# DATA MINING REPORT
# (CRISP-DM APPROACH)

Predictive Analysis on Online News Popularity

Report By

Pedapudi Venkata Sai Vijay Kumar (A0178190Y)

Maddi Kamal Manisha (A0178259M)

Monisha Prasad (A0178265U)

Venkateswara Venkata Krishnan (A0178343Y)

## TABLE OF CONTENTS

# 1. INTRODUCTION

## 1.1. Domain

Mashable.com is a leading independent source for news, information and resources for the connected generation which casts interesting articles and news across diversified topics such as entertainment, tech, business etc. The number of online news portal available today makes it a challenging task for the news portal to grab a formidable position in the market and to get regular visitors for their web page. The more viral a news post becomes, the more revenue it brings through advertisements and marketing methods like Google AdSense, Bing ads, Yahoo Gemini etc.

## 1.2. Definition of task

This project aims to predict the reach/popularity of news article in the social networking platform for an online news/blog post based on the attributes about the news article. The prediction results will be highly influencing in the business by helping the news portal to design the news post to reach maximum number of audience.

## 1.3. Structure of data set

The dataset contains a set of features related to blogs posted by online news website Mashable over a span of 2 years. The data set contains over 35000+ records and 61 attributes. The attribute set includes

a. Basic information such as URL of the post, number of tokens in title, content, average length of the words used etc.
b. Detailed information about the content such as number of images, videos, referencing links with the same web page and other web pages, number of keywords etc.
c. News information such as the category (business, lifestyle, entertainment, tech, social media, world) and the time in which the blog was posted (day of the week)
d. Analysis information such as LDA, subjectivity, positivity, negativity, polarity and best, worst and average keyword

# 2. APPROACH

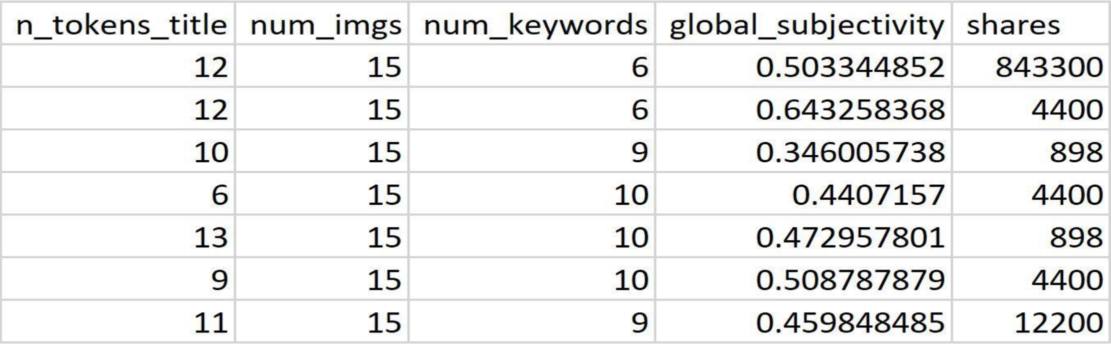
To solve the business objective, we followed the traditional “CRISP-DM” methodology which involves the following tasks.

## 2.1. Business Understanding/Problem Understanding

The news article published online is considered as a valuable resource for the journalists, advertisers, content providers etc. Since the rise of smartphones and internet, online news portal became the primary source of news. To reach a massive crowd of people, the key factor is not just the number of people reading the article, but the number of people sharing the article through social networking platforms like Facebook, LinkedIn, twitter etc. As a well-known parameter, the number of shares determines the popularity of the news which in turn results in the revenue through advertisements. The key challenge in predicting the popularity of a blog post is it vastly depends on human behaviour as mentioned in the research “A survey on predicting the popularity of web content” [2].

### 2.2. Data Understanding

The data source is from UCI Machine learning library [1]. The well-known attributes such as the number of keywords, the category of the blog post was easily understandable. The major hindrance was on the initial understanding of the analytics information provided in the data set such as LDA, polarity, subjectivity which demanded more knowledge on the text mining techniques which was briefly inferred from “Large scale sentiment analysis for news and blogs” [8]. On looking deep into the records, it was evident that the news can be an irrelevant news, which is shared by at most one person or it can be a viral news which is shared by more than half a million people. So, the variety in the data was also a critical factor in deciding the next steps. A sample data explaining the variance is mentioned below. The sample of data below, filtered for the number of images as 15 in a blog post, and the variance can be clearly seen in the number of shares column.



| n_tokens_title | num_imgs | num_keywords | global_subjectivity | shares |
|---:|---:|---:|---:|---:|
| 12 | 15 | 6 | 0.503344852 | 843300 |
| 12 | 15 | 6 | 0.643258368 | 4400 |
| 10 | 15 | 9 | 0.346005738 | 898 |
| 6 | 15 | 10 | 0.4407157 | 4400 |
| 13 | 15 | 10 | 0.472957801 | 898 |
| 9 | 15 | 10 | 0.508787879 | 4400 |
| 11 | 15 | 9 | 0.459848485 | 12200 |

Table 2.2: Sample Data

Refer to appendix A for the complete set of attributes description.

### 2.3. Data Preparation

Discussed in Section 3

### 2.4. Data Modeling & Model Evaluation

Discussed in Section 4

## 3. DATA PREPARATION

### 3.1. Data Cleaning

Dataset has been thoroughly checked using various summarizing techniques for identifying

1. Inconsistent data
2. Inappropriate data
3. Missing data

It’s observed that all the data types i.e. the variable types are consistent. Numeric fields are filled with only numeric values and there are no irrelevant negative values.

While exploring the **data summary** of the input variables few irrelevant values in n_tokens_content attribute is found. The value for n_tokens_content is 0 for few records. With the business understanding the number of words in the news content can never be 0. Upon checking the news post in the website with the URL provided in the dataset, its observed that the rows with value 0 for n_tokens_content are misleading and irrelevant data. So, it is **excluded** and considered as missing data.

Along with data summary, **Boxplot** for various attributes is plotted to detect outliers. Refer to appendix B for box plot of all the variables.

Following are the some of the observations from Box plot analysis.

1. All the records in n_non_stop_words lie in the range from 0 to 1, except 1 record which is 1041 and it has been removed as an outlier. The outlier is detected through the data summary of the n_non_stop_words field, where the mean lies at 0.99 but the maximum value was 1041.
2. One news post has gone **viral** and the number of shares is more than **8,00,000** contrastingly for another post where the number of share is only 1. The average number of shares is 3355 and the median lies at 1400. Understandable, since it is data about an online news portal, the data is highly skewed which led to skewness analysis and data transformation which is discussed in the next section. Before removing the outliers, the data is analyzed furthermore for skewness and transformed. Upon transformation, the outliers are detected on the transformed data and handled accordingly. More on handling the outliers is discussed in the coming sections.

