Credit Card Fraud Detection Report

Kriti Gupta

08/10/2020

Introduction

It is important that credit card companies are able to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase. According to creditcards.com, there was over £300m in fraudulent credit card transactions in the UK in the first half of 2016, with banks preventing over £470m of fraud in the same period. The data shows that credit card fraud is rising, so there is an urgent need to continue to develop new, and improve current, fraud detection methods.

Using this dataset, we will use machine learning to develop a model that attempts to predict whether or not a transaction is fraudlent.The datasets contains transactions made by credit cards in September 2013 by european cardholders.

Due to imbalancing nature of the data, many observations could be predicted as False Negative, in this case Legal Transactions instead of Fraudolent Transaction. For example, a model that predict always **0** (Legal) can archieve an Accuracy of **99.8**. For that reason, the metric used for measuring the score is the **Area Under The Precision-Recall Curve (AUCPR)** instead of the traditional AUC curve. A desiderable result is an AUCPR at least greater than **0.75**.

For archieving the task of classifying credit card fraud detection, they are trained several algorithms such as Naive Bayes Classifier, Logistic Regression,KNN,Random Forest.

In this analysis, a Random Forest model is capable of an AUCPR of **0.768** and Logistic Regression of **0.812**.

Exploratory Data Analysis

The Dataset

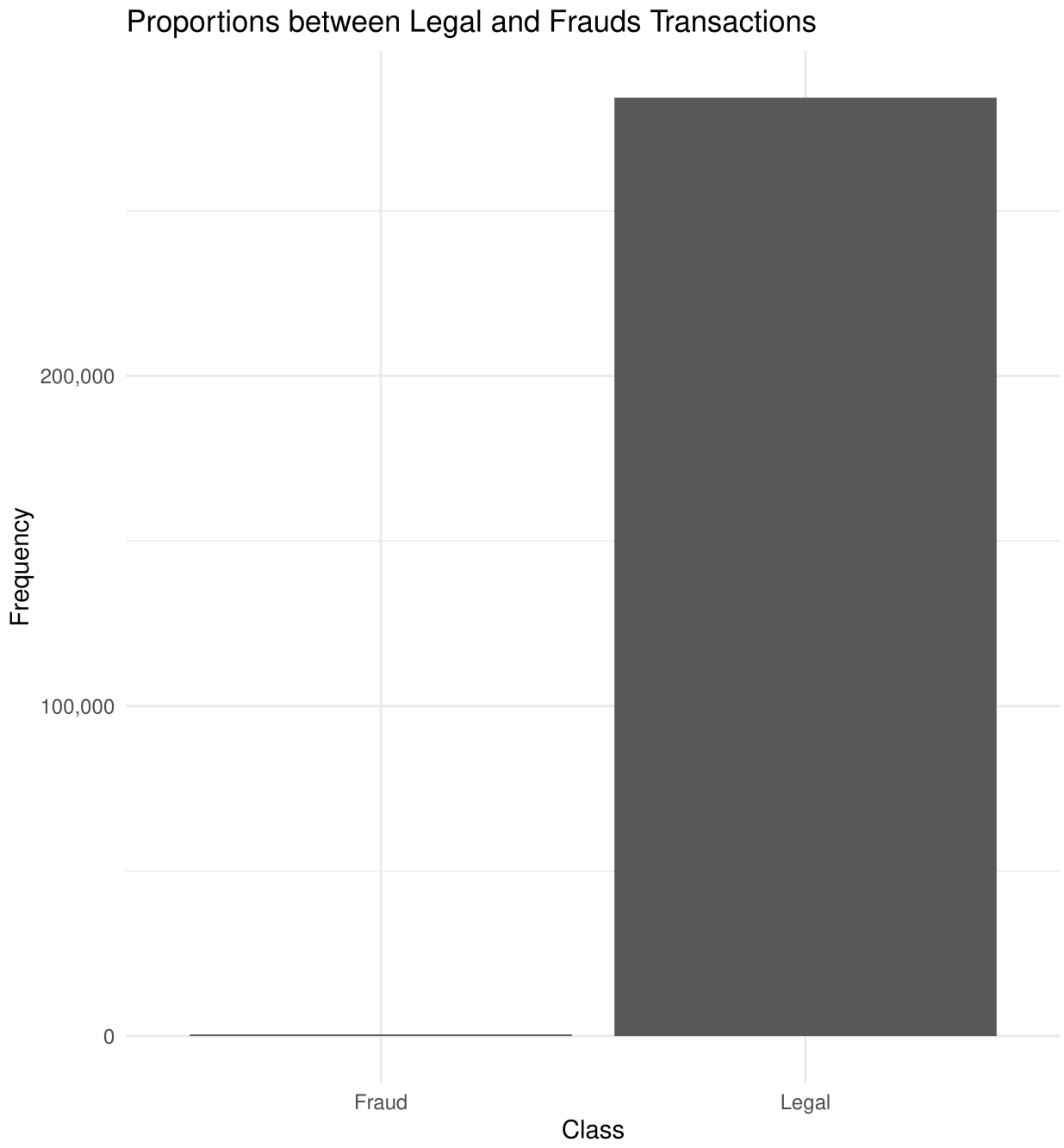
This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions. It contains only numerical input variables which are the result of a PCA transformation. Features V1, V2, ... V28 are the principal components obtained with PCA. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

