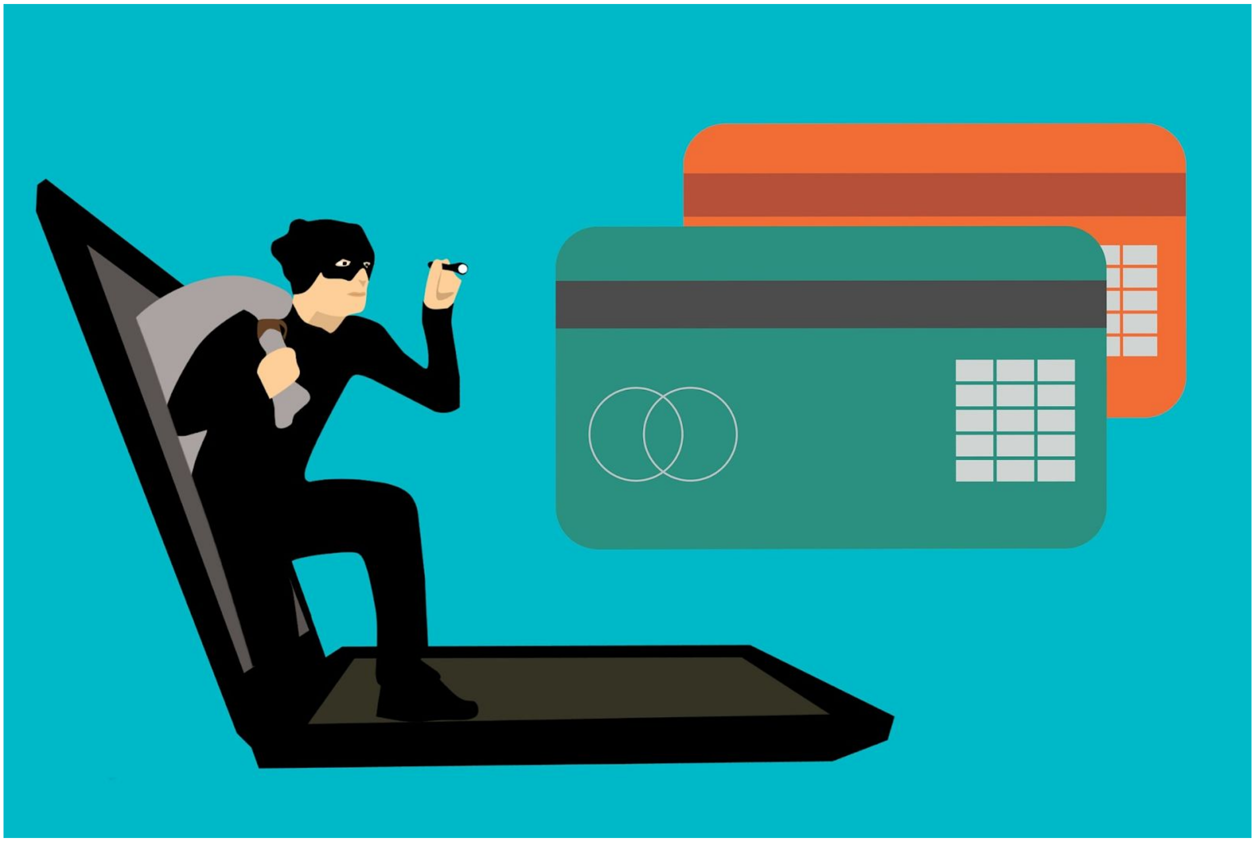
Credit Card Fraud Detection

Statistical Learning Project

MATH 569

Prof. Lulu Kang



Team Members

Aravind Senthil Kumar
A20449302

Amogh Sondur
A20449485

Table of Contents

1 Overview

Every year, millions of Americans fall victim to fraud that costs the national economy billions of dollars. If you're a victim, it can wreak havoc on your personal finances. Luckily, many financial institutions have measures in place to help protect you from credit fraud. Experian also offers tools you can use to protect yourself from identity theft.

Credit card fraud is when someone uses your credit card or credit account to make a purchase you didn't authorize.

Types of Credit Card Fraud
Fraudsters are creative people, and they've come up with many ways to pilfer your personal information and destroy your hard-earned good credit, including:

- **Stealing a credit card**: You look away for a moment and your wallet disappears off the store counter where you placed it while making a purchase. Or, you forget to zip up your purse in a crowd and someone slips your wallet from your bag. When your credit card is stolen, you should immediately notify the card issuer.
- **Using a lost or found credit card**: Accidents happen and it's possible a card falls out of your pocket in a parking lot. Someone who finds the card could try to use it. Always report lost cards to the credit card issuer immediately to reduce the chance of someone doing damage to your balance.
- **Account takeover**: A fraudster can use personal information such as your home address, mother's maiden name, etc., to contact your credit card company or bank, pretend they're you, claim your card has been lost or stolen, or that you've changed addresses, and get the card issuer to send them a new card. Some issuers allow you to have a verbal password when calling them, and this could be a good way to help prevent this type of fraud.

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.

1.1 Data

The datasets contain transactions made by credit cards in September 2013 by european cardholders. 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. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example dependent cost sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

