Detecting credit card frauds using Machine learning

The challenge is to recognize fraudulent credit card transactions so that the customers of credit card companies are not charged for items that they did not purchase.

Main challenges involved in credit card fraud detection are:

- Enormous Data is processed every day and the model build must be fast enough to respond to the scam in time.
- Imbalanced Data i.e most of the transactions (99.8%) are not fraudulent which makes it really hard for detecting the fraudulent ones
- Data availability as the data is mostly private.
- Misclassified Data can be another major issue, as not every fraudulent transaction is caught and reported.
- Adaptive techniques used against the model by the scammers.

How to tackle these challenges?

The model used must be simple and fast enough to detect the anomaly and classify it as a fraudulent transaction as quickly as possible.

Imbalance can be dealt with by properly using some methods which we will talk about in the next paragraph

For protecting the privacy of the user the dimensionality of the data can be reduced.

A more trustworthy source must be taken which double-check the data, at least for training the model.

We can make the model simple and interpretable so that when the scammer adapts to it with just some tweaks we can have a new model up and running to deploy.

Exploratory knowledge Analysis (EDA)

Since nearly all predictors are anonymized, i made a decision to target the non-anonymized predictors time and quantity of the dealings throughout my EDA. the info set contains 284,807 transactions. The norm of all transactions is $88.35 whereas the most important dealings recorded during this knowledge set amounts to $25,691.16. However, as you would possibly be estimate without delay supported the mean and most, the distribution of the value of all transactions is heavily right-skewed. The overwhelming majority of transactions square measure

comparatively tiny and solely a small fraction of transactions comes even on the brink of the utmost.

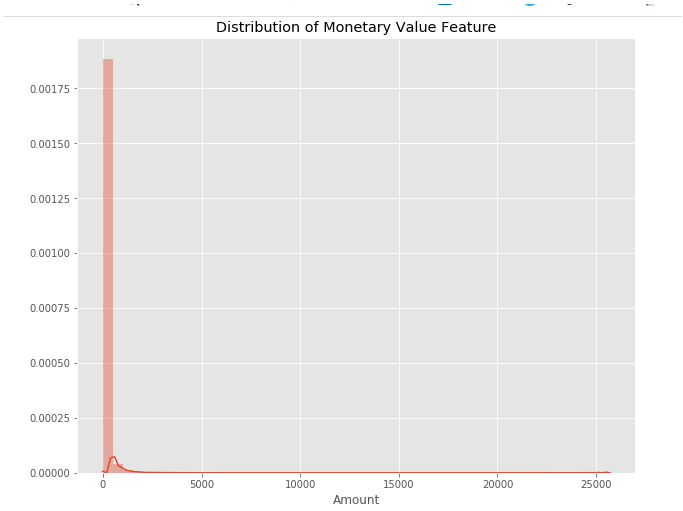


Figure 1: Distribution of Monetary Value Feature

The time is recorded within the range of seconds since the primary group action within the information set. Therefore, we are able to conclude that this information set includes all transactions recorded over the course of 2 days. As against the distribution of the cost of the transactions, it's bimodal. this means that close to twenty eight hours once the primary group action there was a big visit the degree of transactions. whereas the time of the primary group action isn't provided, it'd be affordable to assume that the visit volume occurred throughout the night.

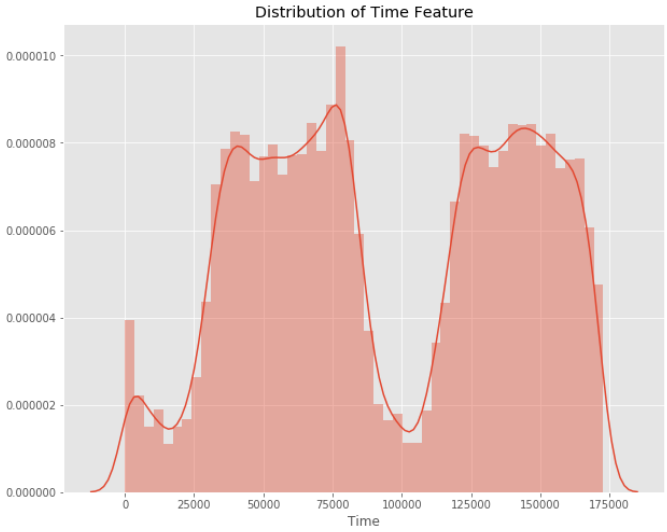


Figure 2: Distribution of Time feature

What regarding the category distributions? Well, as may be expected, most transactions area unit non-fraudulent. In fact, 99.83% of the transactions during this information set weren't deceitful whereas solely zero.17% were deceitful. the subsequent mental image underlines this important distinction.

