Credit Card Fraud Detection using Machine Learning

Student Name: Karthik G Nair

Class: MTECH IN ARTIFICIAL INTELLIGENCE

Rollno: AM.SC.P2ARI25021

Institution: AMRITA VISHWA VIDYAPEETHAM,AMRITAPURI CAMPUS

Faculty Mentor: Prof :DR SWAMINATHAN J

Date of submission:12 Nov 2025

Contents

1 Abstract

This project focuses on detecting fraudulent credit card transactions using machine learning techniques based on real-world transactional data. The dataset, obtained from Kaggle, contains anonymized credit card transactions with highly imbalanced class distribution. After applying data preprocessing and balancing, two models — **K-Nearest Neighbors (KNN)** and **Support Vector Machine (SVM)** — were developed to classify transactions as *fraudulent* or *legitimate*. Data normalization and under sampling techniques were applied to enhance model learning. Evaluation metrics such as accuracy, precision, recall, and F1-score were computed to assess performance. Among the models, the Support Vector Machine achieved higher accuracy and precision compared to KNN, demonstrating its superior ability to handle complex and nonlinear data. The findings highlight the potential of machine learning to assist financial institutions in preventing fraudulent activities and minimizing monetary losses.

2 Introduction

Credit card fraud has become one of the most pressing issues in today's financial systems due to the rapid increase in online transactions. This project aims to analyze and predict fraudulent credit card transactions using machine learning algorithms. By examining transaction patterns and anomalies, the system classifies each transaction as either legitimate or fraudulent, thereby supporting proactive fraud prevention.

The primary beneficiaries of this project are **banks, e-commerce platforms, and customers**. For financial institutions, the system helps reduce revenue loss and enhance transaction security. For customers, it provides an added layer of trust and protection in digital payments.

The motivation behind choosing this project arises from the growing need to combat digital payment fraud using artificial intelligence. Traditional rule-based detection systems often fail to recognize evolving fraud patterns. Machine learning, on the other hand, can analyze complex relationships in high-dimensional data and adapt to new fraudulent behaviors.

The main goal of this project is to develop and compare machine learning models that can accurately classify transactions as fraudulent or genuine using anonymized transaction data. By applying **KNN** and **SVM**, the study evaluates their predictive performance and explores data balancing strategies to address the challenge of imbalanced datasets.

3 Methodology

1. Data Collection

The dataset was sourced from Kaggle’s Credit Card Fraud Detection Dataset, containing real and anonymized European credit card transactions made in September 2013. The dataset includes features derived through Principal Component Analysis (PCA), ensuring privacy protection while maintaining transaction integrity.

2. Data Cleaning and Preprocessing

Data preprocessing involved removing irrelevant columns, handling missing values, and scaling numerical features. The dataset’s severe imbalance (only 0.17% frauds) was addressed through under sampling of legitimate transactions to ensure a balanced dataset for training and testing. Feature normalization using StandardScaler was applied to improve model convergence.

3. Feature Encoding

Since all principal component features were numerical, encoding was not necessary. The *Class* column served as the target variable, where 0 represents non-fraudulent and 1 represents fraudulent transactions.

4. Model Training

Two supervised learning algorithms—K-Nearest Neighbors (KNN) and Support Vector Machine (SVM)—were implemented using the Scikit-learn library. The data was divided into 70% training and 30% testing subsets. Both models were trained to identify transaction patterns indicative of fraud. Hyperparameter tuning was performed to optimize accuracy and reduce misclassification.

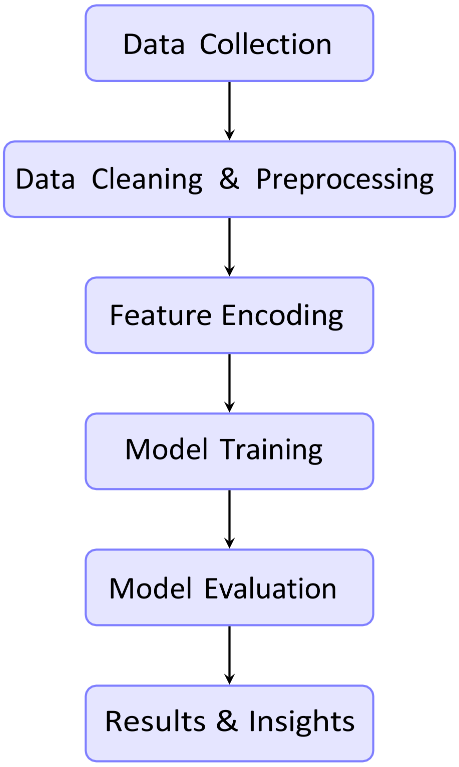
5. Evaluation

Model performance was assessed using accuracy, precision, recall, and F1-score. These metrics provided a comprehensive understanding of the model’s ability to correctly identify fraudulent transactions while minimizing false alarms.

