Online Nedwd Popularity Prediction and Analysis

Andrew Wild, Christopher Tsai, Isabelle Townley-Smith

I. Introduction

In UCI's Machine Learning Repository, there are hundreds of data sets on a variety of topics, ranging from forest fire areas to poker hands. The data set focused on in this paper describes characteristics of online news articles and how popular they were. Our goal is to use machine learning techniques and algorithms to predict this popularity, which is described by the metric of the number of shares. The two approaches we used were linear regression and decision trees, and we split our data into two parts: 90% going into our training set and 10% going into our test set.

II. Data Set and Analysis

The data set contains 39,644 observations, each of which represents an article that was published on the website Mashable prior to January 8th, 2015 (the listed data acquisition date). There are 61 features in total, two of which are listed as non-predictive (the article url and time delta between publication and data acquisition), which we will thus be excluding from our analysis and prediction attempts. The features cover a wide variety of information about the articles, including word analysis, links, images, videos, time of publication, and some natural language processing features, like word polarity. Given this background, it's important for us to consider that these articles are all from the same website (Mashable) and thus the prediction of shares for news articles in this analysis may not be generalizable to all online news sources.

A five number summary (plus the mean) can be found below in Figure 1 for the quantitative variables as well as the qualitative ones. However, the qualitative variables, such as weekday_is_monday, are stored as dummy variables, meaning that within each column there is a 1 or a 0 for whether that qualitative feature is true for that particular data entry. Because each of these variables only contains a 0 or 1, these summary statistics aren't really meaningful. This is true for the following features: data_channel_is_lifestyle, data_channel_is_entertainment, data_channel_is_bus, data_channel_is_socmed, data_channel_is_tech, data_channel_is_world, weekday_is_monday, weekday_is_tuesday, weekday_is_wednesday, weekday_is_thursday, weekday_is_friday, weekday_is_saturday, weekday_is_sunday, and is_weekend.

```
 n_tokens_title n_tokens_content n_unique_tokens    n_non_stop_words    n_non_stop_unique_tokens   num_hrefs
 Min.   : 2.0   Min.   :   0.0   Min.   :  0.0000   Min.   :   0.0000   Min.   :  0.0000          Min.   :  0.00
 1st Qu.: 9.0   1st Qu.: 246.0   1st Qu.:  0.4709   1st Qu.:   1.0000   1st Qu.:  0.6257          1st Qu.:  4.00
 Median :10.0   Median : 409.0   Median :  0.5392   Median :   1.0000   Median :  0.6905          Median :  8.00
 Mean   :10.4   Mean   : 546.5   Mean   :  0.5482   Mean   :   0.9965   Mean   :  0.6892          Mean   : 10.88
 3rd Qu.:12.0   3rd Qu.: 716.0   3rd Qu.:  0.6087   3rd Qu.:   1.0000   3rd Qu.:  0.7546          3rd Qu.: 14.00
 Max.   :23.0   Max.   :8474.0   Max.   :701.0000   Max.   :1042.0000   Max.   :650.0000          Max.   :304.00
 num_self_hrefs        num_imgs          num_videos     average_token_length  num_keywords     data_channel_is_lifestyle
 Min.   :  0.000   Min.   :  0.000   Min.   : 0.00   Min.   :0.000        Min.   : 1.000   Min.   :0.00000
 1st Qu.:  1.000   1st Qu.:  1.000   1st Qu.: 0.00   1st Qu.:4.478        1st Qu.: 6.000   1st Qu.:0.00000
 Median :  3.000   Median :  1.000   Median : 0.00   Median :4.664        Median : 7.000   Median :0.00000
 Mean   :  3.294   Mean   :  4.544   Mean   : 1.25   Mean   :4.548        Mean   : 7.224   Mean   :0.05295
 3rd Qu.:  4.000   3rd Qu.:  4.000   3rd Qu.: 1.00   3rd Qu.:4.855        3rd Qu.: 9.000   3rd Qu.:0.00000
 Max.   :116.000   Max.   :128.000   Max.   :91.00   Max.   :8.042        Max.   :10.000   Max.   :1.00000
 data_channel_is_entertainment data_channel_is_bus data_channel_is_socmed data_channel_is_tech data_channel_is_world
 Min.   :0.000                 Min.   :0.0000      Min.   :0.0000         Min.   :0.0000       Min.   :0.0000
 1st Qu.:0.000                 1st Qu.:0.0000      1st Qu.:0.0000         1st Qu.:0.0000       1st Qu.:0.0000
 Median :0.000                 Median :0.0000      Median :0.0000         Median :0.0000       Median :0.0000
 Mean   :0.178                 Mean   :0.1579      Mean   :0.0586         Mean   :0.1853       Mean   :0.2126
 3rd Qu.:0.000                 3rd Qu.:0.0000      3rd Qu.:0.0000         3rd Qu.:0.0000       3rd Qu.:0.0000
 Max.   :1.000                 Max.   :1.0000      Max.   :1.0000         Max.   :1.0000       Max.   :1.0000
   kw_min_min        kw_max_min        kw_avg_min         kw_min_max        kw_max_max        kw_avg_max        kw_min_avg
 Min.   : -1.00   Min.   :     0   Min.   :   -1.0   Min.   :     0   Min.   :     0   Min.   :     0   Min.   :  -1
 1st Qu.: -1.00   1st Qu.:   445   1st Qu.:  141.8   1st Qu.:     0   1st Qu.:843300   1st Qu.:172847   1st Qu.:   0
 Median : -1.00   Median :   660   Median :  235.5   Median :  1400   Median :843300   Median :244572   Median :1024
 Mean   : 26.11   Mean   :  1154   Mean   :  312.4   Mean   : 13612   Mean   :752324   Mean   :259282   Mean   :1117
 3rd Qu.:  4.00   3rd Qu.:  1000   3rd Qu.:  357.0   3rd Qu.:  7900   3rd Qu.:843300   3rd Qu.:330980   3rd Qu.:2057
 Max.   :377.00   Max.   :298400   Max.   :42827.9   Max.   :843300   Max.   :843300   Max.   :843300   Max.   :3613
   kw_max_avg        kw_avg_avg      self_reference_min_shares self_reference_max_shares self_reference_avg_sharess
 Min.   :     0   Min.   :    0   Min.   :     0             Min.   :     0            Min.   :     0.0
 1st Qu.:  3562   1st Qu.: 2382   1st Qu.:   639             1st Qu.:  1100            1st Qu.:   981.2
 Median :  4356   Median : 2870   Median :  1200             Median :  2800            Median :  2200.0
 Mean   :  5657   Mean   : 3136   Mean   :  3999             Mean   : 10329            Mean   :  6401.7
 3rd Qu.:  6020   3rd Qu.: 3600   3rd Qu.:  2600             3rd Qu.:  8000            3rd Qu.:  5200.0
 Max.   :298400   Max.   :43568   Max.   :843300             Max.   :843300            Max.   :843300.0
 weekday_is_monday weekday_is_tuesday weekday_is_wednesday weekday_is_thursday weekday_is_friday weekday_is_saturday
 Min.   :0.000     Min.   :0.0000     Min.   :0.0000       Min.   :0.0000      Min.   :0.0000    Min.   :0.00000
 1st Qu.:0.000     1st Qu.:0.0000     1st Qu.:0.0000       1st Qu.:0.0000      1st Qu.:0.0000    1st Qu.:0.00000
 Median :0.000     Median :0.0000     Median :0.0000       Median :0.0000      Median :0.0000    Median :0.00000
 Mean   :0.168     Mean   :0.1864     Mean   :0.1875       Mean   :0.1833      Mean   :0.1438    Mean   :0.06188
 3rd Qu.:0.000     3rd Qu.:0.0000     3rd Qu.:0.0000       3rd Qu.:0.0000      3rd Qu.:0.0000    3rd Qu.:0.00000
 Max.   :1.000     Max.   :1.0000     Max.   :1.0000       Max.   :1.0000      Max.   :1.0000    Max.   :1.00000
```

```
 weekday_is_sunday   is_weekend          LDA_00             LDA_01             LDA_02             LDA_03
 Min.   :0.00000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000   Min.   :0.00000   Min.   :0.00000
 1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.02505   1st Qu.:0.02501   1st Qu.:0.02857   1st Qu.:0.02857
 Median :0.00000   Median :0.0000   Median :0.03339   Median :0.03334   Median :0.04000   Median :0.04000
 Mean   :0.06904   Mean   :0.1309   Mean   :0.18460   Mean   :0.14126   Mean   :0.21632   Mean   :0.22377
 3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.24096   3rd Qu.:0.15083   3rd Qu.:0.33422   3rd Qu.:0.37576
 Max.   :1.00000   Max.   :1.0000   Max.   :0.92699   Max.   :0.92595   Max.   :0.92000   Max.   :0.92653
     LDA_04        global_subjectivity global_sentiment_polarity global_rate_positive_words global_rate_negative_words
 Min.   :0.00000   Min.   :0.0000      Min.   :-0.39375          Min.   :0.00000            Min.   :0.000000
 1st Qu.:0.02857   1st Qu.:0.3962      1st Qu.: 0.05776          1st Qu.:0.02838            1st Qu.:0.009615
 Median :0.04073   Median :0.4535      Median : 0.11912          Median :0.03902            Median :0.015337
 Mean   :0.23403   Mean   :0.4434      Mean   : 0.11931          Mean   :0.03962            Mean   :0.016612
 3rd Qu.:0.39999   3rd Qu.:0.5083      3rd Qu.: 0.17783          3rd Qu.:0.05028            3rd Qu.:0.021739
 Max.   :0.92719   Max.   :1.0000      Max.   : 0.72784          Max.   :0.15549            Max.   :0.184932
 rate_positive_words rate_negative_words avg_positive_polarity min_positive_polarity max_positive_polarity
 Min.   :0.0000      Min.   :0.0000      Min.   :0.0000        Min.   :0.00000       Min.   :0.0000
 1st Qu.:0.6000      1st Qu.:0.1852      1st Qu.:0.3062        1st Qu.:0.05000       1st Qu.:0.6000
 Median :0.7105      Median :0.2800      Median :0.3588        Median :0.10000       Median :0.8000
 Mean   :0.6822      Mean   :0.2879      Mean   :0.3538        Mean   :0.09545       Mean   :0.7567
 3rd Qu.:0.8000      3rd Qu.:0.3846      3rd Qu.:0.4114        3rd Qu.:0.10000       3rd Qu.:1.0000
 Max.   :1.0000      Max.   :1.0000      Max.   :1.0000        Max.   :1.00000       Max.   :1.0000
 avg_negative_polarity min_negative_polarity max_negative_polarity title_subjectivity title_sentiment_polarity
 Min.   :-1.0000       Min.   :-1.0000       Min.   :-1.0000       Min.   :0.0000     Min.   :-1.00000
 1st Qu.:-0.3284       1st Qu.:-0.7000       1st Qu.:-0.1250       1st Qu.:0.0000     1st Qu.: 0.00000
 Median :-0.2533       Median :-0.5000       Median :-0.1000       Median :0.1500     Median : 0.00000
 Mean   :-0.2595       Mean   :-0.5219       Mean   :-0.1075       Mean   :0.2824     Mean   : 0.07143
 3rd Qu.:-0.1869       3rd Qu.:-0.3000       3rd Qu.:-0.0500       3rd Qu.:0.5000     3rd Qu.: 0.15000
 Max.   : 0.0000       Max.   : 0.0000       Max.   : 0.0000       Max.   :1.0000     Max.   : 1.00000
 abs_title_subjectivity abs_title_sentiment_polarity     shares
 Min.   :0.0000         Min.   :0.0000               Min.   :     1
 1st Qu.:0.1667         1st Qu.:0.0000               1st Qu.:   946
 Median :0.5000         Median :0.0000               Median :  1400
 Mean   :0.3418         Mean   :0.1561               Mean   :  3395
 3rd Qu.:0.5000         3rd Qu.:0.2500               3rd Qu.:  2800
 Max.   :0.5000         Max.   :1.0000               Max.   :843300
```

