Springboard – Data Science Career Track

Capstone Project 3

Credit Card Fraud Detection

By Zhiling Xie

Capstone 3 Final Report

Credit Card Fraud Detection

1. Introduction

1.1 Problem statement

Credit card fraud is a common form of identity theft. It is important that credit card companies can recognize fraudulent credit card transactions so that they do not suffer financial losses and customers are not charged for items that they did not purchase.

This study focuses on modeling the credit card transactions of European cardholders over two days in September 2013 to identify which transactions were fraudulent.

1.2 Goal

This project aims to develop several classification (supervised) and outlier detection (unsupervised) models and find the best model for this fraud detection problem. The target is to build models with highest combination of precision and recall. The predictive models are used to provide guidance for European credit card companies' fraud alert systems.

The implementation details can be found in the Notebooks in this GitHub repository (https://github.com/zxie9/Capstone-Project-Three).

2. Approach

2.1 Data Acquisition and Wrangling

(1) Dataset

The raw data can be downloaded from the webpage:
https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/data

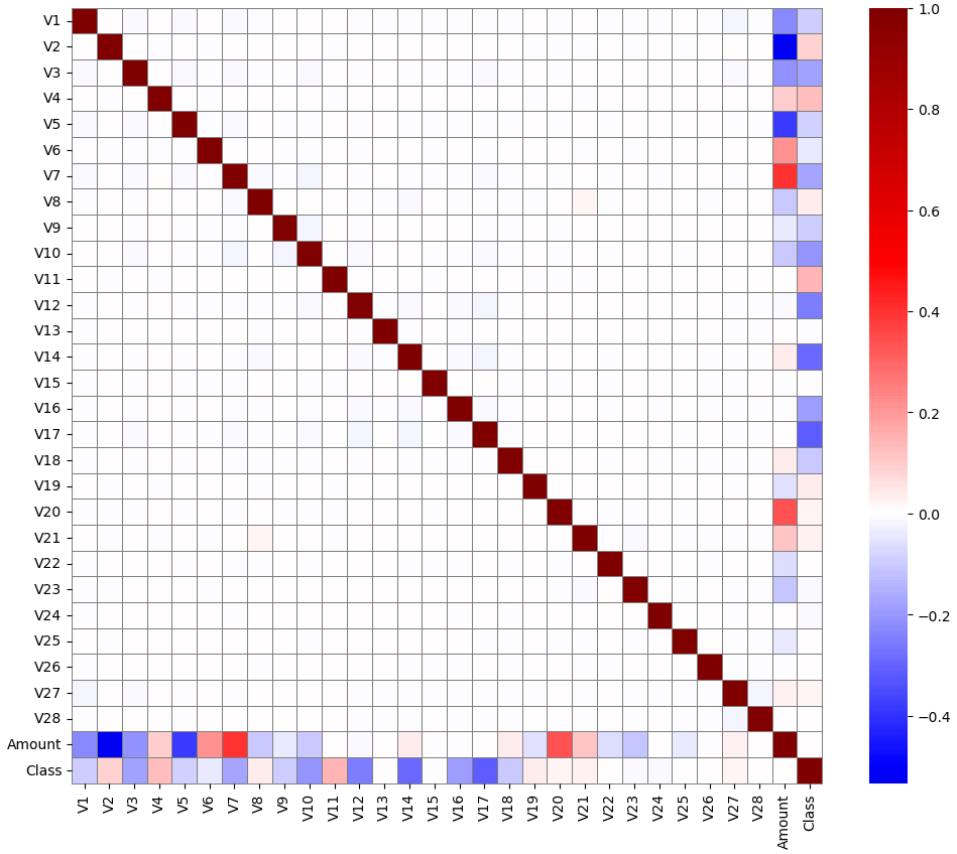
- The dataset contains 492 frauds out of 284,807 transactions. It is highly imbalanced, the positive class (**Fraud**) accounts for only **0.172%** of all transactions.
- Due to confidentiality issues, features **V1, V2,..., V28** are the principal components obtained by applying PCA (rather than the original information).
- **Time** contains the seconds elapsed between each transaction and the first transaction in the dataset. **Amount** is the transaction amount. **Class** is the response variable (**Target**), and it takes value 1 in case of Fraud and 0 otherwise.

