Credit Card Fraud Detection Model

Instructor- Ebrahim Nasrabadi

Submitted by- Sayali Walke

Summary

Overview:

This dataset contains credit card transactions in September 2013 by European cardholders.

Purpose:

The goal is to detect fraudulent and non fraudulent transactions.

Objectives:

1] Data Analysis
2] Data Visualization
3] Data Preprocessing
4] Model Implementation
5] Future Improvements
6] Learning Outcomes

Overview of Dataset

- This dataset has 28 numerical input variables as the result of a PCA transformation.
- The other inputs that have not been transformed are 'Time' and 'Amount’.
- The variable 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset.
- The variable ‘Amount' is the transaction Amount.
- The variable 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise.

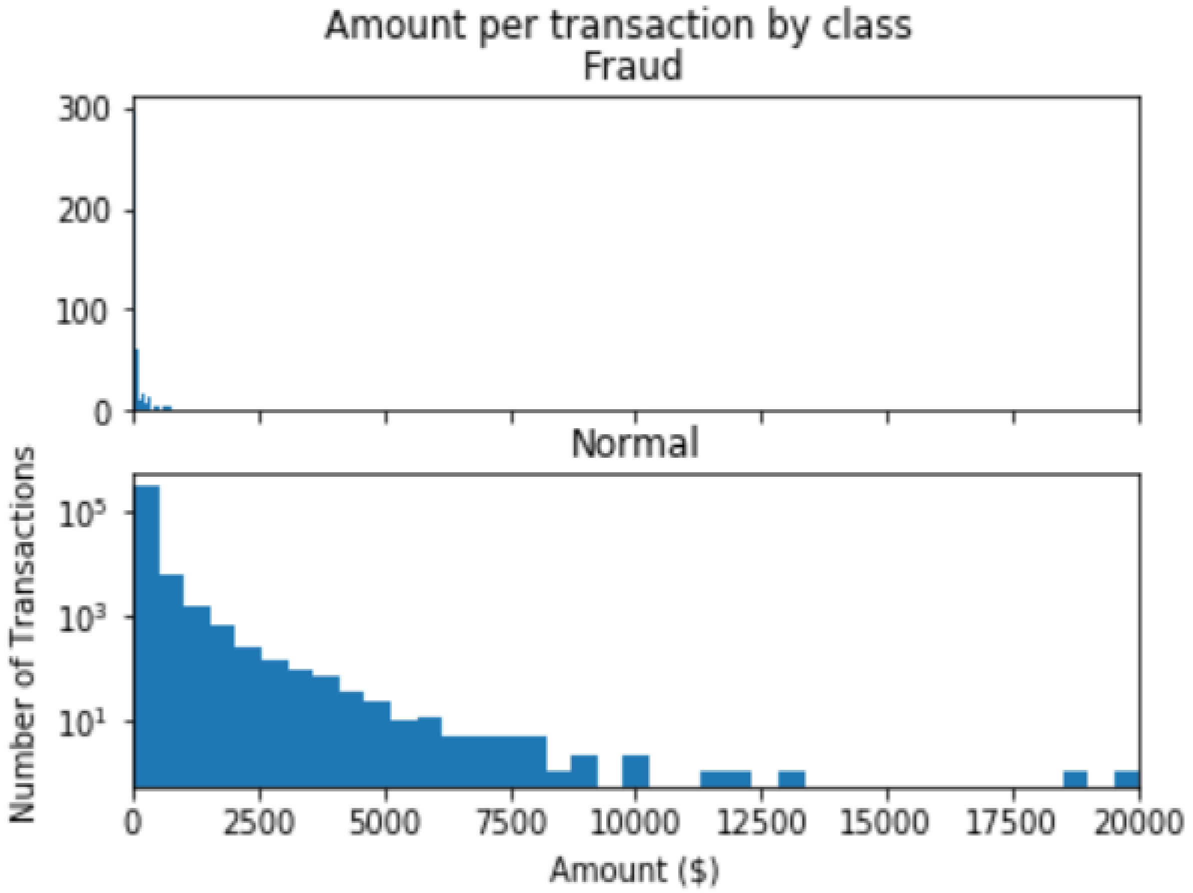
Data Analysis and Data Cleaning

Simple checkpoints:

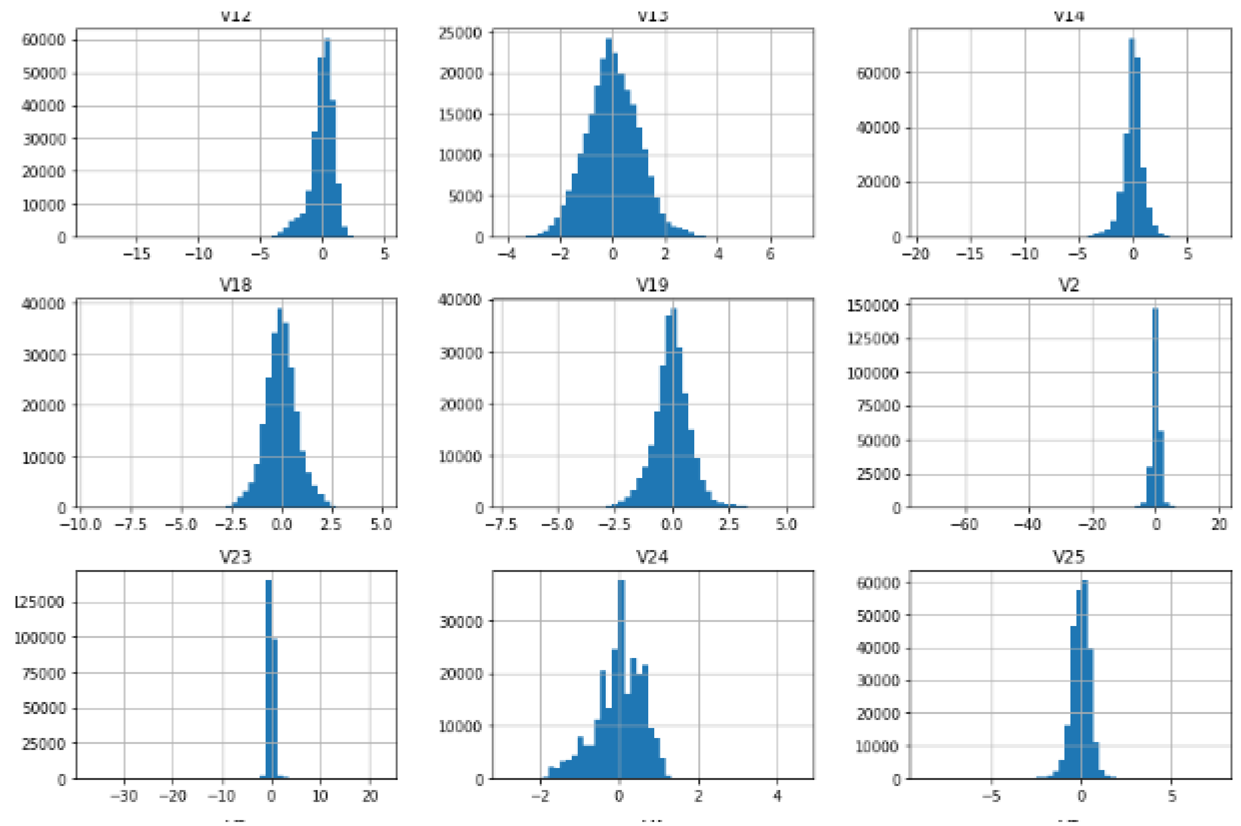
- ➢Missing values- checked for missing values in dataset
- ➢Unique values- checked for unique values of transaction id

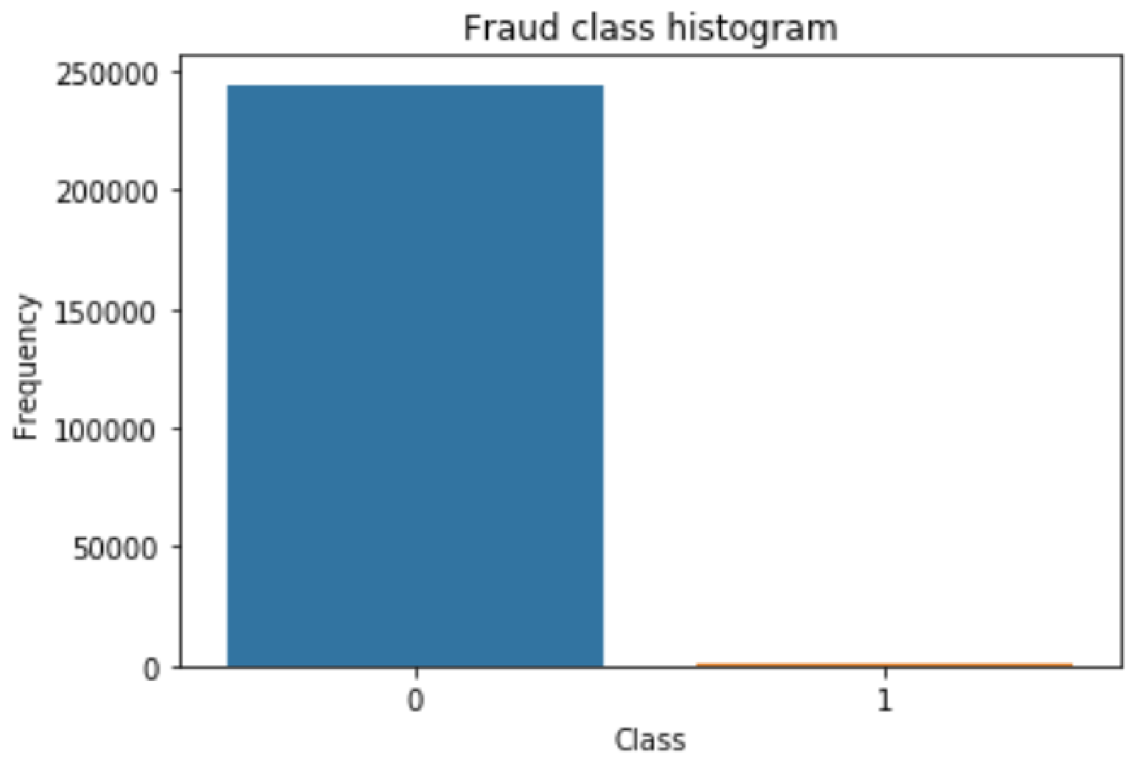
Statistical Analysis:

