

Predicting the Popularity of Mashable.com News Articles

Josh Wilder, Chi-Hua Wu, Vicky Pang, Akshay Pokharkar, Anant Khandelwal

University of Connecticut

Predictive Modeling - OPIM 5604

Professor Jennifer Eigo

04/26/2019

**Table of Contents**

[**Executive Summary**](#_xukh2fppfgwa) **2**

[**Methodology**](#_pyojlgmokrfd) **3**

[Methodology - Sample](#_nukrkn2rcfbi) 3

[1. The Distribution of Shares](#_mb3ngwqfkmby) 4

[2. The Correlations Between Variables. COME BACK TO THIS](#_fw2qpodxizj8) 4

[3. Missing Values and Outliers.](#_vdkabls47vgc) 5

[Methodology - Modify](#_qqy7ucm3wjp) 5

[1. Excluding Missing Data.](#_6vkipmgs1lot) 5

[2. Excluding Outliers.](#_snsycy4hht8w) 6

[3. Principal Component Analysis.](#_rtj55ms6e4tt) 6

[4. Combined Dummy Variables into One Categorical Variable.](#_w8z2uayyd6on) 7

[5. Creating an Alternate Decision Variable.](#_b6bt779l3bej) 7

[6. Variable Selection.](#_gkzoyn6j1wh6) 7

[7. Undersampling.](#_w8be5xgjej4o) 8

[Methodology - Models](#_isr439usxhv6) 8

[1. Logistic Regression.](#_s2iudkbfm08e) 8

[2. Decision Tree.](#_q38ckwvu9o9y) 9

[3. Bootstrap Forest.](#_ievump7wrm9n) 9

[4. Boosted Tree.](#_l9nqcx1uw22k) 10

[5. K-Nearest Neighbor (KNN).](#_ucqic2iznkwq) 10

[6. Neural Network.](#_wi6777mok78g) 10

[7. Linear Discriminant Analysis.](#_xwhmn0tbj65c) 11

[**Results**](#_3uu5xrc8z57x) **12**

[**References**](#_bp007gyk4kg3) **14**

[**Appendix - Tables**](#_eyifmv5xvhtj) **15**

[**Appendix - Figures**](#_gp0v22479sxq) **17**

# 

# Executive Summary

In order for online news companies like Mashable to succeed, they need to determine patterns and trends that contribute to the popularity of their models. Our goal was to create and develop a model predicting which Mashable articles were widely shared on social networks based on several features of online news. Based on the results, we determined which variables would contribute toward future content creations having a wider reach on social media through organic sharing.

Our business insights and recommendations for Mashable are based on our logistic regression model, which was implemented on a dataset built using stratified undersampling. This model provides multiple insights and recommendations that will help Mashable improve their business. These insights includes the impact of image insertions, article categorization, keyword strength, and article release day. We then conclude with direction and specific actions Mashable could take to improve their articles’ virality.

**Problem Statement**

Mashable is a global, multi-platform media and entertainment company, and they post articles of multiple genres online from which they earn revenue from advertisers. In order for companies like Mashable to succeed, they must be aware of the trends within their successful articles. Without learning these trends, a company like Mashable may fail against competitors who also uses data-driven strategy. Therefore, it is imperative to understand and predict what article characteristics are most appealing to readers.

Our solution to this problem involves predicting which articles are widely shared on social media. Shares on social media is a key metric for article virality, and the specific business insight we are looking for is which factors lead to higher article virality. This is very significant for business because it results in more article views without requiring paid marketing.

**Dataset Introduction**

The dataset is called “Online News Popularity Data Set” and it can be accessed from UCI Machine Learning Repository (https://archive.ics.uci.edu). Each row represents a news article and was collected from January 7, 2013 to January 7, 2015. There are 39,797 rows and 61 columns as shown in the Appendix Table 1. The original dataset contained 37 attributes of articles, and several natural language processing features were extracted by previous researchers (Fernandes, Vinagre & Cortez, 2015). In this study, 17 selected predictors are used to build models.

# Methodology

The methodology of this study is followed by SEMMA. SEMMA stands for Sample, Explore, Modify, Models, and Assess (Shmueli, Bruce, Stephens & Patel, 2017, p. 18). During this process, our data visualizations were created by both Tableau and JMP software, and the models were built via JMP software.

## Sample

In the Sample section, the dataset was partitioned into 50% training, 30% validation, and 20% testing set split. The partition was created by JMP (with the fixed random seed of 1234). The training set was used for building models, the validation set was used for accessing the model to avoid overfitting problems, and the test set was used for selecting the best model. However, the results of data visualization showed that this dataset is imbalanced, and we later used undersampling to resolve this problem.

**Explore**

During data exploration, we discovered anticipated and unanticipated relationships between variables. We began our data exploration focusing on our target variable, “shares.” We then continued to use visualizations on all of our data to detect outliers, missing values, abnormal features, and highly correlated predictors.

### 1. The Distribution of Shares

The distribution of shares resembles a Johnson Su distribution, as shown in Appendix **Figure 1**. This distribution was very skewed. Although the majority of the articles have between 1,000 and 3,000 shares, there are thousands of articles with over 10,000 shares, and the highest shared articles had hundreds of thousands of shares. The mean number of shares were over double the median, which is a sign that the distribution was very skewed to the right. Please see Appendix **Figure 2**. We determined that this distribution was a problem, because we would expect models like linear regression to be very sensitive, but wildly inaccurate with very highly shared articles. On the other hand, very highly shared articles were the most important in terms of business results, so keeping them in the data is desirable.

### 2. The Correlations Between Variables.

As the color maps of correlation shows in Appendix **Figure 3**, there were several variables that had high correlations. This meant that they provided similar or overlapping information that could be overly repetitive during the prediction of a target variable. These variables often were similar in meaning, for example, “self\_reference\_max\_shares”, “self\_reference\_min\_shares”, and “self\_reference\_avg\_shares,” whose correlations were close to 1.

### 3. Missing Values and Outliers.

As shown in Appendix **Figure 4** (the portion of scatterplot Matrix), there was a row with abnormally high values for multiple attributes. This included values above 1 for variables that are meant to rate from 0 to 1. In the next section, we detected and excluded them in order to build a robust model to avoid the situation that the extreme values would highly affect the model.