(2) Data wrangling

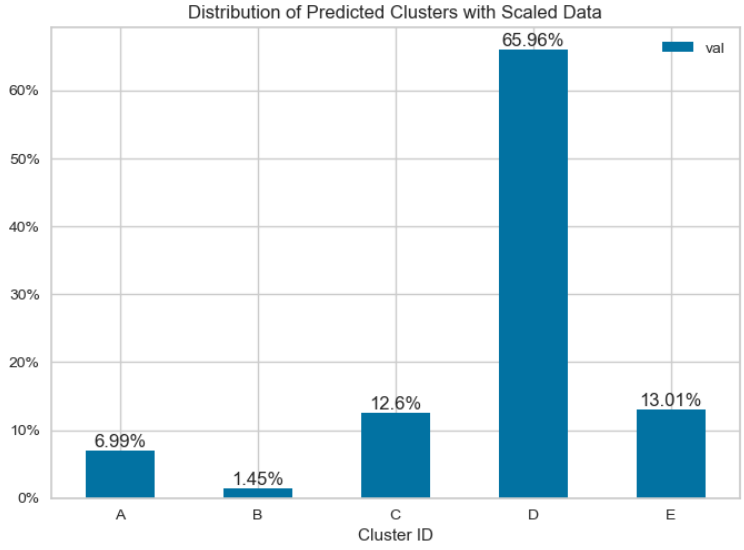
- Target: the **Class** column, binary, minority class is 1 (Fraud).
- Features: **V1, V2,..., V28**, and **Amount**, 29 features in total**.**
- Missing value: all variables contained in the dataset are numerical and there are no missing values.
- Duplicated rows: 1854 rows are duplicated. However, the duplicated rows account for only 0.65% of the total rows. Thus, I simply delete the duplicated ones.

2.2 Exploratory Data Analysis

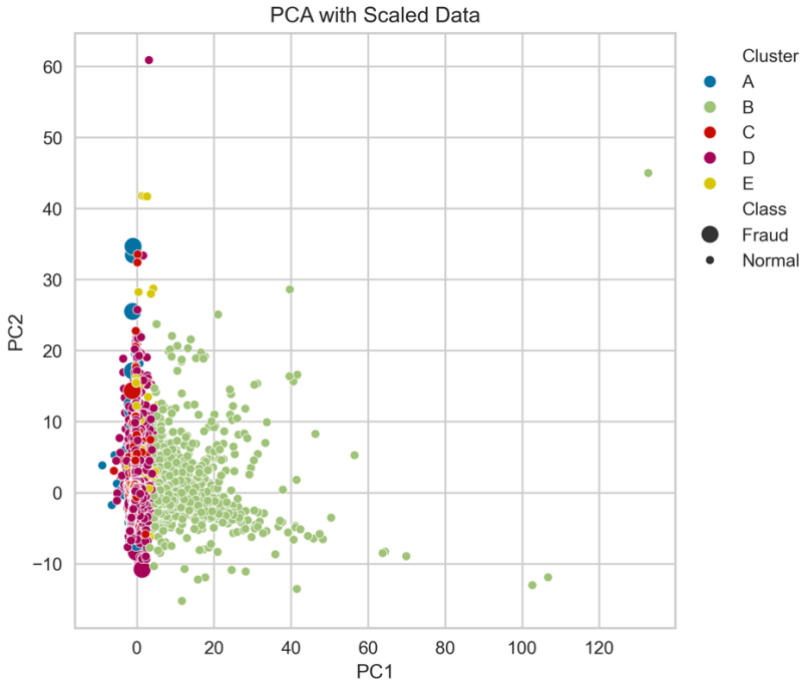
(1) Feature Correlation Heatmap: **V1** to **V28** are not correlated with each other because they are principal components obtained with PCA. **Class**/**Amount** has some positive or negative correlations with several **V** features, while **Class** and **Amount** are not correlated.



