STAT5371_MT_Project

Li Sun

Wednesday, October 21, 2015

DATA SOURCE

Data is from UCI website http://archive.ics.uci.edu/ml/datasets/Online+News+Popularity This is an data analysis on online news popularity by using regression model.

Attribute Information: 0. url: URL of the article (non-predictive) 1. timedelta: Days between the article publication and the dataset acquisition (non-predictive) 2. n_tokens_title: Number of words in the title 3. n_tokens_content: Number of words in the content 4. n_unique_tokens: Rate of unique words in the content 5. n_non_stop_words: Rate of non-stop words in the content 6. n_non_stop_unique_tokens: Rate of unique non-stop words in the content 7. num_hrefs: Number of links 8. num_self_hrefs: Number of links to other articles published by Mashable 9. num_imgs: Number of images 10. num_videos: Number of videos 11. average_token_length: Average length of the words in the content 12. num_keywords: Number of keywords in the metadata 13. data_channel_is_lifestyle: Is data channel 'Lifestyle'? 14. data_channel_is_entertainment: Is data channel 'Entertainment'? 15. data_channel_is_bus: Is data channel 'Business'? 16. data_channel_is_socmed: Is data channel 'Social Media'? 17. data_channel_is_tech: Is data channel 'Tech'? 18. data_channel_is_world: Is data channel 'World'? 19. kw_min_min: Worst keyword (min. shares) 20. kw_max_min: Worst keyword (max. shares) 21. kw_avg_min: Worst keyword (avg. shares) 22. kw_min_max: Best keyword (min. shares) 23. kw_max_max: Best keyword (max. shares) 24. kw_avg_max: Best keyword (avg. shares) 25. kw_min_avg: Avg. keyword (min. shares) 26. kw_max_avg: Avg. keyword (max. shares) 27. kw_avg_avg: Avg. keyword (avg. shares) 28. self_reference_min_shares: Min. shares of referenced articles in Mashable 29. self_reference_max_shares: Max. shares of referenced articles in Mashable 30. self_reference_avg_sharess: Avg. shares of referenced articles in Mashable 31. weekday_is_monday: Was the article published on a Monday? 32. weekday_is_tuesday: Was the article published on a Tuesday? 33. weekday_is_wednesday: Was the article published on a Wednesday? 34. weekday_is_thursday: Was the article published on a Thursday? 35. weekday_is_friday: Was the article published on a Friday? 36. weekday_is_saturday: Was the article published on a Saturday? 37. weekday_is_sunday: Was the article published on a Sunday? 38. is_weekend: Was the article published on the weekend? 39. LDA_00: Closeness to LDA topic 0 40. LDA_01: Closeness to LDA topic 1 41. LDA_02: Closeness to LDA topic 2 42. LDA_03: Closeness to LDA topic 3 43. LDA_04: Closeness to LDA topic 4 44. global_subjectivity: Text subjectivity 45. global_sentiment_polarity: Text sentiment polarity 46. global_rate_positive_words: Rate of positive words in the content 47. global_rate_negative_words: Rate of negative words in the content 48. rate_positive_words: Rate of positive words among non-neutral tokens 49. rate_negative_words: Rate of negative words among non-neutral tokens 50. avg_positive_polarity: Avg. polarity of positive words 51. min_positive_polarity: Min. polarity of positive words 52. max_positive_polarity: Max. polarity of positive words 53. avg_negative_polarity: Avg. polarity of negative words 54. min_negative_polarity: Min. polarity of negative words 55. max_negative_polarity: Max. polarity of negative words 56. title_subjectivity: Title subjectivity 57. title_sentiment_polarity: Title polarity 58. abs_title_subjectivity: Absolute subjectivity level 59. abs_title_sentiment_polarity: Absolute polarity level 60. shares: Number of shares (target)

Read in data and preprocess

Data is read by read.csv function, and preprocessed by adjusting data type and remove "url" variable which will not be used in this analysis

```
#read in data
raw<-read.csv("OnlineNewsPopularity.csv")
dim(raw)
```

```
## [1] 39644    61
```

```
names(raw)
```

```
##  [1] "url"                           "timedelta"
##  [3] "n_tokens_title"                "n_tokens_content"
##  [5] "n_unique_tokens"               "n_non_stop_words"
##  [7] "n_non_stop_unique_tokens"      "num_hrefs"
##  [9] "num_self_hrefs"                "num_imgs"
## [11] "num_videos"                    "average_token_length"
## [13] "num_keywords"                  "data_channel_is_lifestyle"
## [15] "data_channel_is_entertainment" "data_channel_is_bus"
## [17] "data_channel_is_socmed"        "data_channel_is_tech"
## [19] "data_channel_is_world"         "kw_min_min"
## [21] "kw_max_min"                    "kw_avg_min"
## [23] "kw_min_max"                    "kw_max_max"
## [25] "kw_avg_max"                    "kw_min_avg"
## [27] "kw_max_avg"                    "kw_avg_avg"
## [29] "self_reference_min_shares"     "self_reference_max_shares"
## [31] "self_reference_avg_sharess"    "weekday_is_monday"
## [33] "weekday_is_tuesday"            "weekday_is_wednesday"
## [35] "weekday_is_thursday"           "weekday_is_friday"
## [37] "weekday_is_saturday"           "weekday_is_sunday"
## [39] "is_weekend"                    "LDA_00"
## [41] "LDA_01"                        "LDA_02"
## [43] "LDA_03"                        "LDA_04"
## [45] "global_subjectivity"           "global_sentiment_polarity"
## [47] "global_rate_positive_words"    "global_rate_negative_words"
## [49] "rate_positive_words"           "rate_negative_words"
## [51] "avg_positive_polarity"         "min_positive_polarity"
## [53] "max_positive_polarity"         "avg_negative_polarity"
## [55] "min_negative_polarity"         "max_negative_polarity"
## [57] "title_subjectivity"            "title_sentiment_polarity"
## [59] "abs_title_subjectivity"        "abs_title_sentiment_polarity"
## [61] "shares"
```

```
#sapply(raw, class)
raw[,61] <- as.numeric(raw[,61])
raw <- raw[,-1]
```

Loading libraries

```
suppressWarnings(library(ggplot2))
suppressWarnings(library(reshape2))
suppressWarnings(library(DAAG))
```

```
## Loading required package: lattice
```

```
#suppressWarnings(library(leaps))
```

Specify questions

Generally I want to check what attributes of a article will affect its popularity, in other word, what kind of articles are attracting eyes from general public. There are 60 columns, including one dependent variable which is "shares" and 59 independent ones. We want to take a look at these 59 variables first by Exploratory Data Analyasis (EDA).

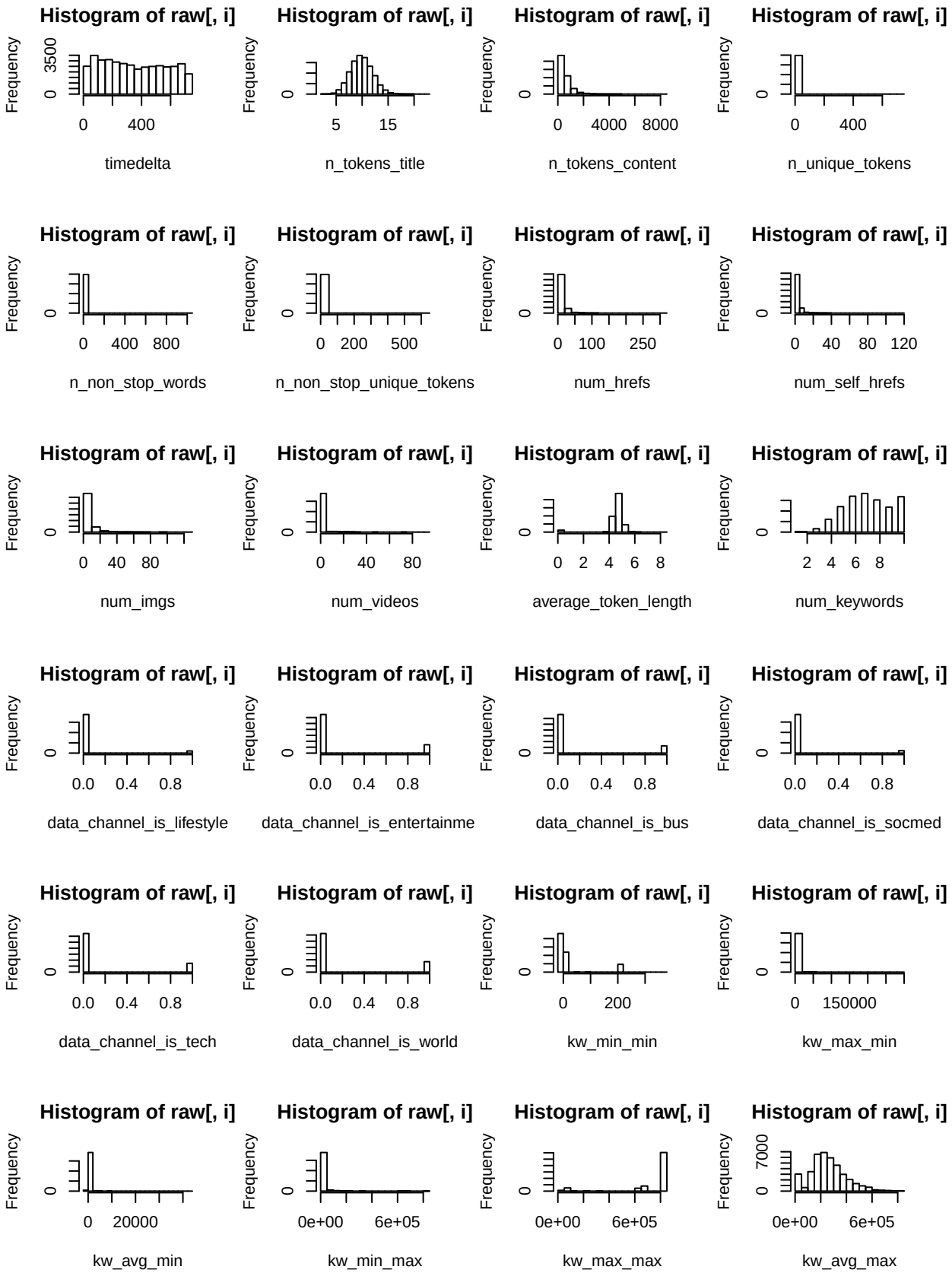
EDA and Variables selection

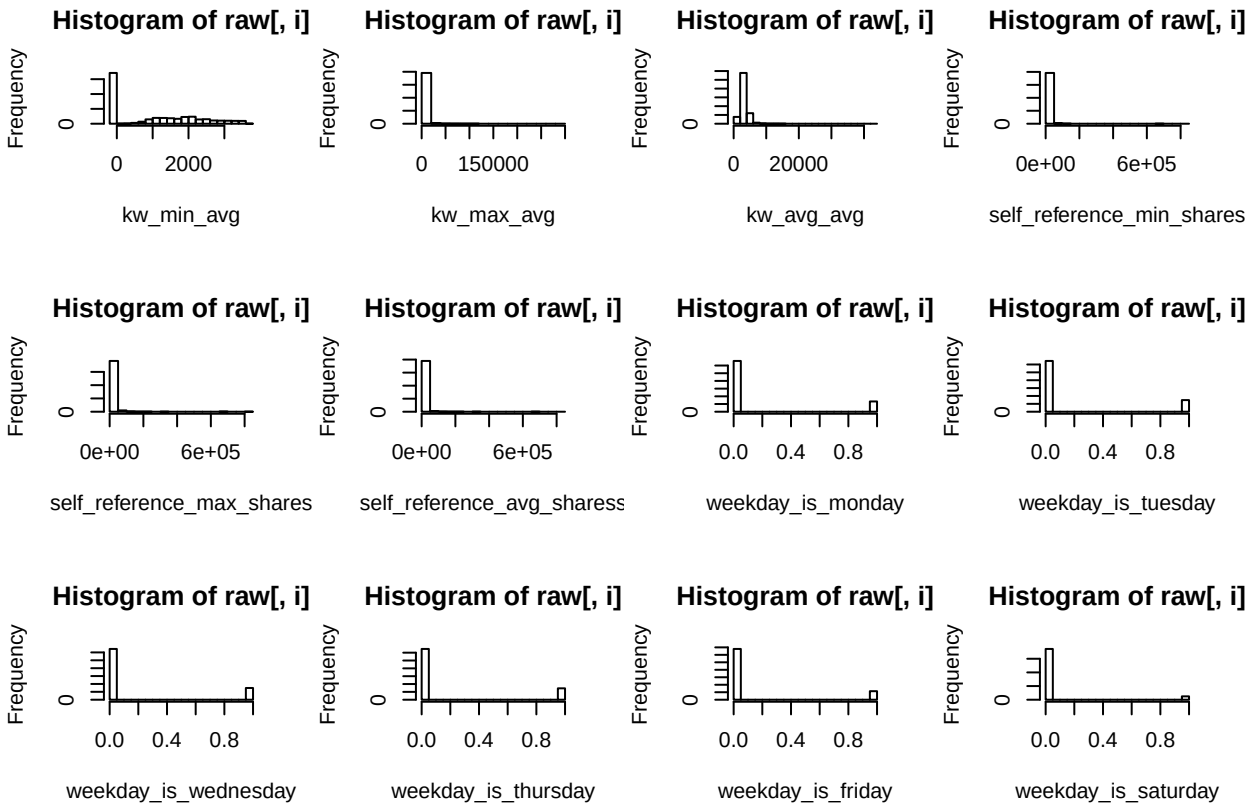
In this section, we take a look and the whole dataset and try to make it better fit regression analysis.