There are 1181 rows which contained a large portion of zero values in “n\_token\_content” to “average\_token\_length”, “self\_reference” related variables, and “global\_subjectivity” to “max\_negative\_polarity.” These zeroes suggest, for example, that these articles have no words. Viewing a handful of these articles allowed us to confirm our suspicion that these articles actually did have words, and there was a flaw in data collection. Therefore, we viewed these rows as ones containing missing values.

## Modify

Using exploratory analysis and visualizations, we found high values that were considered potential outliers. This section would include statistical methods used to detect outliers. This section would also include data preprocessing, undersampling, and variable selection.

### 1. Excluding Missing Data.

Although no data was “missing” as represented by a dot in JMP, this dataset did contain missing data. As shown in Appendix **Figure 5**, we found rows that had zeros in multiple columns. Our challenge was differentiating between zeroes that genuinely represented zeros and zeroes that were just missing data. After checking a couple of articles to make sure that there weren’t wordless video articles, we went ahead and deleted all rows, which had zero in the column. We continued to do this analysis for multiple columns, and some of which included missing data as -1’s instead of zeros (over 2,000 rows were excluded). One downfall of our strategy was that we deleted rows based on missing data in columns that we later deleted. However, we did our best to un-exclude rows that could be used after excluding the problem columns, but this procedure became tedious to do each time. The outcome wouldn’t have had enough impact on the results anyway.

### 2. Excluding Outliers.

Since the popularity of news varies, we applied JMP’s interquartile range outlier detection technique to examine our data, setting tail quantile equal to 0.1 and Q equal to 10. The results are shown in Appendix **Figure 6**. It showed that there were three variables that contained wrong data, “N\_tokens\_content”, “non\_stop\_words”, and “n\_non\_stop\_unique\_tokens” are variables that should be rates between 0 and 1, and values outside of this range are invalid data. The results showed that these invalid values were outliers and we decided to exclude them, and we excluded 1,208 rows of data. Since there were 39K rows of total data, excluding 1,208 rows did not have a detrimental effect on the size of our dataset.

### 3. Principal Component Analysis.

As the result showed in the color map, the correlation of keyword-related variables concluded that they both had positive and negative relationships in between. We then applied the principal component analysis to reduce the number of variables, and then we decided to choose three components to represent six similar variables. The results can be found in Appendix **Figure 7**, and it showed that three components can stand for 84.546% of information.

### 4. Combined Dummy Variables into One Categorical Variable.

Due to the dataset containing several dummy variables in published articles, this would cause problems to have n classes in a model instead of n-1. According to what we’ve learned in our lectures, the JMP software is easier to handle categorical variable rather than several dummy variables. As seen in Appendix **Figure 8**, we used the formula in JMP to create the categorical variable.

### 5. Creating an Alternate Decision Variable.

After observing the skewness of the distribution for shares, we considered altering our response column. We also considered a slight reframing of our business goal, would it make more sense to predict the number of shares that each article gets or should we predict a group of articles that would account for a disproportionality high number of shares? Attempting to determine the feasibility of the second idea, we created a column with a binary response where a 1 is an article which had at least 2900 shares. This came out to be the top 24% most shared articles, and they accounted for 71.68% of the total shares of all Mashable articles on social media. Based on this data exploration, we decided it would make sense (in terms of our business problem) to focus on this group of important articles. We created “IsAtLeast2900Shares” and used it as our target variable. The formula is as shown in Appendix **Figure 9**.

### 6. Variable Selection.

The results of Natural Language Processing usually depends on the researcher’s interpretation and the programming methods. Since the variables contained a large portions of natural language processing, this procedure decided to select a few of them into the model. Based on the results in data exploration and our best business judgement, we selected universal variables for building models, and the models were not limited to all of those variables. As shown in **Table 2** in Appendix, it contained all the variables we selected for building models.

### 7. Undersampling.

The unbalanced proportion in shares over 2900 and less caused the model to predict the class with a large proportion. We undersampled to create a 50/50 balanced proportion of 1’s and 0’s in the response column (in the training and validation data). We used this dataset to build and tune most of our models. The proportion of the data can be found in Appendix **Figure 10**.

## Models

Moving forward, we created our models, which were able to predict categorical variables. This procedure would apply Logistic Regression, Decision Tree, Bootstrap Forest, Boosted Tree, K-Nearest Neighbor, Neural Network, and Discriminant Analysis to predict the desirable outcomes. During this procedure, we’ve tried several combinations of predictors to build models, and based on each results, we will select the best model.

### 1. Logistic Regression.

After our original variable selection, we reduced the numbers of predicting variables by implementing stepwise selection. Regardless of whether we used forward, backward, or mixed selection, the same 17 of the 22 considered variables were always chosen, counting each category of the categorical variable “data channel” as separate variables. We conducted stepwise selection by using Max Validation R-Square as the stopping rule except when possible, but based on p-values. The results can be seen in Appendix **Figure 11**.

Since we were dealing with a “rare event” problem, we decided to use a stratified undersampling dataset to see if it would improve our logistic regression model. We built this second model using the same stepwise selection process. As the results shown in Appendix **Figure 12**, the model out performed our original logistic regression (model comparison is explained in ASSESS section).

Although we acknowledged that this was insufficient evidence to prove that stratified undersampling would improve all of our models, we decided to build the rest of our models on the dataset we built using stratified undersampling, because of the improvement we saw here. A second reason for this choice was that it would eliminate the need for using alternate cutoffs for each model.

### 2. Decision Tree.

We tried a decision tree on the original dataset, but it found less than ten true positives. This section would detail the decision tree implemented on the stratified dataset, and both models used “IsAtLeast2900Shares” as the target variable. During splitting and pruning, we found that the optimal number of splits were 12, achieving the best possible validation R-square, of 0.1037. Screenshots of these models can be found in the Appendix **Figure 13**.

### 3. Bootstrap Forest.

As single trees may not have good predictive ability, we combined results from multiple trees to improve the performance. The multi-tree approach that we used was the Bootstrap Forest technique, which helped us make the most out of the available data. This method had a lesser chance of overfitting and was not sensitive to outliers. We proceeded with 50 to 100 to 1000 of trees, and later found that 50 trees showed that the accuracy was lower. For 100 trees, the accuracy was higher and any trees larger than 100 almost yielded the same results. As shown in Appendix **Figure 14**, the results we obtained from 100 trees were as follows: RMSE = 0.4612 and Accuracy = 65.47%

### 4. Boosted Tree.

The advantage of using the boosted tree technique was that it supported loss function. Each subsequent tree was designed to correct errors of ones that came prior, in this case it helped boost the performance where it made mistakes. By keeping the number of trees as 100, the RMSE came out to 0.4694 and the Accuracy came out to 66.46%. The results is shown in Appendix **Figure 15**.

