Anomaly Detection Challenge 3

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

Malware Classification

December 14, 2015

1 Abstract

The goal of this challenge was to implement a machine learning algorithm (Multi-class Classification) based on static analysis data (peinfo.tar.gz) and antivirus signatures (vt.tar.gz) to classify malware into predefined families (0...9) or determine that a sample does not belong to any family (10). The samples are identified by the md5 signature.

2 Data Preprocessing & Analysis

2.1 Structure of Data

2.1.1 VT Info

Single signature is present per Anti Virus per sample. However, each sample may vary in the presence of an Anti Virus and its signature. We analyzed the variance of different anti viruses and their corresponding range of signatures.

2.1.2 PE Info

Multiple pe_section, pe_resources, pe_import, pe_timestamp, rich_header sections from the static analysis per sample. We analyzed the distribution of different fields and their corresponding range of values.

2.2 Handling Missing Training Samples

There were 429 samples in the training data that did not have the static analysis and anti virus signatures. Due to the large number of samples, we could afford to remove all the samples with missing features from the data set instead of using any replacement technique like replacing by zeros, mean, median, mode, minimum, maximum, etc.

3 Feature Engineering

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

3.1 VT Info

Using a simple One Hot Encoding (OHE) technique would have not worked, given the large number of features (75) and distinct values(average 204) for each of the Anti virus signatures in VT Info. We selected a sub set of Anti Viruses with high variance as features, where Variance $= \frac{TotalNo.ofSamples}{No.ofUniqueSignatures}$.
We compared this with the performance against the one after selecting all Anti Viruses' features.

3.2 PE Info

Most of the sections in PE Info are repeated per sample. Increasing number of features to be used from PE Info did not give better results as compared to pure VT Info Encoding. Hence, we just selected the Numerical Feature of Entropy and used its Mean or Median or Mode as a feature. For the data set, Mean & Median gave the best results.

3.3 Encoding of Features

1. One Hot Encoding (OHE) - Using OHE blows up the features. There would be 75 anti viruses * 203 Signatures as feature on average.

2. Binary Encoding - The feature vectors were encoded based on the presence/absence of Anti virus signature for each sample.

3. Frequency Encoding The feature vectors were encoded based on the variance in the anti virus signatures normalized by the occurrence of the signature. Thus Normalized Variance $= \frac{VarianceofAntivirus}{TotalOccurrencesforthesample}$

The blue, green and red lines in the figure 1 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.

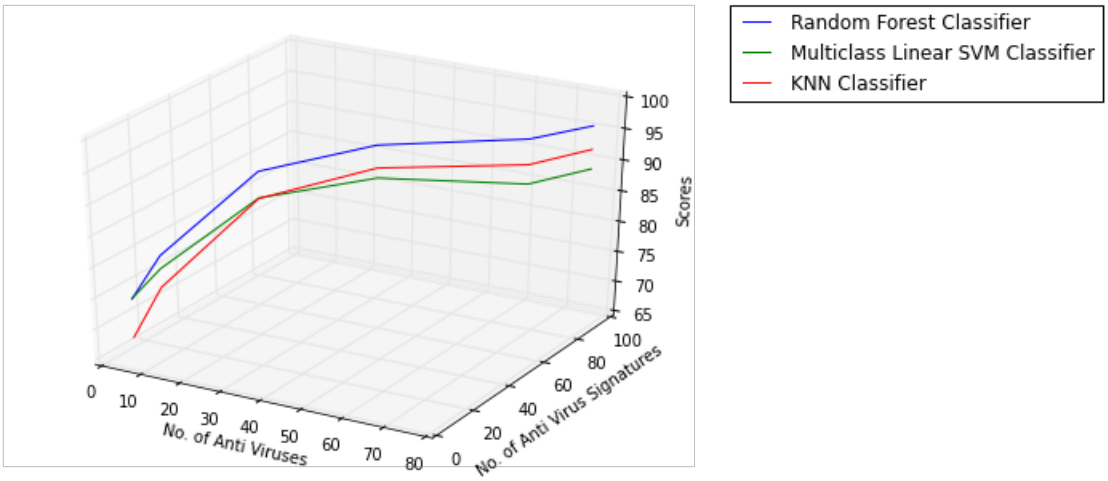
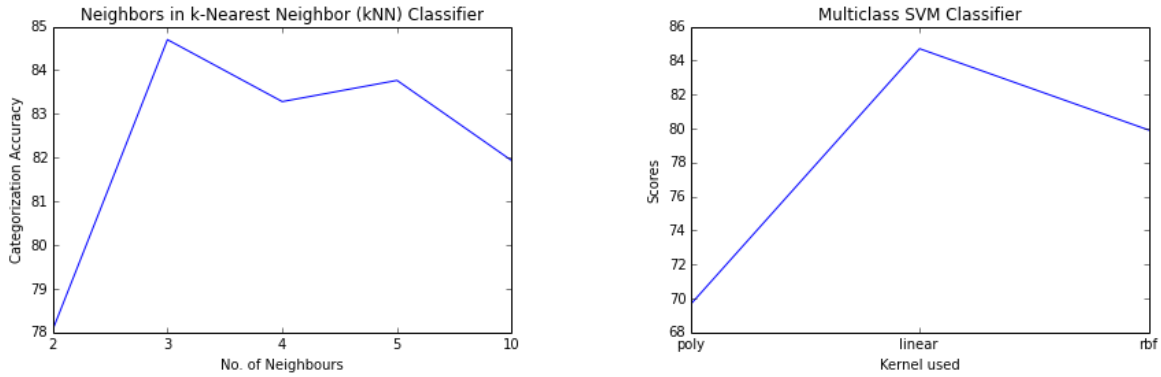


