

# Milestone 4 – Advanced Model Optimization Report

## Introduction

Milestone 4 focuses on training advanced machine-learning models, tuning them, comparing their performance, and generating evaluation graphs. The milestone enhances prediction accuracy over baseline results using boosting algorithms and hyperparameter optimization. The graph generation process is handled separately for cleaner execution and to avoid unnecessary retraining.

## 1. Dataset Loading and Preprocessing

- Loaded the cleaned dataset from: **data/loan\_data\_clean.csv**
- Separated numerical and categorical features
- Automatically removed high-cardinality categorical columns (more than 100 unique values)
- Applied:
  - **StandardScaler** to numerical columns
  - **OneHotEncoder (sparse\_output=False)** to safe categorical columns
- Built a unified preprocessing pipeline for all models

## 2. Train–Test Split

- Dataset split into **80% training** and **20% testing**
- Used **stratified sampling** to maintain class balance
- Ensured fair testing and unbiased model evaluation

### **3. Model Training**

Three models were trained using identical preprocessing steps:

#### **a) Logistic Regression (Baseline)**

- Serves as the benchmark
- Provides a reference AUC score

#### **b) XGBoost Classifier**

- Gradient boosting algorithm optimized for tabular data
- Commonly used for structured datasets

#### **c) LightGBM Classifier**

- Faster and lighter than XGBoost
- Good baseline for gradient boosting performance

Each model's AUC score on the test set was recorded.

### **4. Hyperparameter Tuning**

Performed hyperparameter optimization using **RandomizedSearchCV** on the XGBoost model.

Parameters tuned included:

- Number of estimators
- Learning rate
- Maximum depth
- Column sampling
- Row sampling

The tuned model achieved a higher ROC-AUC score compared to default parameters.

## 5. Model Comparison and Best Model Selection

Compared four AUC results:

- Logistic Regression
- XGBoost
- LightGBM
- Tuned XGBoost

The highest-scoring model was selected as the **final best model**, which was then saved to:

`models/best_model.pkl`

## 6. Graph Generation (Executed Separately)

Graph generation was separated into a dedicated script for flexibility.

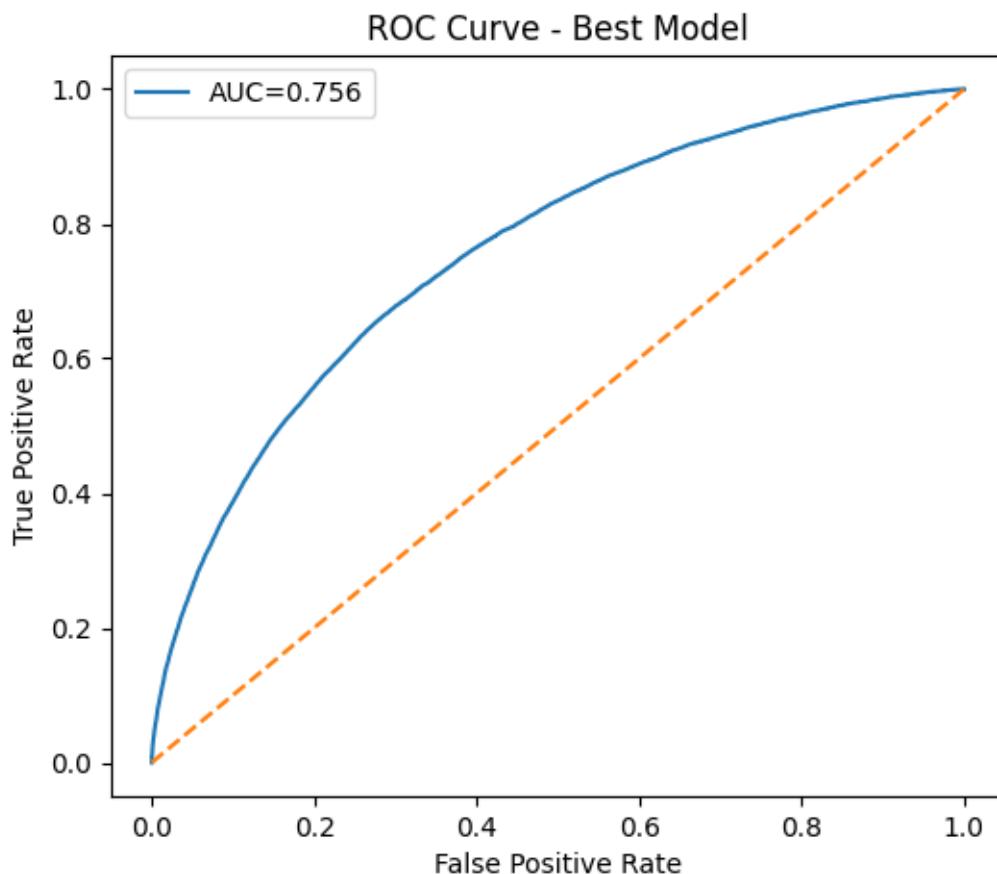
The graph script:

- Loads the saved best model
- Computes predictions
- Generates two evaluation visualizations:
  - **ROC Curve**
  - **Confusion Matrix**
- Saves the output plots into the **models/** directory.

## 7. Graph Descriptions

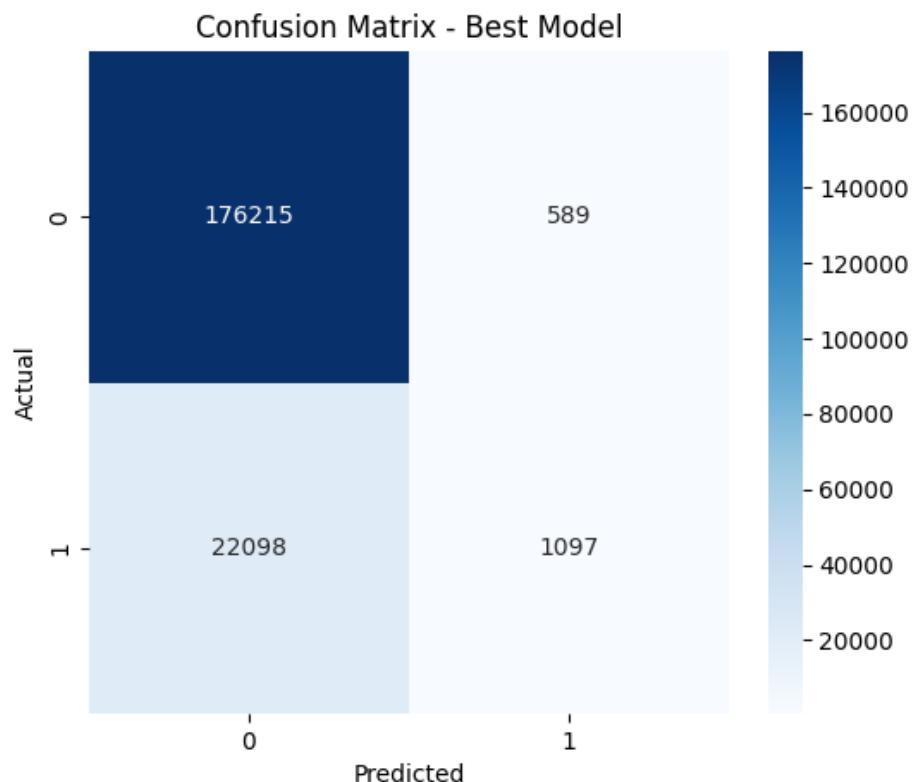
### ROC CURVE (RECEIVER OPERATING CHARACTERISTIC CURVE)

- Shows the model's ability to distinguish between default and non-default cases.
- The curve plots the **True Positive Rate (TPR)** against the **False Positive Rate (FPR)** at various threshold settings.
- The **AUC (Area Under the Curve)** value indicates model performance:
  - **AUC closer to 1.0 = Excellent model**
  - **AUC around 0.5 = No better than random guessing**
- This graph helps visually evaluate how well the selected model separates the two classes.



## CONFUSION MATRIX

- Displays the count of correct and incorrect predictions.
- Contains four key values:
  - **True Positives (TP)**
  - **True Negatives (TN)**
  - **False Positives (FP)**
  - **False Negatives (FN)**
- Helps understand:
  - How many defaults were correctly identified
  - How many non-defaults were misclassified
  - Whether the model is biased toward a particular class
- Useful for evaluating real-world reliability, especially for imbalanced datasets.



## **Conclusion**

Milestone 4 successfully implemented advanced machine learning techniques including model training, boosting algorithms, and hyperparameter tuning. The best-performing model was saved for future use, and evaluation graphs were generated separately to maintain workflow clarity. These improvements significantly enhanced the overall prediction performance of the system.