Predicting the Number of Shares of Articles

Supriya Shingade
sshingad

Due Wed, March 24, at 8:00PM (Pittburgh time)

Contents


```
library("knitr")
library("cmu202")
library("kableExtra")
library("pander")
library("readr")
library("magrittr")
library("car")
library("jtools")
library("leaps")
```

```
social <- readr::read_csv("http://stat.cmu.edu/~gordonw/social.csv")
```

Introduction

Healthcare is a multi billion dollar industry. Even in the Pittsburgh region, UPMC is the largest employer inAllegheny County. With an aging population and the resulting social as well as economic impact of healthcare in the United States, data analytics are increasingly important in the industry. In the present paper,we focus on patient satisfaction within hospitals, and determine whether there are any specific factors thatcontribute to a patient's satisfaction with their visit. # Exploratory Data Analysis

In this sample we analyzed a random sample of 388 articles and the variables shares, videoes, channel, and the day published. Since our interest is the number of shares, this will be our response variable. Here are the descriptions of the varaibles below `Shares` refers to the number of shares an aricle has in a social network, a quantitative variable. `Images` refers to the number of images in the article, a quantitative variable. `Content` refers to the number of words in the article, a quantitative variable. `Day Published` the day that the arctile was published, a categorical varaible that takes the values Monday-Sunday.

Here is some of the data show below:

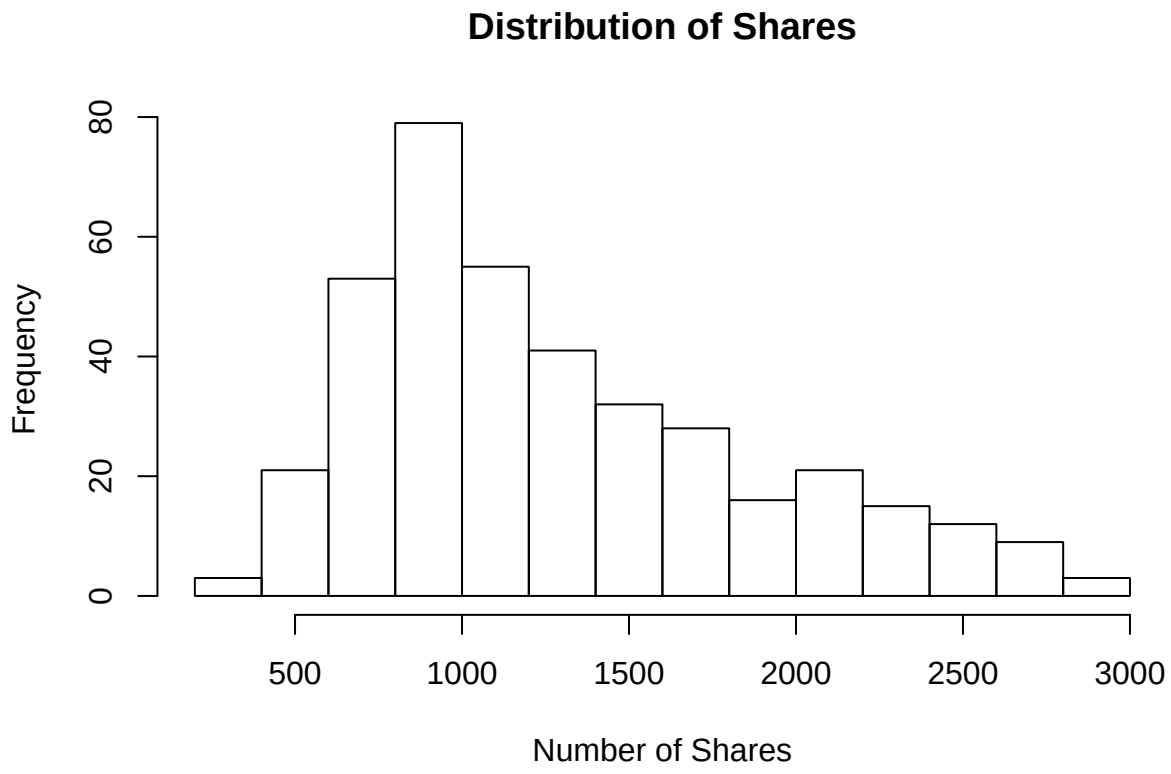
```
head(social)
```

```
## # A tibble: 6 x 4
##   shares content images daypublished
##    <dbl>   <dbl>  <dbl> <chr>
## 1   1100     367      1 Monday
```

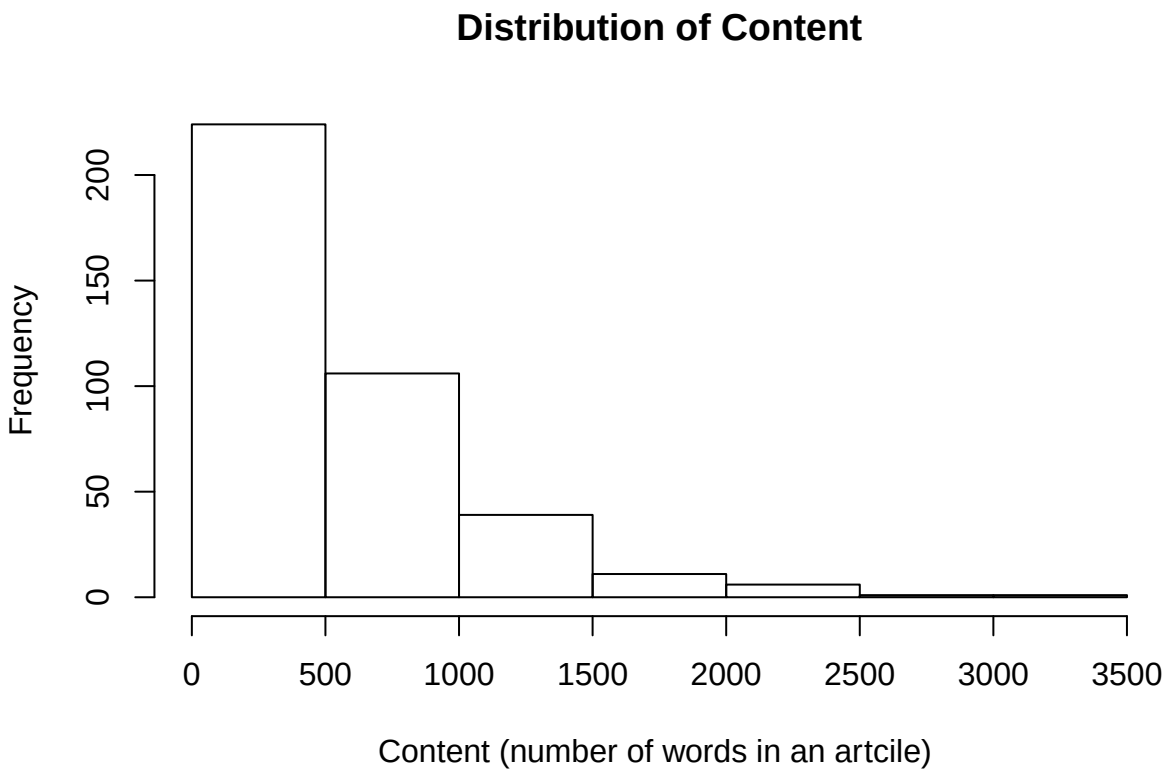
```
## 2   1400     712       1 Monday
## 3    479     291       1 Monday
## 4   2500     463       5 Monday
## 5   1200     498      13 Monday
## 6   1200    1084       1 Monday
```

We will first explore the variables and their distributions. We will check their historgrams to see if we have any skew or outliers in our variables.

