Predictive Credit Risk Management

Project Report

Batch details	PGP-DSE Mar'24
Team members	Abhishek M, Anurag Singh, Saravanan G, Sriharan T, Syed Shayan Abid Hussain
Domain of Project	Big Data Analytics in Finance
Proposed project title	Predictive Credit Risk Management
Group Number	2
Team Leader	Saravanan G
Mentor Name	Ms, Vibha Santhanam

Date: 04-10-2024

Signature of the Mentor

Signature of the Team Leader

1. Summary of Problem Statement, Data, and Findings

The project aims to address the challenge of predicting loan defaults using a comprehensive dataset of customer applications. Loan default risk is a major concern for lenders, and misclassification could lead to significant financial losses. This project uses historical data on customer demographics, financials, and credit history to build a predictive model for default likelihood.

The dataset consists of 307,511 rows and 122 columns, capturing a variety of features such as the client's income, credit amount, days employed, and external credit scores. Key findings revealed a strong correlation between features like **EXT_SOURCE_2**, **EXT_SOURCE_3**, and **DAYS_EMPLOYED** with default status. After thorough data cleaning and feature engineering, we used machine learning models to predict loan defaults, achieving a model with robust performance.

2. Overview of the Final Process

- **Data Overview**: The original data consisted of 307,511 records and 122 features, with missing values in many columns. Some columns, such as **OCCUPATION_TYPE**, had over 30% missing data.
- **Datasets Used**:

  Shape of Application Data : Rows – 307511, Features - 122

  Shape of Previous Application: Rows – 1670214, Features – 37

  Shape of POS Cash Balance: Rows – 10001358, Features – 8

  Shape of Credit Card Balance: Rows – 3840312, Features – 23

- **Data Preprocessing**:
  - Missing data handling involved removing columns with more than 40% missing values, and imputing others with the median for numeric features like **AMT_ANNUITY** and mode for categorical features such as **NAME_TYPE_SUITE**.
  - Feature Engineering: New features like **CREDIT_INCOME_PERCENT** (credit amount to income ratio) and **CREDIT_TERM** (credit amount to annuity ratio) were created to enhance model performance.

    - Categorical features such as **CODE_GENDER** and **FLAG_OWN_CAR** were encoded using binary encoding techniques.

- **Modeling**:
    - We tested various models including **Logistic Regression**, **Random Forest**, and **XGBoost**, **LightGBM**
    - Each model was tuned using grid search for hyperparameter optimization, focusing on minimizing the AUC-ROC score.

3. Step-by-Step Walkthrough of the Solution

- **Exploratory Data Analysis (EDA)**:
    - **Univariate Analysis**: Initial analysis focused on understanding the distributions of key variables such as **AMT_CREDIT**, **AMT_INCOME_TOTAL**, and **DAYS_BIRTH**. Many financial variables exhibited skewness, suggesting the need for transformation.

- **Important Insights from EDA**
    - Young males with lower secondary education and of lower income group and staying with parents or in a rented house, applying for low-range cash contract, should be denied.
    - Females are likely to repay but not if they are on maternity leave. Hence, bank can reduce the loan amount for female applicants who are on maternity leave.
    - Since people taking cash loans for repairs and urgent needs are more likely to default, bank can refuse them.
    - Since the people who have unused offers are more likely to default even though they have comparatively high total income, they can be offered loan at a higher interest rate.
    - Banks can target businessmen, students and working class people with academic degree/ higher education as they have no difficulty in repayment.
    - Bank can also approve loans taken on purpose for buying home or garage as there less chances of defaulting.

- **Feature Engineering**:
    - New features were created to capture additional relationships between variables. For instance:
        - **CREDIT_INCOME_PERCENT** = AMT_CREDITAMT_INCOME_TOTAL\frac{AMT_CREDIT}{AMT_INCOME_TOTAL}AMT_INCOME_TOTALAMT_CREDIT
        - **ANNUITY_INCOME_PERCENT** = AMT_ANNUITYAMT_INCOME_TOTAL\frac{AMT_ANNUITY}{AMT_INCOME_TOTAL}AMT_INCOME_TOTALAMT_ANNUITY
        - **CREDIT_GOODS_DIFF** = AMT_CREDIT−AMT_GOODS_PRICEAMT_CREDIT - AMT_GOODS_PRICEAMT_CREDIT−AMT_GOODS_PRICE, capturing discrepancies between credit and the price of goods.