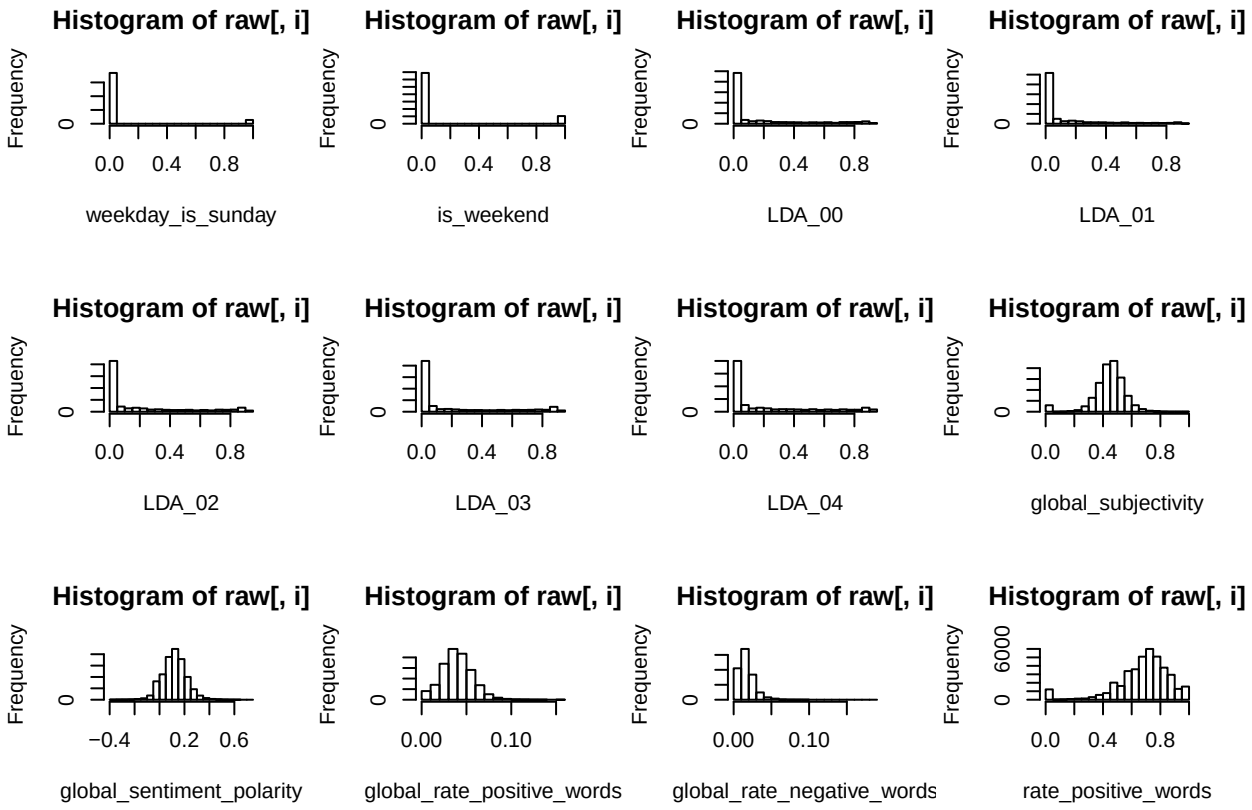
1. Data distributions

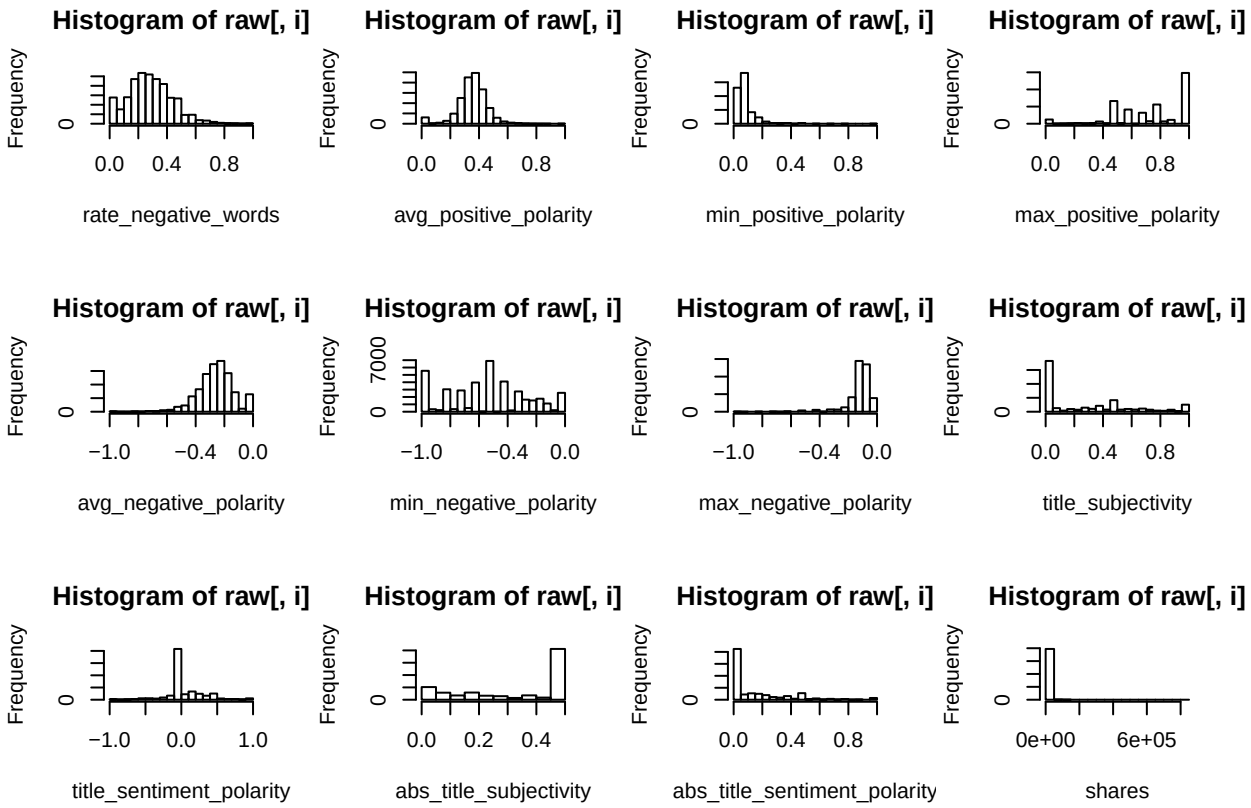
Here I plot all the variable data by histogram to check the distributions.

```
par(mfrow=c(3,4))
for(i in 1:length(raw)){hist(raw[,i], xlab=names(raw)[i])}
```


Histogram of raw[, i]
Frequency
timedelta
n_tokens_title
n_tokens_content
n_unique_tokens
n_non_stop_words
n_non_stop_unique_tokens
num_hrefs
num_self_hrefs
num_imgs
num_videos
average_token_length
num_keywords
data_channel_is_lifestyle
data_channel_is_entertainme
data_channel_is_bus
data_channel_is_socmed
data_channel_is_tech
data_channel_is_world
kw_min_min
kw_max_min
kw_avg_min
kw_min_max
kw_max_max
kw_avg_max


Histogram of raw[, i]
Frequency
0
0
2000
kw_min_avg
Histogram of raw[, i]
Frequency
0
0
150000
kw_max_avg
Histogram of raw[, i]
Frequency
0
0
20000
kw_avg_avg
Histogram of raw[, i]
Frequency
0
0e+00
6e+05
self_reference_min_shares
Histogram of raw[, i]
Frequency
0
0e+00
6e+05
self_reference_max_shares
Histogram of raw[, i]
Frequency
0
0e+00
6e+05
self_reference_avg_sharess
Histogram of raw[, i]
Frequency
0
0.0
0.4
0.8
weekday_is_monday
Histogram of raw[, i]
Frequency
0
0.0
0.4
0.8
weekday_is_tuesday
Histogram of raw[, i]
Frequency
0
0.0
0.4
0.8
weekday_is_wednesday
Histogram of raw[, i]
Frequency
0
0.0
0.4
0.8
weekday_is_thursday
Histogram of raw[, i]
Frequency
0
0.0
0.4
0.8
weekday_is_friday
Histogram of raw[, i]
Frequency
0
0.0
0.4
0.8
weekday_is_saturday


Histogram of raw[, i]
Frequency
weekday_is_sunday
Histogram of raw[, i]
Frequency
is_weekend
Histogram of raw[, i]
Frequency
LDA_00
Histogram of raw[, i]
Frequency
LDA_01
Histogram of raw[, i]
Frequency
LDA_02
Histogram of raw[, i]
Frequency
LDA_03
Histogram of raw[, i]
Frequency
LDA_04
Histogram of raw[, i]
Frequency
global_subjectivity
Histogram of raw[, i]
Frequency
global_sentiment_polarity
Histogram of raw[, i]
Frequency
global_rate_positive_words
Histogram of raw[, i]
Frequency
global_rate_negative_words
Histogram of raw[, i]
Frequency
rate_positive_words



After going over all the distributions of all individual variables, several problems identified

a. outlier in var "n_unique_tokens", "n_non_stop_words", and "n_non_stop_unique_tokens", which might be due to typing error. We will remove that observation.

```
raw <- raw[raw[,4]<1,]
```

b. Missing values are very troubling in this data set because they are coded as 0. So you have to judge if the 0 are missing or real data. By check the distributions, we found around 3000 observations with missing values in 9 different variables. We will remove all cases with missing values.

```
for(i in c(11,20,44,45,46,48,49,50,53))raw <- raw[raw[,i]!=0,]
```

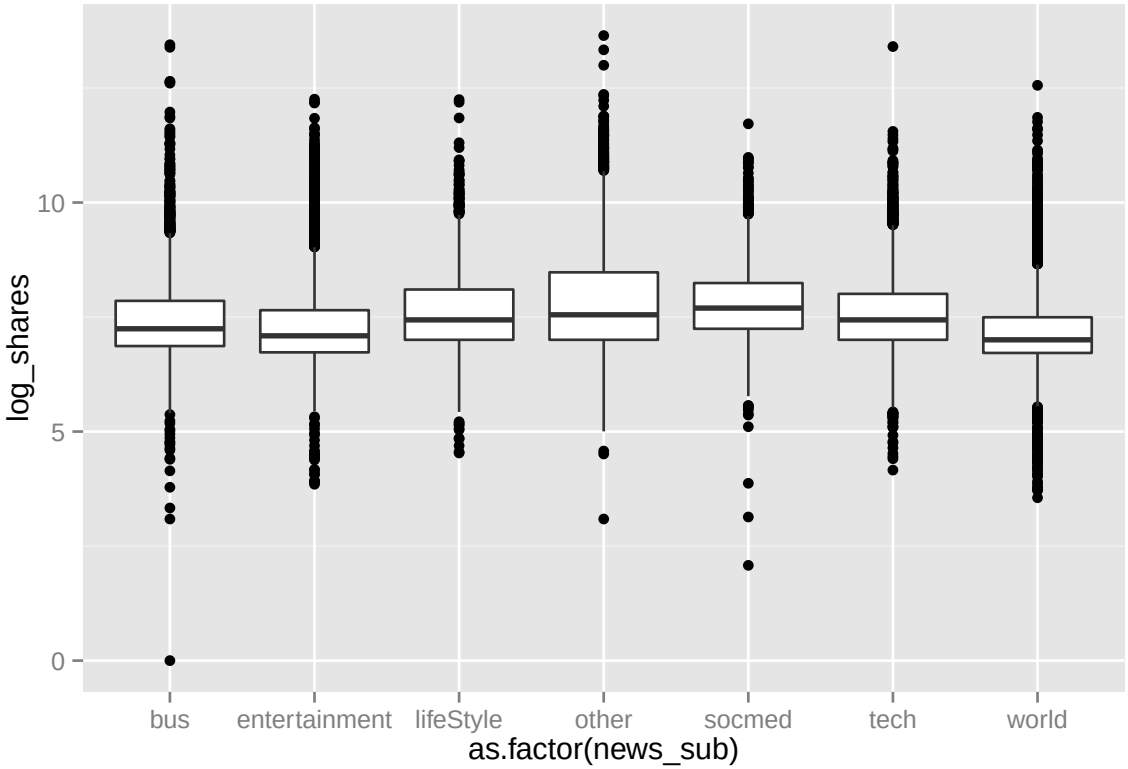
c. Skewed data. Lots of the variables are heavily right skewed, including the response "shares". So we will transform them to reduce the skewness. For those variables with all values bigger than 0, we use log, and other variable with 0, we use square root to transform them.

```
for(i in c(3,7,8,9,10,22,26:30,39:43,47, 60)){
  if(!sum(raw[,i]==0)){raw[,i] <- log(raw[,i]); names(raw)[i] <- paste("log_",names(raw)[i], sep="")}
  else{raw[,i] <- sqrt(raw[,i]); names(raw)[i] <- paste("sqrt_",names(raw)[i], sep="")}
}
```

d. 19th, 21st, 23rd, 25th Variables contains negative values that cannot be explained by information available, so they will be removed

```
raw <- raw[, -c(19,21,23,25)]
```

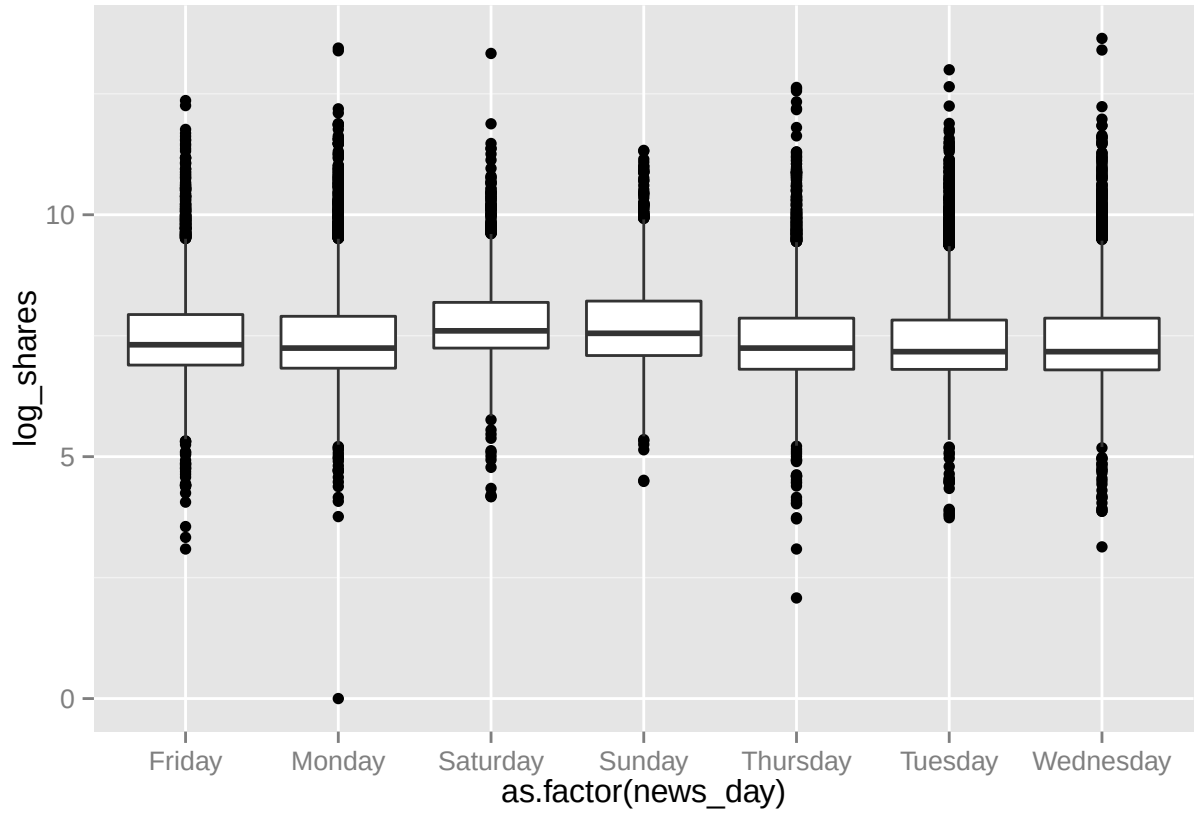
2. Does subjects and publishing days of news matters?