- ➢There is no metadata about the original features provided, so pre-analysis or feature study could not be done.
- ➢The transaction amount is relatively small.
- ➢The mean of all the amounts made is approximately USD 88.
- ➢The maximum transaction amount is USD 25691.



Data Visualization





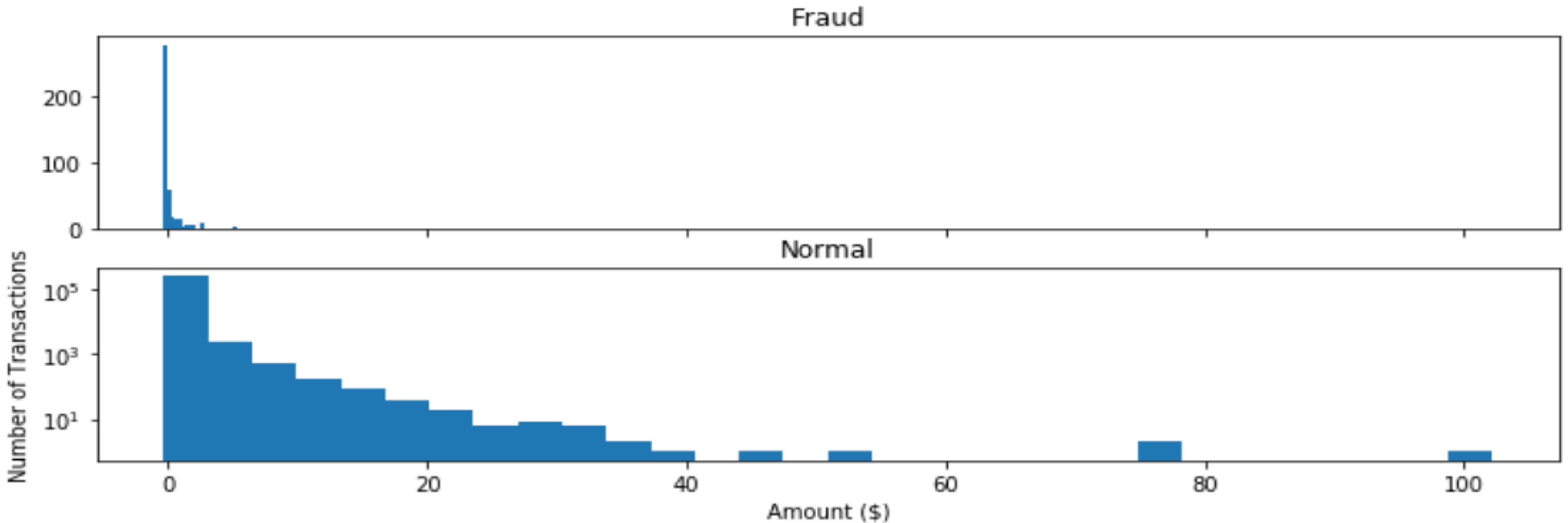
Observations:

- All the PCA transformed features are scaled
- Amount and Time input are not scaled
- In fig 2. we can see that most of the transactions(99.9%) are non –fraud and very few(0.172%) are fraud

Data Preprocessing

Scaling:

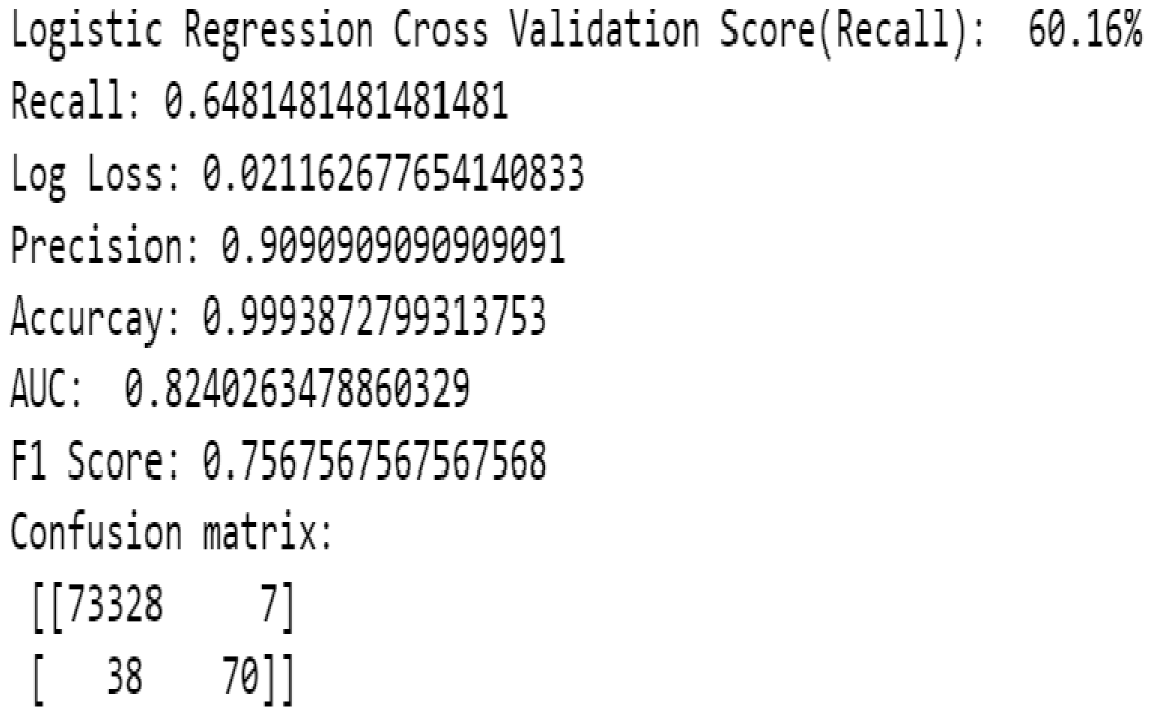
- ➢As observed from histograms most of the features are scaled except Amount and Time
- ➢I have scaled the Amount column and created new column norm_amount
- ➢ Dropped Time, Amount, ID columns
- ➢In the fig. we can see number of transactions v/s for Fraud and Non- Fraud Categories



Data Preprocessing

Need of sampling :

- Fit logistic regression on imbalanced dataset
- Got accuracy of 99.9% But it’s not true.
- As most of the labels 0, even random guess gives 99% accuracy.
- So use Recall as a accuracy measure which measures the ability of model to predict right for a given label.
- Recall is very Low 64.814%

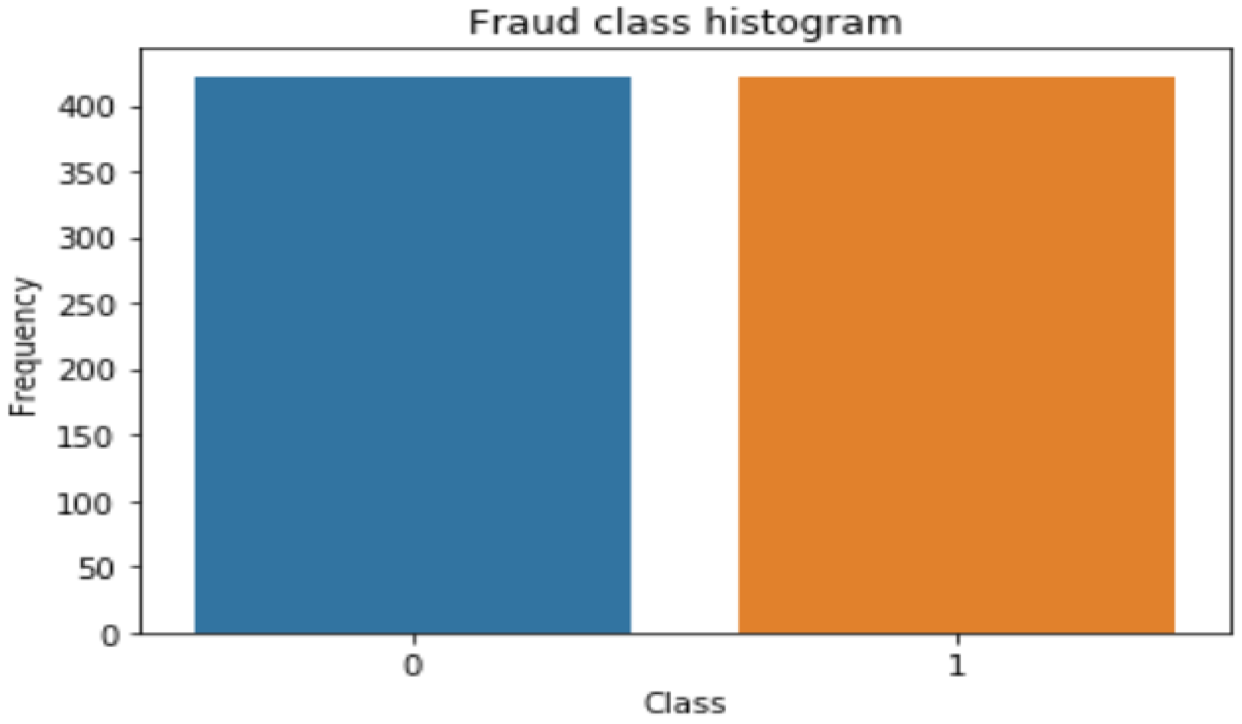


```
Logistic Regression Cross Validation Score(Recall):  60.16%
Recall: 0.6481481481481481
Log Loss: 0.021162677654140833
Precision: 0.9090909090909091
Accurcay: 0.9993872799313753
AUC:  0.8240263478860329
F1 Score: 0.7567567567567568
Confusion matrix:
 [[73328     7]
 [   38    70]]
```

