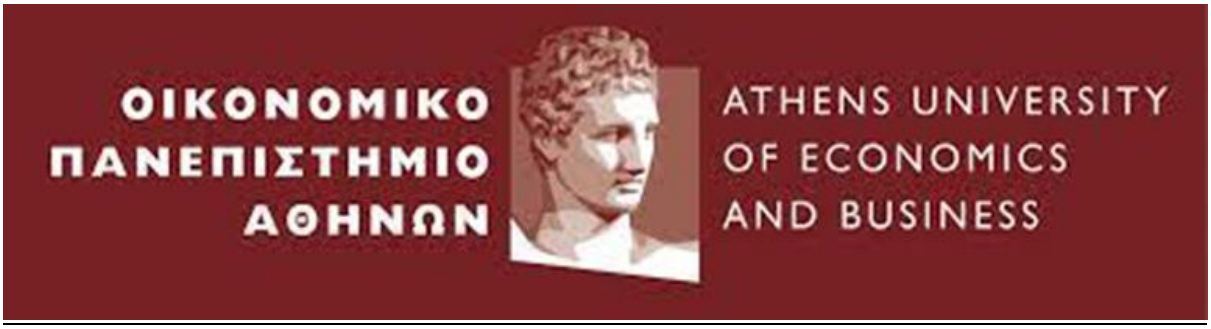


ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
DEPARTMENT OF MANAGEMENT, SCIENCE & TECHNOLOGY
MSc BUSINESS ANALYTICS

"MASHABLE POPULARITY NEWS ASSIGNMENT"
"Training Data Set: alldata_onlinenews_21"

Full Name: ATHANASIOS ALEXANDRIS
Register Number:p2822202

ATHENS, 2023

TABLE OF CONTENTS

1.INTRODUCTION

1.1 Description of the problem

The data of this assignment refer to characteristics of the popular website of Mashable (www.mashable.com) downloaded in January 2015. The main variable of the study is the number of shares which measures the popularity of the site/post. We are interested to identify the ingredients of a successful post and what it takes to for a post to become a viral. In our study we try to identify the best model that predicts the number of shares of a post.

1.2 TRAINING DATASET

For our study we used two datasets, one for for training the model and another for evaluating the model. The first one included 3.000 records and the second one 10.000 records. The dependent variable of the study is the number of shares, which serves as a measure of the post's popularity, and there are 60 independent variables (58 explanatory attributes, 2 non-explanatory).

2. EXPLORATORY/DESCRIPTIVE ANALYSIS

2.1 Data cleaning

Before we start our analysis we had to clean and modify our dataset, in a way that would help us to our next steps to make an effective and sufficient analysis. Below are listed all actions we did regarding the data cleaning :

1. We removed the following columns:
X: this column is not listed in the description of the dataset and we do not have sufficient information for this column
url: non explanatory column, with information not useful for our analysis
timedelta: non explanatory column, with information not useful for our analysis
is_weekend: this column give information that we already have from columns weekday_is_saturday and weekday_is_sunday.
2.We checked for blanks, NAs, NaN or infinite values, but we did not find any of these.
3. We set the int columns as numeric.
4.We set the categorical columns (nominal) as factors.
By the end of the cleaning 58 variables had been remained to our dataset, 45 numerical variables and 13 factors.

2.2 Visualizations

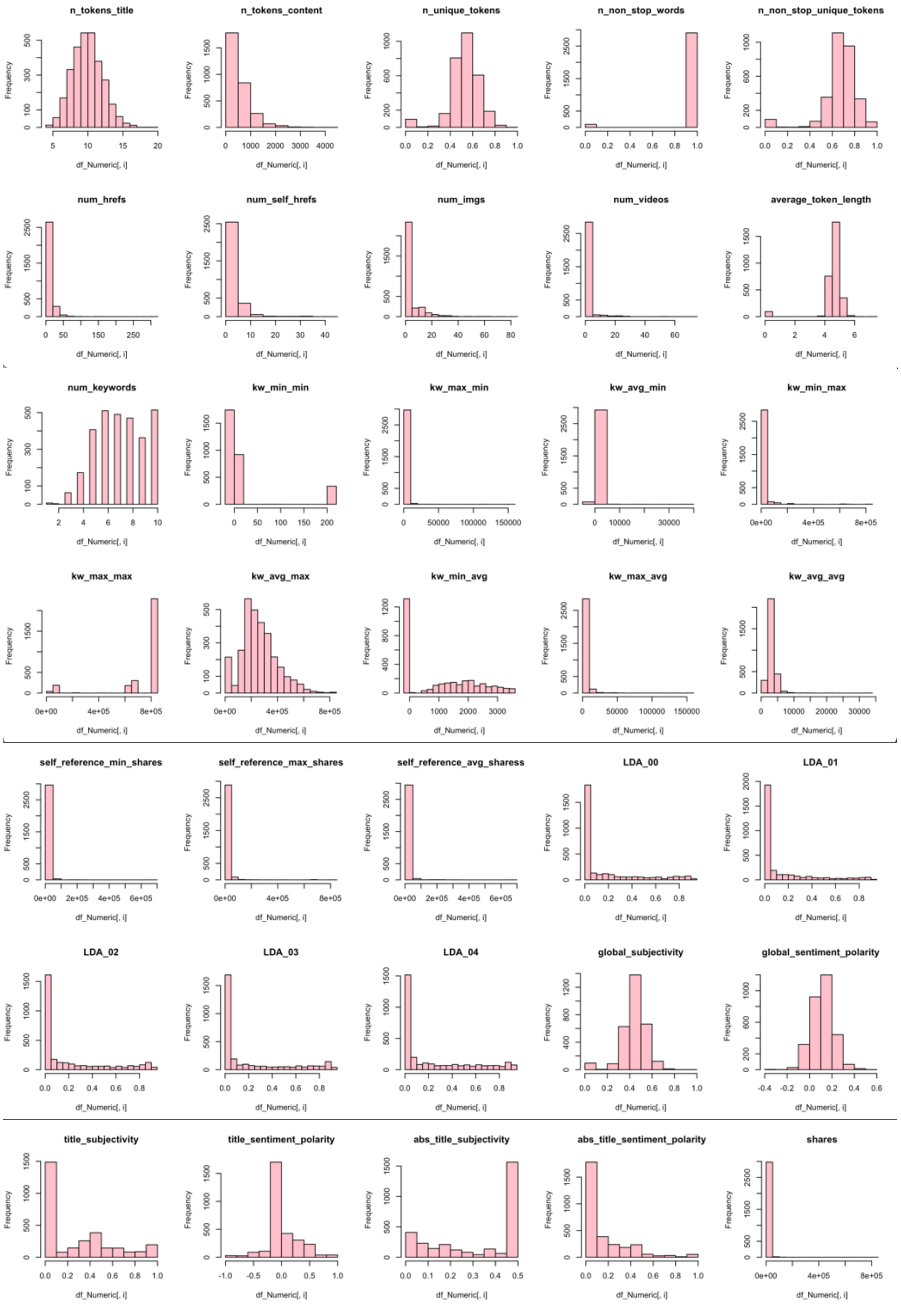
2.2.1 Numerical variables

Regarding the numerical variables, we made histograms to understand the type of distribution and the symmetry of each variable. By observing Figure 1 we can assume the following:

1. The variables: n_tokens_title, n_unique_tokens, global_subjectivity, global_sentiment_polarity, avg_positive_polarity, seem to follow normal distribution.

2. The rest numerical variables seem to have asymmetry and significant positive/negative skewness. It is important to mention at this point specially the significant skewness of our dependent variable ‘shares’.

Figure 1



2.2.2 Factor variables

In Figure 2 we have designed barplots for the factor variables.

Figure 2.