a. Subjects matters?
As you see, all subjects look similar regarding share numbers. So I will remove the 7 variables about subjects.

b. Publishing Days might or might not affect shares, let's look at it.

```
raw$news_day <- rep("Sunday", nrow(raw))
raw$news_day[raw$weekday_is_monday==1] <- "Monday"
raw$news_day[raw$weekday_is_tuesday==1] <- "Tuesday"
raw$news_day[raw$weekday_is_wednesday==1] <- "Wednesday"
raw$news_day[raw$weekday_is_thursday==1] <- "Thursday"
raw$news_day[raw$weekday_is_friday==1] <- "Friday"
raw$news_day[raw$weekday_is_saturday==1] <- "Saturday"
#Check
p1 <- ggplot(data=raw, aes(as.factor(news_day), log_shares))
p1 + geom_boxplot()
```



Publishing day didn't show much influence on shares neither. So I will get rid of all the indicators but leave "is_weekend" because I do see some difference bewtween weekdays and weekend data.

```
#remove 7 publishing day var and 7 subject indicator var
raw2 <- raw[,-c(13:18, 27:33, 57,58)]
```

3. PCA analysis

PCA analysis can tell us where the variance of our independent variables are from? How they form the shape of our data variance.

```
x <- as.matrix(scale(raw2[,-43]))
dim(x)
```

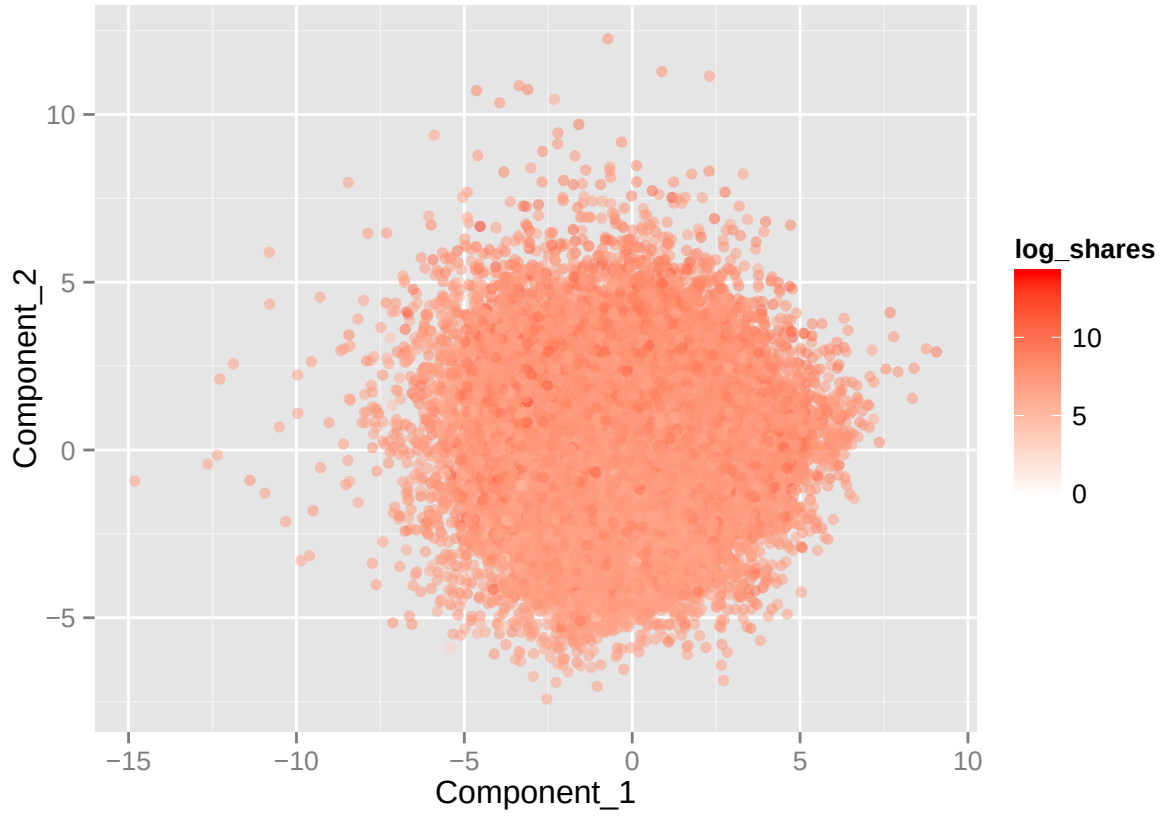
```
## [1] 36274    42
```

```
corx <- cor(x)
dim(corx)
```

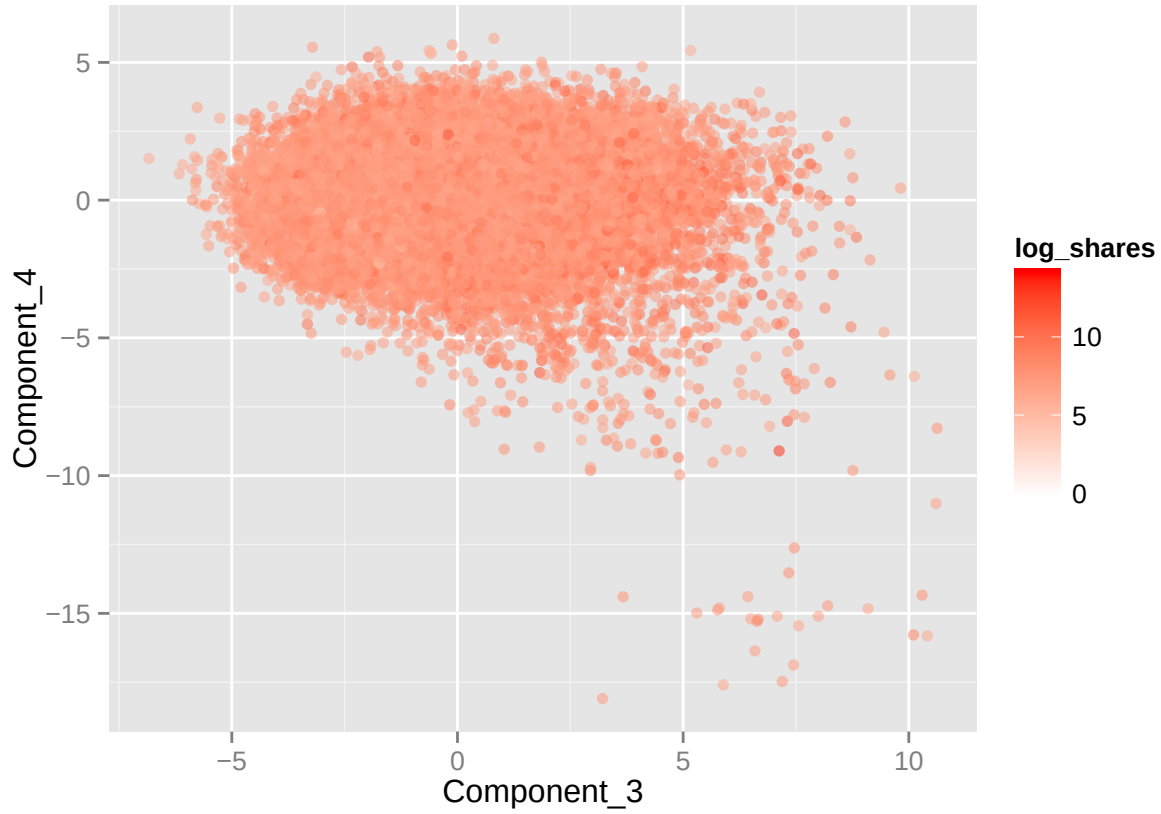
```
## [1] 42 42
```

```
evd<-svd(corx)
w <- x %*% evd$u
pca2 <- as.data.frame(cbind(w[,1:4], raw2$log_shares))
```

```
names(pca2) <- c("Component_1", "Component_2", "Component_3", "Component_4", "log_shares")
#Map share on first two components
pcaplot <- ggplot(aes(Component_1, Component_2, colour=log_shares), data=pca2)
pcaplot + geom_point(alpha = 0.5) + scale_colour_gradient(limits=c(0, 14), low="white", high="red")
```




```
#Map share on 3rd and 4th components
pcaplot <- ggplot(aes(Component_3, Component_4, colour=log_shares), data=pca2)
pcaplot + geom_point(alpha = 0.5) + scale_colour_gradient(limits=c(0, 14), low="white", high="red")
```

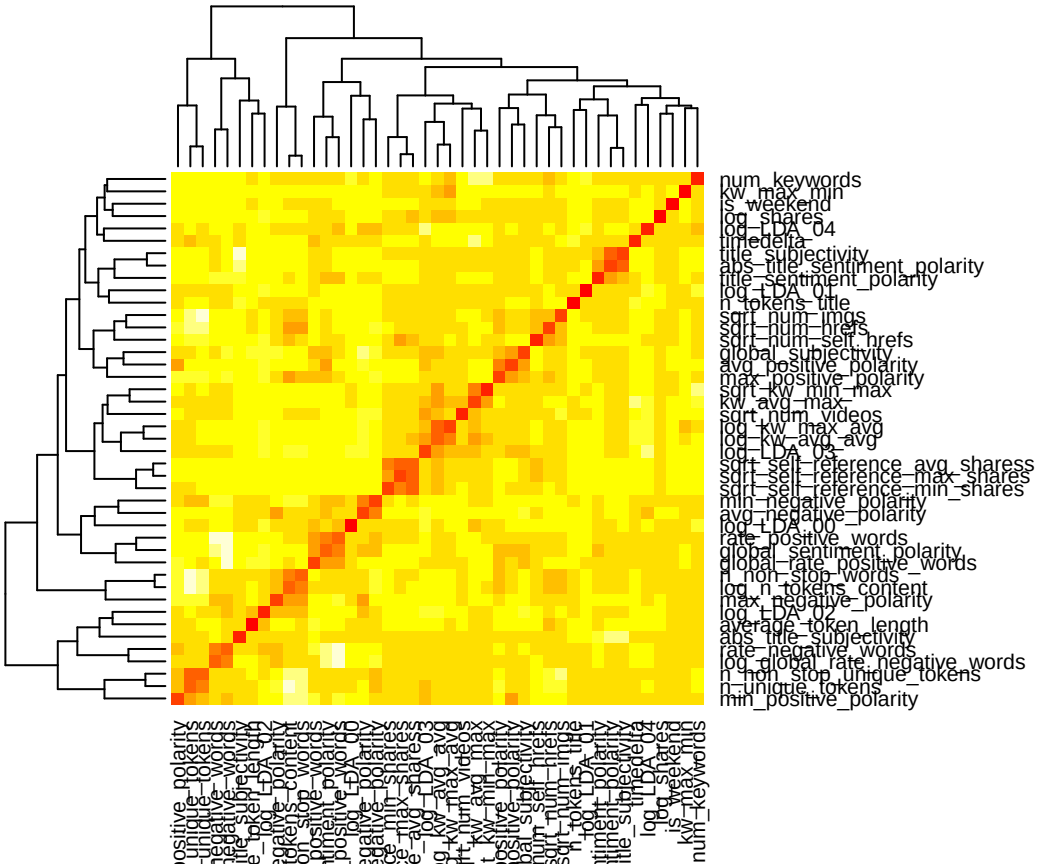


Conclusion, the variance of share number is not aligned with the first 4 major components of variance from independent variables.In other words, most of the infomation from our independent variables might not be related to our dependent variable. So there are too much non-relevant information we should get rid of.

