A

MINI PROJECT REPORT

On

" Credit Card Fraud Prediction"

Submitted by,

Divyani Mandale	TE, SEM VI, DIV. B, Roll No. 05
Saud Hussain	TE, SEM VI, DIV. B, Roll No. 13
Aniruddh Nagare	TE, SEM VI, DIV. B, Roll No. 15

Under the guidance of

Prof. R. A. Jolhe

Department of Information Technology

Datta Meghe College of Engineering,

Sector-3, Airoli, Navi Mumbai – 400 708, (M.S.), INDIA

2 0 2 4 - 2 5

R1 (2 marks)	R2 (2 marks)	R3 (1 mark)	Total (5 marks)	SIGNATURE

TABLE OF CONTENT

ABSTRACT

Credit card fraud poses a significant threat in the digital economy, with increasing cases driven by identity theft, phishing, and unauthorized transactions. Early detection of fraudulent activity is essential for protecting financial assets and maintaining trust in online transactions. This study employs data mining techniques and a **Random Forest classification model** to analyze key indicators of fraud, including transaction amount, location, merchant details, time patterns, device information, and IP address. Transactions are classified as fraudulent or legitimate based on these features. Results indicate that anomalous transaction times, unusual locations, and mismatched device/IP data are strong predictors of fraud. The findings highlight the potential of integrating such predictive models into real-time payment systems and banking applications to enhance security and reduce financial losses.

.

INTRODUCTION

Credit card transactions have become an integral part of modern financial systems, but with the increasing volume of online payments, the risk of fraud has also surged. Detecting fraudulent transactions in real-time is a complex yet critical task that financial institutions must address to protect user data and assets. This project investigates the use of **machine learning algorithms**, specifically **Random Forest and other classification models**, to develop an intelligent fraud detection system capable of identifying suspicious activities with high accuracy.

Traditional fraud detection relies heavily on rule-based systems and manual audits, which are often insufficient in responding to rapidly evolving fraud patterns. By leveraging data mining techniques and historical transaction datasets, this project applies **predictive analytics** to uncover hidden patterns and relationships among features such as transaction amount, location, time, device information, and user behavior. Through proper data cleaning, feature encoding, and modeling, the system effectively classifies transactions as either fraudulent or legitimate.

The importance of this project lies not only in enhancing transaction security but also in paving the way for **automated, real-time fraud prevention**. With the growing integration of AI in financial platforms, such intelligent models can be deployed within payment gateways and mobile banking apps to flag anomalies instantly. As digital transactions continue to grow, the adoption of ML-driven fraud detection systems will be crucial in **strengthening financial cybersecurity, reducing monetary losses, and increasing consumer trust** in online financial services.

PROBLEM DEFINITION

Credit card fraud is a major concern in today's digital economy. This project aims to analyze transaction data based on various parameters such as transaction amount, location, merchant category, device information, and timing. Using a **Machine Learning model**, we classify transactions as either legitimate or fraudulent based on these factors. With the increasing number of online transactions, financial fraud has become more prevalent and sophisticated. Factors such as unusual spending behavior, mismatched locations, and suspicious merchant details often indicate fraudulent activities. Detecting these anomalies early can help prevent financial losses and enhance the security of online payment systems. This study leverages **data mining techniques** to predict fraud and provides actionable insights that can assist financial institutions in making real-time, data-driven decisions.

DATA MINING TASK SELECTION

The primary data mining task in this project is **classification**, which involves categorizing financial transactions into predefined labels—**fraudulent** or **legitimate**. Classification is critical in the context of fraud detection because it allows the model to predict the likelihood of a transaction being fraudulent based on learned patterns from historical data. This targeted prediction enables timely interventions and helps prevent unauthorized financial activity.

Classification was chosen over other data mining tasks such as clustering or association rule mining because it provides clear, actionable results. While clustering groups transactions based on similarity, it does not assign specific fraud labels, making it less effective for real-time decision-making. In contrast, classification models such as **Random Forest** or **Naive Bayes** learn from labeled data and can make precise predictions on new, unseen transactions. This makes classification the most suitable approach for developing intelligent, automated fraud detection systems that enhance the security and reliability of digital payments.

DATASET DESCRIPTION

- **Source of Data**

The dataset used in this study comprises structured transaction records obtained from simulated or publicly available financial datasets tailored for fraud detection. Each record represents a transaction with various attributes related to user behavior, payment details, and contextual information. The data is designed to help train machine learning models to distinguish between fraudulent and legitimate transactions.

