Hillary Mahoney
MSDS 454 – Sec 55
May 2019

Introduction

"When you press "send" on an e-mail, you can't just assume it will reach an inbox. One out of four American business e-mails was marked as e-mail spam or went missing in 2015," (Yesware Blog, 2018). Webmail providers are increasingly using recipient engagement to classify an e-mail as spam or not. The Hewlett-Packard Internal-only Technical Report contains a collection of spam e-mails that came from their postmaster and individuals who had filed spam. The collection of non-spam e-mails came from filed work and personal e-mails, therefore, the word 'george' and the area code '650' are indicators of non-spam. These are useful when constructing a personalized spam filter.

The goal of this analysis is to determine whether a given e-mail is spam or not based on the frequency of certain words or characters occurring within the e-mail. False positives (marking good mail as spam) are undesirable, but if we insist on zero false positives in the training/testing set, then 20-25% of the spam passed through the filter. To accurately predict whether an e-mail is spam or not, it is worth understanding the trade-offs that are made when dealing with precision (accuracy of the positive prediction) and recall (positive instances that are correctly detected) rates. It is impossible to have both a high precision and high recall. If precision is increased, the correct spam classifier will be better predicted, but that would allow more actual spam to pass through the filter (lower recall). Therefore, we need to determine if it's better to have high precision or high recall for labeling and filtering potential spam e-mails.

Data Quality Check

A preliminary check of the data revealed that there were 4601 observations, 58 initial variables and zero missing values. Also, there were three additional columns in the dataset used to split the data into training and test sets. Table 1.1 displays the attribute name, data type, and a brief description of the data. Most of the variables represent the frequency of a given word occurring within an e-mail, expressed as a percentage. A word in this context is any string of alphanumeric characters bounded by non-alphanumeric characters or end-of-string. The run-length attributes represent the length of sequences of consecutive capital letters and the character frequency attributes represent the frequency of a specific character occurring within an e-mail. Lastly, the spam attribute is a binary variable denoting whether the e-mail was considered spam (1) or not (0), i.e. an unsolicited commercial e-mail. In the full data set, about 60% of the observations are labeled as non-spam and 40% are labeled as spam e-mails.

TABLE 1.1: DATA ATTRIBUTES

ATTRIBUTE	DATA TYPE	DESCRIPTION/CALCULATION
word_freq_*WORD*	Continuous	Percentage of words in the e-mail that match *WORD* =100*(number of times *WORD* appears in e-mail) / total number of words in e-mail
char_freq_*CHAR*	Continuous	Percentage of characters in the e-mail that match *CHAR* =100*(number of *CHAR* occurrences) / total characters in e-mail
capital_run_length_average	Continuous	Average length of uninterrupted sequences of capital letters
capital_run_length_longest	Continuous	Length of longest uninterrupted sequence of capital letters
capital_run_length_total	Continuous	Total number of capital letter in the e-mail
spam	Binary Factor	Denotes whether the e-mail was considered spam or not

Because the word frequency and character frequency attributes are percentages, their values should not be less than zero or greater than 100. A quick check of the minimum and maximum values for each of these variables validated that the ranges were as expected with no values exceeding 100% and no values less than 0%. However, it was noticed that most of the word and character frequency attributes had high frequencies of 0% (i.e. high counts of zero occurrences within an e-mail); this will be addressed in greater detail in the EDA section.

The distributions for the remaining variables were examined and were also highly right skewed, as displayed in Figure 1.1. Additionally, Table 1.2 shows the summary statistics for the capital_run_length attributes and since the minimum values for each of these variables is greater than zero, a natural log or log transformation could be successfully applied to help with normality. The topic of transformations will be explored further in the EDA section.