4. Does all potential predictors are independent to each other and relevant to our dependent variable?

I will use heatmap to check correlation matrix of the 43 left columns

```
corm<-1-cor(raw2)
heatmap(corm)
```

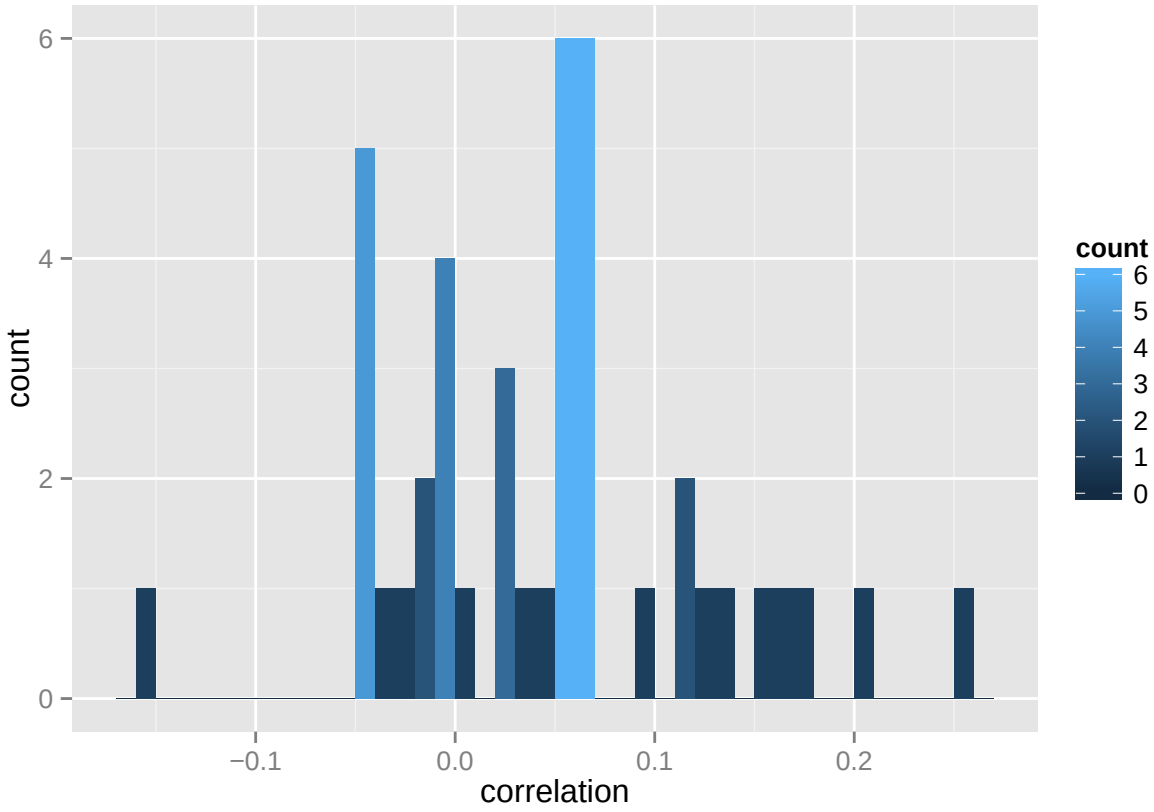


Let's further loot at the correlations of different variables with our dependenet variables.

```
summary(corm[43,-43])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.7462  0.9317  0.9489  0.9540  1.0060  1.1530
```

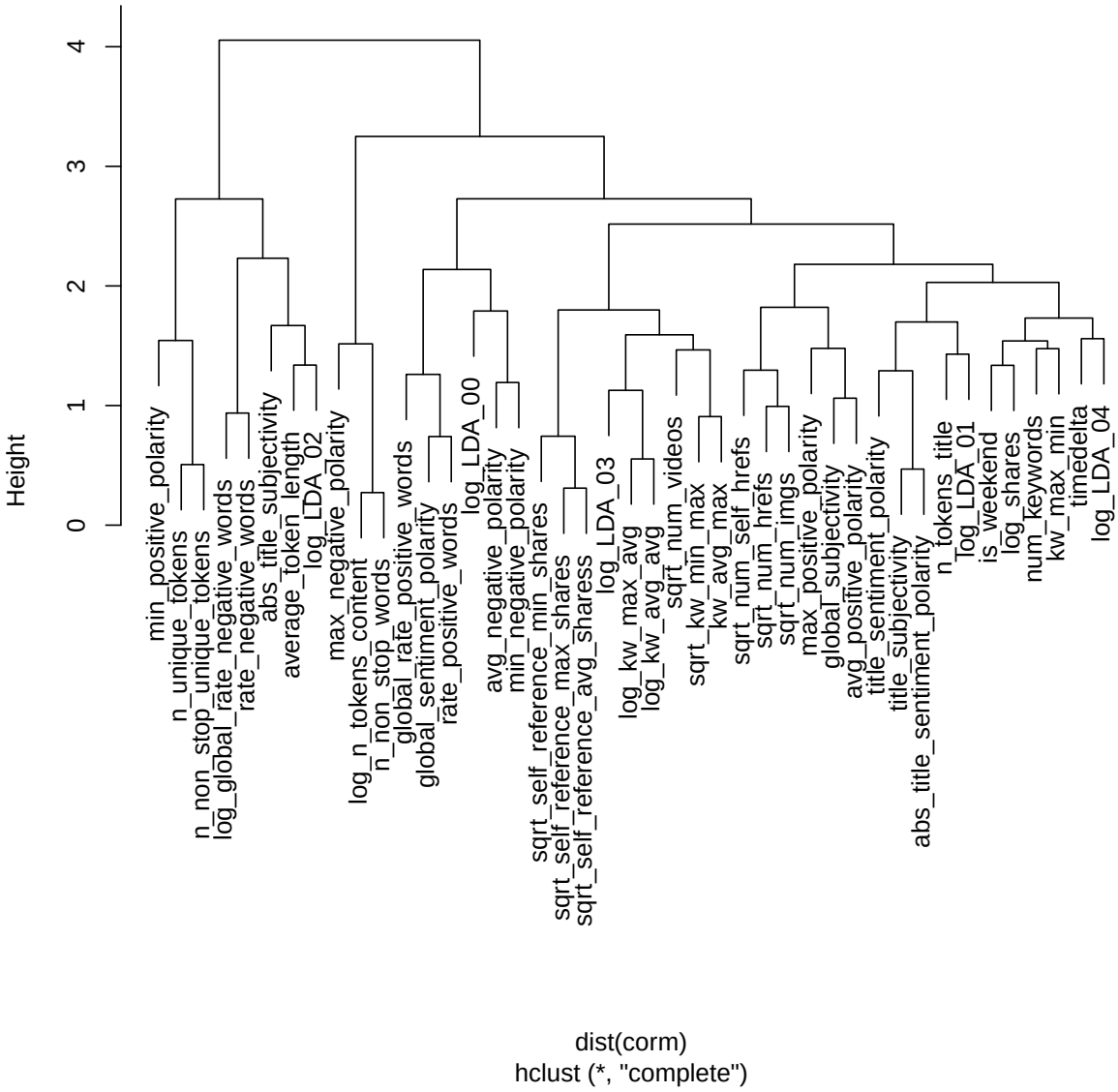
```
qplot(1-corm[43,-43], binwidth=0.01, fill=..count.., geom="histogram", xlab="correlation")
```



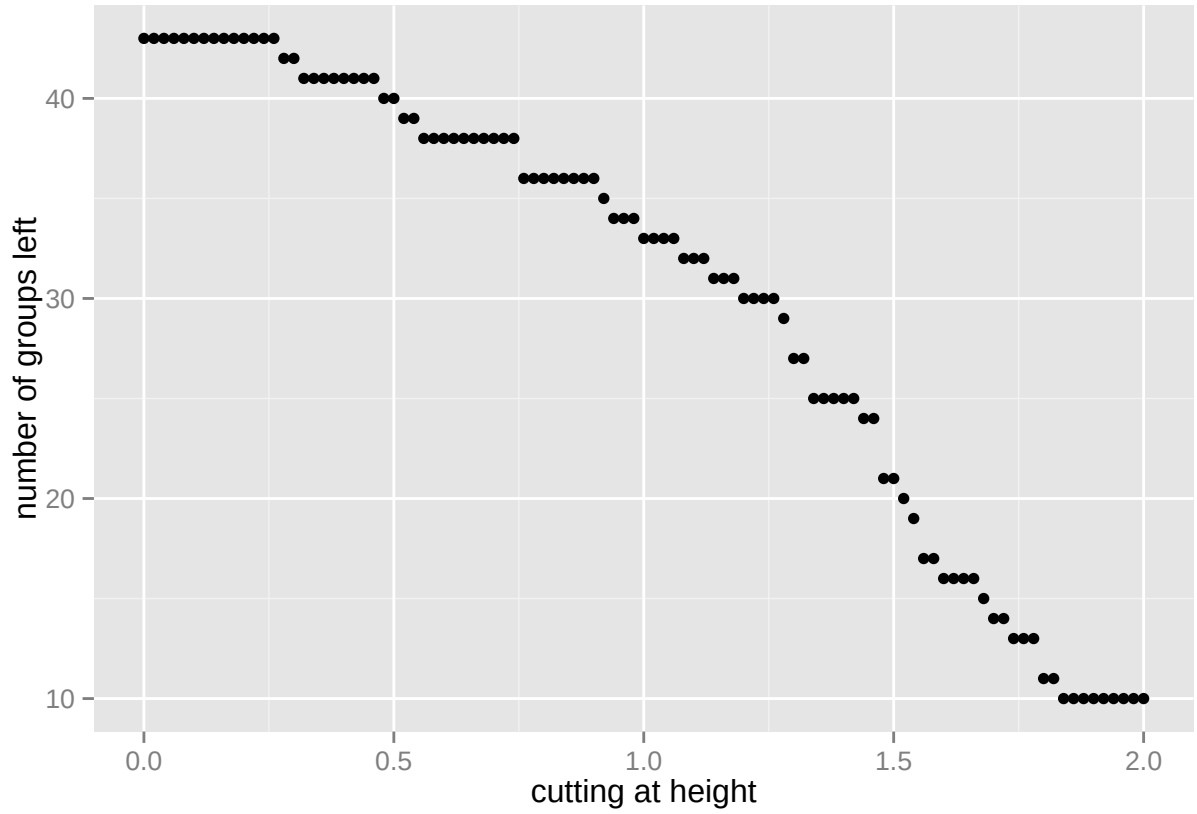
Generally all the predictors have pretty low correlations with our log(shares). We have some variables have nearly 0 correlations with share numbers. Considering most variables relative low correlations, we will not exclude any variables because of low correlation.

From the heatmap above, we can tell there indeed are some groups of variables which are pretty close to each other. Let's build the tree and cut it to get groups!

```
hc <- hclust(dist(corm))
par(mfrow=c(1,1))
plot(hc)
```




```
y<-numeric()
for(i in seq(0,2,0.02)){
  y <- c(y, length(unique(cutree(hc,h=i))))
}
x <- seq(0,2,0.02)
ggplot(aes(x=x,y=y), data=as.data.frame(cbind(x,y)))+geom_point()+labs(x="cutting at height", y="number
```



According to those plots, I will maintain all 43 variables.

Exploratory Modeling

I use all 43 variables to build a regression model and analyze it.

```
raw2$is_weekend <- as.factor(raw2$is_weekend)
xx <- raw2[,-43]
yy <- raw2[,43]
full <- lm(log_shares~ ., data=raw2)
summary(full)
```

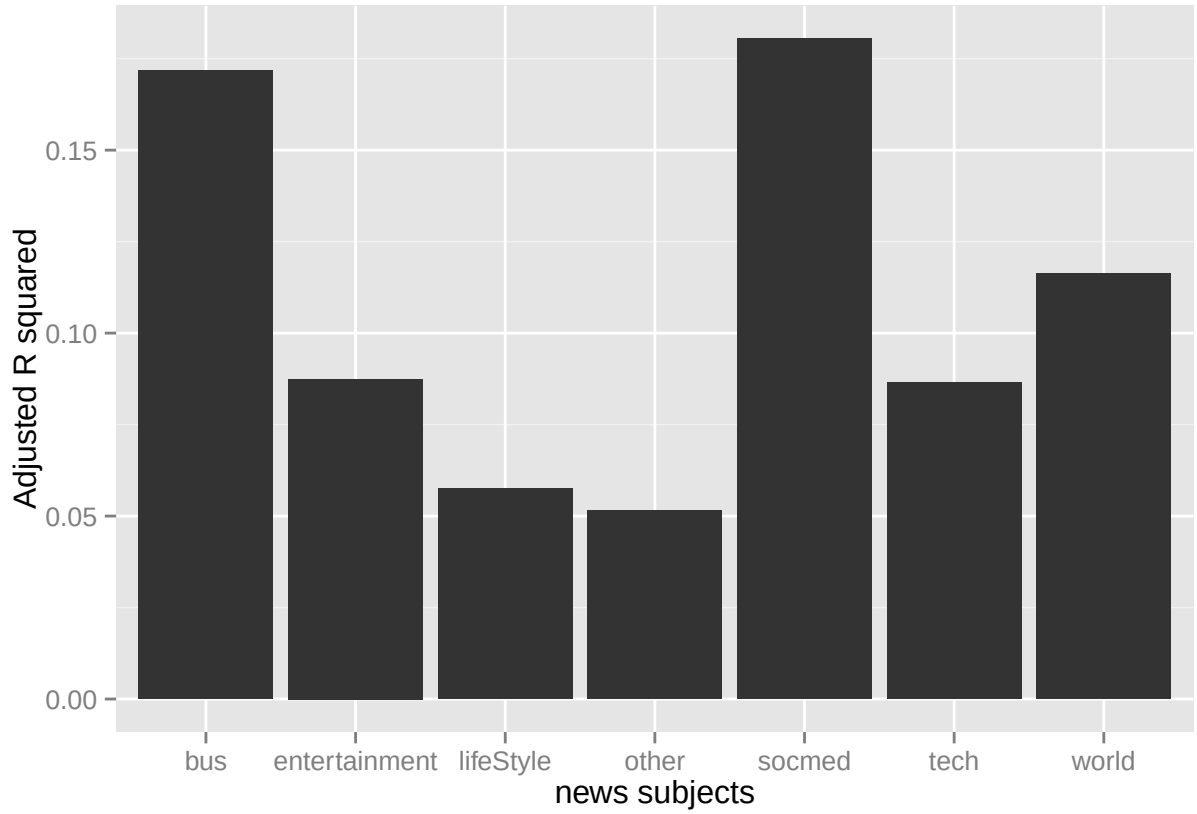
```
##
## Call:
## lm(formula = log_shares ~ ., data = raw2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1865 -0.5533 -0.1592  0.4032  5.7973
##
## Coefficients: (2 not defined because of singularities)
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    2.800e+00  2.587e-01  10.822  < 2e-16 ***
## timedelta                      2.095e-04  2.726e-05   7.686 1.56e-14 ***
## n_tokens_title                 5.046e-03  2.298e-03   2.196 0.028124 *
```

```
## log_n_tokens_content            -1.126e-01  2.020e-02  -5.574 2.50e-08 ***
## n_unique_tokens                 -8.456e-01  2.071e-01  -4.084 4.44e-05 ***
## n_non_stop_words                        NA         NA      NA       NA
## n_non_stop_unique_tokens         2.252e-01  1.418e-01   1.588 0.112383
## sqrt_num_hrefs                   4.686e-02  4.598e-03  10.191  < 2e-16 ***
## sqrt_num_self_hrefs             -3.157e-02  6.249e-03  -5.053 4.38e-07 ***
## sqrt_num_imgs                    2.486e-02  4.346e-03   5.721 1.07e-08 ***
## sqrt_num_videos                  3.935e-02  5.459e-03   7.208 5.79e-13 ***
## average_token_length            -7.385e-02  1.934e-02  -3.818 0.000135 ***
## num_keywords                     7.047e-03  3.081e-03   2.288 0.022172 *
## kw_max_min                      -7.789e-06  1.255e-06  -6.204 5.56e-10 ***
## sqrt_kw_min_max                 -3.955e-04  6.507e-05  -6.079 1.22e-09 ***
## kw_avg_max                      -3.673e-07  5.429e-08  -6.765 1.36e-11 ***
## log_kw_max_avg                  -9.067e-02  2.138e-02  -4.241 2.23e-05 ***
## log_kw_avg_avg                   7.768e-01  3.239e-02  23.982  < 2e-16 ***
## sqrt_self_reference_min_shares   1.121e-03  2.778e-04   4.035 5.48e-05 ***
## sqrt_self_reference_max_shares  -3.232e-04  3.028e-04  -1.067 0.285884
## sqrt_self_reference_avg_sharess  1.609e-03  5.335e-04   3.017 0.002557 **
## is_weekend1                      2.629e-01  1.366e-02  19.253  < 2e-16 ***
## log_LDA_00                       4.155e-02  4.601e-03   9.030  < 2e-16 ***
## log_LDA_01                      -3.196e-02  4.776e-03  -6.692 2.23e-11 ***
## log_LDA_02                      -1.759e-02  4.788e-03  -3.674 0.000239 ***
## log_LDA_03                      -1.448e-03  5.235e-03  -0.277 0.782151
## log_LDA_04                       3.808e-02  4.623e-03   8.236  < 2e-16 ***
## global_subjectivity              4.791e-01  6.735e-02   7.113 1.16e-12 ***
## global_sentiment_polarity       -3.027e-01  1.346e-01  -2.249 0.024489 *
## global_rate_positive_words      -1.309e+00  6.423e-01  -2.038 0.041535 *
## log_global_rate_negative_words   5.583e-02  2.299e-02   2.428 0.015177 *
## rate_positive_words              5.042e-01  1.252e-01   4.026 5.68e-05 ***
## rate_negative_words                     NA         NA      NA       NA
## avg_positive_polarity            6.234e-03  1.075e-01   0.058 0.953776
## min_positive_polarity           -1.879e-01  9.165e-02  -2.050 0.040376 *
## max_positive_polarity           -2.411e-02  3.407e-02  -0.708 0.479070
## avg_negative_polarity            1.033e-01  9.764e-02   1.058 0.290129
## min_negative_polarity           -5.074e-02  3.575e-02  -1.419 0.155858
## max_negative_polarity            2.458e-02  8.194e-02   0.300 0.764195
## title_subjectivity               6.045e-02  2.149e-02   2.813 0.004905 **
## title_sentiment_polarity         7.322e-02  1.980e-02   3.698 0.000217 ***
## abs_title_subjectivity           1.273e-01  2.874e-02   4.431 9.42e-06 ***
## abs_title_sentiment_polarity     3.214e-02  3.117e-02   1.031 0.302516
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8668 on 36233 degrees of freedom
## Multiple R-squared:  0.123,  Adjusted R-squared:  0.122
## F-statistic: 127.1 on 40 and 36233 DF,  p-value: < 2.2e-16
```