## 3.2 Skewness Analysis

### 3.2.1 Skewness measure on target variable

The target variable is highly skewed to 34.95. On applying log transformation on the target variable, the skewness is reduced to 1.02. Even after transformation for the target variable, there are more than 1430 unique variables which will be hard for a linear algorithm to predict the exact number of shares. Further handling of the target variable for model building is discussed in the modeling section

### 3.2.2. Skewness measure on other variables

Skewness and the kurtosis measure is applied on the other input variables excluding the identifiers and non-predictive variables. Among multiple dependent variables, many variables like n_tokens_content, num_hrefs, min_positive_polarity are right skewed. Variables like avg_token_length, n_non_stop_words are left skewed. The target variable “shares” is also highly skewed on the right side.

Since it is a highly variant data, variables with skewness below -2 and above 2 are considered to be highly skewed. Skewness removal/transformation will be done on the input variables after the transformation, outlier removal and category generation of target variable.

To remove the skewness, the highly skewed columns are normalized using log transformation and square root transformation. Variables which have 0 as a value cannot be transformed using log transformation, so square root transformation is applied on that.

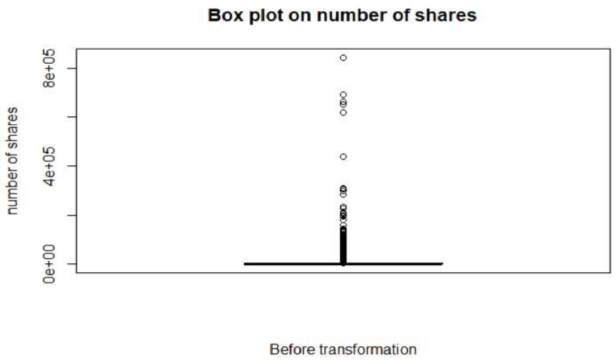


Fig 3.2.1: Boxplot on shares before transformation

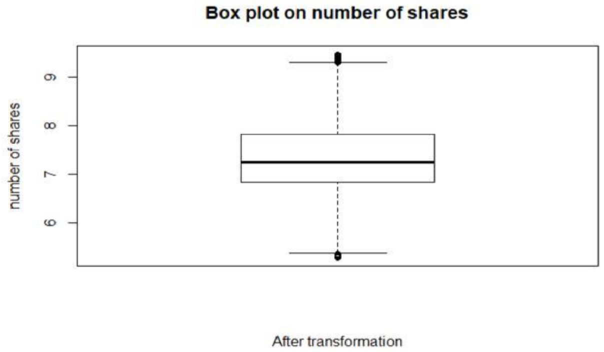


Fig 3.2.2: Boxplot on shares after transformation

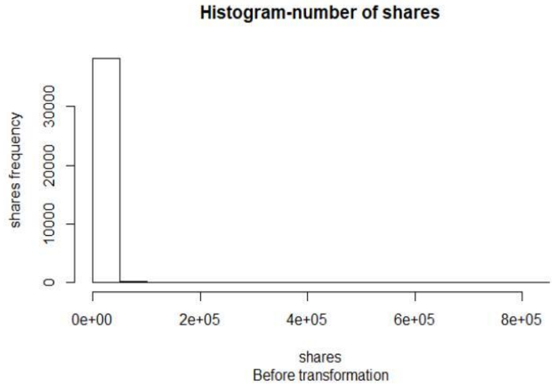


Fig 3.2.3: Histogram on shares before transformation

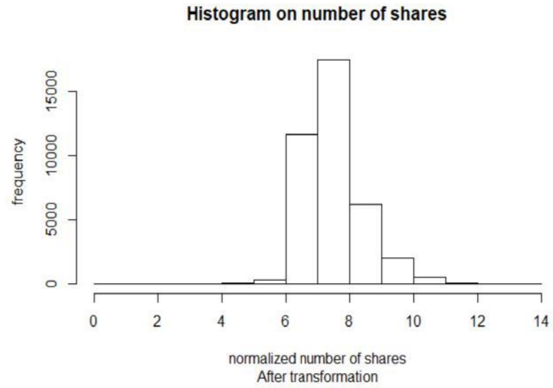


Fig 3.2.4: Histogram on shares after transformation

## 3.3. Handling Outliers in Target variable

To have an accurate model, the outlier in the target variable (shares) is ignored. But considering the fact that, in social media the news post can go viral suddenly, we raised the outlier fence to include the maximum number of valid data to train our model.

a. From the box plot statistics, the lower outer fence and the upper outer fence is constructed using the fence measure to consider the mild outliers and exclude the extreme outliers

b. The primary reason of not excluding the entire outliers is the domain itself where the number of shares isn't under the unanimous range. The below formula is used to expand the fence.

```
hspread<-upperhinge-lowerhinge
lowerouterfence<-lowerhinge-3*hspread
upperouterfence<-upperhinge+3*hspread
```

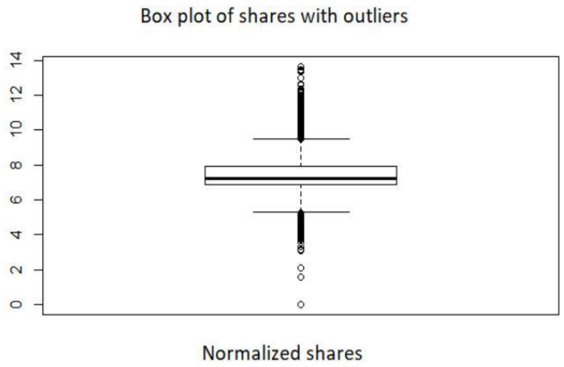


Fig 3.3.1: boxplot on normalized shares with outliers

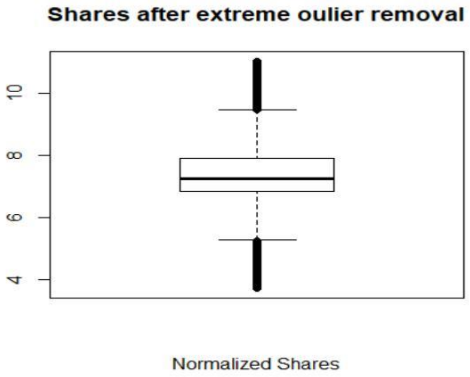


Fig 3.3.2: boxplot on normalized shares without outliers

### 3.4 Outlier and Skewness considerations

Upon removing the outlier and skewness in most of the variables, still skewness was observed in few variables. Looking detailed into that, it was clearly going hand in hand with the business. For instance, the attribute num_imgs which represents the number of images present in the news post is skewed moderately even after transformation. The reason being, there might be few news where there is extensive use of images and there will be few news with no use of images. Same consideration applies to number of videos, and few other variables like self-referencing min shares, self-referencing avg shares, etc.

### 3.5 Feature Selection

The preliminary level of selection of attributes are done with the knowledge gained from the business understanding. The initial number of attributes is 61, excluding the identifier attribute “URL”, the number of numeric input variables is 60.