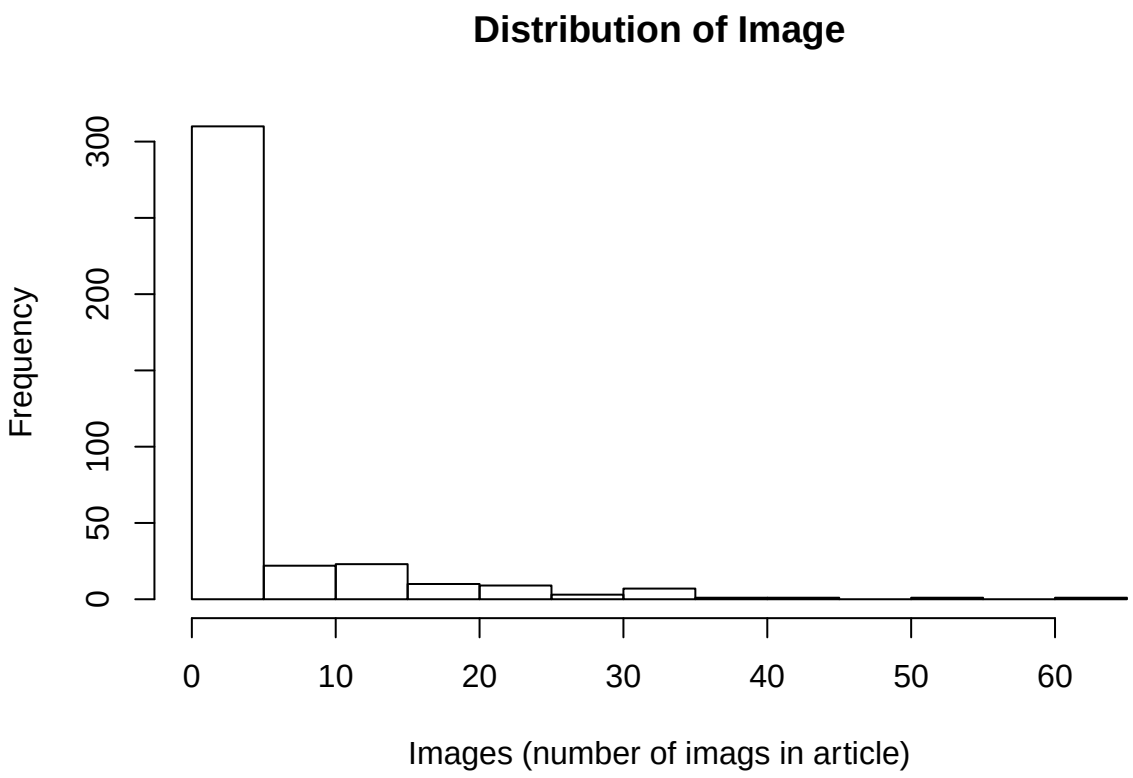
```
hist(social$shares,
     xlab = "Number of Shares",
     main = "Distribution of Shares")
```




```
hist(social$content,
     xlab = "Content (number of words in an artcile) ",
     main = "Distribution of Content")
```



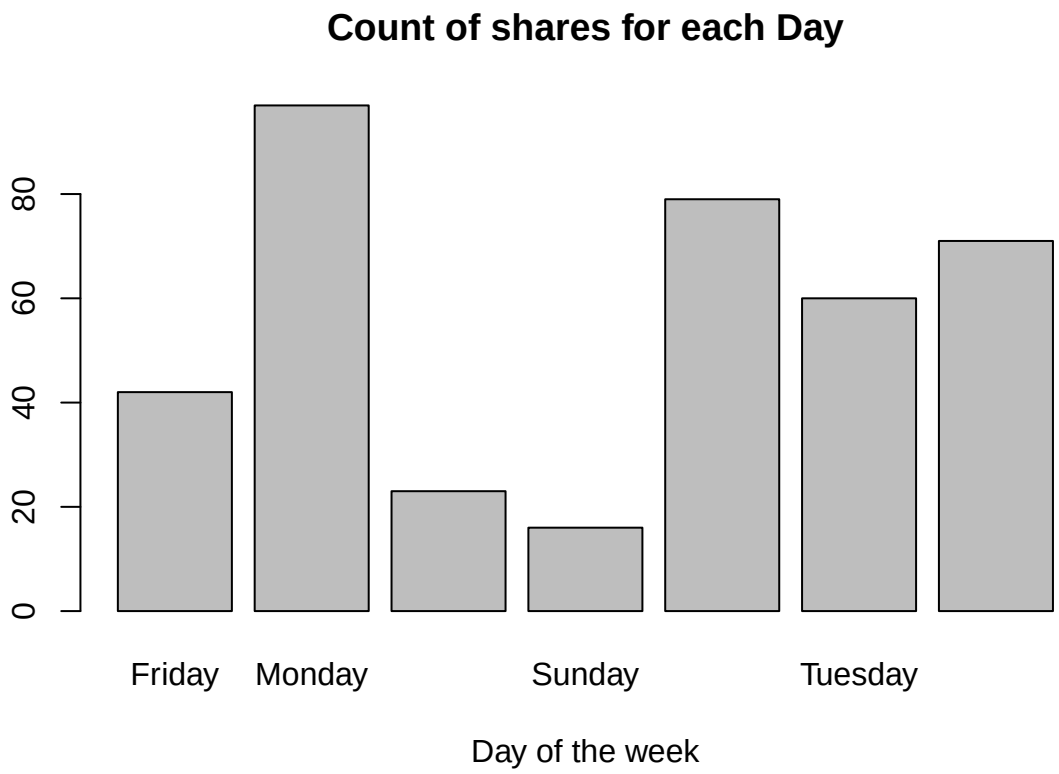

```
hist(social$images,
     xlab = "Images (number of imags in article)",
     main = "Distribution of Image")
```




```
tab <- table(social$daypublished)
names(tab)
```

```
## [1] "Friday"    "Monday"    "Saturday"  "Sunday"    "Thursday"  "Tuesday"
## [7] "Wednesday"
```

```
barplot(tab, main="Count of shares for each Day",
        xlab = "Day of the week")
```



Looking at the bar plot we can see that there is a peak for shares on Monday and Thursday. It seems most of the data has a lot of data from Monday and Thursday.