Fig. 1

Given the above summaries, general information about the data can be seen, like that the average number of shares that these articles had was 3395. Also the minimum number of shares is 1, meaning that articles that received no shares on the website were not included (or no articles on the website had 0 shares). However, due to the many predictive variables present in the data set, it is difficult to gain real insight just from looking at the summary statistics, thus we begin our analysis by looking at plots of some of the variables that we think intuitively might be important.

In Figure 2 below, there appears to be a very high number of news articles receiving close to 0 shares; however, this distribution is highly skewed by a few very high outliers in the data.

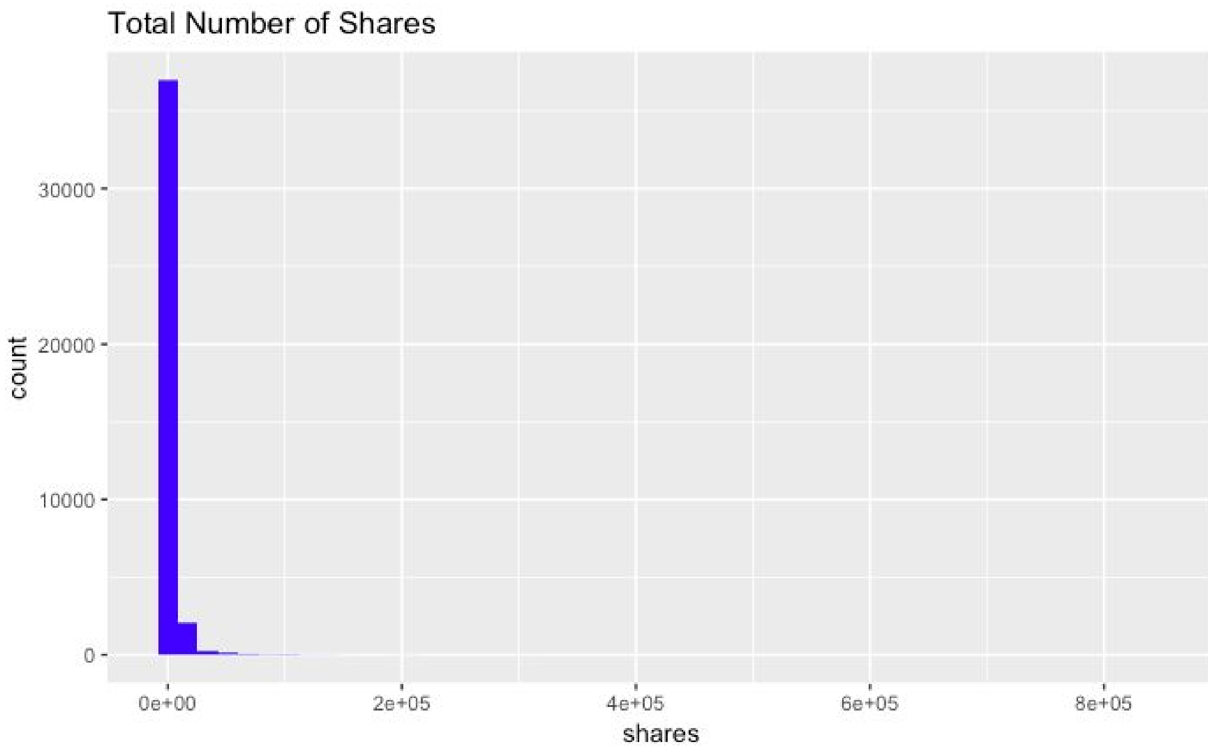


Fig. 2

The 95th percentile of the data is 10800, and if data points above this cut off are excluded, the pattern in the number of shares looks much different, as shown in Figure 3. With these much larger shares excluded, it becomes more clear that the vast majority of articles received under a few thousand shares, which makes sense given that the 3rd Quartile calculated in the summary statistics above was 2800.

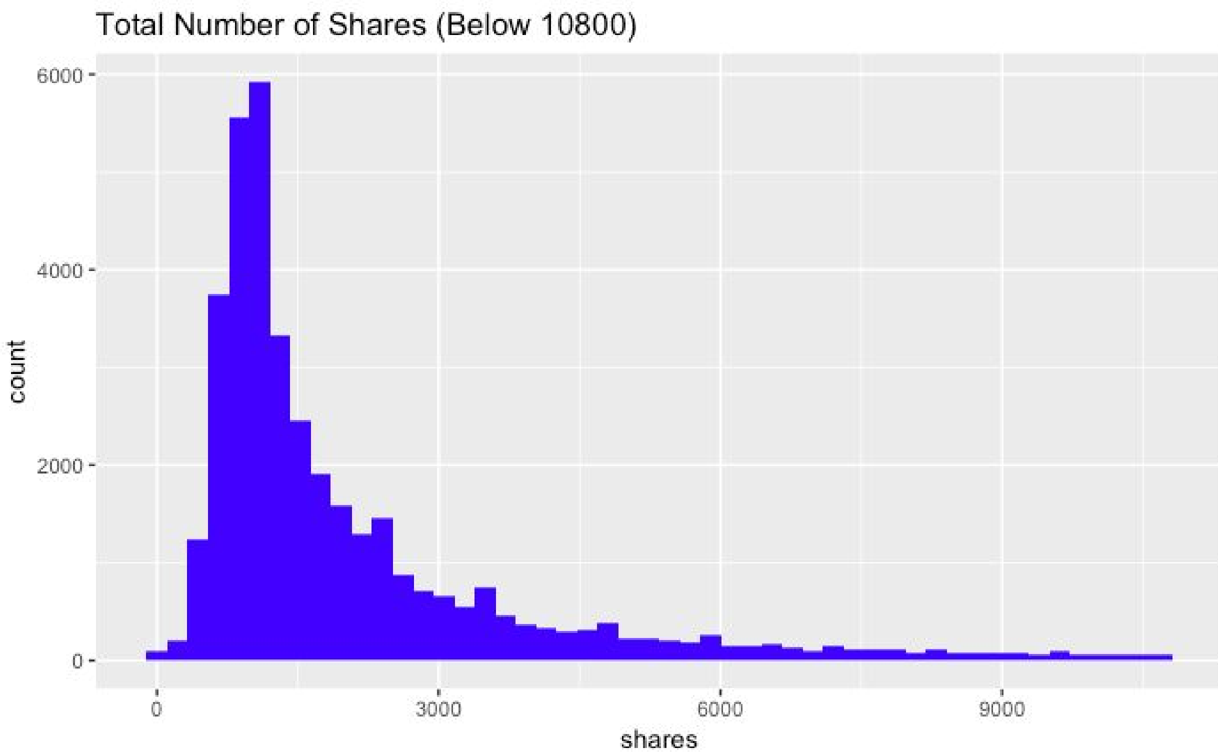


Fig. 3

We theorize that the number of words in the title of the articles may be important, as titles that are too short may not give enough information, and titles that are too long may cause people to lose interest. Thus in Figure 4 we plot the number of words in the title versus the total number of shares the articles received.

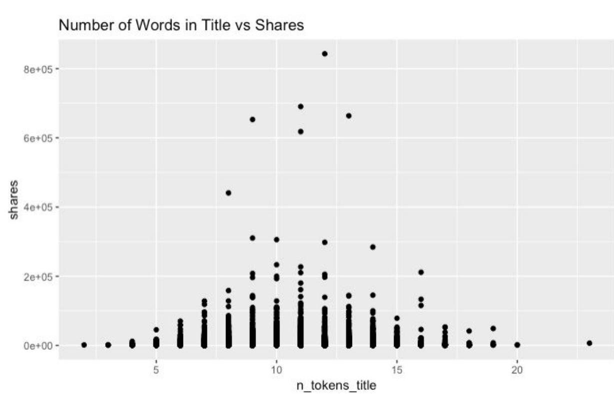


Fig. 4

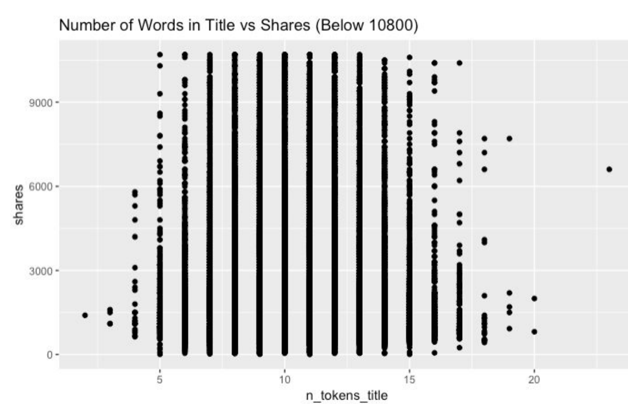


Fig. 5

From this plot, there does appear to be a relationship between the number of words in the title and how many shares the article got, with the articles that got the most shares having a “medium” amount of words, of around 7-12. However, if we again remove articles above the 95th percentile of shares, as in Fig. 5, we see that there is virtually no relationship between the number of words and the shares. This tells us that if an article gets an incredibly high number of shares, it almost certainly has a medium length title, but just because an article has a medium length title does not mean that it will get a lot of shares, and thus this variable may not be very helpful in predicting the number of shares for the majority of the data.