To reduce the complexity in model building and to have a better trade-off between performance and the execution time of the model, the number of attributes is gracefully reduced by careful observation and understanding of the domain

The clear attributes like **n_token_content** i.e. the number of tokens in the content, **num_keywords** i.e. the number of keywords used, **timedelta** i.e. the time between the day when the share was posted and the day when the dataset is acquired are taken into consideration.

Box plot (refer image below) for the day published and the number of shares is plotted, which showed news posted on **weekend** has slightly more shares compared to other news posted on weekdays.

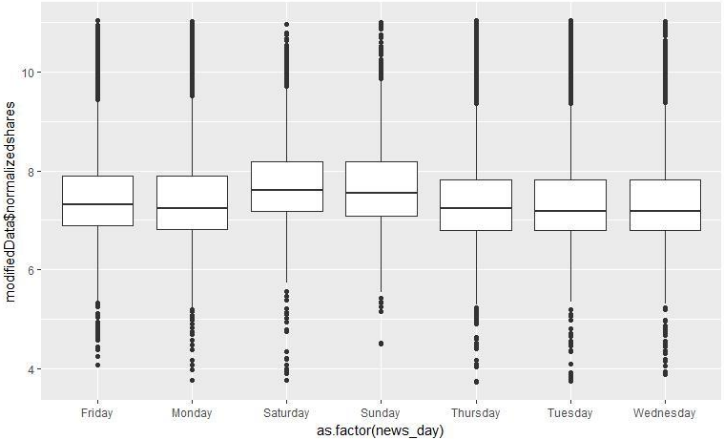


Fig 3.5.1: Attribute Importance

There are few more variables which explained about the subjectivity, polarity, maximum shares of the referenced websites, LDA analysis, the channel of the news post i.e. the category like social media, tech, the day of week published etc. To understand the highly important variables under each of these sections it will be highly time consuming to manually understand the data. So, it is highly appropriate to use the machine computing power to decide the importance based on data and carry on further based on business knowledge.

For deriving the variable importance, a simple random forest model (number of trees - 500, number of splits - 7) is applied on the entire data. On observing the Gini importance and the mean decrease in accuracy, attributes which have more than 20% impact in the trees are considered for the final model building after multiple iteration of selection done with the variables with impact more that 20%.

All the variables which were assumed initially to be a part of the model building had enough importance in the random forest model also. Upon understanding each variable's importance through the plain random forest model, the following attributes are considered for modeling. Refer to appendix A for detailed explanation on each attribute.

| | |
|---|---|
| content and referenced content related attributes | n_tokens_content, timedelta, n_unique_tokens, n_non_stop_unique_tokens, num_imgs, num_hrefs, n_non_stop_words, average_token_length, self_reference_avg_sharess, is_weekend |
| LDA and category related attributes | LDA_00, LDA_01, LDA_02, LDA_03, LDA_04, data_channel_is_entertainment, data_channel_is_tech, data_channel_is_socmed |
| keyword related attributes | kw_avg_avg, kw_max_avg, kw_min_avg |
| NLP attributes (Sentiment analysis and Subjectivity) | global_subjectivity, global_sentiment_polarity, rate_positive_words, rate_negative_words, global_rate_positive_words, global_rate_negative_words |

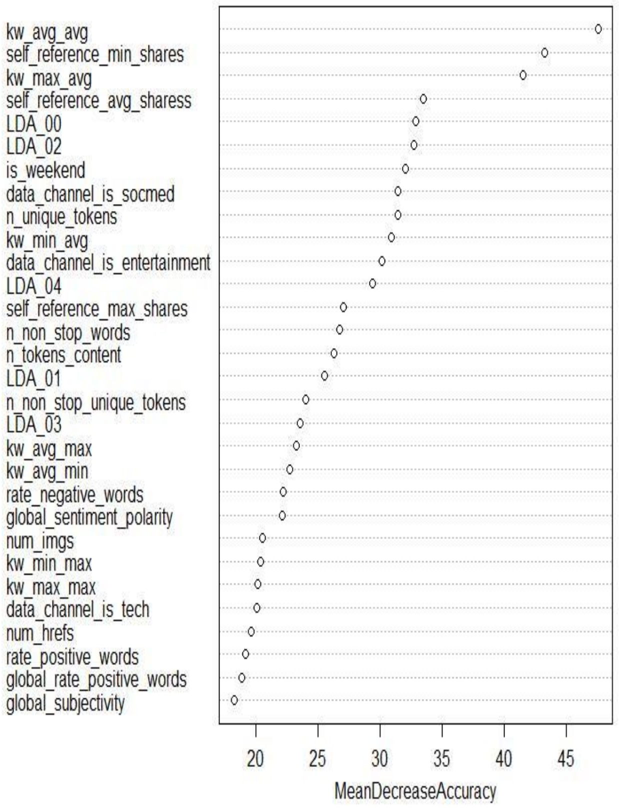


Fig 3.5.2: Attribute selection

# 4. DATA MODELING

After preparing the data by removing all the outliers and irrelevant data completely, the dependent variables were identified in the previous section. To understand the relationship between the dependent variables and the target variable, the correlation matrix is generated. But the highest correlation was at 0.1739 which is totally unelevated. Also, the target variable range is highly varying with more than 1400 different values for prediction. So, the target variable is binned into different categories to predict the right popularity category such as popular, less popular, viral etc. which the new post belongs to. Before binning the target variable into categories, a linear regression model was applied to understand the problem better and to understand the variance in the data.

## 4.1 Regression

The selected attributes, as listed in the previous section are all numeric. So linear regression model is built to predict the exact value of the target variable. Since the correlation was very mere value the linear regression model couldn't fit the target value.

The data is partitioned at 80-20% for training and testing. The linear model produced the mean R-Square value of 0.1235, which is very low, which undeniably explains the variance in the data. Also, the residual quantile plot doesn't have a normal distribution as shown below. So, to make the prediction better, the variables are binned under different number of categories and predicted. More about binning and new attribute generation i.e. the **popularity** attribute is discussed in the next section

