Project Report: Predicting Credit Card Default

Abstract

This study builds a supervised classification pipeline to predict default payments on credit cards using a real-world dataset with 23 explanatory variables and a binary target. The workflow covers data audit, preprocessing (label fixing, rescaling, and class-imbalance handling via SMOTE), exploratory data analysis (EDA), baseline modeling (Logistic Regression, Decision Tree, Random Forest, SVM, Gradient Boosting, XGBoost), and model selection via cross-validation with hyperparameter tuning. Performance was assessed using accuracy, precision, recall, F1, AUC, ROC curves, and confusion matrices. Baseline Random Forest achieved the strongest test-set performance; after cross-validation and hyperparameter tuning, XGBoost obtained the best test accuracy (87.10%) and AUC (0.874), indicating robust discrimination.

1. Introduction

Accurate prediction of credit card default supports proactive risk management and portfolio health monitoring. The objective is to build and compare modern classification algorithms and select a model that balances generalization performance and interpretability. The study also documents the end-to-end engineering choices required to deploy such a pipeline.

2. Data Description

The dataset contains demographic variables (e.g., SEX, EDUCATION, MARRIAGE, AGE), historical repayment status ('PAY_*'), billing amounts ('BILL_AMT1–BILL_AMT6'), prior payments ('PAY_AMT1–PAY_AMT6'), and the binary target 'IsDefaulter'. An initial audit confirmed no missing values or duplicates. Variable labels were standardized for readability.

3. Methodology

3.1 Preprocessing

・Relabeling and encoding: categorical codes were mapped to human-readable labels where appropriate.

- Class imbalance: Synthetic Minority Over-sampling Technique (SMOTE) was applied to create a balanced training set.
- Scaling: Standardization (zero mean, unit variance) was applied to features prior to model fitting where required.
- Train/test split: A stratified split (80/20) preserved the original class ratio in the test set (random_state=42).

3.2 Baseline Models

The following classifiers were trained and evaluated on the held-out test set: Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, Gradient Boosting, and XGBoost. Metrics computed included train/test accuracy, precision, recall, F1, AUC, and confusion matrices. ROC curves were plotted to compare discriminative power.

3.3 Model Selection & Tuning

Model selection considered both performance and overfitting risk. Grid/Randomized search with cross-validation was used for:

- Logistic Regression: penalty, C, max_iter.
- Decision Tree: max_depth, min_samples_split, min_samples_leaf.
- SVM: C, kernel (RBF).
- Random Forest: n_estimators, max_depth, min_samples_split, min_samples_leaf.
- Gradient Boosting: n_estimators, learning_rate, max_depth.
- XGBoost: key hyperparameters via randomized search (scoring=ROC-AUC).

4. Exploratory Data Analysis (EDA)

Target variable distribution showed class imbalance (fewer defaulters than non-defaulters). Category-wise inspection (SEX, EDUCATION, MARRIAGE, AGE bands) revealed differences in default rates across groups. A correlation heatmap highlighted strong relationships among adjacent billing amounts, payment amounts, and repayment status variables. These insights informed feature engineering choices and motivated imbalance mitigation.

5. Experiments & Results

5.1 Baseline Comparison

Among baseline models, Random Forest achieved the strongest combination of test accuracy, F1, and AUC. Decision Tree and Random Forest exhibited larger train–test gaps, indicating overfitting risk in their untuned forms.

5.2 Tuned Models

Hyperparameter tuning markedly improved generalization. Notably, the tuned XGBoost model achieved a test accuracy of 87.10% and an AUC of 0.874 (as reported in the notebook). These values exceeded other tuned baselines, making XGBoost the preferred model.

5.3 Error Analysis

Confusion matrices across models showed the classical precision–recall trade-off: models with higher recall for defaulters incurred more false positives. ROC curves illustrated superior ranking performance for ensemble methods (Random Forest, XGBoost).

6. Feature Importance & Interpretation

Random Forest feature importance (top-15 chart) indicated that recent repayment status (e.g., PAY_0/PAY_2), recent billing amounts (BILL_AMT1–BILL_AMT6), and recent payment amounts (PAY_AMT1–PAY_AMT6) were influential predictors. These align with domain expectations: recent delinquency signals, current outstanding balances, and repayment behavior are strongly associated with near-term default risk.

7. Discussion

SMOTE improved class balance and helped models learn the minority class decision boundary; however, careful validation was used to avoid synthetic data leakage into the test set. Standardization benefitted SVM and Logistic Regression. Ensemble methods, especially XGBoost, captured non-linear interactions and delivered superior AUC. Overfitting observed in tree-based baselines was mitigated via depth and leaf constraints and cross-validated tuning.

8. Limitations

Future work should include calibration analysis (e.g., reliability curves), fairness checks across demographics, and cost-sensitive evaluation given asymmetric misclassification costs in credit risk.

9. Conclusion

An end-to-end pipeline for credit default prediction was developed and compared across multiple algorithms. Baseline Random Forest performed best among initial models, yet cross-validated tuning crowned XGBoost as the final choice (Test Accuracy: 87.10%, AUC: 0.874). The study underscores the value of imbalance handling, proper scaling, and systematic hyperparameter optimization for high-stakes financial classification tasks.

10. Recommendations

- Adopt the tuned XGBoost model for deployment, with periodic retraining and drift monitoring.
- Add probability calibration (Platt/Isotonic) for threshold setting aligned to business costs.
- Implement cost-sensitive thresholds to reduce false negatives (missed defaulters) under specified loss matrices.
- Conduct stability selection / SHAP analysis for robust interpretability.
- Log model inputs/outputs for auditability and performance governance.