Fraud Detection – Data Science Project Report

Project Title

Fraud Detection in Financial Transactions (Classification Problem)

Project Goal

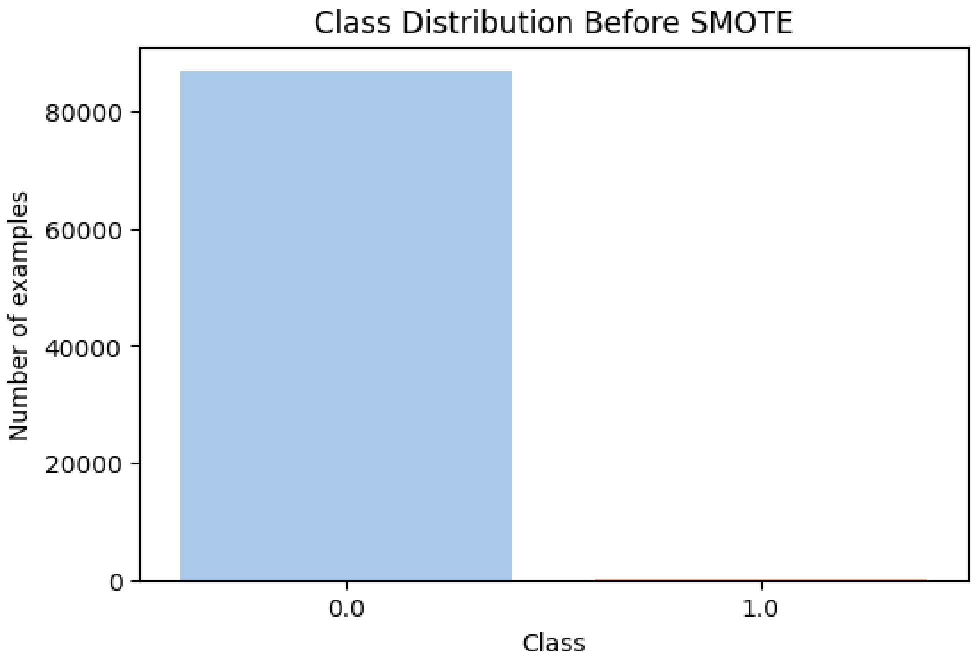
The goal of this project is to detect fraudulent financial transactions using Machine Learning and Neural Network models.
The task is a **binary classification problem**, where the model predicts whether a transaction is:

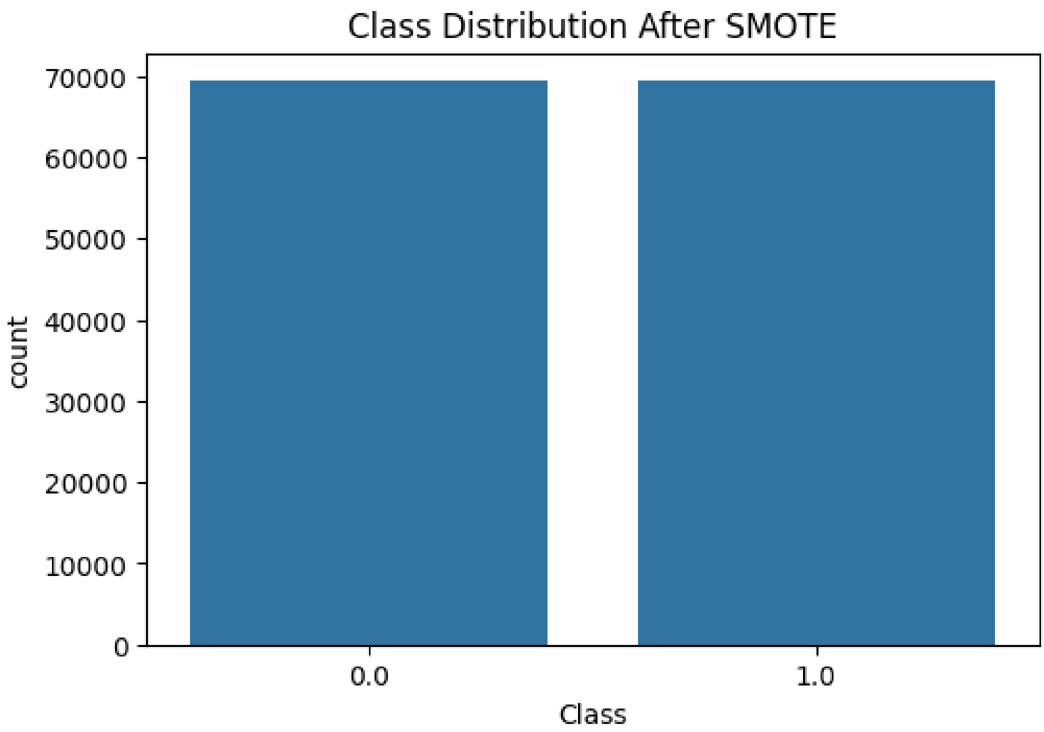
- **0 → Legitimate (Non-Fraud)**
- **1 → Fraudulent**