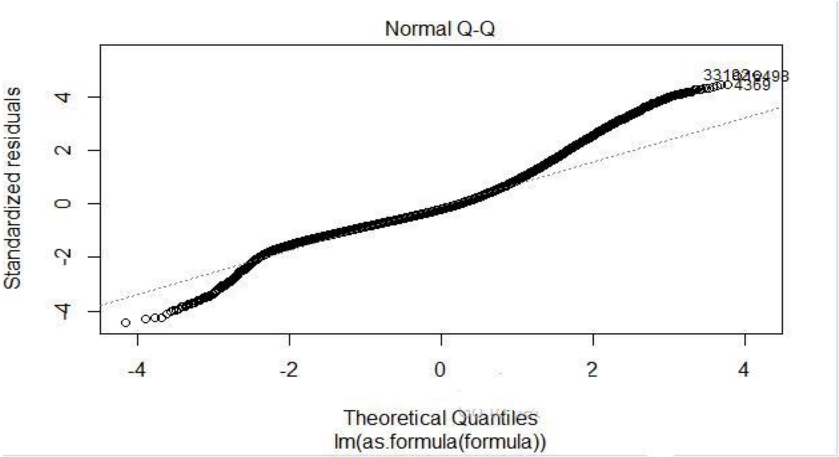


Fig 4.1: Residual Quantile Plot

## 4.2 Attribute Generation

Understanding from the previous section, the linear prediction model will not be able to provide accurate solutions. So, a new categorical attribute called "**popularity**" is created and all the shares (normalized value of shares) are categorized/binned under either 2,3 or 4 categories. To start optimistically the model generation is carried out to predict 2 categories, and later the number of categories is increased based on the accuracy rate of models for each category. The category names for 4,3,2 categories are "Least popular, Average, Popular, Viral", "Least popular, popular, Viral" and "flop, hit" respectively. The box plot statistics are used for binning the normalized share value. Steps for creating categorical variable:

1. Using the box-plot stats the values for Lower Hinge, Upper Hinge, Median, and the outer fences.
2. The extreme outliers are removed, as discussed in section 3.3 using the outer fence values, those boundaries are also considered. So, the binning points are lower outer fence, lower hinge, median, upper hinge, upper outer fence.

Deciding on the model is highly dependent on the variables involved. The output variable in the case of 2 categories, is a binary output. A Logistic regression model will be fast in predicting since all the input variables are numeric. In case of 3 categories the data is biased towards popular category, so a random forest model will be apt since it is robust to biased data. For 4 categories, the data is fairly split across categories, so any classification model can be applied. More on modeling for classification for each type of binning is discussed in the next sections.

### 4.2.1 Data Partitioning

The data is split into training and testing set for model building and evaluation. Initially 80% of the data is considered for training and 20% of the data is partitioned for testing. Before creating the data chunks, the data is shuffled randomly, and the partition is made. Upon deciding the right modeling technique, the model is fine-tuned using K-fold cross validation technique.

| Boundaries/ Binning points (from box-plot stats) | Categories | Number of record under each category |
|---|---|---|
| lower outer fence, Median, upper outer fence | Hit | 18788 |
| | Flop | 19541 |
| lower outer fence, lower hinge, upper hinge, upper outer fence | Least popular | 9624 |
| | Popular | 19232 |
| | Viral | 9473 |
| lower outer fence, lower hinge, median, upper hinge, upper outer fence | Least Popular | 9624 |
| | Average | 9917 |
| | Popular | 9315 |
| | Viral | 9473 |

Table 4.2.1: Binning

## 4.3 Modeling for 2 categories

### 4.3.1 Logistic Regression

As mentioned in the previous section, the training partition is taken to build the model. The logistic model requires binary values 0 & 1 as output. So, the category “Hit” is modified as 1 and “flop” is modified as 0 before building the model. Below are the results of logistic regression

#### 4.3.1.1 Logistic Regression Evaluation

| | | | | | |
|---|---|---|---|---|---|
| Accuracy | 0.6468 | | Confusion Matrix | | |
| Sensitivity | 0.6781 | | Prediction | 0 | 1 |
| Specificity | 0.6151 | | 0 | 2612 | 1468 |
| Balanced Accuracy | 0.6466 | | 1 | 1240 | 2346 |

Table 4.3.1.1.1: Logistic Regression Evaluation

From the above table, it’s evident that, the logistic regression model predicts the two category results with a balanced accuracy of 64%. Accuracy curve for different cut-offs is as below, so to maximize the accuracy the threshold limit is chosen as 0.5. With this threshold Confusion matrix and the ROC curve(area=0.6462) have been studied.

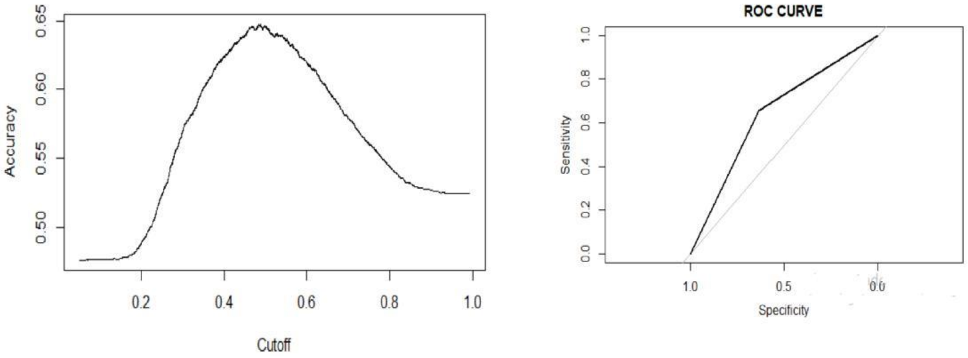


Fig 4.3.1.1.2: ROC

Before proceeding further on logistic regression model, the random forest model is applied, since random forest is apt for classification with ensemble learning.

### 4.3.2 Random Forest

The random forest model is built with the same set of input variables, testing set and training set which was used for logistic regression to understand the accuracy of the models.

The random forest model is initially set with 500 trees and 7 splits. Later, confirming the right model, the number of trees and the number of splits will be tuned to find the optimized results.

| | | | | |
|---|---|---|---|---|
| Accuracy | 0.6645 | Confusion Matrix | | |
| Sensitivity | 0.6752 | Prediction | 0 | 1 |
| Specificity | 0.6536 | 0 | 2601 | 1321 |
| Balanced Accuracy | 0.6644 | 1 | 1251 | 2493 |

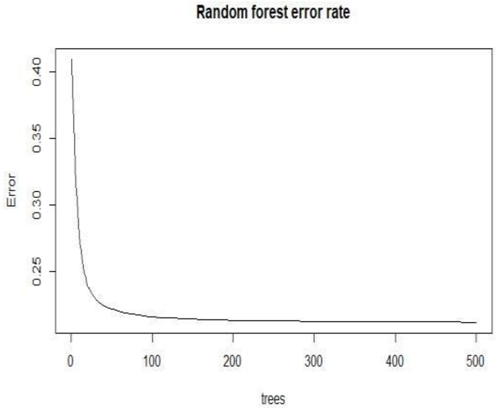


Fig 4.3.2: Random Forest Error Rate

### 4.3.3 Support Vector Machines

The other classification model which is with almost matching accuracy to random forest is support vector machines. The same input formula is used to predict the target variable. Despite SVM having a processing time considerably higher than the linear model and the random forest model, there is not much difference in the accuracy.

| | | | | | |
|---|---|---|---|---|---|
| Accuracy | 0.6551 | | Confusion Matrix | | |
| Sensitivity | 0.6846 | | Prediction | 0 | 1 |
| Specificity | 0.6253 | | 0 | 2637 | 1429 |
| Balanced Accuracy | 0.6550 | | 1 | 1215 | 2385 |

Fig 4.3.3: SVM Results

Before deciding on the right model for 2 category predictions, the category binning is expanded for 3 category and prediction models are built. More details about the 3 category prediction data modeling is dealt in the next section.