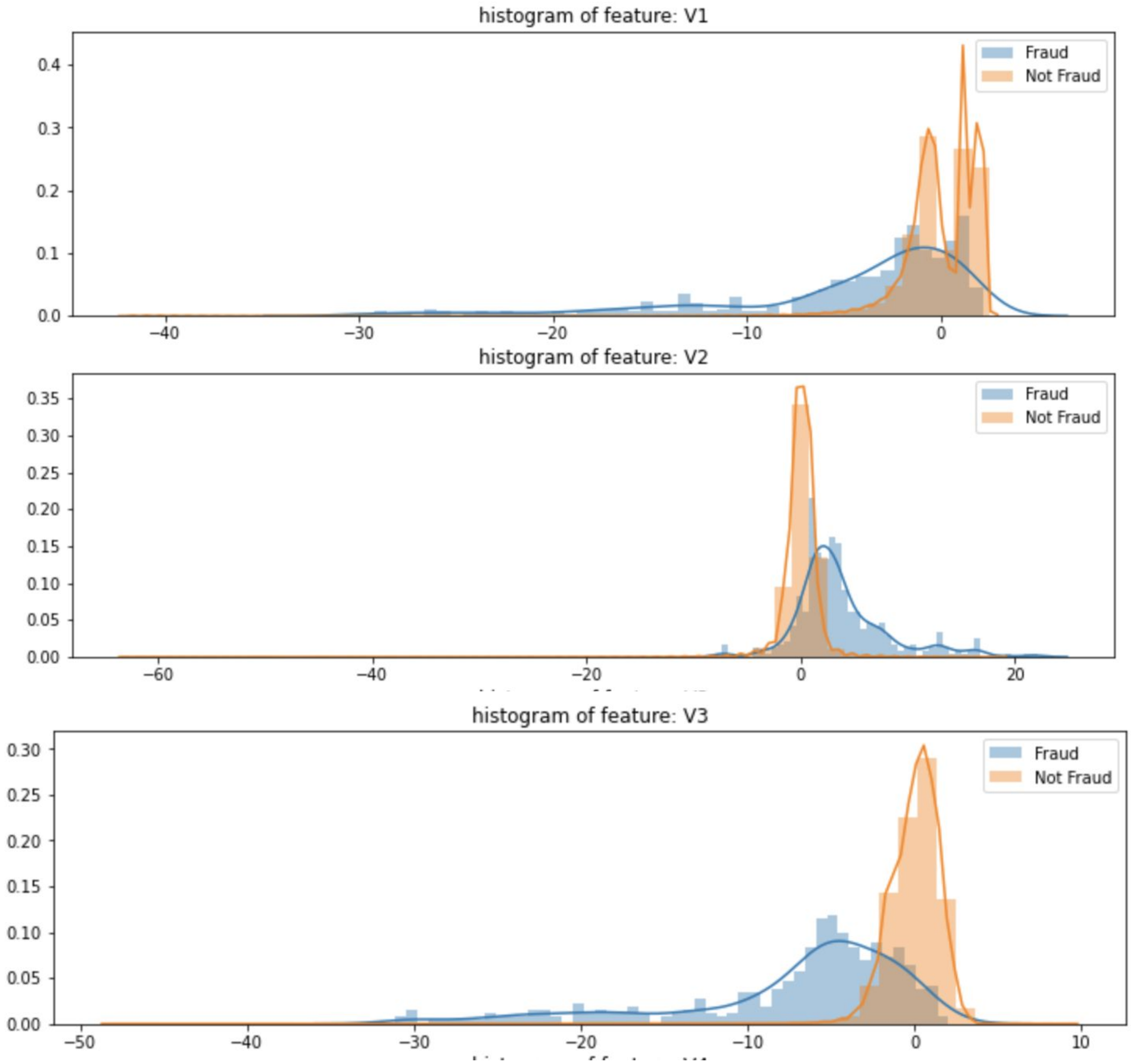
1.2 Acknowledgements

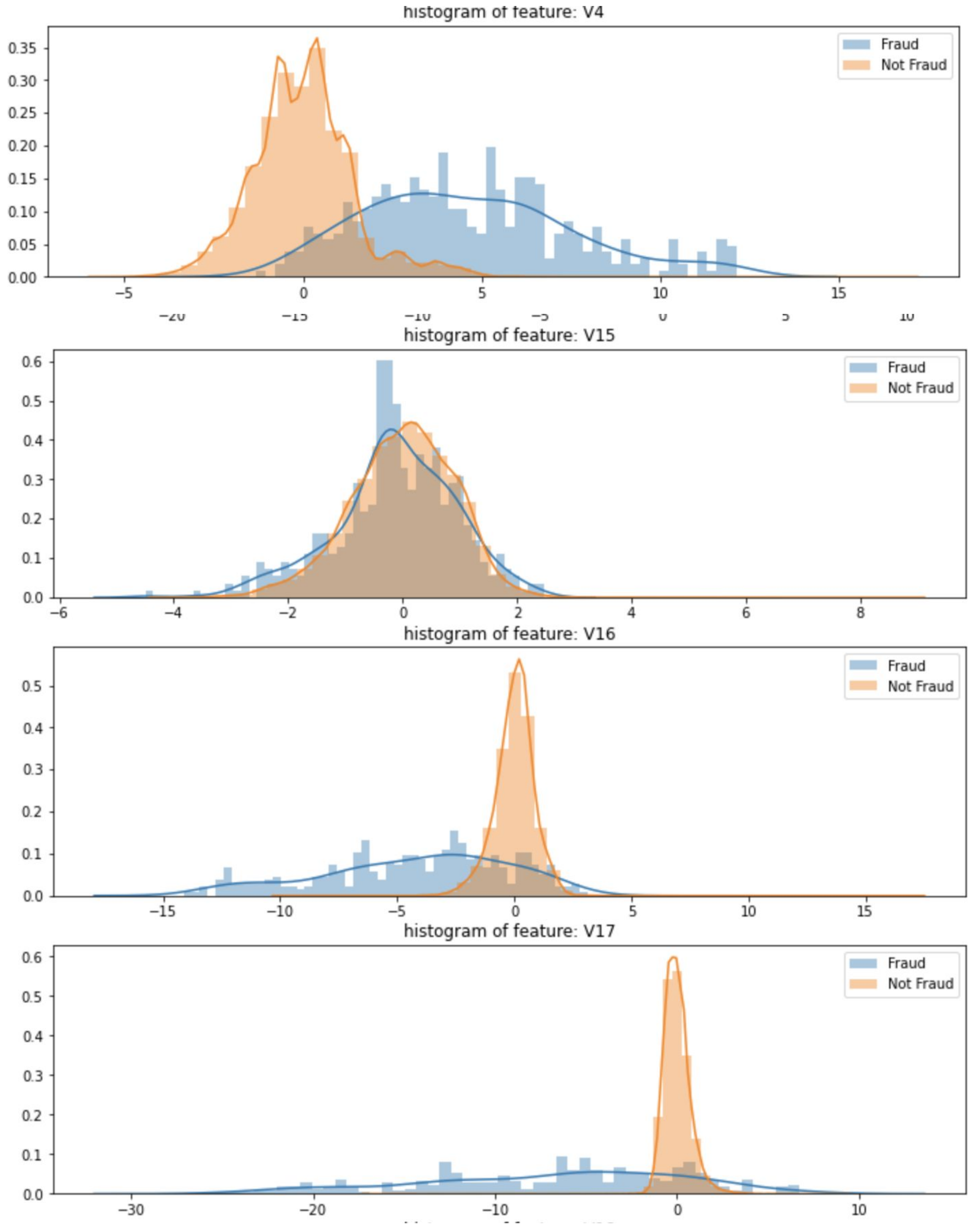
The dataset has been collected and analysed during a research collaboration of Worldline and the Machine Learning Group (http://mlg.ulb.ac.be) of ULB (Université Libre de Bruxelles) on big data mining and fraud detection. More details on current and past projects on related topics are available on https://www.researchgate.net/project/Fraud-detection-5 and the page of the DefeatFraud project

2 Data Preprocessing

2.1 Exploratory Data Analysis

For all the variables, a density plot is created to understand the distribution of each variable based on the output class. Few of them are shown below





As we can clearly see from the histograms, the distribution of features vary a lot based in the transaction type

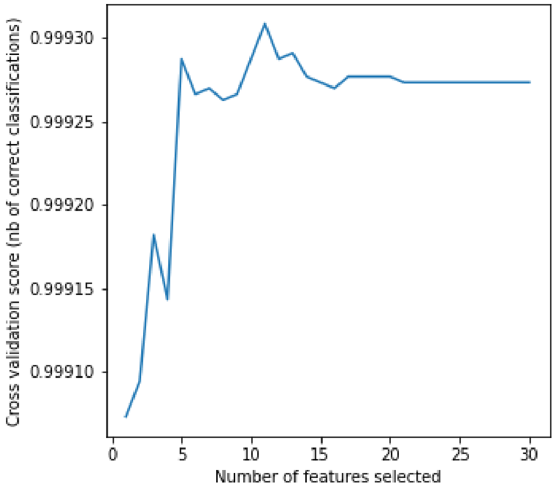
2.2 Feature Selection

Feature Selection is one of the core concepts in machine learning which hugely impacts the performance of your model. The data features that you use to train your machine learning models have a huge influence on the performance you can achieve. In our dataset, we have 28 features. It is important to select import features to improve accuracy, reduce overfitting, and thus reduce the training time of our model.

