STAT 451 Project: Android Malware Detection

Group 1
Yuval Lerman, John Oh, Jedidiah Schloesser, Brian Slupecki, Kasey White

Introduction

Given a dataset that describes various Android softwares that are potentially malware, the goal of our project was to create a model that detects malware.

Our dataset had 214 permission-based features and 27 API based features, all binary, and then a label of either Malware or Goodware.

Guiding Questions:

1.) What type of model is best suited for detecting malware, and what accuracy can we get with it?

2.) Which features are the more important?

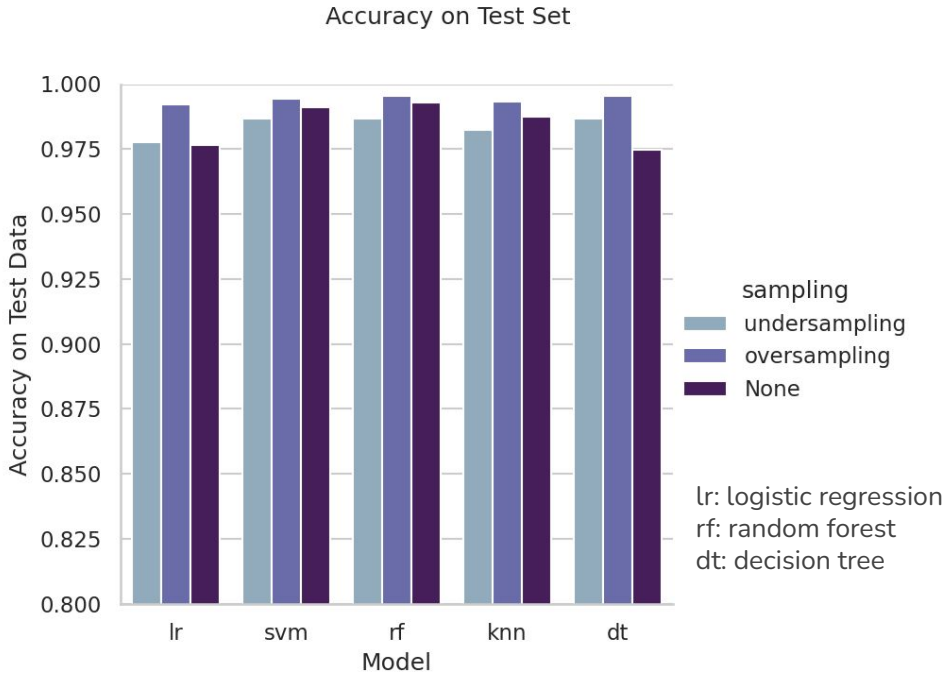
3.) Can we achieve similar accuracy using one-class classification?

Best Model to Detect Malware

Composition of dataset:

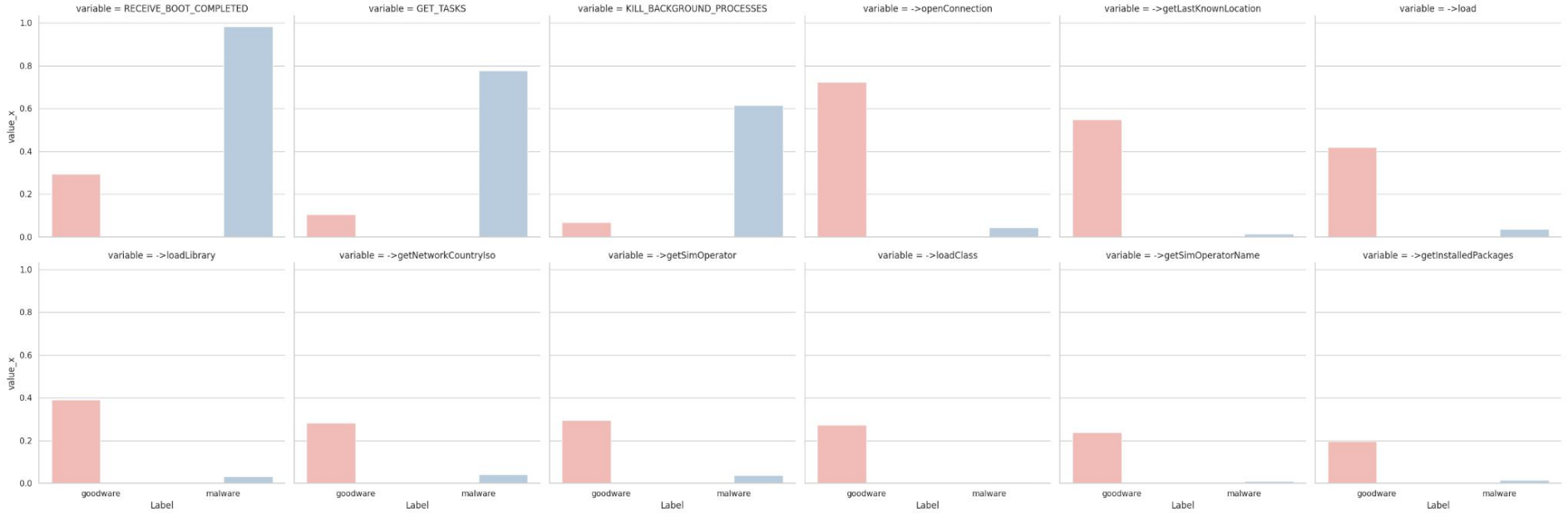
- 80% malware
- 20% goodware.