Data Preprocessing

Under sampling:

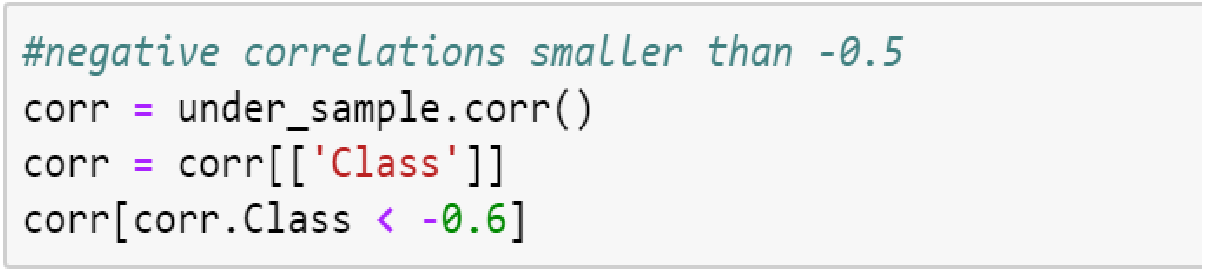
- Under sampling is one of the techniques used for handling class imbalance.
- In this technique, we under sample majority class to match the minority class and make sure that the training data has equal amount of fraud and non-fraud samples.
- In the fig we can see that training dataset has equal number of fraud and non-fraud Transactions.



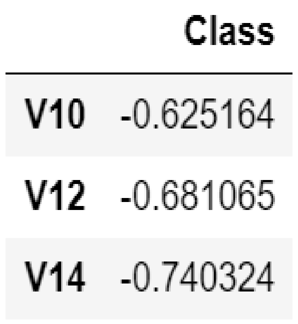
Feature Selection

Filter Method using correlation:

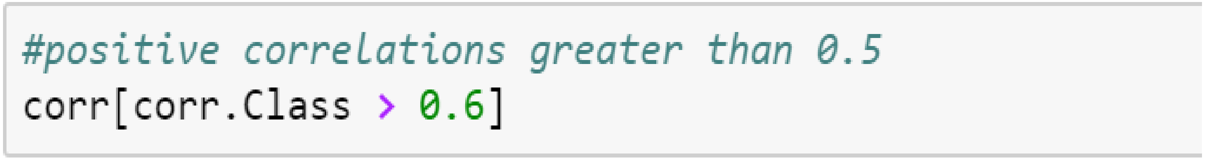
- ➢If we try to correlate class and features on imbalanced dataset then it will be of no use because we will not see true correlations of features with result.
- ➢Try correlating class and features on under sampled dataset
- ➢Fitted the model with best possible features
- ➢But it resulted in poor performance



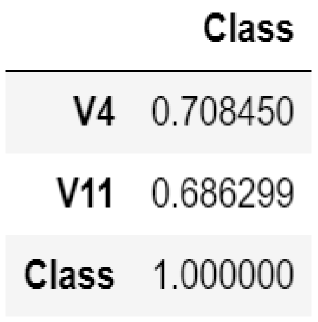
```
#negative correlations smaller than -0.5
corr = under_sample.corr()
corr = corr[['Class']]
corr[corr.Class < -0.6]
```