It seems that the days of the week when most of the articles tend to be published are the working days of the week, and especially the three middle working days (Tuesday to Thursday). Mondays and Fridays seem to have slightly lower frequency, and finally Saturday and Sunday are the days with the less observations.

Regarding the type of data channel, in first place of frequency seem to be the world category, slightly second the tech category, next entertainment and then the categories bus and unknown have smaller but important frequency.

As we have mentioned before, our dependent variable has a significant right skewness, so we decided to transform the values of this variable to logarithms, as we believe it will help us to our next steps.

2.3 Pairwise comparisons

2.3.1 Comparisons between 'shares' and numerical variables

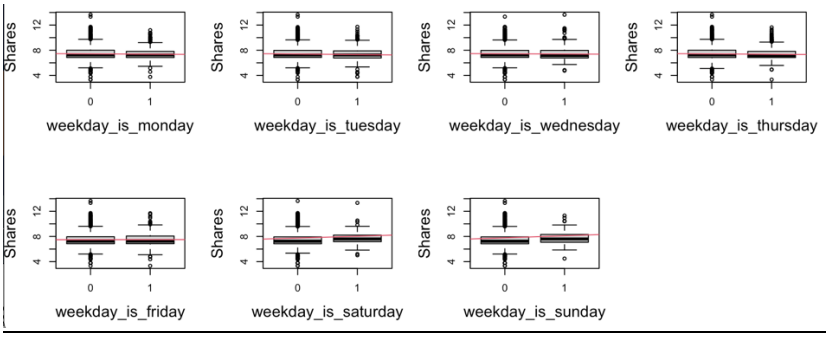
At this point we made the visualizations of bivariate associations, between the log of our dependent variable 'shares' and the rest numerical variables. By observing the diagrams on the left of Figure 3 we can say that 'shares' variable does not have a linear relationship with any of the numerical variables.

The analysis of the corplots on the right side, lead us to the outcome, that our dependent variable has not significant correlation with any of the other variables. The log_shares variable has the strongest correlations (|10%|<=r<=|19%|) with variables 'kw_avg_avg' (r=18%), 'LDA_02' (r=-17%), num_hrefs (14%), num_imgs (13%), LDA_03 and self_reference_max_shares (11%), self_reference_max_shares (10%), kw_min_ avg (10%).

2.3.2 Comparisons between 'shares' and factor variables

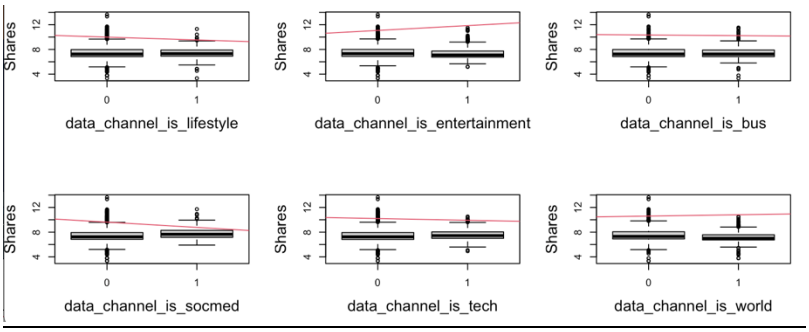
In Figure 4 we designed box plots for the categorical variables that indicate the day of the post in comparison with the response, log(shares). As we can observe log(shares) seem to be higher for articles that have been posted on days 'Saturday' and 'Sunday' in comparison with the rest working days that we observe that they do not have important impact to our dependent variable log(shares).

Figure 4



In Figure 5 we designed box plots for the categorical variables that indicate the channel type in comparison with the response log(shares). As we can observe log(shares) seem to be higher for posts of type ‘entertainment’ and ‘world’. For the types ‘bus’ and ‘tech’ log(shares) seem to be slightly lower and finally for the rest types ‘lifestyle’ and ‘socmed’ log(shares) end to be much lower.

Figure 5

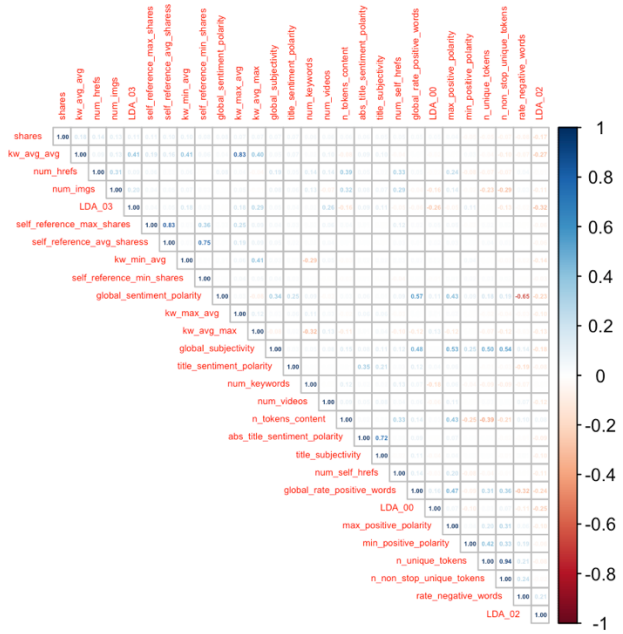


3.PREDICTIVE/DESCRIPTIVE MODELS

3.1 Attribute selection

In order to identify the most significant numerical variables for our model, we designed (Figure 4) a corrplot to identify the most strong correlations attributes to our response 'shares'. As we found out that the response variable has very low correlations with the other variables we decided to set a low threshold, correlation higher of 0.04 or lower than -0.04 to the variable 'shares'.

Figure 4.

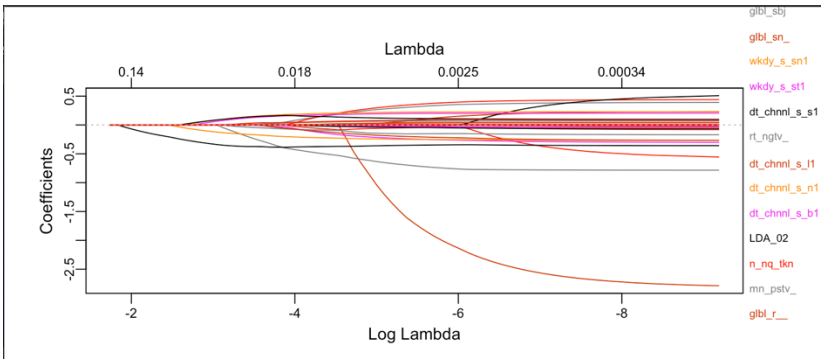


3.2 Prediction Model

Next, we used as input for the lasso the 28 highly correlated variables that are listed in Table1. We use lasso to remove additional variables that are not significant for our model and also get the remained variables regularized.

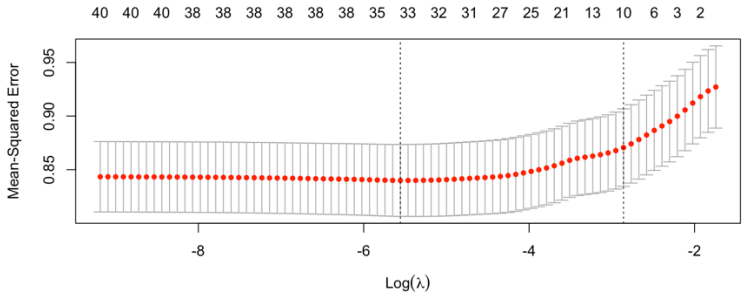
First, we designed the plot for lasso with the coefficients against the value of Log Lambda and, as shown in Figure 5.

Figure 5.



Moreover the diagramm Figure 6 shows us that Lasso chooses 10 variables out of 40 (27 numeric, 13 factors) that we give it as input. We decided to use lambda.1se (right vertical line), as we can have a more clear model and more regularized model.