We will also look at the summaries of these variables:

```
summary(social$shares)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   319.0   859.8  1200.0  1325.1  1700.0  2900.0
```

```
sd(social$shares)
```

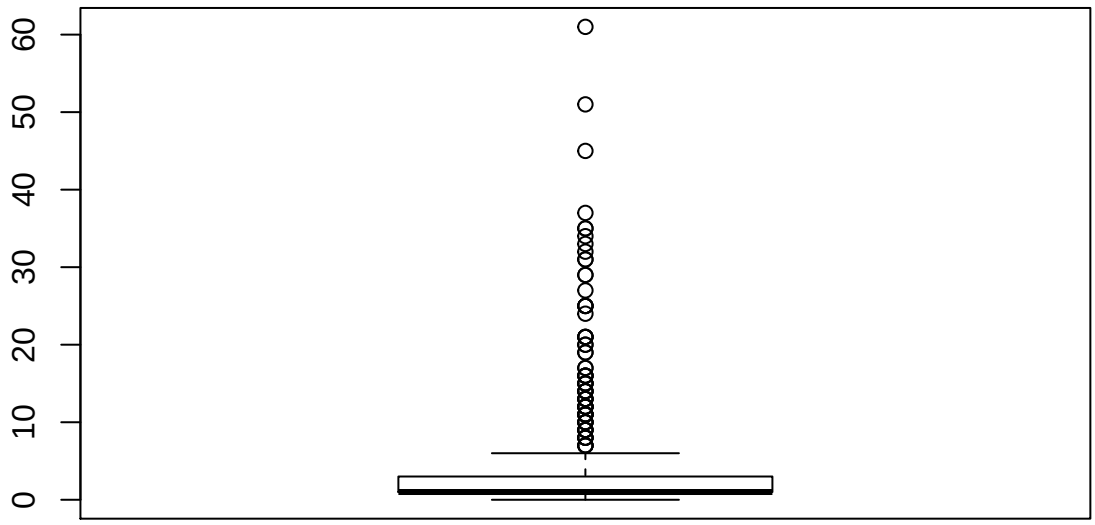
```
## [1] 598.5999
```

The distribution of shares was unimodel, but it was skew a little to the right. The median and mean of the data were not that far from each other however the mean was a more than the median (mean being 1325.1 and the median as 1200), which is also reflected in the fact that the mode of the data is less than the mean (somewhere beteen 800-1000). The standard deviation of shares was 598.6. The distribution of content is skew right, the mean and median are apart from each other. Applying a log transformation will help the shares become more normal.

```
summary(social$content)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   276.5   433.0   586.1   734.2  3174.0
```

```
boxplot(social$images)
```




```
sd(social$content)
```

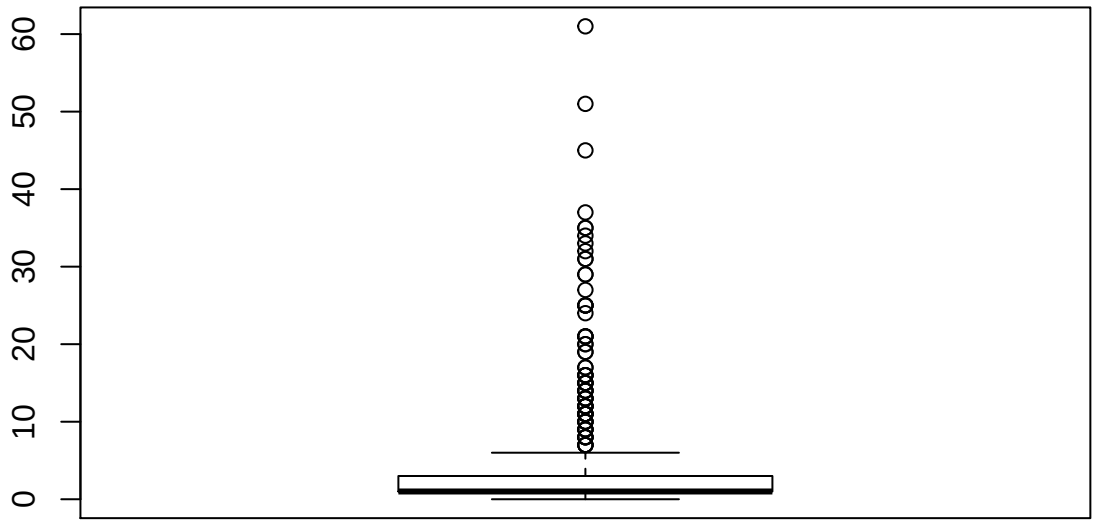
```
## [1] 470.5251
```

The distribution of content is unimodal, but is is skew right. The mean and median of the data are more than 100 words different from each other (where the median is 433 and mean is 586). The most of the data has content from 0 to 500 words which is evidence with the median. The standard deviation of the data is 470.52. The boxplot also suggests that there are a good number of outliers pulling the mean up. We could apply the log transformation to Normalize the data.

```
summary(social$images)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   1.000   4.433   3.000  61.000
```

```
boxplot(social$images)
```




```
sd(social$images)
```

```
## [1] 8.211868
```

The distribution of the images is unimodal, but it is very skew right. This means that most of the data is either 1 or 0 images per article. Looking at the box plot we can see that there are outliers that pull the mean up, to 4.3 but the median is still 1.0. I don't think applying any transformations would help make the data more normal because it is so skewed to the right. We could try log, but it is best to stick with no transformation because the images variable will probably produce the same relationship with shares.

```
summary(social$daypublished)
```