	Class
V10	-0.625164
V12	-0.681065
V14	-0.740324

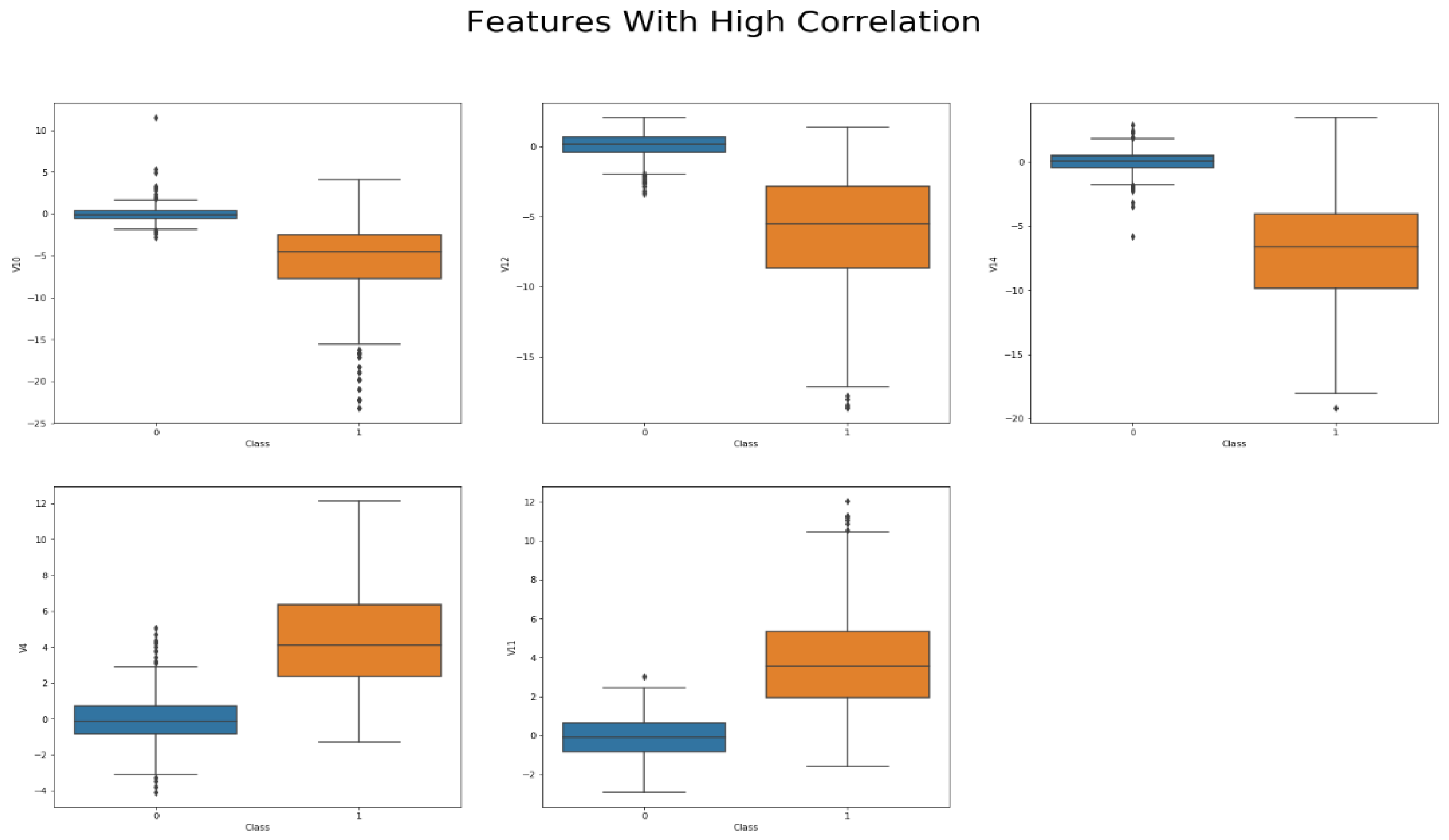


```
#positive correlations greater than 0.5
corr[corr.Class > 0.6]
```



	Class
V4	0.708450
V11	0.686299
Class	1.000000

Box plots



Outliers

- Removed the extreme outliers which were outside of 2.5 times IQR
- Fitted models with and without outliers
- Got better result before removing outliers
- Before even thinking of removing outliers there should be enough evidence that these observations are actual outliers
- It should be done by in depth statistical analysis to make sure that these observations are actual outliers because different ML methods used for detecting fraud, are based on anomaly detection and they treat such extreme outliers as frauds.
- So, by removing them we delete the most important observations, that have higher probability of being frauds