FIGURE 1.1: DISTRIBUTIONS OF CAPITAL_RUN_LENGTH ATTRIBUTES

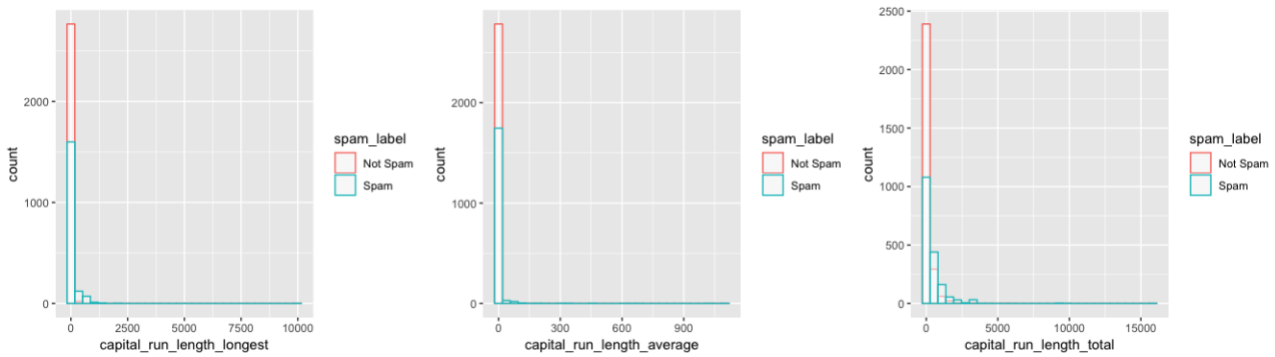


TABLE 1.2: SUMMARY STATISTICS OF CAPITAL_RUN_LENGTH ATTRIBUTES

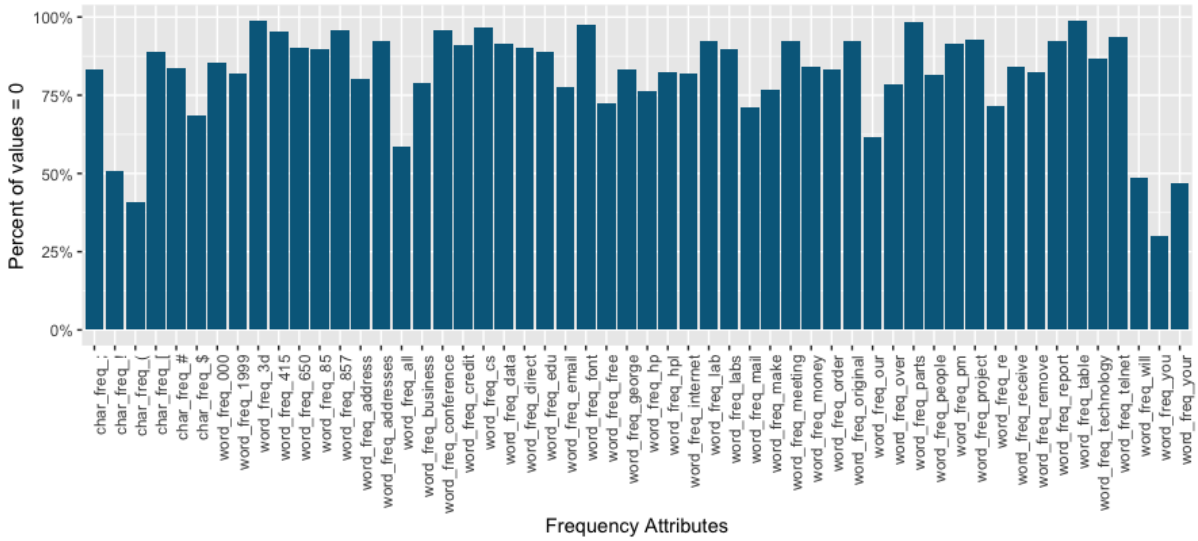
Statistic	Mean	St. Dev.	Min	1st QRT	3rd QRT	Max
capital_run_length_average	5.192	31.729	1.000	1.588	3.706	1,102.50
capital_run_length_longest	52.173	194.891	1.000	6.000	43.000	9,989.00
capital_run_length_total	283.289	606.348	1.000	35.000	266.000	15,841.0

Exploratory Data Analysis

When examining the summary statistics and histograms for each attribute, one of the most notable observations was that for a majority of the attributes at least 75% of the observations were zero inflated. In other words, for most of the word_frequency and char_frequency variables, less than 25% of the observations had a value greater than zero. The distributions for these variables were extremely right-skewed with varying lengths of tails.

