Anomaly Detection Challenge 4

Vishal Bhalla Matriculation No: 03662226
Nithish Raghunandanan Matriculation No: 03667351

Spam Detection

January 10, 2016

1 Abstract

The goal of this challenge was to implement a machine learning algorithm (Binary Classification) to classify email messages into SPAM or HAM. The email messages were in the format of an .eml file as defined in RFC822, and information on recent standard of email, i.e., MIME (Multipurpose Internet Mail Extensions) can be found in RFC2045-2049. The Training set consisted of the email message along with a label identifying whether it was a SPAM/HAM message.

2 Data Preprocessing & Analysis

2.1 Structure of Data

We analyzed the distribution of different fields and their corresponding range of values.

2.1.1 Fields of Interest

Out of the many fields in each email, we found the Sender information, the date of delivery of the mail, the message subject and body to be interesting. However, processing each of these fields was cumbersome due to the different formats in which the email was present. In particular, the email body was either of Plain Text or MIME type and contained Multiple sections in it.

3 Feature Engineering

There were many qualitative features in the data set which we had to map to numerical features.

3.1 Message Length

We used the length of features.

3.2 Spam Measure of an email

1. Spammicity of message
   The feature vectors for a message subject and body were encoded based on the Spam Measure of its constituent words. For each word, we constructed a dictionary from the training set t email in the anti virus signatures normalized by the occurrence of the signature. Thus
   $$\textbf{Spammicity(word)} = \frac{Number\ of\ occurences\ of\ that\ word\ in\ a\ SPAM\ email}{Total\ \ Occurrences\ of\ that\ word\ in\ both\ SPAM\ \&\ HAM\ emails}$$

   Finally, to calculate the overall Spammicity of a message we multiply the individual word spammicities. Therefore,
   $$\textbf{Message Spammicity} = \prod_{w \in Message} Spammicity(w)$$

The blue, green and red lines in the figure **??** indicate the accuracy of Random Forest, Multi-class SVM and KNN Classifiers based on the number of Anti Viruses and their signatures used as features. As depicted in the graph, increasing the number of Anti Viruses as features definitely improved the performance. Initially, increasing the use of signature as features inside each Anti Virus did improve the classification accuracy but as we increased the number of Anti Viruses, this was irrelevant.

Hence, we decided to use just one feature per Anti Virus and that was the Anti Virus's frequency encoding. Moreover, we minimized the training samples with Null Feature Vectors after the encoding.

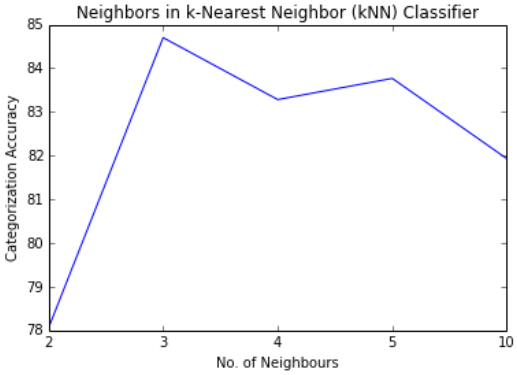
4 Model

We tried fitting three different models to our data viz. Random Forest, Multi-class SVM and KNN Classifiers. Some of the observations and conclusions during Model Selection over the cross-validation dataset were as below:

1. Plotting the optimal number of neighbors (fig. 1a) for kNN classifier, we found k=21 to give optimum results.

2. Tuning the criterion as Gini or Entropy and the number of trees for Random Forest Classifier. Fig. 2a shows that a Random Forest with Gini and 200 trees to give the best results over the cross-validation dataset.

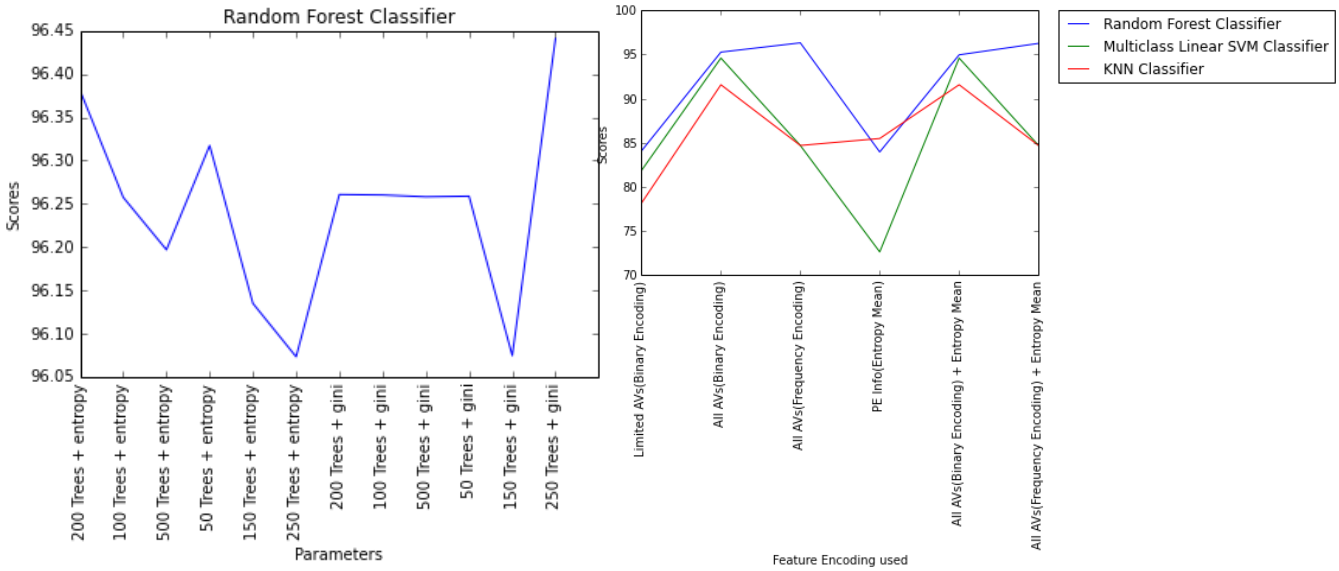
3. Vowpal Wabbit



(a) Optimal no. of neighbors for kNN classifier

Figure 1: Tuning Parameters

4. Splitting the training samples in the ratio of 4:1 and using Stratified K-Fold Cross-Validation as a model evaluation metric fig. 2b, we infer Vowpal Wabbit to be the best classifier.