- Reduces Overfitting: Less redundant data means less opportunity to make decisions based on noise.
- Improves Accuracy: Less misleading data means modeling accuracy improves.
- Reduces Training Time: fewer data points reduce algorithm complexity and algorithms train faster.

LDA: Linear Discriminant Analysis seeks to best separate (or discriminate) the samples in the training dataset by their class value. Specifically, the model seeks to find a linear combination of input variables that achieves the maximum separation for samples between classes (class centroids or means) and the minimum separation of samples within each class.

RFE: Recursive Feature Elimination (RFE) is a brute force approach to feature selection. The RFE method from sklearn can be used on any estimator with a .fit method that once fitted will produce a coef_ or feature_importances_ attribute.[1] It works by removing the feature with the least importance from the data and then reevaluates the feature importance, repeating the process until it is told to stop.

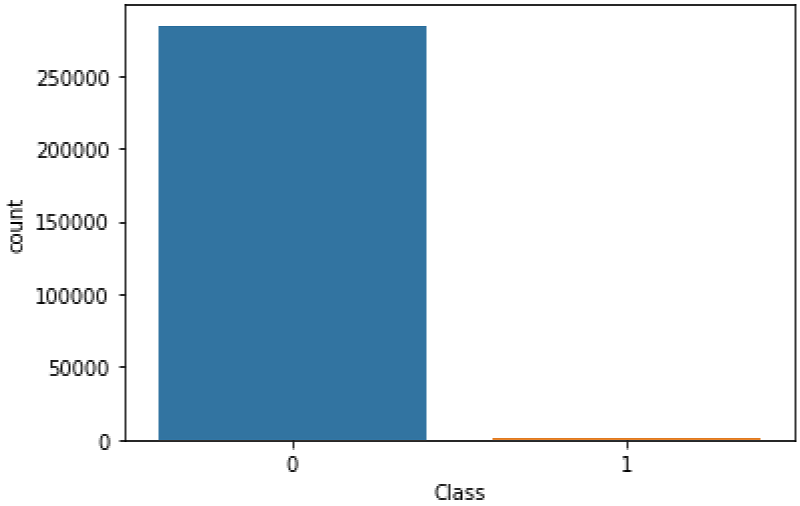


But here, since the data is already provided as principal components, we will not be implementing feature reductions

2.3 Stratified Sampling

Data sampling provides a collection of techniques that transform a training dataset in order to balance or better balance the class distribution. Once balanced, standard machine learning algorithms can be trained directly on the transformed dataset without any modification. This allows the challenge of imbalanced classification, even with severely imbalanced class distributions, to be addressed with a data preparation method.

Transforming Training Dataset The ratio of **0.17%** between the Fraud and Normal classes is showing strongly unbalanced data in favor of the Normal class. Resampling is used to transform the Training dataset, in which we will oversample the Normal class, and make the Dataset balanced out between the Classes, this prevents the fitting model from overfitting on the majority class.



Overall Normal vs Fraud - 0: 284315, 1: 492

Before performing SMOTE, let’s split the data into train and test for assessing the model performance. We have implemented stratified sampling to make sure the positive class distribution percentage remains the same in train and test

1. Class distribution in train data : Fraud - 394 and Not Fraud - 227,451 with a positive (Fraud) class percentage of 0.1732%
2. Class distribution in train data : Fraud - 98 and Not Fraud - 56,864 with a positive (Fraud) class percentage of 0.1723%

2.4 Scaling

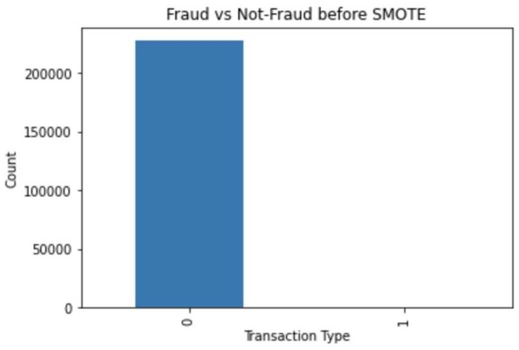
StandardScaler() is used for scaling the train data to normalize and standardize to scale the features. Same scale is used to normalize and standardize the test data

2.5 SMOTE

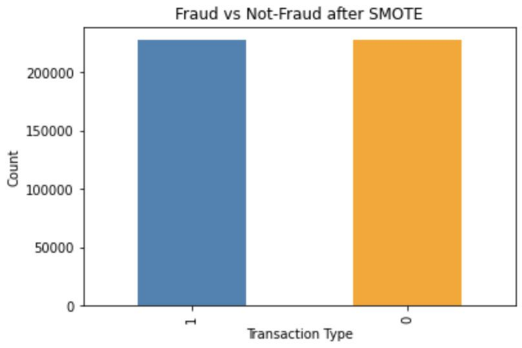
Perhaps the most widely used approach to synthesizing new examples is called the Synthetic Minority Oversampling Technique or SMOTE for short.

SMOTE works by selecting examples that are close in the feature space, drawing a line between the examples in the feature space, and drawing a new sample at a point along that line. Specifically, a random example from the minority class is first chosen. Then k of the nearest neighbors for that example is found (typically k=5). A randomly selected neighbor is chosen and a synthetic example is created at a randomly selected point between the two examples in feature space.

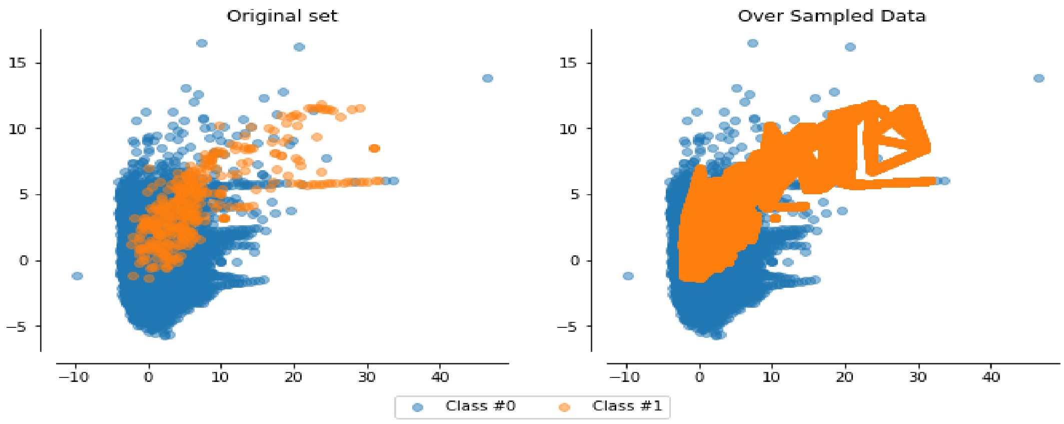
The graphs displayed below are for train dataset after applying SMOTE