(2) K-Means Clustering: The 29 features are used to perform clustering. The result of the Elbow Method suggests that **K=5** be selected. We can see that the cluster D accounts for more than 65% of the data, following by clusters E and C.



(3) Visualize Clustering Results with PCA: The large dots denote the **Fraud** Class. Although the number is small, we can still see some large dots in clusters A, C, and D.



2.3 Modeling Part I : Classification (Supervised)

The classification models are built with labeled data (including the **Class** column for training). I run each model using the original data and resampled data, separately. For resampling, I used random undersampling and oversampling. Oversampling gives much better prediction skills than undersampling.

(1) Logistic Regression (LR) : Estimating the parameters of a logistic model (the coefficients in the linear combination). The F1 score of the LR model is 0.69 without resampling and 0.11 with resampling.
(2) XGBoost: An optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. The F1 score of the XGBoost model is 0.85 regardless of resampling.
(3) LightGBM: A gradient boosting framework that uses tree-based learning algorithms. accuracy. The F1 score of the LGBM model is 0.56 without resampling and 0.84 with resampling.

2.4 Modeling Part II: Anomaly Detection (Unsupervised)

The unsupervised models are built with unlabeled data (without the **Class** column for training). Outlier detection is also known as unsupervised anomaly detection. The scikit-learn project provides a set of machine learning tools that can be used both for novelty and outlier detection. The following two models have very low F1 scores but high Recall.

(1) Isolation Forest: 'Isolates' observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature. The Recall of the Isolation model is 0.80.
(2) Local Outlier Factor (LOF): Computes a score (called local outlier factor) reflecting the degree of abnormality of the observations. It measures the local density deviation of a given data point with respect to its neighbors. The Recall of the LOF model is 0.84.

3. Findings

3.1 Modeling Results Comparison: Prediction Scores

The Table below summarizes the Precision, Recall, and F1 score of Label 1 class (Fraud) for each model. Precision is about minimizing the number of FALSE Positives, while Recall is about minimizing the number of FALSE Negatives. F1 score combines the precision and recall scores of a model.

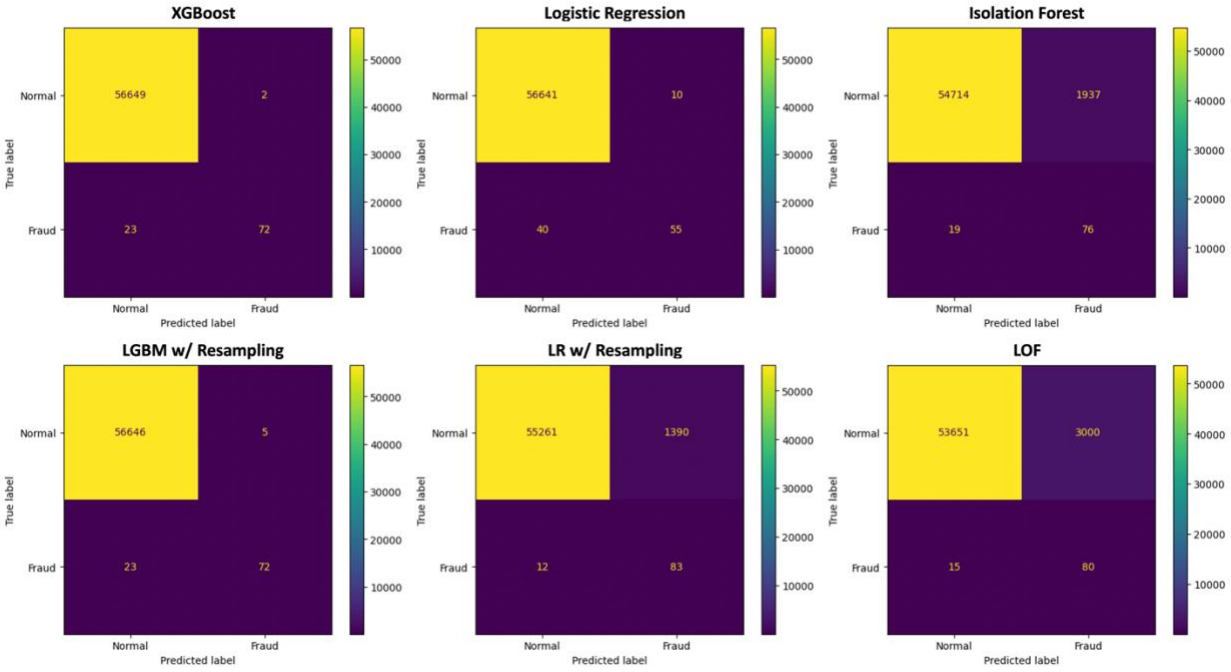
- **Logistic Regression**: Have a high Precision and a fair Recall. After applying resampling, F1 score and Precision become much worse, but Recall becomes very high.
- **XGBoost**: No matter with or without resampling, it has the highest Precision of 0.97 and the highest F1 score of 0.85. Its Recall is also very good (0.76).
- **LightGBM**: After applying resampling, the result becomes much better, with higher Precision, Recall and F1 score, almost the same as XGBoost.
- **Isolation Forest and LOF**: The two unsupervised models give similar results, both having a very high Recall but a very low Precision. Comparing the two, the Isolation Forest is very computationally efficient, while the LOF's computation is much slower.

