Project Introduction

Overview

The data was posted by Microsoft company on Kaggle. The original goal of this competition is to predict a Windows machine's probability of getting infected by various families of malware, based on properties of that machine. Beside prediction of the infection rate, I propose to inquiry the

The raw dataset is about 4.38GB and contains almost 9 million records. Each record represent a machine and there are 82 features describing properties of a given machine. The response is a logical value, indicating if a machine is infected by malwares.

Feature Engineering

Most features are categorical variables and require further processing.

1. lots of categories, but can extract information from it, merge some categories: e.g. EngineVersion: 1.1.15100.1, 1.1.14600.4, etc.can be split into 4 sub-features by "." and treat each sub-feature differently.
2. lots of categories, unable to extract information from it, but features has meaning, e.g. country code, etc. If we use them directly, the data might be too noisy and contain irrelevant information. We could convert the original feature categories by using techniques such as **Greedy TS** or **binary encoding**.
3. Remove the **collinearity** within features, such as features "Census_OSEdition" and "Census_OSSkuName" are strongly positively correlation, feature "Census_OSEdition" is removed from the dataset. Similarly, features "Census_OSArchitecture", "Processor", "OsBuildLab", "Census_OSBranch", "Census_OSBuildNumber" , and "Census_OSBuildRevision" are correlation since their information are nested.

After the first level processing, we end up with 249 features in total.

Evaluation

After feature engineering, I applied lightGBM to the datasets with raw features and processed features and found that **the accuracy is improved from 63.33% to 70.59%**. Thus we could assume that the processed features have following advantages:

1. when raw features are used, all features are transformed to categories which results in lots of levels for some features. There are lots of noisy information and affect the model performance. In addition, it's could also cause computation burden.
2. Since the processed features are manually handled and it retains meaningful information, so it facinate the following result interpretation.

Some fun facts

- There is a feature named "IsProtect" indicating if there is an active antivirus product on a given machine. In the training data, about 95% machines have active protection. However, for machines without antivirus product, about 38% machines have been infected; for machines with antivirus product, the infection rate is increased by 12%. This fact indicate that the antivirus product does not work.
- The data also shows that about 40% machines are gamer devices, for gamer machine, the infection rate is 54% and for non-gamer machine, the infection rate is 48%. Thus gamer machine tends to be more vulnerable.

- I observed that virtual machine has infection rate of 20% which is much lower comparing to the infection rate of 50% for ordinary machine.

Modeling

Besides lightGBM, I plan to try several other models, such as random forest, neural network, and XGBoost. Ensemble techiniques, such as stacking can be applied to combine results from different models.