Dataset Description

- The dataset consists of anonymized numerical features named:
  V1, V2, …, V28
- Includes a “Class” column (0 = normal, 1 = fraud).
- The dataset is highly **imbalanced**, with fraudulent cases being much fewer.

Before SMOTE





Project Flow

Dataset Preparation

- Loaded the dataset.
- Separated features (X) and target (y).
- Split into training and testing sets
- **Train:** 80% **Test:** 20%

Data Preprocessing

Performed the following preprocessing steps:

1. **Handling Missing Values**
   – There were no missing values in this dataset.

2. **Handling Imbalanced Data using SMOTE**

   - Applied SMOTE to oversample the minority fraud class.

3. **Feature Scaling**

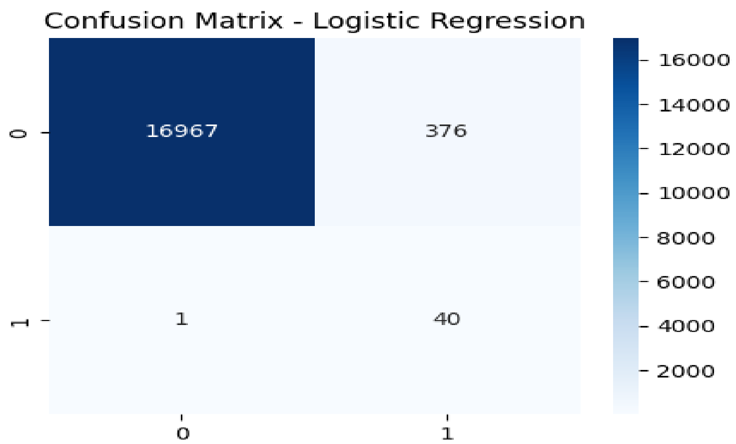
   - Used StandardScaler to normalize numerical features.

