Microsoft Malware Prediction

Instructor: Prof. Daniel Acuna

Team Members: Chaitanya Chaphalkar, Eashani Deorukhkar, Jeet Ganatra, Ruoming Wang

1. PROBLEM STATEMENT

In today's Information age, the world relies on advancement in computer technology and the ease it brings to our everyday life. Computers can, however, be attacked digitally using malware which may breach a user or an organization's privacy and leak sensitive data.

Anti-malware security suites usually have a reactive approach where they detect the security breach after malware has already infected the system. In many cases, the damage is irreversible, and a complete system reset is needed.

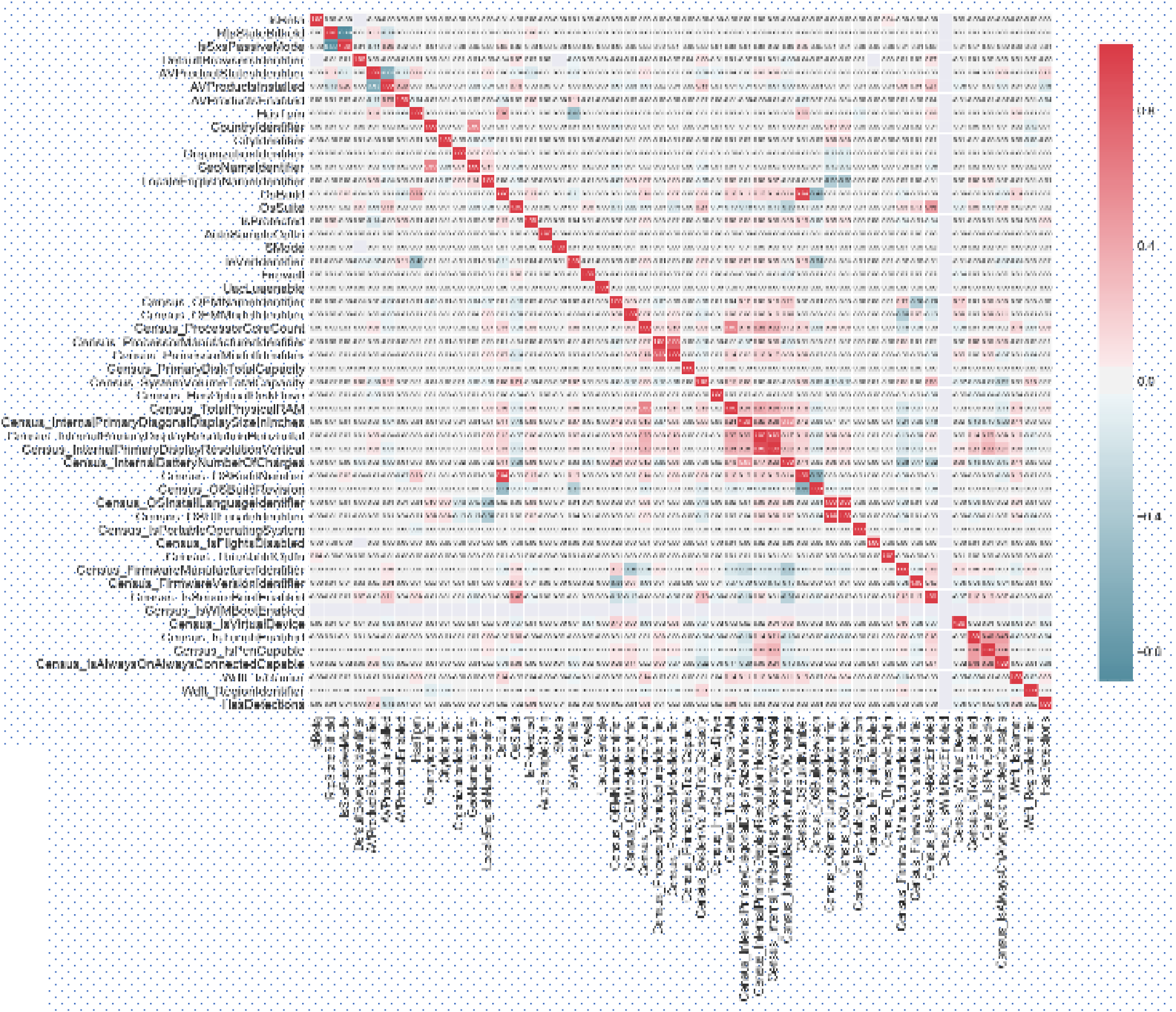
While there is no solution that can make a system completely invulnerable to malware attacks, we can, however, analyze how vulnerable a system currently is and what can be done to make it more resistant. We aim to analyze user information (system specific and excluding sensitive information) provided by Microsoft which lists hardware and software configurations of the users against the Windows Defender's malware detection tag.

