Springboard - Data Science Track
Jacqueline (Xuan) Guo

First Capstone Project – Milestone Report

May 27, 2020

Title:

Improve the efficacy of fraudulent transaction alerts

Introduction:

When doing some transactions with a debit card or credit card, we often face a situation that the card cannot work normally. Imagine there is a bunch of people waiting behind you and you tried your card once, twice, three times, but it's still not working. This is a very embarrassing situation when fraud detection went wrong. Therefore, improving the accuracy of fraudulent transaction alerts would help to avoid this embarrassing situation. Also, reduce the risk when the detection undiscovered leading people to lose money.

Companies such as JP Morgan Chase, Bank of America, and Credit Card Processing would love to use the improved model to predict fraud activities. If this project is successful, it will help hundreds of thousands of businesses reduce their fraud losses and increase their revenue. Customers may also want to choose banks or financial companies with high accuracy fraud detection.

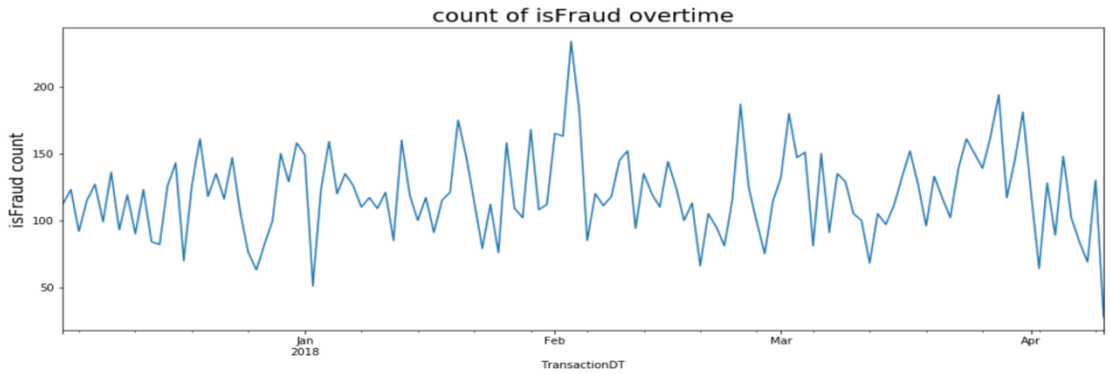
Dataset:

The data comes from Kaggle Competition - IEEE-CIS Fraud Detection**,** Kaggle's data comes from Vesta's real-world e-commerce transactions and contains a wide range of features from device type to product features. The dataset contains 20 kinds of features in total such as *DeviceType* (transaction device type), *addr* (where this transaction was made), and *card* (what kind of card the customer used). Then using those features to analyze the relationship between features and labeled fraud and build machine learning models to improve better accuracy fraud prediction.

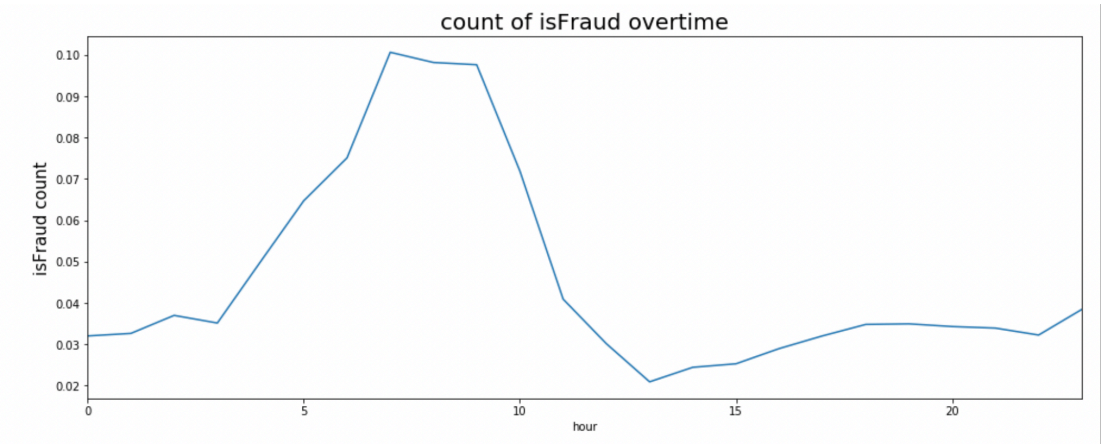
Preprocessing

Since there are many available features in our samples, I decide to do feature selection first to see if there are some features have strong correlations to fraud results.