Very low R square around 0.122. So the data really does not contain the info about popularity of news? Remember I discard some info in the beginning about subjects of news. So what about in different subjects, do we have higher prediction power in each subject?

```
sublist<-split(raw2, raw$news_sub)
RsqrSub <- data.frame("sub"=names(sublist), "Rsqr"=rep(0,7))
for(i in 1:7){
```

```
  temp<-lm(log_shares~ ., data=sublist[[i]])
  RsqrSub[i,2]<-summary(temp)$adj.r.squared
}
ggplot(aes(x=factor(sub), y=Rsqr), data=RsqrSub) + geom_bar(stat="identity") + labs(x="news subjects", y
```



We do see some difference across different subjects. However, they are not high in general. The highest one is social media news. So I will only use the subset of news about social media to build a model for practise. First start with full model and do some diagnositc plots

```
social <- sublist[[5]]
dim(social)
```

```
## [1] 2162   43
```

```
socfull <- lm(log_shares ~ ., data=social)
summary(socfull)
```

```
##
## Call:
## lm(formula = log_shares ~ ., data = social)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8479 -0.4487 -0.1083  0.3724  3.9674
##
```

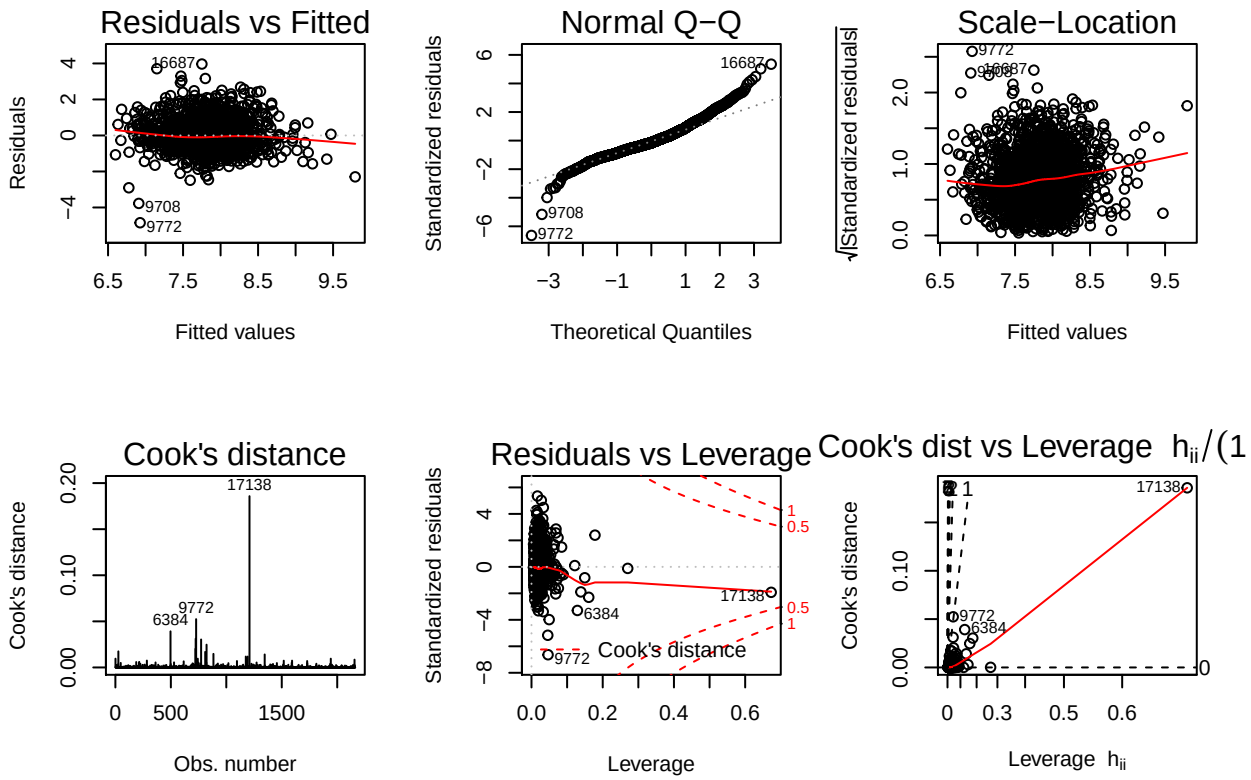
```
## Coefficients: (2 not defined because of singularities)
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     3.376e+00  9.660e-01   3.495 0.000484 ***
## timedelta                       2.862e-04  1.187e-04   2.412 0.015961 *
## n_tokens_title                 -1.294e-03  8.068e-03  -0.160 0.872608
## log_n_tokens_content           -2.915e-02  7.627e-02  -0.382 0.702398
## n_unique_tokens                -1.206e+00  7.549e-01  -1.598 0.110132
## n_non_stop_words                       NA         NA      NA       NA
## n_non_stop_unique_tokens       -1.436e-01  5.205e-01  -0.276 0.782606
## sqrt_num_hrefs                 -3.786e-02  1.683e-02  -2.249 0.024603 *
## sqrt_num_self_hrefs            -1.427e-02  2.056e-02  -0.694 0.487489
## sqrt_num_imgs                  -3.180e-02  1.565e-02  -2.031 0.042343 *
## sqrt_num_videos                 3.780e-02  2.013e-02   1.878 0.060520 .
## average_token_length           -3.725e-02  7.617e-02  -0.489 0.624831
## num_keywords                    1.669e-02  1.022e-02   1.633 0.102559
## kw_max_min                     -8.694e-06  4.329e-06  -2.008 0.044719 *
## sqrt_kw_min_max                -6.825e-04  2.237e-04  -3.052 0.002305 **
## kw_avg_max                     -1.445e-07  2.309e-07  -0.626 0.531530
## log_kw_max_avg                 -1.380e-01  7.957e-02  -1.734 0.083027 .
## log_kw_avg_avg                  8.156e-01  1.080e-01   7.555 6.21e-14 ***
## sqrt_self_reference_min_shares  1.626e-03  7.979e-04   2.038 0.041630 *
## sqrt_self_reference_max_shares -3.283e-04  7.286e-04  -0.451 0.652323
## sqrt_self_reference_avg_sharess 1.455e-03  1.373e-03   1.059 0.289606
## is_weekend1                     1.146e-01  4.903e-02   2.338 0.019494 *
## log_LDA_00                      9.841e-02  1.952e-02   5.041 5.03e-07 ***
## log_LDA_01                     -7.151e-02  2.215e-02  -3.228 0.001265 **
## log_LDA_02                     -3.168e-02  1.713e-02  -1.849 0.064567 .
## log_LDA_03                     -4.938e-02  1.934e-02  -2.553 0.010749 *
## log_LDA_04                     -3.107e-02  1.710e-02  -1.817 0.069408 .
## global_subjectivity            -2.483e-01  2.519e-01  -0.986 0.324402
## global_sentiment_polarity       5.817e-01  4.795e-01   1.213 0.225193
## global_rate_positive_words      9.367e-01  2.075e+00   0.451 0.651811
## log_global_rate_negative_words -6.438e-02  8.306e-02  -0.775 0.438375
## rate_positive_words            -6.572e-01  4.828e-01  -1.361 0.173538
## rate_negative_words                    NA         NA      NA       NA
## avg_positive_polarity          -4.101e-01  4.045e-01  -1.014 0.310848
## min_positive_polarity          -9.321e-01  3.208e-01  -2.906 0.003703 **
## max_positive_polarity          -2.314e-01  1.223e-01  -1.893 0.058498 .
## avg_negative_polarity          -9.291e-03  3.610e-01  -0.026 0.979471
## min_negative_polarity          -1.771e-01  1.352e-01  -1.310 0.190459
## max_negative_polarity          -9.746e-02  2.930e-01  -0.333 0.739448
## title_subjectivity              1.041e-01  8.068e-02   1.290 0.197282
## title_sentiment_polarity       -1.019e-01  8.288e-02  -1.230 0.218944
## abs_title_subjectivity          2.670e-01  1.044e-01   2.558 0.010605 *
## abs_title_sentiment_polarity    2.370e-01  1.231e-01   1.925 0.054309 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7478 on 2121 degrees of freedom
## Multiple R-squared:  0.1956, Adjusted R-squared:  0.1804
## F-statistic: 12.89 on 40 and 2121 DF,  p-value: < 2.2e-16
```

We have two variables gives out NA in coeficients calculation, indicating those variables are linear combinations of other variables. So we will remove those two.

```
social<-social[,names(social)!="n_non_stop_words"&names(social)!="rate_negative_words"]
```

Diagnostic plot

```
par(mfrow=c(2,3))
plot(socfull, which = c(1:6))
```



According to 6 diagnostic plots, the assumptions of spherical error and normal distributed error roughly hold. However outliers identified. So let's remove those observations

```
outliers <- c(6384, 9708, 9772, 16687, 17138)
soc<-social[!rownames(social) %in% outliers,]
```

Model selection

1. Backwards model selection

```
socfull <- lm(log_shares ~ ., data=soc)
backstep <- step(socfull, direction= "backward", trace = 0)
#backstep$coefficients
summary(lm(formula(backstep), data=soc))
```

```
##
## Call:
## lm(formula = formula(backstep), data = soc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5173 -0.4585 -0.0946  0.3831  3.7823
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     3.453e+00  5.739e-01   6.017 2.09e-09 ***
## timedelta                       3.540e-04  8.426e-05   4.202 2.76e-05 ***
## n_unique_tokens                -1.220e+00  1.925e-01  -6.335 2.89e-10 ***
## sqrt_num_hrefs                 -4.941e-02  1.279e-02  -3.862 0.000116 ***
## sqrt_num_imgs                  -3.367e-02  1.439e-02  -2.340 0.019374 *
## sqrt_num_videos                 3.605e-02  1.857e-02   1.941 0.052357 .
## num_keywords                    2.158e-02  8.661e-03   2.491 0.012798 *
## log_kw_avg_avg                  5.561e-01  6.530e-02   8.516  < 2e-16 ***
## sqrt_self_reference_min_shares  2.400e-03  5.282e-04   4.543 5.86e-06 ***
## sqrt_self_reference_avg_sharess 7.203e-04  4.500e-04   1.601 0.109573
## is_weekend1                     1.070e-01  4.703e-02   2.276 0.022967 *
## log_LDA_00                      1.010e-01  1.863e-02   5.422 6.54e-08 ***
## log_LDA_01                     -6.743e-02  2.122e-02  -3.178 0.001505 **
## log_LDA_02                     -2.494e-02  1.647e-02  -1.514 0.130222
## log_LDA_03                     -4.699e-02  1.837e-02  -2.558 0.010593 *
## log_LDA_04                     -3.767e-02  1.599e-02  -2.356 0.018572 *
## min_positive_polarity          -9.753e-01  2.536e-01  -3.846 0.000124 ***
## max_positive_polarity          -2.812e-01  8.670e-02  -3.243 0.001199 **
## min_negative_polarity          -1.120e-01  6.434e-02  -1.741 0.081901 .
## title_subjectivity              2.011e-01  5.803e-02   3.465 0.000542 ***
## abs_title_subjectivity          2.676e-01  9.875e-02   2.710 0.006781 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7265 on 2136 degrees of freedom
## Multiple R-squared:  0.1972, Adjusted R-squared:  0.1897
## F-statistic: 26.24 on 20 and 2136 DF,  p-value: < 2.2e-16
```

Backwards elimination ends up with 20 variables left.