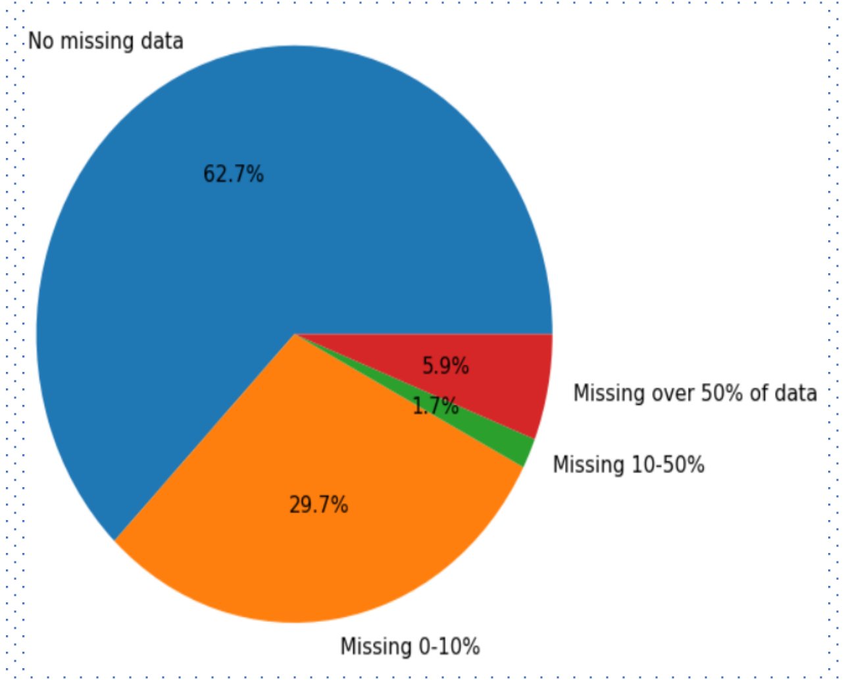
2. GOALS

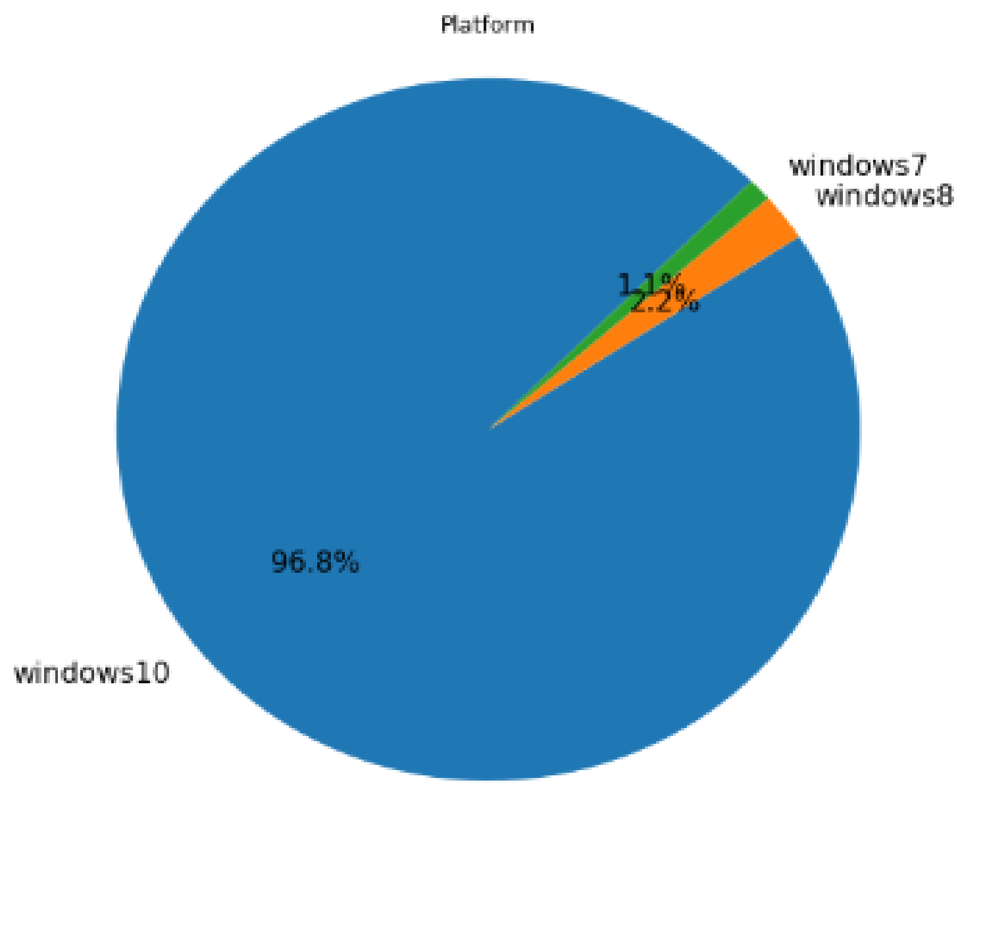
1. Generate insights to determine configurations that make a system vulnerable/resistant to malware attacks.
2. To implement a model which can detect the vulnerability of a system given its configuration as input.
3. Predict the combination of OS configurations and anti-malware mechanisms that will help secure a system