### 5. K-Nearest Neighbor (KNN).

The most challenging aspect of KNN models were software crashes. When we used all of the dataset (39,644 total row numbers) to build a model with data partition rule 50-30-20, the JMP software crashed for two group members who has attempted to build a KNN model on two different computers. A third group member tried KNN on the undersampled dataset with 23,617 total row numbers and succeeded. We believed that the reduced number of rows made computation easier. We used the same variables that we used in the regression model after the stepwise selection. Tuning to minimize validation set misclassification rate, we decided to use K equals 81, considering all Ks are up to 100. The accuracy of this model was 69.307%, as shown in Appendix **Figure 16**. Unfortunately, the KNN models had limited interpretation. Aside, suggesting “make articles similar to ones that have succeeded in the past,” there was little to recommend for business without actually using the model.

### 6. Neural Network.

During this procedure, we used the same variables that we used in the regression model after the stepwise selection. We tested a multitude of different neural network structures in terms of the number and types of nodes in each hidden layer. Sparing the details, we ultimately found that a two hidden layer model with 3 tanh nodes, 3 linear nodes, and 4 gaussian nodes in both layers worked best. We implemented this model using 12 tours in order to improve accuracy. The difficulty of model interpretation limits the amount of business insights that could be drawn from this model. The results can be found in Appendix **Figure 17**.

### 7. Linear Discriminant Analysis.

We used stepwise selection in order to achieve the lowest validation misclassification rate. The resulting model only used four predictors: the first two Keyword Strength principal components, self\_reference\_avg\_shares, and num\_hrefs. As shown in Appendix **Figure 18**, the validation misclassification rate was 31.95%. This model was limited because it could only be used for continuous predictors and it does not determine probabilities. Therefore, alternate cutoff points could not be considered.

**Assess**

The first two models that we compared were both logistic regressions that also used the stepwise selection to select predictor variables. The first model was built on the original dataset (with excluded data), and the second model was built on the undersampled dataset. At first, we thought to compare the models using RMSE, Accuracy, or Accuracy of 1’s, but realized that these were all problematic. The first model used a cutoff of .50, which excelled in terms of these test metrics, but only actually found 149 true positives, while the second model found over a thousand. We then realized that we could compare the models using test AUC (area underneath the test ROC curve) in order to quickly compare these models considering all possible cutoffs. We knew that selecting a specific alternate cut off wasn’t necessary for business recommendations.

The second model, which was the logistic regression model was built on the undersampled dataset, had high test AUC. Therefore, we selected this as the best model, and decided to continue testing models built on the undersampled data. Due to the structure of the undersampled data, models built upon it did not have an issue with finding too few true positives. We compared all of these models using RMSE and Accuracy, both of which ranked our models the same way. We found that our KNN model was the best model statistically, but decided it would not help us make good business insights. Due to interpretive ability, we chose the logistic regression model as our best model for making business insights and recommendations. Please see model comparisons in Appendix **Figure 19**.

# Results

Our best model gave us a better understanding of which variables were most and least significant towards shared articles. One variable that should be eliminated is the number of images. We discovered that it didn’t have a big impact whether included or not. Based on variable significance and the formula of our model, we believe that it’s best that Mashable create less world, business, and entertainment articles (Appendix Figure 15). We also recommend Mashable to create more articles categorized as social media, tech, lifestyle, and similar articles that would help increase their shared numbers. The results of our logistic regression model also focuses on keyword strength. Based on the statistical significance, we believe that having popular keywords is crucial to having a popular article. Another important factor of successfully shared articles, results also focuses on the popularity of shared articles on a weekend.

We made this recommendation to Mashable based on assuming the goal of maximizing shares on social media. We also recognized that although making more social media articles may not translate to more total views or profit. In terms of business practice, we would recommend having at least a couple employees whose main job is to improve keyword strength. We believe that an expert in this specific task would do much better than writers or editors, and would have a significant impact on article virality on social media. To measure success, we also understand that is important to also engage the readers by providing current news in real-time and especially before competitors. However, for all other articles that are less time-sensitive, they should be published on the weekend in order to reach a larger audience particularly through social media shares.

**Conclusion**

In conclusion, in order to improve social media virality, we suggest Mashable focus most on publishing the right types of articles, with the right keywords, at the right time. Specifically, efforts like publishing articles on the weekend would cost Mashable little to nothing and would greatly increase social media visibility, while efforts like including many images in articles cost Mashable time and money and achieve very little. With our insights and recommendations, we believe that the average popularity of Mashable articles on social media would increase considerably.

# 

# 

# References

Taylor, C. (2016, October 26). How Internet Affects the Newspaper Business. Retrieved from:

https://smallbusiness.chron.com/

Bryant, M. (2016, 3 Mar).*“20 Years Ago Today, the World Wide Web Was Born”,* TNW Insider*.*

Retrieved from: https://thenextweb.com/insider/

Laird, S., & Laird, S. (2012, April 18). *How Social Media Is Taking Over the News Industry*

[INFOGRAPHIC]. Retrieved from: https://mashable.com/

Shmueli, G., Bruce, P. C., Stephens, M. L., & Patel, N. R. (2017). *Data Mining for Business*

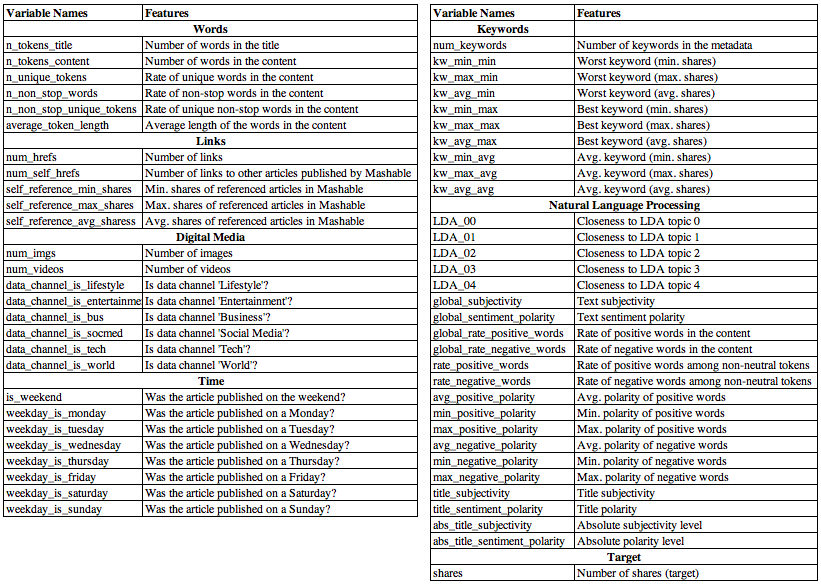
*Analytics: Concepts, Techniques, and Applications with JMP Pro.*John Wiley & Sons.