Figure 6



Then we implemented the AIC method, which is more appropriate for predictions, in order to end up to the most effective model. We ended up with a model with 9 variables, where the AIC has its minimum value (-520.45) , as shown in Table 2.

Finally, we used VIF to understand if we have multicollinearity in our model. However, all coefficients of our model are below 10, so we accept them all (Table 3).

The summary of our model (Table 4), shows us that all the covariates have p-value lower than 0.05, so we cannot remove additional variables. Also, the adjusted R-squared is really low (0,09604) but it has the highest values of all the models, that we have produced.

As a last step we tried to remove the intercept from our model. Although, the summary of this model showed R-squared 0.9853, the real adjusted R-squared was 0.09875, so the impact of removing the intercept is really small, and so decided not remove it.

The mathematical formulation of our model described below:

Log(shares)= 7.192+ 8.149e-05*kw_avg_avg + 8.142e03*num_imgs +3.047e06*self_reference_avg_sharess – -5.046e-01*LDA_02 + 2.540e01*weekday_is_saturday1 + 2.656e-01+weekday_is_sunday1 - 2.717e01*data_channel_is_entertainment1 + 2.387e-01*data_channel_is_socmed1 + ε ~ N(0, 0.9154)

3.3 Model Assumptions

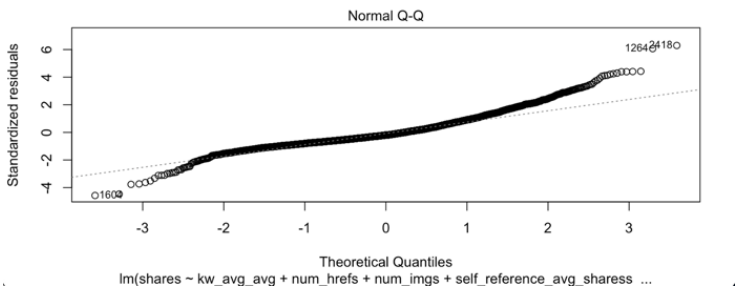
At this point we must check the assumptions of the linear regression model. These assumptions are the following:

- Normality of residuals
- Homoscedasticity of residuals variance
- Linearity of residuals
- Independence of residuals error terms

I. Normality of residuals

The tests we performed for normality assumption have both p-value below 0.05 (Lilliefors KS p = 2.22e-16 < 0.05, Shapiro-Wilk p =2.22e-16 < 0.05), so the Ho, that the residuals come from a normally distributed population is rejected (Table 5). Moreover, since the observations are more than 50, we also designed a QQ (Figure 7) plot, to evaluate the normality assumption. The residuals do not lie exactly on this line, so this an additional clue that there is not normality.

Figure 7.

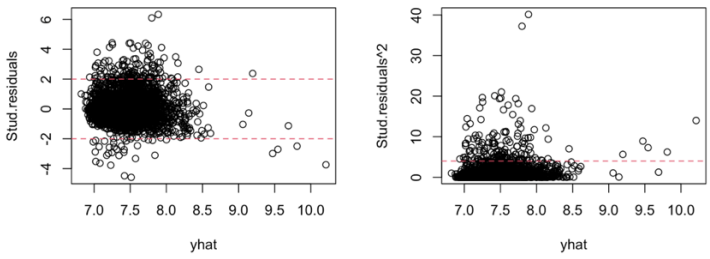


II. Homoscedasticity of residuals variance

Moreover, from Figure 8 it can be observed that the variances of the residuals tend to increase as the value of the fitted outcome variable increases, and that indicates non-constant variances in the residuals' errors, so meaning there is heteroscedasticity. Also we executed Levene and

ncvTest (Table 6), where the p-values were below of 0.05 for both of them, so we must reject the Ho that the variance of the residuals is constant and conclude that there is heteroscedasticity.

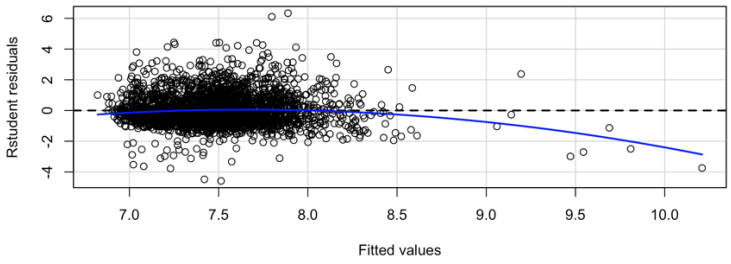
Figure 8



III. Linearity of the data.

The assumption of linearity of the residuals can be rejected by the plot where we can easily observe that they are not linear related, since the blue line of the residuals do not follow the straight black line of linearity (Figure 9).

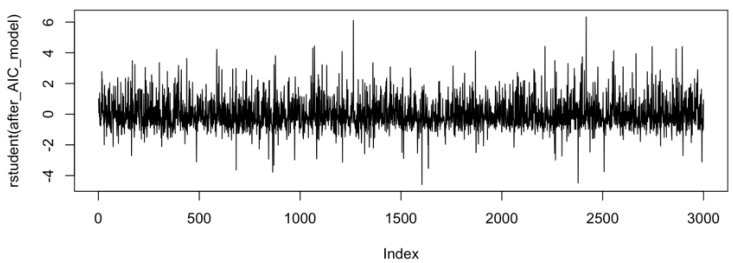
Figure 9



IV. Independence of residuals error terms.

We executed Durbin Watson test and the p-value was 0.728 >0.05 (Table 7), so we do not reject the Ho, that the autocorrelation of the disturbances is 0 (Figure 10). As a result, the residuals may have a zero autocorrelation.

Figure 10



The conclusion of the assumptions check is that only the IV assumption is not violated. In order to fix the violations, we recommend creating a new model including polynomials, as we have already put log in our response.

3.4 Polynomials model and assumptions

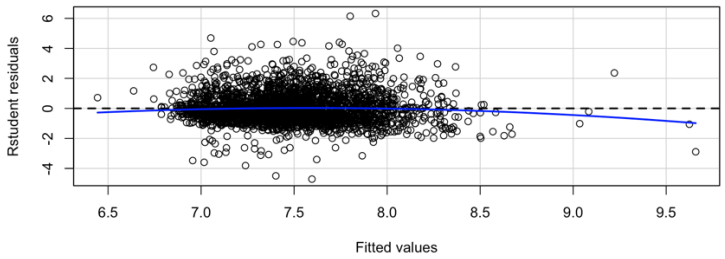
Trying to fix the violations of assumptions that we discussed above, we created a new model with polynomials up to fifth level. At Table 8 the summary of the new model is described.

The mathematical formulation of our model described below:

Log(shares)= 6.687+ 3.279e-04* kw_avg_avg -2.667e-08 * kw_avg_avg^2 + 5.050e-13* kw_avg_avg^3 + 7.758e-03*num_hrefs + 7.183e-03*num_imgs + 3.350e-06*self_reference_avg_sharess -4.092e-01*LDA_02 + 2.365e-01*weekday_is_saturday1+ 2.525e-01*weekday_is_Sunday1 - 2.769e-01data_channel_is_entertainment1 + 2.478e-01* data_channel_is_socmed1 + $\varepsilon \sim N(0, 0.9089)$

Once again, we performed the test for the four assumptions we have discussed. The p-values for the normality tests are below 0.05 (Lilliefors KS p = 2.2e-16, Shapiro-Wilk p = 2.2e-16) so the Ho is rejected (Table 9). The p-value for homoscedasticity test is below 0.05 (ncvTest p = 2.22e-16 and Levene p = 4.645e-16) (Table 10). On the other hand, the assumption of linearity of the residuals cannot be rejected by the plot where we can easily observe that they are linear related, since the blue line of the residuals do follow the straight black line of linearity (Figure 11). Finally, we executed Durbin-Watson test and the p-value was 0.468 >0.05 (Table 11), so we do not reject the Ho, that the autocorrelation of the disturbances is 0. As a result, the residuals may have a zero autocorrelation.