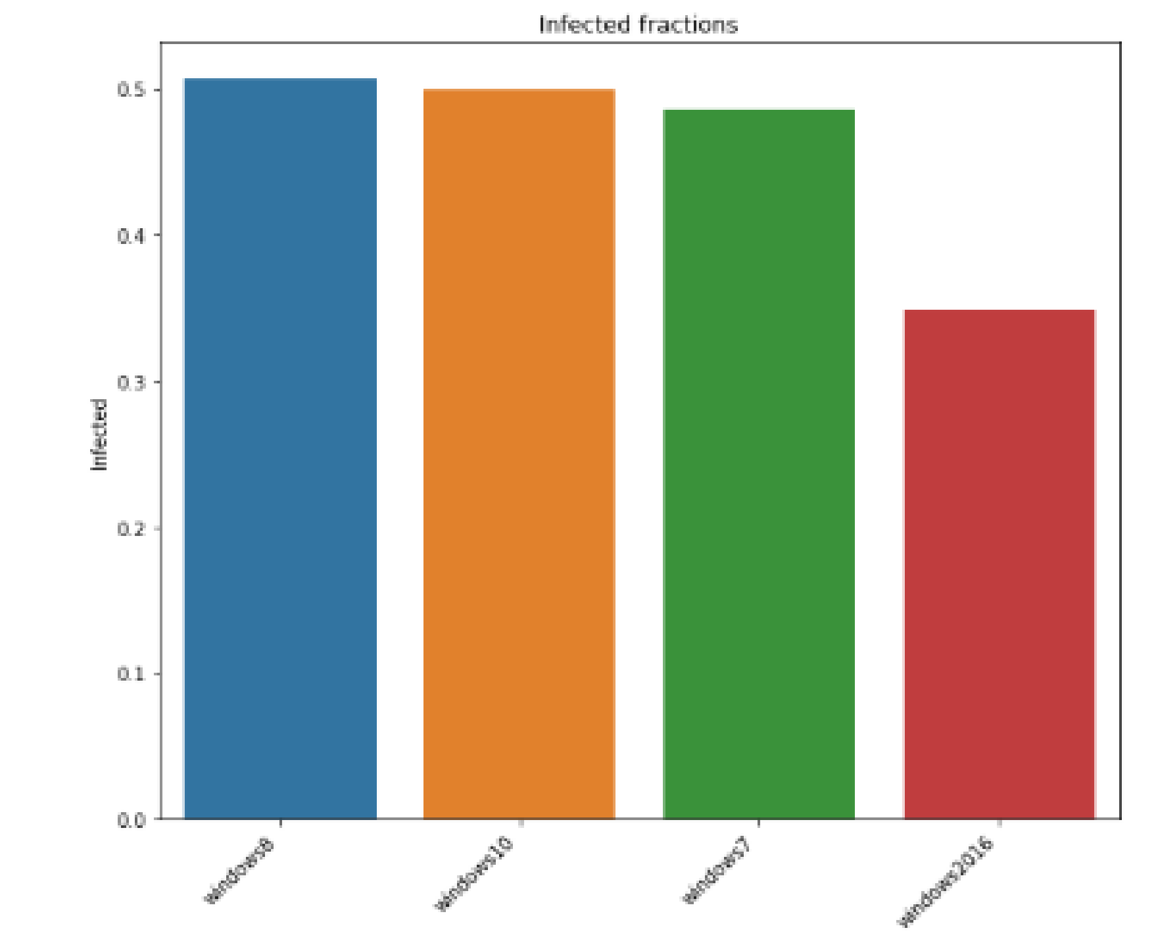
3. DATASET

We are using a dataset provided by Microsoft on Kaggle containing 83 features and around 8 million instances. It contains 29 categorial variables, 41 numerical variables, and 13 binary variables