Normal - 227,451 and Fraud - 394



Normal - 227,451 and Fraud - 227,451



3 Modeling

The response variable is a categorical variable. Gives the type of transaction - Fraudulent or not. Hence, we have implemented 8 different classification models

For all the models, the dataset size is as follows:

- Train data : records - 227,845, features - 29
- Test data : records - 56,962, features - 29
- Evaluation Metric : Since we are dealing with class imbalance, we are assessing a model based on Precision, Recall, F1 Score rather than accuracy

3.1 Logistic regression:

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression(or logit regression) is estimating the parameters of a logistic model (a form of binary regression). Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail which is represented by an indicator variable, where the two values are labeled "0" and "1"

Model:

1. Hyperparameter tuning : Grid Search
2. Cross-Validation : Stratified k fold cross-validation with k = 10

Model Performance:

Train Classification Report:

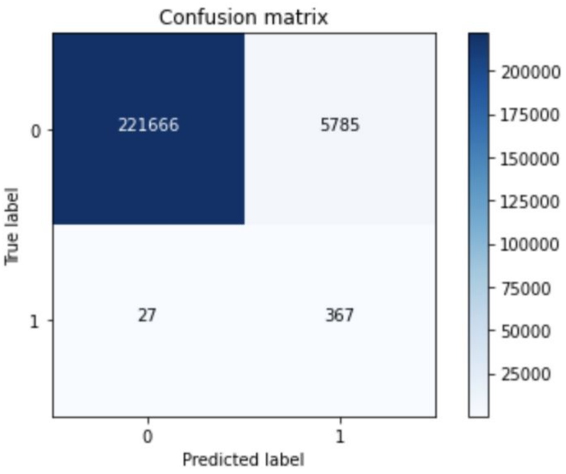
Class	Precision	Recall	F1 Score	Support
0	1	0.97	0.99	227451
1	0.06	0.93	0.11	394

Test Classification Report:

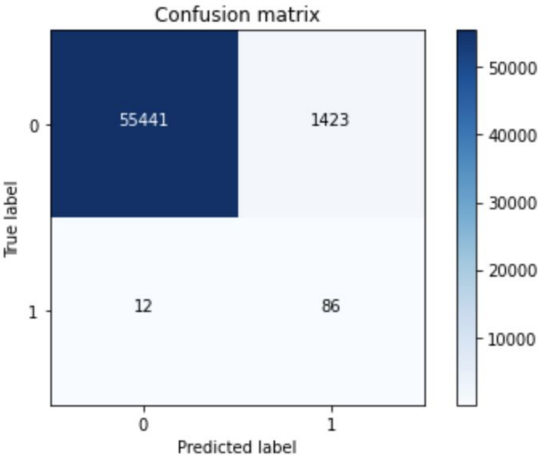
Class	Precision	Recall	F1 Score	Support
0	1	0.97	0.99	227451
1	0.06	0.88	0.11	394

Confusion Matrix:

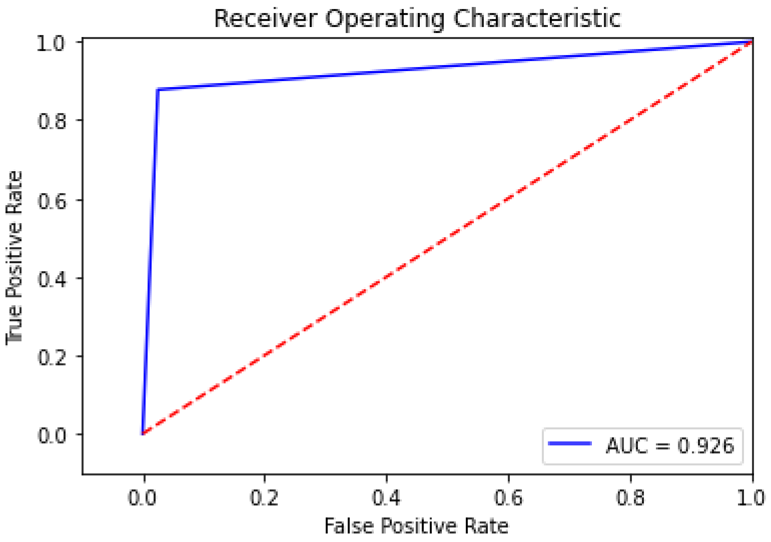
Train



Test



Test ROC curve:



Though the AUC value is high, as we can clearly see from the classification report the precision and F1 score are very low. This means, we will be ending up with a lot of false positives. Let’s consider this as the baseline model and try to improve the classifier’s performance using difference classification techniques

3.2 Random Forest

Random forests is an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees Random decision forests correct for decision trees habit of overfitting to their training set Random forests generally outperform decision trees.

Model:

1. Hyperparameter tuning : Randomized Grid Search
2. Cross-Validation : 3 fold cv across 100 different combinations of parameters

Model Performance:

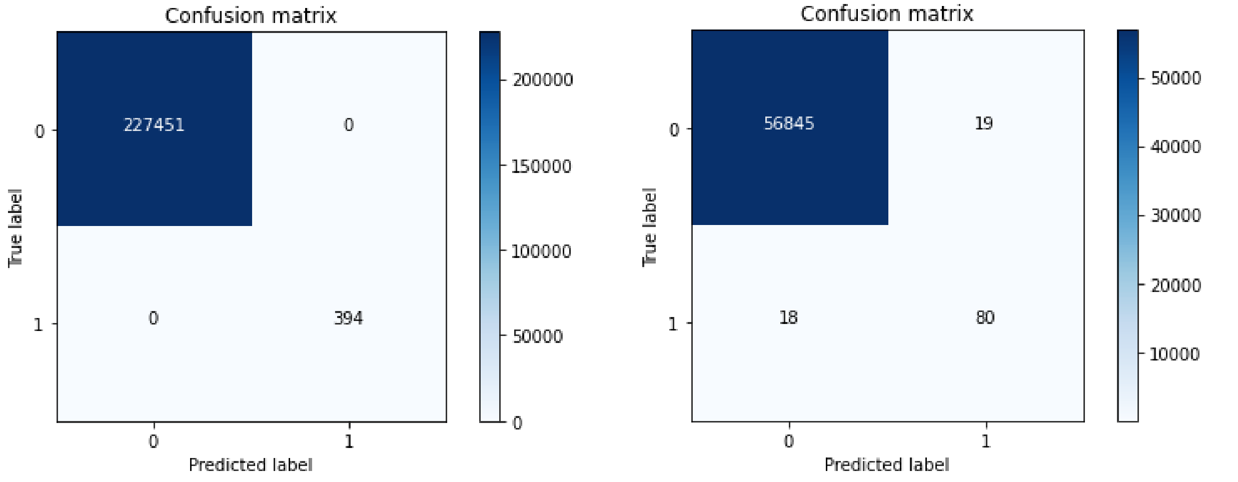
Train Classification Report:

Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1.0	1.0	1.0	227451
1	1.0	1.0	1.0	394

Test Classification Report:

