PRML MINOR PROJECT
CREDIT CARD FRAUD DETECTION

Team:

Gandi Gagan (B21ES010)
Hima Varshitha Nandi (B21CS049)
Pujari Bheemesh (B21EE053)

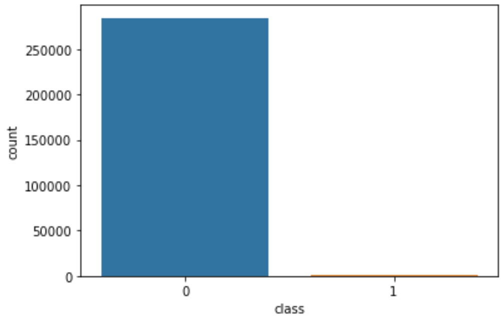
AIM: The aim of this project is to predict whether, given the details about the credit card, it is real or fake. The dataset contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

OVERVIEW: An end-to-end machine learning pipeline is implemented for the task given in the project. The following classifiers are used to train the data:

- Random Forest
- Decision Tree
- Logistic Regression
- XG Boost

For the performance evaluation of the entire pipeline confusion matrix, ROC curves are plotted. Accuracy, precision, recall, F1 scores are calculated and compared to get better results.

ANALYSIS: In the given dataset, the features V1, V2,……V28 are separated from the data. Scaling is performed on the columns of Time and Amount by importing StandardScaler from sklearn.preprocessing. The data is highly imbalanced because the number of samples in class 0(non-fraud) are 284315 and in class 1(fraud) are 492 which have a great difference.



This problem can be solved by performing oversampling or undersampling.

Undersampling - The technique of reducing the size of the majority class in an imbalanced dataset by randomly choosing samples from it equal to the number of samples in the minority class. Random Undersampling technique is used.

Oversampling - It is increasing the size of the minority class in an imbalanced dataset by replicating samples from it. ADASYN technique is used for oversampling. **ADASYN** is based on the idea of adaptively generating minority data samples according to their distributions: more synthetic data is generated for minority class samples that are harder to learn compared to those minority samples that are easier to learn.

The data is split into training and test sets of 80:20 using train_test_split from sklearn.model_selection. Now the classifiers are trained using this data.

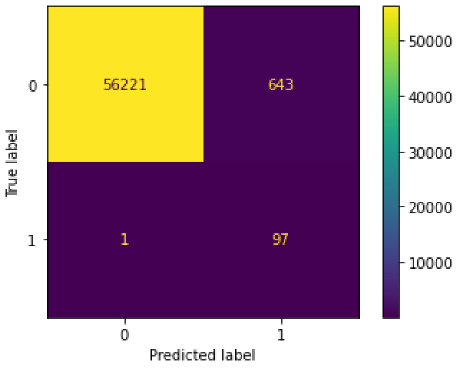
NOTE: In this type of imbalance datasets, the important evaluation metric is recall (not accuracy) because it helps us to find the most fraudulent transactions. Due to the imbalancing of the data, many observations could be predicted as False Negatives, being that we predict a normal transaction, but it is in fact a fraudulent one. Recall captures this.

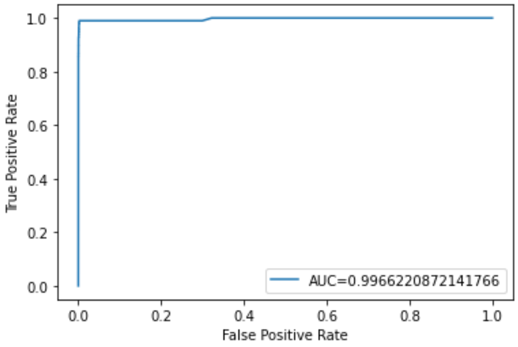
OBSERVATIONS:

- **Random Forest Classifier**

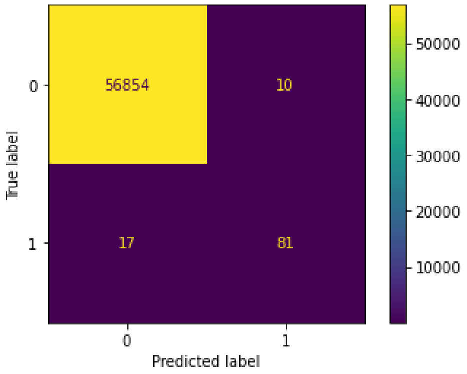
  The classifier is trained using undersampled and oversampled training data by importing RandomForestClassifier from sklearn.ensemble