If we look at another variable we theorize might be important, like the number of images present in the article, we find a similar kind of issue.

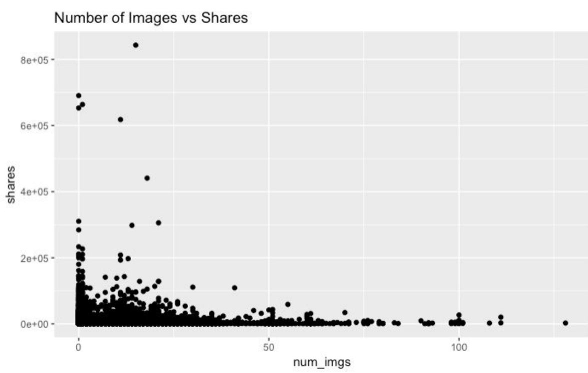


Fig. 6

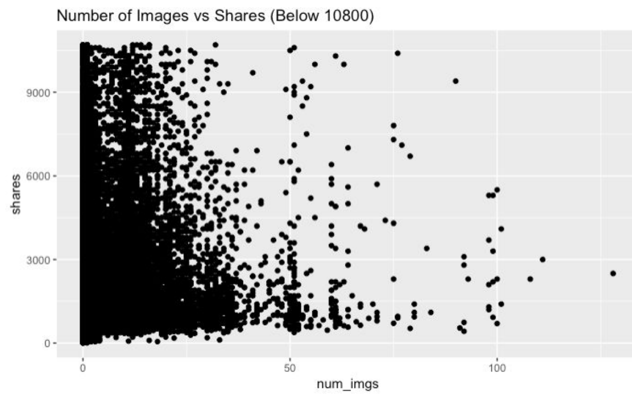


Fig. 7

In Fig. 6, it seems that high shares corresponds strongly to having fewer images and that most articles have very few images. However, when we look at the same plot with those higher shares excluded, that relationship again begins to disappear. This kind of issue happens again and again with other variables, including other kinds of variables like the number of videos, the average article length, and the rate of positive words. A pattern that appears to be very distinct disappears when the seemingly outlier articles are removed. Because features seem to affect the number shares differently depending on different tiers, this leads us to theorize that a regression tree may be effective for predicting with this data.

Another potential solution to dealing with such skewed data is to log transform it in order to try to normalize the data and use linear regression to model it. If the response variable (shares) is log transformed, it looks as below in Figure 8. While still slightly skewed, it is much closer to a normal distribution and should thus make a much better regression model.