Figure 11



As conclusion for the second model, we can say that only linearity fixes and the assumptions of normality and heteroscedasticity are still violated.

4.MODEL EVALUATION

4.1 LOOCV & 10-fold cross validation methods

The last part of our project is to test the model that we have selected on a new dataset containing 10,000 observations, with the same format, but with different data. So, after the transformations we did to the new dataset, in order to be in the same format with the train dataset, we implemented the leave-one-out and 10-fold cross-validation on our test dataset, to identify the model with the best predictability, amongst the two models that we presented earlier. The criteria for our decision is based on the combination with a low RMSE and a high R-squared.

Amongst the two models we decided to select the second model, as for both methods it has lower RMSE(0.8805<0.8848 , 0.8808<0.885) and higher R-squared(0.1067> 0.10012, 0.10548 > 0.09678) (Table 12). Also, a really important factor that drove us to this selection is that the second model is the one which satisfies the most Regression assumptions.

Concluding, although the model we have selected it is better than the first one, it has a low predictability, and it is not safe for predictions.

4.2 Interpretation of the model

The summary of our model is presented in Table 8.

The mathematical formulation of our model described below:

Log(shares)= 6.687+ 3.279e-04* kw_avg_avg -2.667e-08 * kw_avg_avg^2 + 5.050e-13* kw_avg_avg^3 + 7.758e-03*num_hrefs + 7.183e-03*num_imgs + 3.350e-06*self_reference_avg_sharess -4.092e-01*LDA_02 + 2.365e-01*weekday_is_saturday1+ 2.525e-01*weekday_is_Sunday1 - 2.769e-01data_channel_is_entertainment1 + 2.478e-01* data_channel_is_socmed1 + $\varepsilon \sim N(0, 0.9089)$

Interpretation:

Constant is the expected value of the dependent variable when all predictors are zero. In our case when all independent variables will be 0 the log(shares) will be 6.687.

Regarding the interpretation of the coefficients, if we increase by 1 unit one of them, without increase or decrease any of the rest predictors, the log(shares) will be increased by bi. For example, if we increase the ‘num_hrefs’ by 1, this log(shares) will be increased by 7.183e-3.

The Residual Standard Error is an indicator of the model's fit to the data. In our case, the Residual Standard Error has a very large value (0.9089), that shows us that the model prediction is not accurate.

The p-value of the independent variables is the probability the variable not to be relevant. If the p-value is smaller than 0.05, indicate that we must reject the null hypothesis, that the coefficient must be equal to 0. In our model all variables have p-value less than 0.05, so they are all important for the value of the response.

The Adjusted R-squared shows at which level the data are explained by the model. In our case the Adjusted R-squared is very low (0.1088) , and indicates that the model does not fit good to our data.

5. CONCLUSION

The aim of this project was to construct a model for making predictions, regarding articles popularity and especially the prediction of the number of shares for the articles depending on the variables of the train dataset that we had. Finally, we ended up to a linear regression model, with 11 predictors and the constant. Our model has a low Adjusted R-squared 0.1088, which practically means that nearly 11% of the variance is explained in our model. The model satisfies the assumptions of linearity of residuals and independence of residuals error terms, but violates the assumptions of normality of the residuals and Homoscedasticity of residuals variance. Our best model has very high RMSE, so it has not great ability to predictions.

Finally, from our analysis we understand that a post might have good chances to be viral if the following conditions are satisfied:

- The average number of shares of the average keyword of the article to be high
- The day of the post is Saturday or Sunday
- The category of the data channel is ‘socmed’
- The post contains high number of images
- The average number of shares of referenced articles in Mashable is high.
- The topic of the article has not to be close to category LDA_02

Appendix A: Tables and Figures

Table 1

```
> high_correlations
 [1] "shares"                        "kw_avg_avg"
 [3] "num_hrefs"                     "num_imgs"
 [5] "LDA_03"                        "self_reference_max_shares"
 [7] "self_reference_avg_sharess"    "kw_min_avg"
 [9] "self_reference_min_shares"     "global_sentiment_polarity"
[11] "kw_max_avg"                    "kw_avg_max"
[13] "global_subjectivity"           "title_sentiment_polarity"
[15] "num_keywords"                  "num_videos"
[17] "n_tokens_content"              "abs_title_sentiment_polarity"
[19] "title_subjectivity"            "num_self_hrefs"
[21] "global_rate_positive_words"    "LDA_00"
[23] "max_positive_polarity"         "min_positive_polarity"
[25] "n_unique_tokens"               "n_non_stop_unique_tokens"
[27] "rate_negative_words"           "LDA_02"
```

Table 2

```
Step:  AIC=-520.45
shares ~ kw_avg_avg + num_hrefs + num_imgs + self_reference_avg_sharess +
    LDA_02 + weekday_is_saturday + weekday_is_sunday + data_channel_is_entertainment +
    data_channel_is_socmed

                               Df Sum of Sq    RSS     AIC
<none>                                      2505.4 -520.45
+ self_reference_max_shares     1     0.479 2505.0 -519.02
- data_channel_is_socmed        1     9.091 2514.5 -511.58
- num_imgs                      1    11.032 2516.5 -509.27
- weekday_is_saturday           1    11.109 2516.5 -509.18
- weekday_is_sunday             1    12.459 2517.9 -507.57
- self_reference_avg_sharess    1    12.874 2518.3 -507.07
- num_hrefs                     1    25.872 2531.3 -491.63
- data_channel_is_entertainment 1    30.278 2535.7 -486.41
- kw_avg_avg                    1    30.756 2536.2 -485.85
- LDA_02                        1    54.007 2559.4 -458.47
```

Table 3

```
> vif(after_AIC_model)
                    kw_avg_avg                       num_hrefs
                      1.123000                        1.113653
                    num_imgs     self_reference_avg_sharess
                      1.136165                        1.030516
                      LDA_02            weekday_is_saturday
                      1.145489                        1.008336
          weekday_is_sunday  data_channel_is_entertainment
                      1.017332                        1.077836
       data_channel_is_socmed
                      1.021337
```

Table 4

```
Call:
lm(formula = shares ~ kw_avg_avg + num_hrefs + num_imgs + self_reference_avg_sharess +
    LDA_02 + weekday_is_saturday + weekday_is_sunday + data_channel_is_entertainment +
    data_channel_is_socmed, data = dataset_21)

Residuals:
    Min      1Q  Median      3Q     Max
-4.1813 -0.5723 -0.1828  0.4382  5.7565

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                   7.192e+00  5.234e-02 137.401  < 2e-16 ***
kw_avg_avg                    8.149e-05  1.345e-05   6.058 1.55e-09 ***
num_hrefs                     8.142e-03  1.465e-03   5.557 2.99e-08 ***
num_imgs                      8.541e-03  2.354e-03   3.628 0.000290 ***
self_reference_avg_sharess    3.047e-06  7.773e-07   3.920 9.06e-05 ***
LDA_02                       -5.046e-01  6.285e-02  -8.028 1.41e-15 ***
weekday_is_saturday1          2.540e-01  6.977e-02   3.641 0.000276 ***
weekday_is_sunday1            2.656e-01  6.887e-02   3.856 0.000118 ***
data_channel_is_entertainment1 -2.717e-01  4.520e-02  -6.011 2.07e-09 ***
data_channel_is_socmed1       2.387e-01  7.245e-02   3.294 0.001000 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9154 on 2990 degrees of freedom
Multiple R-squared:  0.09875,   Adjusted R-squared:  0.09604
F-statistic:  36.4 on 9 and 2990 DF,  p-value: < 2.2e-16
```