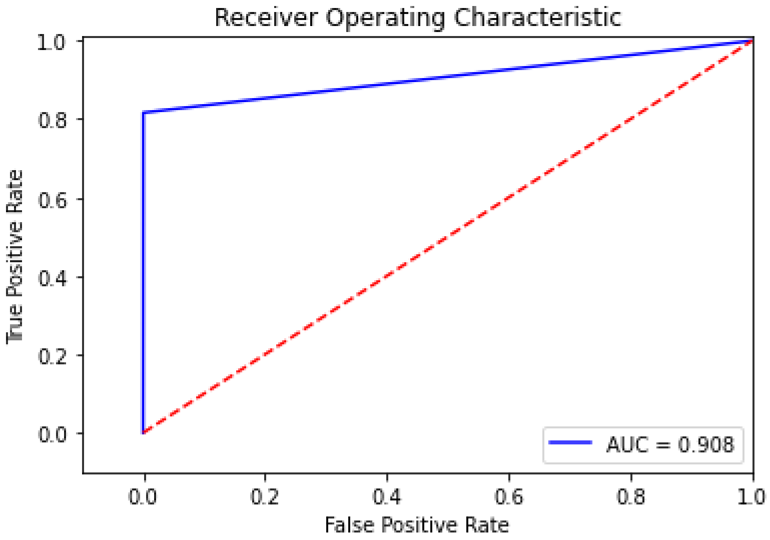
Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1.0	1.0	1.0	56864
1	0.81	0.82	0.81	98

Confusion Matrix:



Train **Test**

Test ROC curve:



Though the AUC value is slightly less than the baseline model, we can clearly see from the classification report that the precision and F1 score has increased a lot. This means the model will output very less False Positives.

3.3 SVM

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well.

Model architecture:

1. Hyperparameter tuning : Randomized Grid Search - scoring based on roc_auc
2. Cross-Validation : 10 fold repeated stratified cv

Model Performance:

Train Classification Report:

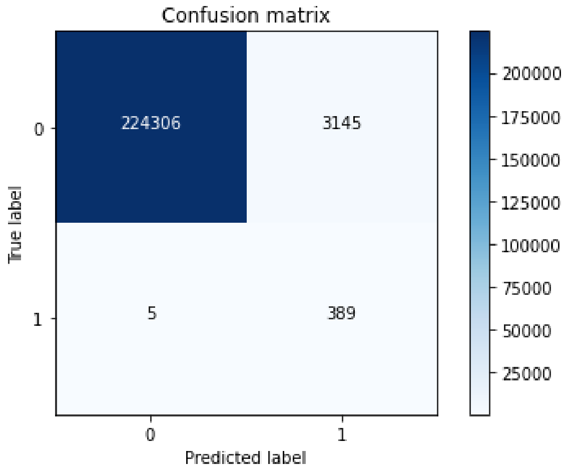
Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1.0	0..99	0.99	227451
1	0.11	0.99	0.20	394

Test Classification Report:

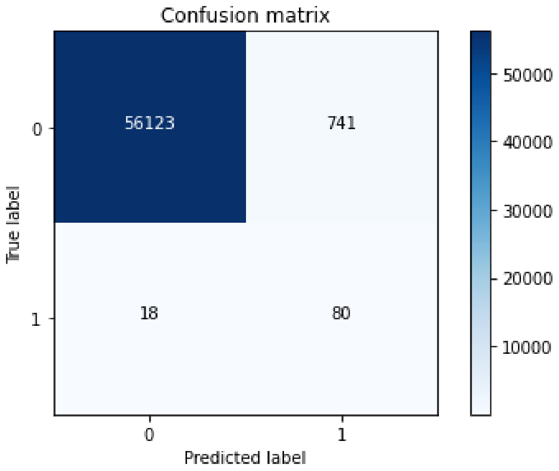
Class	Precision	Recall	F1 Score	Support
0	1.0	0.99	0.99	56864
1	0.10	0.82	0.17	98

Confusion Matrix:

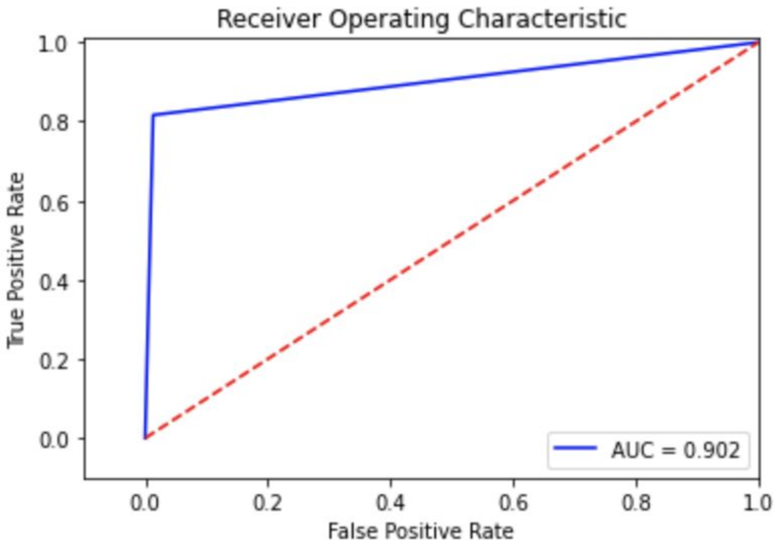
Train



Test



Test ROC curve:



Though the AUC value is slightly less than the baseline model, we can clearly see from the classification report that the precision and F1 score has increased a lot. This means the model will output very less False Positives.

3.4 LDA

Linear discriminant analysis (LDA), normal discriminant analysis (NDA), or discriminant function analysis is a generalization of Fisher's linear discriminant, a method used in statistics and other fields, to find a linear combination of features that characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier, or, more commonly, for dimensionality reduction before later classification.

Model architecture:

1. Hyperparameter tuning : Randomized Grid Search - scoring based on roc_auc
2. Cross-Validation : 10 fold repeated stratified cv

Model Performance:

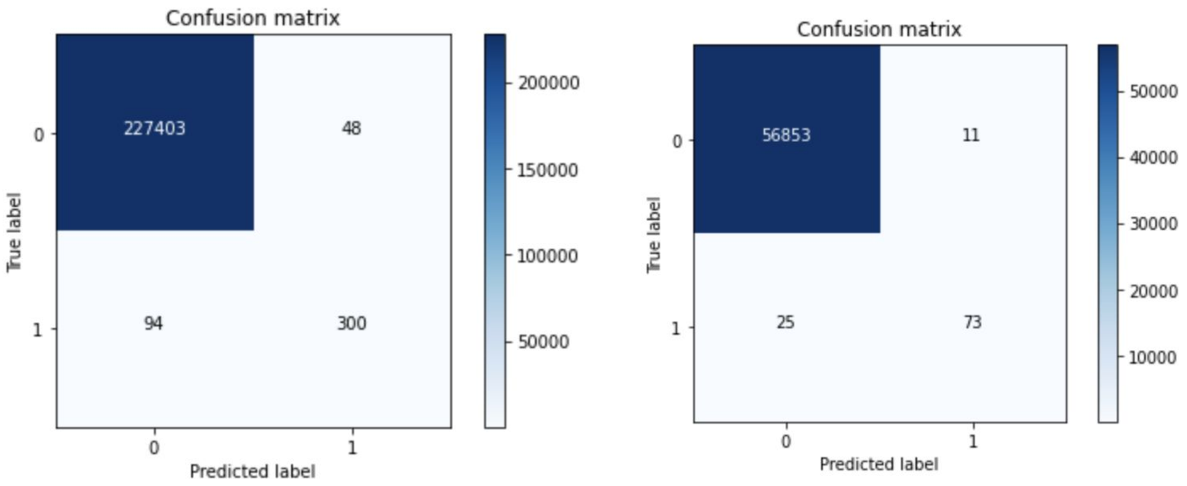
Train Classification Report:

Class	Precision	Recall	F1 Score	Support
0	1.0	1.0	1.0	227451
1	0.86	0.76	0.81	394

Test Classification Report:

Class	Precision	Recall	F1 Score	Support
0	1.0	1.0	1.0	56864
1	0.87	0.74	0.80	98

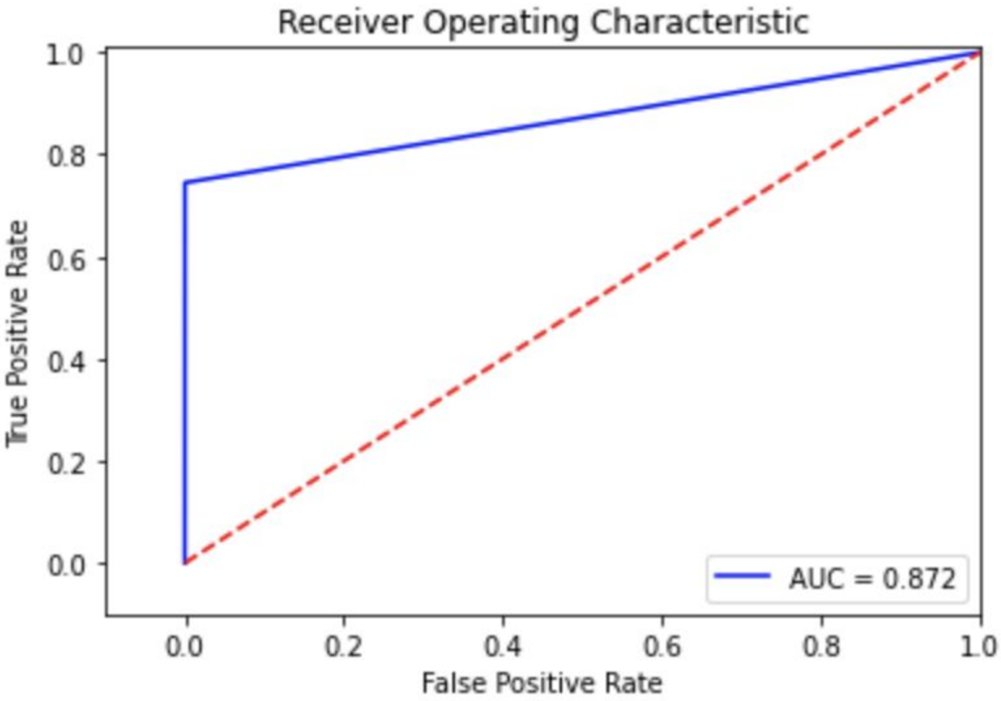
Confusion Matrix:



Train

Test

Test ROC curve:



The performance of LDA is better than the baseline model but not even close to random forest

3.5 QDA

QDA is a variant of LDA in which an individual covariance matrix is estimated for every class of observations. QDA is particularly useful if there is prior knowledge that individual classes exhibit distinct covariances. A disadvantage of QDA is that it cannot be used as a dimensionality reduction technique.

Model architecture:

1. Hyperparameter tuning : NA
2. Cross-Validation : NA

Model Performance:

Train Classification Report:

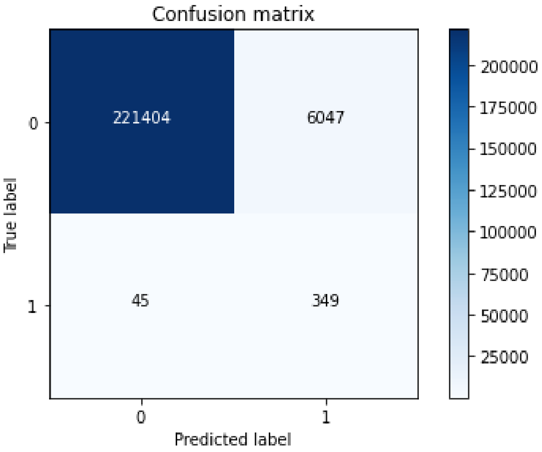
Class	Precision	Recall	F1 Score	Support
0	1.0	0.97	0.99	227451
1	0.05	0.89	0.10	394

Test Classification Report:

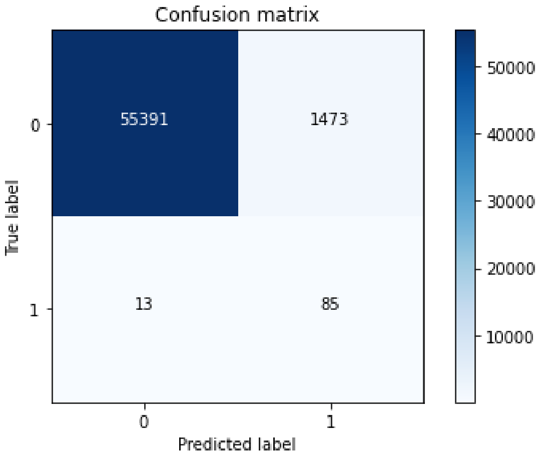
Class	Precision	Recall	F1 Score	Support
0	1.0	0.97	0.99	56864
1	0.05	0.87	0.10	98

Confusion Matrix:

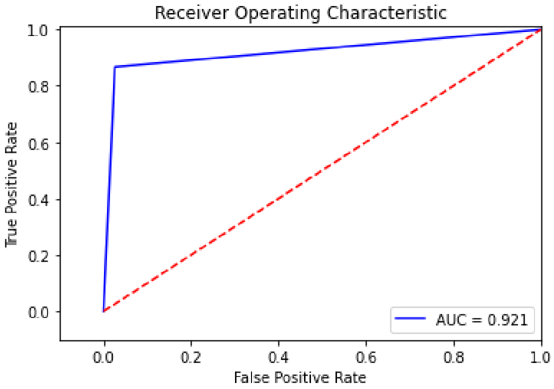
Train



Test



Test ROC:



QDA performs very similar to logistic regression

3.6 XGBoost

XGBoost is a special implementation of the Gradient Boosting and XGBoost stands for Extreme Gradient Boosting, XGBoost uses more accurate approximations by employing second-order gradients and advanced regularization.