	Precision	Recall	F1 Score
Logistic Regression	0.85	0.58	0.69
LR w/ Resampling	0.06	0.87	0.11
XGBoost	0.97	0.76	0.85
XGB w/ Resampling	0.96	0.77	0.85
LightGBM	0.71	0.46	0.56
LGBM w/ Resampling	0.94	0.76	0.84
Isolation Forest	0.04	0.80	0.07
Local Outlier Factor	0.03	0.84	0.05

3.2 Confusion Matrix

The figure below shows the Confusion Matrix of each model. In a confusion matrix, the lower right corner and the upper left corner are the number of TRUE Positives and TRUE Negatives, respectively; the upper right corner is the number of FALSE Positives, and the lower left corner is the number of FALSE Negatives.

- **XGBoost** and **LightGBM with Resampling** are similar (Left Two), with very few FALSE Positives. The number of FALSE Negatives is also relatively small.
- For **Logistic Regression** (Middle Two), resampling significantly reduces FALSE Negatives but remarkably increases FALSE Positives (from 10 to 1390).
- Both **Isolation Forest** and **LOF** have a very large number of FALSE Positives and a small number of FALSE Negatives.



4.Conclusions and Discussion

(1) Best model for fraud detection

- Considering the combination of Recall and Precision and computational efficiency, **XGBoost** is the **best model**. No need for resampling, having the highest F1 score and Precision and also high Recall, and its computing speed is fast.
- LGBM with Resampling has similar high scores but is considerably slower to compute.
- If only **Recall** is considered (i.e. focusing on NOT treating "Fraud" as a normal transaction), Logistic Regression with Resampling and two unsupervised models (Isolation Forest and LOF) have very high Recall scores.

(2) Practical implications

- In fraud detection, identifying a fraudulent transaction as a normal one can result in significant financial losses, which means the cost of False Negatives is very high. In this case, **Recall** is a better metric. Therefore, those models with high Recall may also be good choices.

- On the other hand, a low **Precision** score indicates a large number of False Positives, meaning that many normal transactions are identified as fraudulent. This situation can increase complaints from credit card users and cause credit card companies to lose customers.

5. Consulted Resources

[1] Combat Imbalanced Dataset: http://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/

[2] XGBoost: https://xgboost.readthedocs.io/en/stable/

[3] LightGBM: https://lightgbm.readthedocs.io/en/stable/

[4] Novelty and Outlier Detection: https://scikit-learn.org/stable/modules/outlier_detection.html