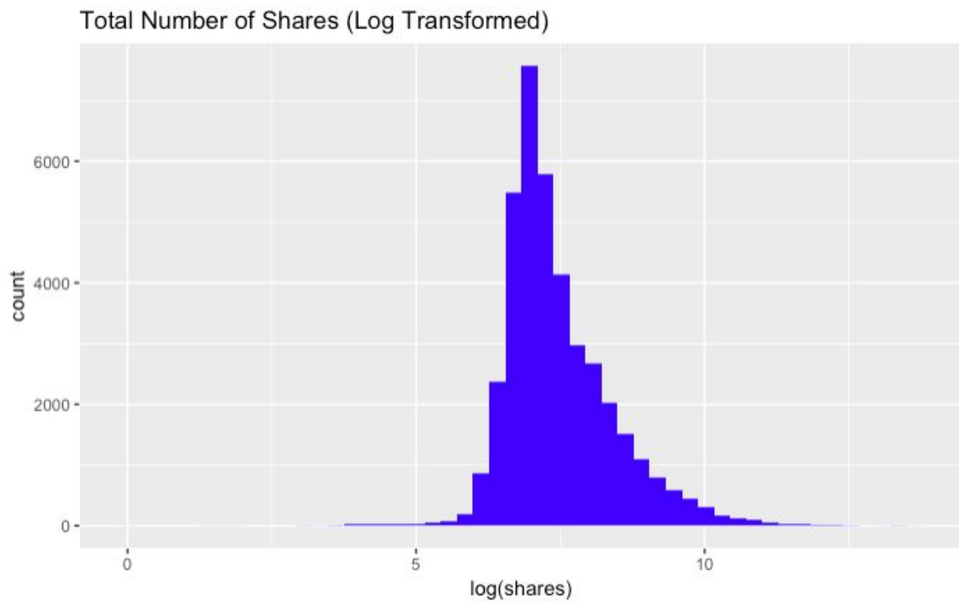


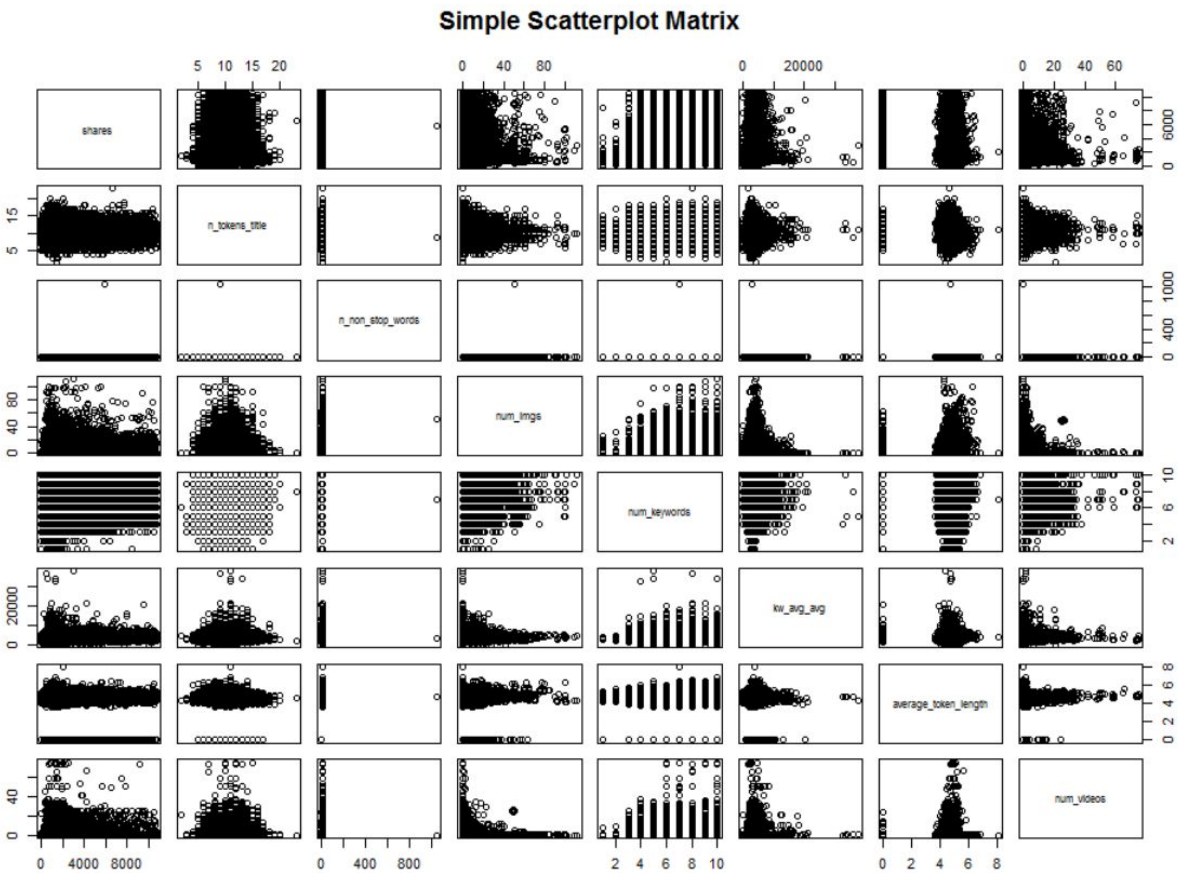
Fig. 8

III. Model Predictions

Linear Regression

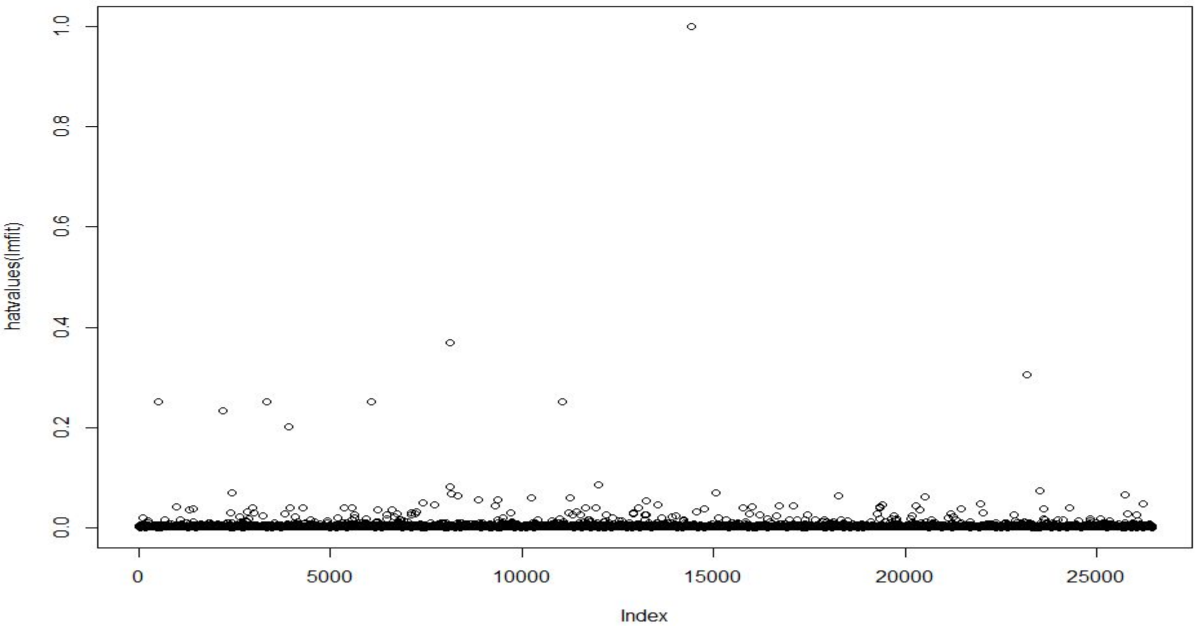
Linear Regression is one of the most basic regression models to predict qualitative features. Although our data is definitely not suited and is not easily translatable to a linear regression model, we wanted to to compare the results of an extremely basic model to a more complex model (random forests). Linear regression is not a very flexible method for predicting a model with so many variables that are not all quantitative; however there are methods such as transforming the data that might help fit the model a little better.

In order for one to perform linear regression, there are a few assumptions that must be met, but the most important one that we are looking for is that there must be a linear relationship with each independent variable and the response. After this, we would want to look at the diagnostics of our fitted model and check for normality in our residuals along with multicollinearity in our variables. From the data analysis above, we see that most of our variables do not satisfy the linear relationship condition, and to fix that, we will attempt to