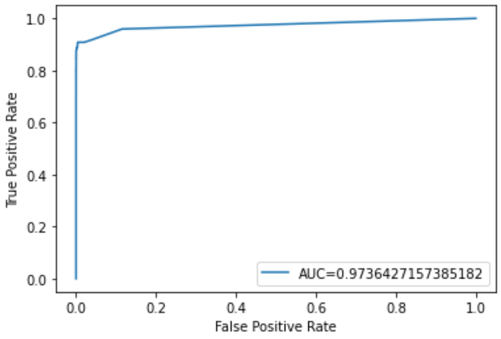
  The confusion matrix and ROC curve by using undersampled data:





  The confusion matrix and ROC curve by using oversampled data are

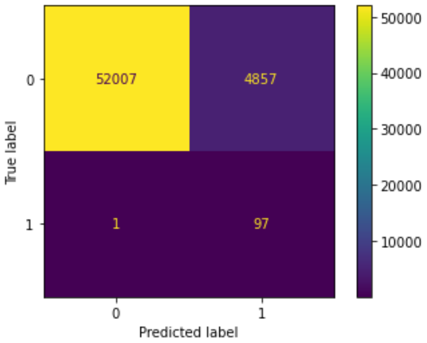


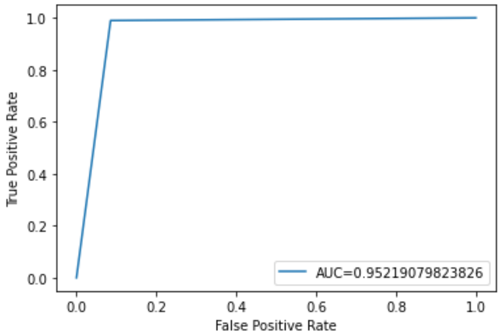


- **Decision Tree Classifier**

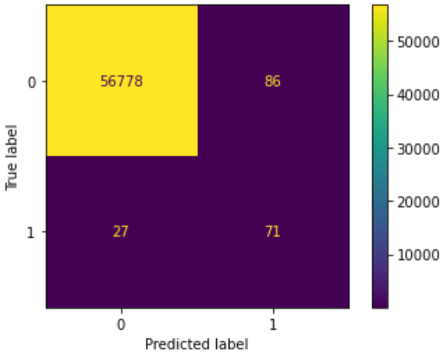
  The classifier is trained using undersampled and oversampled training data by importing DecisionTreeClassifier from sklearn.tree

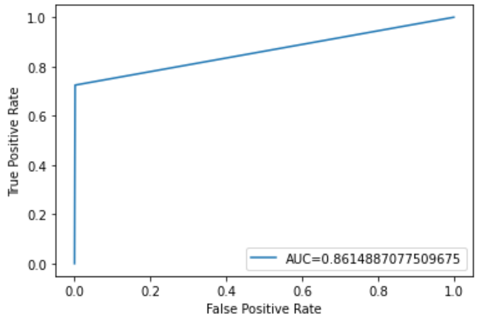
  The confusion matrix and ROC curve by using undersampled data are





  The confusion matrix and ROC curve by using oversampled data are

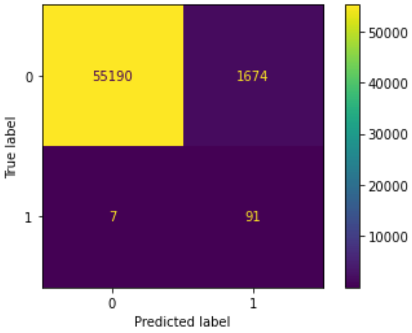


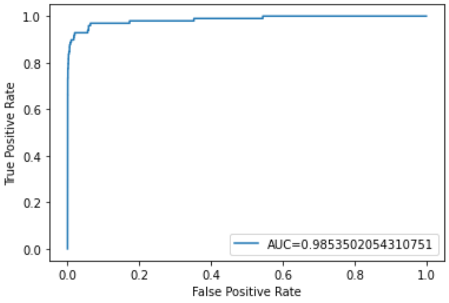


- **Logistic regression**

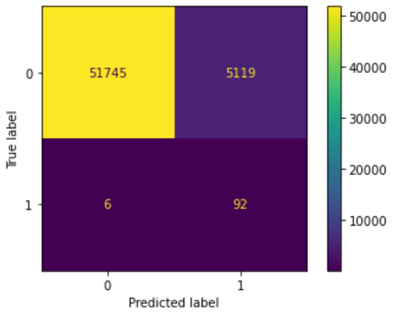
  The classifier is trained using undersampled and oversampled training data by importing LogisticRegression from sklearn.linear_model

