

# CSCI 181 / E-181 Spring 2014 Practical 3

Kaggle Team "Capt. Jingleheimer"

David Wihl  
davidwihl@gmail.com

Zack Hendlin  
zgh@mit.edu

March 12, 2014

## Warm-Up

### Baseline

### Warmup Topic 1

### Warmup Summary

## Classifying Malicious Software

### Preliminary Data Analysis

NOTES: 4GB of XML to parse and process, first step was to split the training and the testing. Refactored sample code to separate feature extraction and classification steps.

### Using Cross-validation

Ran 5-10 CV sets of train / CV data 70/30, 80/20 and 90/10 for each classifier. Enabled us to experiment with many permutations of features, classification algorithms, and hyper parameters.

## Approaches considered

### Feature Engineering

Aggregate Features per training file: selected all process features (e.g. 'startreason', 'terminationreason', 'username', 'executionstatus', 'applicationtype') and summary thread features (num of each type of system call).

used CV to generate Logistic Regression weights. Took mean and std of resulting matrix, then eliminated any features where  $\text{abs}(\text{mean}) < 0.001$  and  $\text{std} < 0.01$ .

Further examined  $R^2$  score to eliminate features that were not adding any value.

Created separate model of thread metrics. NOTE: still unknown how to pull in thread metrics to file level.

### **Selection of fitting technique**

Tried LogisticRegression and SVM with a number of different C values, none of which made a significant difference.

Attempted kNN and Theano but ran into technical difficulties.

Had “sanity check” of resulting submission file to check for aberrant distributions. (See plot).

### **Exploratory Data Analysis**

### **Conclusion**