Fernandes, K., Vinagre, P., & Cortez, P. (2015, September). *A proactive intelligent decision*

*support system for predicting the popularity of online news*. In Portuguese Conference on Artificial Intelligence (pp. 535-546). Springer, Cham.

# Appendix - Tables

**Table 1**. List of attributes by category.



**Table 2**. Variable Selection List

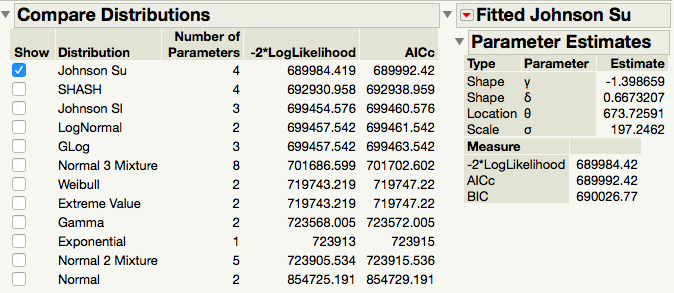
# 

**Table 3**. The Comparison of models

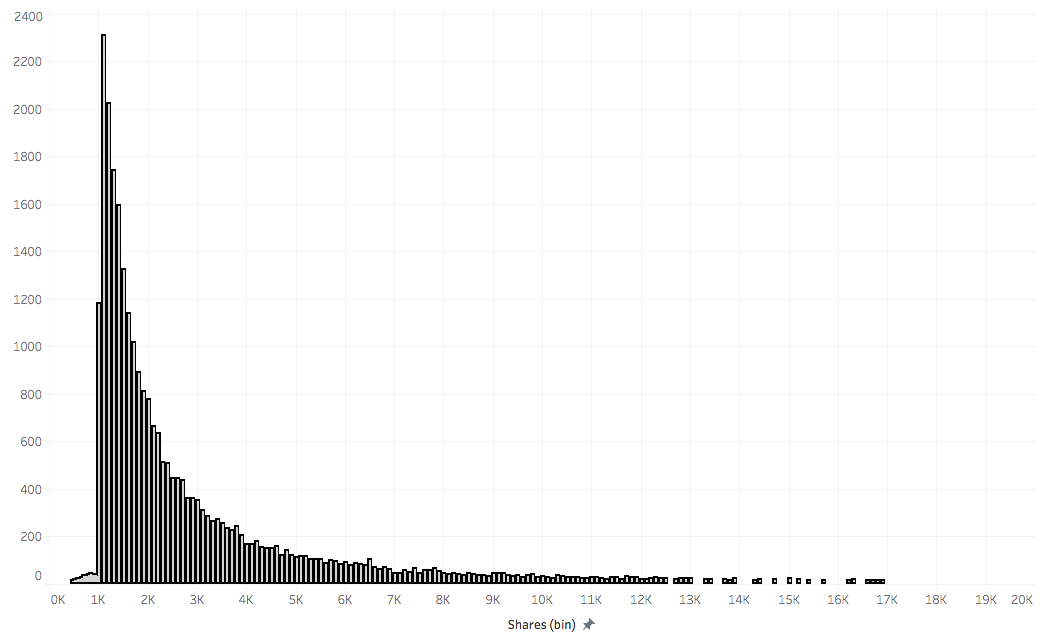
# 

# Appendix - Figures

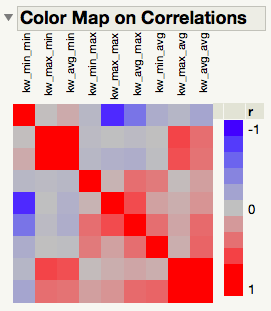
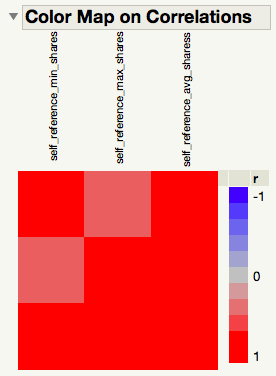
**Figure 1**. The Result for the Distribution for the number of shares



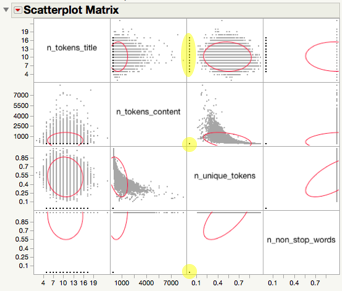
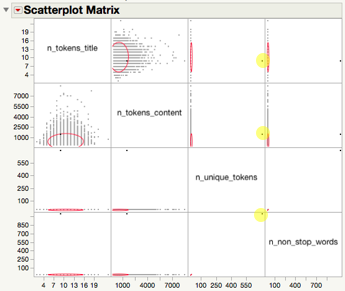
**Figure 2**. The distribution of the count of shares when shares under 20K.