Figure 2.1 shows each of the word_frequency and char_frequency attributes and the percent of values that were equal to zero. Only 11 out of the 54 attributes had less than 75% of observations equal to zero. Some of the attributes, such as word_freq_3d or word_freq_parts, had extremely high percentages of zero counts where almost 98% of the total observations were equal to zero.

FIGURE 2.1: PERCENT OF ZERO COUNTS PER FREQUENCY ATTRIBUTE

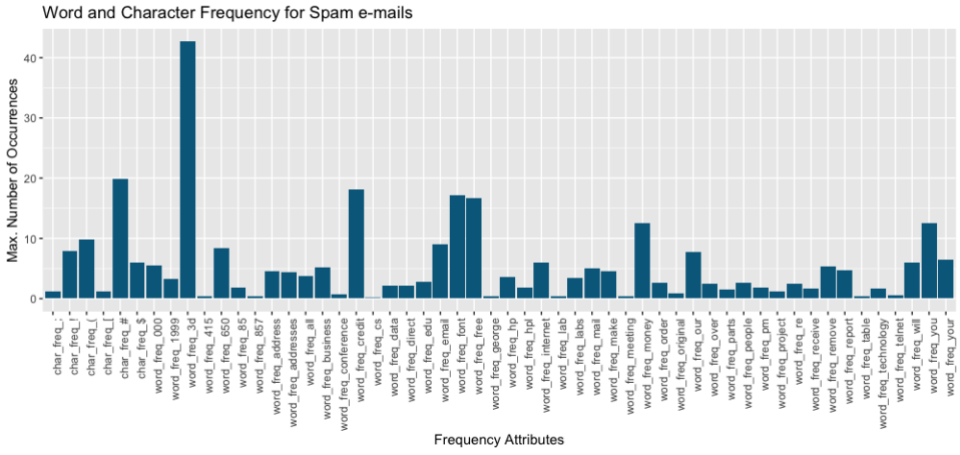


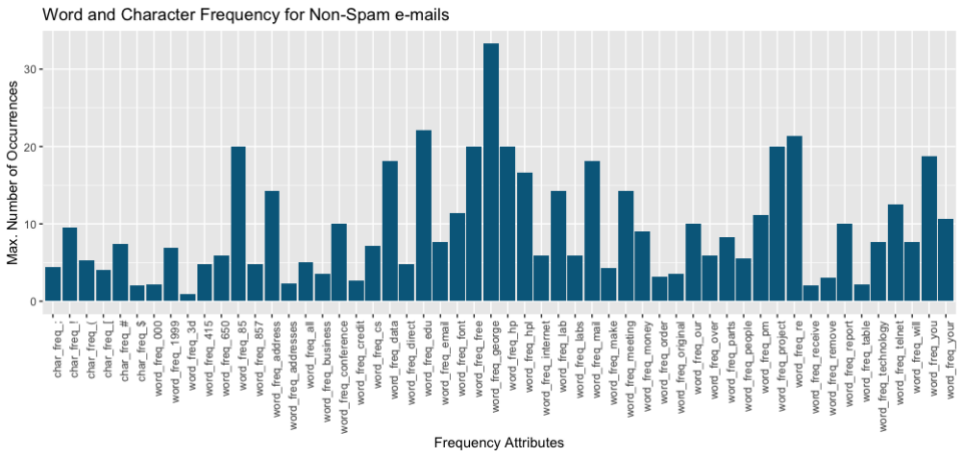
Zeros can arise for several different reasons each of which may have to be treated differently. Depending on the modeling problem, a transformation could be necessary to assist with statistical analysis, but the models explored in this analysis are more flexible and do not rely on normality assumptions. Additionally, given that the

word and character frequency attributes are counts represented as a percentage of a total and knowing they have zero-inflated distributions that are heavily right skewed, a reasonable transformation may not be possible. Some of the typical transformations that are normally considered, such as natural log or square root, are not applicable with zero-inflated data sets as any monotonic, non-deterministic transformation will transform the zeros to a new minimum or maximum with a similar distribution. Therefore, no transformations were made to the data.