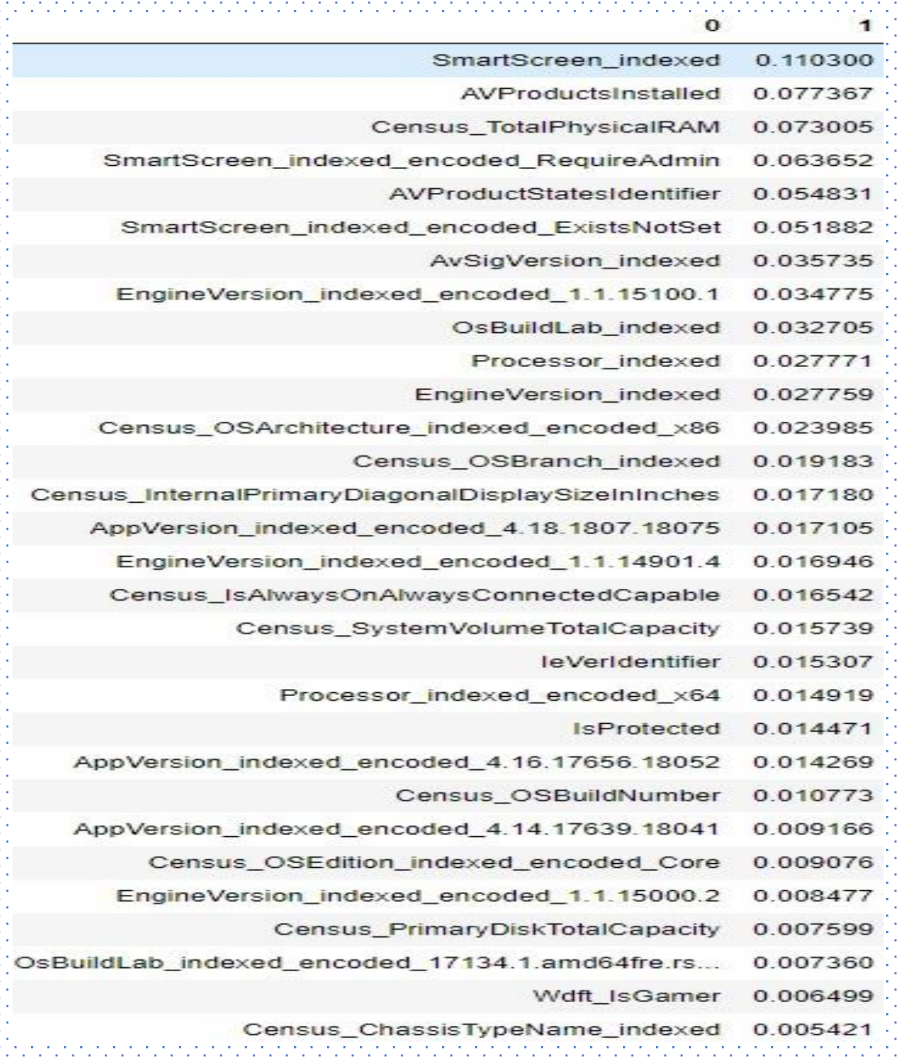




4. MODELS AND FEATURES

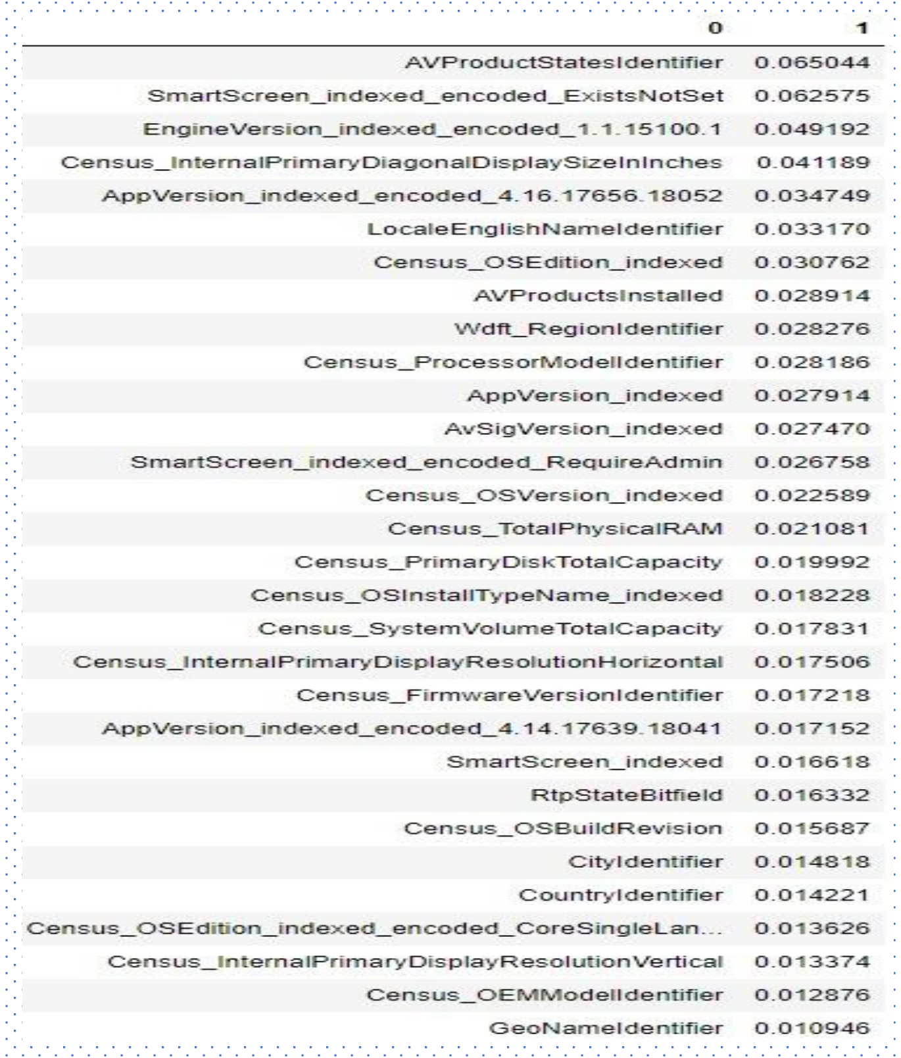
Model	No. of features	Techniques	Evaluation
Logistic Regression	73	Spark pipelines, One Hot Encoding, Dummy variables, Cross validation	AUC
Random Forest	73	Spark pipelines, One Hot Encoding, Dummy variables, Cross validation	AUC
Gradient Boosting Trees	73	Spark pipelines, One Hot Encoding, Dummy variables, Cross validation	AUC

We performed 5-fold cross validation for random forest and GBT by splitting the training data into 3 sets of train, validation and test.