4.Model Selection and Training:

Logistic Regression - Base Model

We initially trained a Logistic Regression model, achieving an ROC-AUC score of 0.60 on our Kaggle dataset. This score indicated room for improvement in our model's performance.

Ensemble Techniques

To enhance our model, we decided to explore ensemble techniques, starting with Decision Trees as our base model.

Decision Tree Performance

The Decision Tree model yielded a disappointing ROC-AUC score of 0.54, which was lower than our Logistic Regression model. However, after hyperparameter tuning, we improved the ROC-AUC score to 0.71, with recall scores of 0.65 for class 1 and 0.66 for class 0.

Addressing Overfitting

Given that our Decision Tree model exhibited slight overfitting (with a training accuracy of 0.67 and a test accuracy of 0.66), we opted to implement a Bagging technique using Random Forest Classification to mitigate this issue.

Random Forest Performance

With default hyperparameters, the Random Forest model achieved a ROC-AUC score of 0.73.

Hyperparameter Tuning for Random Forest

After tuning the hyperparameters of the Random Forest model, we obtained a ROC-AUC score of 0.747, along with improved recall scores of 0.73 for class 0 and 0.64 for class 1. However, the recall for class 1 was still lower compared to the Decision Tree.

Boosting Techniques

To further improve performance, particularly given the imbalanced nature of our target variable, we explored boosting techniques. We implemented XGBoost, which resulted in a ROC-AUC score of 0.756, with recall scores of 0.69 for both classes.

LightGBM Implementation

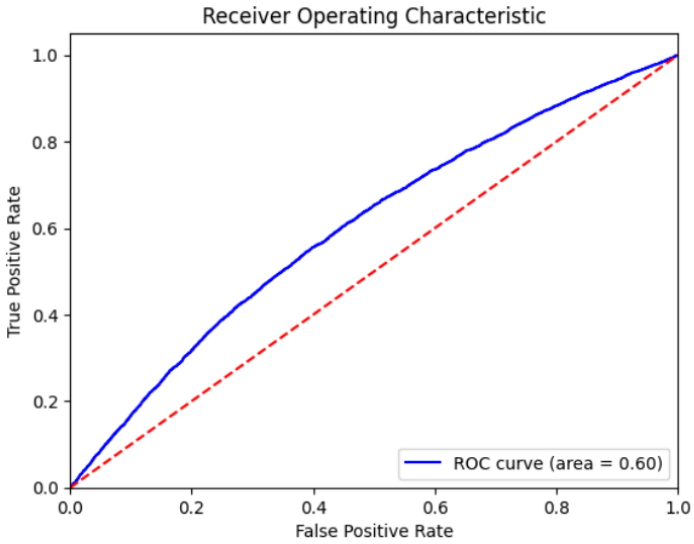
Finally, we employed LightGBM, which is well-suited for large datasets. This model achieved the highest ROC-AUC score of 0.76, with a recall of 0.69 for class 1, outperforming the other models.

Computational Limitations

Due to the large size of our dataset, we faced computational constraints that prevented us from performing GridSearchCV for hyperparameter optimization.

4. Model Evaluation

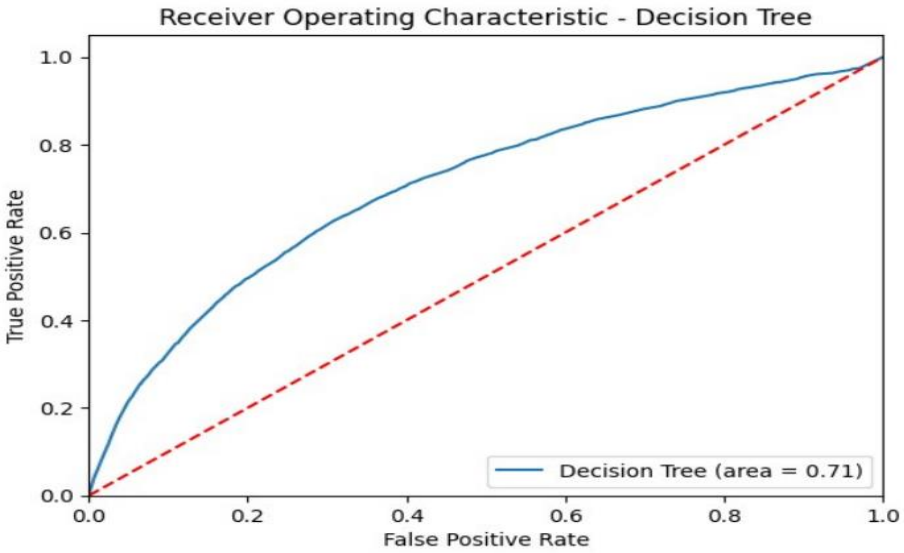
- Base Model - **Logistic Regression**
    - Logistic regression is a simple yet powerful algorithm for binary classification, making it ideal for predicting loan defaulters.
    - Logistic regression serves as a strong baseline model for classification tasks, and it's easy to implement.
    - It's faster than many complex models, allowing quick iterations and evaluations
    - The coefficients can help understand the importance of different features, useful for making data-driven decisions.