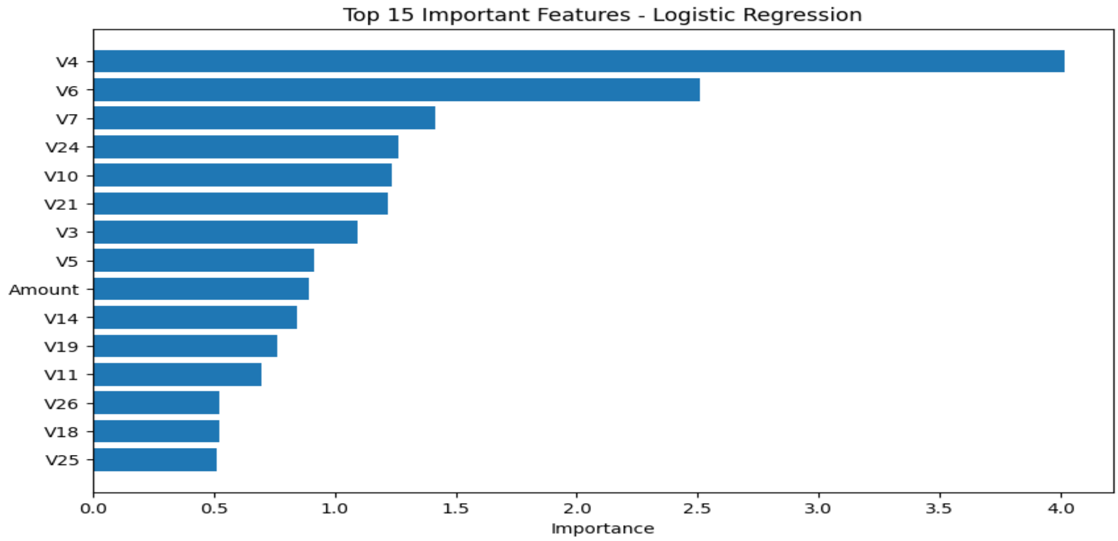
4. **Checking Class Balance**

Classical Machine Learning Models

Logistic Regression

- **Accuracy: 98%**
- **Precision (Fraud): 0.10**
- **Recall (Fraud): 0.98**
- **F1-Score: 0.18**