0	1
SmartScreen_indexed	0.110300
AVProductsInstalled	0.077367
Census_TotalPhysicalRAM	0.073005
SmartScreen_indexed_encoded_RequireAdmin	0.063652
AVProductStatesIdentifier	0.054831
SmartScreen_indexed_encoded_ExistsNotSet	0.051882
AvSigVersion_indexed	0.035735
EngineVersion_indexed_encoded_1.1.15100.1	0.034775
OsBuildLab_indexed	0.032705
Processor_indexed	0.027771
EngineVersion_indexed	0.027759
Census_OSArchitecture_indexed_encoded_x86	0.023985
Census_OSBranch_indexed	0.019183
Census_InternalPrimaryDiagonalDisplaySizeInInches	0.017180
AppVersion_indexed_encoded_4.18.1807.18075	0.017105
EngineVersion_indexed_encoded_1.1.14901.4	0.016946
Census_IsAlwaysOnAlwaysConnectedCapable	0.016542
Census_SystemVolumeTotalCapacity	0.015739
IeVerIdentifier	0.015307
Processor_indexed_encoded_x64	0.014919
IsProtected	0.014471
AppVersion_indexed_encoded_4.16.17656.18052	0.014269
Census_OSBuildNumber	0.010773
AppVersion_indexed_encoded_4.14.17639.18041	0.009166
Census_OSEdition_indexed_encoded_Core	0.009076
EngineVersion_indexed_encoded_1.1.15000.2	0.008477
Census_PrimaryDiskTotalCapacity	0.007599
OsBuildLab_indexed_encoded_17134.1.amd64fre.rs...	0.007360
Wdft_IsGamer	0.006499
Census_ChassisTypeName_indexed	0.005421

Random forest feature importance



0	1
AVProductStatesIdentifier	0.065044
SmartScreen_indexed_encoded_ExistsNotSet	0.062575
EngineVersion_indexed_encoded_1.1.15100.1	0.049192
Census_InternalPrimaryDiagonalDisplaySizeInInches	0.041189
AppVersion_indexed_encoded_4.16.17656.18052	0.034749
LocaleEnglishNameIdentifier	0.033170
Census_OSEdition_indexed	0.030762
AVProductsInstalled	0.028914
Wdft_RegionIdentifier	0.028276
Census_ProcessorModelIdentifier	0.028186
AppVersion_indexed	0.027914
AvSigVersion_indexed	0.027470
SmartScreen_indexed_encoded_RequireAdmin	0.026758
Census_OSVersion_indexed	0.022589
Census_TotalPhysicalRAM	0.021081
Census_PrimaryDiskTotalCapacity	0.019992
Census_OSInstallTypeName_indexed	0.018228
Census_SystemVolumeTotalCapacity	0.017831
Census_InternalPrimaryDisplayResolutionHorizontal	0.017506
Census_FirmwareVersionIdentifier	0.017218
AppVersion_indexed_encoded_4.14.17639.18041	0.017152
SmartScreen_indexed	0.016618
RtpStateBitfield	0.016332
Census_OSBuildRevision	0.015687
CityIdentifier	0.014818
CountryIdentifier	0.014221
Census_OSEdition_indexed_encoded_CoreSingleLan...	0.013626
Census_InternalPrimaryDisplayResolutionVertical	0.013374
Census_OEMModelIdentifier	0.012876
GeoNameIdentifier	0.010946

GBT feature importance

The feature importance values derived from random forest and gradient boosting trees. While the order of the variables may be different, they both on average have the same distribution of important features. Features related to antivirus settings like AVProductsInstalled or AVProductsStatesIdentifier as expected are the most important features.