I. I've noticed that there are features about the date and time in the transaction data set. I want to measure if the fraud action is somehow related to the time of transaction. Therefore, I draw a fraud count graph with daytime and try to find if there's any obvious trends:

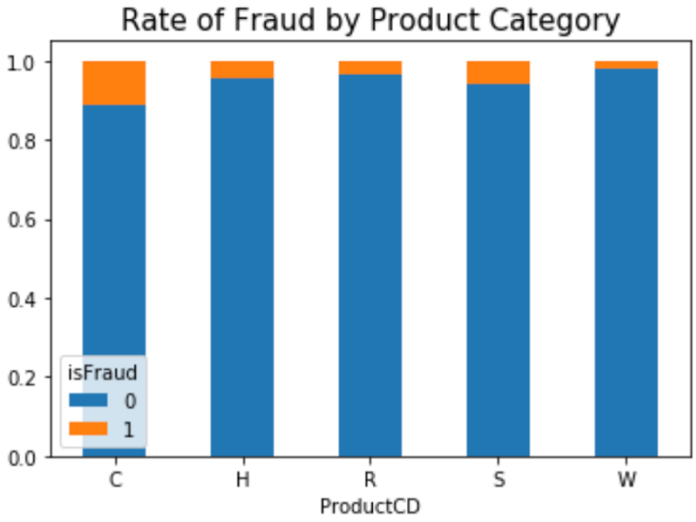


However, according to the figure above: no significant pattern visually based on days. Thus, I changed the unit to hours:



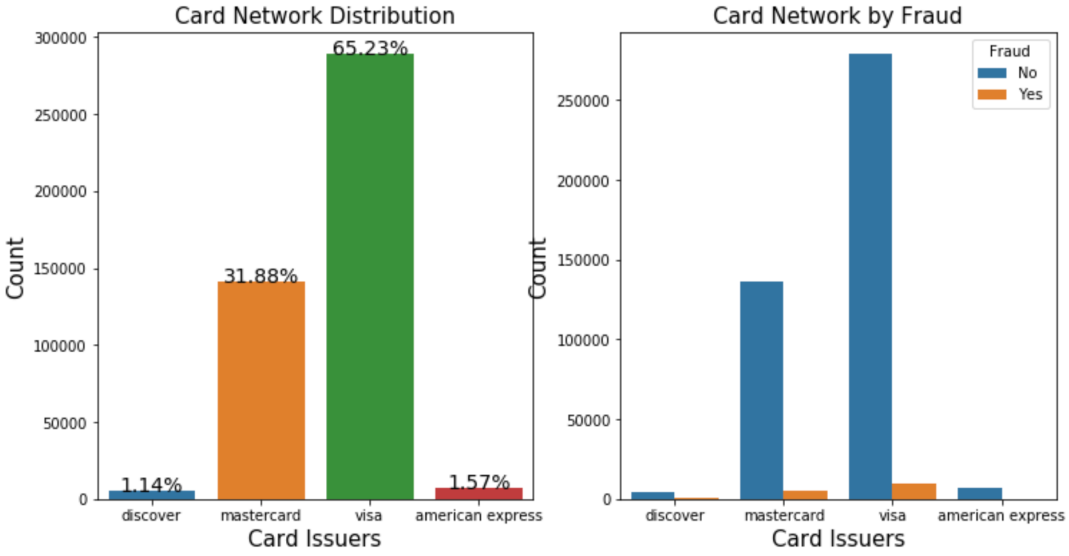
Then we can notice that there's a sharp increase between 5 am to 10 am. Thus we could conclude that the fraud action seems to peak during early morning.