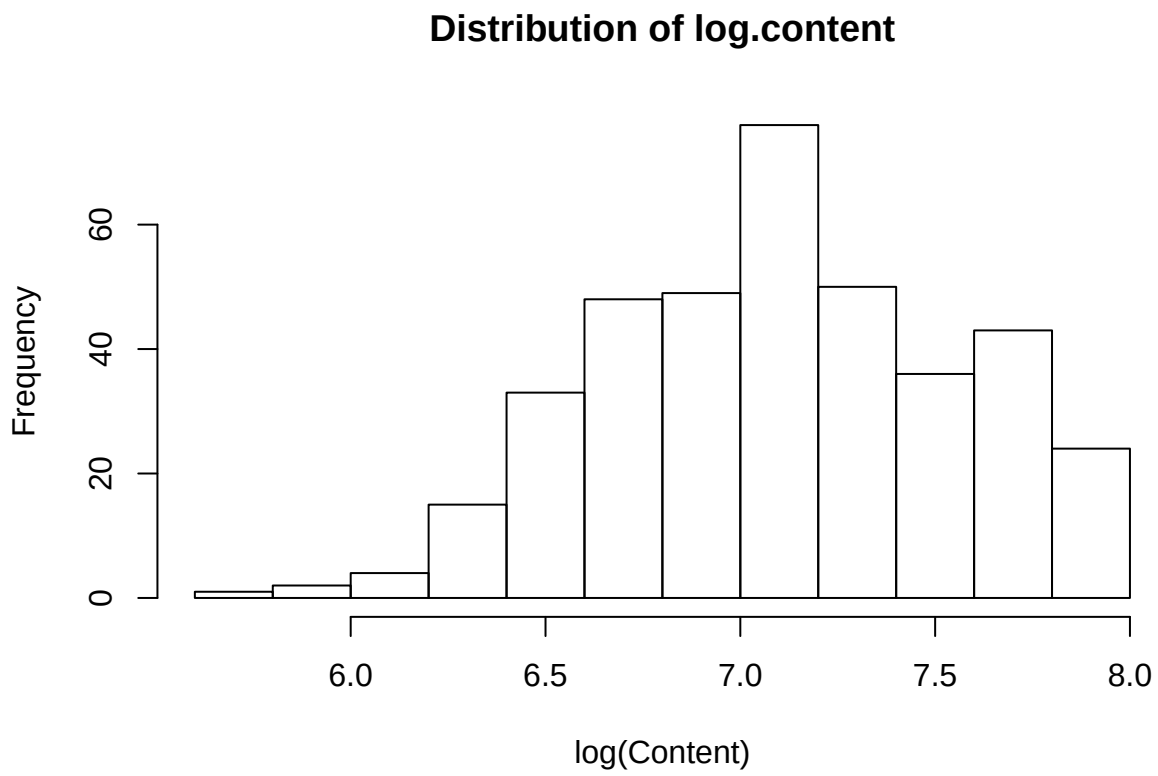
```
##    Length     Class      Mode
##       388 character character
```

Bivariate

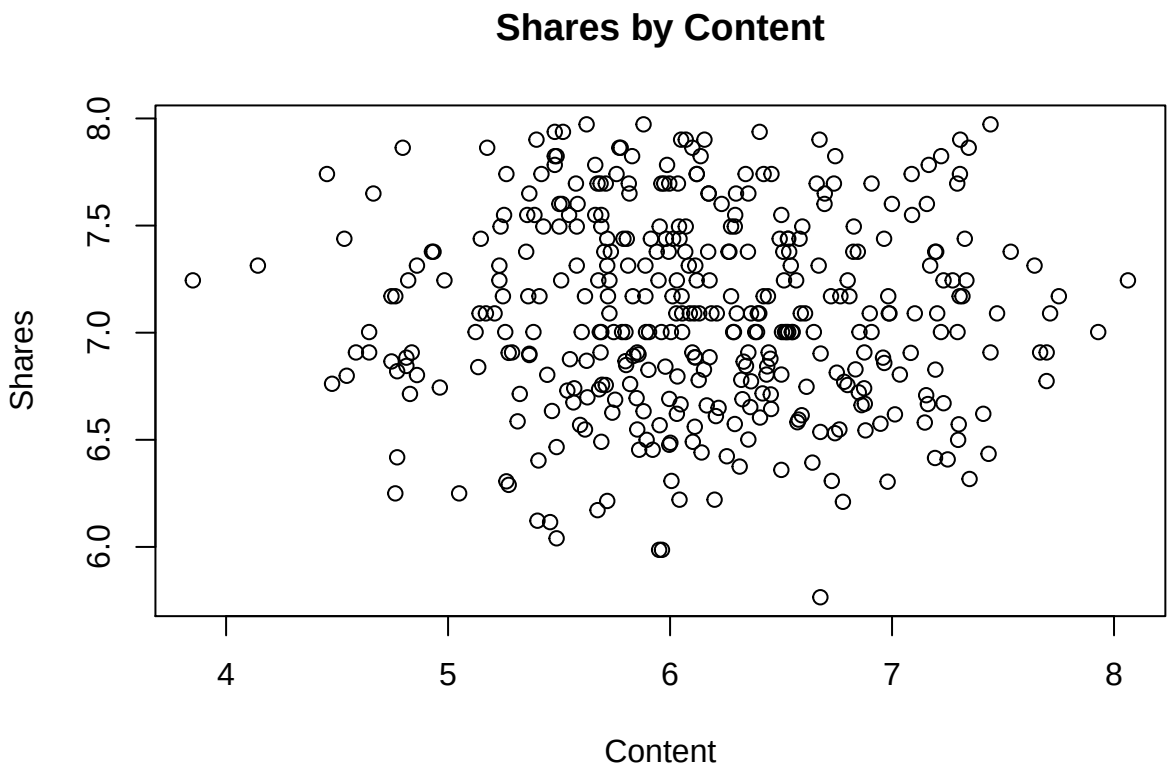
We will first compare the response variable share to the content variable. We will transform the shares to make it more normal. We will see if transforming the content variable by also taking its log will make a difference.

```
social$log.shares <- log(social$shares)
social.sub0 <- subset(social, content >= 1)
social.sub0$log.content <- log(social.sub0$content)
social.sub0$log.shares <- log(social.sub0$shares)
```