Because we are trying to classify an e-mail as spam or not based on the occurrence of specific words or characters, it would be interesting to know how often certain words or characters occur for spam e-mails compared to non-spam e-mails. Figure 2.2 displays the maximum percent frequency of occurrences for each of the words and characters split by spam versus non-spam.

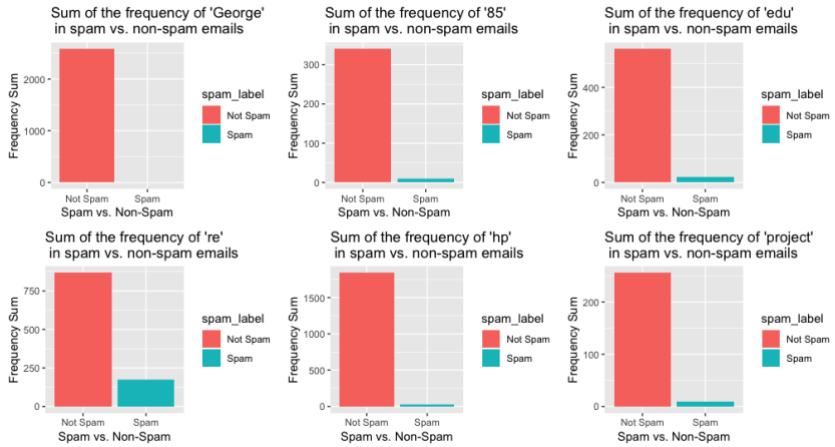
FIGURE 2.2: SPAM VS. NON-SPAM MAXIMUM OCCURANCE PER FREQUENCY ATTRIBUTE





There is a clear demarcation between the maximum frequency specific words or characters appear in spam e-mails versus non-spam e-mails. For spam e-mails, the word '3d' appears 40 times (at a maximum, on average) out of every 100 words. Similarly, the '#' character has a higher occurrence frequency for spam e-mails compared to non-spam e-mails. Figures 2.3, 2.4 and 2.5 show examples of attributes that are more likely to occur in spam versus non-spam e-mails and vice-versa. Figure 2.5 focuses specifically on the frequency of characters in spam versus non-spam e-mails and it's interesting that half of the characters are more likely to show up in non-spam e-mails and the other half of the characters appear in spam e-mails more frequently.

FIGURE 2.3: WORDS THAT HAVE HIGHER FREQUENCIES IN NON-SPAM E-MAILS



For non-spam e-mails it appears that words such as ‘George’ or ‘project’ occur more frequently. Similarly, characters like ‘[‘ and ‘(‘ have higher frequencies in non-spam e-mails compared to spam e-mails. Therefore, as the occurrence of these specific words or characters increases within an e-mail, the less likely it is spam. Alternatively, the more often words like ‘3d’,’free’, or ‘money’ appears in an e-mail, the more likely it is spam. Also, longer sequences of capital letters have higher frequencies in spam e-mails compared to non-spam e-mails. All three of the capital_run_length attributes had higher frequencies in spam e-mails.

