Credit Card Fraud Detection

This presentation covers our project on credit card fraud detection. We will explore the methods and implementation. Our goal is to protect against financial fraud.

Presented by:

Sana Parween 23scse1280060

Pranavi Jain 23scse1010720

Shiwangee Singh 23scse1012283

Muskan Shah 23scse1010456

Introduction to Credit Card Fraud

Significant Losses

Global fraud losses reached $27.86 billion in 2018. This highlights a critical financial threat.

Consumer Impact

Fraudulent transactions disproportionately affect consumers. This erodes trust and causes distress.

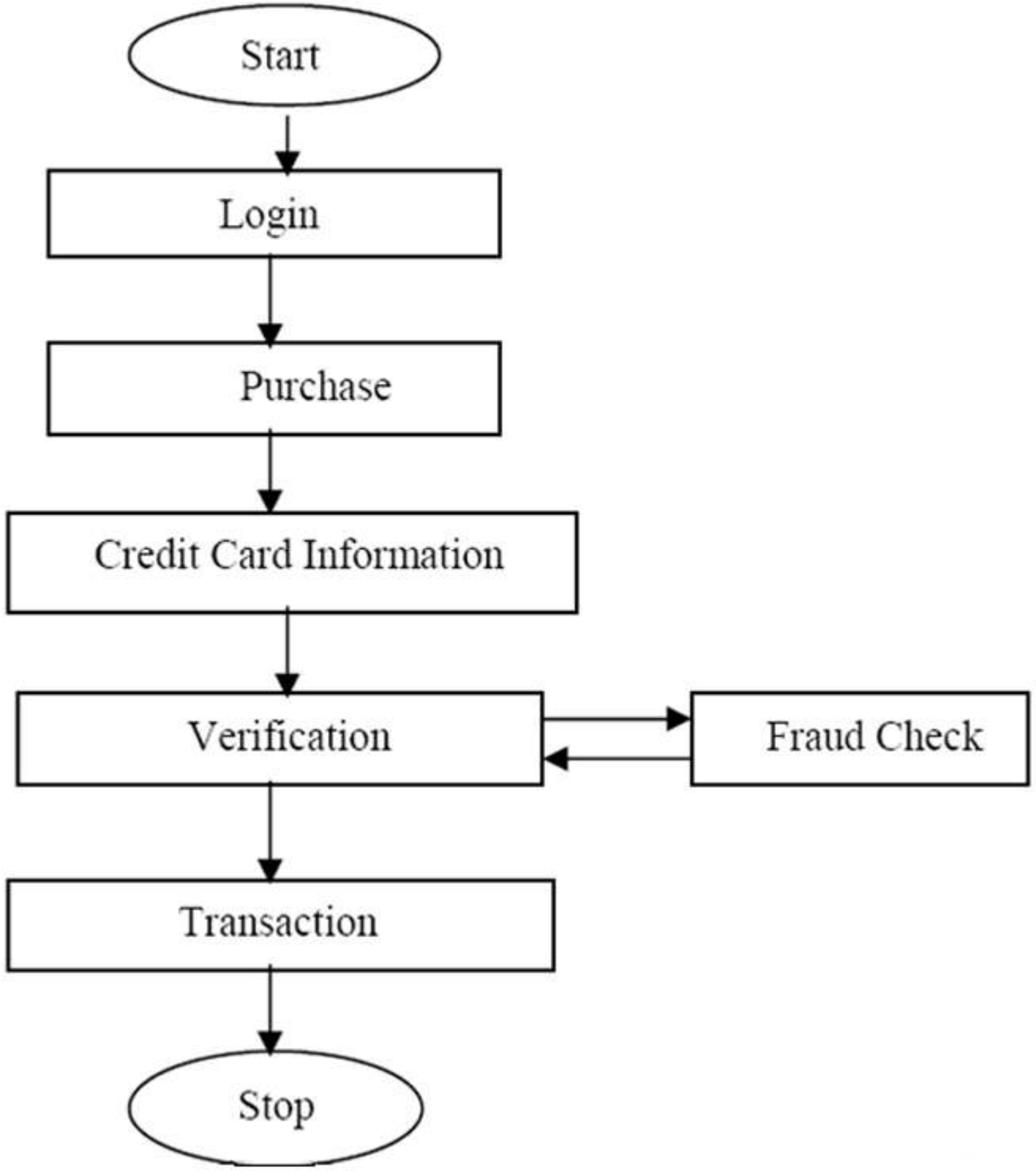
Automation Needed

Machine Learning offers superior detection. It surpasses traditional rule-based systems.

High Accuracy Objective

Our main goal is to achieve high accuracy. This ensures effective fraud detection.