2. Foward model selection

```
minfit <- lm(log_shares ~ 1, data=soc)
forstep <- step(minfit, scope = formula(socfull), direction = "forward", trace = 0)
summary(lm(formula(forstep), data=soc))
```

```
##
## Call:
## lm(formula = formula(forstep), data = soc)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -3.3432 -0.4530 -0.0958  0.3743  3.7457
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    3.401e+00  5.867e-01   5.796 7.79e-09 ***
## sqrt_self_reference_avg_sharess  7.659e-04  4.521e-04   1.694 0.090418 .
## log_LDA_00                      1.167e-01  1.601e-02   7.291 4.30e-13 ***
## log_kw_avg_avg                  7.021e-01  1.036e-01   6.778 1.58e-11 ***
## min_positive_polarity          -9.905e-01  2.543e-01  -3.895 0.000101 ***
## sqrt_self_reference_min_shares   2.344e-03  5.304e-04   4.419 1.04e-05 ***
## n_unique_tokens                -1.233e+00  1.927e-01  -6.396 1.95e-10 ***
## sqrt_num_imgs                  -3.378e-02  1.450e-02  -2.330 0.019916 *
## kw_avg_max                      5.337e-08  2.215e-07   0.241 0.809651
## max_positive_polarity          -2.792e-01  8.667e-02  -3.221 0.001296 **
## log_LDA_01                     -5.296e-02  2.031e-02  -2.607 0.009196 **
## sqrt_num_hrefs                 -5.225e-02  1.284e-02  -4.071 4.86e-05 ***
## abs_title_sentiment_polarity    1.430e-01  1.007e-01   1.420 0.155811
## abs_title_subjectivity          2.867e-01  9.925e-02   2.888 0.003912 **
## is_weekend1                     1.067e-01  4.703e-02   2.268 0.023434 *
## num_keywords                    2.397e-02  9.453e-03   2.536 0.011275 *
## timedelta                       3.849e-04  1.128e-04   3.413 0.000654 ***
## min_negative_polarity          -1.169e-01  6.437e-02  -1.816 0.069446 .
## sqrt_kw_min_max                -3.790e-04  2.178e-04  -1.740 0.081984 .
## title_subjectivity              1.324e-01  7.693e-02   1.721 0.085451 .
## sqrt_num_videos                 3.066e-02  1.860e-02   1.648 0.099478 .
## log_LDA_03                     -3.106e-02  1.640e-02  -1.894 0.058401 .
## log_LDA_04                     -2.611e-02  1.526e-02  -1.711 0.087246 .
## log_kw_max_avg                 -1.099e-01  7.408e-02  -1.483 0.138121
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7263 on 2133 degrees of freedom
## Multiple R-squared:  0.1987, Adjusted R-squared:  0.1901
## F-statistic:    23 on 23 and 2133 DF,  p-value: < 2.2e-16
```

3. Compare the coefieients contained in 2 models

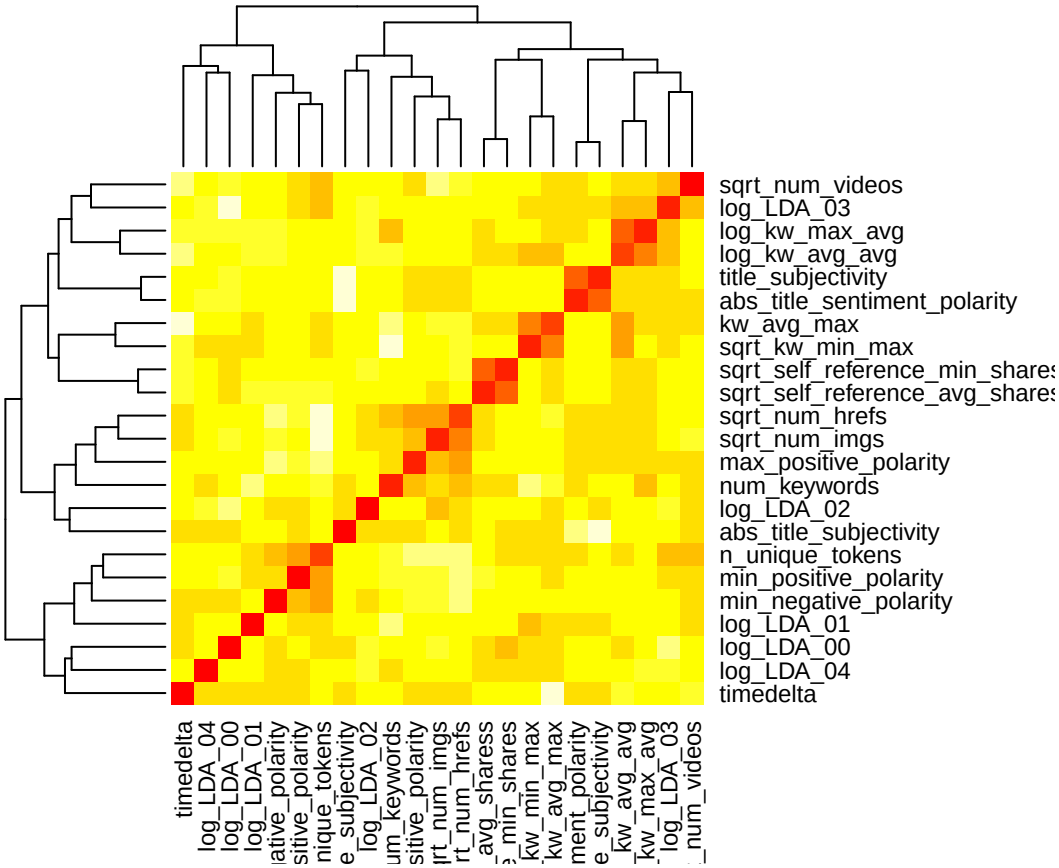
```
bc <- names(backstep$coefficients)
fc <- names(forstep$coefficients)
c <- unique(c(bc, fc))
bcind <- c %in% bc
fcind <- c %in% fc
varcompare <- data.frame("Variables"=c, "backward"=bcind, "forward"=fcind)
varcompare
```

```
##                        Variables backward forward
## 1                    (Intercept)     TRUE    TRUE
## 2                      timedelta     TRUE    TRUE
## 3                 n_unique_tokens     TRUE    TRUE
## 4                  sqrt_num_hrefs     TRUE    TRUE
## 5                    sqrt_num_imgs     TRUE    TRUE
```

```
## 6                  sqrt_num_videos     TRUE    TRUE
## 7                     num_keywords     TRUE    TRUE
## 8                   log_kw_avg_avg     TRUE    TRUE
## 9   sqrt_self_reference_min_shares     TRUE    TRUE
## 10 sqrt_self_reference_avg_sharess     TRUE    TRUE
## 11                      is_weekend1    TRUE    TRUE
## 12                       log_LDA_00    TRUE    TRUE
## 13                       log_LDA_01    TRUE    TRUE
## 14                       log_LDA_02    TRUE   FALSE
## 15                       log_LDA_03    TRUE    TRUE
## 16                       log_LDA_04    TRUE    TRUE
## 17           min_positive_polarity     TRUE    TRUE
## 18           max_positive_polarity     TRUE    TRUE
## 19           min_negative_polarity     TRUE    TRUE
## 20              title_subjectivity     TRUE    TRUE
## 21          abs_title_subjectivity     TRUE    TRUE
## 22                       kw_avg_max    FALSE    TRUE
## 23    abs_title_sentiment_polarity    FALSE    TRUE
## 24                 sqrt_kw_min_max    FALSE    TRUE
## 25                  log_kw_max_avg    FALSE    TRUE
```

4. Correlations between those variable?

```
designM <- soc[,names(soc) %in% c]
heatmap(1-cor(designM))
```




```
meltcor<-melt(cor(designM))
meltcor <- meltcor[meltcor$value!=1,]
meltcor <- meltcor[order(meltcor$value, decreasing =T),]
names(meltcor) <- c("var1", "var2", "correlation")
head(meltcor)
```

```
##                                 var1                            var2
## 242 sqrt_self_reference_avg_sharess  sqrt_self_reference_min_shares
## 264  sqrt_self_reference_min_shares sqrt_self_reference_avg_sharess
## 483    abs_title_sentiment_polarity              title_subjectivity
## 527              title_subjectivity    abs_title_sentiment_polarity
## 194                  log_kw_avg_avg                 log_kw_max_avg
## 216                  log_kw_max_avg                 log_kw_avg_avg
##     correlation
## 242   0.7872656
## 264   0.7872656
## 483   0.7320615
## 527   0.7320615
## 194   0.6946758
## 216   0.6946758
```

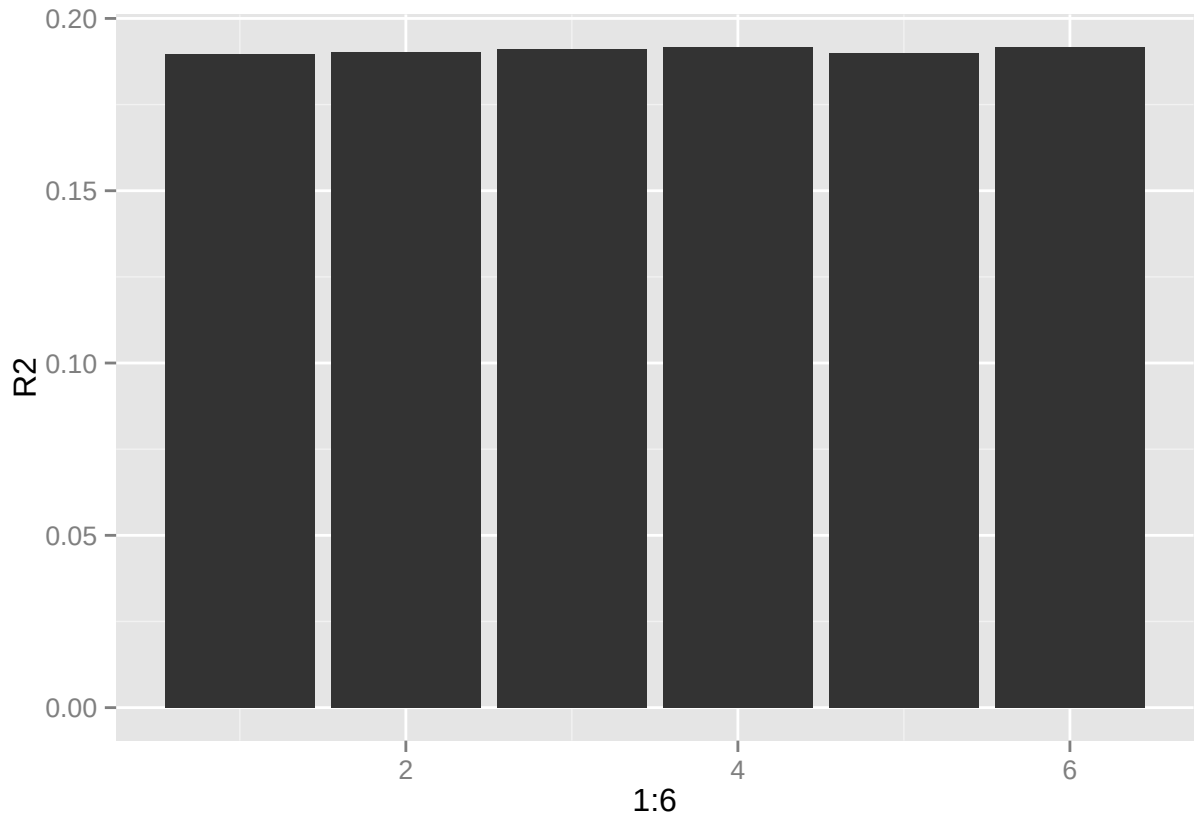
Let's add interacting terms to the existant models to generate several more models

```
intfit1 <- update(backstep, .~.+sqrt_self_reference_avg_sharess:sqrt_self_reference_min_shares)
intfit2 <- update(forstep, .~.+sqrt_self_reference_avg_sharess:sqrt_self_reference_min_shares)
intfit3 <- update(forstep, .~.+abs_title_sentiment_polarity:title_subjectivity)
intfit4 <- update(forstep, .~.+sqrt_self_reference_avg_sharess:sqrt_self_reference_min_shares+abs_title_
```

5. Compare all six models

Now we have 6 models we need to compare. Lets look at their adjusted R squared first

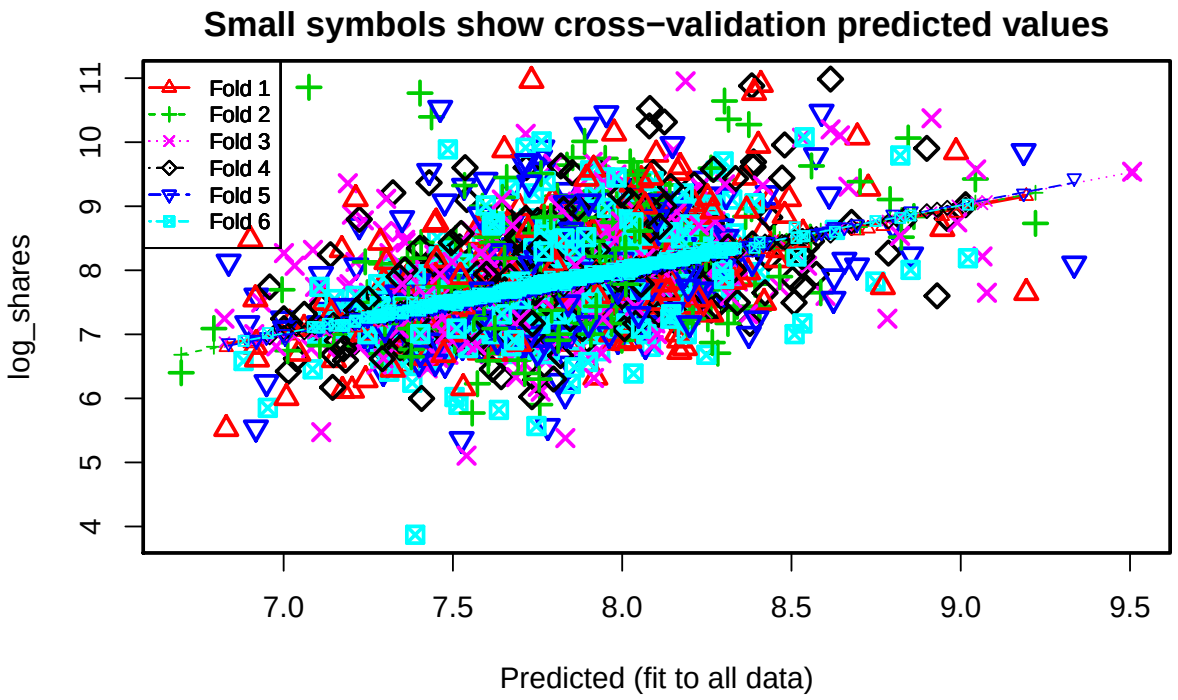
```
R2<-c(summary(backstep)$adj.r.squared,summary(forstep)$adj.r.squared,summary(intfit1)$adj.r.squared,summ
qplot(1:6, y=R2, geom="bar", stat="identity")
```