FIGURE 2.4: WORDS THAT HAVE HIGHER FREQUENCIES IN SPAM E-MAILS

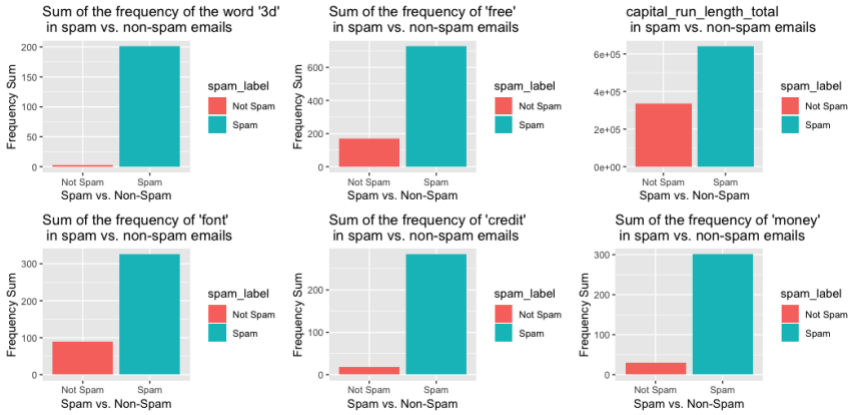
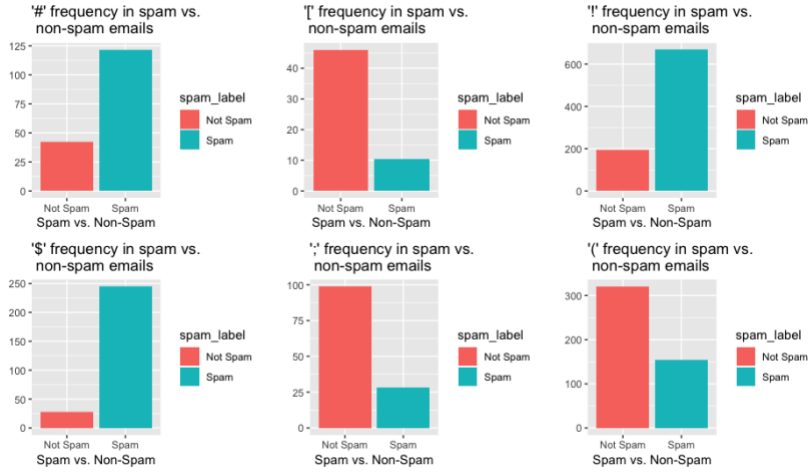


FIGURE 2.5: CHARACTER FREQUENCIES IN SPAM VS. NON-SPAM E-MAILS



Now that we know the more often certain words or characters occur indicates whether an e-mail is spam or not, we want to examine any potential relationships

between dependent variables to determine if the occurrence of a specific word or character influences the occurrence of a different word or character. If these relationships do exist, do they impact the likelihood of an e-mail being classified as spam or not spam? To aid with the exploratory analysis section, a classification tree model was created to identify and provide insight into potentially influential variables. One of the many qualities of decision trees is that they require very little data preparation, particularly, they don't require feature scaling or centering which make them great tools for initial exploration of the data.

FIGURE 2.6: CLASSIFICATION TREE PLOT

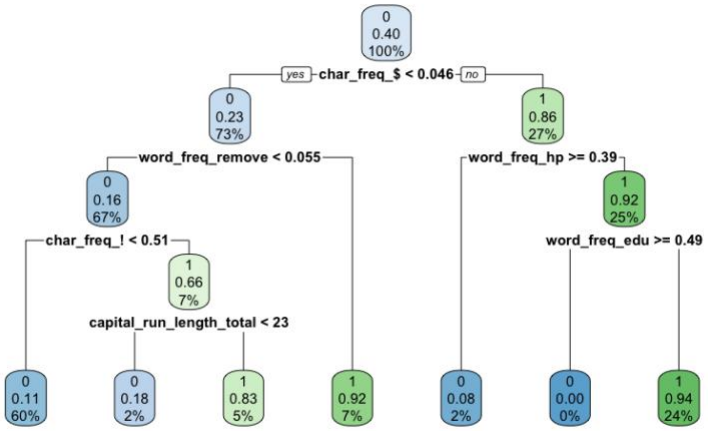


Figure 2.6 contains the tree plot of the classification model; from this we can see that the char_frequency attributes and the capital_run_length_total attribute appear to be important in determining the e-mail as spam which we also saw in Figures 2.4 and 2.5. High frequencies of the words 'HP' and 'edu' in e-mails indicates an e-mail is not spam, whereas e-mails containing low frequencies of these words were more likely to be considered spam which is consistent with what we saw in Figures 2.2 and 2.3.

Model Build

1. Logistic Regression Model

Outside of the decision tree constructed during EDA, the first model built was a generalized linear model using backwards variable selection. The model output displayed in Table 3.1 contains the model coefficients, their standard errors, and the associated p-values.