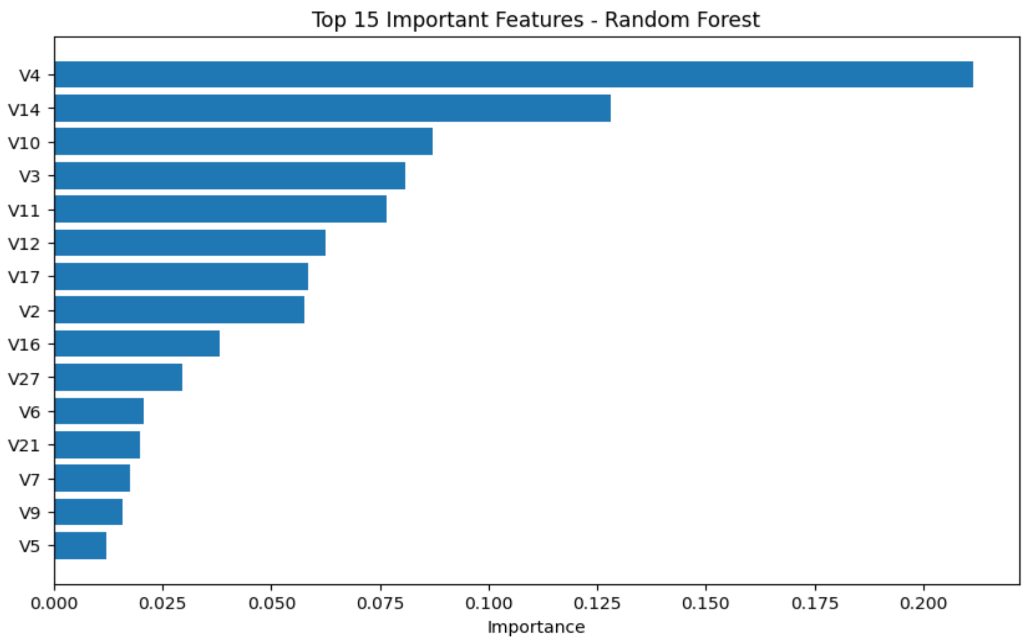


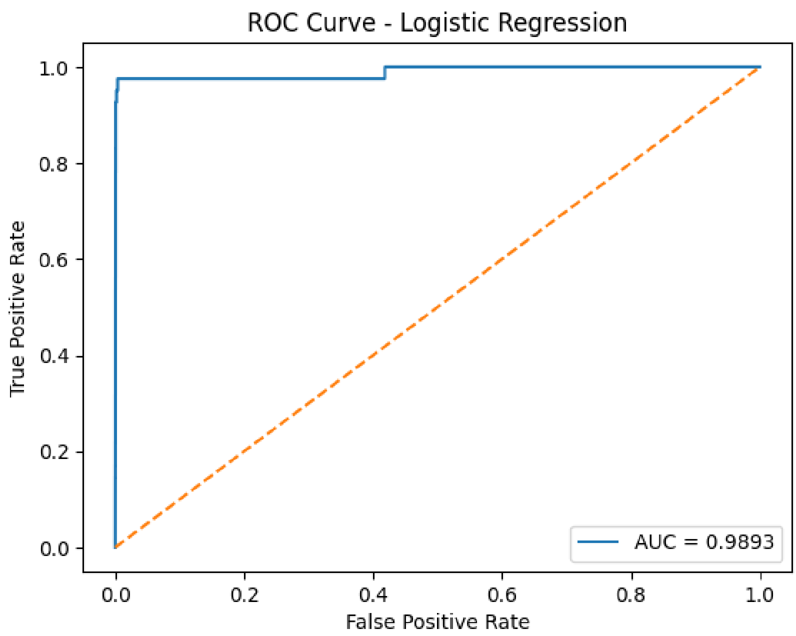


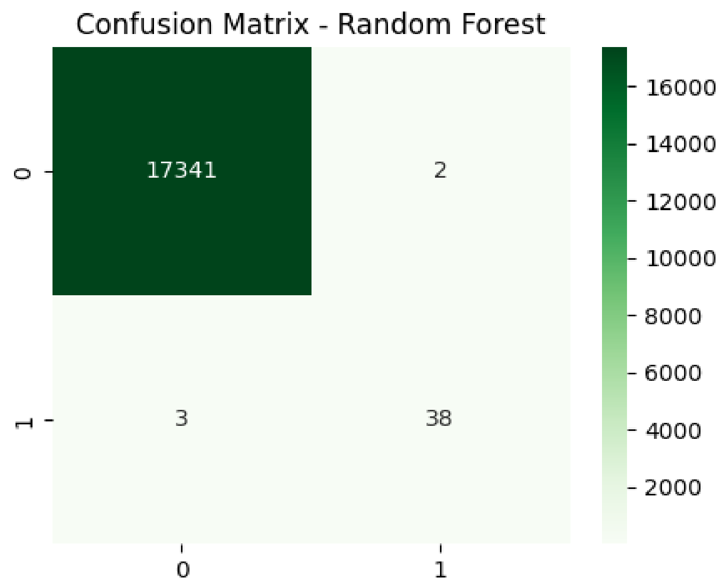
Although Logistic Regression achieved high overall accuracy, it performs poorly in precision for fraud detection.
This means it classifies almost all fraud correctly (high recall), but predicts many false positives, making it unreliable for real fraud systems.

Random Forest Classifier

- Accuracy: ~100%
- Precision (Fraud): 0.95
- Recall (Fraud): 0.93
- F1-Score: 0.94







Random Forest performs extremely well in detecting fraud.
It achieves a strong **balance between precision and recall**, making it the best classical model in this project.

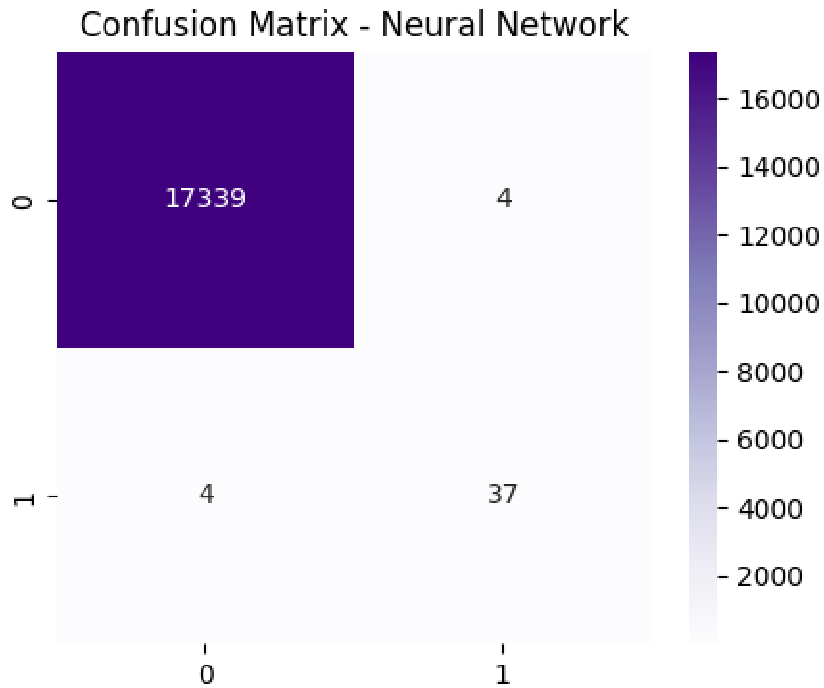
conclusion:

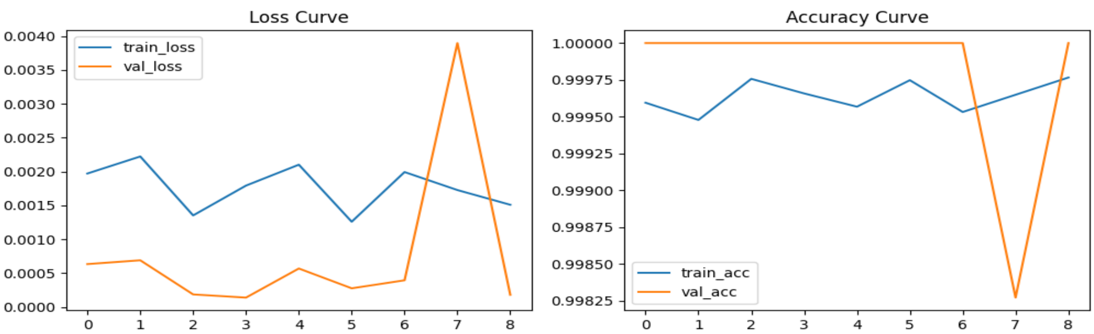
- **Random Forest is the best classical model** in terms of balanced performance.
- It significantly outperforms Logistic Regression, especially in detecting minority fraud cases.

Neural Network Model

Experiment 1

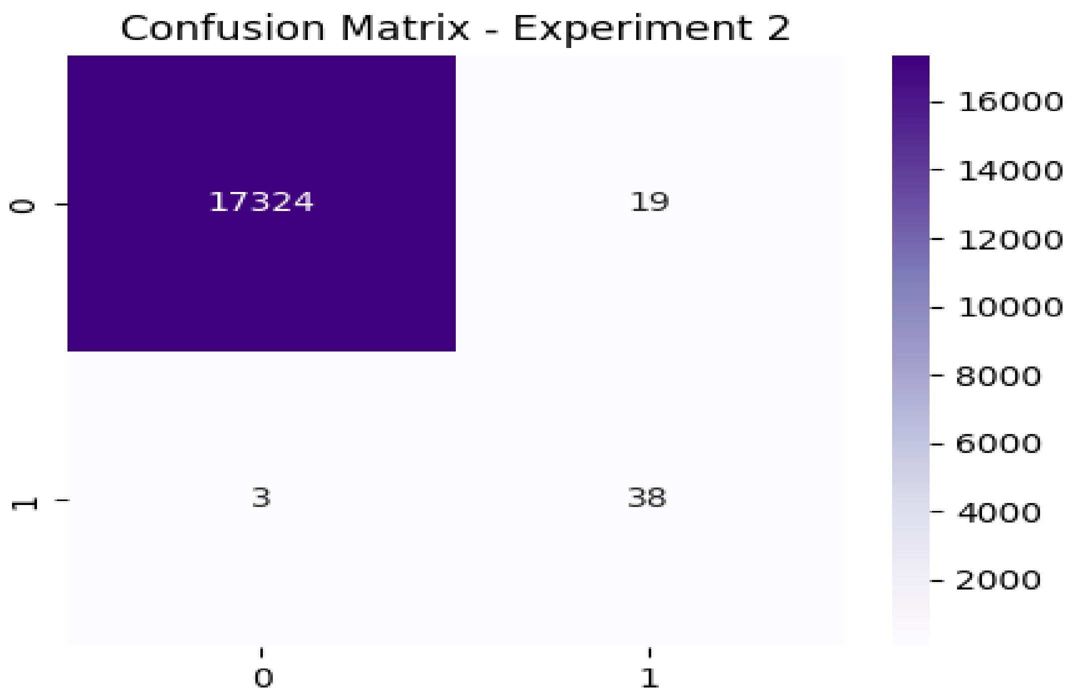
- Layers: [32, 16]
- Dropout: 0.2
- Optimizer: Adam(0.001)
- Batch size: 32
- Epochs: 50

