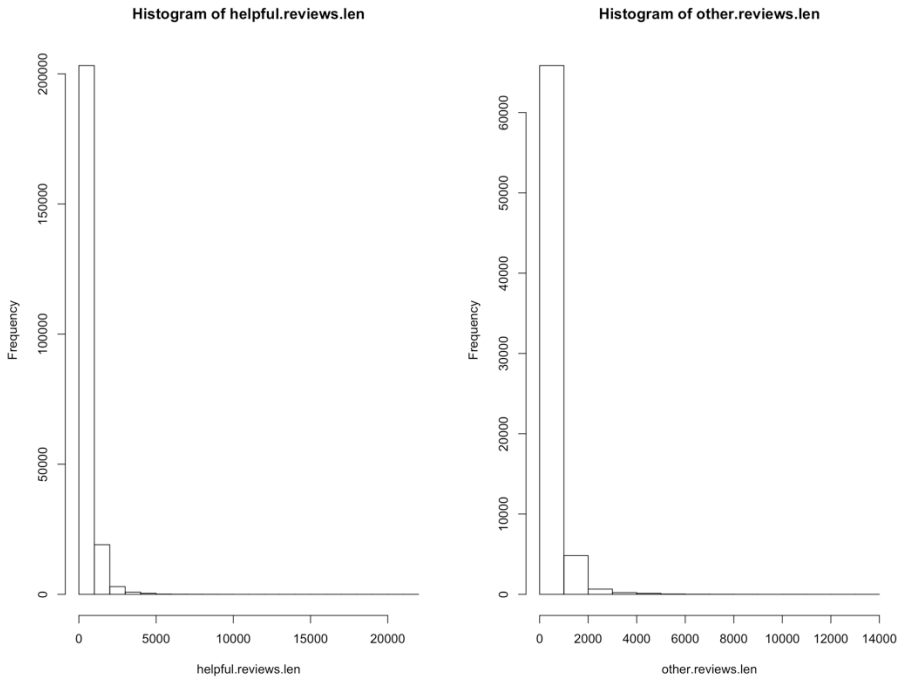
```
helpful.reviews <- frac.helpful[which(frac.helpful > 0.5)]
```

g. Are helpful reviews longer than unhelpful ones? Start by making appropriate graphical summaries to determine whether the review length should be transformed. Then do a hypothesis test of whether the typical length of helpful reviews is longer the typical length of than unhelpful reviews. State your conclusion in context.

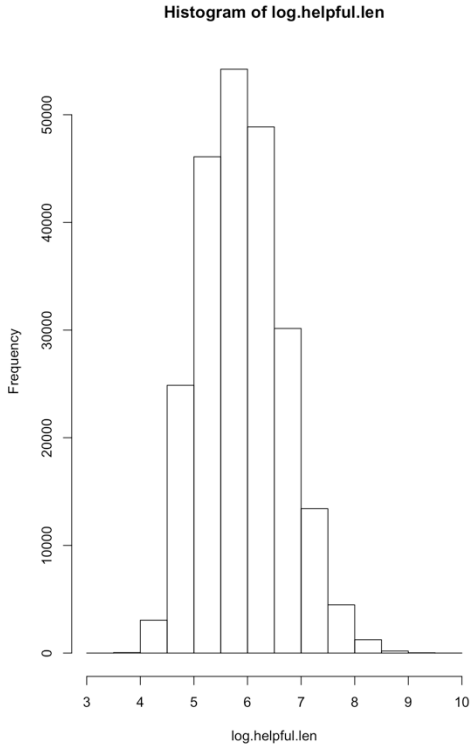
Reviews are right skewed.

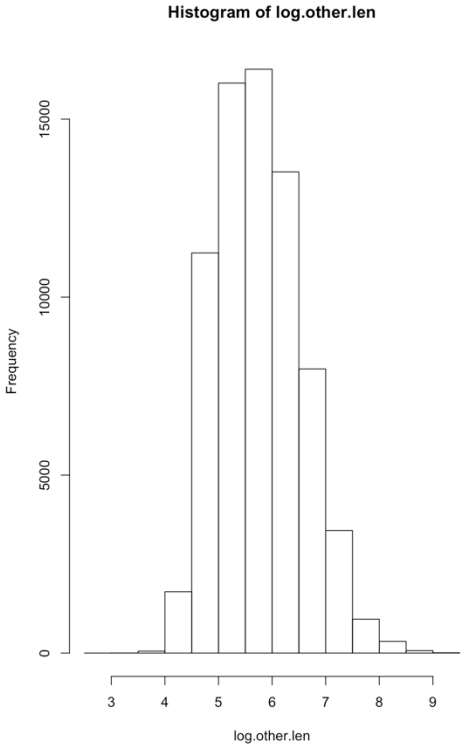
```
helpful.reviews.len <- review.length[which(frac.helpful > 0.5)]
other.reviews.len <- review.length[which(frac.helpful <= 0.5)]
# compare helpful reviews vs. non helpful ones
hist(helpful.reviews.len)
hist(other.reviews.len)
```

Both might benefit from a log transform.