(a) Random Forest - Tuning trees & criterion

(b) Model Comparison

Figure 2: Random Forest as best model after tuning parameters across all models

5. Static Rules

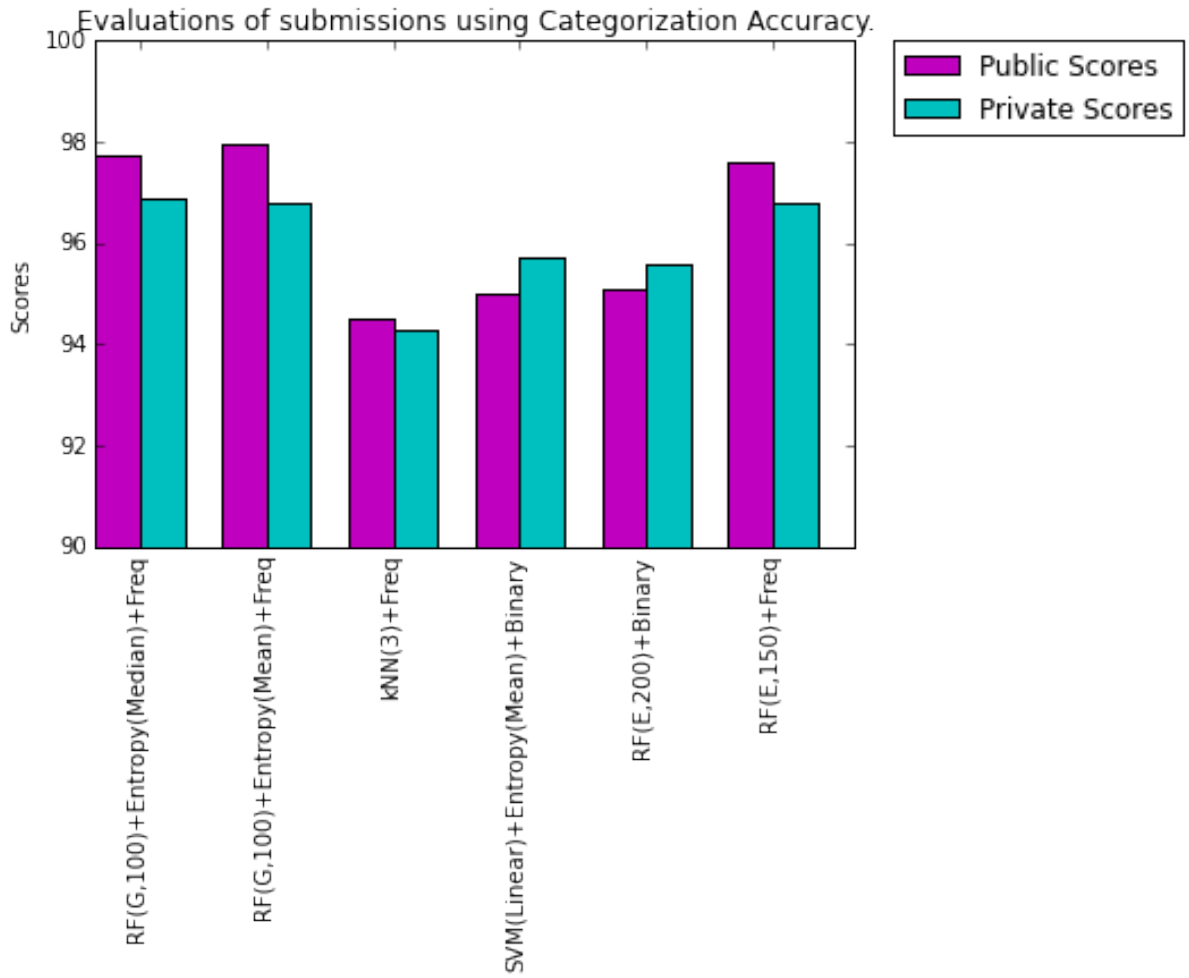


Figure 3: Kaggle Evaluation Results on different models : Public vs Private Scores

5 Results

The performance is evaluated by computing the Categorization Accuracy i.e. the percentage of correct predictions. The final evaluation results from Kaggle using different models on the Malware Dataset are depicted in the graph fig. 3

6 Conclusion

By trying out different models on our training set, after removing missing training samples, best results were observed for Random Forest (100 trees with Gini as a criterion) on Frequency Encoded all Anti Viruses' VT Info data with a Public classification accuracy of 97.971% (96.778% Private Score). Using frequency encoding of Anti Viruses signatures from VT Info and mean/median values of Entropy from PE Info as features gives near about

similar result (in fact the same result for Private scores) as pure Frequency based VT Info encoding on all Anti Viruses with a classification accuracy of 97.613% (96.778% Private Score). As most of the data was categorical in nature and these categorical features had no relation with other fields / features, we deduce that K Nearest Neighbors classifier and Multi Class SVM did not give better results as compared to Random Forests.