Results

- After fitting the logistic regression model it gave ROC AUC score as 0.60
- It gave a recall for defaulters as 0.55
- This score indicates that the model has limited ability to distinguish between defaulters and non-defaulters.
- The recall of 0.55 means that the model correctly identifies 55% of the defaulters, which implies it misses 45% of the actual defaulters.

- **Decision Tree**
    - Decision trees can model complex, non-linear relationships between features and the target variable
    - No Assumption of Linearity and it Handles Missing Data
    - Decision trees can easily overfit the training data, especially with large datasets, unless pruning or limiting tree depth is applied.
    - For very large datasets, decision trees can become slower to train and more memory-intensive compared to logistic regression

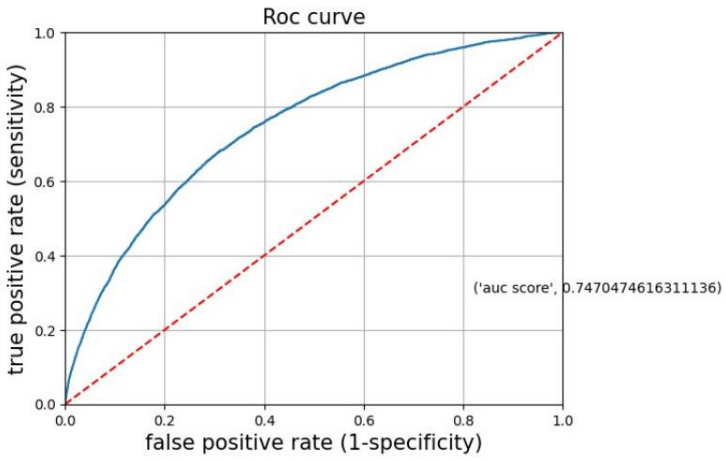


Results

- After fitting the Decision tree model it gave ROC AUC score as 0.71 with
- hyper parameteers criterion=gini',max_depth=9,min_samples_split=3,min_samples_leaf=3,class_weight='balanced'
- It gave a recall for defaulters as 0.65

- This is an improvement over the logistic regression model (which had a ROC AUC of 0.60), suggesting that the Decision Tree captures more complex patterns in the data.
- The recall of 0.65 means that the model correctly identifies 65% of the defaulters, which implies it misses 35% of the actual defaulters.

- **Random Forest**
  - Random Forest combines the predictions of multiple decision trees, reducing overfitting and capturing more complex patterns in the data.
  - By averaging multiple trees, Random Forest reduces the risk of overfitting, making it more robust for complex datasets.
  - Random Forest handles missing data better and is more effective with imbalanced datasets, especially with class weighting or sampling techniques.
  - Random Forests require more time and computational resources to train, especially with large datasets like Home Credit.