- **Attributes and Features in the Dataset**

The dataset contains 8000 rows and 20 relevant columns, including:

- Transaction Date and Time – Timestamp of the transaction
- Transaction Amount – Value of the transaction (Numerical)
- Cardholder Name – Encrypted or anonymized
- Card Number – Hashed/Encrypted for privacy
- Merchant Name – Where the transaction occurred
- Merchant Category Code (MCC) – Type of merchant (e.g., travel, retail)
- Transaction Location – City or ZIP code
- Transaction Currency – Currency used
- Card Type – Visa, MasterCard, etc.
- Card Expiration Date – Card's expiry details
- CVV Code – Encrypted security code
- Transaction Response Code – Processor status code
- Transaction ID – Unique identifier
- Fraud Label – 1 for fraud, 0 for non-fraud
- Previous Transactions – Historical data for the cardholder
- Transaction Source – Mobile app, web browser, etc.
- IP Address – Online transactions' originating IP
- Device Information – Device type, browser used
- User Account Info – Account-level behavior
- Transaction Notes – Comments or tags

- **Preprocessing Steps (Data Cleaning, Feature Selection, Handling Missing Data)**

1. **Data Cleaning:** Removal of duplicate or irrelevant transaction records and inconsistent values.

2. **Handling Missing Data:** Used SimpleImputer from sklearn to fill missing values in fields like 'User Account Information' using statistical techniques.

3. **Feature Selection:** Eliminated non-contributing features to enhance model efficiency and prevent overfitting.

4. **Encoding Categorical Variables:** Applied One-Hot Encoding and Label Encoding to transform text-based categories into numerical values.

5. **Normalization/Standardization:** Rescaled numerical features (e.g., transaction amount) to ensure uniformity across features.

6. **Data Splitting:** Divided the dataset into 80% training and 20% testing subsets for model training and evaluation.

ALGORITHM SELECTION

In this project, Random Forest is selected as the primary classification algorithm for detecting fraudulent credit card transactions. Random Forest is an ensemble learning technique that builds multiple decision trees on random subsets of the dataset and aggregates their outputs to enhance predictive performance. For classification tasks like fraud detection, it uses majority voting across trees to determine the final outcome..

Why Random Forest ?

- **High Accuracy**: Random Forest delivers strong classification performance, especially in imbalanced datasets like fraud detection, where it helps minimize false positives and false negatives.

- **Resistant to Overfitting**: Unlike individual decision trees, Random Forest reduces overfitting by averaging predictions across many trees trained on different data slices.

- **Feature Importance Insight**: It ranks and reveals the most influential features, such as transaction time, amount, and device info—helpful for understanding key fraud indicators.

- **Robust to Missing Data:** It can handle datasets with partial information, which is common in real-world financial data.

- **Handles Diverse Data Types**: Capable of processing both categorical (e.g., card type, merchant) and numerical (e.g., amount, time) attributes efficiently.

- **Scalable and Efficient**: Random Forest can be parallelized, making it ideal for large-scale datasets with thousands of transaction records.

ALGORITHM IMPLEMENTATION

The **Random Forest classifier** is employed to identify whether a credit card transaction is **fraudulent or legitimate** based on various transaction attributes. Below is the step-by-step implementation workflow:

- **Load the dataset:**
  - Import the dataset containing transaction records.
  - Explore the structure: rows, columns, data types, and class distribution (fraud vs. non-fraud).

- **Data Preprocessing:**
  - **Handle Missing Values:** Use techniques like `SimpleImputer` to fill missing data (e.g., user account info).
  - **Encode Categorical Variables:** Apply One-Hot or Label Encoding to features like card type, merchant category, etc.
  - **Standardize Numerical Attributes:** Normalize continuous data such as transaction amounts and timestamps to a standard scale.

- **Data Splitting:**
  - Split the dataset into:
    - **Training Set (80%)**
    - **Testing Set (20%)**
  - Ensure class balance is considered to avoid biased predictions
    - **Train the Random Forest Model:**

- **Train the Random Forest Model**
  - Define hyperparameters:
    - `n_estimators` (number of trees)
    - `max_depth` (depth of each tree)
    - `min_samples_split` (minimum samples to split a node)
  - Train the model using the training dataset.

- **Feature Selection**
  - Analyze the importance scores for each feature.
  - Identify top contributors to fraud detection like:
    - Unusual transaction times
    - Suspicious locations
    - IP address mismatches
    - High transaction amounts

.