Figure 1: Scores on different models with Signature in Anti Viruses as features.

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. 2a) for kNN classifier, we found k=3 to give optimum results.



(a) Optimal no. of neighbors for kNN classifier

(b) Choosing the best Kernel

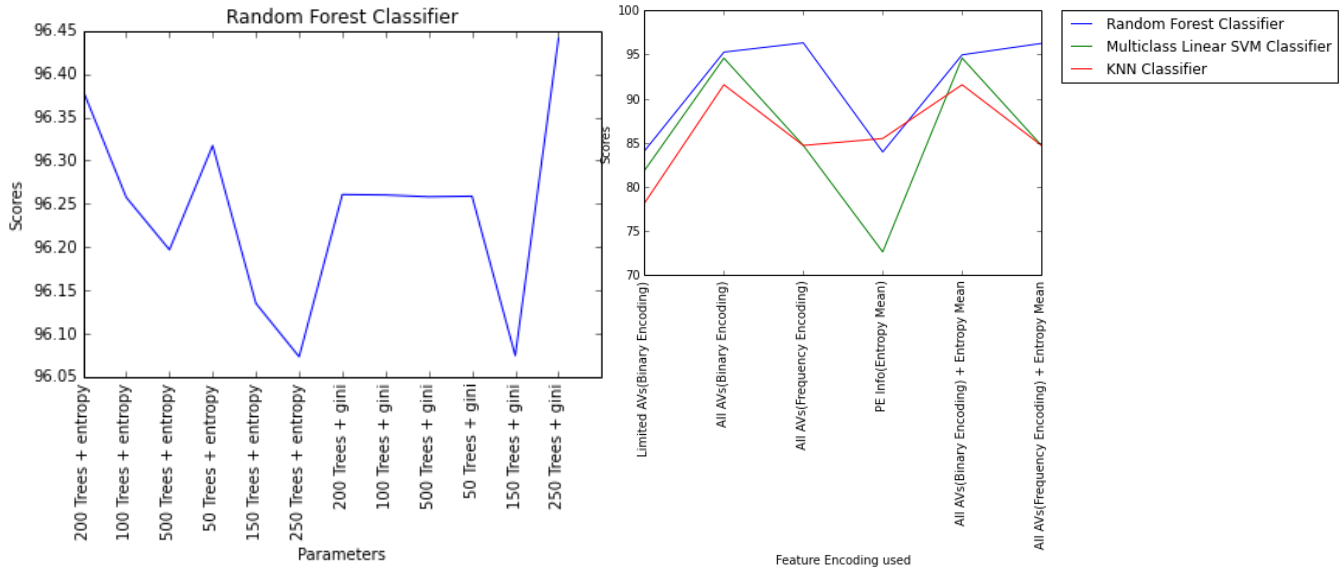
Figure 2: Tuning Parameters

2. Deciding on the type of kernel to be used in Multi Class SVM classifier. We found the linear kernel (fig. 2a) to give the best results over the cross-validation dataset.