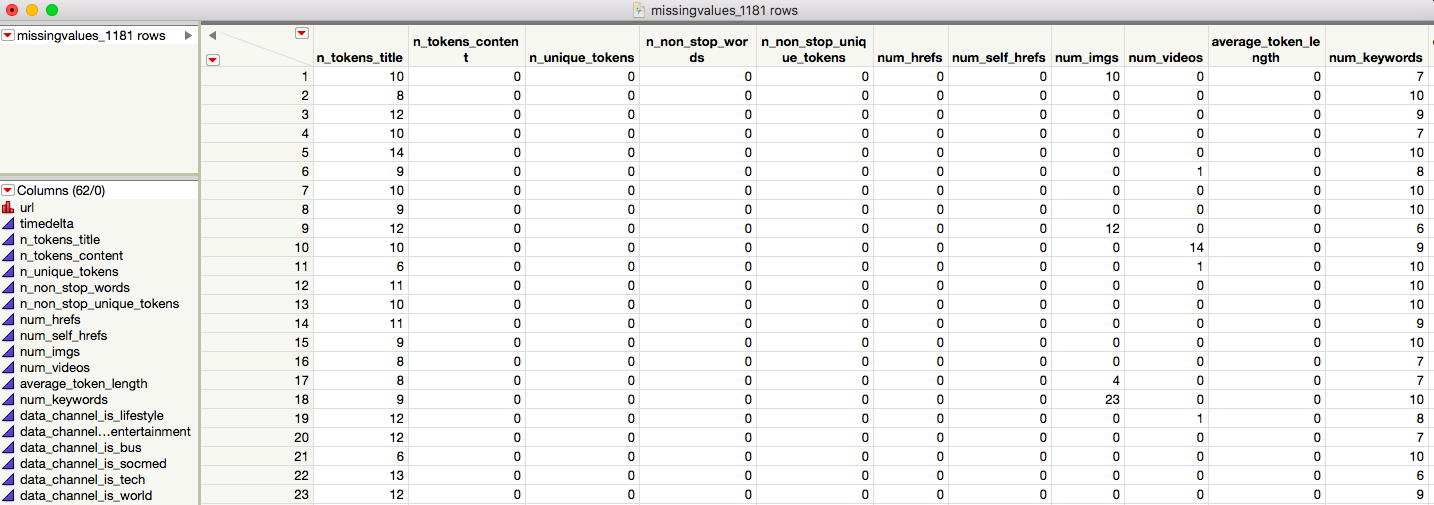
**Figure 3**. The color Map on Correlations



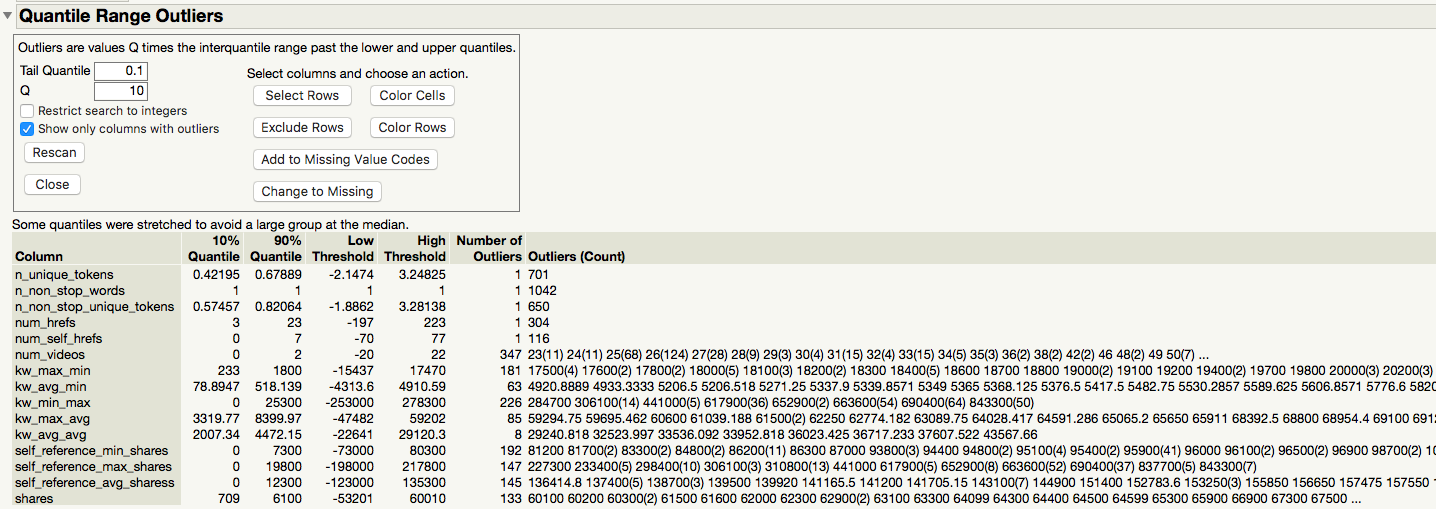
**Figure 4.** Scatterplot Matrix (a portion of the variables)



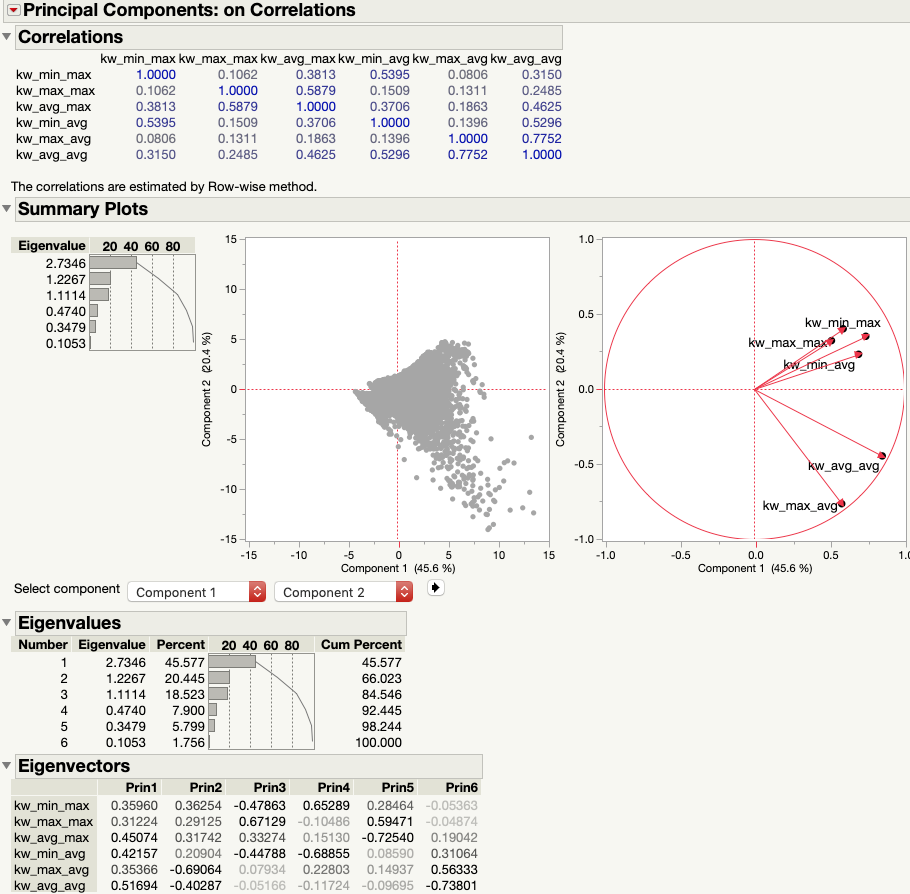
**Figure 5**. An Example of Rows with Missing Data Shown as Zeroes.