Model Implementation

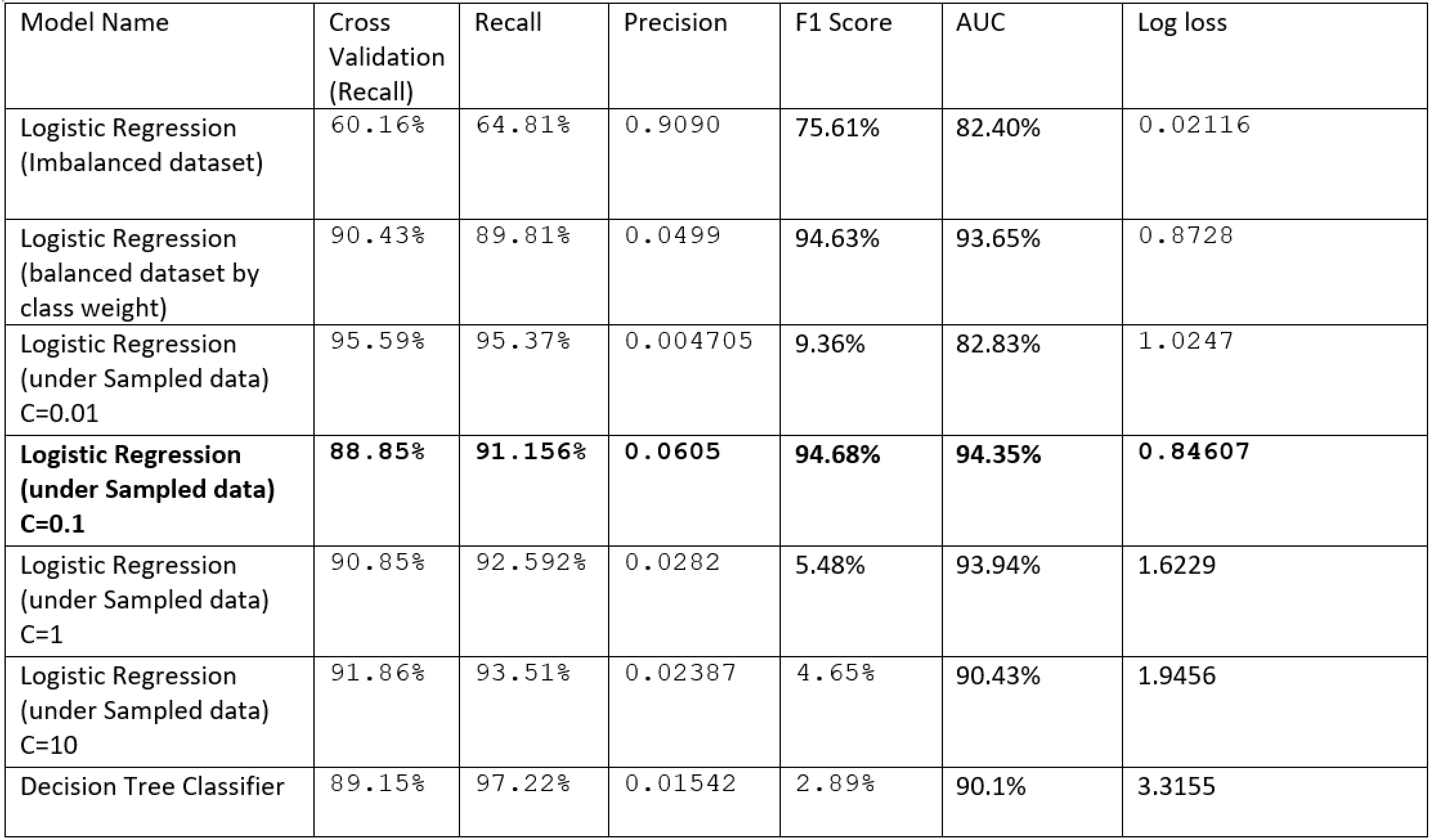
1] Logistic regression on imbalanced dataset

2] Logistic regression using class_weight: scikit-learn logistic regression has a option named class_weight when specified does class imbalance handling implicitly.

3] Logistic regression with tuning parameters[0.001, 0.1,1,10]