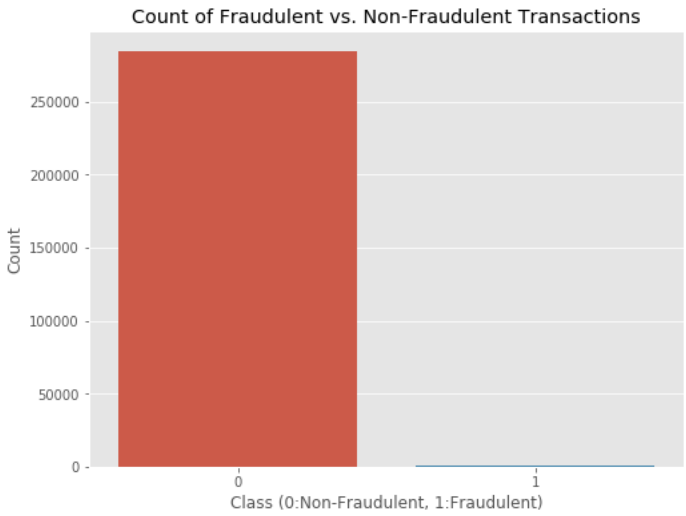


Figure 3: Count of fraudulent and non-fraudulent

Finally, it'd be fascinating to grasp if there square measure any vital correlations between our predictors, particularly with regards to our category variable. one in every of the foremost visually appealing ways in which to work out that's by employing a heatmap.

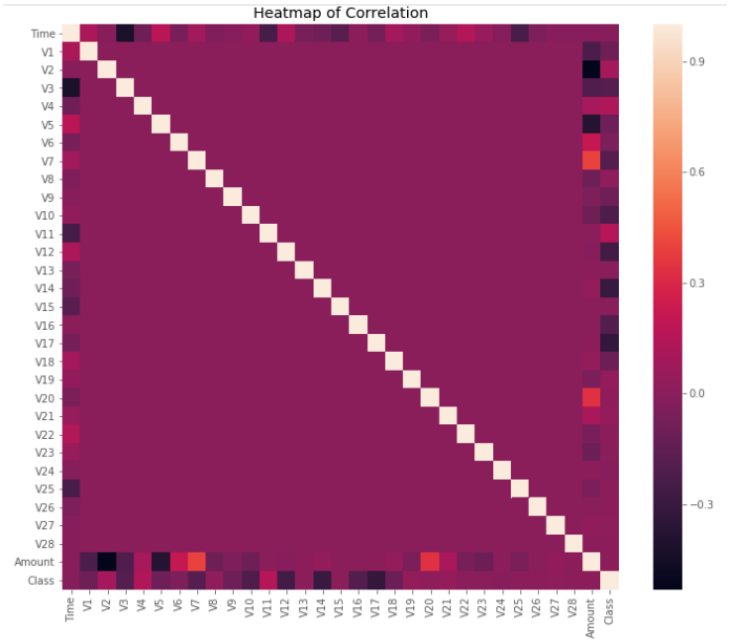


Figure 4: Heatmap of Correlation

As you'll see, a number of our predictors do appear to be related to with the category variable. still, there appear to be comparatively very little important correlations for such an enormous variety of variables. this will most likely be attributed to 2 factors:

- The data was ready employing a PCA, so our predictors are principal elements.
- The huge category imbalance would possibly distort the importance of bound correlations with regards to our category variable.

Data Preparation

Before continued with our analysis, it's vital to not forget that whereas the anonymized options are scaled and appear to be focused around zero, our time and quantity options haven't. Not scaling them likewise would end in bound machine learning algorithms that provide weights to options (logistic regression) or think about a distance live (KNN) playacting abundant worse. To avoid this issue, I standardized each the time and quantity column. Luckily, there aren't any missing values and that we, therefore, don't have to be compelled to worry concerning missing worth imputation.

Creating a Training Set for a Heavily Imbalanced Data Set

Now comes the difficult part: making a coaching knowledge set which will permit our algorithms to select up the precise characteristics that build a dealing a lot of or less possible to be dishonorable. victimization the initial knowledge set wouldn't encourage be a decent plan for a awfully straightforward reason: Since over ninety nine of our transactions square measure non-fraudulent, AN algorithmic program that invariably predicts that the dealing is non-fraudulent would accomplish AN accuracy more than ninety nine. however, that's the alternative of what we wish. we have a tendency to don't need a ninety nine accuracy that's achieved by ne'er labeling a dealing as dishonorable, we wish to discover dishonorable transactions and label them in and of itself.