Table 5

```
> lillie.test(res)

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  res
D = 0.093083, p-value < 2.2e-16

> shapiro.test(res)

        Shapiro-Wilk normality test

data:  res
W = 0.93643, p-value < 2.2e-16
```

Table 6

```
> ncvTest(after_AIC_model)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 98.82563, Df = 1, p = < 2.22e-16
> yhat.quantiles<-cut(yhat, breaks=quantile(yhat, probs=seq(0,1,0.25)), dig.lab=6)
> table(yhat.quantiles)
yhat.quantiles
(6.82178,7.26867] (7.26867,7.45675] (7.45675,7.63583] (7.63583,10.2095]
              749               750               750               750
> leveneTest(rstudent(after_AIC_model)~yhat.quantiles)
Levene's Test for Homogeneity of Variance (center = median)
       Df F value    Pr(>F)
group   3   21.09 1.617e-13 ***
      2995
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Table 7

```
> durbinWatsonTest(after_AIC_model)
 lag Autocorrelation D-W Statistic p-value
   1    -0.007744416      2.014865   0.728
 Alternative hypothesis: rho != 0
```

Table 8

```
Call:
lm(formula = shares ~ kw_avg_avg + I(kw_avg_avg^2) + I(kw_avg_avg^3) +
    num_hrefs + num_imgs + self_reference_avg_sharess + LDA_02 +
    weekday_is_saturday + weekday_is_sunday + data_channel_is_entertainment +
    data_channel_is_socmed, data = dataset_21)

Residuals:
    Min      1Q  Median      3Q     Max
-4.2636 -0.5678 -0.1760  0.4353  5.7078

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    6.687e+00  1.004e-01  66.593  < 2e-16 ***
kw_avg_avg                     3.279e-04  4.774e-05   6.869 7.85e-12 ***
I(kw_avg_avg^2)               -2.667e-08  6.080e-09  -4.387 1.19e-05 ***
I(kw_avg_avg^3)                5.050e-13  1.499e-13   3.370 0.000761 ***
num_hrefs                      7.758e-03  1.456e-03   5.328 1.07e-07 ***
num_imgs                       7.183e-03  2.346e-03   3.062 0.002217 **
self_reference_avg_sharess     3.350e-06  7.804e-07   4.292 1.82e-05 ***
LDA_02                        -4.092e-01  6.403e-02  -6.391 1.91e-10 ***
weekday_is_saturday1           2.365e-01  6.932e-02   3.411 0.000656 ***
weekday_is_sunday1             2.525e-01  6.841e-02   3.691 0.000228 ***
data_channel_is_entertainment1 -2.769e-01  4.489e-02  -6.169 7.81e-10 ***
data_channel_is_socmed1        2.478e-01  7.217e-02   3.434 0.000603 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9089 on 2988 degrees of freedom
Multiple R-squared:  0.1121,    Adjusted R-squared:  0.1088
F-statistic:  34.3 on 11 and 2988 DF,  p-value: < 2.2e-16
```

Table 9

```
> lillie.test(res2)

        Lilliefors (Kolmogorov-Smirnov) normality test

data:  res2
D = 0.092982, p-value < 2.2e-16

> shapiro.test(res2)

        Shapiro-Wilk normality test

data:  res2
W = 0.93863, p-value < 2.2e-16
```

Table 10

```
> ncvTest(model2)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 76.72054, Df = 1, p = < 2.22e-16
> leveneTest(rstudent(model2)~yhat.quantiles2)
Levene's Test for Homogeneity of Variance (center = median)
        Df F value    Pr(>F)
group    3  25.139 4.645e-16 ***
      2995
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