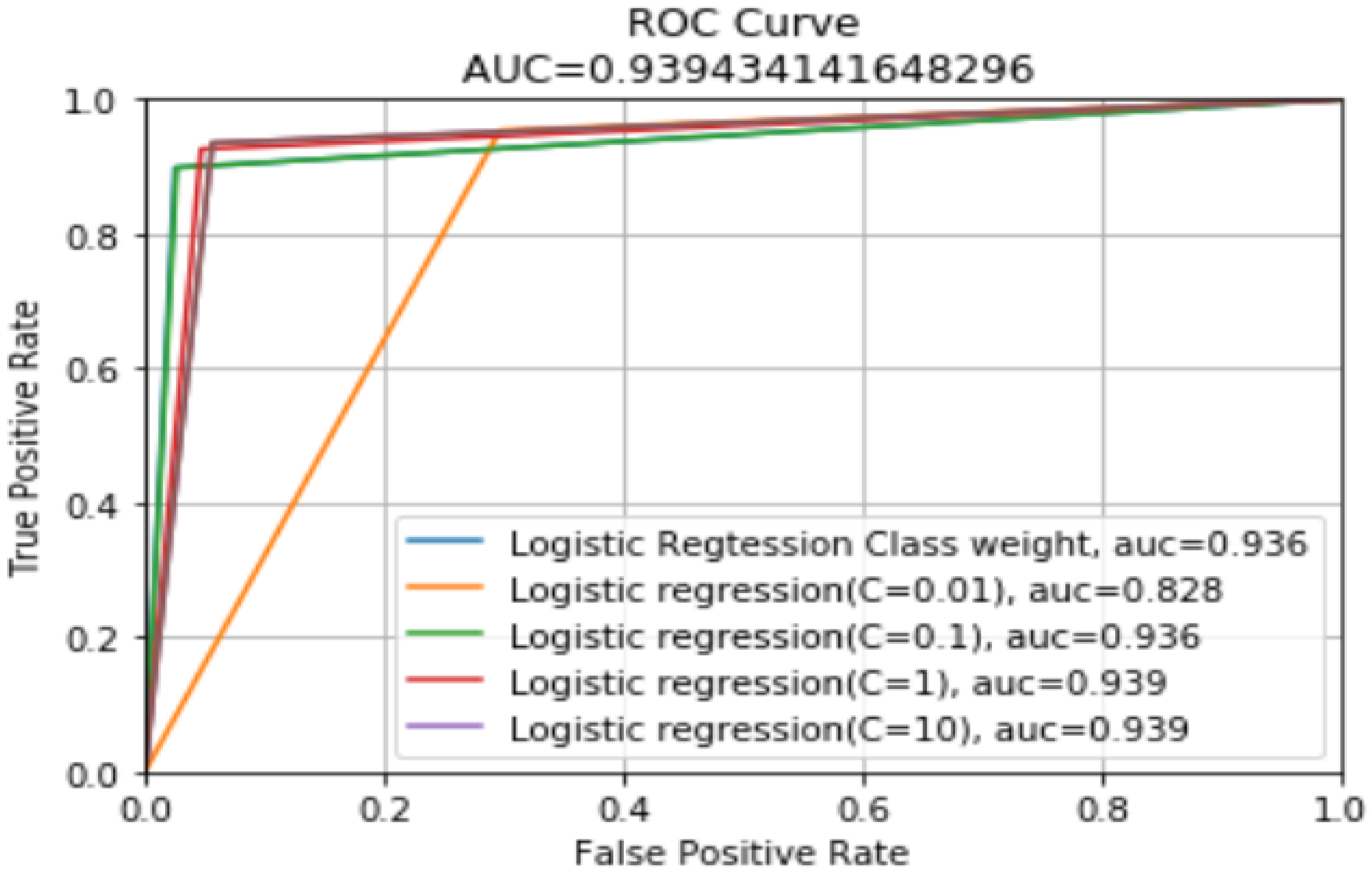
4] Decision Tree Classifier

Summary of models



Model Name	Cross Validation (Recall)	Recall	Precision	F1 Score	AUC	Log loss
Logistic Regression (Imbalanced dataset)	60.16%	64.81%	0.9090	75.61%	82.40%	0.02116
Logistic Regression (balanced dataset by class weight)	90.43%	89.81%	0.0499	94.63%	93.65%	0.8728
Logistic Regression (under Sampled data) C=0.01	95.59%	95.37%	0.004705	9.36%	82.83%	1.0247
Logistic Regression (under Sampled data) C=0.1	**88.85%**	**91.156%**	**0.0605**	**94.68%**	**94.35%**	**0.84607**
Logistic Regression (under Sampled data) C=1	90.85%	92.592%	0.0282	5.48%	93.94%	1.6229
Logistic Regression (under Sampled data) C=10	91.86%	93.51%	0.02387	4.65%	90.43%	1.9456
Decision Tree Classifier	89.15%	97.22%	0.01542	2.89%	90.1%	3.3155

Summary of models



Future Improvements

- More techniques can be explored for sampling like Over sampling, SMOTE
- In depth analysis for outlier detection
- Explore different techniques for feature selection
- Explore different machine learning algorithms like Random Forest Classifier, Support Vector etc
- Explore classification by changing threshold

Learning Outcomes

- Dealing with imbalanced dataset
- Learned analyzing and processing large dataset
- Implemented and interpreted some good visualizations
- Implemented various machine learning algorithm and evaluated their performances based on various accuracy metrics

Thank you!