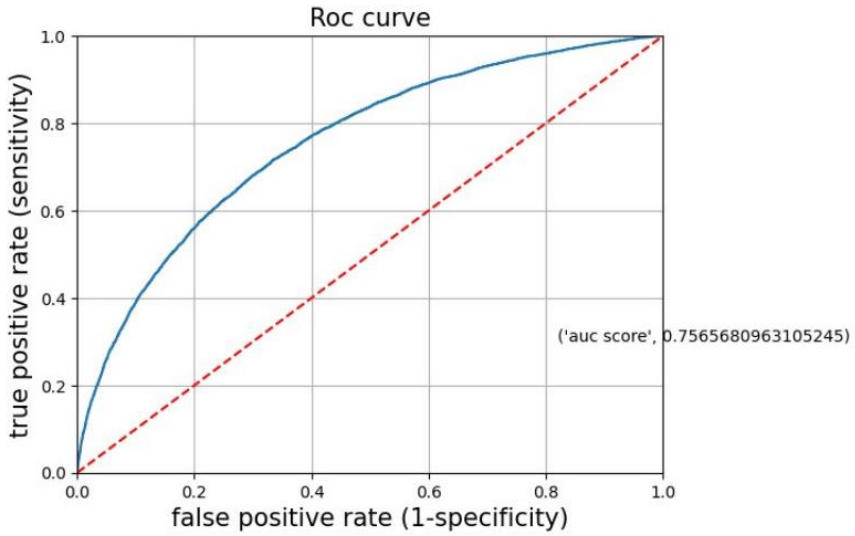


Results

- After fitting the Random forest model it gave ROC AUC score as 0.747 with
- hyper parameteers n_estimators=500, criterion=entropy,max_depth=9,min_samples_split=5,min_samples_leaf=2,class_weight='balanced'
- It gave a recall for defaulters as 0.64

- This is an improvement over the Decision tree model (which had a ROC AUC of 0.71), suggesting that Random forest captures more complex patterns in the data.
- The recall of 0.64 means that the model correctly identifies 65% of the defaulters, which implies it misses 36% of the actual defaulters.

- **XG Boost Classifier**
    - XGBoost is a powerful boosting algorithm that often outperforms simpler models like logistic regression on structured/tabular data.
    - XGBoost uses boosting to improve prediction performance by sequentially building trees that focus on correcting errors from previous iterations, leading to better accuracy.
    - XGBoost can handle class imbalance better by using techniques like custom loss functions or adjusting the scale_pos_weight parameter.
    - XGBoost has a large number of hyperparameters (e.g., learning rate, max_depth, subsample, etc.), and optimizing them can be time-consuming and complex.

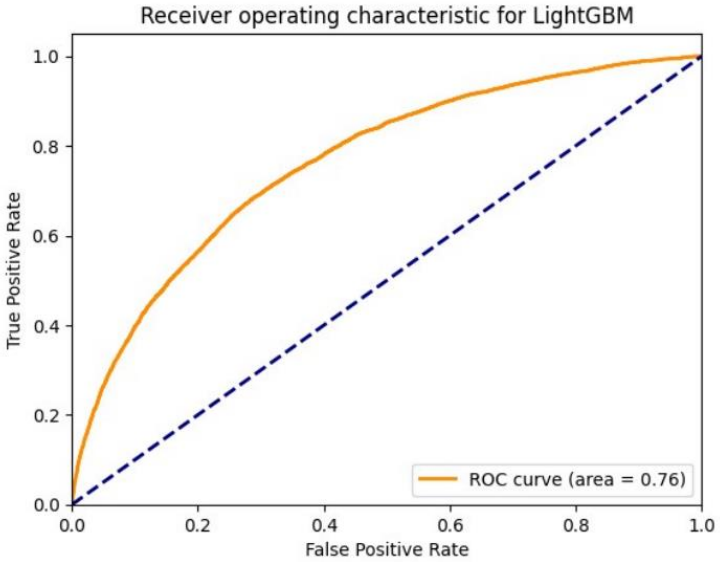


Results

- After fitting the XG Boost model it gave ROC AUC score as 0.7565
- It gave a recall for defaulters as 0.69

- This is an improvement over the Random forest model (which had a ROC AUC of 0.747), suggesting that XG Boost captures more complex patterns in the data.
- The recall of 0.69 means that the model correctly identifies 69% of the defaulters, which implies it misses 31% of the actual defaulters.
- Best_params = { 'objective': 'binary:logistic', 'scale_pos_weight': sum(ytrain == 0) / sum(ytrain == 1), # Handle imbalance 'eval_metric': 'aucpr', # Precision-Recall AUC 'max_depth': 3, # Control overfitting 'min_child_weight': 6, # Regularization 'gamma' # Prevents overfitting (pruning) : 4, # Regularization 'lambda': 1.0, # L2 regularization 'learning_rate': 0.1, # Lower learning rate for better generalization 'n_estimators': 100 }

- **Final Model - LightGBM Classifier**
  - LightGBM is specifically optimized for efficiency and speed on large datasets.
  - LightGBM is highly optimized for speed and can handle large datasets with many features and samples much faster than logistic regression and even XGBoost.
  - LightGBM has built-in handling for imbalanced datasets, allowing you to adjust for class imbalance without extensive preprocessing
  - If not carefully tuned or regularized, LightGBM can overfit, particularly when the dataset is small or noisy. Regularization techniques and early stopping are often necessary.

