Capstone project

Summary

In response to the rising incidence of credit card fraud, our project aims to develop an effective solution for real-world fraud detection scenarios. As criminals increasingly use fake identities and advanced technologies to exploit users, formulating efficient strategies has become crucial. Our proposed model addresses this challenge by leveraging machine learning and other relevant IT fields.

To adapt to the evolving nature of fraudulent activities, we employ logistic regression, decision tree, and XGBoost algorithms to classify credit card transactions as either fraudulent or genuine. By analyzing extensive datasets and incorporating real-time user data, our model enhances the accuracy of fraud detection. Additionally, we utilize data visualization techniques to process key attributes, enabling the identification of fraudulent patterns.

We evaluate our techniques based on metrics such as accuracy, precision, ROC curves, and recall. Our results demonstrate the superior performance of the XGBoost algorithm, achieving significant accuracy levels.

Introduction

Detecting credit card fraud presents significant challenges in the financial sector, with billions lost annually due to fraudulent activities. Traditional methods, which involve analyzing spending patterns and implementing government and bank safeguards, are often insufficient as fraudsters continuously adapt to evade detection.

Although instances of credit card fraud comprise only about 0.1% of all card transactions, they result in substantial financial losses due to the high value of fraudulent transactions. It is crucial for credit card companies to recognize fraudulent transactions to ensure that customers are not charged for items they did not purchase. An effective algorithm to classify transactions as fraudulent or non-fraudulent is needed to benefit both credit card companies and customers who endure the aftermath of fraud.

To address this issue, our project focuses on predictive analytics, utilizing three classification algorithms: Logistic Regression, Decision Trees, and XGBoost. By analyzing a credit card dataset, we aim to identify the most effective model for distinguishing between genuine and fraudulent transactions.

Problem Statement

Credit cards are widely used for online purchases and payments, offering convenience but also posing risks. Credit card fraud, which involves the unauthorized use of someone else's credit card or credit card information to make purchases or withdraw cash, is a significant concern. It is essential for credit card companies to detect fraudulent transactions to prevent customers from being unfairly charged.

The dataset provided contains transactions made by European cardholders in September 2013, covering a two-day period with 492 fraudulent transactions out of a total of 284,807 transactions. This dataset is highly imbalanced, with fraudulent transactions representing only 0.172% of the total.

Our objective is to develop a classification model to predict whether a transaction is fraudulent or not.

Focus on this project

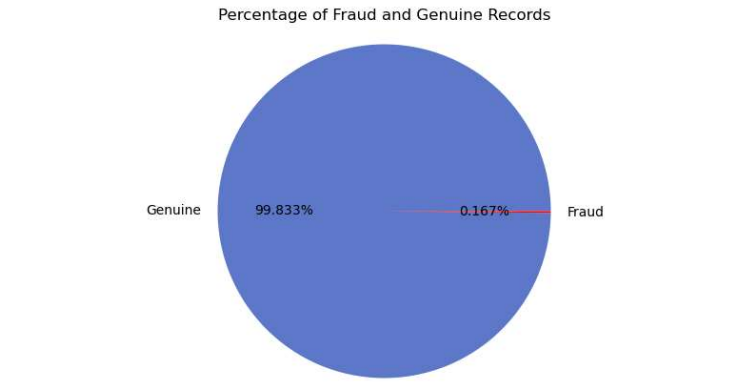
- **Exploratory Data Analysis:** Analyze and understand the data to identify patterns, relationships, and trends in the data by using Descriptive Statistics and Visualizations
- **Data Cleaning:** This might include standardization, handling the missing values and outliers in the data.
- **Dealing with Imbalanced data**: This data set is highly imbalanced. The data should be balanced using the appropriate methods before moving onto model building.
- **Feature Engineering**: Create new features or transform the existing features for better performance of the ML Models.
- **Model Selection:** Choose the most appropriate model that can be used for this project.
- **Model Training:** Split the data into train & test sets and use the train set to estimate the best model parameters.
- **Model Validation:** Evaluate the performance of the model on data that was not used during the training process. The goal is to estimate the model's ability to generalize to new, unseen data and to identify any issues with the model, such as overfitting.

Dataset

The dataset used in this project consists of transactions made by European cardholders in September 2013. It includes a total of 284,807 transactions, with 492 labeled as fraudulent. The dataset is highly imbalanced, as fraudulent transactions account for only 0.172% of all transactions.