Dimensions

Length	Columns
284807	31

Imbalanced Dataset

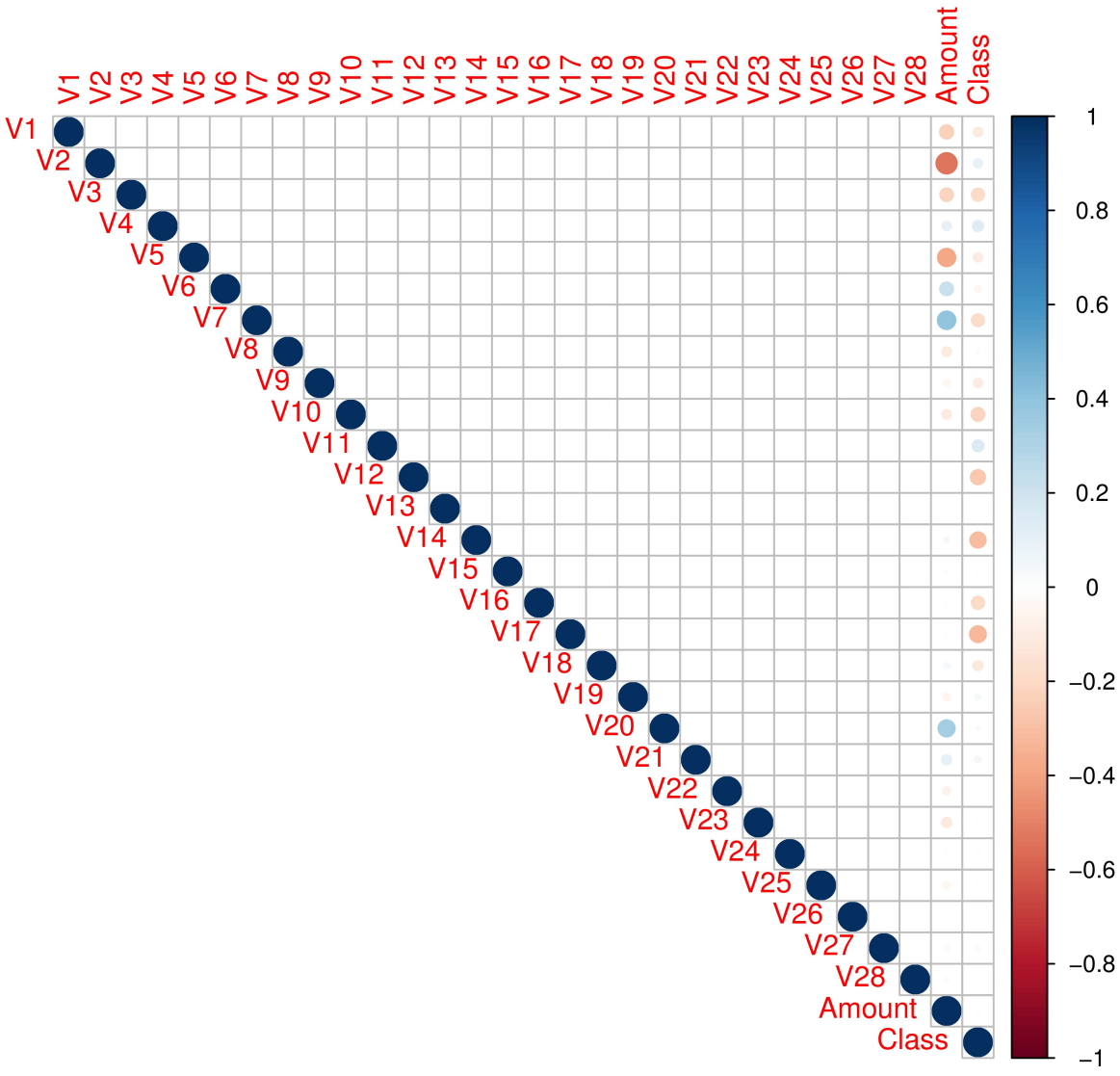
This is a very imbalanced dataset.In this case, only **492** transactions are frauds, represented by **1** and **284315** are not, represented by **0**.



