Project 3

Home Credit Default Risk

1. Introduction

In the financial world, risk assessment is critical for institutions that provide loans to customers. One of the most important concerns is evaluating whether a borrower is likely to default. The objective of this project is to build a predictive model that can assess a loan applicant's ability to repay the loan. By leveraging historical data from Home Credit's previous applicants, we aim to predict the likelihood of default and assist banks in making informed lending decisions.

This project uses a real-world dataset provided by Home Credit, consisting of multiple interlinked tables that contain information about the clients' applications, previous credit history, cash balances, and payment behaviors. Through extensive data cleaning, exploratory data analysis, feature engineering, and model building, we aim to develop a robust solution for default risk classification.

2. Data Understanding

The dataset is made up of several CSV files, with the primary file being application_train.csv. This file contains the current loan applications, labeled by the target variable TARGET (0 = no default, 1 = default). Other auxiliary datasets provide historical and behavioral data of the applicants:

- bureau.csv & bureau_balance.csv: Credit history from other institutions.
- previous_application.csv: Previous loan applications and their status.
- POS_CASH_balance.csv: Point of sale and cash loan balance history.
- installments_payments.csv: Payments made for previous loans.
- credit_card_balance.csv: Monthly credit card balances.

Each file contains a common client identifier (SK_ID_CURR or SK_ID_BUREAU) used for joining. The datasets have varying shapes and degrees of missing data.

3. Data Preprocessing

3.1 Missing Value Treatment

A significant challenge was the presence of missing values across almost all datasets. Here's a summary of how missing values were handled:

- Features with **>40% missing values** such as OWN_CAR_AGE, AMT_REQ_CREDIT_BUREAU_*, and many categorical features were **dropped** if they showed low correlation with the target.
- For features like DAYS_EMPLOYED, which had a large number of 365243 values (indicating missing), we **imputed with median**.
- Numeric missing values were imputed using:
    - Median for skewed distributions.
    - Mean for normally distributed features.
- Categorical missing values were imputed with 'Unknown' or mode.

3.2 Outlier Detection and Treatment

- Outliers in AMT_INCOME_TOTAL, AMT_CREDIT, and DAYS_EMPLOYED were identified using boxplots and z-score thresholds.
- Values exceeding 3 standard deviations were **clipped** to the 99th percentile to minimize their influence.
- Visualizations were used to validate the impact of these changes on model performance.

3.3 Feature Transformation

- DAYS_BIRTH, DAYS_EMPLOYED, DAYS_REGISTRATION were converted from negative days to positive integers.
- Ratios were created like CREDIT_INCOME_RATIO, ANNUITY_INCOME_RATIO, and EMPLOYMENT_AGE_RATIO.

4. Exploratory Data Analysis (EDA)

4.1 Target Variable Distribution

- Target variable (TARGET) is **highly imbalanced**:
    - 0 (no default): ~92%
    - 1 (default): ~8%

- Addressed later during model training using class weights and oversampling.

4.2 Univariate Analysis

- **Income and Credit**: Most applicants fall under income categories of 1.5–2.5 lakhs; credit amounts follow a similar distribution.
- **Age**: Most defaulters fall in the 25–35 years range.
- **Education**: Applicants with **lower education levels** tend to have higher default rates.
- **Employment Type**: Lower default rate observed in **State servants**.

4.3 Bivariate Analysis

- AMT_CREDIT vs TARGET: Higher loan amounts correlate with increased risk.
- DAYS_EMPLOYED vs TARGET: Irregular employment history is a risk factor.
- NAME_CONTRACT_TYPE: Cash loans are riskier than revolving loans.

5. Feature Engineering

Feature engineering was a major focus for improving model performance:

5.1 New Features Created

- **Ratio Features**:
  - CREDIT_INCOME_RATIO = AMT_CREDIT / AMT_INCOME_TOTAL
  - ANNUITY_INCOME_RATIO = AMT_ANNUITY / AMT_INCOME_TOTAL
  - EMPLOYMENT_AGE_RATIO = DAYS_EMPLOYED / DAYS_BIRTH

6. Modeling and Evaluation

6.1 Algorithms Used

Several classification models were implemented and evaluated:

Model	Accuracy	Precision	Recall	F1 Score	ROC-AUC
Logistic Regression	0.8579	0.1878	0.2288	0.2063	0.6549
Decision Tree	0.7819	0.1286	0.2948	0.1791	0.6399

K-Nearest Neighbors	0.7278	0.1122	0.3432	0.1691	0.5873
MLP Classifier	0.8827	0.1870	0.1351	0.1569	0.6619
Random Forest	0.8999	0.1816	0.0682	0.0992	0.6823
XGBoost	0.9090	0.2606	0.0690	0.1092	0.7051
LightGBM	**0.9103**	**0.2675**	**0.0636**	**0.1028**	**0.7071**

6.2 Observations

- Logistic Regression provided balanced recall and interpretability. It was scaled and tuned using grid search (C=0.1, solver='liblinear').
- XGBoost and LightGBM delivered the highest accuracy and ROC-AUC, but with low recall, suggesting they're good at identifying non-defaulters but not as effective at catching defaulters.
- KNN had poor performance across all metrics and is not recommended.
- MLP showed decent accuracy but lacked generalizability due to poor recall.
- Recommended Model for Production: **Logistic Regression** (if interpretability and regulatory compliance are required) OR
- **LightGBM** (if highest predictive power and AUC are prioritized, with caution for imbalanced recall).

6.3 Class Imbalance Handling

- SMOTE (Synthetic Minority Oversampling) was applied, but slightly reduced precision.
- Final model used **LightGBM with class weights** to balance loss function.

7. Challenges Faced

7.1 Data Quality Issues

- Missing values in critical columns like employment duration and income.
    - Addressed using statistical imputations and careful feature dropping.

7.2 Dataset Size and Memory Constraints

- Merging auxiliary datasets increased memory usage significantly.
    - Resolved by chunk processing and aggregations before merge.

7.3 Target Imbalance

- With only 8% positive class, models struggled to learn minority patterns.
    - Solved using class weighting and ROC-AUC for evaluation.

7.4 Feature Correlation and Redundancy

- High correlation among derived features led to overfitting.
    - Applied PCA and correlation-based feature selection to reduce redundancy.

8. Conclusion

This project demonstrates a complete end-to-end machine learning workflow:

- Comprehensive data cleaning and feature engineering
- Smart handling of large and messy auxiliary datasets
- Informed model selection based on metrics (not just accuracy)
- Use of both simple (Logistic) and advanced (LightGBM) models

Outcome: LightGBM provided the **best overall ROC-AUC**, while Logistic Regression gave **interpretable and balanced results**.

This solution can be deployed as a **credit scoring engine** for pre-screening applicants in real-time to reduce loan default risk.

9. Future Scope

- Deploy as API using Flask or FastAPI
- Build dashboard using Streamlit for business users
- Use SHAP values to explain LightGBM predictions
- Experiment with time-series modeling for payment behavior