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y_i}^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t)$$

The Objective function is solved by using Second-order Taylor polynomial approximation and Simply put XgBoost tries to find the optimal output value for a tree ft in an iteration t that is added to minimize the above loss function across all data points,

Model architecture:

1. Hyperparameter tuning : Adjusted learning rate, max depth, number of estimators, number of leaves, boosting type and lambda
2. Cross-Validation : NA

Model Performance:

Train Classification Report:

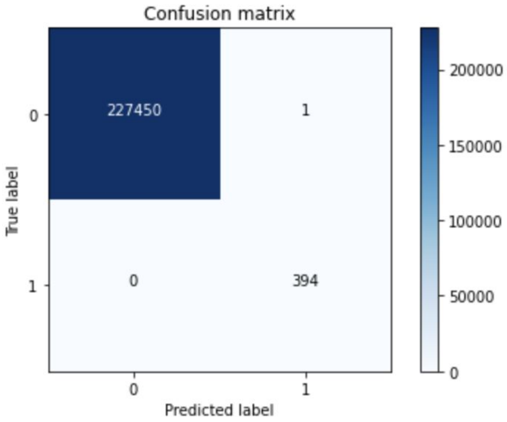
Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1.0	1.0	1.0	227451
1	1.0	1.0	1.0	394

Test Classification Report:

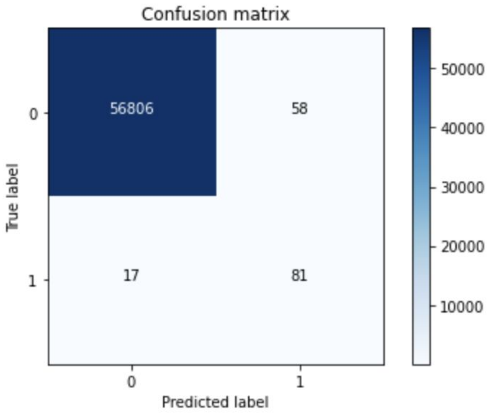
Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1.0	1.0	1.0	56864
1	0.58	0.83	0.68	98

Confusion Matrix:

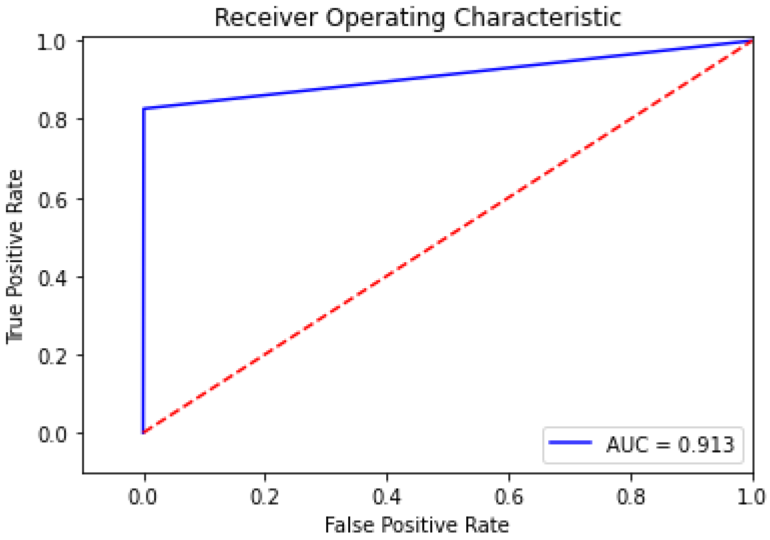
Train



Test



ROC Curve:



XGBoost performs similarly to Random forest. Though it captures 1 more True positive than randomforest, it also misclassified more normal transactions as fraud.

3.7 catBoost

CatBoost is a recently open-source machine learning algorithm from Yandex. It can easily integrate with deep learning frameworks like Google’s TensorFlow and Apple’s Core ML. It can work with diverse data types to help solve a wide range of problems that businesses face today. To top it up, it provides best-in-class accuracy.
It is especially powerful in two ways:

- It yields state-of-the-art results without extensive data training typically required by other machine learning methods, and
- Provides powerful out-of-the-box support for the more descriptive data formats that accompany many business problems.

“CatBoost” name comes from two words “Category” and “Boosting”.

Train Classification Report:

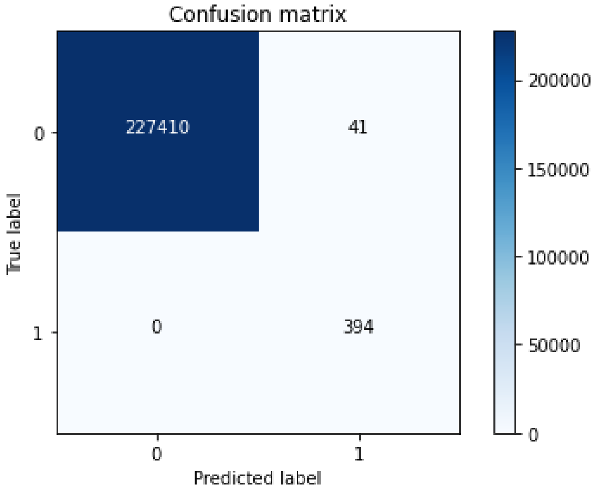
Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1	1	1	227451
1	0.91	1	0.95	394

Test Classification Report:

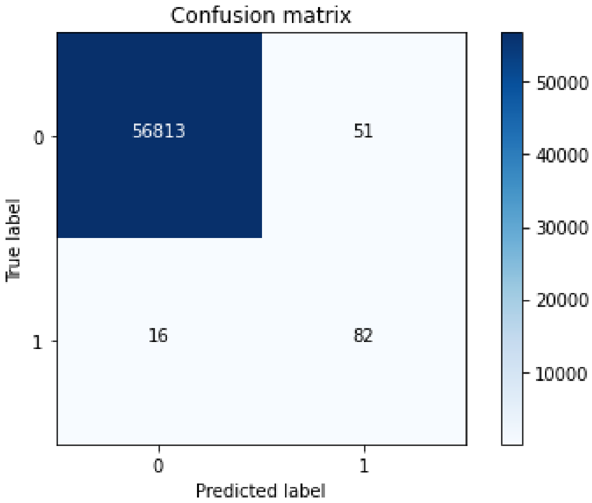
Class	**Precision**	**Recall**	**F1 Score**	**Support**
0	1	1	1	56864
1	0.62	0.84	0.71	98