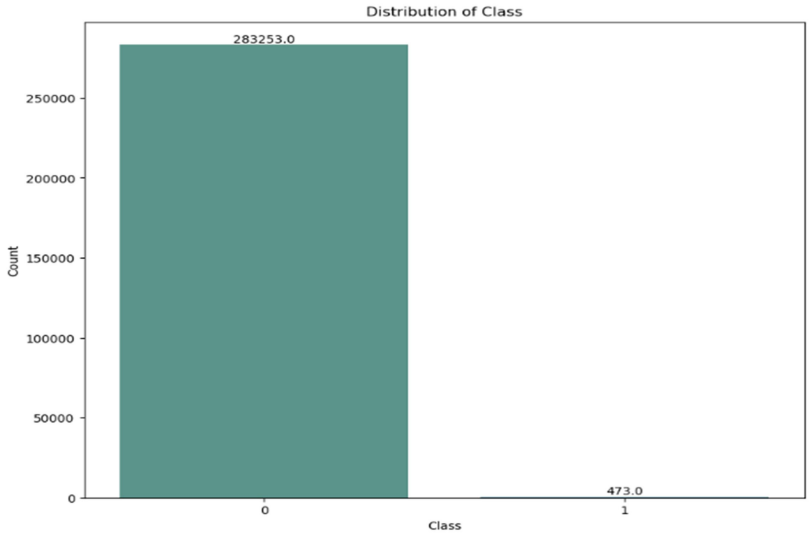
Data cleaning

- **Missing Value:** There are no missing values in this dataset.
- **Feature Scaling:** Through the above dataset analysis, it is seen that all the columns are scaled except the Amount & Time features. These are the only features which have not been transformed. Feature Time contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature Amount is the transaction Amount. We have scaled these features using the Standardization method.
- **Encoding:** Not performing any encoding for this dataset. Because all the columns are in numerical format.
- **Imbalance data:** As there are fewer number of transactions for a particular class, the data can be said to be unbalanced. The unbalanced class distribution can be visualized in a diagram given below.



Genuine transactions make up approximately 99.833% of the dataset, while fraudulent transactions represent only 0.167%.

Bar distribution of classes

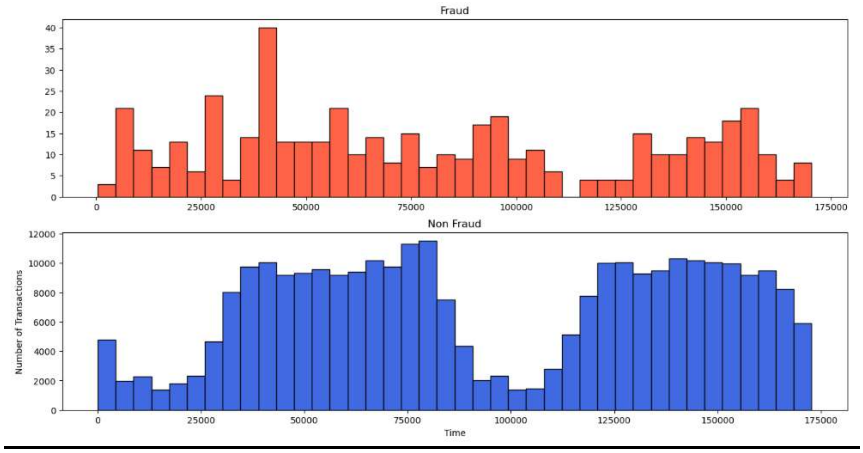


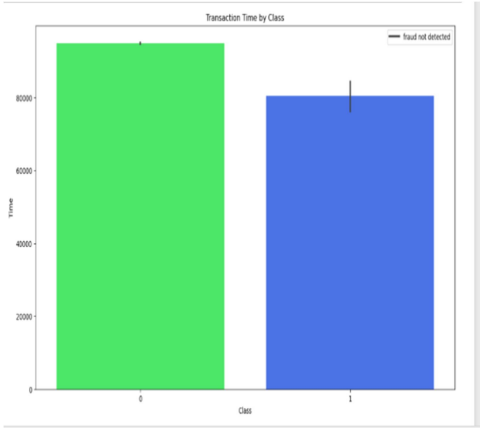
- Here, 283253 transactions are genuine (class 0), and 473 transactions are fraud (class 1) which clearly depicts the high imbalance in our dataset.
- **Handling imbalance data:** To make the data balancing **Oversampling technique** is used. We will use SMOTE (Synthetic Minority Over-sampling Technique). SMOTE is a technique used to generate synthetic samples for the minority class to balance the class distribution in the dataset. By creating synthetic samples, SMOTE helps mitigate the impact of class imbalance and improves the performance of machine learning models in predicting the minority class.

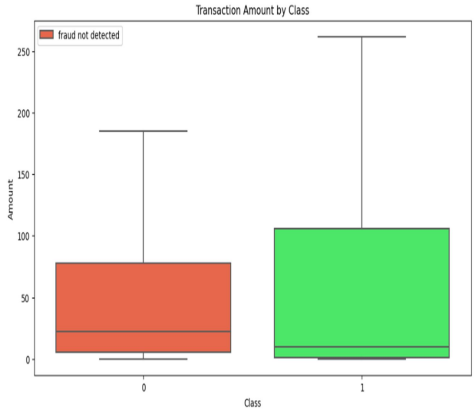
  Using Oversampling (SMOTE) instead of undersampling because the difference between the two classes (0 and 1) is huge, so if undersampling technique is used, it will lead to losing of most of the sensitive information from the data. Therefore, to overcome that oversampling is used.