3. Tuning the criterion as Gini or Entropy and the number of trees for Random Forest Classifier. Fig. 3a shows that a Random Forest with Gini and 150 trees to give the

best results over the cross-validation dataset.

4. Splitting the training samples in the ratio of 4:1 and using Stratified K-Fold Cross-Validation as a model evaluation metric fig. 3b, we infer Random Forest to be the best classifier.



(a) Random Forest - Tuning trees & criterion (b) Model Comparison

Figure 3: Random Forest as best model after tuning parameters across all models

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. 4

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

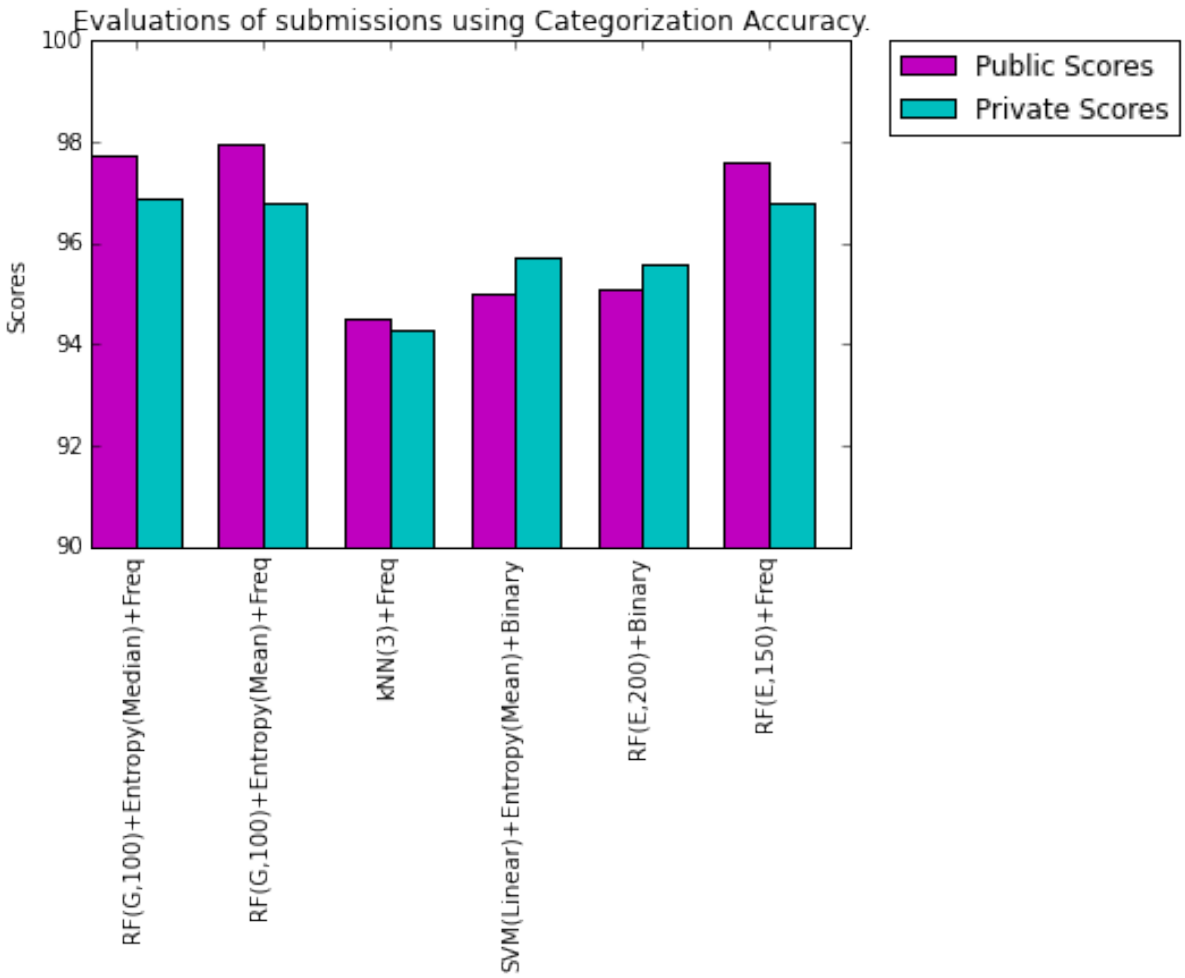


Figure 4: Kaggle Evaluation Results on different models : Public vs Private Scores

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.