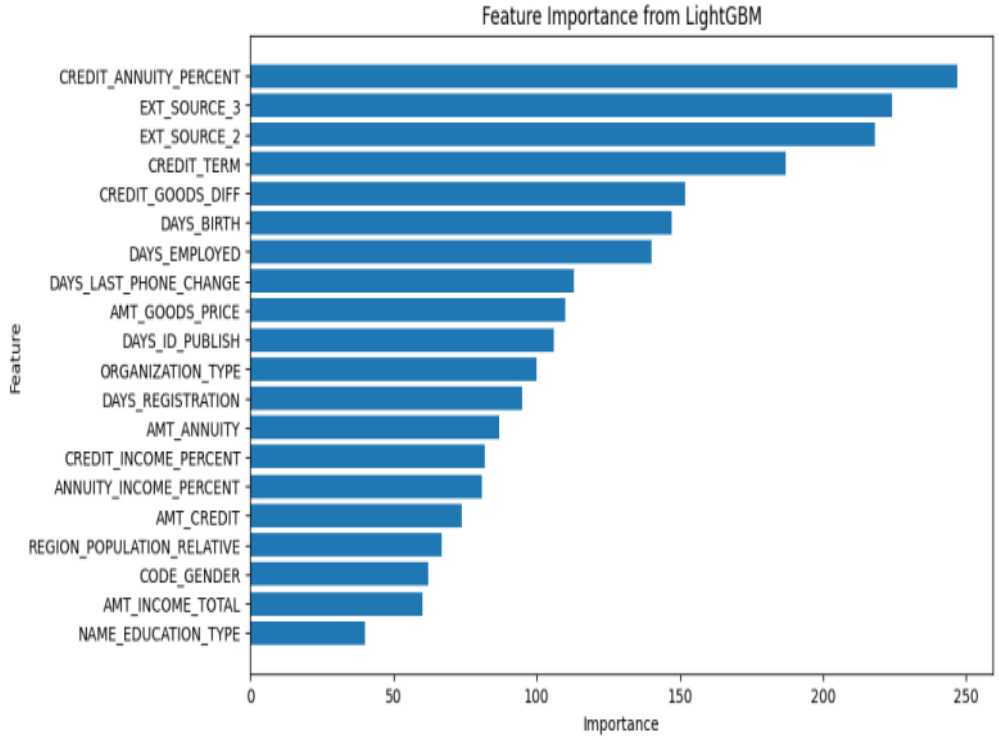


Results

- After fitting the XG Boost model it gave ROC AUC score as 0.76
- It gave a recall for defaulters as 0.69
- This is an improvement over the XG Boost model (which had a ROC AUC of 0.75), suggesting that Lightgbm captures more complex patterns in the data.
- The recall of 0.69 means that the model correctly identifies 69% of the defaulters, which implies it misses 31% of the actual defaulters.
- Best Parameters: {'learning_rate': 0.05, 'max_depth': -1, 'min_child_samples': 40, 'n_estimators': 200(no of boosting iterations), 'num_leaves': 31(no.of leaves each tree should have)}
- Best AUC Score: 0.758 (~0.76)