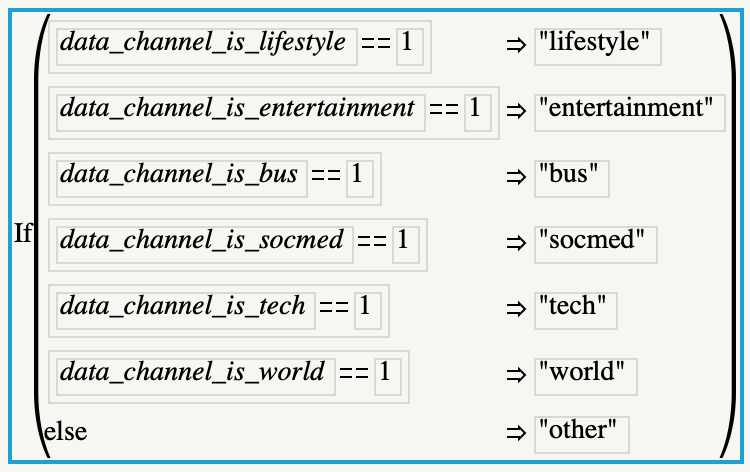
**Figure 6.**  The results of using Quantile Range Outliers.



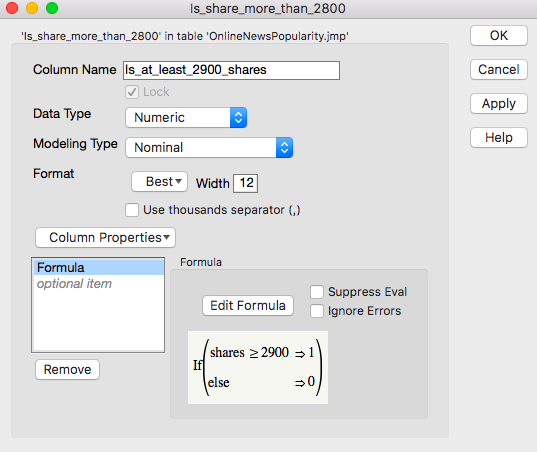
**Figure 7**. The Result of Principal Components Analysis



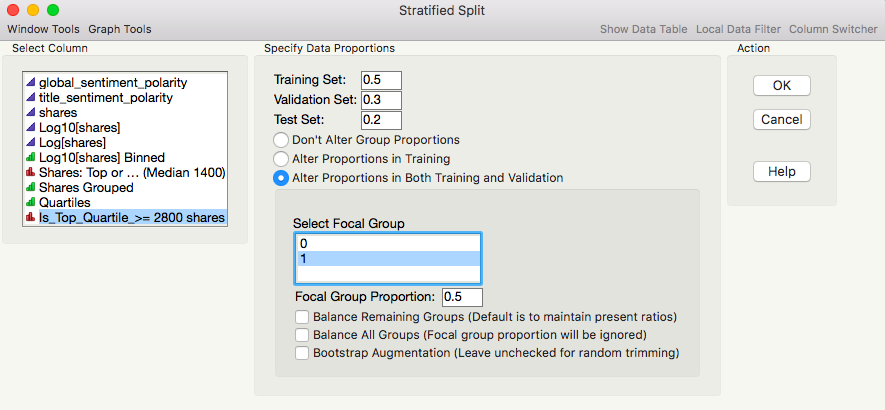
**Figure 8**. The Formula to Combine Dummy Variables into One Categorical Variable