## 4.4 Modelling for 3 categories

All the models built for 2 categories binning provided almost similar accuracy around 65% which is a good result considering the data is related to social media. To increase the depth of prediction, 3 category binning is chosen. The issue with 3 category binning is the data is highly biased to the popular category as seen in the table in section 4.2.1. Classification models like SVM are weak with the biased data whereas random forest is robust towards biasing. So random forest model is applied on the training set with 3 category data.

### 4.4.1 Random Forest Model

The random forest model is built by initially setting the number of trees as 500 and the number of split as 7. Below are the results of random forest model built for 3 categories.

| Overall Accuracy | 0.5522 | Least Popular | Popular | Viral |
|---|---|---|---|---|
| Sensitivity | | 0.2971 | 0.8287 | 0.23197 |
| Specificity | | 0.9254 | 0.3133 | 0.92751 |
| Balanced Accuracy | | 0.6112 | 0.5710 | 0.57974 |

| Confusion Matrix | | | |
|---|---|---|---|
| | Least-Popular | Popular | Viral |
| Least-Popular | 565 | 337 | 93 |
| Popular | 1248 | 3237 | 1334 |
| Viral | 89 | 332 | 431 |

Table 4.4.1.1: Random Forest Model for 3 categories

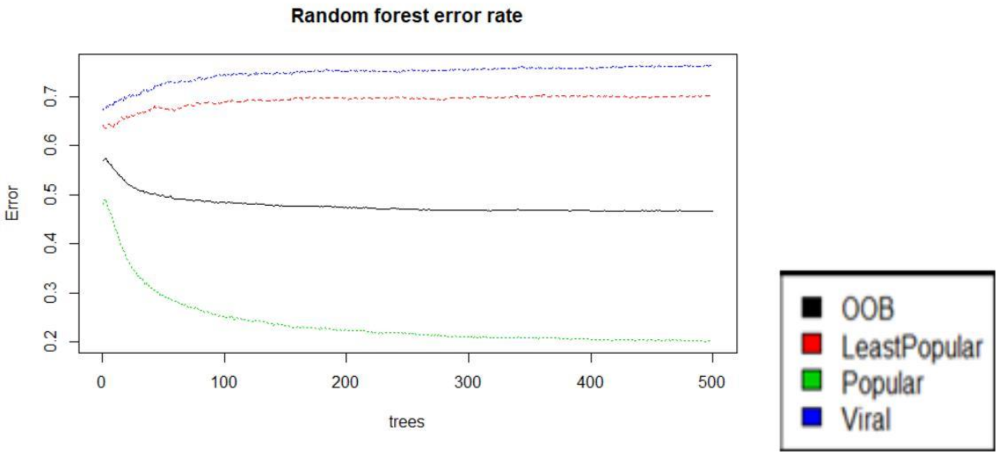


Fig 4.4.1.2: Random Forest Error Rate

From the above table, it is observed that, there is a certain level of decrease in the accuracy from 65% to 53%. Though it is a biased set, the steep decrease in accuracy let to binning them further and predicting with 4 categories to get complete clarity on the model behaviour. More about prediction model with 4 categories is discussed in the next section.

### 4.5 Modeling for 4 categories

As done for the previous two binning, the same input set is used by varying the target variable's category. From the table in section 4.2.1, it is clear that the data is unbiased towards one category. So, both Random forest model and Support vector machines can be applied to test the accuracy of the model.

### 4.5.1 Random Forest Model

For better comparison with the previous category binning, the same number of trees (500) and splits (7) are used for building the prediction model. Below are the results of random forest model built for 4 categories.

| Overall Accuracy | 0.3996 | Least Popular | Average | Popular | Viral |
|---|---|---|---|---|---|
| Sensitivity | | 0.5472 | 0.27076 | 0.26823 | 0.5099 |
| Specificity | | 0.7875 | 0.80653 | 0.83048 | 0.7747 |
| Balanced Accuracy | | 0.6674 | 0.53864 | 0.54935 | 0.6423 |

Table 4.5.1.1: Random Forest Model for 4 categories

| Confusion Matrix | | | | |
|---|---|---|---|---|
| | Least-Popular | Average | Popular | Viral |
| Least-Popular | 1054 | 612 | 350 | 258 |
| Average | 393 | 525 | 408 | 307 |
| Popular | 216 | 388 | 504 | 377 |
| Viral | 263 | 414 | 617 | 980 |

Fig 4.5.1.2: Random Forest Model results

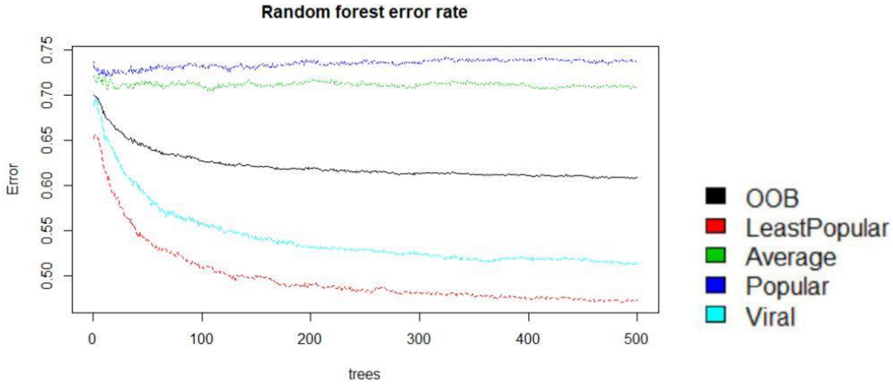


Fig 4.5.1.3: Random Forest Model Error rate

The accuracy faced even drastic decrease, when model is built using 4 categories. This led to a serious research on understanding and applying the predictive model to data in social media which provided interesting information about behaviour of people in social media. More about this is discussed in the section 4.6.

### 4.5.2 Support Vector Machines

As the data for 4 categories is unbiased, and have almost equal binning in all 4 categories, support vector machine model is applied to compare the performance with the random forest model applied earlier. Below are the results of support vector machine model built for 4 categories.

| Overall Accuracy | 0.3844 | Least Popular | Average | Popular | Viral |
|---|---|---|---|---|---|
| Sensitivity | | 0.6262 | 0.18412 | 0.25333 | 0.4724 |
| Specificity | | 0.7068 | 0.85525 | 0.82530 | 0.7918 |
| Balanced Accuracy | | 0.6665 | 0.51968 | 0.53931 | 0.6321 |

Table 4.5.2.1: SVM results for 4 categories

Confusion Matrix

| | Least-Popular | Average | Popular | Viral |
|---|---|---|---|---|
| Least-Popular | 1206 | 816 | 490 | 377 |
| Average | 266 | 357 | 336 | 227 |
| Popular | 210 | 391 | 476 | 410 |
| Viral | 244 | 375 | 577 | 908 |

Table 4.5.2.2: Confusion Matrix - SVM results for 4 categories

## 4.6 Prediction on social Media Data

The right ingredients for understanding the popularity of a web page is still not clearly defined. The research article "A survey on predicting the popularity of web content" by Alexandru Tatar [2] clearly explains the fact that predicting the popularity of an online news is a challenging task as the evolution of content popularity may be described by complex online interactions and information cascades that are difficult to predict.