perform transformations that would make the relationship more linear. I selected a few important variables to plot a scatterplot matrix to see the correlation between each variable and the response.



However, we see that many of the variables in the scatterplot above have little to no linear relationship with our response (shares); as such, it is very difficult to perform any transformations on any of the variables. Other variables that are not in this scatterplot are either (binary or nominal) or have less significance than these variables. There were a few variables that we did log transform, as it did make the relationship a little more linear, but the fact that most of our variables held little relationship to our response means that those transformed variables were of little significance.

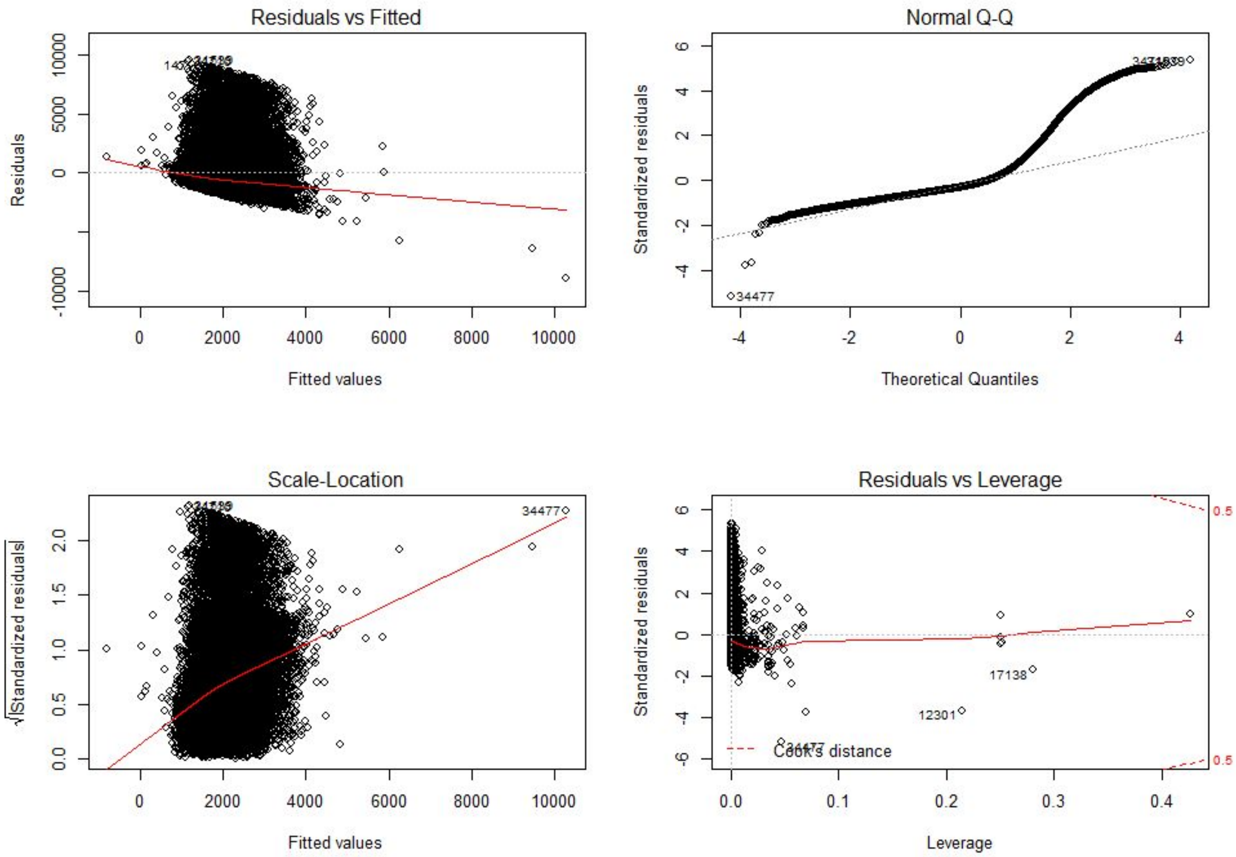
Before we proceeded with fitting the model, we found it important to remove outliers from our training set, as linear regression does not do well with extreme outliers. We justified removing these outliers because we do not want our model to be using these extreme outliers in its prediction.
Below, we see what the leverage statistics would look like if the outliers were not removed in the training set.