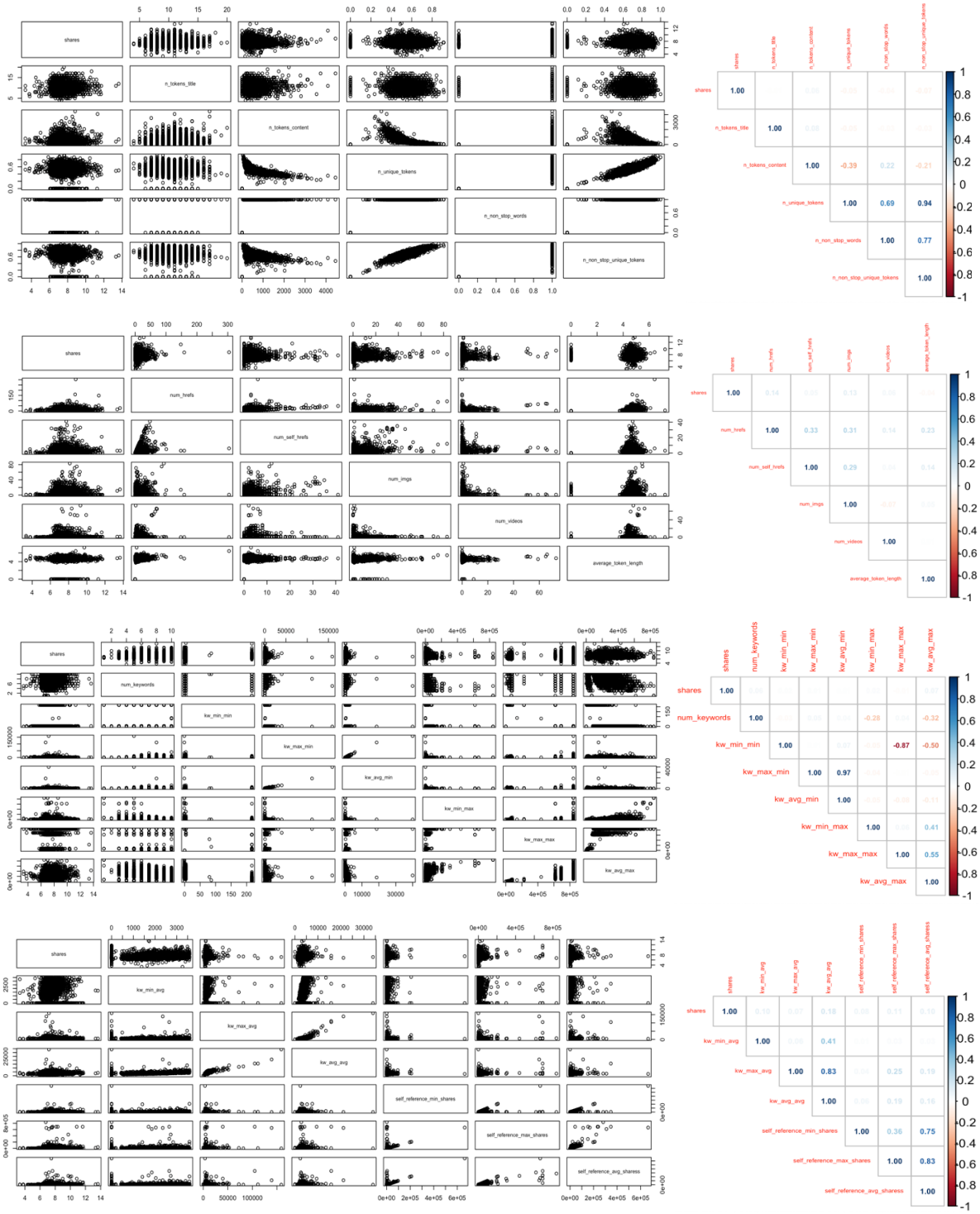
Table 11

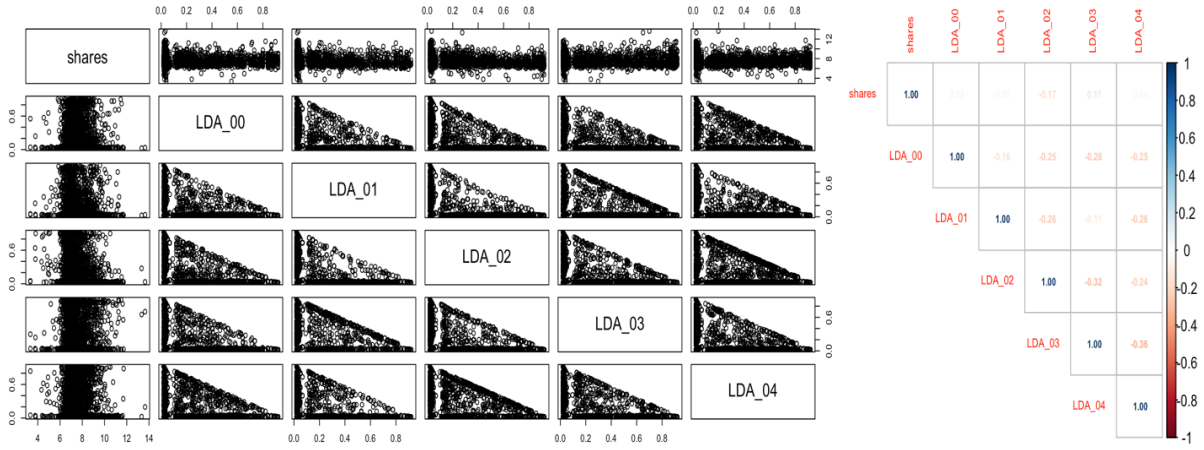
```
> durbinWatsonTest(model2)
 lag Autocorrelation D-W Statistic p-value
   1     -0.01391233      2.027127   0.468
 Alternative hypothesis: rho != 0
```

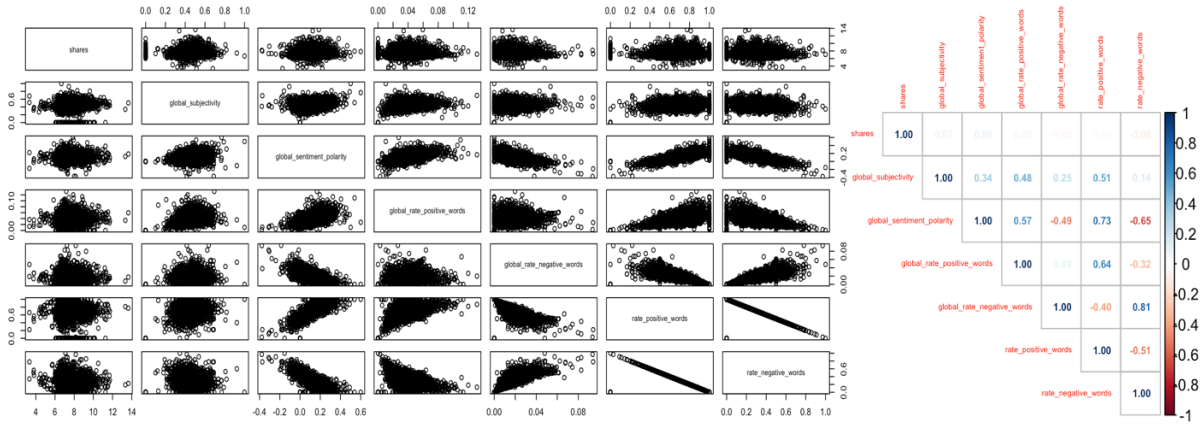
Table 12

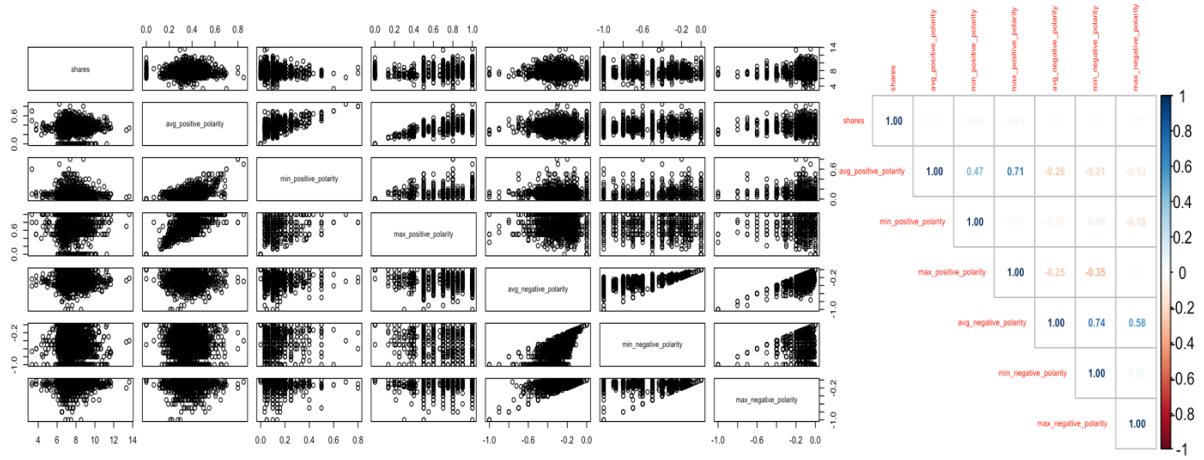
```
> results
              intercept      RMSE   Rsquared       MAE
model_cv_10_1      TRUE 0.8848125 0.10012408 0.6588026
model_LOOCV_1      TRUE 0.8850674 0.09678675 0.6587589
model_cv_10_2      TRUE 0.8805020 0.10673961 0.6567443
model_LOOCV_2      TRUE 0.8807918 0.10548199 0.6567894
```

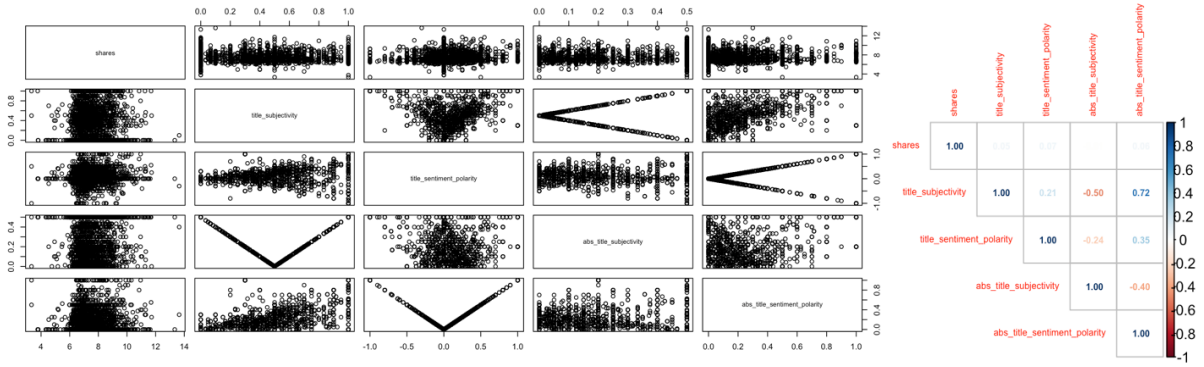
Figure 3




shares
LDA_00
LDA_01
LDA_02
LDA_03
LDA_04


shares
global_subjectivity
global_sentiment_polarity
global_rate_positive_words
global_rate_negative_words
rate_positive_words
rate_negative_words


shares
avg_positive_polarity
min_positive_polarity
max_positive_polarity
avg_negative_polarity
min_negative_polarity
max_negative_polarity


shares
title_subjectivity
title_sentiment_polarity
abs_title_subjectivity
abs_title_sentiment_polarity