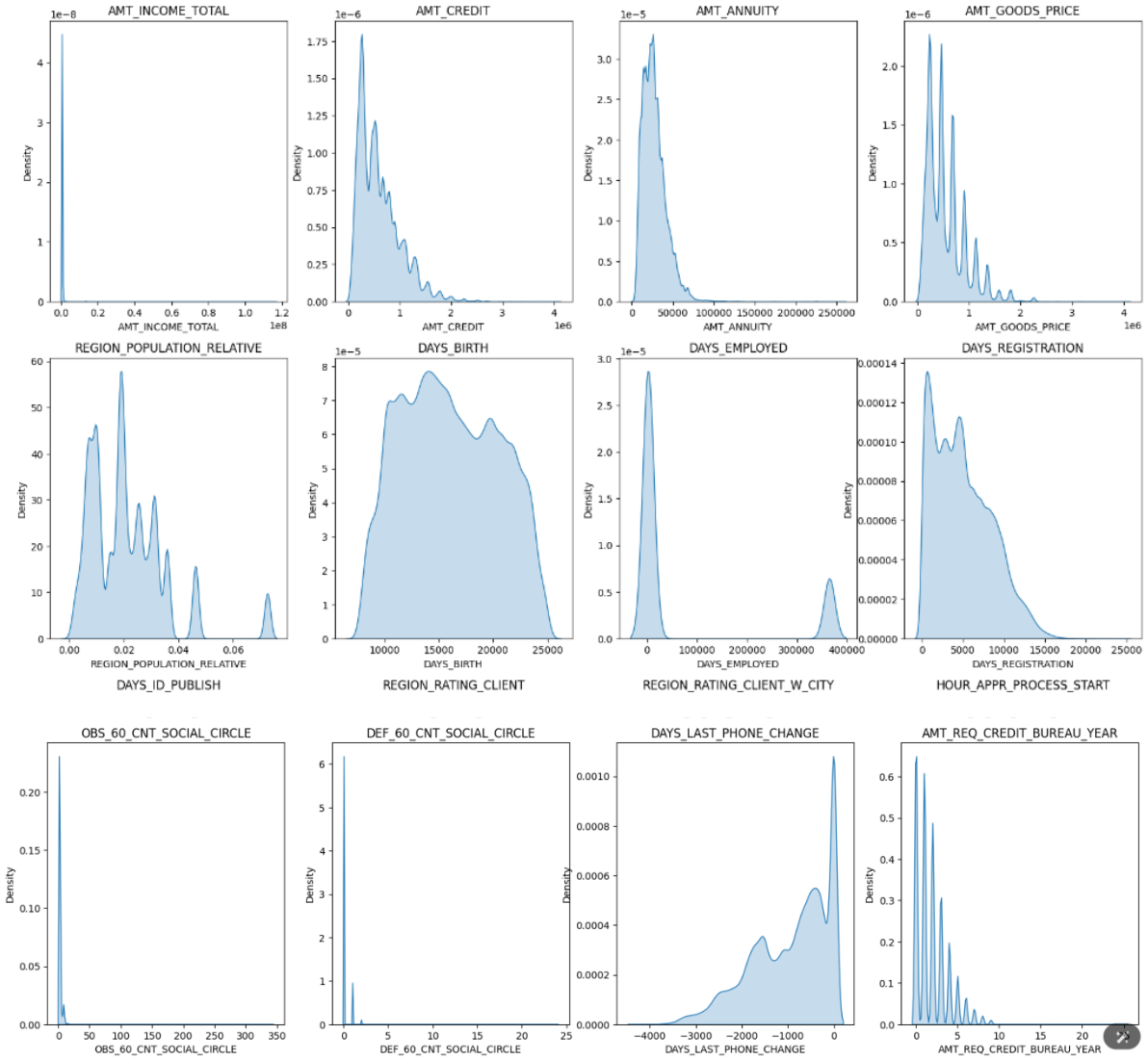
- **Evaluation Metrics**:
  - **AUC-ROC** was the primary evaluation metric due to its robustness in imbalanced datasets. The final model achieved an AUC score of **0.76**, representing a significant improvement over the baseline logistic regression model.
  - Precision, Recall, and F1-Score were also computed to measure the trade-offs between precision and recall.

Top 20 Features from Lightgbm



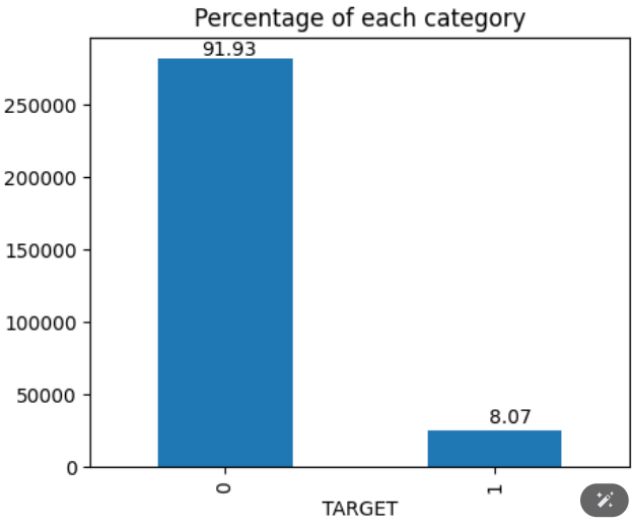
6. Visualizations

- **Univariate Visualizations**:
    - The distribution of **AMT_CREDIT** and **AMT_INCOME_TOTAL** revealed significant skewness, which required log transformation.