Research information from "The Coefficient of Determination, r-squared" from Penn-State Eberly college of science [6] states that "Social scientists who are often trying to learn something about the huge variation in human behaviour will tend to find it very hard to get *r*-squared values much above, say 25% or 30%. Engineers, on the other hand, who tend to study more exact systems would likely find an *r*-squared value of just 30% merely unacceptable."

With enough evidences from the above-mentioned research works, the accuracy of all the obtain models have its own significance in the prediction outcome.

Choosing the right model is always a trade-off between multiple factors. More about choosing the right model is discussed in the next section.

## 4.7 Model's Comparison

The complete summary of all the models is listed in the below table.

| Model | number of categories | Accuracy<br>(on a scale of 0 to 1) | Time elapsed for model building (general GPU) | Area under ROC Curve (AUC) |
|---|---|---|---|---|
| Logistic Regression | 2 | 0.6468 | 0.30 secs | 0.6446 |
| Random Forest | 2 | 0.6645 | 6.70 mins | 0.6638 |
| Support Vector Machine | 2 | 0.6551 | 10.69 mins | 0.6517 |
| Random Forest | 3 | 0.5522 | 4.36 mins | |
| Random Forest | 4 | 0.3996 | 3.32 mins | |
| Support Vector Machine | 4 | 0.3844 | 11.54 mins | |

Table 4.7.1: Models Summary for comparison

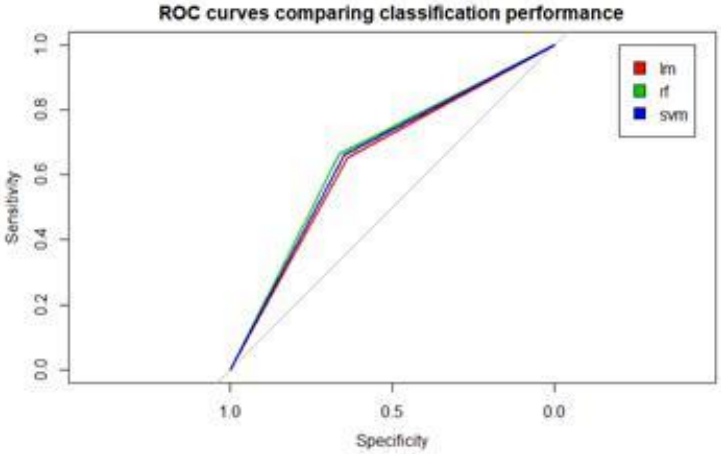


Fig 4.7.2: ROC for Comparison of Models

From the above table, the accuracy decreases with the increase in the number of categories. Even 39% accuracy is acceptable for a social media data as discussed in the previous section. But from the business perspective, the accuracy also plays a key role in prediction of the popular post which would result in more views and more business. So, considering accuracy as a primary factor, the models built with 2 categories are chosen. Under the models for two categories, considering the time elapsed for building the model, logistic regression model would be apt. But when we look at the accuracy the random forest model provides more than 2% of increase in the accuracy with comparatively less time than the SVM model. Also, the random forest model has better ROC than other models. Considering **time elapsed** as the primary factor, **Logistic Regression** will be the suited model, and considering **accuracy** the **random forest** model with 2 categories prediction will provide business-oriented results.

### 4.8 Fine Tuning the Model

### 4.8.1 Fine Tuning the Random Forest Model

The hyper parameters considered for building the random forest model are

1.Data Dimension
2.Number of trees
3.Number of splits at each level

### 4.8.1.1 Data Dimension

Since the attributes are clearly identified and filtered at the data preparation stage itself, the input variables can’t be reduced further. To reduce the time elapsed for building the model, the size of the training set can be varied which will provide better results with less training data. To enable this, K-fold cross validation with 10 folds is performed to determine the accuracy with increase in the size of the training data. Below is the plot explaining the accuracy with the size of training set.

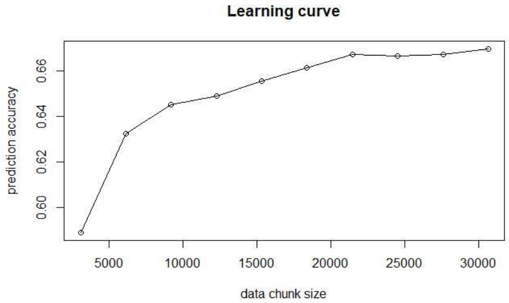


Fig 4.8.1.1: K-fold Cross Validation

### 4.8.1.2 Number of trees

The initial model was built with 500 trees. To reduce the time elapsed, the number of trees is set at various levels without compromising the accuracy. The random forest model was built with 300, 400, 500 and 600 trees with constant number of splits as 7. The 500 trees model gave the highest accuracy compared to other trees, with not much variation in the time elapsed.

### 4.8.1.3 Number of splits

Since the 500 trees model provided best results, the number of trees is kept as 500 and the number of splits is adjusted to determine the best prediction results. Both 6 trees and 8 trees gave a marginally lesser result compared to 7 trees.

| Number of trees | Accuracy | | Number of splits | Accuracy |
|---|---|---|---|---|
| number of splits – 7 | | | number of trees - 500 | |
| 300 | 0.6517 | | 6 | 0.6611 |
| 400 | 0.6659 | | 7 | 0.6686 |
| 500 | 0.6686 | | 8 | 0.6610 |
| 600 | 0.6657 | | | |

Table 4.8.1.3: Prediction Results

## 5.CONCLUSION

The business problem was started with the aim of predicting the reach/popularity of the news article. After multiple levels of cleaning and pre-processing the data is stabilized for model building. Since the linear model couldn’t produce better results because of the variance in the data, various number of bins are used, and classification algorithms are applied. Upon analyzing various models, the suited random forest model is fine tuned. Though 2 categories classification provides better and relevant results, it assumes popularity as a definitive output rather than a ranking methodology. In the next steps, instead of categorizing the news article, a ranking mechanism can be built using the bag of words and other text mining, clustering methodologies [7]. The ranking methodology can be improved over the period using the reinforcement learning by adding the words from the popular articles to the bag of words.