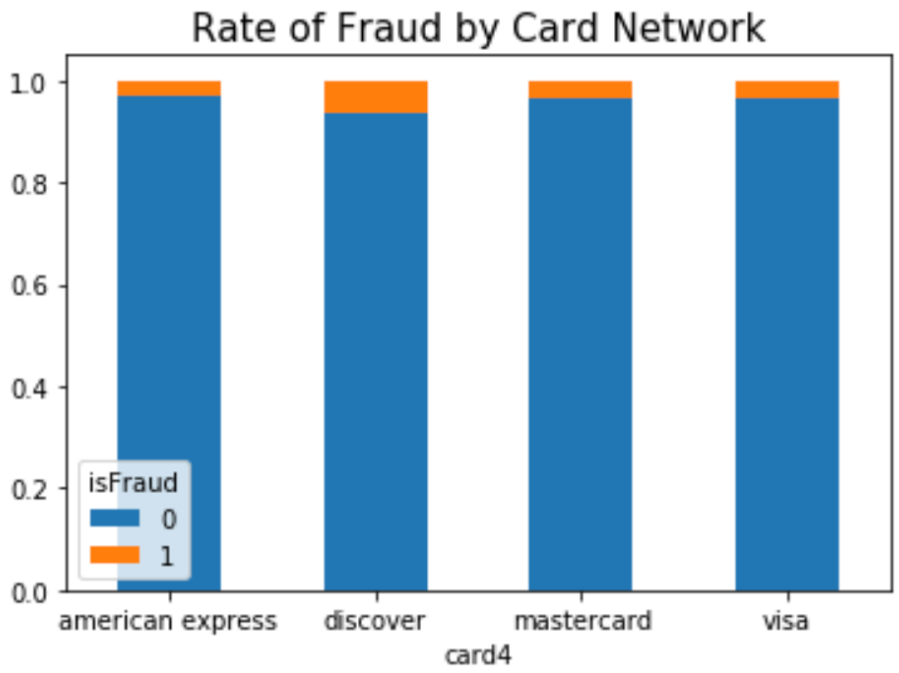
II. In addition, I've noticed that different product type may cause more or less frauds as the following figure shows:



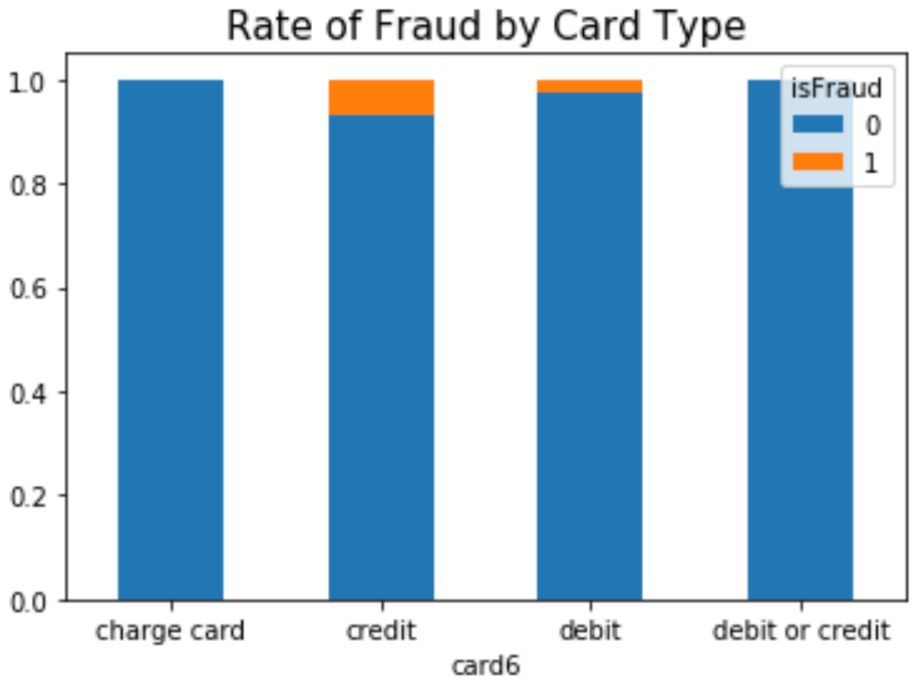
As the figure shows, product C takes up 67.5% of fraud cases for transactions that have identity. It also has highest rate of fraud -- 12%, more than double any other class of product.

III. I've reduced some of the card features and only keep card issuers and card types as the elements of our analysis. I used data visualization to figure out the relationship between card issuers, card types and fraud actions as the following figures show:



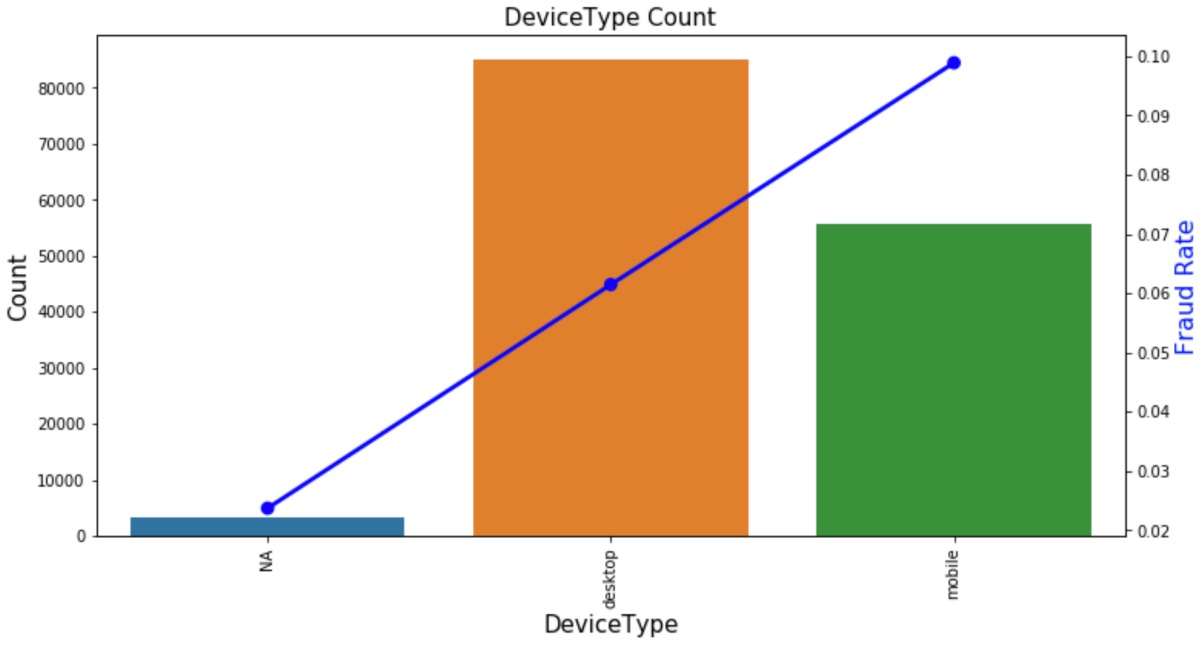


As we can see, although visa has the highest counts of "isfraud", this is because visa is most used type of card. If we divide it by its population then we can see that American Express have a lower fraud rate.



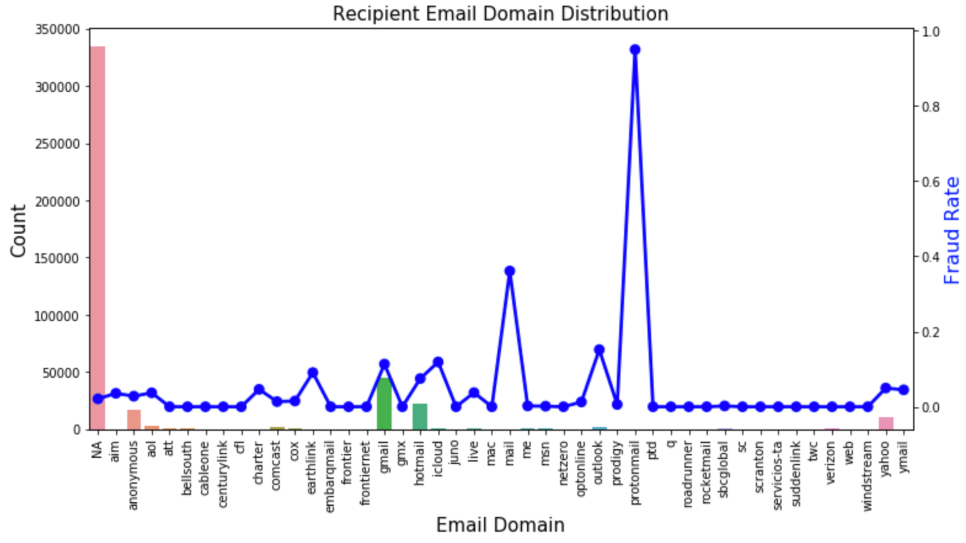
The above figure indicates that credit card tends to have a higher fraud rate.

IV. Moreover, there are features about different device information. It's worth to find out if there are some correlation between different device types and fraud rate. Thus, I draw a figure as the below:



We've noticed that the most way for people to pay the bill is desktop. Although it has higher fraud rate than null type, but it because the transaction amount of this pay device is very high. On the contrary, the fraud rate is very high on mobile payments. It's probably more security hazard in mobile payment service.