- Bar plots for categorical variables such as **CODE_GENDER** and **NAME_INCOME_TYPE** helped visualize the distribution of key demographic features

TARGET :

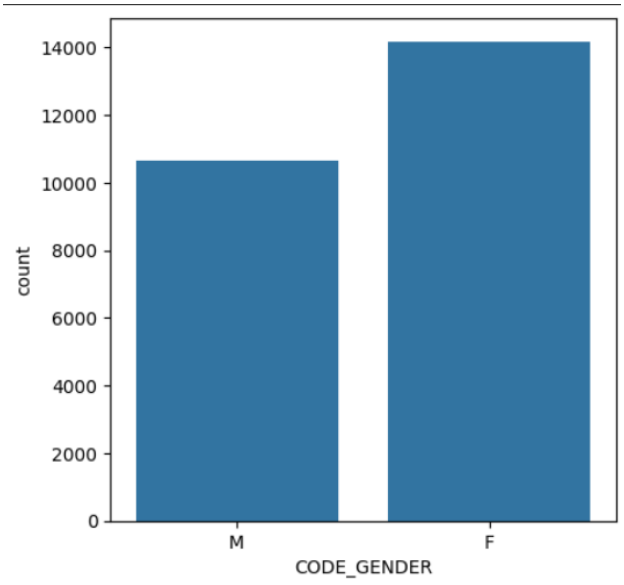


1– Defaulters (~8%)

0- Non Defaulters (~92%)

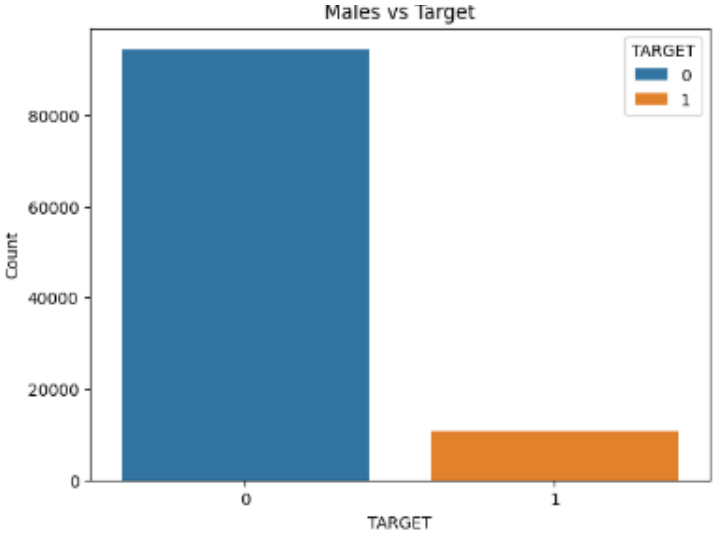
Target imbalance was one of the key challenges we faced during the modelling process. We used class weights method to handle the imbalance.