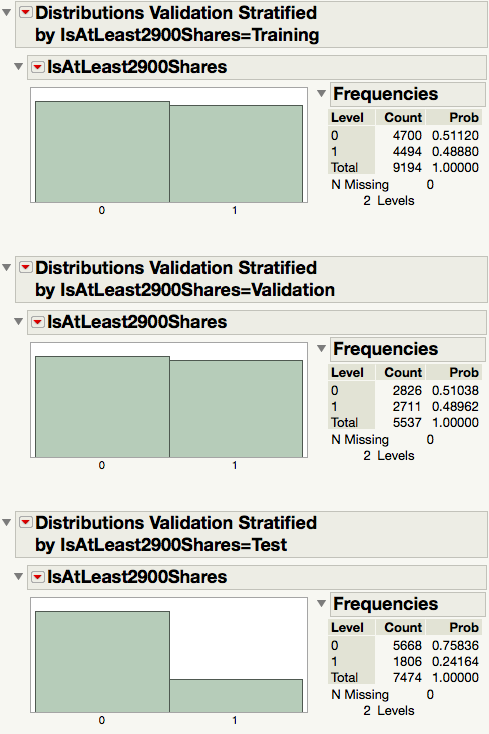


**Figure 9**. The Formula to Create an Alternative Decision Variable

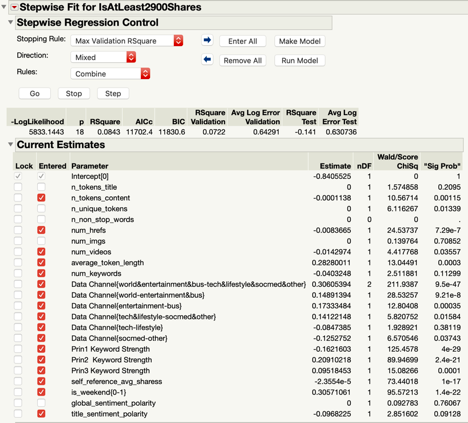
****

**Figure 10.** The Results of Undersampling

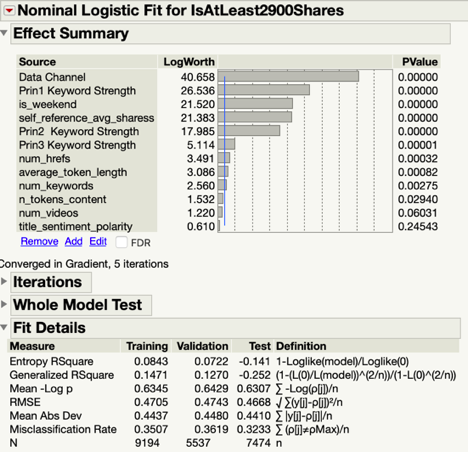
****

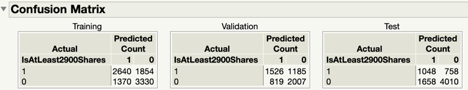


**Figure 11**.The Results of Stepwise

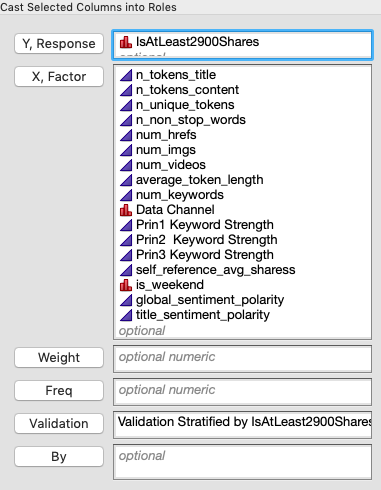


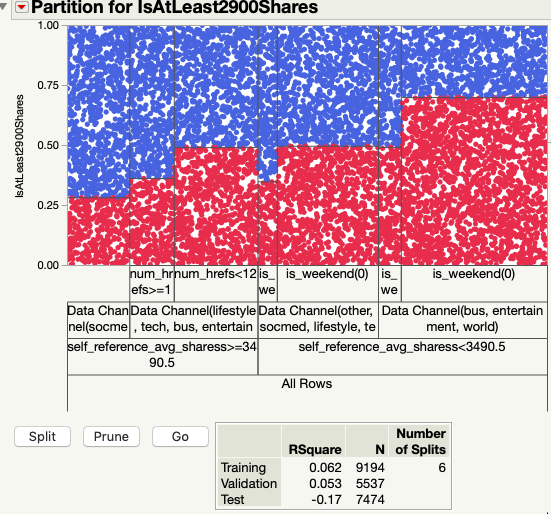
**Figure 12**. The Results of Logistic Regression



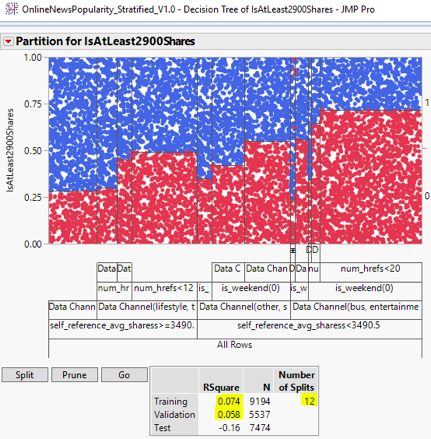


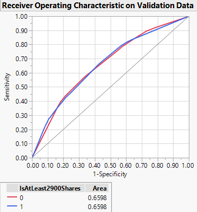
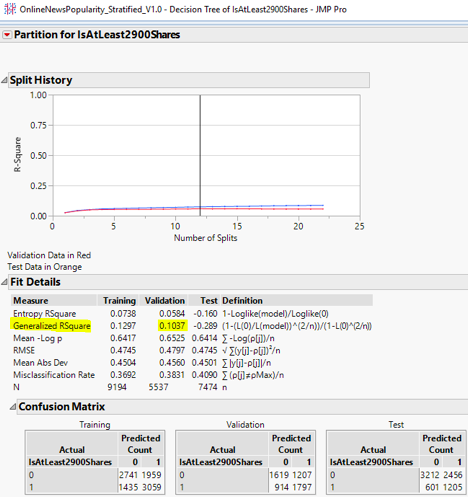
**Figure 13**. The Results of Decision Tree



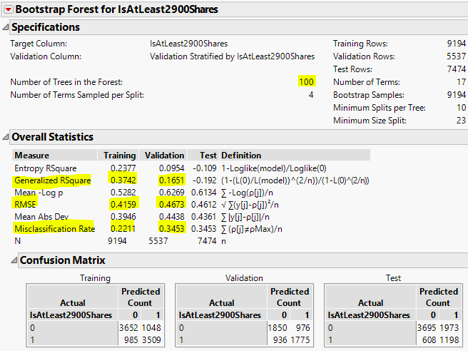


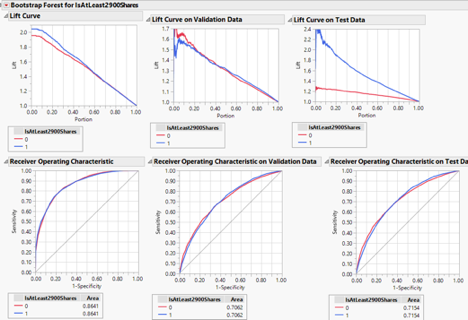
**Figure 13**. The Results of Decision Tree (continued)



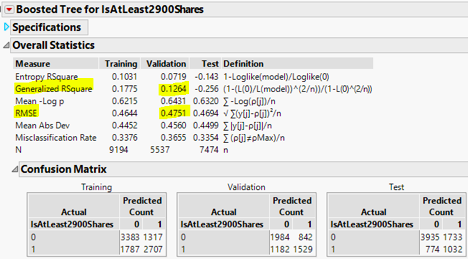


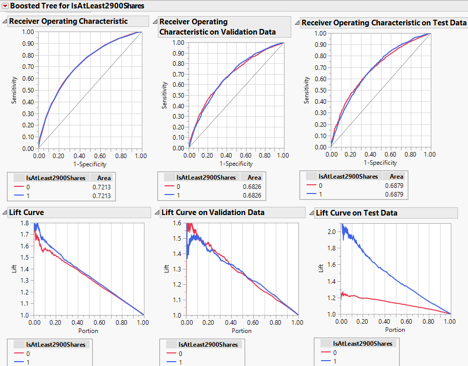
**Figure 14.** The Results of Bootstrap Forest