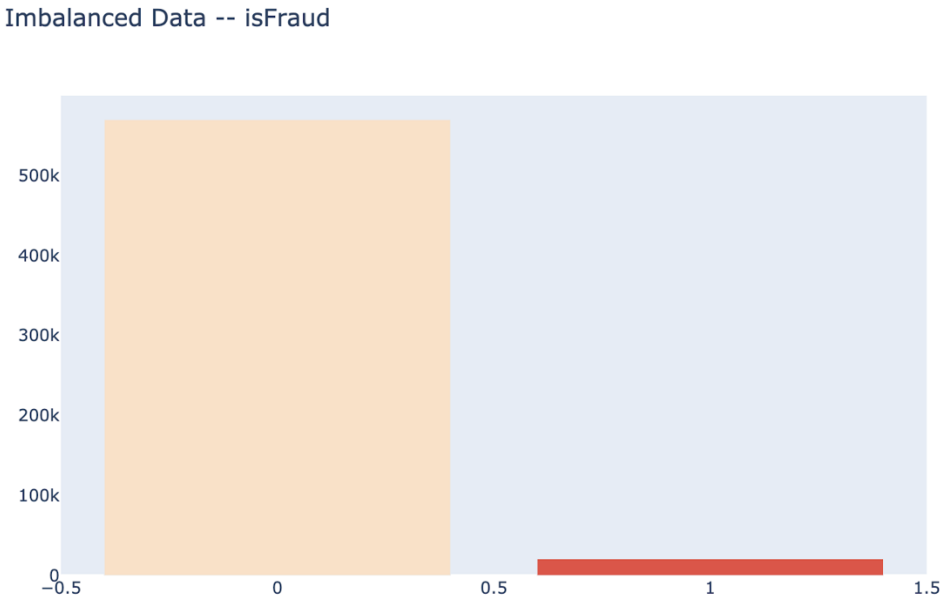
V. I did the same analysis of email domain features and found the fraud rate may also has high correlation to email domain.



In the future research and analysis, we can focus on those features that have strong relationship with the fraud rate and improve the detection from those aspects.

Limitations:

I. I checked if the labeled data is reasonable for us to do data analysis. However, I found the labeled data "isFraud" are imbalance data as the following graph shown:

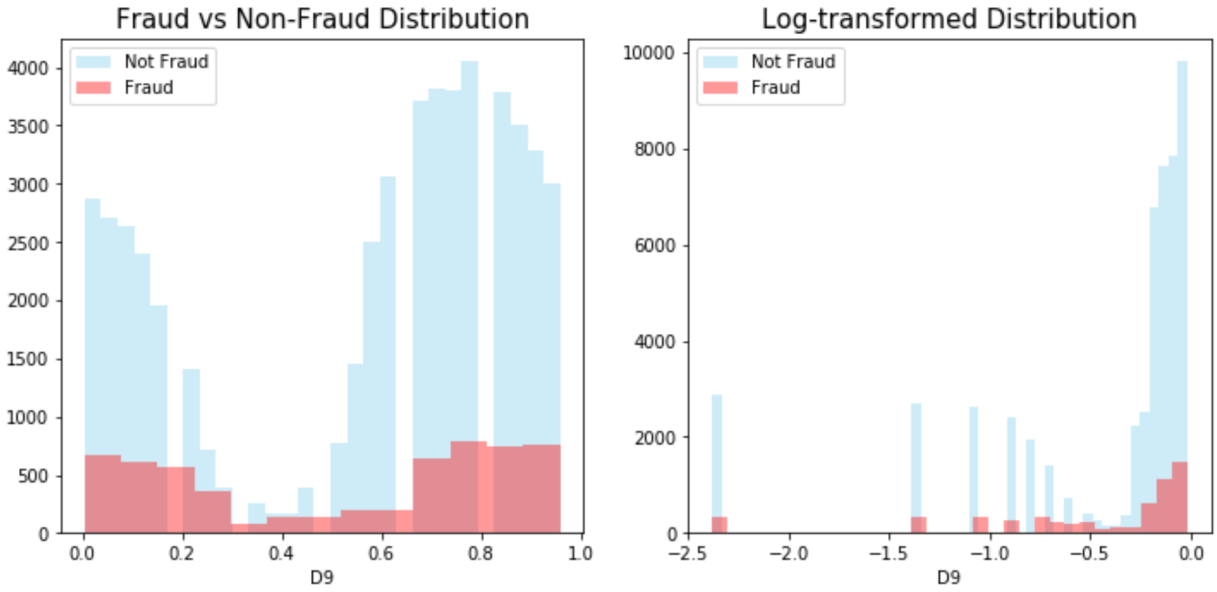


II. I found some of the features is hard to analyze because we could not know the feature meaning due to security reasons. For example, the card information, 4 of them are numeric data and we cannot know what's the meaning of them. What I did is drop them because I think it's probably some personal information such as card expiration date or card number. Thus, those kinds of information would not influence our analysis. I also did the Pearson correlation coefficient between those features to check if it's necessary to be included in our analysis.