There square measure 2 key points to concentrate on to assist U.S.A. solve this. First, we have a tendency to square measure about to utilize random under-sampling to form a coaching dataset with a balanced category distribution which will force the algorithms to discover dishonorable transactions in and of itself to attain high performance. Speaking of performance, we have a tendency to don't seem to be about to have confidence accuracy. Instead, we have a tendency to ar} about to build use of the Receiver in operation Characteristics-Area beneath the Curve or ROC-AUC performance measure (I have connected any reading below this article). primarily, the ROC-AUC outputs a price between zero and one, whereby one could be a excellent score and nil the worst. If AN algorithmic program encompasses a ROC-AUC score of higher than zero.5, it's achieving a better performance than random estimation.

To create our balanced coaching knowledge set, I took all of the dishonorable transactions in our knowledge set and counted them. Then, I willy-nilly elite identical variety of non-fraudulent

transactions and concatenated the 2. when shuffling this recently created knowledge set, I made a decision to output the category distributions another time to examine the distinction.

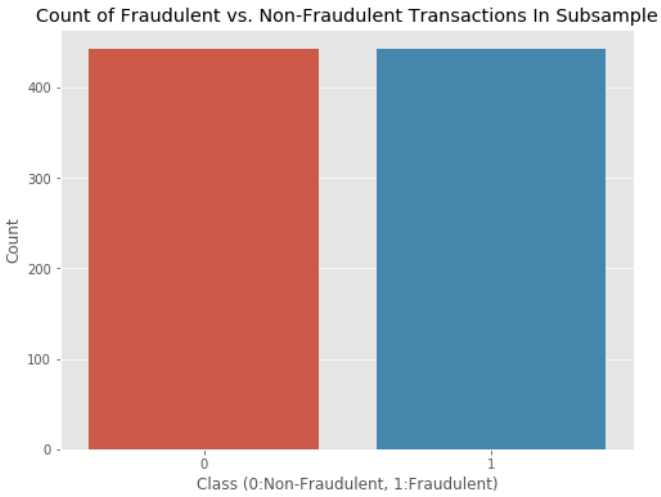
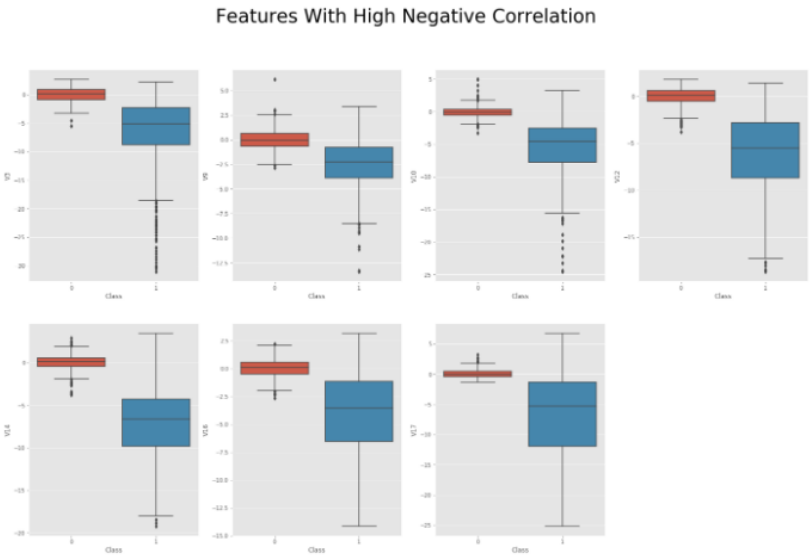
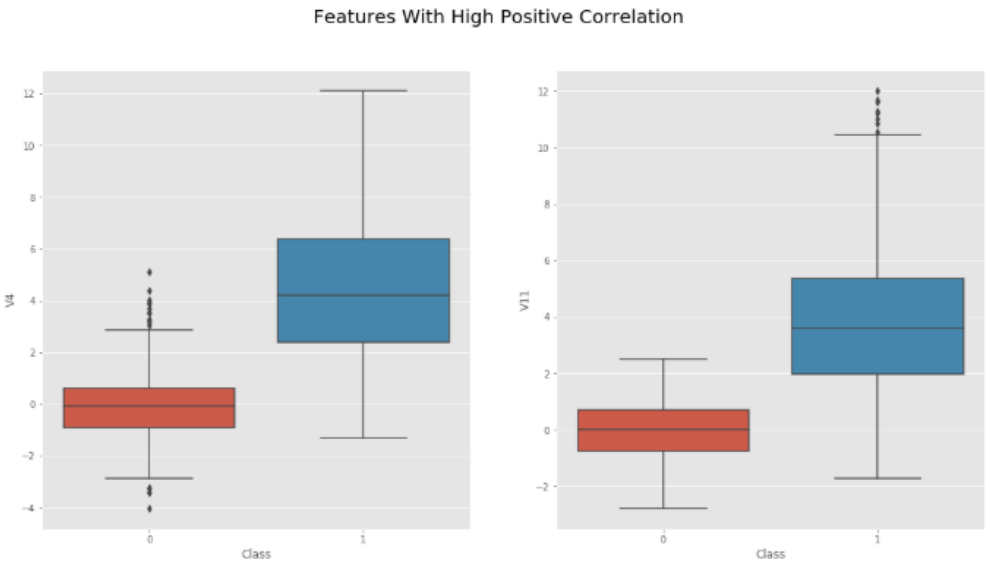