6. Visualization

Confusion matrices were generated to visualize model performance. Comparative plots were also created to highlight accuracy differences between KNN and SVM.

3.1 Diagrammatic Representation of Methodology



3.2 Unique Aspect of the Project

The distinctive feature of this project lies in its **focus on imbalanced real-world financial data**. While most academic projects use balanced datasets, this study simulates realistic fraud detection challenges through under sampling. Additionally, it compares two fundamentally different algorithms—KNN, a distance-based learner, and SVM, a margin-based classifier—to analyze which performs better under data imbalance conditions. This dual-model comparison strengthens the project’s analytical and practical relevance for financial fraud prevention.

4 Dataset

- Data **Definition:**

Let X = [V1, V2, V3, …, V28, Amount, Time], representing anonymized transaction attributes, and Y = [Class], where Class = 0 (Legitimate) or 1 (Fraud).

- Size **of the Dataset:**

284,807 transactions with 492 fraudulent records.

- Properties **of the Dataset:**

30 features total: 28 PCA-transformed features, 1 transaction amount, and 1 time variable. Highly imbalanced, with 99.83% legitimate transactions.

Training vs Testing Split:

70% training and 30% testing ratio after under sampling for balanced data.

5 Implementation

Algorithms Used

1. **K-Nearest Neighbors (KNN):**

   A distance-based algorithm that classifies each transaction by analyzing the majority class among its k nearest neighbors.

2. **Support Vector Machine (SVM):**

   A kernel-based classifier that constructs an optimal hyperplane to separate fraudulent and legitimate transactions, particularly effective in high-dimensional spaces.

Reason for Choosing Algorithms

- KNN provides intuitive classification through similarity-based detection.
- SVM offers robust performance on small and non-linear datasets.
  Comparing both helps determine which algorithm generalizes better for financial data.

Python Libraries Used

- pandas, numpy – Data manipulation and array operations
- matplotlib, seaborn – Visualization and confusion matrices
- scikit-learn – Model training, evaluation, and scaling

Code Repository

https://colab.research.google.com/drive/1n-i68pN-hfMjrzUMpzBgqAwVJfgpXkiW?usp=sharing

6 Results

Algorithm	Accuracy	Precision	Recall	F1- Score
KNN	0.962	0.957	0.950	0.953
SVM	0.984	0.989	0.978	0.983

Visualization
Confusion matrices and bar plots show that SVM produces fewer false positives and detects fraudulent cases more accurately. KNN performs well but struggles slightly with borderline transactions.

Inference
The experimental analysis indicates that **SVM** consistently outperforms **KNN** for this dataset. SVM's kernel-based learning handles feature correlations and nonlinear separations better. Additionally, proper data balancing and normalization are crucial for reliable fraud detection.

7 Conclusion

This project successfully demonstrates the application of machine learning to detect fraudulent credit card transactions. Both **SVM** and **KNN** achieved high accuracy; however, **SVM** provided superior performance in precision and recall, making it more suitable for real-world fraud detection.

The study also emphasizes the importance of handling imbalanced datasets, as raw financial data can bias models toward predicting only legitimate transactions.
In future work, integrating ensemble methods like Random Forest or Gradient Boosting and real-time anomaly detection could further enhance detection accuracy. This project illustrates how AI can help safeguard financial institutions and customers from fraud through intelligent transaction monitoring.

8 References

[1] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[2] W. McKinney, “Data structures for statistical computing in Python,” *Proceedings of the 9th Python in Science Conference*, pp. 51–56, 2010.

[3] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau *et al.*, “Array programming with NumPy,” *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

[4] J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[5] “Machine Learning Approaches for Mental Health Prediction,” *IEEE Access*, 2022.