Baseline (guessing) has 80% accuracy

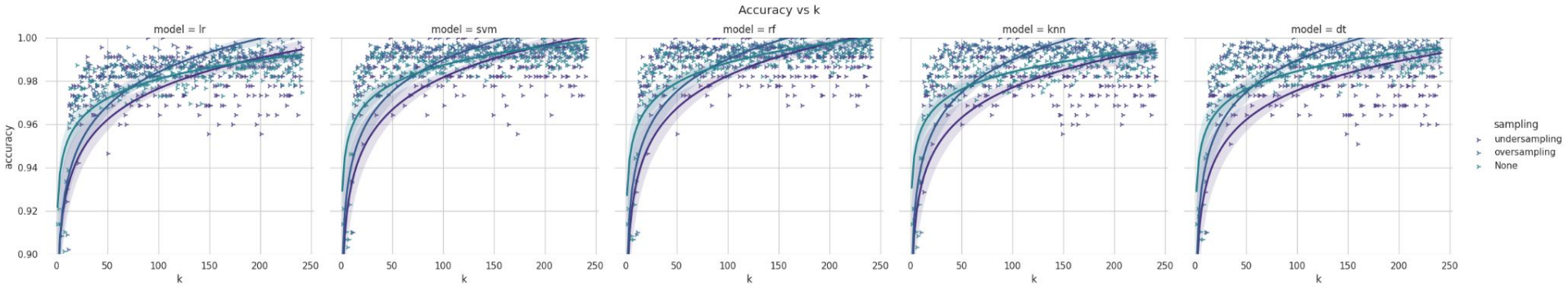


Accuracy from each model is with optimal hyperparameters selected

Important Features



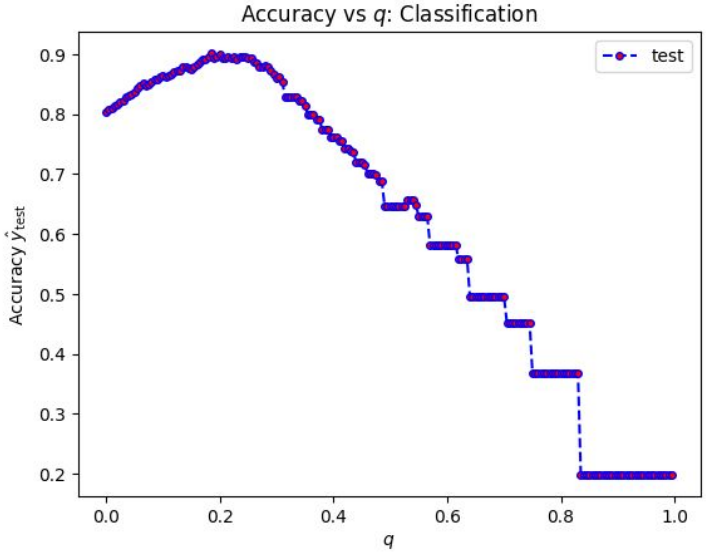
Number of Features



One-class Classification Accuracy Score

Results from Gaussian Mixture Models

- Highest accuracy model:
  - Accuracy just over 90%
    - Guessing benchline is 80% accuracy
  - Over 5% of malware testing examples were classified as goodware
- More practical model:
  - If we choose a classification threshold that yields an accuracy of about 80%, we have just over 2% of malware testing examples classified as goodware



Conclusion

- The best model to detect malware would be a random forest model
- This model was able to achieve 99% prediction accuracy on the test data.
- We were not able to achieve the same results when evaluating the data using a one-class classification model.