







# Credit Card Fraud Detection

A machine learning project to detect fraudulent credit card transactions using Logistic Regression. The model was trained and tested on an imbalanced dataset to evaluate its performance.

## Objectives

The primary objectives of this project are:

-  **Detect fraudulent credit card transactions** using a supervised machine learning algorithm.
-  **Handle class imbalance** in the dataset where fraudulent cases are significantly fewer than legitimate ones.
-  **Build and train a Logistic Regression model** to classify transactions as fraudulent (1) or legitimate (0).
-  **Evaluate the model** using not just accuracy, but also **precision, recall, and F1-score**, which are more appropriate for imbalanced datasets.
-  **Generate insights** from model performance to understand strengths and areas for improvement in fraud detection.
-  Lay the foundation for future enhancements such as:
  - Incorporating more advanced models (e.g., Random Forest, XGBoost)
  - Using resampling techniques like **SMOTE**
  - Deploying the model for real-time fraud detection

## ## Project Overview

Credit card fraud detection is essential for minimizing losses in the financial industry. In this project, we built a classification model using **Logistic Regression** to predict whether a transaction is fraudulent or not.

## ## Dataset

The dataset used is from [Kaggle - Credit Card Fraud Detection] (<https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>), which contains transactions made by European cardholders in September 2013.

- Total Records: 284,807

- Features: 30 (V1-V28 PCA components + Time, Amount)

- Target: `Class` (0 for normal, 1 for fraud)

### **Technologies Used**

- Python
- Pandas, NumPy
- Scikit-learn (LogisticRegression, train\_test\_split, accuracy\_score)
- Jupyter Notebook / VS Code

### ## **Model Training**

```
python
```

```
model = LogisticRegression(solver='liblinear', max_iter=1000)
```

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

### **Key Insights**

- **High Precision (0.98) for Fraud Class:** Very few legitimate transactions were wrongly flagged as fraud.
- **Strong F1-Score (0.91) for Fraud Detection:** Balanced performance between precision and recall.
- **Slightly Lower Recall (0.86):** Indicates a few fraudulent transactions were missed.
- Overall, the model performs reliably in identifying frauds with minimal false alarms