III. There are some outliers in the data set, but this data is from the real-world collection. It's hard to determine if the outliers are due to real world phenomenon or some human mistakes. Therefore, it's hard to count outliers into our analysis. For instance, the distance between billing address and shipping address of some credit card transactions are super far, then we could not say it's an outlier or something else. Since it's possibly

that the consumer traveling out of their country and made the purchase. Thus, how to deal with the outliers is hard to determine.

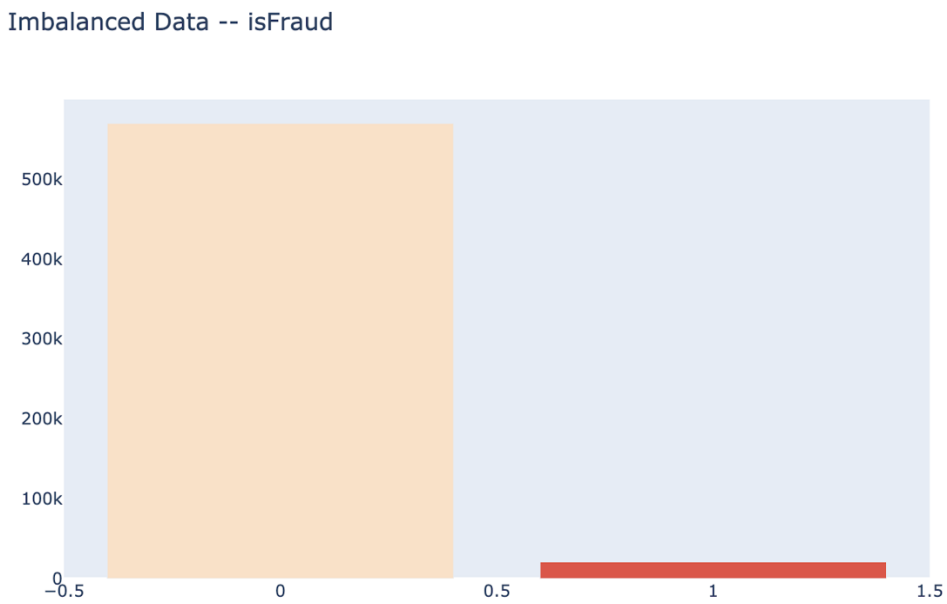
IV. In addition, we could not know the meaning of some features even if we found they are related to fraud rate. We do know the feature 'd8' is strongly related (as the following picture shown) to fraud rate, but it's hard to infer more deeper.



Models:

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

II. 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.

III. 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 features 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

Discuss and Conclusion:

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 K-fold. I separated last 20% as hold-out and got predictions for it by setting another unshuffled K-fold with the first 80% of the train data. The final recall is ~0.73

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.

In this capstone, I spent most of the time on exploring data and feature engineering. I learnt a lot of new ways to gain insight among features and reinforced my previously learnt knowledge. Furthermore, I realized how visualization can be more helpful in large data size. I have better understanding of some of the technique I have learnt, for example, when reducing the dimension of features, I reviewed details of PCA and find out that I can look at eigenvalue of each principle component to determine the dimension to reduce without losing too much information on original data. Through evaluating the model, I learnt all sorts of metrics such as precision, recall, f1 score, AUC ROC, etc.

Future:
This Kaggle competition data set has lots of features for credit card transaction and identity information. To make the prediction more accuracy, sometimes we should abandon some weak features and only use features that strong related to fraud rate. However, I could not completely understand the relations between features, so it's may cause overfitting.

In addition, each feature may have more aspects for us to understand this problem and do our analysis. Take the device information as an example, we've already found different device may cause the different level of fraud rate. However, on the other hand, the device system version may also indicate the probability of fraud action happening. More detail on this point, if the consumer's device system is up to date (newest version), then it means new techniques may already applied on this consumer's device to protect his/her from fraud. On the contrary, if a consumer didn't pay any attention to their system or any updates, then they might too lazy to update their security protection way such as password. This also could lead more fraud.

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

Reference:

Kaggle.com. 2019. IEEE-CIS Fraud Detection | Kaggle. [online] Available at: <https://www.kaggle.com/c/ieee-fraud-detection/overview> [Accessed 28 May 2020].