TABLE 3.1: GLM FAMILY=BINOMIAL SUMMARY OUTPUT

Dependent variable: *spam*

	Estimate	Std. Error	Pr(>|z|)		Estimate	Std. Error	Pr(>|z|)
word_freq_address	-0.137	0.085	0.107	**word_freq_telnet**	-3.945	2.719	0.147
word_freq_3d	1.580	1.614	0.328	**word_freq_data**	-0.791	0.408	0.053.
word_freq_our	0.742	0.135	0.000***	**word_freq_415**	-16.485	4.025	0.000***
word_freq_over	0.790	0.289	0.006**	**word_freq_85**	-2.542	0.906	0.005**
word_freq_remove	2.174	0.377	0.000***	**word_freq_technology**	1.105	0.345	0.001**
word_freq_internet	0.502	0.181	0.005**	**word_freq_parts**	-0.556	0.483	0.250
word_freq_order	1.202	0.345	0.000***	**word_freq_pm**	-0.873	0.421	0.038*
word_freq_mail	0.150	0.087	0.084.	**word_freq_cs**	-41.968	29.170	0.150
word_freq_receive	-0.802	0.319	0.012*	**word_freq_meeting**	-3.742	1.473	0.011*
word_freq_will	-0.167	0.093	0.073.	**word_freq_project**	-1.638	0.598	0.006**
word_freq_free	0.943	0.163	0.000***	**word_freq_re**	-0.659	0.148	0.000***
word_freq_business	1.057	0.261	0.000***	**word_freq_edu**	-1.330	0.287	0.000***
word_freq_you	0.082	0.042	0.048*	**word_freq_table**	-3.259	2.523	0.196
word_freq_credit	1.009	0.594	0.089.	**word_freq_conference**	-3.290	1.850	0.075.
word_freq_your	0.273	0.062	0.000***	**`char_freq_;`**	-0.815	0.327	0.013*
word_freq_000	2.274	0.550	0.000***	**`char_freq_[`**	-2.128	1.765	0.228
word_freq_money	0.729	0.271	0.007**	**`char_freq_!`**	1.016	0.185	0.000***
word_freq_hp	-1.788	0.327	0.000***	**`char_freq_$`**	5.209	0.768	0.000***
word_freq_hpl	-1.284	0.531	0.016*	**`char_freq_#`**	2.624	0.849	0.002**
word_freq_george	-16.415	2.691	0.000***	**capital_run_length_longest**	0.014	0.003	0.000***
word_freq_650	0.413	0.239	0.084.	**capital_run_length_total**	0.001	0.000	0.003**
word_freq_lab	-1.933	1.495	0.196				
(Constant)	-1.807	0.159	0.000***				
Observations				3439			
Log Likelihood				-650.1525			
Akaike Inf. Crit.				1388.3			

Note: *pp***p<0.01**

Most of the predictors are statistically significant at the alpha = 0.05 level with the exception of word_freq_3d, word_freq_address, word_freq_lab, word_freq_telnet, word_freq_data, word_freq_parts, word_freq_cs, word_freq_table, and char_freq_[. The logistic regression coefficients give the change in the log-odds of the outcome for a one unit increase in the predictor variable. For every one-unit change in the frequency of the word 'our', the log odds of spam (versus non-spam) increases by 0.742. Or, if we convert the coefficients to odds-ratios, the odds of an e-mail being classified as spam increases by a factor of 2.1 ($exp^{0.742}$) when the frequency of the word 'our' increases by one unit.

TABLE 3.2: GLM TEST SET CONFUSION MATRIX

	Not Spam (0)	Spam (1)
Not Spam (0)	686 (TN)	28 (FP)
Spam (1)	60 (FN)	388 (TP)

Table 3.2 displays the confusion matrix for the logistic regression model. The rows represent an actual target whereas the columns represent the predicted classes. The model accuracy is calculated by summing the true positives (denoted TP) plus the true negatives (TN) over the total number of observations. The out-of-sample accuracy is 92.42%. Other measures that can be considered are precision and recall. Precision is the accuracy of the positive prediction and recall is the ratio of positive instances that are correctly detected by the classifier. There is a concave relationship between precision and recall so it is impossible to have both high precision and high recall, however, since we are interested in avoiding false positives for this problem we can focus on precision. Therefore, the precision for this model is 91.63%.