# 6. REFERENCES

[1] “Online News Popularity Data Set”
https://archive.ics.uci.edu/ml/datasets/online+news+popularity
[2] “A survey on predicting the popularity of web content”
https://link.springer.com/article/10.1186/s13174-014-0008-y
[3] “A Survey of Prediction Using Social Media”
https://arxiv.org/ftp/arxiv/papers/1203/1203.1647.pdf
[4] “Predicting and Evaluating the Popularity of Online News” http://cs229.stanford.edu/proj2015/328_report.pdf
[5] “The Pulse of News in Social Media: Forecasting Popularity”
https://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/viewFile/4646/4963
[6] “The Coefficient of Determination, r-squared”
https://onlinecourses.science.psu.edu/stat501/node/255
[7] “Modelling and predicting news popularity”
https://link.springer.com/article/10.1007/s10044-012-0314-6
[8] “large scale sentiment analysis for news and blogs”
http://www.uvm.edu/pdodds/files/papers/others/2007/godbole2007a.pdf

# APPENDIX

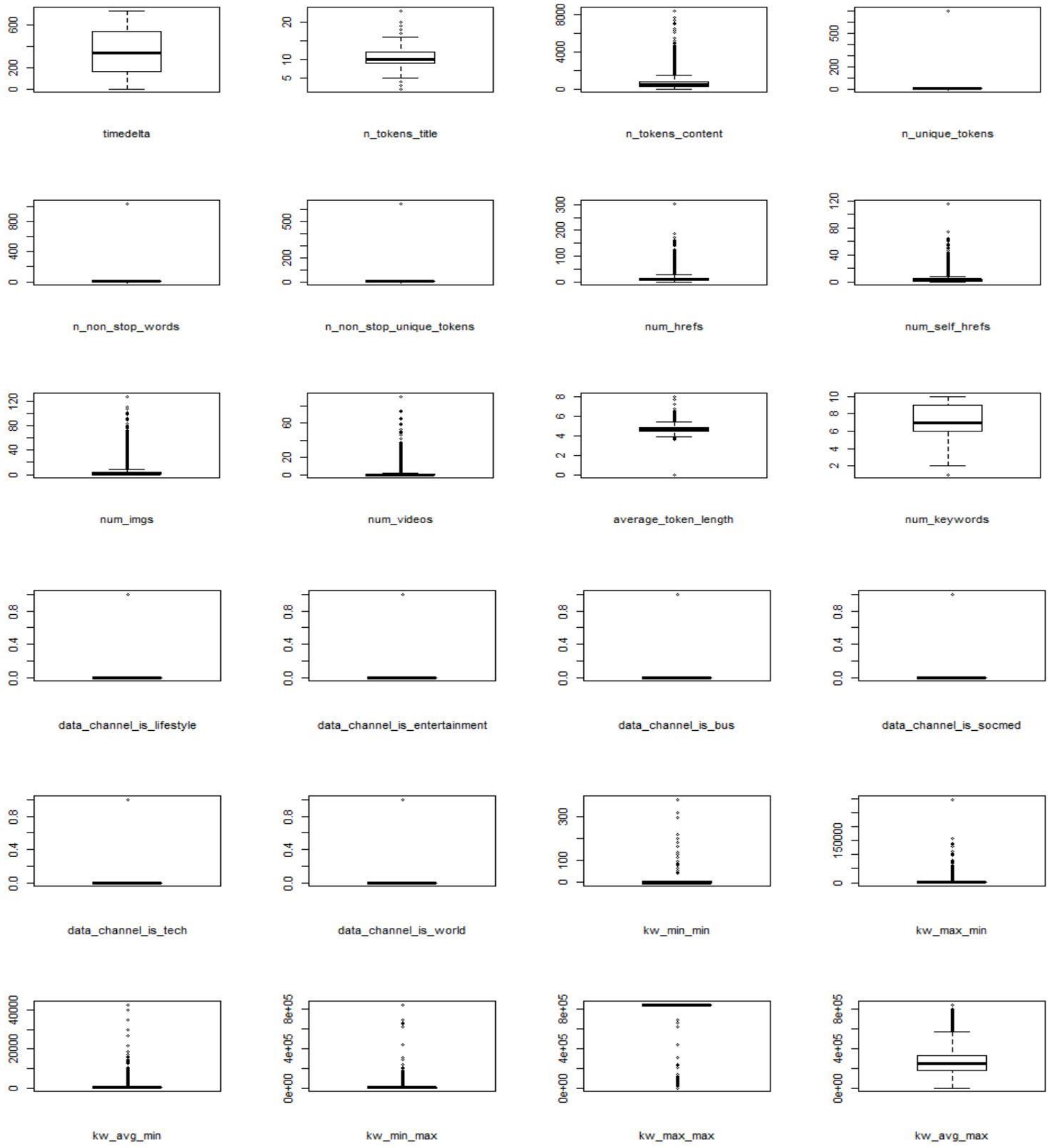
## APPENDIX A - ATTRIBUTES DESCRIPTION

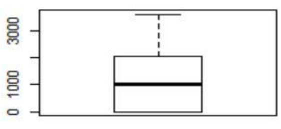
Below are the initial set of attributes and the description from the data source

[1] Identifier - url: URL of the article (non-predictive)

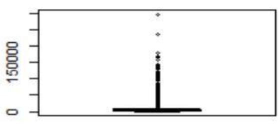
1. timedelta: Days between the article publication and the dataset acquisition (non-predictive)
2. n_tokens_title: Number of words in the title
3. n_tokens_content: Number of words in the content
4. n_unique_tokens: Rate of unique words in the content
5. n_non_stop_words: Rate of non-stop words in the 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)