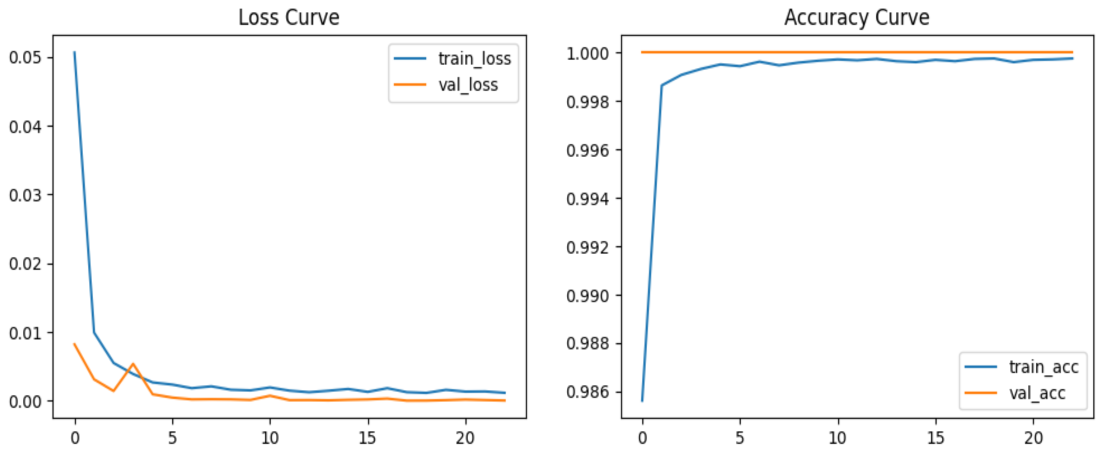




Experiment 2

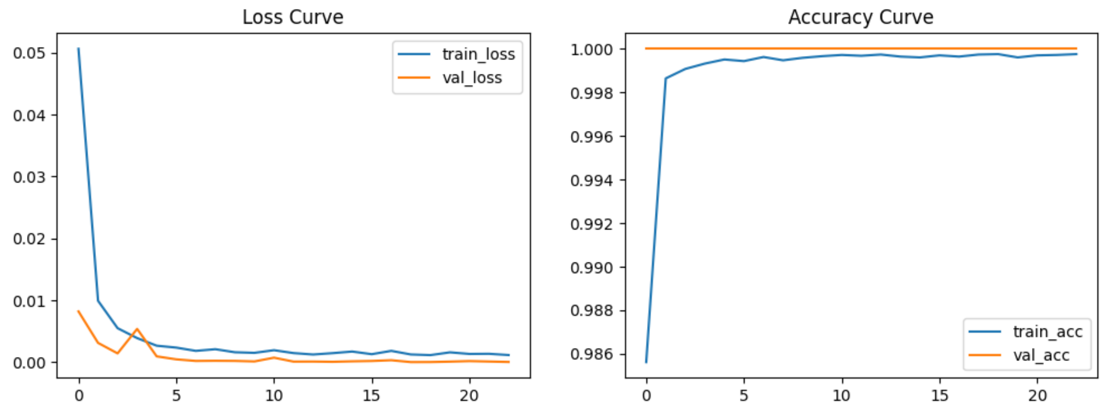
- Layers: [64, 32, 16]
- Dropout: 0.3
- Optimizer: RMSprop(0.0005)
- Batch size: 32

