CREDIT CARD FRAUD DETECTION

Project Report Submitted By

ABHIJITH A

ABSTRACT

This project focuses on developing an efficient and robust credit card fraud detection system. By leveraging machine learning algorithms, specifically logistic regression, and data analysis techniques, we aim to accurately identify and prevent fraudulent transactions.

The project involves collecting and integrating various datasets, including transaction time and amount, user location, merchant details, and historical transaction data. Exploratory Data Analysis (EDA) techniques are employed to gain insights into the datasets, identify patterns, and understand the characteristics of fraudulent transactions.

Logistic regression is employed to predict the likelihood of credit card fraud based on features such as transaction amount, user behavior patterns, and historical transaction data. The model analyzes factors like transaction velocity, geographical location, and spending habits to classify transactions as legitimate or fraudulent.

The project also involves the development of a robust data management system to store and manage the vast amount of transaction data efficiently. This system facilitates data

retrieval, analysis, and model updates, enabling real-time fraud detection and continuous improvement of the system.

By implementing this fraud detection system, financial institutions can significantly reduce financial losses, enhance customer trust, and improve overall security. The system provides a proactive approach to fraud prevention, enabling timely intervention and minimizing the impact of fraudulent activities.

INTRODUCTION

This project aims to develop an efficient and robust system for detecting fraudulent credit card transactions. By leveraging machine learning algorithms and data analysis techniques, the system will analyze historical transaction data to identify patterns and anomalies indicative of fraudulent activity. The objective is to build a predictive model capable of accurately classifying transactions as either legitimate or fraudulent in real-time. This will enable proactive fraud detection, reduce financial losses, and enhance security for both customers and financial institutions.

SCOPE OF THE PROJECT

1. Data Collection and Integration:

- Data Acquisition: Collect historical credit card transaction data from various sources such as databases or APIs.
- Data Cleaning and Preprocessing: Clean the dataset by handling missing values, addressing outliers, and transforming data types as needed to ensure consistency and accuracy.
- Feature Engineering: Create new features from existing ones (e.g., time-based features, transaction amounts, and customer behavior patterns) to improve the performance of the fraud detection model.
-

2. Exploratory Data Analysis (EDA):

- Descriptive Statistics: Calculate summary statistics (mean, median, standard deviation) for key variables to understand the data's central tendencies and variability.
- Data Visualization: Use visualizations (e.g., histograms, box plots, scatter plots) to analyze feature distributions, detect patterns, and identify any anomalies in the dataset.
- Feature Importance Analysis: Evaluate the relationships between the features and the target variable (fraudulent or legitimate transactions) to identify the most significant predictors of fraud.

- Class Imbalance Handling: Investigate and address the issue of class imbalance, where fraudulent transactions are typically fewer than legitimate ones, by using techniques like oversampling, undersampling, or generating synthetic data.
-

3. Machine Learning Model Development:

- Model Selection: Choose suitable machine learning algorithms for fraud detection, such as logistic regression, decision trees, random forests, support vector machines, or neural networks.
- Model Training: Train the selected models on the preprocessed data, and evaluate them using relevant metrics such as accuracy, precision, recall, F1-score, and AUC-ROC.
- Model Optimization: Fine-tune the hyperparameters of the models to enhance their performance, ensuring they generalize well to unseen data and are resistant to overfitting.
-

4. Model Deployment and Monitoring:

- Deployment: Integrate the trained fraud detection model into a real-time or near-real-time system to monitor and flag potential fraudulent transactions.
- Monitoring: Continuously track the model's performance in production, monitoring key metrics like detection rates, false positives, and false negatives.

- Model Retraining: Periodically retrain the model using new data to adapt to emerging fraud patterns and ensure its continued accuracy.
- Alerting System: Develop a notification system to alert relevant personnel (e.g., fraud analysts) of suspected fraudulent transactions for further review.
-

5. Testing and Validation:

- Rigorous Testing: Test the system thoroughly using diverse datasets and scenarios to ensure its robustness, accuracy, and reliability under various conditions.
- Performance Evaluation: Assess the model's performance based on predefined metrics and compare it with existing fraud detection systems to ensure competitiveness and effectiveness.
- User Acceptance Testing: Involve relevant stakeholders (e.g., fraud analysts, risk managers) in testing to confirm that the system meets their expectations and operational requirements.
-

6. Documentation and Reporting:

- Documentation: Develop comprehensive documentation that outlines the data sources, preprocessing steps, model selection process, training, evaluation, and deployment methodologies.