# APPENDIX B - BOX PLOT FOR EACH ATTRIBUTE

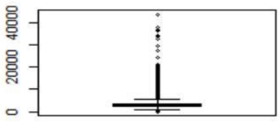




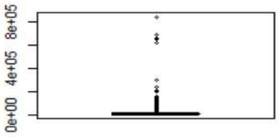
kw_min_avg



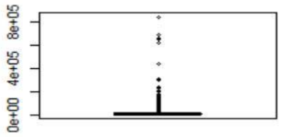
kw_max_avg



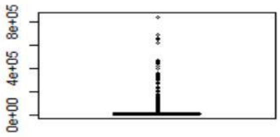
kw_avg_avg



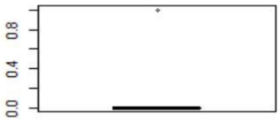
self_reference_min_shares



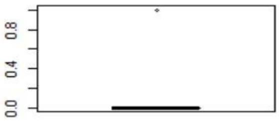
self_reference_max_shares



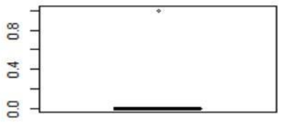
self_reference_avg_sharess



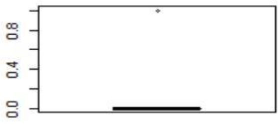
weekday_is_monday



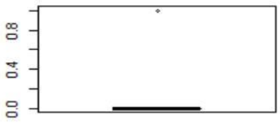
weekday_is_tuesday



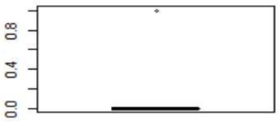
weekday_is_wednesday



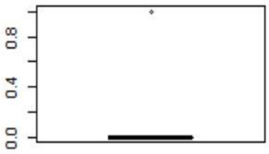
weekday_is_thursday



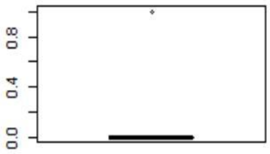
weekday_is_friday



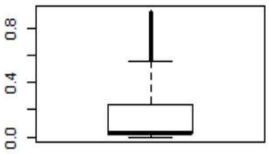
weekday_is_saturday



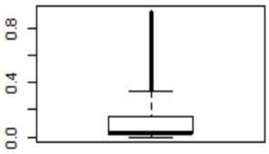
weekday_is_sunday



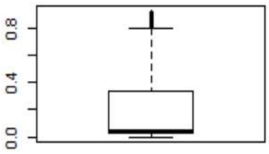
is_weekend



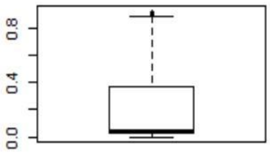
LDA_00



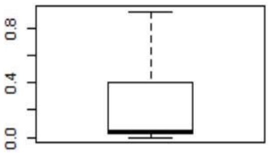
LDA_01



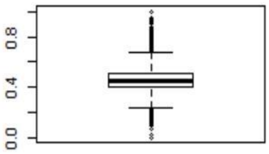
LDA_02



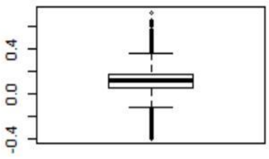
LDA_03



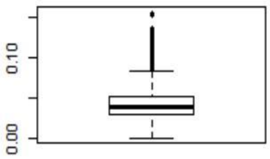
LDA_04



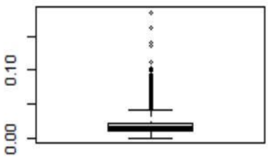
global_subjectivity



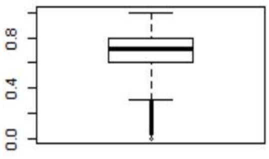
global_sentiment_polarity



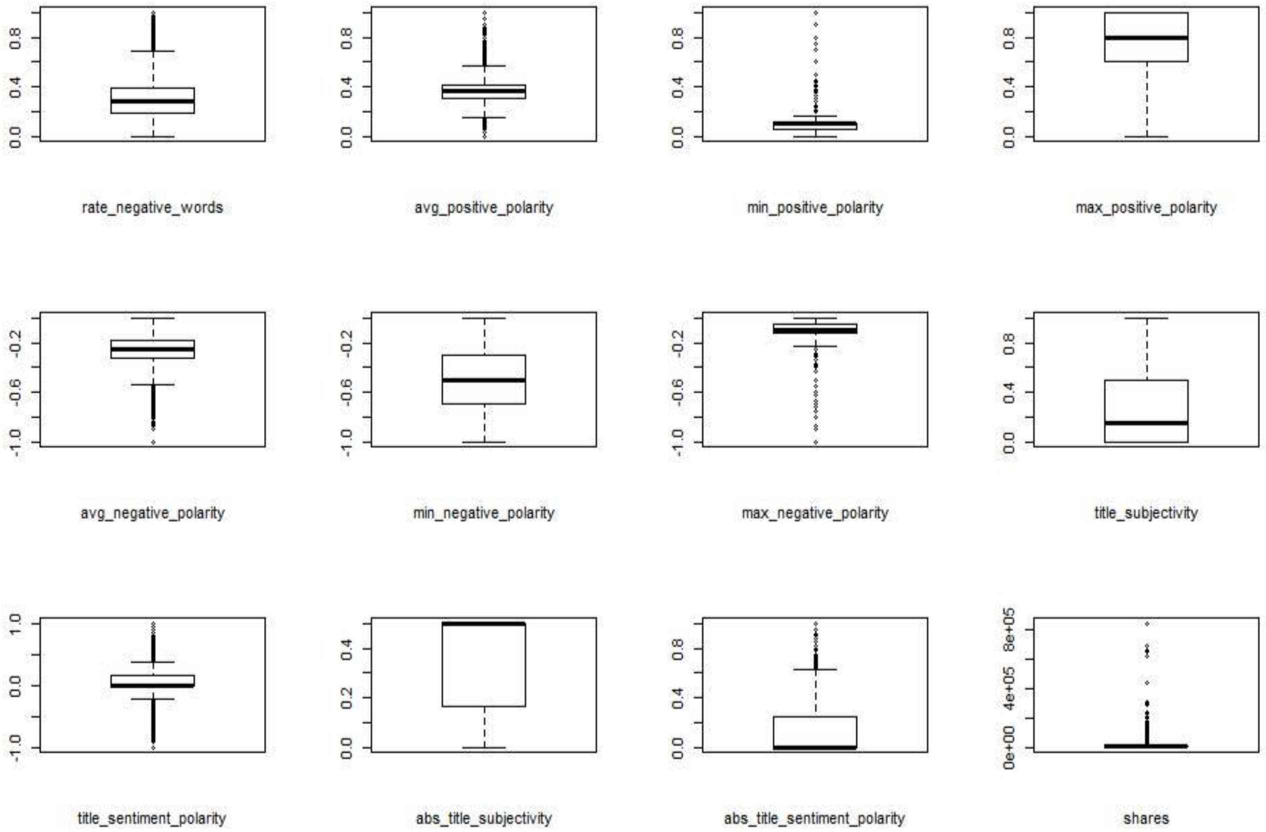
global_rate_positive_words



global_rate_negative_words



rate_positive_words



## APPENDIX C – SKEWNESS BEFORE & AFTER

| Attribute Name | Skewness before | Skewness after |
|---|---|---|
| num_hrefs | 4.056402 | 1.383346 |
| num_self_hrefs | 5.210713 | 0.7197383 |
| num_imgs | 3.967806 | 1.537008 |
| num_videos | 6.943412 | 2.713274 |
| kw_max_min | 36.51198 | 6.077105 |
| kw_min_max | 10.61884 | 3.832506 |
| kw_max_avg | 16.82633 | 5.136376 |
| kw_avg_avg | 6.048811 | 1.418065 |
| self_reference_min_shares | 26.3385 | 5.339222 |
| self_reference_max_shares | 13.77502 | 4.291197 |
| self_reference_avg_sharess | 17.87423 | 4.19713 |
| n_tokens_content | 3.02832 | 0.04682788 |
| min_positive_polarity | 3.211573 | 1.249393 |