After removing the outliers from the training set, we began to fit the model with all of the predictors given. A small summary of the first 25 variables is given.

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	2.881e+05	9.530e+05	0.302	0.762411	
n_tokens_title	-2.719e+00	4.796e+00	-0.567	0.570774	
n_tokens_content	8.906e-02	3.798e-02	2.345	0.019046	*
n_unique_tokens	-1.050e+02	3.233e+02	-0.325	0.745310	
n_non_stop_words	-1.025e+02	9.198e+02	-0.111	0.911280	
n_non_stop_unique_tokens	-1.565e+02	2.743e+02	-0.571	0.568263	
num_hrefs	6.669e+00	1.148e+00	5.812	6.24e-09	***
num_self_hrefs	-1.534e+01	2.962e+00	-5.179	2.25e-07	***
num_imgs	4.107e+00	1.517e+00	2.707	0.006798	**
num_videos	3.941e+00	2.695e+00	1.462	0.143690	
average_token_length	-1.305e+02	4.100e+01	-3.183	0.001457	**
num_keywords	2.997e+01	6.231e+00	4.810	1.52e-06	***
data_channel_is_lifestyle	-1.957e+02	6.723e+01	-2.912	0.003597	**
data_channel_is_entertainment	-3.213e+02	4.337e+01	-7.409	1.30e-13	***
data_channel_is_bus	-3.376e+02	6.488e+01	-5.204	1.96e-07	***
data_channel_is_socmed	2.920e+02	6.300e+01	4.636	3.57e-06	***
data_channel_is_tech	2.031e+02	6.296e+01	3.225	0.001260	**
data_channel_is_world	-7.770e+01	6.381e+01	-1.218	0.223313	
kw_min_min	6.992e-01	2.728e-01	2.563	0.010389	*
kw_max_min	2.749e-02	8.935e-03	3.077	0.002094	**
kw_avg_min	-2.466e-01	5.480e-02	-4.500	6.83e-06	***
kw_min_max	-2.708e-04	1.946e-04	-1.391	0.164123	
kw_max_max	-1.556e-04	9.678e-05	-1.608	0.107932	
kw_avg_max	-7.958e-04	1.407e-04	-5.656	1.56e-08	***

We see that there are many variables that do not hold significant p-values, which led us to believe that performing feature selection on our model would improve our error greatly. Forward selection is a good criterion for variable selection where we start with no variables in the model and add more variables until it can no longer improve the model. However, before we were able run forward selection, we looked at the diagnostics of our fitted model, and found many things wrong with our fit.

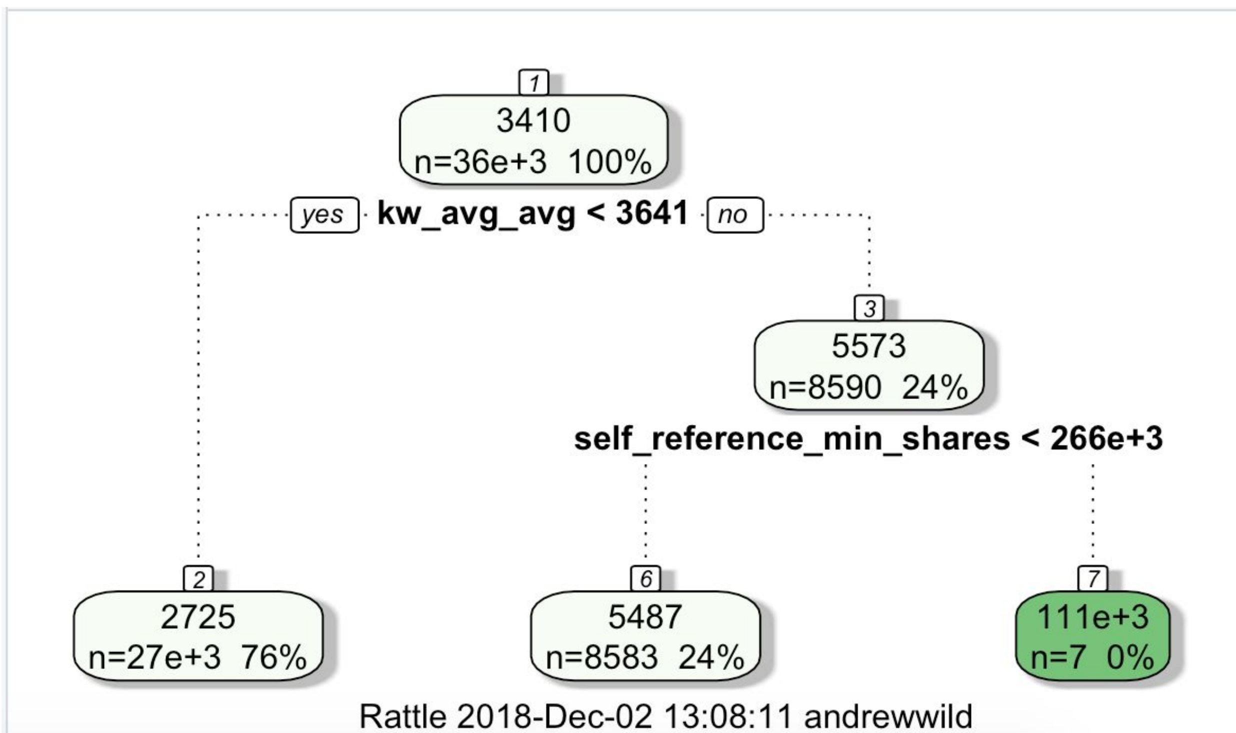


A large problem that is observed from our residuals vs. fitted plot is that most of our values are fitted between 0 and 4000, with a very large gap of residuals from 0 to 10000. Ideally, we would want our fitted values to have more spread and have residuals around 0 for the most part. Moving on to our second major problem, our Normal Q-Q plot shows that our residuals are not normal, which breaks one of the assumptions in linear regression. (One thing to note is that even if we log transformed the response variable so that our normal Q-Q plot would look linear, it actually produces a worse error than the fitted model above. Using RMSE as our loss function the log regression had a RMSE of 2800 compared to the first regression which had a RMSE of 1950).