- **Model Prediction:**
  - Predict the **fraud labels** for the test set transactions.
  - Output binary classification (0 = Legitimate, 1 = Fraud).

- **Performance Evaluation:**
  - Compute key evaluation metrics:
    - **Accuracy**
    - **Precision**
    - **Recall**
    - **F1-score**
  - Generate a **Confusion Matrix** to visualize true vs. predicted classes and identify false positives/negatives

- **Visualization:**
  - Plot **feature importance scores** to understand model decisions.
  - Display **ROC curves** or **bar charts** for model performance metrics

- **Model Deployment (Optional):**
  - Integrate the trained model into a **banking dashboard**, **payment gateway**, or **mobile app**.
  - Enable real-time detection and alerting for suspicious transactions

RESULTS AND ANALYSIS

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8000 entries, 0 to 7999
Data columns (total 20 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Transaction Date and Time             8000 non-null   object
 1   Transaction Amount                    8000 non-null   float64
 2   Cardholder Name                       8000 non-null   object
 3   Card Number (Hashed or Encrypted)     8000 non-null   object
 4   Merchant Name                         8000 non-null   object
 5   Merchant Category Code (MCC)          8000 non-null   int64
 6   Transaction Location (City or ZIP Code)  8000 non-null   object
 7   Transaction Currency                  8000 non-null   object
 8   Card Type                             8000 non-null   object
 9   Card Expiration Date                  8000 non-null   object
 10  CVV Code (Hashed or Encrypted)        8000 non-null   object
 11  Transaction Response Code             8000 non-null   int64
 12  Transaction ID                        8000 non-null   object
 13  Fraud Flag or Label                   8000 non-null   int64
 14  Previous Transactions                 8000 non-null   object
 15  Transaction Source                    8000 non-null   object
 16  IP Address                            8000 non-null   object
 17  Device Information                    8000 non-null   object
 18  User Account Information              3990 non-null   object
 19  Transaction Notes                     8000 non-null   object
dtypes: float64(1), int64(3), object(16)
memory usage: 1.2+ MB
```

```
data.describe()
```

	Transaction Amount	Merchant Category Code (MCC)	Transaction Response Code	Fraud Flag or Label
count	8000.000000	8000.000000	8000.000000	8000.000000
mean	2496.356036	5484.150375	5.637500	0.498625
std	1451.221326	2608.164617	4.928147	0.500029
min	1.090000	1000.000000	0.000000	0.000000
25%	1242.580000	3230.750000	0.000000	0.000000
50%	2492.460000	5455.000000	5.000000	0.000000
75%	3739.522500	7761.000000	12.000000	1.000000
max	4996.700000	9999.000000	12.000000	1.000000

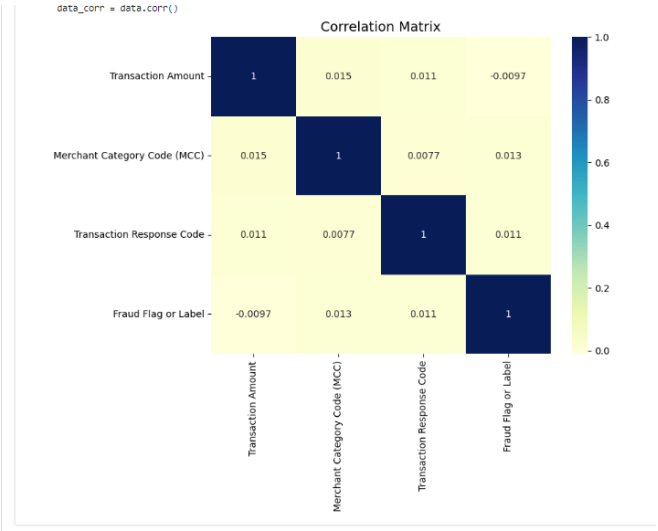
```
data.isnull().sum()
```