Confusion Matrix

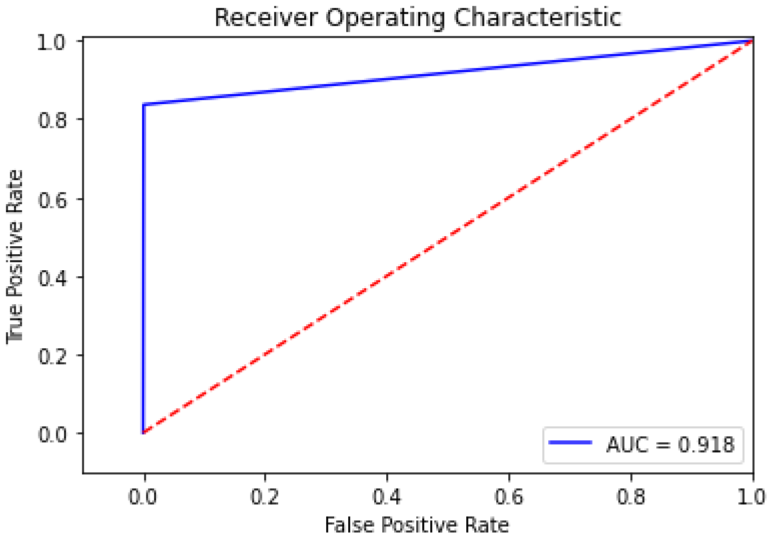
Train



Test



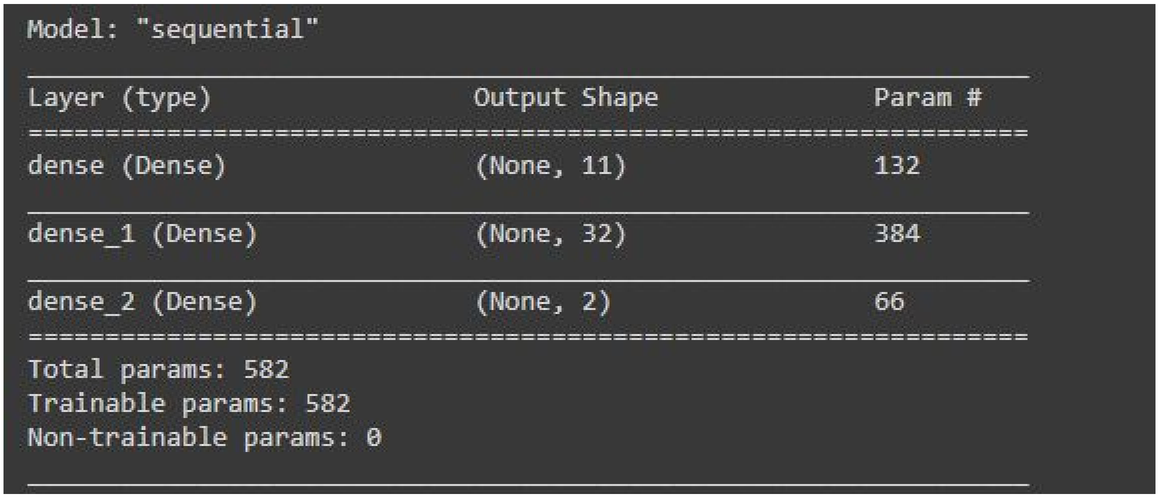
ROC:



The model performs well with the AUC values of 0.918

3.8 Neural Net

Model Architecture:

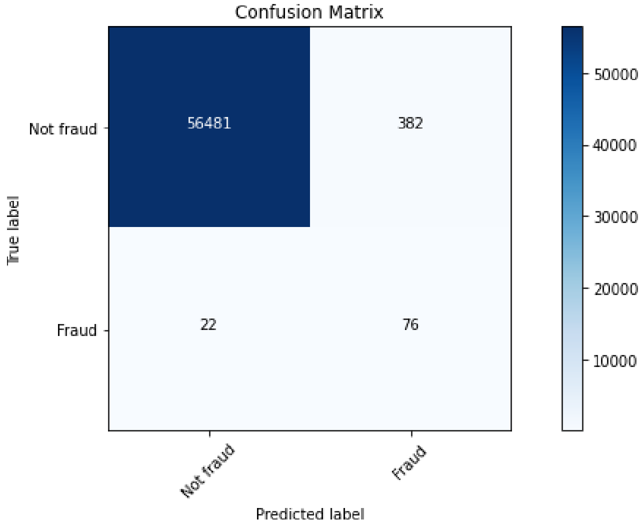


```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 11)                132
_________________________________________________________________
dense_1 (Dense)              (None, 32)                384
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 66
=================================================================
Total params: 582
Trainable params: 582
Non-trainable params: 0
_________________________________________________________________
```

Test Classification Report

Class	Precision	Recall	F1 Score	Support
0	1	0.99	1	56864
1	0.17	0.78	0.27	98

Test Confusion Matrix



Neural net performs marginally better than QDA and logistic Regression but worse than random forest and boosting techniques.

4 Conclusion

Although there are several fraud detection techniques available today, none is able to detect all frauds completely when they are actually happening, they usually detect it after the fraud has been committed. This happens because a very minuscule number of transactions from the total transactions are actually fraudulent in nature. So we need a technology that can detect the fraudulent transaction when it is taking place so that it can be stopped then and there and that too at a minimum cost. So the major task of today is to build an accurate, precise and fast detecting fraud detection system for credit card frauds that can detect not only frauds happening over the internet like phishing and site cloning but also tampering with the credit card itself i.e. it signals an alarm when the tampered credit card is being used. The major drawback of all the techniques is that they are not guaranteed to give the same results in all environments. They give better results with a particular type of dataset and poor or unsatisfactory results with other types. Thus, the results are purely dependent on the dataset type used.

Machine learning techniques like Logistic regression, Random Forest SVM, LDA, QDA, and XGBoost classifiers were used to detect fraud in credit card systems. Sensitivity, Specificity, accuracy and error rate were used to evaluate the performance for the proposed system. From the experiments, the result that has been concluded is that Random Forest and LDA performed significantly better than other models while considering the accuracy, precision and recall.

5. References

1. Andrea Dal Pozzolo, Olivier Caelen, Reid A. Johnson and Gianluca Bontempi. Calibrating Probability with Undersampling for Unbalanced Classification. In Symposium on Computational Intelligence and Data Mining (CIDM), IEEE, 2015
2. Dal Pozzolo, Andrea; Caelen, Olivier; Le Borgne, Yann-Ael; Waterschoot, Serge; Bontempi, Gianluca. Learned lessons in credit card fraud detection from a practitioner perspective, Expert systems with applications,41,10,4915-4928,2014, Pergamon
3. Dal Pozzolo, Andrea; Boracchi, Giacomo; Caelen, Olivier; Alippi, Cesare; Bontempi, Gianluca. Credit card fraud detection: a realistic modeling and a novel learning strategy, IEEE transactions on neural networks and learning systems,29,8,3784-3797,2018,IEEE
4. Dal Pozzolo, Andrea Adaptive Machine learning for credit card fraud detection ULB MLG PhD thesis (supervised by G. Bontempi)
5. Carcillo, Fabrizio; Dal Pozzolo, Andrea; Le Borgne, Yann-Aël; Caelen, Olivier; Mazzer, Yannis; Bontempi, Gianluca. Scarff: a scalable framework for streaming credit card fraud detection with Spark, Information fusion,41, 182-194,2018,Elsevier
6. Carcillo, Fabrizio; Le Borgne, Yann-Aël; Caelen, Olivier; Bontempi, Gianluca. Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization, International Journal of Data Science and Analytics, 5,4,285-300,2018,Springer International Publishing
7. Bertrand Lebichot, Yann-Aël Le Borgne, Liyun He, Frederic Oblé, Gianluca Bontempi Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection, INNSBDDL 2019: Recent Advances in Big Data and Deep Learning, pp 78-88, 2019
8. Fabrizio Carcillo, Yann-Aël Le Borgne, Olivier Caelen, Frederic Oblé, Gianluca Bontempi Combining Unsupervised and Supervised Learning in Credit Card Fraud Detection Information Sciences, 2019
9. https://hyperopt.github.io/hyperopt/?source=post_page
10. https://machinelearningmastery.com/cost-sensitive-svm-for-imbalanced-classification/