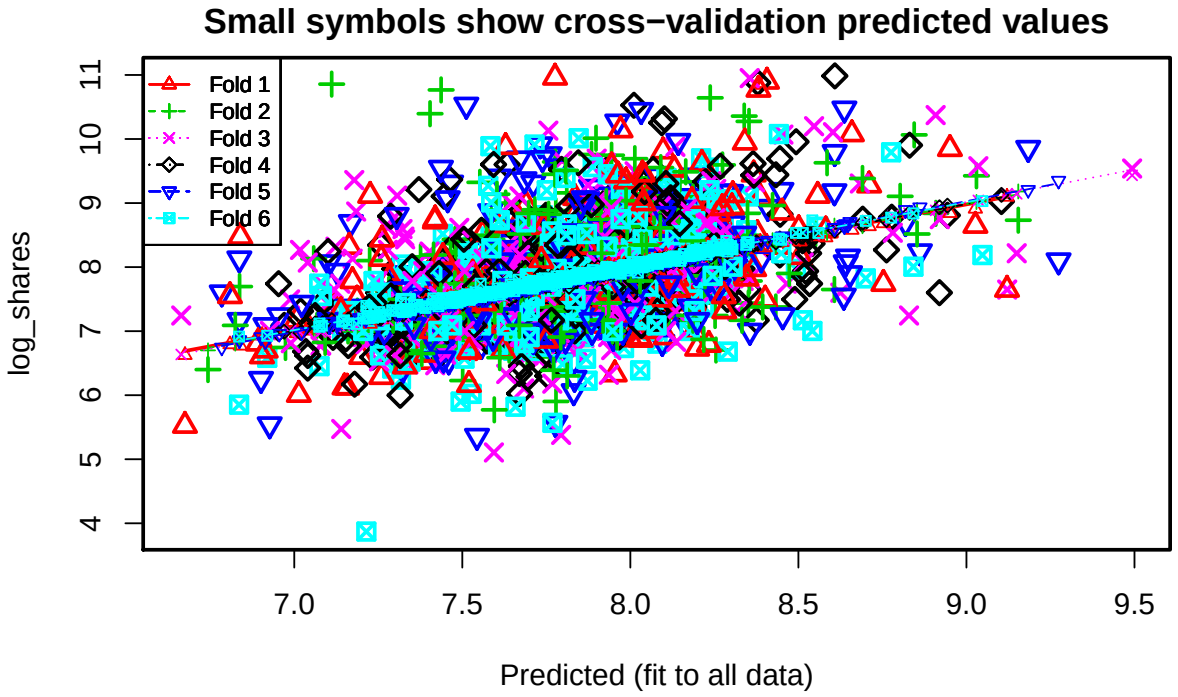


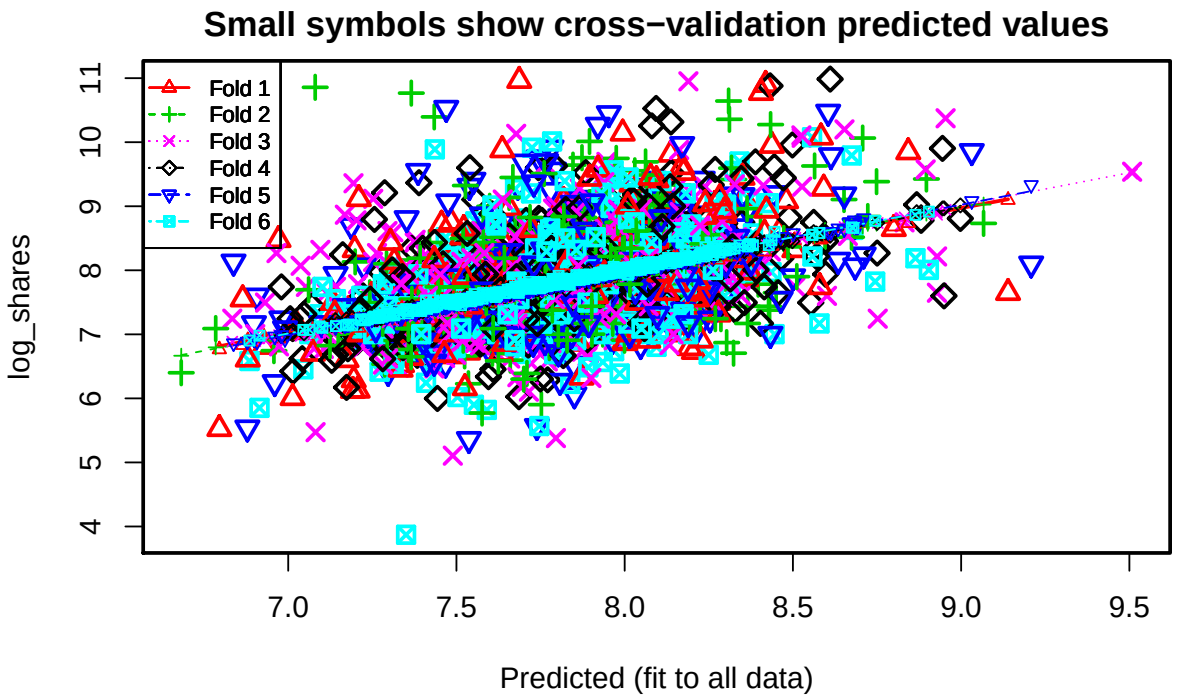
They are almost the same

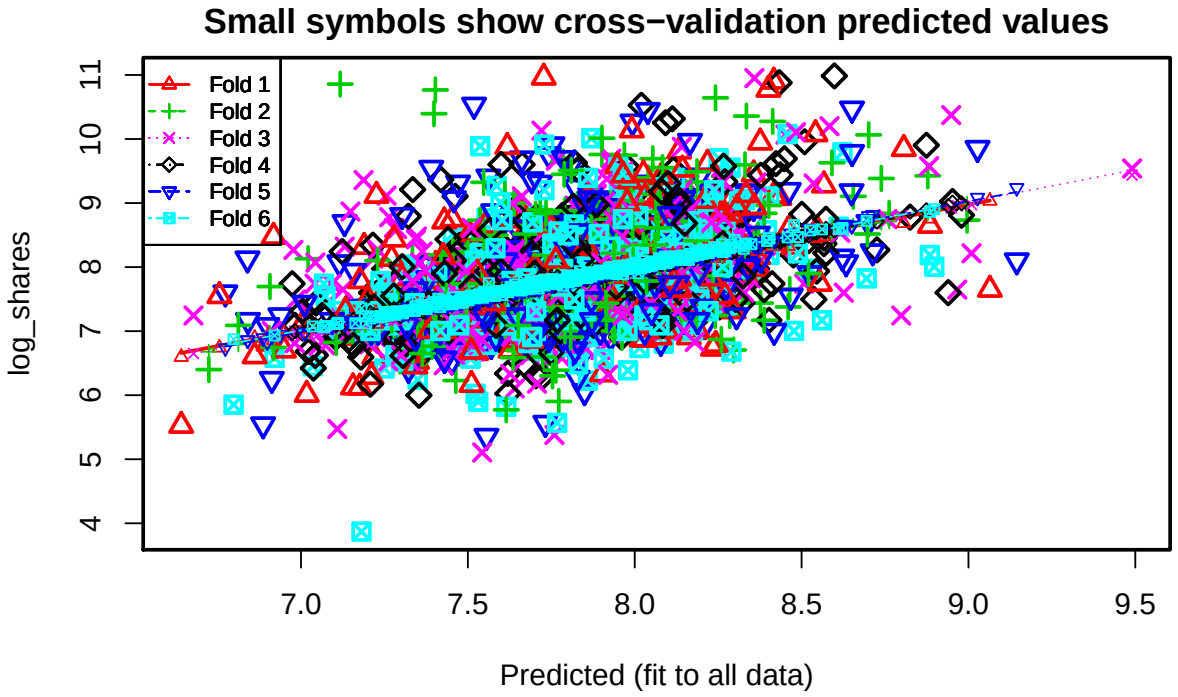
Now lets look at prediction power by cross validation of the six models

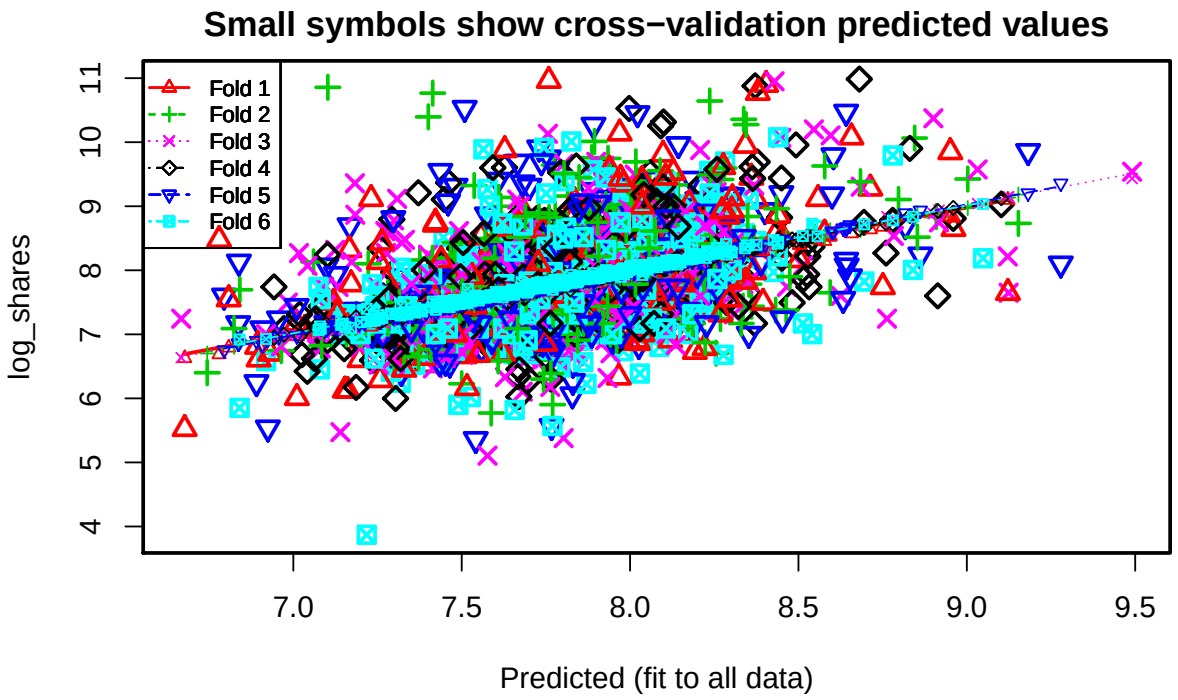
```
modellist <- list(backstep, forstep, intfit1, intfit2, intfit3, intfit4)
ms <- numeric()
df <- numeric()
for(i in 1:6){
  cv <- suppressWarnings(CVlm(data=soc, modellist[[i]], m=6, printit=F))
  ms <- c(ms, attributes(cv)$ms)
  df <- c(df, attributes(cv)$df)
}
```

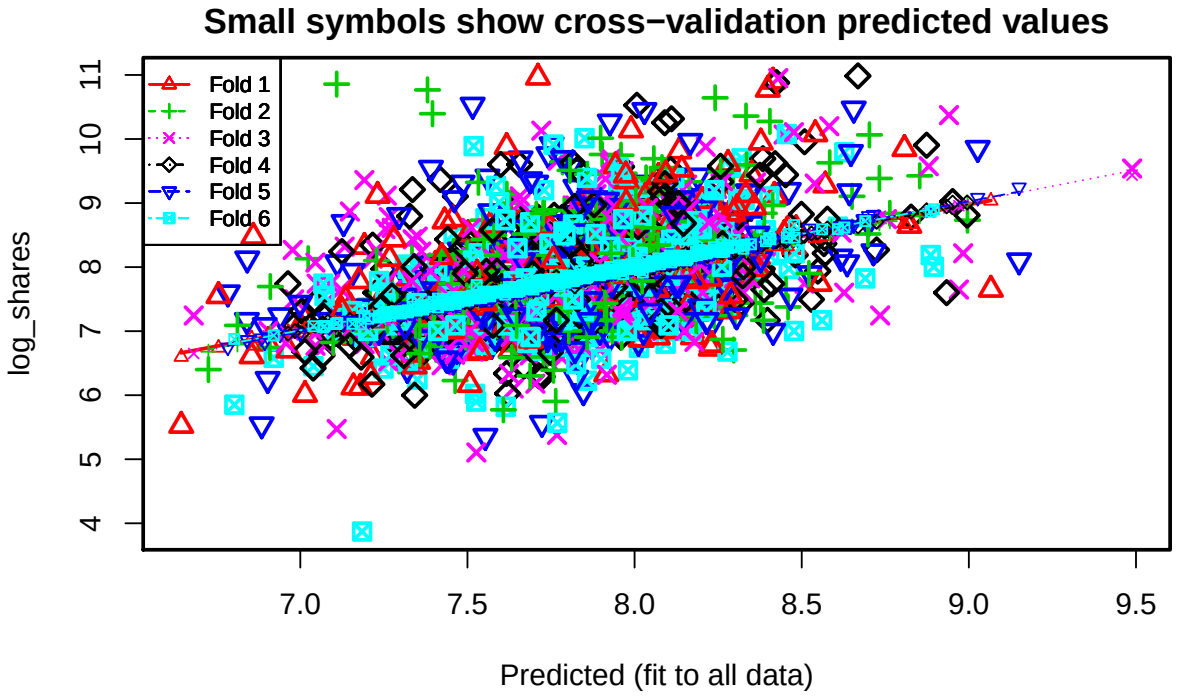

Small symbols show cross−validation predicted values
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
log_shares
4
5
6
7
8
9
10
11
7.0
7.5
8.0
8.5
9.0
9.5
Predicted (fit to all data)