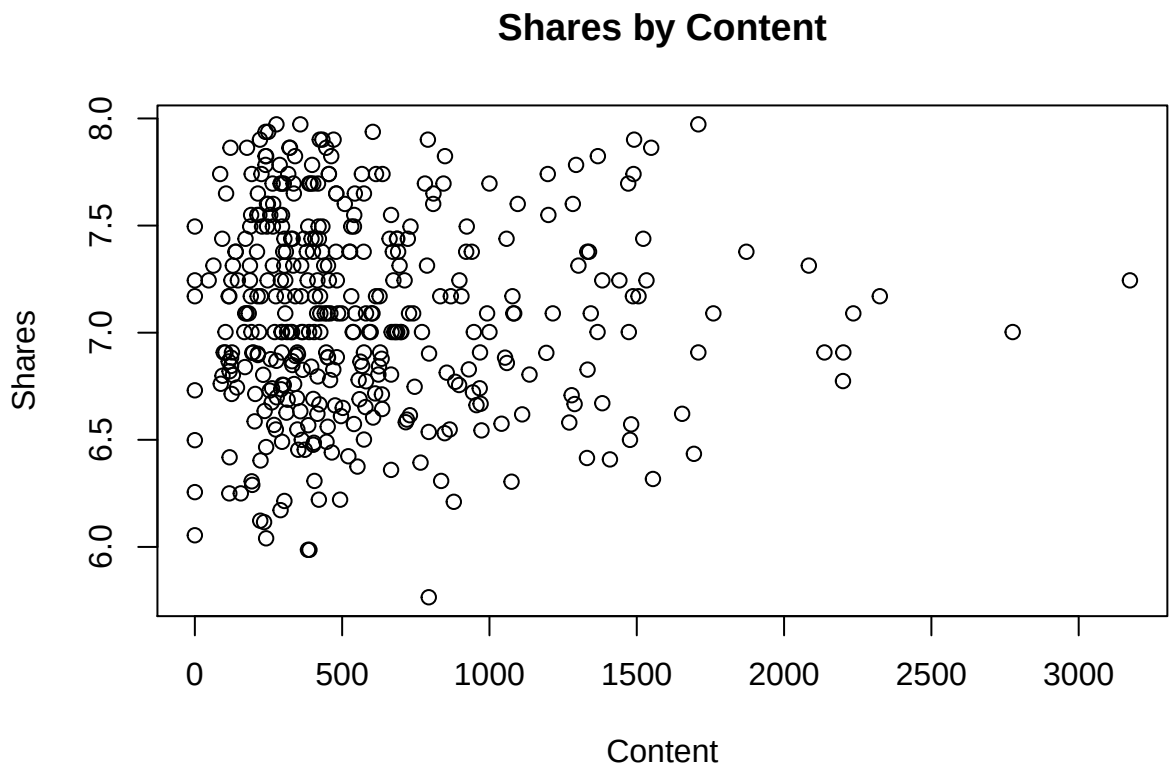
```r
hist(social.sub0$log.shares,
     xlab = "log(Content)",
     main = "Distribution of log.content")
```




```r
plot(social.sub0$log.shares~social.sub0$log.content,
     xlab = "Content",
     ylab = "Shares",
     main = "Shares by Content")
```



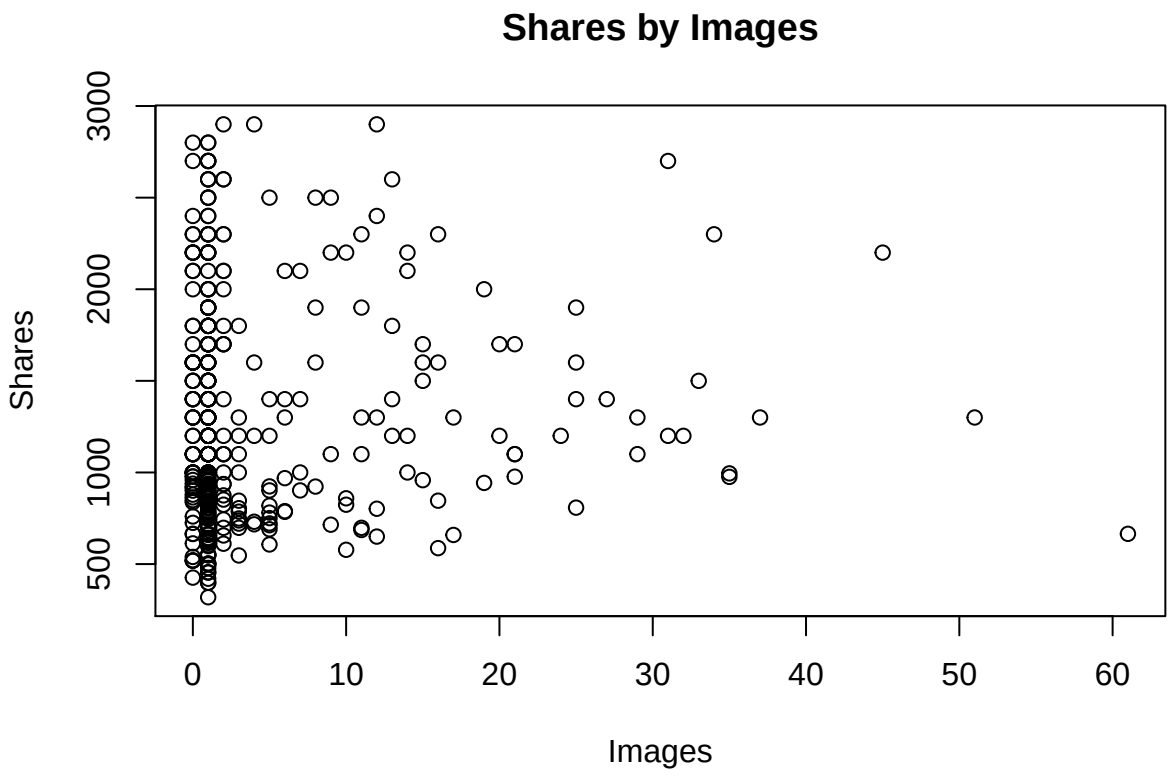

```
plot(social$log.shares~social$content,
     xlab = "Content",
     ylab = "Shares",
     main = "Shares by Content")
```



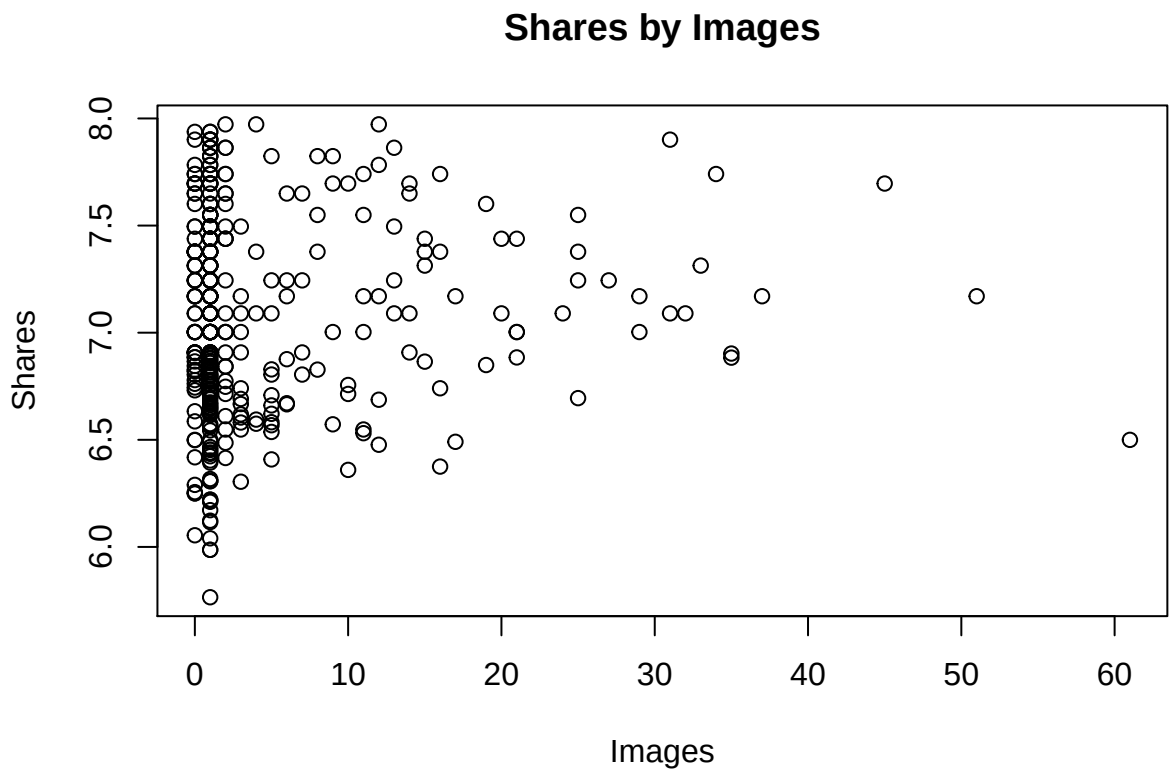
It seems that the data has gotten a little better by transforming the shares. It spread the data out more across the y access. However if we also transformed the content variable it spreads out the points across the x axis however the relationship between the two variables does not seem that clear. With log.shares vs content graph you can see a weak negative, linear relationship. As the content increases, on average we see a very weak trend of the number of shares decreasing.