- Reporting: Generate periodic reports on the system's performance, detailing fraud detection rates, false positives, model updates, and any issues encountered during deployment to keep stakeholders informed.

IMPLEMENTATION

1. Data Collection:

- Gather historical credit card transaction data, including transaction amounts, merchant details (category, location), user behavior (frequency, time of day, location), and timestamps.
- Collect additional transaction-specific information, such as the time of the transaction and the amount involved, to capture spending patterns.
- Collect user profile data (if available and ethically permissible), including demographics (age, location), past spending habits, and device information.
- Obtain external data such as historical fraud cases, transaction patterns, and regulatory requirements for fraud detection (e.g., PCI DSS compliance, fraud reporting standards).

2. Data Preprocessing:

- Data Cleaning: Clean the data by handling missing values, addressing outliers, and correcting inconsistencies (e.g., incorrect timestamps, invalid amounts).

- Feature Engineering: Create new, relevant features from the raw transaction data, such as:

  - Transaction frequency
  - Time-based features
  - Transaction velocity
  - User behavioral patterns

- Data Encoding: Encode categorical variables using techniques like one-hot encoding or label encoding to prepare the data for model training.

3. Model Development:

- Model Selection: Choose appropriate machine learning algorithms for fraud detection, such as logistic regression, decision trees, random forests, support vector machines, or neural networks.
- Model Training: Train the selected models on the preprocessed data and evaluate performance using appropriate metrics like accuracy, precision, recall and F1-score.
- Model Optimization: Fine-tune hyperparameters of the models using techniques like grid search or random search to enhance performance, ensuring the models generalize well on unseen data and effectively detect fraudulent transactions.

4. Testing and Validation:

- Testing: Conduct thorough testing of the fraud detection system, including unit testing, integration testing, and system testing, to ensure the solution is reliable and error-free.
- Model Validation: Perform model validation using techniques like k-fold cross-validation or hold-out validation to assess model performance, check for overfitting, and verify generalization ability.
- Performance Evaluation: Evaluate model performance using predefined metrics, such as fraud detection rate, false positive rate, precision, recall, F1-score, and AUC-ROC, to ensure that the system accurately detects fraudulent transactions with minimal false positives.
- User Acceptance Testing: Conduct user acceptance testing (UAT) with relevant stakeholders, such as fraud analysts or risk managers, to ensure the system meets their needs, provides actionable insights, and aligns with operational requirements.

5. Deployment and Maintenance:

- Deployment: Deploy the fraud detection model into a production environment, ensuring real-time integration with existing transaction processing systems and implementing necessary security protocols.

- Ongoing Maintenance: Provide continuous maintenance and support, which includes regular retraining of the models with updated data, addressing emerging fraud patterns, and resolving any performance or scalability issues.
- Monitoring and Feedback: Continuously monitor system performance by tracking key metrics like fraud detection rates, false positive rates, alert volumes, and transaction processing times. Analyze logs and system behavior to identify opportunities for improvement.
- Continuous Improvement: Regularly evaluate the system based on new data, evolving fraud trends, and feedback from stakeholders to adapt the model and maintain high detection accuracy over time.

EXPLORARTORY DATA ANALYSIS

The chart below gives info regarding various columns in the datasets :

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 284807 entries, 0 to 284806

Data columns (total 31 columns):

#	Column	Non-Null Count	Dtype
0	Time	284807 non-null	float64
1	V1	284807 non-null	float64
2	V2	284807 non-null	float64
3	V3	284807 non-null	float64
4	V4	284807 non-null	float64
5	V5	284807 non-null	float64
6	V6	284807 non-null	float64
7	V7	284807 non-null	float64
8	V8	284807 non-null	float64
9	V9	284807 non-null	float64
10	V10	284807 non-null	float64
11	V11	284807 non-null	float64
12	V12	284807 non-null	float64
13	V13	284807 non-null	float64
14	V14	284807 non-null	float64

