Credit Card Fraud Detection Model Comparison

By: Brian Faraira

Motivation

My motivation to try this project was that I was interested in how banking companies decide whether something is a fraudulent transaction or not. I had experiences where my card was locked due to fraudulent activities, so I was interested and now I've learned that they use machine learning models in order to predict if something is fraudulent or not through random forest models and decision tree models.

Importance of the Problem

This problem is very important to everybody's daily life. If your credit card is stolen or your information is stolen then the only measure stopping the thief from taking all of your savings are these machine learning algorithms. Although these models can be inaccurate and annoying at times, it's better if your card is locked than watching someone stealing your life savings without any measures to stop them.

Approach

I first loaded my CSV file containing all of the transactions. I split the data into X and Y, with X containing all of the features, and Y containing the class label. I then used the StandardScaler() which standardizes a feature by subtracting the mean and then scaling to unit variance. I only applied this to the features and not the labels. I then split features and labels into training and test variables.

Implementing Models

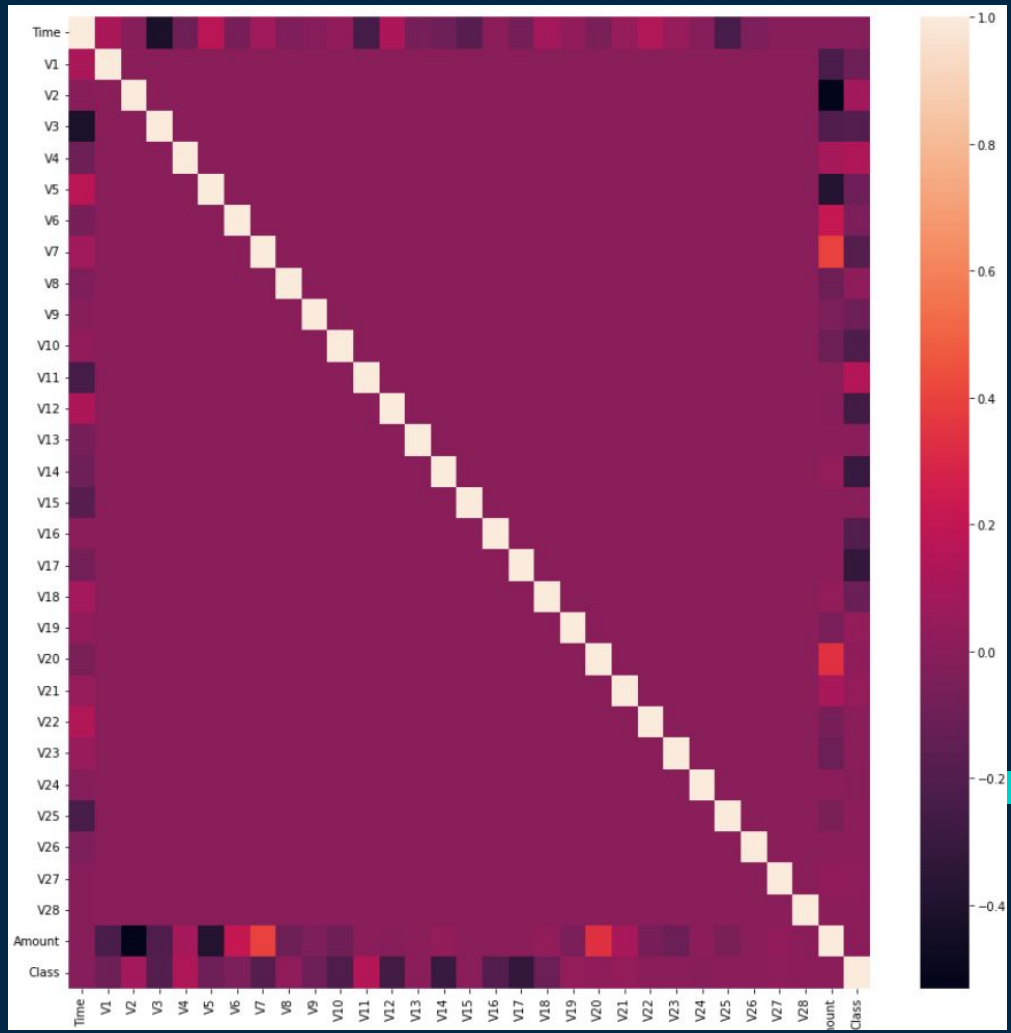
After using the train test split I began to implement logistic regression, SVM, and the random forest model using sklearn. For logistic regression, I had to set the iterations to 1000 and then I fit and predicted the classes for Y. Then I used the recall score, precision score, and the F1 score in order to evaluate the model. I then repeated this for SVM and random forest. The number of fraudulent transactions is only 492 and the number of normal transactions is 284,315. This difference between the classes leads to a highly unbalanced dataset which would mainly affect the SVM model. For SVM, I had standardized the dataset earlier so that the SVM model would work for an unbalanced dataset.

Evaluation Metrics

The classification accuracy metric would usually be the metric used for this dataset, however, since the dataset is highly imbalanced we need to use another evaluation metric. The best metrics for unbalanced data would be precision, recall, and the F1 score. Precision can be used to measure the number of positive instances among the total number of positives. Recall is used to measure the number of true positives out of all positive predictions that could have been made. The F1 score is a combination of both precision and recall and is used to find a balance between the two. The recall score is the most useful score since we don’t mind false positives and we want to avoid false negatives.

Correlations

There is little correlation between features V1 through V28. There are only a few features that have a high correlation with an amount which is shown by the bright red which shows that these features may be redundant. Since there is no red on class which is the label this means there are no irrelevant features.



Results

Logistic Regression Recall: 0.6376811594202898

Logistic Regression Precision: 0.8712871287128713

Logistic Regression F1_score: 0.7364016736401673

SVC Recall: 0.717391304347826

SVC Precision: 0.9801980198019802

SVC F1_score: 0.8284518828451881

Random Forest Recall: 0.8115942028985508

Random Forest Precision: 0.9572649572649573

Random Forest F1_score: 0.8784313725490197

Results

The results showed me that the random forest model was the best model to use for credit card fraud detection. This model had the highest precision, recall, and F1 score since it's a decision tree that has both low bias and low variance due to the bootstrapping implemented into random forest models. The model with the second-highest recall score was SVM and this was due to the fact that the dataset was standardized to be fit for the SVM model. Before standardization, the model had a much lower recall. Logistic regression had the lowest recall score as well as the lowest F1 score and precision score. This is most likely due to the fact that logistic regression is more prone to overfitting when compared to the SVM model.

Lessons Learned

I learned that machine learning algorithms are used in many applications daily and that they aren’t always 100% accurate. They are sometimes when a machine learning model is made to be skewed towards false negatives since some real-world applications benefit more from this. I also learned how to implement the standard scalar method and how some unbalanced datasets need to be balanced before you can use the SVM model. I also learned about using other evaluation metrics when you are presented with an unbalanced dataset.