We will obersve the relationship between images and shares. Since a most of the data for images is 0,1, or possibly 2, the transformation would not have a significant effect on the relationship between images and shares. However to normalize/spread out the shares we can still try a log transformation.

```
plot(social$shares~social$images,
     xlab = "Images",
     ylab = "Shares",
     main = "Shares by Images")
```

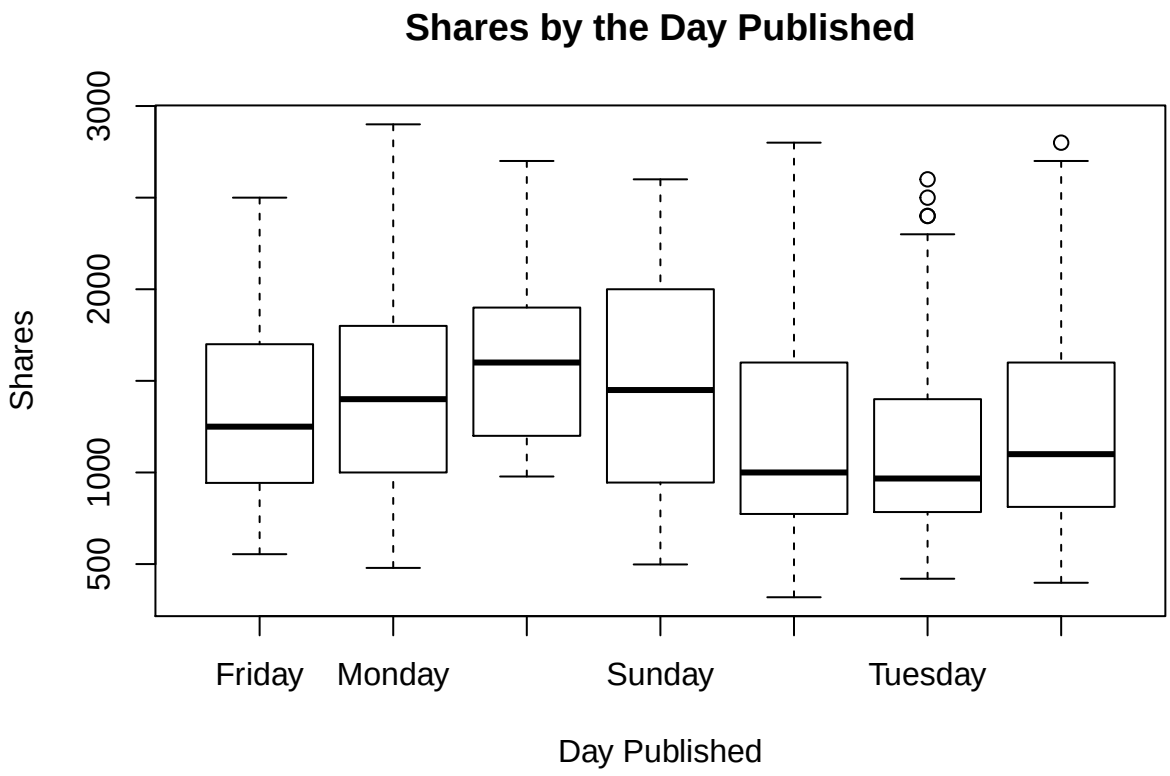



```
plot(social$log.shares~social$images,
     xlab = "Images",
     ylab = "Shares",
     main = "Shares by Images")
```



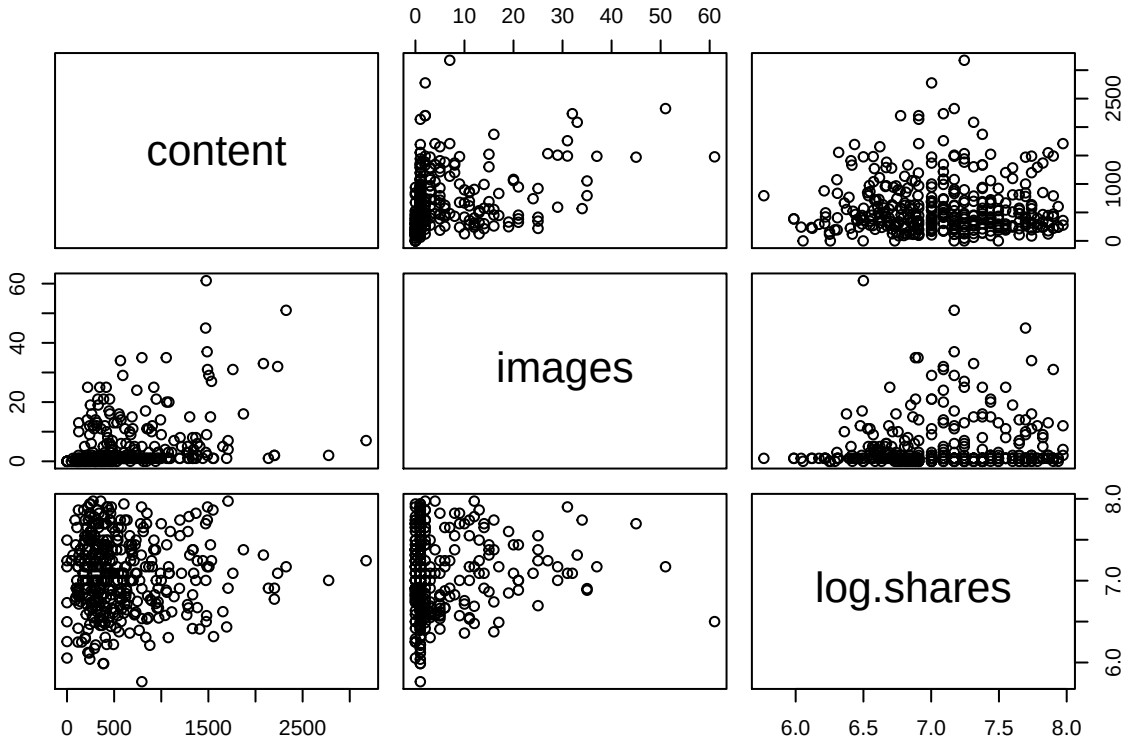
There seems to be a very weak postitive linear relationship however most data points below 5 seem to be randomly scattered.

```
boxplot(social$shares ~ social$daypublished,
        xlab = "Day Published",
        ylab = "Shares",
        main = "Shares by the Day Published")
```