Figure 4: Count of Fraudulent and Non-Fraudulent Transactions is subsample

Outlier Detection & Removal

Outlier detection is a complex topic. The trade-off between reducing the number of transactions and thus volume of information available to my algorithms and having extreme outliers skew the results of your predictions is not easily solvable and highly depends on your data and goals. In my case, I decided to focus exclusively on features with a correlation of 0.5 or higher with the class variable for outlier removal. Before getting into the actual outlier removal, let's take a look at visualizations of those features:





Box plots give U.S.A. with an honest intuition of whether or not we'd like to stress concerning outliers as all transactions outside of one.5 times the IQR (Inter-Quartile Range) square measure sometimes thought-about to be outliers. However, removing all transactions outside of one.5 times the IQR would dramatically decrease our coaching knowledge size, that isn't terribly massive, to start with. Thus, i made a decision to solely specialise in extreme outliers outside of two.5 times the IQR.

Dimensionality Reduction With t-SNE for Visualization

Visualizing our categories would influence be quite attention-grabbing and show USA if they're clearly severable. However, it's unfeasible to provide a 30-dimensional plot exploitation all of our predictors. Instead, employing a spatiality reduction technique like t-SNE, we have a tendency to area unit able to project these higher dimensional distributions into lower-dimensional visualizations. For this project, i made a decision to use t-SNE, AN rule that I had not been operating with before. If you'd prefer to apprehend additional concerning however this rule works, see here.

Projecting our information set into a two-dimensional area, we have a tendency to area unit able to manufacture a scatter plot showing the clusters of dishonest and non-fraudulent transactions:

Classifications Algorithms

Onto the half you've most likely been expecting all this time: coaching machine learning algorithms. To be able to take a look at the performance of our algorithms, I initial performed Associate in Nursing 80/20 train-test split, rending our balanced knowledge set into 2 items. To avoid overfitting, I used the quite common resampling technique of k-fold cross-validation. This merely means you separate your coaching knowledge into k elements (folds) and so suit your model on k-1 folds before creating predictions for the kth hold-out fold. You then repeat this method for each single fold and average the ensuing predictions.

To get a higher feeling of that formula would perform best on our knowledge, let's quickly control a number of the foremost in style classification algorithms:

- Logistic Regression
- Linear Discriminant Analysis
- K Nearest Neighbors (KNN)
- Classification Trees
- Support Vector Classifier
- Random Forest Classifier
- XGBoost Classifier

The results of this spot-checking can be visualized as follows:

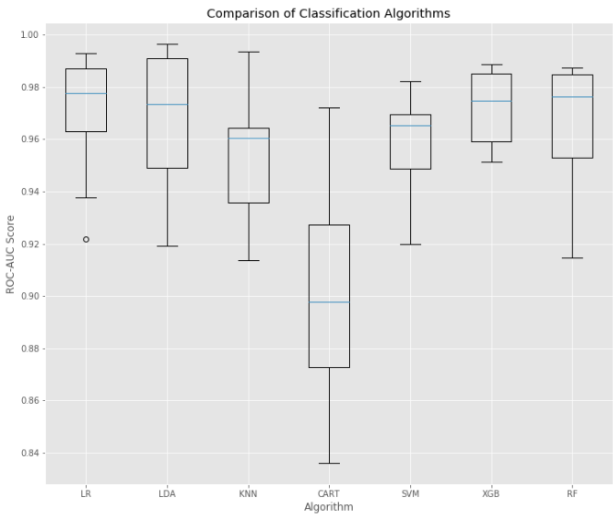


Figure 5: Comparison of classification Algorithms

As we are able to see, there are some algorithms that quite considerably outperformed the others. Now, what rule will we choose? As mentioned on top of, this project had not solely the main target of achieving the best accuracy however additionally to make business price. Therefore, selecting Random Forest over XGBoost could be an inexpensive approach so as to attain a better degree of comprehensiveness whereas solely slightly decreasing performance. To any illustrate what I mean by this, here may be a image of our Random Forest model that might simply be accustomed justify terribly merely why an exact call was made:

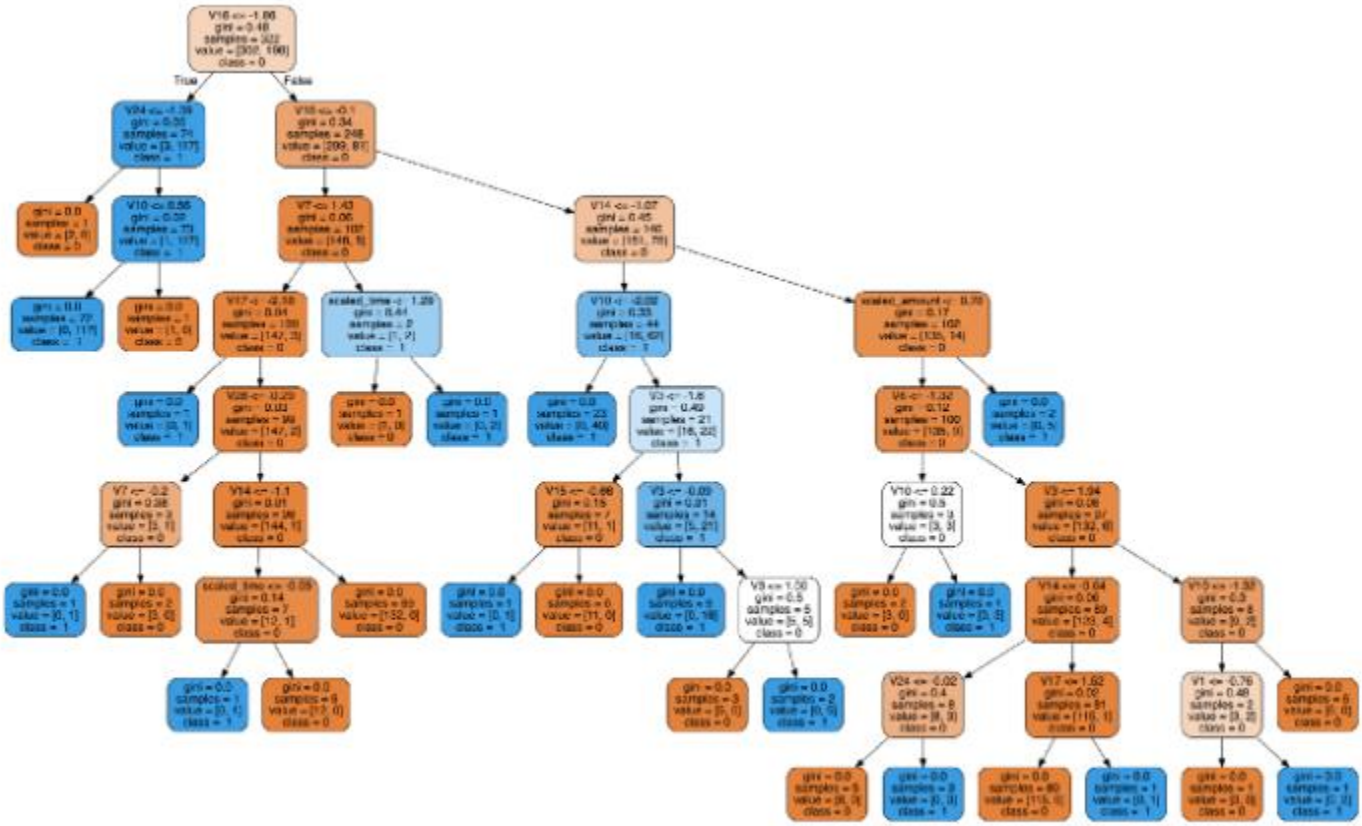


Figure 6: Random Forest Model