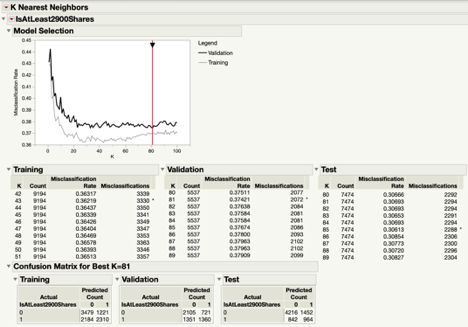


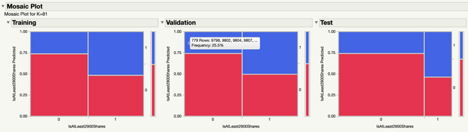
**Figure 15.** The Results of Boosted Tree



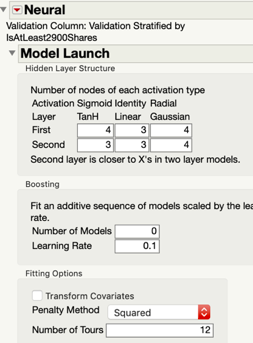


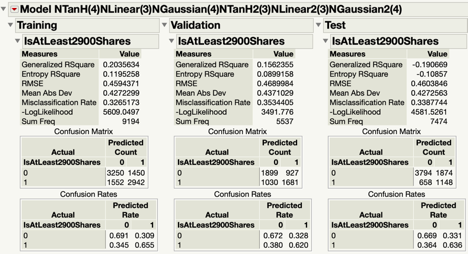
**Figure 16.** The Results of K-Nearest Neighbor



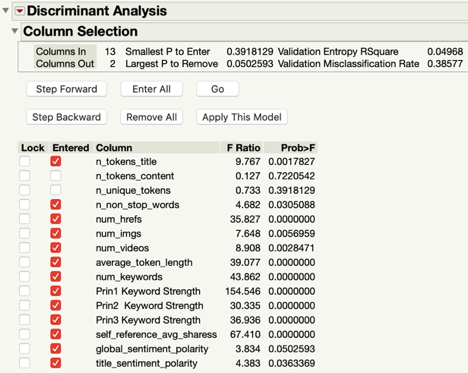


**Figure 17**. The Results of Neural Network





**Figure 18.** The Results of Linear Discriminant Analysis

****

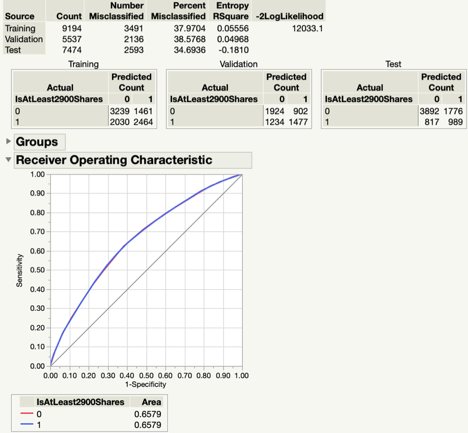
****

****

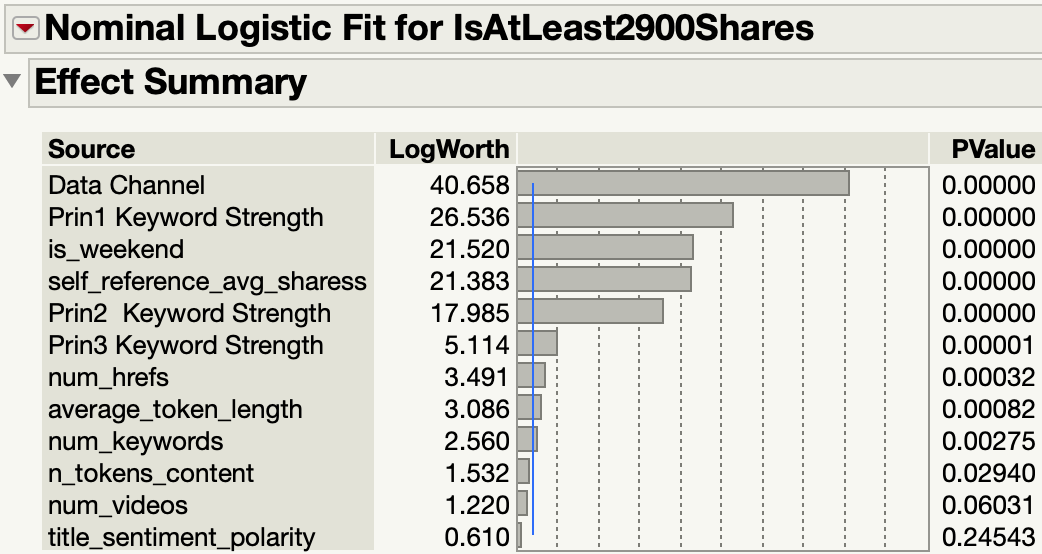
****

****

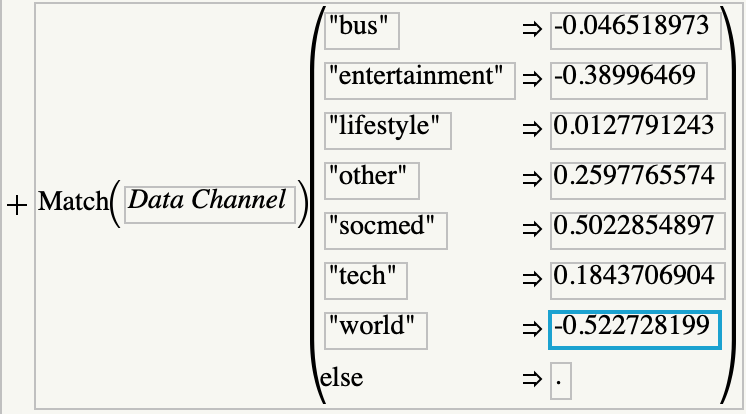
**Figure 18.** The Results of Linear Discriminant Analysis (continued)

****

**Figure 19**. Best Model Interpretation



Statistical Significance



Coefficients for specific Data Channels