Adding all of these problems together, it is hard to ignore the fact that our data is just not very compatible with linear regression, due to the more rigid assumptions that linear regression requires. After realizing that these problems could not be ignored, we decided not to improve the linear regression model and to move on to another method, decision trees.

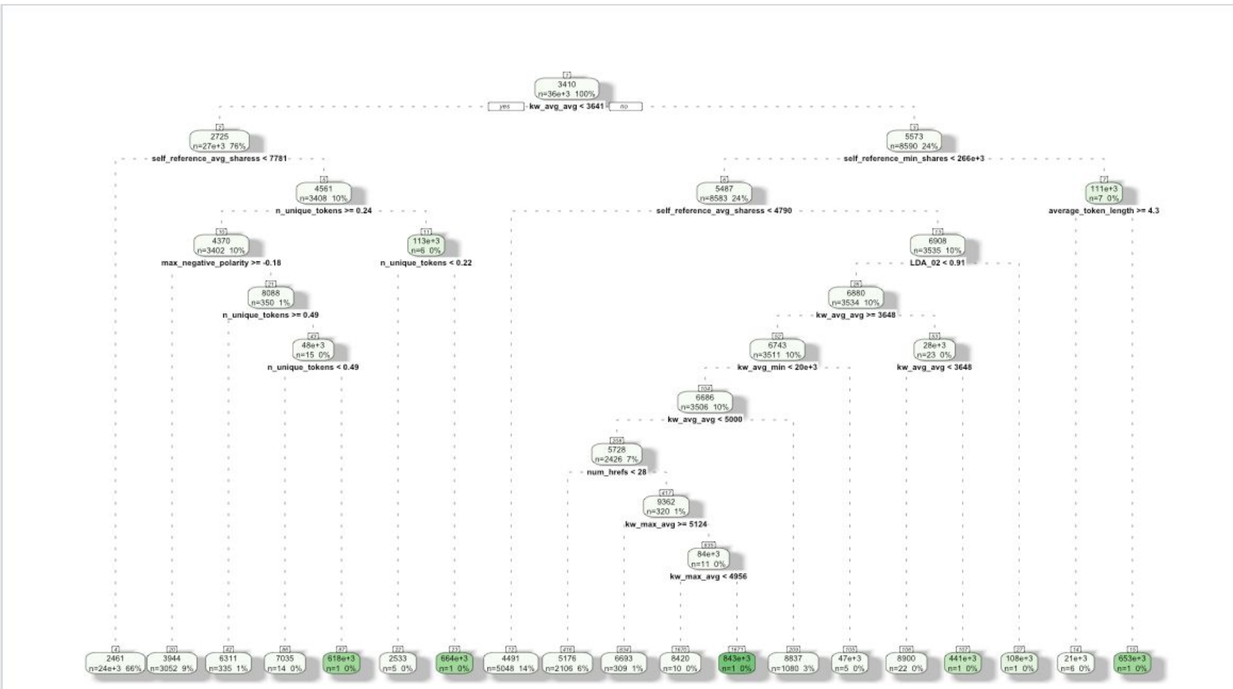
Decision Trees

The initial results from a decision tree on our training data were not ideal for deep analysis. Without making any specifications on what we want from the tree and using defaults for minimum node sizes and complexity parameter. There are just two cuts made, but the second just sorts outs seven outliers based on the minimum shares of any referenced article within the target article. 25% of the articles get sorted into a highly performing node with an average of 5487 shares, and the rest into a node with an average of 2725 shares.



To get a tree that we could dig more into, we set the node minimum to one and halved the default complexity parameter. Keywords continue to be the an important factor, but in a different version. Now, the second most important variable is the maximum amount an article with the average keyword from our target variable has received. The most important variable has now become the rate of unique words in the article, which is the sort of readability statistic we initially expected to be more predictive of the article’s response. The average length of words in the article is the third most important variable.

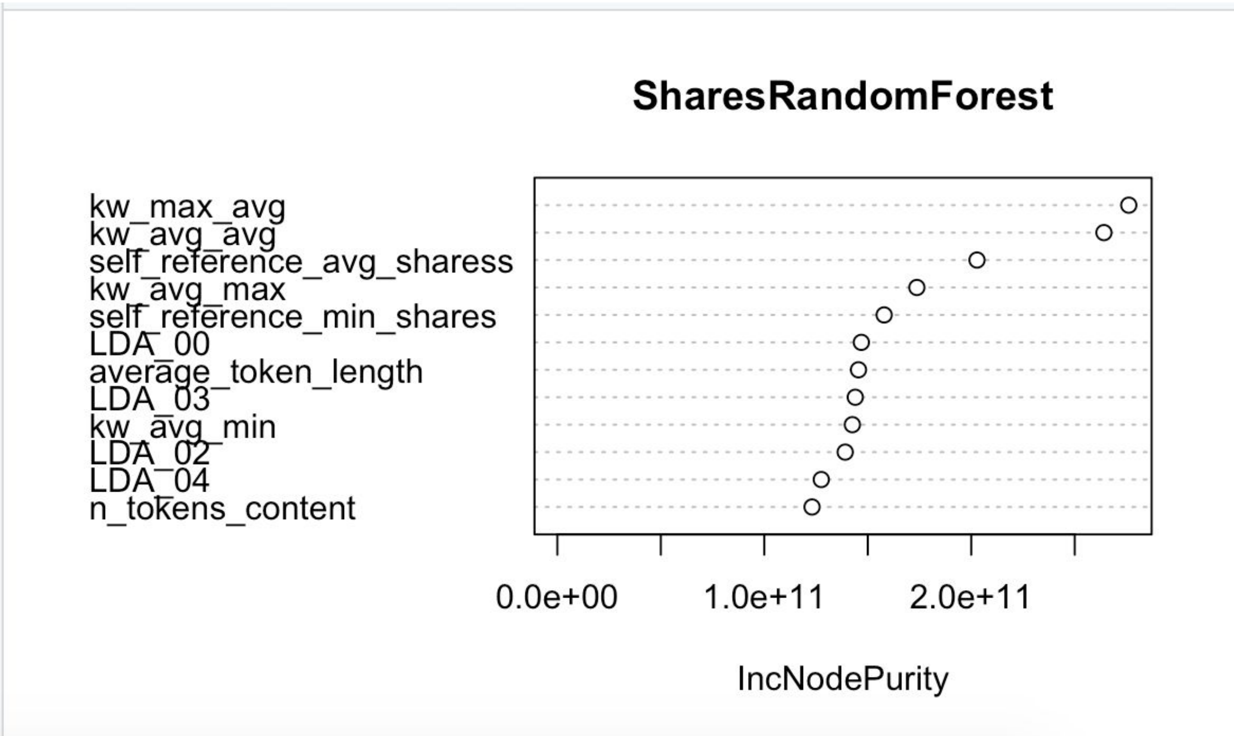
```
Variable importance
           n_unique_tokens                 kw_max_avg       average_token_length
                        32                         29                         14
                kw_avg_avg  self_reference_min_shares self_reference_avg_sharess
                        10                          4                          3
  n_non_stop_unique_tokens  global_sentiment_polarity  self_reference_max_shares
                         2                          1                          1
                kw_min_avg                   num_imgs                     LDA_03
                         1                          1                          1
```