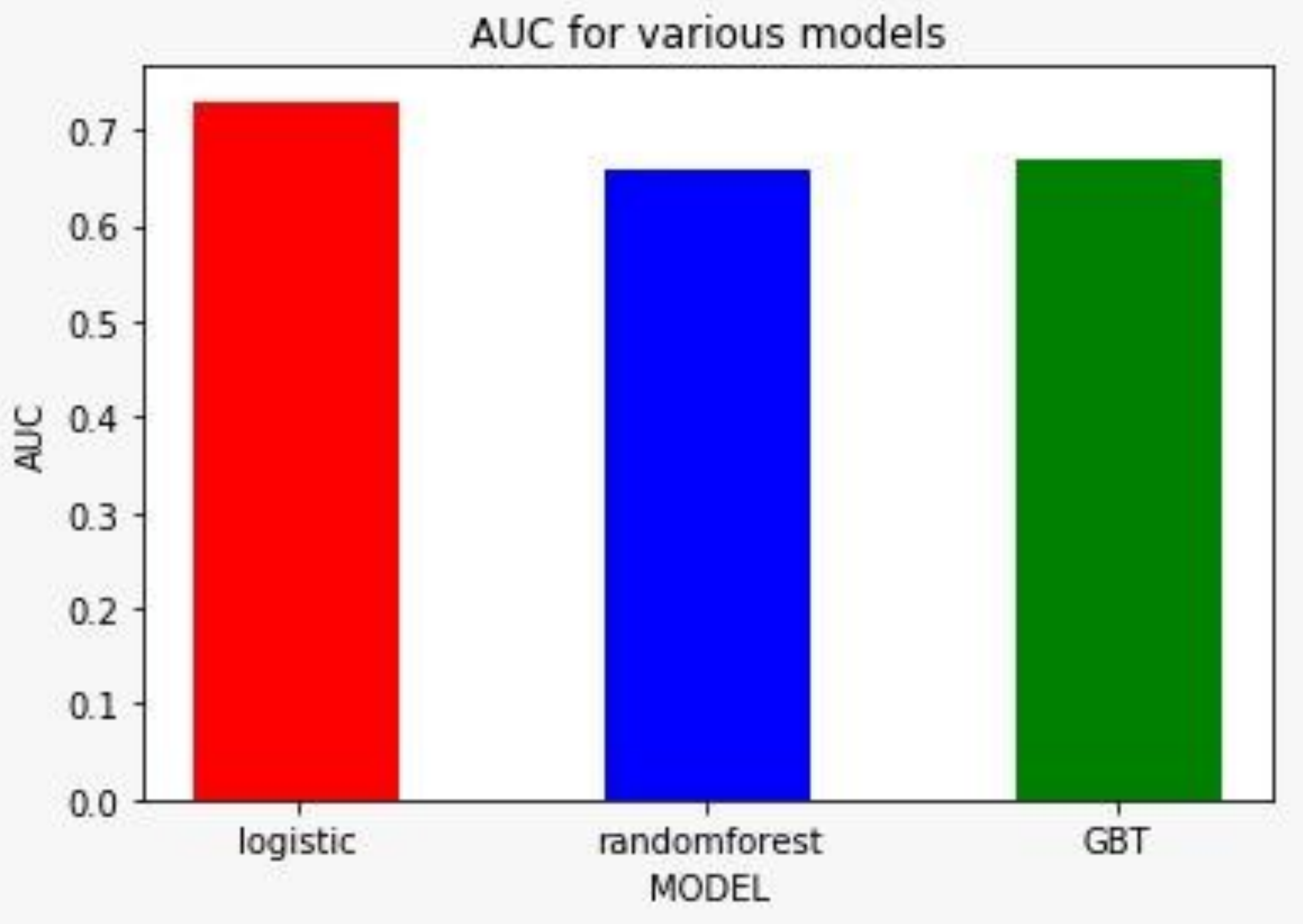
However, a strange observance was noticed in the features. The column "SmartScreen" was also very important to the model. This was a peculiar fact that came up in our results.

5. Model Comparison (Validation)

Data was randomly split by ratio 60% (training), 30% (validation), 10% (testing)

Classification Models

Area under the ROC curve (AUC) – the higher the better



6. Conclusion

By using the following models, anti-malware software developers and OS companies can determine configurations that make a machine susceptible to malware attacks and implement in-built security mechanisms to preventively protect a machine and advise its users on making some configuration changes to make it secure.

7. PROBLEMS

The major problems faced were in relation to the size of the data. Only a very small part of the original dataset could be analyzed (100K) rows. Another problem was the imputation of values. A lot of the string features had to be dropped.

Also, as the description of the data mentions that this dataset is not an accurate representation of real-world data. It could be difficult to translate the models to practical applications.