The box plot seems to signify that there is significant differences between the days of the week, the mean number of shares for satuday is higher than the other days. Tuesday seems to have the lowest mean number of shares. Although there is overlap between the boxes, there still could be a relantioship between the number of shares and the day published since boxes like Saturday vs. Tuesday seem to have much less overlap.

```
social.no.daypub <- subset(social,
                           select = -c(daypublished, shares) )
pairs(social.no.daypub)
```




```
round(cor(social.no.daypub),
      digits=2)
```

```
##             content images log.shares
## content        1.00   0.37      -0.01
## images         0.37   1.00       0.06
## log.shares    -0.01   0.06       1.00
```

```
social.sub2 <- subset(social,
                      select = -c(daypublished, content) )
```

There seems to be a stronger linear relationship between images and content than either variable with response so we might run into multicolinearity problems. We can run a vif test to see if we should be include both variables in one model.

```
social.con.im.mod <- lm(log.shares ~ content + images + daypublished,
                        data = social)
car::vif(social.con.im.mod)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## content       1.172911  1        1.083010
## images        1.187833  1        1.089878
## daypublished 1.031635  6        1.002599
```

There don't seem to be any multicolinearity issues so this model seems safe to go with, we can have a safe model with both the predictors.

Modeling

I first tried a model with all the variables with no transformations. When looking at the qq plots I realized that we need to apply a transformation on shares, similar to what we did when we were doing our bivariate analysis because the tails of the qq plot were big and strayed far from the line. The log of the shares made the residual of the errors more normal, as it decreased the tails and most of the data is on the line. However we were not getting any significant pvalues. I tried adding some interaction terms to see if there is any significant interaction between daypublished and content or images. When building that model I found that there is, infact, significant interaction between the content and daypublished as the pvalue for wednesday and content:wednesday were lower than 0.05. I also tried adding the images variable to the regression model and found that the multiple R^2 values increases, even though it doesn't have a pvalue lower than 0.05 it is worth having in th model because the R^2 value increases.

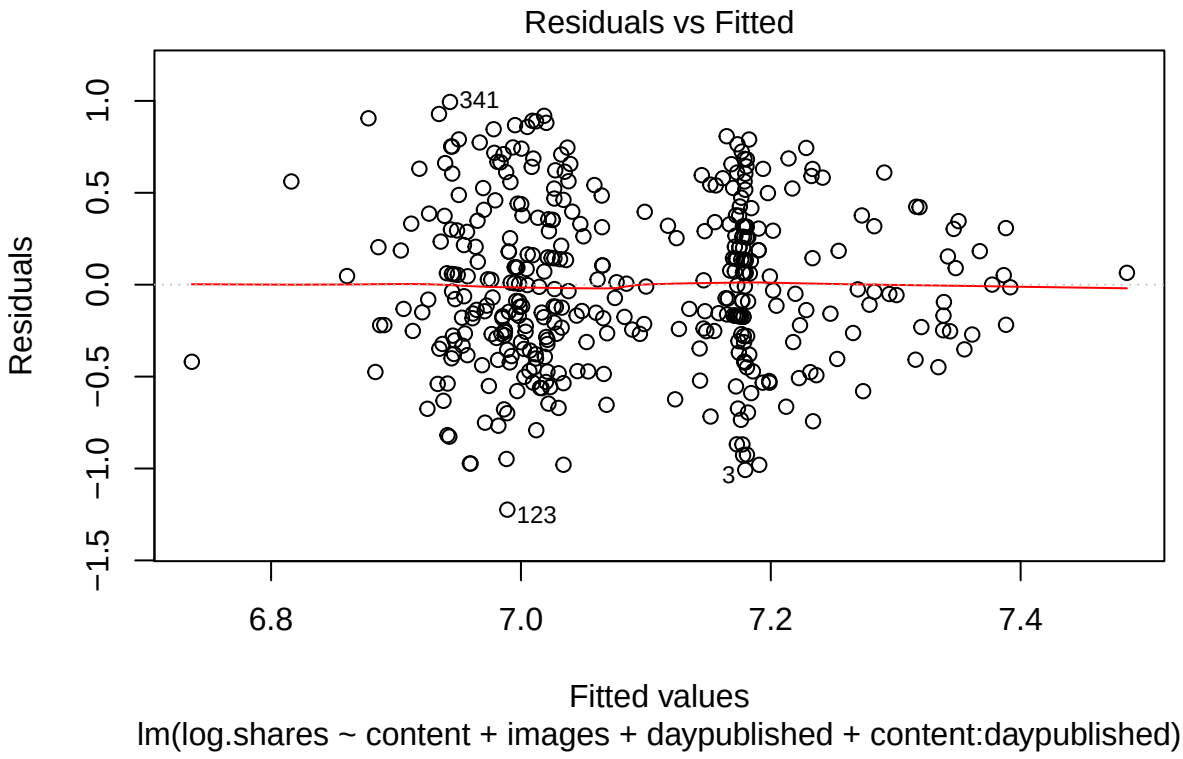
```
social.mod.1 <- lm(log.shares ~ content+ images + daypublished+ content:daypublished ,
                   data = social)
summary(social.mod.1)
```

```
##
## Call:
## lm(formula = log.shares ~ content + images + daypublished + content:daypublished,
##     data = social)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.22389 -0.28575 -0.02861  0.31652  0.99419
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    7.2875760  0.1260766  57.803   <2e-16 ***
## content                       -0.0003572  0.0002081  -1.716   0.0869 .
## images                         0.0044120  0.0030717   1.436   0.1517
## daypublishedMonday            -0.1066694  0.1448176  -0.737   0.4618
## daypublishedSaturday           0.1210846  0.1950857   0.621   0.5352
## daypublishedSunday            -0.1572302  0.2640220  -0.596   0.5519
## daypublishedThursday          -0.2535236  0.1488585  -1.703   0.0894 .
## daypublishedTuesday           -0.2781202  0.1559735  -1.783   0.0754 .
## daypublishedWednesday         -0.3754106  0.1516792  -2.475   0.0138 *
## content:daypublishedMonday     0.0003374  0.0002295   1.470   0.1424
## content:daypublishedSaturday   0.0002027  0.0002720   0.745   0.4565
## content:daypublishedSunday     0.0004209  0.0003612   1.165   0.2446
## content:daypublishedThursday   0.0002950  0.0002347   1.257   0.2095
## content:daypublishedTuesday    0.0002521  0.0002421   1.042   0.2983
## content:daypublishedWednesday  0.0004681  0.0002327   2.012   0.0450 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4487 on 373 degrees of freedom
## Multiple R-squared:  0.07198,    Adjusted R-squared:  0.03715
## F-statistic: 2.067 on 14 and 373 DF,  p-value: 0.01295
```