2. Decision Tree Model

In the EDA section, a simple classification tree using all available predictors was constructed and we found that the frequency of characters such as '$' and '!' and the frequency of words like 'edu' and 'HP' impact the classification of an e-mail as spam or not spam. Without any pruning or tuning of parameters, this tree had a 90% accuracy rate on the test set, a precision rate of 96%, and a recall rate of 89%. Precision is a good measure to consider when the cost of false positives is high. For email spam detection, a false positive means that an email that is non-spam (actual negative) has been identified as spam (predicted spam) so the email user might lose important emails if the precision is not high for the spam detection model.

TABLE 3.3: CLASSIFICATION TREE TEST SET CONFUSION MATRIX

	Not Spam (0)	Spam (1)
Not Spam (0)	688 (TN)	26 (FP)
Spam (1)	85 (FN)	363 (TP)

The main drawback of decision trees is that they are prone to overfitting, because if grown deep, they are able to fit all kinds of variations in the data, including noise. However, it is possible to partially address this by pruning, so the classification tree from the EDA section was pruned by setting three as the minimum number of observations in the node before the algorithm performs a split, the minimum number of observations in the final leaf was set to 15, and the maximum depth of any node was set to eight.

FIGURE 3.1: PRUNED CLASSIFICATION TREE PLOT

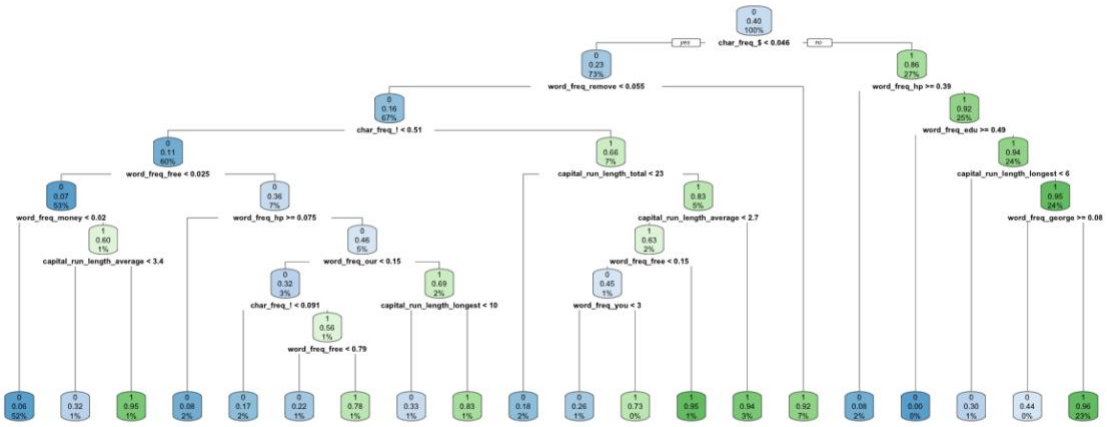


TABLE 3.4: PRUNED CLASSIFICATION TREE TEST SET CONFUSION MATRIX

	Not Spam (0)	Spam (1)
Not Spam (0)	685 (TN)	29 (FP)
Spam (1)	70 (FN)	378 (TP)

As we can see from Figure 3.4, the pruned tree is substantially larger and more complex, but still relatively easy to interpret. Additionally, pruning proved to be somewhat successful as the accuracy rate increased to 91.48% at the cost of a small decrease in precision from 96.3% to 95.9%.

3. Support Vector Machine

Support Vector Machines (SVMs) seek an optimal hyperplane for separating two classes in a multidimensional space. The hyperplane maximizes the margin between the two classes' closest points. SVMs are flexible in that they can construct classification boundaries that are linear, polynomial, radial and sigmoid in shape. Two SVM models were constructed, one using a linear boundary and the other using a radial boundary.

The first model with a linear boundary uses repeated cross validation with 10 re-sampling iterations and is repeated three times. Additionally, the model is tuned using a manual grid search and a tune length of 10, meaning the algorithm tries 10 different

default values for the main parameter. Table 3.5 displays the confusion matrix for this model; the out-of-sample accuracy rate is 93.03% and the precision rate is 94.96%.

TABLE 3.5: LINEAR SUPPORT VECTOR MACHINE CONFUSION MATRIX

	Not Spam (0)	Spam (1)
Not Spam (0)	678 (TN)	45 (FP)
Spam (1)	36 (FN)	403 (TP)

The second model uses a radial boundary with a similar manual grid search and a tune length of 10. Table 3.6 displays the confusion matrix for this model; the out-of-sample accuracy rate for the radial model is slightly better than the linear approach at 93.89%. Additionally, the precision rate is 97.33% which is also higher than the precision rate of the linear approach.