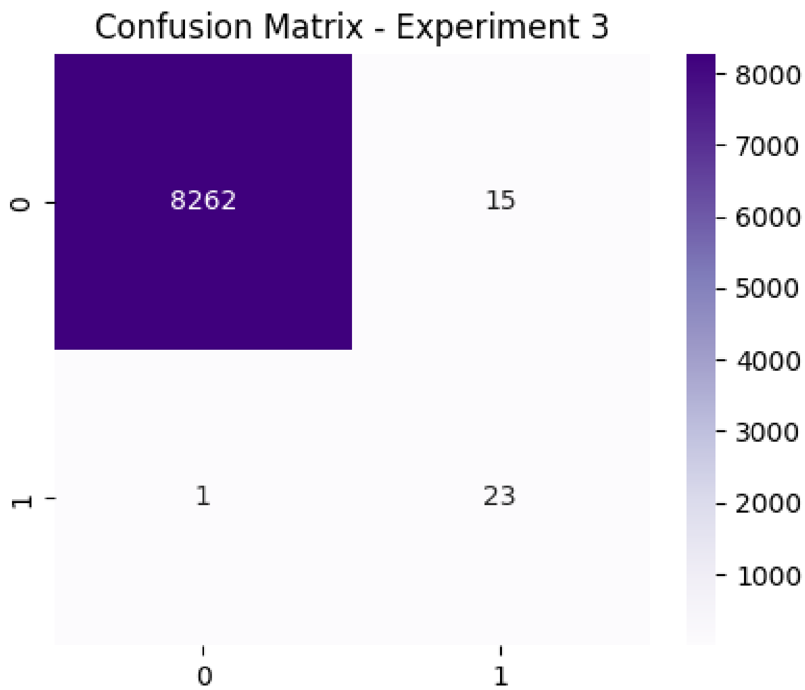




Experiment 3

- Layers: [128, 64]
- Dropout: 0.3
- Optimizer: SGD + momentum
- Batch size: 16





Hyperparameter Tuning Summary

Experiment	Layers	Dropout	Learing_rate	Optimizer	Batch	Notes
1	**[32,16]**	0.2	0.001	Adam	32	Best precision + stable performance
2	[64,32,16]	0.3	0.0005	RMSprop	32	Best balance between recall & precision
3	[128,64]	0.3	0.01	SGD	16	Highest recall but lower precision

Adjusting hyperparameters like learning rate, layers, dropout, and optimizer significantly affected performance. Experiment 2 provided the best balance for detecting fraud.

Final Model Selection

Best Model: Experiment 2 Neural Network

- Achieves high precision and high recall
- Avoids overfitting
- More balanced than Experiment 1 or 3.

Prediction on Unseen Data

- Model was tested on new unseen examples.
- The model successfully identified fraudulent patterns accurately.

Comparative Summary (Classical vs Neural Network)

The project included two classical machine learning models (Logistic Regression and Random Forest) and three Neural Network experiments.

Random Forest achieved the highest performance among classical models, with excellent precision and recall for fraud detection.

The Neural Network experiments showed competitive performance and were able to detect fraud effectively, especially with tuned architectures and dropout layers.

Comparing both categories, Random Forest and the optimized Neural Network configurations deliver strong results, while Logistic Regression falls behind due to low precision for the fraud class.

- model is suitable for real-world fraud detection

References

▪ Dataset: Credit Card Fraud Detection Dataset 2023 (Kaggle)
▪ UCI Machine Learning Repository
▪ SMOTE — imblearn documentation
▪ Keras / TensorFlow documentation
▪ Scikit-learn documentation

Dataset: Credit Card Fraud Detection Dataset 2023
Project Notebook:
https://colab.research.google.com/drive/10cyqdVF7_ZeonYUb2tpzfjJuzR0FT8iG?usp=sharing

Bonus Points Achieved

1. Model Enhancements / Modifications

- Applied **Early Stopping** to prevent overfitting.

- Added **Dropout layers** to improve generalization

2-Using a Cloud Platform

._Used **Google Colab** as a cloud platform to execute and experiment with the model.

_Although CPU was used due to the project being lightweight, utilizing Colab helped **facilitate the pipeline and ensure reproducibility of results**.