15 V15 284807 non-null float64

16 V16 284807 non-null float64

17 V17 284807 non-null float64

18 V18 284807 non-null float64

19 V19 284807 non-null float64

20 V20 284807 non-null float64

21 V21 284807 non-null float64

22 V22 284807 non-null float64

23 V23 284807 non-null float64

24 V24 284807 non-null float64

25 V25 284807 non-null float64

26 V26 284807 non-null float64

27 V27 284807 non-null float64

28 V28 284807 non-null float64

29 Amount 284807 non-null float64

30 Class 284807 non-null int64

dtypes: float64(30), int64(1)

memory usage: 67.4 MB

BUILDING THE MODEL

1.Data Loading and Preprocessing:

- o Load the credit card fraud dataset (creditcard.csv) using pandas.
- o Explore the data by looking at the head, distribution of the 'Amount' feature for legitimate and fraudulent transactions, and group-wise means.
- o To address class imbalance (more legitimate transactions than fraudulent ones), you randomly undersample the majority class (legitimate) to match the size of the minority class (fraudulent).
- o Create a new balanced dataset (new_dataset) by combining the undersampled legitimate transactions and all fraudulent transactions.
- o Split the data into features (X) and target variable (Y).
- o Perform train-test split to create training and testing sets for model evaluation (stratified split to maintain class balance in both sets).

2. Model Training and Evaluation:

- o Instantiate a Logistic Regression model.
- o Train the model on the training data (X_train, Y_train).
- o Evaluate the model's performance on the training data using

accuracy score.

- Evaluate the model's performance on the testing data (X_test, Y_test) using accuracy score.

The code demonstrates a good initial approach to building a fraud detection model. Here are some additional considerations and potential improvements:

IMPORT NECESSARY LIBRARIES AND MODELS:

Import libraries:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

EDA:

```
creditcard_data=pd.read_csv('creditcard.csv')
creditcard_data.head()
creditcard_data.tail()
creditcard_data.info()
creditcard_data.isnull().sum()
creditcard_data['Class'].value_counts()
legit=creditcard_data[creditcard_data.Class==0]
fraud=creditcard_data[creditcard_data.Class==1]
legit.shape
fraud.shape
```

statical measures of data:

```
legit.Amount.describe()
fraud.Amount.describe()
```

comparing values of both data:

```
creditcard_data.groupby('Class').mean()
```

under sampling:

```
legit_sample=legit.sample(n=492)
```

concatnate two dataframes:

```
new_dataset=pd.concat([legit_sample,fraud],axis=0)
new_dataset['Class'].value_counts()
new_dataset.groupby('Class').mean()
```

splitting data into features and targets:

```
X=new_dataset.drop(columns='Class',axis=1)
Y=new_dataset['Class']
X.head()
Y.head()
```

Split training data and testing data:

```
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0
.2,stratify=Y,random_state=2)
X.shape
X_train.shape
X_test.shape
```

model training:

```
model=LogisticRegression()
model.fit(X_train,Y_train)
```

accuracy score:

accuracy on training data

```
X_train_prediction=model.predict(X_train)
traning_data_accuracy=accuracy_score(X_train_prediction,Y
_train)
traning_data_accuracy
```

accuracy on testing data:

```
X_test_prediction=model.predict(X_test)
testing_data_accuracy=accuracy_score(X_test_prediction,Y_
test)
testing_data_accuracy
```

CONCLUSION

The development of an effective credit card fraud detection system is crucial in today's digital economy. This project has demonstrated the potential of leveraging data analysis, machine learning, and advanced technologies to identify and prevent fraudulent transactions. By implementing a robust fraud detection system, financial institutions can significantly reduce financial losses, enhance customer trust, and improve overall security.

Moving forward, continued research and development in areas such as anomaly detection, deep learning, and real-time fraud monitoring will be essential to stay ahead of evolving fraud tactics and ensure the continued effectiveness of fraud detection systems. Collaboration between financial institutions, technology providers, and regulatory bodies will also be vital in sharing knowledge, developing best practices, and combating fraud effectively.