Appendix B: R-CODE

```
dataset_21<-read.csv("/Users/thanosalexandris/Desktop/R BUSINESS ANALYTICS/FINAL
PROJECT/datasets/alldata_onlinenews_21.csv",header=TRUE,dec=",",sep=";")

str(dataset_21)
names(dataset_21)
View(dataset_21)

#removing non-useful columns
dataset_21 = subset(dataset_21, select = -c(X,url,timedelta,is_weekend))

# null/na/inf
dataset_21[dataset_21==""]<-NA

for (i in 1:ncol(dataset_21))
{
  NAS21 <- is.na(dataset_21[,i])
  INF21 <- is.infinite(dataset_21[,i])
}
any(INF21)
any(NAS21)

# Factors
dataset_21$weekday_is_monday<-factor(dataset_21$weekday_is_monday)
dataset_21$weekday_is_tuesday<-factor(dataset_21$weekday_is_tuesday)
dataset_21$weekday_is_wednesday<-factor(dataset_21$weekday_is_wednesday)
dataset_21$weekday_is_thursday<-factor(dataset_21$weekday_is_thursday)
dataset_21$weekday_is_friday<-factor(dataset_21$weekday_is_friday)
dataset_21$weekday_is_saturday<-factor(dataset_21$weekday_is_saturday)
dataset_21$weekday_is_sunday<-factor(dataset_21$weekday_is_sunday)

dataset_21$data_channel_is_bus<-factor(dataset_21$data_channel_is_bus)
dataset_21$data_channel_is_entertainment<-factor(dataset_21$data_channel_is_entertainment)
dataset_21$data_channel_is_lifestyle<-factor(dataset_21$data_channel_is_lifestyle)
dataset_21$data_channel_is_socmed<-factor(dataset_21$data_channel_is_socmed)
dataset_21$data_channel_is_tech<-factor(dataset_21$data_channel_is_tech)
dataset_21$data_channel_is_world<-factor(dataset_21$data_channel_is_world)
```

```
#Numeric variables
for (i in 1:(ncol(dataset_21)))
{if(class(dataset_21[,i])=='integer'){
  dataset_21[,i]<-as.numeric(dataset_21[,i])
}
}
#  numerics dataframe
library(psych)
index <- sapply(dataset_21, class) =='numeric'
numeric21 <- dataset_21[,index]
n <- nrow(numeric21)

str(numeric21)

# factors dataframe
factors21 <- dataset_21[,!index]

# Numerics visualization
par(mfrow=c(2,5))
for (i in 1:10)
{
  h1 <- hist(numeric21[,i], main=names(numeric21)[i], col='pink')
}
par(mfrow=c(2,5))
for (i in 11:20)
{
  h2 <- hist(numeric21[,i], main=names(numeric21)[i], col='pink')
}
par(mfrow=c(2,5))
for (i in 21:30)
{
  h3 <- hist(numeric21[,i], main=names(numeric21)[i], col='pink')
}
par(mfrow=c(2,5))
for (i in 31:40)
{
  h4 <- hist(numeric21[,i], main=names(numeric21)[i], col='pink')
```

```
}
par(mfrow=c(2,5))
for (i in 41:45)
{
  h5 <- hist(numeric21[,i], main=names(numeric21)[i], col='pink')
}

# factors visualization
barplot(sapply(factors21[,c(1:6)],table)/n, names.arg = c("Lifestyle", "Entertainment", "Bus", "Socmed",
"Tech", "World"), horiz=T, las=1, col=2:3, ylim=c(0,11), cex.names=0.8)
legend('topleft', fil=2:3, legend=c('No','Yes'), ncol=2,cex=0.5)
barplot(sapply(factors21[,c(7:13)],table)/n, names.arg = c("Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday", "Sunday"), horiz=T, las=1, col=2:3, ylim=c(0,11), cex.names=0.8)
legend('topleft', fil=2:3, legend=c('No','Yes'), ncol=2, cex=0.5)

numeric21$shares<- log(numeric21$shares)
dataset_21$shares <- log(dataset_21$shares)

# Pairs of shares and other numerical variables
pairs(numeric21 [,c(45,1,2,3,4,5)])
pairs(numeric21 [,c(45,6,7,8,9,10)])
pairs(numeric21 [,c(45,11,12,13,14,15,16,17)])
pairs(numeric21 [,c(45,18,19,20,21,22,23)])
pairs(numeric21 [,c(45,24,25,26,27,28)])
pairs(numeric21 [,c(45,29,30,31,32,33,34)])
pairs(numeric21 [,c(45,35,36,37,38,39,40)])
pairs(numeric21 [,c(45,41,42,43,44)])

require(corrplot)
corrplot(cor(numeric21 [,c(45,1,2,3,4,5)]),method='number',type = "upper", number.cex = .5, tl.offset = 0.37,
tl.cex=0.37)
corrplot(cor(numeric21 [,c(45,6,7,8,9,10)]),method='number',type = "upper", number.cex = .5, tl.offset = 0.37,
tl.cex=0.37)
corrplot(cor(numeric21 [,c(45,11,12,13,14,15,16,17)]),method='number',type = "upper", number.cex = .5,
tl.offset = 0.55, tl.cex=0.55)
corrplot(cor(numeric21 [,c(45,18,19,20,21,22,23)]),method='number',type = "upper", number.cex = .5, tl.offset
= 0.4, tl.cex=0.4)
```

```
corrplot(cor(numeric21 [,c(45,24,25,26,27,28)]),method='number',type = "upper", number.cex = .5, tl.offset = 0.6, tl.cex=0.6)
corrplot(cor(numeric21 [,c(45,29,30,31,32,33,34)]),method='number',type = "upper", number.cex = .5, tl.offset = 0.4, tl.cex=0.4)
corrplot(cor(numeric21 [,c(45,35,36,37,38,39,40)]),method='number',type = "upper", number.cex = .5, tl.offset = 0.4, tl.cex=0.4)
corrplot(cor(numeric21 [,c(45,41,42,43,44)]),method='number',type = "upper", number.cex = .5, tl.offset = 0.5, tl.cex=0.5)

# Shares on each factor variable
par(mfrow=c(2,3))
for(j in 1:6){
  boxplot(dataset_21[,58]~factors21[,j], xlab=names(factors21)[j], ylab='Shares',cex.lab=1.5)
  abline(lm(dataset_21[,1]~factors21[,j]),col=2)
}
par(mfrow=c(2,4))
for(j in 7:13){
  boxplot(dataset_21[,58]~factors21[,j], xlab=names(factors21)[j], ylab='Shares',cex.lab=1.5)
  abline(lm(dataset_21[,58]~factors21[,j]),col=2)
}


###selecting attributes########
library(corrplot)
correlations <- cor(numeric21, use="pairwise.complete.obs")
cor_sorted <- as.matrix(sort(correlations[,"shares"], decreasing = TRUE))
#select high correlations
high_correlations <- names(which(apply(cor_sorted, 1, function(x) (abs(x) < -0.04 || abs(x)> 0.04))))
correlations2 <- correlations[high_correlations, high_correlations]
corrplot(correlations2,method='number',type = "upper", number.cex = .28, tl.offset = 0.41, tl.cex=0.41)
high_correlations
#variable shares has stronger corellations with the variables included in matrix high_correlations

require(glmnet)
library(glmnet)
fullmodel21 <- lm(shares
~kw_avg_avg+num_hrefs+num_imgs+LDA_03+self_reference_max_shares+self_reference_avg_sharess+kw_
min_avg+
```

```
self_reference_min_shares+global_sentiment_polarity+kw_max_avg+kw_avg_max+global_subjectivity+title_sentiment_polarity+num_keywords+num_videos+

n_tokens_content+abs_title_sentiment_polarity+title_subjectivity+num_self_hrefs+global_rate_positive_words+LDA_00+max_positive_polarity+

min_positive_polarity+n_unique_tokens+n_non_stop_unique_tokens+rate_negative_words+LDA_02+

weekday_is_monday+weekday_is_tuesday+weekday_is_wednesday+weekday_is_thursday+weekday_is_friday+weekday_is_saturday+

weekday_is_sunday+data_channel_is_bus+data_channel_is_entertainment+data_channel_is_lifestyle+data_channel_is_socmed+
        data_channel_is_tech+data_channel_is_world, data = dataset_21)