Check for NAs There are no NA values in the data.

```
##    Time      V1      V2      V3      V4      V5      V6      V7      V8      V9     V10
##       0       0       0       0       0       0       0       0       0       0       0
##     V11     V12     V13     V14     V15     V16     V17     V18     V19     V20     V21
##       0       0       0       0       0       0       0       0       0       0       0
##     V22     V23     V24     V25     V26     V27     V28  Amount   Class
##       0       0       0       0       0       0       0       0       0
```

Correlations between variables



The correlation matrix graphically gives us an idea of how features correlate with each other. We can clearly see that most of the features do not correlate to other features but there are some features that either has a positive or a negative correlation with each other. For example, V2 and V5 are highly negatively correlated with the feature called Amount. We also see some correlation with V20 and Amount.

Data Pre-Processing

This involves 2 steps:

1. Remove the "Time" column from the dataset.
2. Split the dataset into train and test. If you train the network more, then you will get a higher accuracy with your testing sample . By testing 40% of data we are not overtraining as we can test more data.

Modeling and Analysis

Model 1 - Naive Baseline Algorithm

This model always predicts class as "Legal" transaction. Accuracy is **99.8** because the data is imbalanced. But AUCPR is **0**.

Model	AUC	AUCPR
Naive Baseline	0.5	0

Model 2 - Naive Bayes Algorithm

The performance improves a little but is still a poor result according to the metric of interest.

Model	AUC	AUCPR
Naive Baseline	0.5000	0.0000
Naive Bayes	0.9176	0.0549

Model 3 - Logistic Regression

It is typically a good idea to start out with a simple model and move on to more complex ones. A simple logistic regression model achieved nearly 100% accuracy.

Model	AUC	AUCPR
Naive Baseline	0.50000	0.00000
Naive Bayes	0.91760	0.05490
Logistic Regression	0.98411	0.81251

Model 4 - K Nearest Neighbours

A KNN Model is used to achieve significant improvement in respect to baseline models in regard of AUCPR. We use default value of k = 5.

Model	AUC	AUCPR
Naive Baseline	0.50000	0.00000
Naive Bayes	0.91760	0.05490
Logistic Regression	0.98411	0.81251
K-Nearest Neighbors	0.81627	0.57976

Model 5 - Random Forest

The ensemble methods are capable of a significant increase in performance.

Model	AUC	AUCPR
Naive Baseline	0.50000	0.00000
Naive Bayes	0.91760	0.05490
Logistic Regression	0.98411	0.81251
K-Nearest Neighbors	0.81627	0.57976
Random Forest	0.89793	0.76835

	MeanDecreaseGini
V1	8.6303
V2	6.9510
V3	8.8552
V4	15.5475
V5	8.3396
V6	10.0785
V7	16.2357
V8	7.2956
V9	24.1753
V10	52.1565
V11	35.2323
V12	78.0863
V13	6.3955
V14	65.8827
V15	6.2699
V16	46.2725
V17	98.9816
V18	15.3196
V19	8.0340
V20	8.8723
V21	11.6151
V22	5.5115
V23	5.4286
V24	5.9804
V25	5.4993
V26	10.3433
V27	9.0177
V28	6.3441
Amount	8.3428

It is interesting to compare variable importances of the RF model with the variables identified earlier as correlated with the “Class” variable. The top 3 most important variables in the RF model were also the ones which were most correlated with the “Class” variable. Especially for large datasets, this means we could save disk space and computation time by only training the model on the most correlated/important variables, sacrificing a bit of model accuracy.

Results

This is the summary results for all the models built.

Model	AUC	AUCPR
Naive Baseline	0.50000	0.00000
Naive Bayes	0.91760	0.05490
Logistic Regression	0.98411	0.81251
K-Nearest Neighbors	0.81627	0.57976
Random Forest	0.89793	0.76835

Conclusion

This project has explored the task of identifying fraudlent transactions based on a dataset of anonymised features. It has been shown that even a very simple logistic regression model can achieve good recall, while a much more complex Random Forest model improves upon logistic regression in terms of AUC. For future improvements there are more models which can improve on both fronts such as XGBoost and GBM. Techniques like SMOTE which help in sampling high-dimensional and imbalanced datasets like this one, can also be used.