Small symbols show cross−validation predicted values
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
log_shares
4
5
6
7
8
9
10
11
7.0
7.5
8.0
8.5
9.0
9.5
Predicted (fit to all data)


Small symbols show cross−validation predicted values
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
log_shares
4
5
6
7
8
9
10
11
7.0
7.5
8.0
8.5
9.0
9.5
Predicted (fit to all data)


Small symbols show cross−validation predicted values
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
log_shares
4
5
6
7
8
9
10
11
7.0
7.5
8.0
8.5
9.0
9.5
Predicted (fit to all data)






```
ms
```

```
## [1] 0.5325773 0.5331490 0.5317718 0.5324277 0.5336441 0.5329404
```

```
df
```

```
## [1] 2157 2157 2157 2157 2157 2157
```

6. What about using "leekasso"

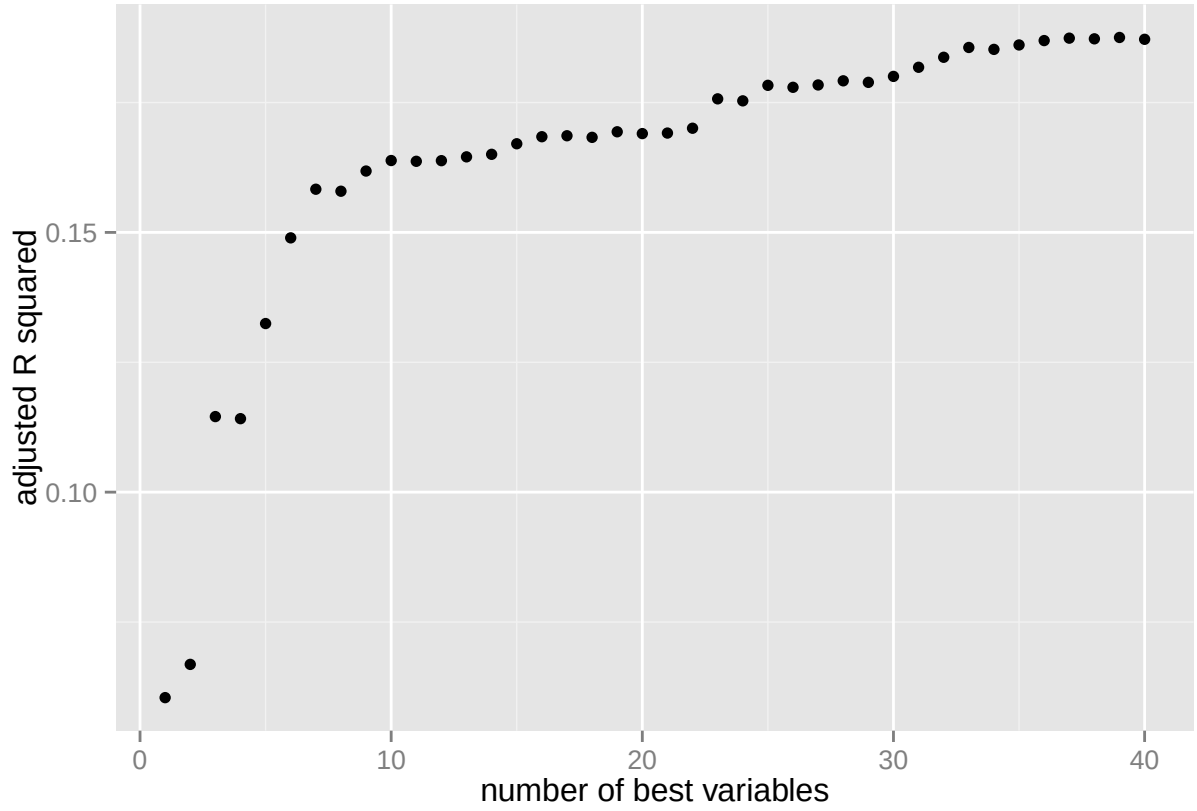
leekasso is a method chooses top 10 variables with least p-values in F-tests. Let's try this out

```
y <- soc$log_shares
x <- soc[,-41]
x0 <- rep(1, nrow(x))
n <- nrow(x)
projM <- function(x){m <- x %*% solve(t(x) %*% x) %*% t(x); return(m)}
Px0 <- projM(x0)
pvals <- data.frame(var = names(x), pval = rep(0, ncol(x)))
for(i in 1:ncol(x)){
  xa <- cbind(x0, x[,i])
  Pxa <- projM(xa)
  MS1 <- t(y) %*% (Pxa - Px0) %*%y
  MS2 <- t(y) %*% (diag(n) - Pxa) %*%y / (n-2)
  fstat <- MS1/MS2
  pval <- df(fstat, 1, n-2)
  pvals[i,2] <- pval
}
pv <- pvals[order(pvals[,2], decreasing=F),]
leekvar <- pv$var[1:10]
leekdat <- cbind(x[,names(x) %in% leekvar], y)
leekasso <-lm(y ~., data=leekdat)
summary(leekasso)
```

```
##
## Call:
## lm(formula = y ~ ., data = leekdat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4773 -0.4734 -0.1050  0.3758  3.8772
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    4.5295666  0.4915665   9.215  < 2e-16 ***
## n_unique_tokens               -1.1136935  0.3891767  -2.862  0.00426 **
## n_non_stop_unique_tokens       0.3354168  0.3998979   0.839  0.40170
## log_kw_avg_avg                 0.4619421  0.0607258   7.607 4.17e-14 ***
## sqrt_self_reference_min_shares 0.0020666  0.0007754   2.665  0.00775 **
## sqrt_self_reference_max_shares -0.0008498  0.0006249  -1.360  0.17403
## sqrt_self_reference_avg_sharess 0.0019766  0.0012590   1.570  0.11658
## log_LDA_00                     0.1352310  0.0135600   9.973  < 2e-16 ***
## log_LDA_01                    -0.0634083  0.0191756  -3.307  0.00096 ***
## avg_positive_polarity         -0.5222996  0.2097435  -2.490  0.01284 *
## min_positive_polarity         -0.7693897  0.2824643  -2.724  0.00650 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.738 on 2146 degrees of freedom
## Multiple R-squared:  0.1677, Adjusted R-squared:  0.1638
## F-statistic: 43.25 on 10 and 2146 DF,  p-value: < 2.2e-16
```

This method gave me 16% R squared. Not as better as previous model. However, it only contains 10 models. I am now interested in how R squared will change along the number of best variables I chose

```
arsq <- as.numeric()
ordvar<- pvals[order(pvals[,2], decreasing=F),]
for(i in 1:nrow(ordvar)){
  tempdata <- cbind(x[,names(x) %in% ordvar[1:i,1]], y)
  arsq <- c(arsq, summary(lm(y~., data=as.data.frame(tempdata)))$adj.r.squared)
}
qplot(x=1:ncol(x), y=arsq, geom="point", xlab="number of best variables", ylab = "adjusted R squared")
```



The R squared increased along the number of variable been involved but still cannot get over 20%.

Conclusion

Fit the best model in my hand

```
bestmodel <- modellist[order(ms)][[1]]
summary(bestmodel)
```

```
##
## Call:
## lm(formula = log_shares ~ timedelta + n_unique_tokens + sqrt_num_hrefs +
##     sqrt_num_imgs + sqrt_num_videos + num_keywords + log_kw_avg_avg +
```

```
##     sqrt_self_reference_min_shares + sqrt_self_reference_avg_sharess +
##     is_weekend + log_LDA_00 + log_LDA_01 + log_LDA_02 + log_LDA_03 +
##     log_LDA_04 + min_positive_polarity + max_positive_polarity +
##     min_negative_polarity + title_subjectivity + abs_title_subjectivity +
##     sqrt_self_reference_min_shares:sqrt_self_reference_avg_sharess,
##     data = soc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4791 -0.4575 -0.1015  0.3779  3.7769
##
## Coefficients:
##                                                                  Estimate
## (Intercept)                                                     3.425e+00
## timedelta                                                       3.747e-04
## n_unique_tokens                                                -1.195e+00
## sqrt_num_hrefs                                                 -4.943e-02
## sqrt_num_imgs                                                  -3.436e-02
## sqrt_num_videos                                                 3.533e-02
## num_keywords                                                    2.077e-02
## log_kw_avg_avg                                                  5.490e-01
## sqrt_self_reference_min_shares                                  3.778e-03
## sqrt_self_reference_avg_sharess                                 9.673e-04
## is_weekend1                                                     1.048e-01
## log_LDA_00                                                      1.003e-01
## log_LDA_01                                                     -6.664e-02
## log_LDA_02                                                     -2.391e-02
## log_LDA_03                                                     -4.757e-02
## log_LDA_04                                                     -3.779e-02
## min_positive_polarity                                          -9.679e-01
## max_positive_polarity                                          -2.666e-01
## min_negative_polarity                                          -1.328e-01
## title_subjectivity                                              1.979e-01
## abs_title_subjectivity                                          2.689e-01
## sqrt_self_reference_min_shares:sqrt_self_reference_avg_sharess -6.308e-06
##                                                                Std. Error
## (Intercept)                                                     5.736e-01
## timedelta                                                       8.474e-05
## n_unique_tokens                                                 1.927e-01
## sqrt_num_hrefs                                                  1.278e-02
## sqrt_num_imgs                                                   1.438e-02
## sqrt_num_videos                                                 1.856e-02
## num_keywords                                                    8.662e-03
## log_kw_avg_avg                                                  6.533e-02
## sqrt_self_reference_min_shares                                  8.315e-04
## sqrt_self_reference_avg_sharess                                 4.641e-04
## is_weekend1                                                     4.700e-02
## log_LDA_00                                                      1.862e-02
## log_LDA_01                                                      2.120e-02
## log_LDA_02                                                      1.647e-02
## log_LDA_03                                                      1.835e-02
## log_LDA_04                                                      1.598e-02
## min_positive_polarity                                           2.534e-01
## max_positive_polarity                                           8.689e-02
```

```
## min_negative_polarity                                          6.501e-02
## title_subjectivity                                             5.800e-02
## abs_title_subjectivity                                         9.867e-02
## sqrt_self_reference_min_shares:sqrt_self_reference_avg_sharess 2.941e-06
##                                                               t value
## (Intercept)                                                     5.972
## timedelta                                                       4.422
## n_unique_tokens                                                -6.200
## sqrt_num_hrefs                                                 -3.867
## sqrt_num_imgs                                                  -2.389
## sqrt_num_videos                                                 1.904
## num_keywords                                                    2.397
## log_kw_avg_avg                                                  8.404
## sqrt_self_reference_min_shares                                  4.543
## sqrt_self_reference_avg_sharess                                 2.084
## is_weekend1                                                     2.230
## log_LDA_00                                                      5.387
## log_LDA_01                                                     -3.143
## log_LDA_02                                                     -1.452
## log_LDA_03                                                     -2.592
## log_LDA_04                                                     -2.365
## min_positive_polarity                                          -3.819
## max_positive_polarity                                          -3.069
## min_negative_polarity                                          -2.043
## title_subjectivity                                              3.413
## abs_title_subjectivity                                          2.725
## sqrt_self_reference_min_shares:sqrt_self_reference_avg_sharess -2.145
##                                                               Pr(>|t|)
## (Intercept)                                                   2.74e-09
## timedelta                                                     1.03e-05
## n_unique_tokens                                               6.78e-10
## sqrt_num_hrefs                                                0.000114
## sqrt_num_imgs                                                 0.016961
## sqrt_num_videos                                               0.057108
## num_keywords                                                  0.016595
## log_kw_avg_avg                                                 < 2e-16
## sqrt_self_reference_min_shares                                5.84e-06
## sqrt_self_reference_avg_sharess                               0.037268
## is_weekend1                                                   0.025823
## log_LDA_00                                                    7.93e-08
## log_LDA_01                                                    0.001697
## log_LDA_02                                                    0.146619
## log_LDA_03                                                    0.009614
## log_LDA_04                                                    0.018114
## min_positive_polarity                                         0.000138
## max_positive_polarity                                         0.002177
## min_negative_polarity                                         0.041154
## title_subjectivity                                            0.000655
## abs_title_subjectivity                                        0.006478
## sqrt_self_reference_min_shares:sqrt_self_reference_avg_sharess 0.032054
##
## (Intercept)                                                   ***
## timedelta                                                     ***
## n_unique_tokens                                               ***
```

```
## sqrt_num_hrefs                                                  ***
## sqrt_num_imgs                                                   *
## sqrt_num_videos                                                 .
## num_keywords                                                    *
## log_kw_avg_avg                                                  ***
## sqrt_self_reference_min_shares                                  ***
## sqrt_self_reference_avg_sharess                                 *
## is_weekend1                                                     *
## log_LDA_00                                                      ***
## log_LDA_01                                                      **
## log_LDA_02
## log_LDA_03                                                      **
## log_LDA_04                                                      *
## min_positive_polarity                                           ***
## max_positive_polarity                                           **
## min_negative_polarity                                           *
## title_subjectivity                                              ***
## abs_title_subjectivity                                          **
## sqrt_self_reference_min_shares:sqrt_self_reference_avg_sharess *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7259 on 2135 degrees of freedom
## Multiple R-squared:  0.199,  Adjusted R-squared:  0.1911
## F-statistic: 25.25 on 21 and 2135 DF,  p-value: < 2.2e-16
```

This analysis is about online news popularity. I tried to explore the provided variables and find a linear model to explain why some news have more shares whiles others dont. Without transformation, throw all variables into a lm function will only give you .02 R squared. Which indicates irrelevant information been used. After transformation and variable selection, I can at best increase the R squared to around 0.2. So I think this data set is not appropriate to analyze news popularity, more information needed, which make sense because all these data are from superficial text mining results. They are very general and we know that, people like to share different news at different time according to many other events occuring all over the world, so without the variable measuring the media environment, it is hard to believe there is a general rule that one kind of article will draw more attention than others.

Thanks for reading!