summary(fullmodel21)

X_21 <- model.matrix(fullmodel21)[,-1]
lasso <- glmnet(X_21, numeric21$shares, standardize = TRUE,alpha = 1)
library(plotmo)
plot_glmnet(lasso,label=15)
lasso2 <- cv.glmnet(X_21,numeric21$shares, alpha = 1)
plot(lasso2)
#1se0.06905 min#0.0051
lassoModel1 <- coef(lasso2, s = lasso2$lambda.1se)
lassoModel


#Return coefficients with values
rownames(coef(lasso2, s = 'lambda.1se'))[coef(lasso2, s = 'lambda.1se')[,1]!= 0]

#Stepwise with coefficients of 'lambda.1se'

#AIC step
new_model <- lm(shares
~kw_avg_avg+num_hrefs+num_imgs+self_reference_max_shares+self_reference_avg_sharess+
```

```
LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = dataset_21)
summary(new_model)

AIC<-step(new_model, direction='both')
summary(AIC)

anova(AIC)

after_AIC_model <- lm(shares ~kw_avg_avg+num_hrefs+num_imgs+self_reference_avg_sharess+

LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = dataset_21)
summary(after_AIC_model)

after_AIC_model_no_intercept<-lm(shares ~kw_avg_avg+num_hrefs+num_imgs+self_reference_avg_sharess+

LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d-1, data = dataset_21)
summary(after_AIC_model_no_intercept)

true.r2 <- 1-sum(after_AIC_model_no_intercept$residuals^2)/((n-1)*var(dataset_21$shares))

#VIF
install.packages('psych')
require(car)
vif(after_AIC_model)#all below 10



#Normality of the residuals

plot(after_AIC_model, which=2)
res <- after_AIC_model$residuals

library(nortest)
lillie.test(res)
shapiro.test(res)
```

```
#constant variance
Stud.residuals<-rstudent(after_AIC_model)
yhat <- fitted(after_AIC_model)
par(mfrow=c(1,2))
plot(yhat, Stud.residuals)
abline(h=c(-2,2), col=2, lty=2)
plot(yhat, Stud.residuals^2)
abline(h=4, col=2, lty=2)

library(car)
ncvTest(after_AIC_model)
yhat.quantiles<-cut(yhat, breaks=quantile(yhat, probs=seq(0,1,0.25)), dig.lab=6)
table(yhat.quantiles)
leveneTest(rstudent(after_AIC_model)~yhat.quantiles)

#non-linearity#
library(car)

par(mfrow=c(1,1))
residualPlot(after_AIC_model, type='rstudent')
residualPlots(after_AIC_model, plot=F, type = "rstudent")

#Independence of errors
library(car)
durbinWatsonTest(after_AIC_model)

plot(rstudent(after_AIC_model), type='l')

##Use polynomials##
model2<-lm(shares
~kw_avg_avg+I(kw_avg_avg^2)+I(kw_avg_avg^3)+num_hrefs+num_imgs+self_reference_avg_sharess+

LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = dataset_21)

residualPlots(model2,plot=F)
summary(model2)
```

```
#Normality of the residuals
plot(model2, which=2)
res2 <- model2$residuals

library(nortest)
lillie.test(res2)
shapiro.test(res2)

#constant variance
Stud.residuals2<-rstudent(model2)
yhat2 <- fitted(model2)
par(mfrow=c(1,2))
plot(yhat2, Stud.residuals2)
abline(h=c(-2,2), col=2, lty=2)
plot(yhat2, Stud.residuals2^2)
abline(h=4, col=2, lty=2)

library(car)
ncvTest(model2)
yhat.quantiles2<-cut(yhat2, breaks=quantile(yhat2, probs=seq(0,1,0.25)), dig.lab=6)
table(yhat.quantiles2)
leveneTest(rstudent(model2)~yhat.quantiles2)

#non-linearity#
library(car)

par(mfrow=c(1,1))
residualPlot(model2, type='rstudent')
residualPlots(model2, plot=F, type = "rstudent")

#Independence of errors


library(car)
durbinWatsonTest(model2)

plot(rstudent(model2), type='l')
```

```
#### Prediction - test dataset ########################################

test_dataset<-read.csv("/Users/thanosalexandris/Desktop/R BUSINESS ANALYTICS/FINAL
PROJECT/datasets/OnlineNewsPopularity_test.csv",header=TRUE,dec=",",sep=";")
View(test_dataset)

test_dataset = subset(test_dataset, select = -c(X,url,timedelta,is_weekend))

test_dataset$weekday_is_monday<-factor(test_dataset$weekday_is_monday)
test_dataset$weekday_is_tuesday<-factor(test_dataset$weekday_is_tuesday)
test_dataset$weekday_is_wednesday<-factor(test_dataset$weekday_is_wednesday)
test_dataset$weekday_is_thursday<-factor(test_dataset$weekday_is_thursday)
test_dataset$weekday_is_friday<-factor(test_dataset$weekday_is_friday)
test_dataset$weekday_is_saturday<-factor(test_dataset$weekday_is_saturday)
test_dataset$weekday_is_sunday<-factor(test_dataset$weekday_is_sunday)

test_dataset$data_channel_is_bus<-factor(test_dataset$data_channel_is_bus)
test_dataset$data_channel_is_entertainment<-factor(test_dataset$data_channel_is_entertainment)
test_dataset$data_channel_is_lifestyle<-factor(test_dataset$data_channel_is_lifestyle)
test_dataset$data_channel_is_socmed<-factor(test_dataset$data_channel_is_socmed)
test_dataset$data_channel_is_tech<-factor(test_dataset$data_channel_is_tech)
test_dataset$data_channel_is_world<-factor(test_dataset$data_channel_is_world)



test_dataset$shares <- log(test_dataset$shares)


#1- - fold cross validation  Choose min RMSE
#First Model
install.packages('caret')
library(caret)
set.seed(1)
train_control_CV_10 <- trainControl(method = "CV", number = 10)
model_cv_10_1 <- train(shares ~kw_avg_avg+num_hrefs+num_imgs+self_reference_avg_sharess+
```

```
LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = test_dataset,
            trControl=train_control_CV_10,
            method="lm")
model_cv_10_1

# Leave one out cross validation
train_control_LOOCV <- trainControl(method = "LOOCV")
model_LOOCV_1<- train(shares ~kw_avg_avg+num_hrefs+num_imgs+self_reference_avg_sharess+

LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = test_dataset,
                trControl=train_control_LOOCV, method="lm")
model_LOOCV_1

#1- - fold cross validation  Choose min RMSE
#Second model
model_cv_10_2 <- train(shares
~kw_avg_avg+I(kw_avg_avg^2)+I(kw_avg_avg^3)+num_hrefs+num_imgs+self_reference_avg_sharess+

LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = test_dataset,
                trControl=train_control_CV_10,
                method="lm")
model_cv_10_2

# Leave one out cross validation
model_LOOCV_2<- train(shares
~kw_avg_avg+I(kw_avg_avg^2)+I(kw_avg_avg^3)+num_hrefs+num_imgs+self_reference_avg_sharess+

LDA_02+weekday_is_saturday+weekday_is_sunday+data_channel_is_entertainment+data_channel_is_socme
d, data = test_dataset,
                            trControl=train_control_LOOCV, method="lm")
model_LOOCV_2

results<- rbind(model_cv_10_1$results[1:4],model_LOOCV_1$results,
                model_cv_10_2$results[1:4],model_LOOCV_2$results)
```

```
rownames(results)<- c("model_cv_10_1","model_LOOCV_1","model_cv_10_2","model_LOOCV_2")
```