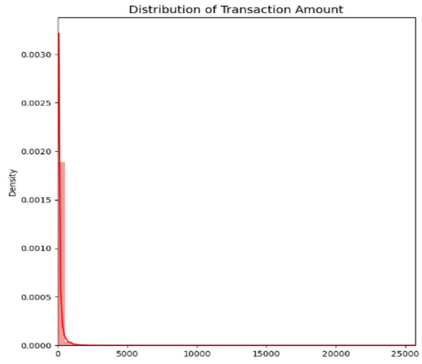
Exploratory Data Analysis

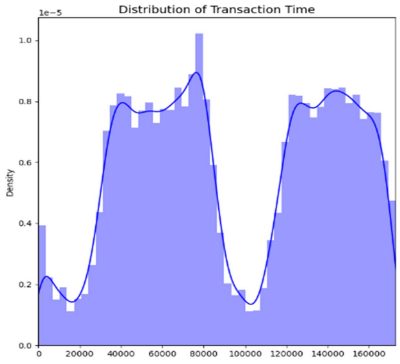
We will utilize various visualization techniques for basic Exploratory Data Analysis (EDA), including subplots, bar plots, box plots, histograms, and heatmaps.











Model Selection, Building and Evaluation:

We build the model with train-test split in the ratio 80-20% (80% training data and 20% test data). Then we have found which Machine Learning model works well with the balanced dataset.
The algorithms used are: -

1. Logistic Regression
2. Decision Tree
3. XGBoost

- **Logistic regression:**

Logistic regression is an essential tool for classification tasks, particularly when the target variable is categorical. It uses a logistic function to model a binary response variable. Although there are different types of logistic regression, binary logistic regression is suitable for this project, accommodating two distinct classes (0 and 1). Within this framework, predictions represent the likelihood of outcomes belonging to each class.

Model evaluation for logistic regression is shown below:

Logistic Regression Model Evaluation

Metric	Value	Interpretation
Accuracy	94.52%	The model correctly predicted transactions for 94.52% of the cases.
Precision (Fraud)	97.26%	Out of all class as fraud, only 97.26% were actually fraud.
Recall (Fraud)	91.66%	The model identified 94.66% of the actual fraud transaction.

The evaluation of the Logistic Regression model in the class domain reveals its performance in predicting fraud transaction. While it achieved an accuracy of 94.52%, indicating overall correctness.

- **Decision Tree:**

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

Model Evaluation of decision tree shown below:

Decision Tree Model Evaluation

Metric	Value	Interpretation
Accuracy	99.82%	The model correctly predicted transactions for 99.82% of the cases.
Precision (Fraud)	99.73%	Out of all class as fraud, only 99.73% were actually fraud.
Recall (Fraud)	99.91%	The model identified 99.91% of the actual fraud transaction.

- **XGBoost:**

XGBoost is an enhanced version of gradient boosting, offering additional features like a parallel tree learning algorithm and regularization for optimal split determination. It follows the principle of gradient boosting but incorporates differences in modeling details. Specifically, XGBoost uses a more regularized model formalization to control overfitting, resulting in better performance.

The name XGBoost refers to the engineering goal of maximizing computational resources for boosted tree algorithms. This optimization is a key reason for its widespread use. For the model, it may be more accurately described as regularized gradient boosting.

Model evaluation for XGBoost is shown below:

XgBoost Model Evaluation

Metric	Value	Interpretation
Accuracy	98.28%	The model correctly predicted transactions for 98.28% of the cases.
Precision (Fraud)	98.84%	Out of all transactions as fraud, only 98.84% were actually fraud.
Recall (Fraud)	97.72%	The model identified 97.72% of the actual fraud transactions.

Hyperparameter tuning

Hyperparameter tuning is the process of selecting the optimal values for a machine learning model's hyperparameters. Hyperparameters are settings that control the learning process of the model, such as the learning rate, the number of neurons in a neural network, or the kernel size in a support vector machine. The goal of hyperparameter tuning is to identify the settings that lead to the best performance for a given task.

Using the XGBoost Classifier with optimal hyperparameters, the model evaluation after tuning is shown below:

XGBoost Model Evaluation (after tuning)

Metric	Value	Interpretation
Accuracy	99.75%	The model correctly predicted transactions for 99.75% of the classes.
Precision (Fraud)	99.84%	Out of all transactions predicted as fraud, only 99.84% were actually fraud.
Recall (Fraud)	99.66%	The model identified 99.66% of the actual fraud transactions.

Conclusion

The project aimed to develop algorithms for detecting fraudulent credit card transactions. Various models were tested after balancing the data to determine the best performer across balanced class distributions. Analysis of the outcomes revealed that leveraging XGBoost with oversampled data and fine-tuned hyperparameters yields the best results for identifying fraudulent transactions. This approach consistently demonstrates high precision and recall rates across multiple folds, highlighting its effectiveness and accuracy in managing imbalanced datasets and delivering precise predictions.