Credit Card Default Prediction Analysis

By Akhilesh Deepak Jichkar – 2025AA05613
Github Link: AkhiJix/ML_Assignment_2 Streamlink App: ML Model Evaluator · Streamlit

a. Problem Statement

The objective of this project is to develop a robust machine learning system to predict the likelihood of credit card default for clients based on their demographic data and payment history.
Financial institutions face significant risks when lending money. Predicting which clients are likely to default in the next month allows the bank to take proactive measures such as adjusting credit limits or engaging in early intervention to minimize financial losses. This project implements and compares six different classification models to identify the most effective solution for this task.

b. Dataset Description

The analysis uses the "Default of Credit Card Clients" dataset from the UCI Machine Learning Repository.

- **Source:** UCI Machine Learning Repository.
- **Sample Size:** 30,000 observations (clients).
- **Features (X):** 23 attributes including Age, Sex, Education, Credit Limit, Repayment Status, and Bill Amounts.
- **Target Variable (Y):** 'default.payment.next.month' (1 = Default, 0 = No Default).
- **Class Imbalance:** The dataset contains significantly more non-defaulters than defaulters.

c. Models Used

The following six supervised learning algorithms were implemented and evaluated on the same training/testing split. Random Forest and XGBoost emerged as the top performers.

Model Name	Accuracy	AUC Score	Precision	Recall	F1 Score	MCC Score
Logistic Regression	0.807667	0.707636	0.686825	0.239638	0.355307	0.324443
Decision Tree	0.809667	0.718170	0.622193	0.354936	0.452015	0.367154
KNN Classifier	0.792833	0.701435	0.548724	0.356443	0.432161	0.323267
Naive Bayes	0.752500	0.724930	0.451474	0.553881	0.497462	0.338620
Random Forest	0.813333	0.751038	0.636005	0.364732	0.463602	0.380948
XGBoost	0.811833	0.756469	0.628906	0.363979	0.461098	0.376398

d. Observations

ML Model Name	Observation about model performance
Logistic Regression	Achieved a baseline accuracy of 80.7% but had the lowest Recall (0.24). This suggests that as a linear model, it struggled to capture the non-linear relationships in payment history, significantly under-predicting the minority class (defaulters).
Decision Tree	Showed improved Recall (0.356) over Logistic Regression by capturing non-linear splits in the bill amount and payment status. However, it is prone to higher variance, leading to an MCC score (0.368) that is slightly lower than the ensemble methods.
kNN	Performance was highly sensitive to feature scaling. While it matched the Decision Tree in Recall (0.356), its lower Precision (0.548) suggests it produced more False Positives, likely due to the high dimensionality of the 23 features (the "Curse of Dimensionality").
Naive Bayes	The "Outlier" of the group. It had the lowest Accuracy (75.2%) but the highest Recall (0.55). This indicates that while it's less precise, it is the most aggressive at identifying potential defaulters, making it useful if the "cost" of a missed default is very high.
Random Forest (Ensemble)	Emerged as the top performer with the highest MCC Score (0.378) and Accuracy (81.3%). By averaging multiple decision trees, it successfully reduced overfitting and handled the imbalanced nature of the dataset more robustly than single-tree models.
XGBoost (Ensemble)	Performed nearly identical to Random Forest (MCC: 0.376). It excelled in the AUC Score (0.756), proving that its gradient boosting approach is highly effective at distinguishing between classes by iteratively correcting the errors of previous weak learners.

e. Screenshot