Springboard - Data Science Track
Jacqueline (Xuan) Guo

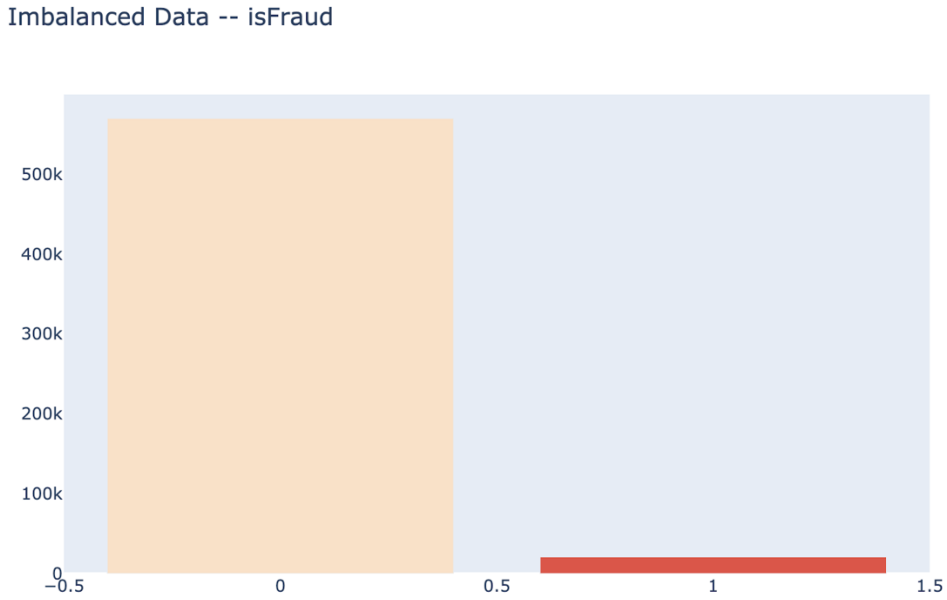
First Capstone Project – In-depth Analysis

May 27, 2020

Objective:
Utilize supervised and unsupervised learning techniques to build predictive models for the transaction data.

Introduction:
So far I have gone through the data, did some exploratory analysis on the dataset. With the understanding gained, this report would focus on feature engineering and selection for the model, building the model and evaluating the model.

Metrics:
Before jumping into feature engineering and model, we need to think about the metrics used to evaluate the model. Using simpler metrics like accuracy can be misleading. In a dataset with highly unbalanced classes, if the classifier always "predicts" the most common class without performing any analysis of the features, it will still have a high accuracy rate, this is obviously something I want to avoid. And in this case, the data is indeed highly imbalanced:



Simply predicts everything true would have ~92% accuracy. Since our problem is fraud detection, what we really care is to detect as much as frauds as possible while it's ok to classify non-fraud action as fraud. That means we should be focusing on minimizing the false negative

rate instead of false positive. Therefore, I decided to use recall (the number of true positives divided by the number of positive values) as the metric.

Feature engineering:

Based on exploratory data analysis, here are the feature engineering I did (aside from minor data cleaning such as lowercase all device info):

- Oversample the fraud class.
- Fill missing values with median and create a new column that counts the missing value.
- Create new feature representing transaction hour in a day.
- Create two new aggregated feature using card1-2 and addr1-2 to represent user id.
- Change device version into latest and outdated where outdated is represented in a numerical scale.
- Only retain most popular and highest fraud rate email domain, changing rest of them to 'other'.
- use the correlation to drop some features that are high correlated with each other

Model:

As this is a classification problem – classify if a data point is fraud or not. Decision tree-based model is a good choice, so I used decision model. The recall was ~0.65.

Then I used random forest model and applied parameter tuning on the model using grid search and randomized search. I also used unshuffled KFold. I separated last 20% as hold-out and got predictions for it by setting another unshuffled KFold with the first 80% of the train data. The final recall is ~0.73

Discuss and Conclusion:

Feature engineering does not always improve the model. There are various cases that I make modification on features based on statistical inference, for example I remove all date information except hour because only hour shows slightly stronger correlation with the label, but then the model performs worse.

Deliverables:

1. Code notebooks
2. Report on the capstone project
3. Presentation on the capstone project