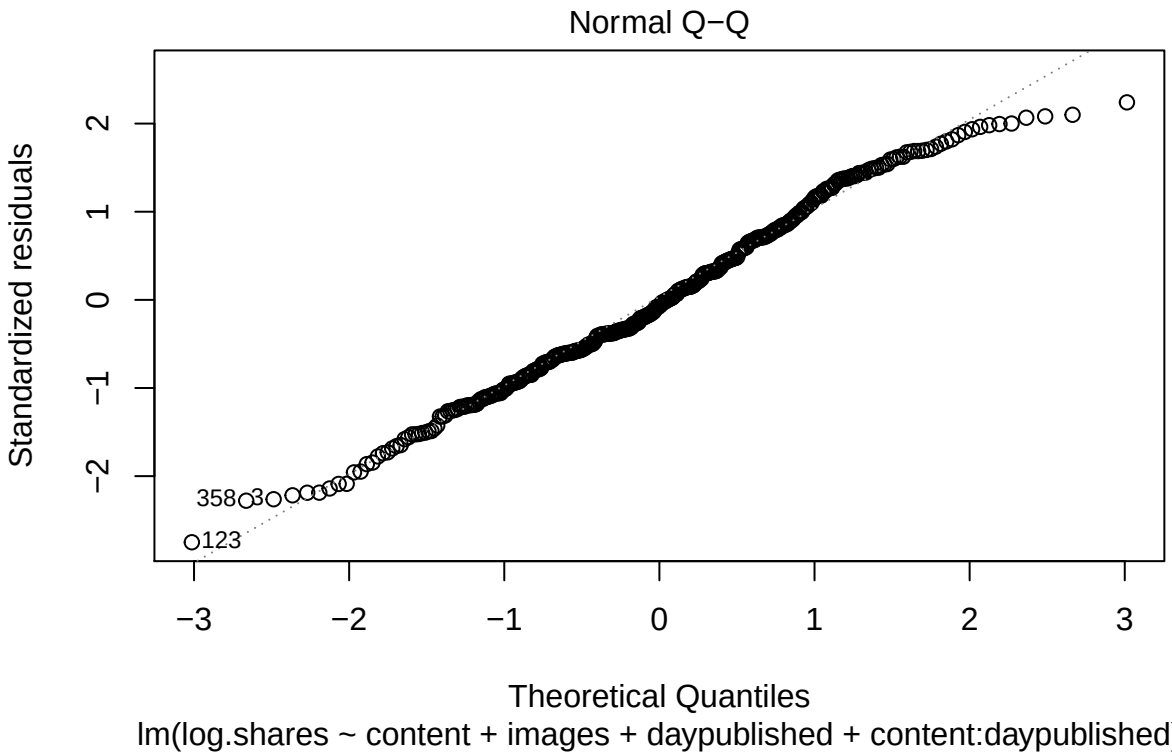
We can now obeserve the residual plot of the data. The residuals have a constant spread, indepedence, and a mean of zero, which validates the assumptions, becdause the residuals seem about equal above and below spread from the zero line and there doesn't seem to be any obvious patterns. I do acknwoledge that there is a cluster of data at 7.0 and 7.18-9 but this doesn't seem to lend to any distinct pattern. There are also a couple ouliers on the bottom, but they seem dismissible because there does not seem to be any pattern

arising from them either. Looking at the qq plot we can see that the data is about normal, except for the tails of the graph, and the outliers on the bottom. However most of the data falls on the line. These were the best results we could get out of all the other models that we tried.

```
plot(social.mod.1, which =1)
```




```
plot(social.mod.1, which =2)
```



This model seems to have the best results. It has the lowest p value for the f statistic (0.01295) and also the highest multiple r^2 value (0.07198). We can see that content and shares are negatively related by the negative coefficient as well as images and shares are postively related by the postive coefficient, as we thought from the EDA. We can see that Satudray has a positive coefficient, which was what our EDA reflected as we found it to have the highest mean number of shares. We also see Wednesday have a negative coefficient which was also something we observed, as it had one of the lowest mean of shares.

Since out reponse variable is transformed, it is important to note that our predictors can help explain the transformed reponse so in this case low content, higher number of pictures, and saturday seem to be associated with higher number of log shares (and in turn higher number of shares).

Prediction

We have a reasonably valid model that satisfies the assumptions. We can use this model to predict the number of shares of an article with 627 words, three images, and was published on Saturday: log.shares = 7.2875760 + 0.1210846 + (-0.0003572+0.0002027)*content + 0.0044120* images E^(log.shares) = shares

```
exp(7.2875760 + 0.1210846 + (-0.0003572+0.0002027)*627 + 0.0044120*3)
```

```
## [1] 1517.812
```

About **1517.812** shares. The software creates a dummy variable for each day of the week with Friday as the default value. Thus for Saturday's dummy model, it would be 1 if Saturday and 0 if any other day. Thus the interactoin term present would be the one that relates content and Saturday together. To interpert this result we would say that the model predicts that if the article had 627 words, three images and was published on Saturday it would have 1517.812 shares. This prediction does not seem that high because it is within one standard deviation of the mean.

Discussion

In this report we found that the number of shares and article gets is related to the content (number of words in the article), the number of images, and what day it is published. When making the model we found the residual plot to have some clusters of data and a line of residuals at 7.0 and 7.18-9. This could be concerning because this has the potential to lead to a pattern in the residual plots. To solve this issue, it may be helpful to get more data and see if a larger set of data causes a more distinct pattern to emerge in the residual plot which would violate the independence condition of the error assumptions which would make this model invalid.

An area we could further investigate is if we tried to get data where each day published has the same number of articles and compare their means on average to see if a specfic day has more shares on average holding all else constant, which could give more insight into what day published would have the most number of shares (on average). Some other variables we could investigate would be the category of information the article provides (for example lifestyle, business, healthcare etc.) and we could also investigate where it was shared (for example social media, new providers website, link, etc). This information could affect the number of shares because there could be more share depending on the topic and the medium it is shared on (there could even be a relationship between the two).

This analysis could be useful when trying to optimize the number of shares a certain artile gets. Especially when there is some important finding or urgent article that should be shared, using this model can help find an optimal number of words, number of images, and what day to publish on.