Flowchart


Start
Login
Purchase
Credit Card Information
Verification
Fraud Check
Transaction
Stop

Steps:-

The steps are broadly divided into below steps. The sub steps are also listed while we approach each of the steps.

1. Reading, understanding and visualising the data
2. Preparing the data for modelling
3. Building the model
4. Evaluate the model

In [2]:

```
# This was used while running the model in Google Colab
# from google.colab import drive
# drive.mount('/content/drive')
```

In [3]:

```
# Importing the libraries
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

import warnings
warnings.filterwarnings('ignore')
```

Dataset Overview

Source	European Cardholders, Sept 2013
Imbalance	0.172% fraudulent out of 284,807
Features	28 anonymized (V1-V28), Time, Amount
Target	Class (0: Normal, 1: Fraudulent)

The dataset presents a significant imbalance. Anonymized features protect privacy. The target variable is crucial for classification.

Methodology: Addressing Data Imbalance

Resampling Techniques

Random Under-Sampling reduces majority instances. SMOTE synthesizes minority class samples.

Evaluation Metrics

Precision, Recall, F1-score, and AUC-ROC are used. These evaluate model performance comprehensively.

Model Selection

Logistic Regression, SVM, and Random Forest were chosen. These algorithms are well-suited for classification.

Model Training and Evaluation

Data Splitting

We split the data into 70% training and 30% testing. This ensures robust model validation.

Feature Scaling

StandardScaler normalized feature ranges. This prevents dominance by large values.

Algorithm Performance

Logistic Regression showed competitive results. Random Forest demonstrated high accuracy.

SMOTE Impact

SMOTE significantly improved recall. However, it slightly reduced precision.

Results and Findings

Exploratory data analysis

Reading and understanding the data

In [5]:
```
# Reading the dataset
df = pd.read_csv('creditcard.csv')
df.head()
```

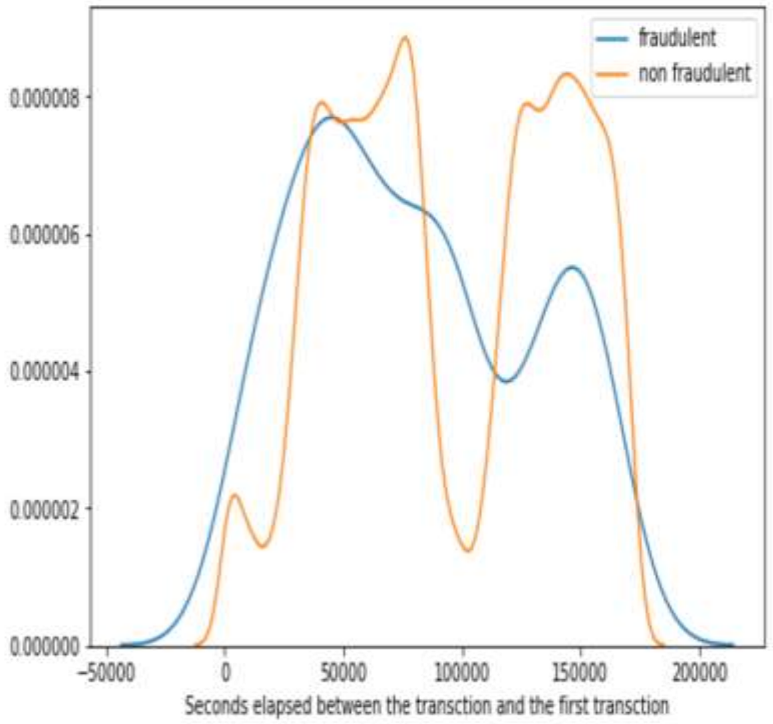
Out[5]:

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	0.090794
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	-0.166974
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	0.207643
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	-0.054952
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	0.753074

In [6]:
```
df.shape
```

Out[6]: (284807, 31)

In [7]:
```
df.info()
```



Analysis

We do not see any specific pattern for the fraudulent and non-fraudulent transctions with respect to Time. Hence, we can drop the `Time` column.

In [17]:
```
# Dropping the Time column
df.drop('Time', axis=1, inplace=True)
```

Project Outcomes and Future Enhancements

Impact

Reduces financial losses and builds customer trust.

Scalability

Handles large transaction volumes in real-time easily.

Future Work

Explore XGBoost and Neural Networks. Implement anomaly detection. Develop a real-time system. Add location and IP address features.

Conclusion

ML for Fraud

Machine learning is effective. It robustly detects credit card fraud.

SMOTE's Role

SMOTE successfully handles imbalanced data. This improves model performance.

Continuous Improvement

Further enhancements will boost system performance. Ongoing development is key.

Resource

Access the project on GitHub: sana-khan05/Credit-Card-Fraud-Detection

Thank You!

GitHub Link:- https://github.com/sana-khan05/Credit-Card-Fraud-Detection