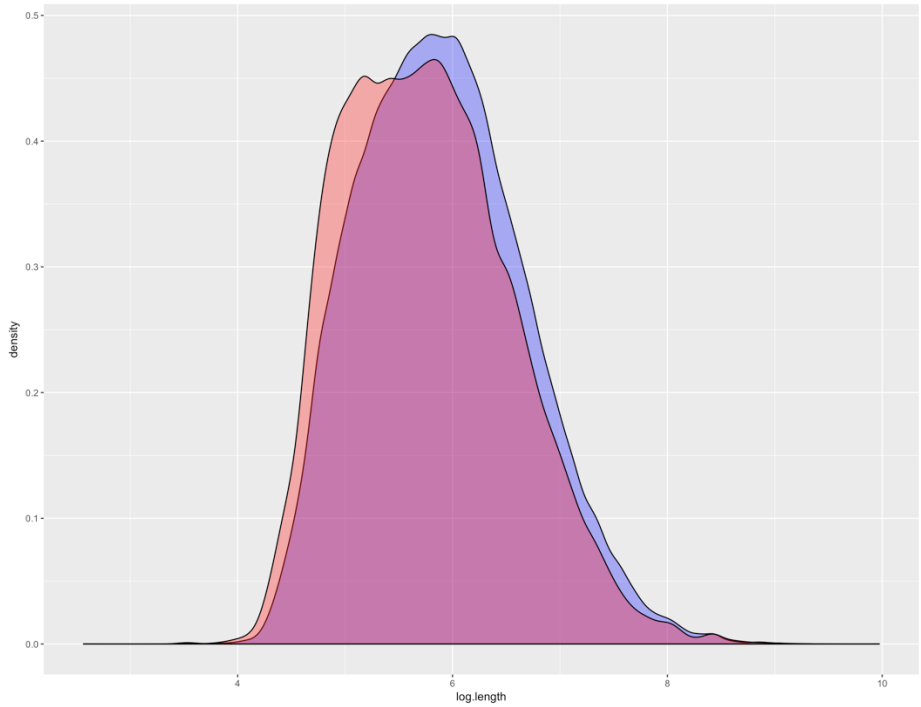


```r
# log transform right skew
log.helpful.len <- log(helpful.reviews.len)
log.other.len <- log(other.reviews.len)
hist(log.helpful.len)
hist(log.other.len)
```





```
library(ggplot2)

#Plot.
ggplot() +
  geom_density(aes(x=log.helpful.len), fill = 'blue', alpha = 0.35) +
  geom_density(aes(x=log.other.len), fill = 'red', alpha =  0.35) +
  xlab('log.length')
```



With a p-value close to zero, we can reject the null hypothesis that the two sample means are equal. The 95th percent confidence interval indicates that the mean log of the length of the helpful reviews is longer than the rest of the reviews.

```
> t.test(log.helpful.len, log.other.len, mu=0, alternative = 'greater')

        Welch Two Sample t-test

data:  log.helpful.len and log.other.len
t = 42.103, df = 119160, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.1343221       Inf
sample estimates:
mean of x mean of y
 5.903569  5.763786
```

h. In parts h-k, you will investigate whether products with more reviews tend to have more votes on their reviews. First, use tapply to find the maximum number of votes received by any of the product's reviews. Then count the number of reviews for each product ID (using tapply or another method you can think of).

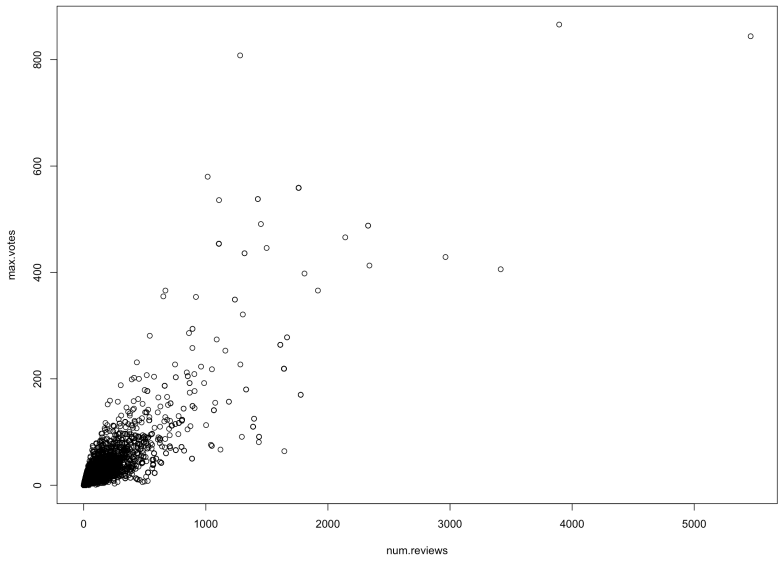
```
# max votes recieved by any of the reviews for a product
max.votes <- tapply(votes, d[,'ProductID'], FUN = max)
max.votes
length(unique(d[,'ProductID'])) == length(max.votes)

# count number of reviews for each product
num.reviews <- tapply(as.numeric(d[,'NumReviews']), d[,'ProductID'], FUN = sum, na.rm = T)
num.reviews
sum(as.numeric(d[,'NumReviews']), na.rm = T) == sum(as.vector(num.reviews), na.rm = T)
```

```
> length(unique(d[,'ProductID'])) == length(max.votes)
[1] TRUE
> sum(as.numeric(d[,'NumReviews']), na.rm = T) == sum(as.vector(num.reviews), na.rm = T)
[1] TRUE
```

i. Make a scatterplot of max number of votes as a function of number of reviews. Is there a visible trend? If so, describe it.