```
Transaction Date and Time                0
Transaction Amount                       0
Cardholder Name                          0
Card Number (Hashed or Encrypted)        0
Merchant Name                            0
Merchant Category Code (MCC)             0
Transaction Location (City or ZIP Code)  0
Transaction Currency                     0
Card Type                                0
Card Expiration Date                     0
CVV Code (Hashed or Encrypted)           0
Transaction Response Code                0
Transaction ID                           0
Fraud Flag or Label                      0
Previous Transactions                    0
Transaction Source                       0
IP Address                               0
Device Information                       0
User Account Information              4010
Transaction Notes                        0
dtype: int64
```

VISUALIZATION TECHNIQUES USED

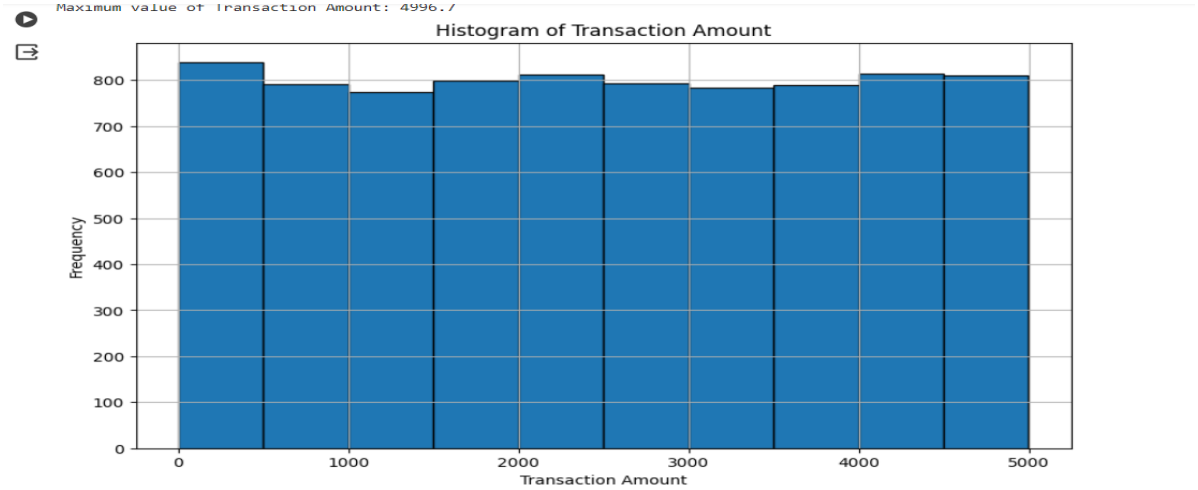
1. Feature Correlation Heatmap

A heatmap visually represents the correlation between features, helping identify strongly related variables for better model performance



2. Histogram

A histogram represents the distribution of numerical data by grouping values into bins. It helps in understanding the spread and frequency of data points.



BUSINESS INTELLIGENCE DECISION

Business Intelligence (BI) techniques enable financial institutions to make informed, data-driven decisions by uncovering key factors contributing to fraudulent activities. By analyzing structured transaction data and applying machine learning models like **Random Forest**, we derive actionable insights that enhance fraud prevention strategies and improve customer security.

Key Findings:

1. **Unusual Transaction Time as a Strong Indicator**
    - Transactions occurring at odd hours (e.g., late night or early morning) are more likely to be flagged as fraudulent.
2. **Location and Device Mismatch Raises Red Flags**
    - Significant variation between known user locations and transaction origin (IP/device) often indicates suspicious behavior.
3. **High Transaction Amounts with New Merchants Pose Risks**
    - Unfamiliar merchants combined with high transaction values have a higher probability of being fraudulent.
4. **Repeated Small Transactions Can Indicate Testing Attempts**
    - Fraudsters often test stolen card details with low-value transactions before initiating larger fraud.
5. **Customer Behavior Modeling Enhances Risk Scoring**
    - Tracking and learning normal transaction behavior improves model accuracy and helps in personalized fraud detection.

By leveraging Business Intelligence and machine learning, banks and digital payment platforms can proactively detect fraud, protect user trust, and reduce financial losses—building a more secure and resilient financial ecosystem.

CONCLUSION

This study successfully demonstrates the application of **data mining and machine learning techniques** in predicting and identifying fraudulent credit card transactions using real-world transaction data. The implementation of the **Random Forest algorithm** provided high classification accuracy, showcasing its reliability and effectiveness in detecting suspicious behavior.

The insights derived from the model emphasize the importance of features such as **transaction timing, location anomalies, device inconsistencies, and unusual merchant activity** in flagging potentially fraudulent transactions. These findings reinforce the value of intelligent fraud detection systems in enhancing financial security.

Looking forward, future enhancements may include the integration of **real-time fraud monitoring** with banking systems and mobile apps, enabling immediate detection and intervention. Moreover, the use of **larger, more diverse datasets** and the exploration of **advanced deep learning models** could further improve prediction accuracy and expand the applicability of this system across global financial networks.

By continuing to evolve and innovate, such fraud detection solutions have the potential to significantly **reduce financial losses, improve customer trust, and contribute to a safer digital payment ecosystem.**