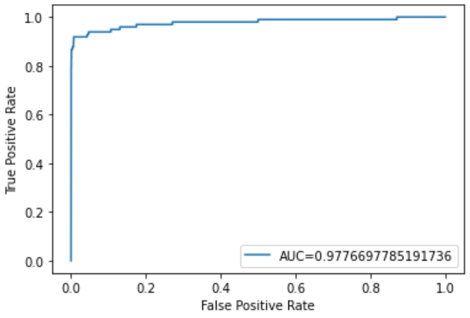
  The confusion matrix and ROC curve by using undersampled data are



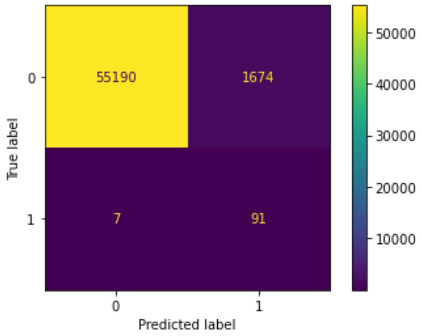


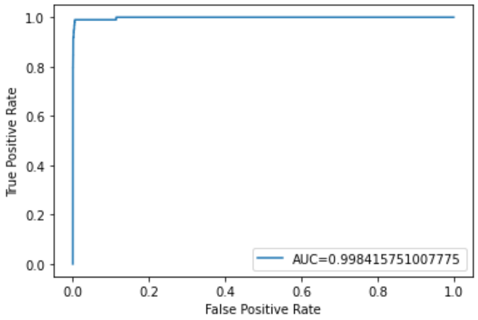
The confusion matrix and ROC curve by using oversampled data are



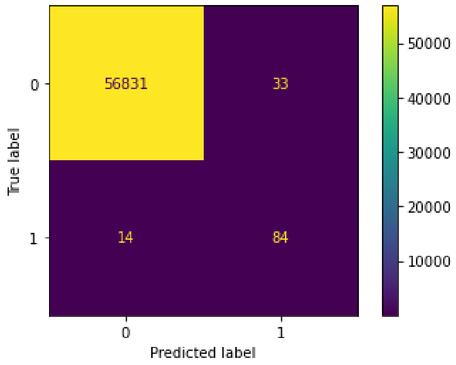


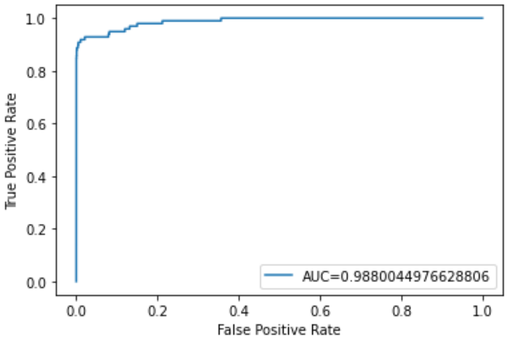
- **XGBoost Classifier**
  The classifier is trained using undersampled and oversampled training data. Xgboost is installed using the command !pip install xgboost
  The confusion matrix and ROC curve by using undersampled data are





The confusion matrix and ROC curve by using oversampled data are





		Random Forest	Decision Tree	Logistic Regression	XGBoost
By using Under sampled Data	Recall	98.9	98.9	92.8	92.8
	Precision	13.1	1.9	5.1	6.1
	F1-score	23.1	3.8	9.7	9.8
	Accuracy	98.8	91.4	97	96.2
By using Over sampled Data	Recall	82.6	72.4	93.8	85.7
	Precision	89	45.2	1.7	71.7
	F1-score	85.7	55.6	3.4	78.1
	Accuracy	99.9	99.8	91	99.9

CONCLUSION:

- We can observe that by using undersampling technique, we got high recall scores but very less precision. Which indicates that these models are not able to predict the non-fraudulent credit cards correctly. They can correctly predict fraudulent credit cards. This is happening because we are only training the model using undersampled data which leads to a lot of information loss. So this is not preferable.
- In the oversampling, we can observe that recall and precision are relatively better compared to undersampling. This is because they are training the model using oversampled data which does not lead to any information loss. So oversampling technique is preferred as it is working well.
- By comparing the models, we can observe that the recall scores of logistic regression and XGBoost are better than others. But logistic regression has very low precision, but XGBoost has decent precision. Also the AUC value is better for XGBoost than Logistic Regression. So, XGBoost will be a better model in classifying both fraudulent and non-fraudulent samples.