There seems to be somewhat of a linear relationship between max votes and number of reviews when viewing the data as a whole. There is a large cluster of data points with less than 500 reviews and under 100 votes as well. I'd be interested in exploring deeper into the pattern within this cluster.

```
# the the data
plot(num.reviews, max.votes)
```



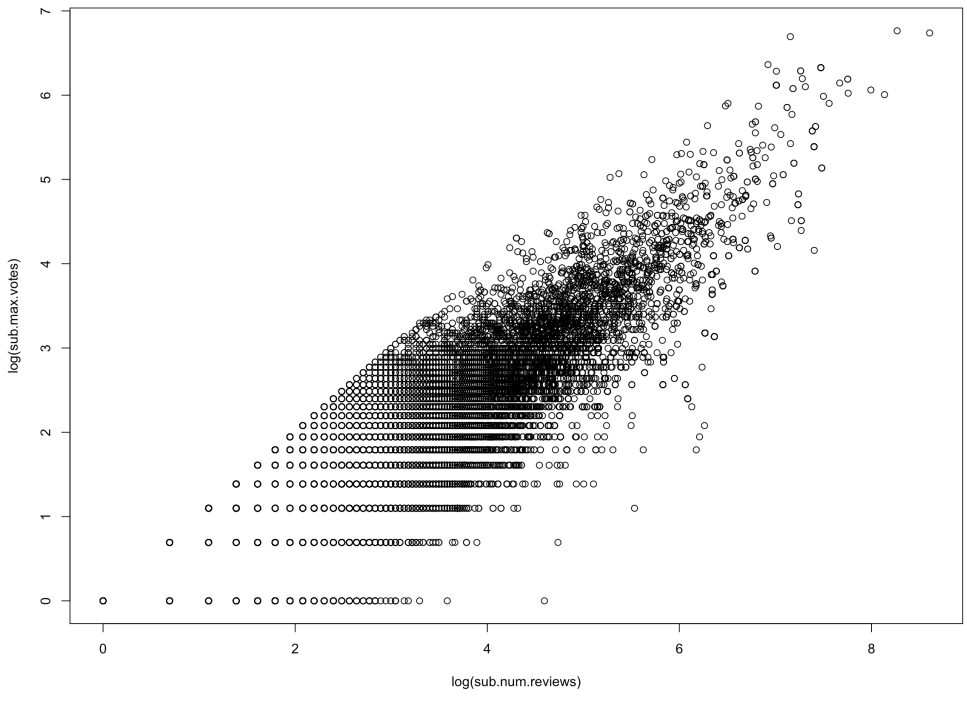
j. Histograms of the review counts and number of votes indicate that both variables are right-skewed. (You can check this for yourself.) So, a log transformation might be helpful in investigating the relationship between them. However, some product IDs had 0 votes, which would result in an error if we tried to take the log. Subset the variables max.votes and number.of.reviews (or whatever you called them) to only those values corresponding to products with 1 or more votes.

We want to exclude the corresponding reviews that have no votes, and corresponding votes that have no reviews (which should not be possible) so we can log transform.

```
# subset variables to exclude zeros
sub.max.votes <- max.votes[which(max.votes>0 & num.reviews>0)]
sub.num.reviews <- num.reviews[which(max.votes>0 & num.reviews>0)]
```

k. Make a scatterplot of log(max.votes) as a function of log(number.of.reviews). Is there a visible trend? If so, describe it. Does this tell us anything about the relationship between the untransformed max.votes and number.of.reviews?

This reinforces my suspicion that the relationship between max.votes and num.reviews is linear. I was curious if there was some sort of exponential growth for a votes from low number of reviews, but this does not appear to be the case. I think the relationship between the log votes and log reviews extends to untransformed simply because we transformed both variables.



Further exploring the untransformed data and fitting a linear model further supports the suspicious of a linear relationship, due to a low p-value and a healthy adjusted R-squared.

```r
# the the data
plot(num.reviews, max.votes)
mod<-lm(max.votes~num.reviews)
abline(mod, col='red', lwd=2)
summary(mod)
```

```
Call:
lm(formula = max.votes ~ num.reviews)

Residuals:
    Min      1Q  Median      3Q     Max
-231.62   -1.17   -0.53    0.47  577.26

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.1680765  0.0315359   37.04   <2e-16 ***
num.reviews 0.1792169  0.0003897  459.93   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.4 on 74254 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.7402,    Adjusted R-squared:  0.7402
F-statistic: 2.115e+05 on 1 and 74254 DF,  p-value: < 2.2e-16
```