We now get a much more satisfying five nodes with more than 2% of the dataset, with the highest predicted value for a large sized node being predicted at nearly 9,000 views. The largest node contains 66% of the dataset and is projected for a low average of under 2,500 views, and occurs for low values of average keyword and referenced article performance.

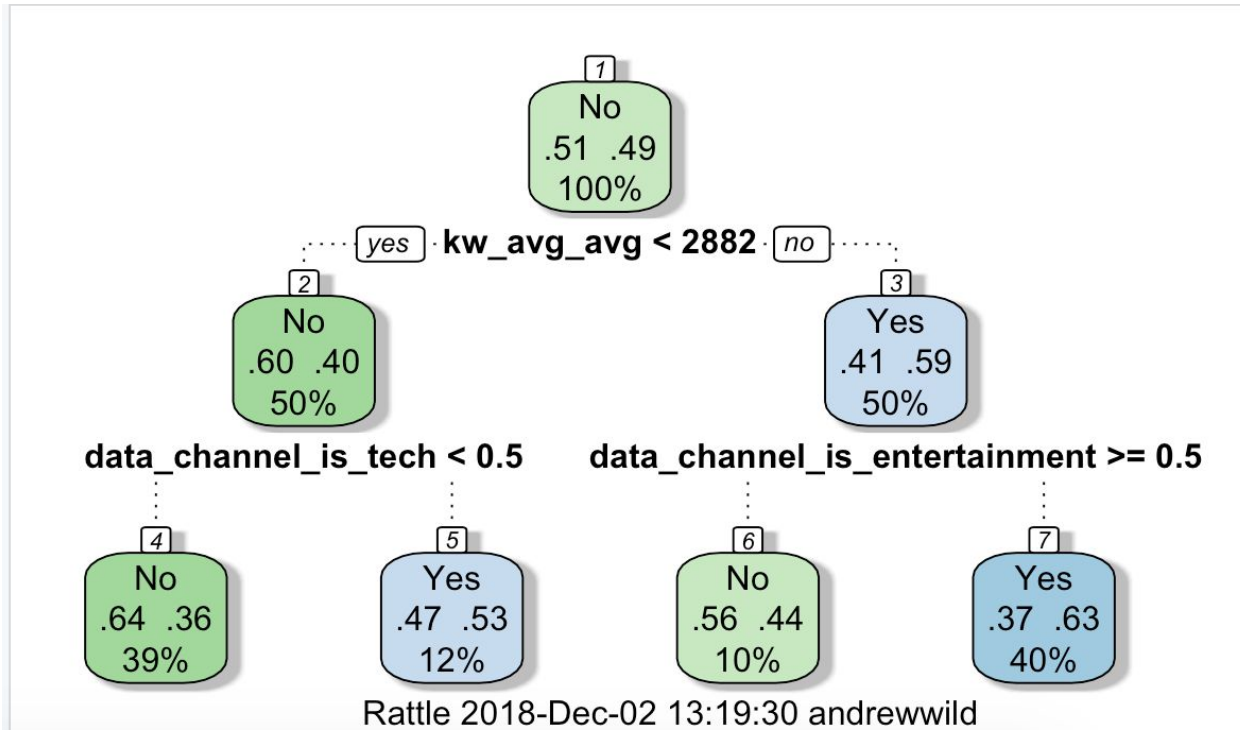
The mean absolute error on the test set for this tree is roughly 3,345, which is obviously not ideal when the median number of shares is less than half of that. We can account for some outliers skewing the error values by looking at median error, but that comes out to 1,590, still above the median share value.

A random forest model proved to surprisingly be no better, with a mean and median absolute error of 3,263 and 1,672 respectively, and from what we can tell from the variable importances, it worked similar to our other trees, placing primary importance on the keywords in the article followed by referenced articles.



We know from the heavy skew of digital shares in general that trying to predict the actual number of shares is a somewhat doomed effort, so we moved onto trying to model a more realistic target. To think like an editor, we built a decision tree where the target variable was whether or not the article had a shares above the median value, which we can see an editor gauging as being the most basic assessment of whether an article performed well or not.

We got a four node tree with three decision splits, the first being on the average keyword performance, as we saw in the first, most basic tree we built. The next two decisions were entirely new in our analysis however, and did seem to model how editors think.



It simply came down to the subject of the article. For articles with poor performing keywords, tech articles did the best and that's what an editor would want to push generally. For good keyword performance, it counter-intuitively seemed that entertainment focused articles did worse, and editors would push their writers in different directions. We can't account for exactly would make an editor think those choices are right, but the pattern of using keywords and article subject seems realistic. This very simple tree predicted whether or not an article would perform above the median 61% of the time, which while not astounding, seems like it'd be a very useful tool for online publishers.

IV. Conclusion

The heavy skew of the data meant that any sort of unadjusted linear regression was certain to have poor results, but it was worth doing to see what variables still had some amount of predictive power. In an attempt to combat this issue while still making use of the linear regression model, the response variable was log transformed before using it within the model again; although this did solve the problem of our residuals not being normal, it actually produced an error larger than our initial regression. The initial decision trees did not fare much better, but the failure of a black box system like random forest was more surprising. We knew from both how hard it is to predict online performance for those in the industry and the skew of the data that there weren't going to be obvious well-performing systems, but it seemed that perhaps the sheer number of trees inherent in random forest were going to strike on something, but it wasn't the case.

If we consider the real-world application of modeling this data, it becomes clear that trying to predict the exact number of shares that an article would receive is not something that an editor would be particularly interested in, and thus a measurement of error like the mean and median prediction errors are not particularly helpful in this context. An editor is much more likely to be interested in if their article will generally perform “well” or “poorly.” Thinking like an editor and trying to gauge more qualitatively how an article would do was a more successful effort. Although being able to predict whether an article will do above or below the median performance with accuracy a bit above 60% won’t redefine the industry, it’s a legitimate result. A classification tree that can predict qualitative article success with greater than 50% accuracy could be a useful model for people working in this industry to at least get a baseline for how their article will perform.