TABLE 3.6: RADIAL SUPPORT VECTOR MACHINE CONFUSION MATRIX

	Not Spam (0)	Spam (1)
Not Spam (0)	695 (TN)	52 (FP)
Spam (1)	19 (FN)	396 (TP)

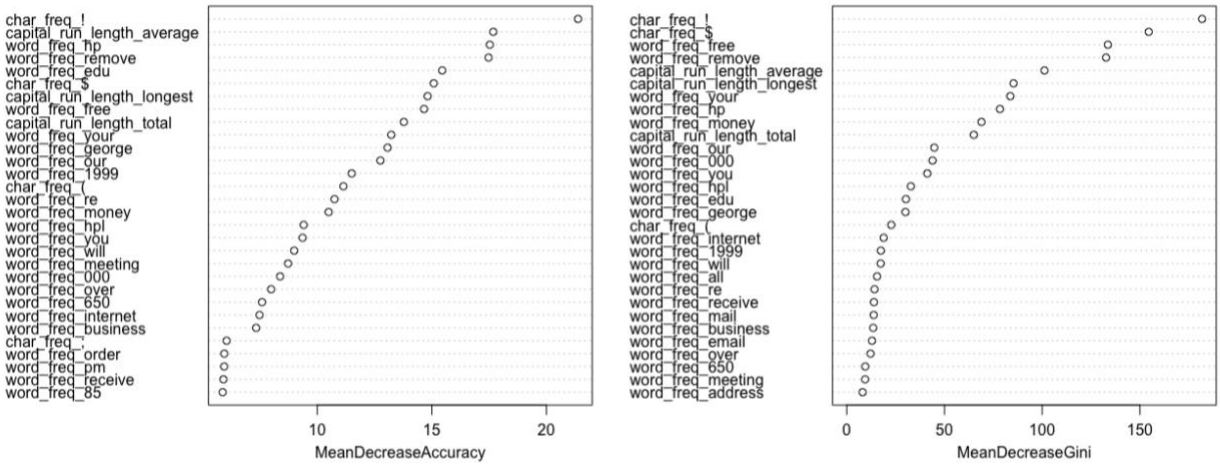
4. Random Forest

Random forests are built on the same fundamental principles as decision trees, but is a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of overcoming over-fitting problem of individual decision tree. Figure 3.2 displays the variable importance plot for the top 20 most important variables, that is, the variables with the most predictive power. The accuracy rate for the random forest model is 95.49% and the precision rate is 96.7%.

TABLE 3.7: RANDOM FOREST CONFUSION MATRIX

	Not Spam (0)	Spam (1)
Not Spam (0)	700 (TN)	14 (FP)
Spam (1)	36 (FN)	412 (TP)

FIGURE 3.2: VARIABLE IMPORTANCE PLOT (TOP 20 VARIABLES)



Model Comparison

The models considered for this problem were flexible in that they allowed the response variables to have a distribution form other than normal which is ideal considering the zero-inflated distributions observed during the exploratory analysis phase. Surprisingly all of the models performed well both in-sample and out-of-sample; in the past decision trees have proved to be less accurate compared to other techniques and we saw the same thing here. However, the final classification tree did have a notable out-of-sample precision rate. Accuracy is highly important, but if we are trying to avoid false positives, then overall the best model would be the support vector machine using the radial boundary and a manual grid search.

TABLE 4.1: FINAL MODEL COMPARISON

Model	In-Sample Accuracy	Out-of-Sample Accuracy	Out-of-Sample Precision
GLM	93.3%	92.4%	91.6%
Classification Tree	92.4%	91.5%	95.9%
Support Vector Machine	94.3%	93.9%`	97.3%
Random Forest	97.1%	95.5%	96.7%

REFERENCES

Musumano, Elise. “How To Avoid Email Spam Filters: The Complete Guide.” *Yesware Blog*, 14 Nov. 2018, www.yesware.com/blog/email-spam/.