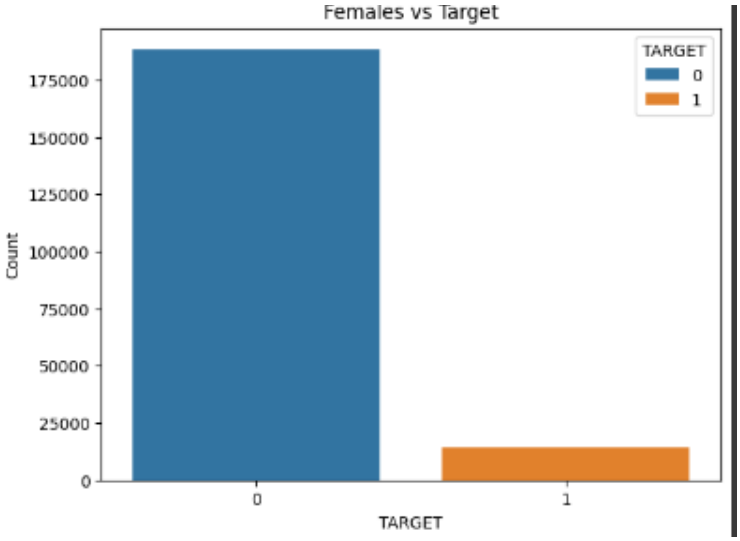
CODE GENDER-

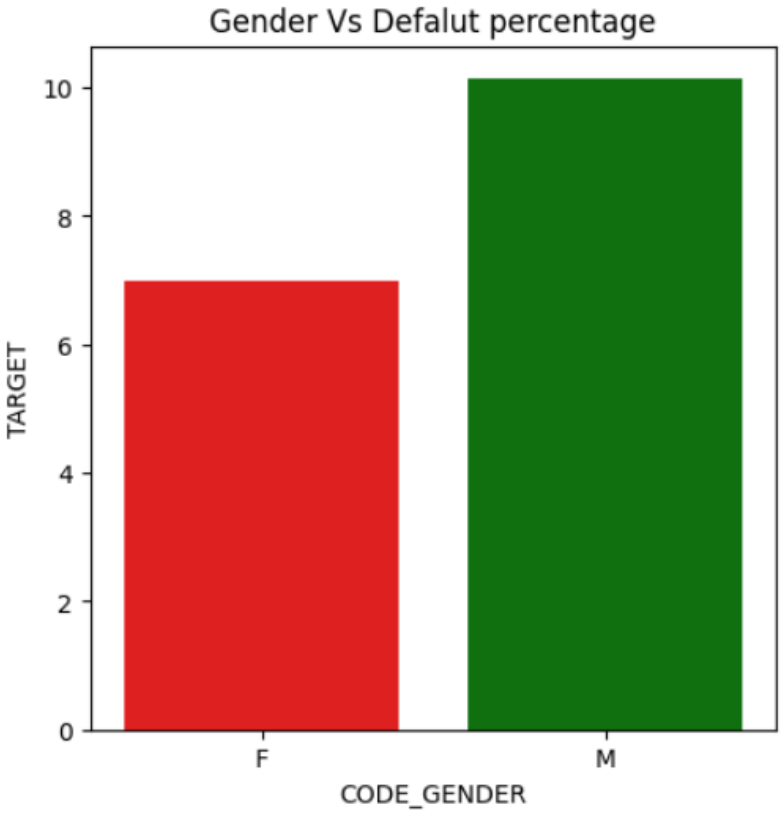


Most of the clients were Females

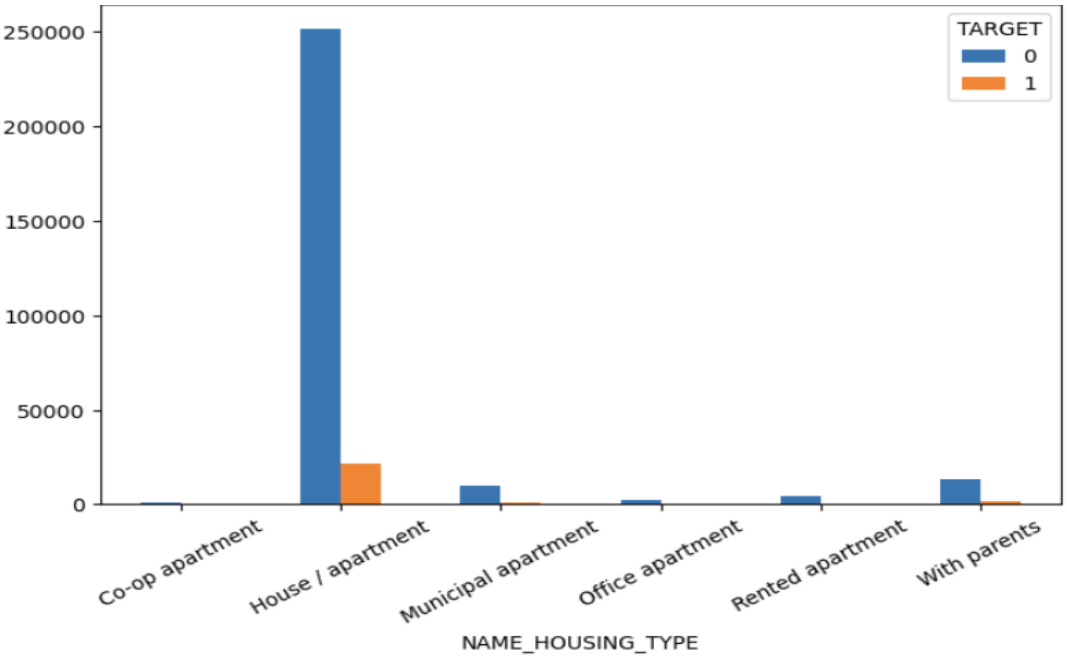
- **Bivariate Visualizations**:

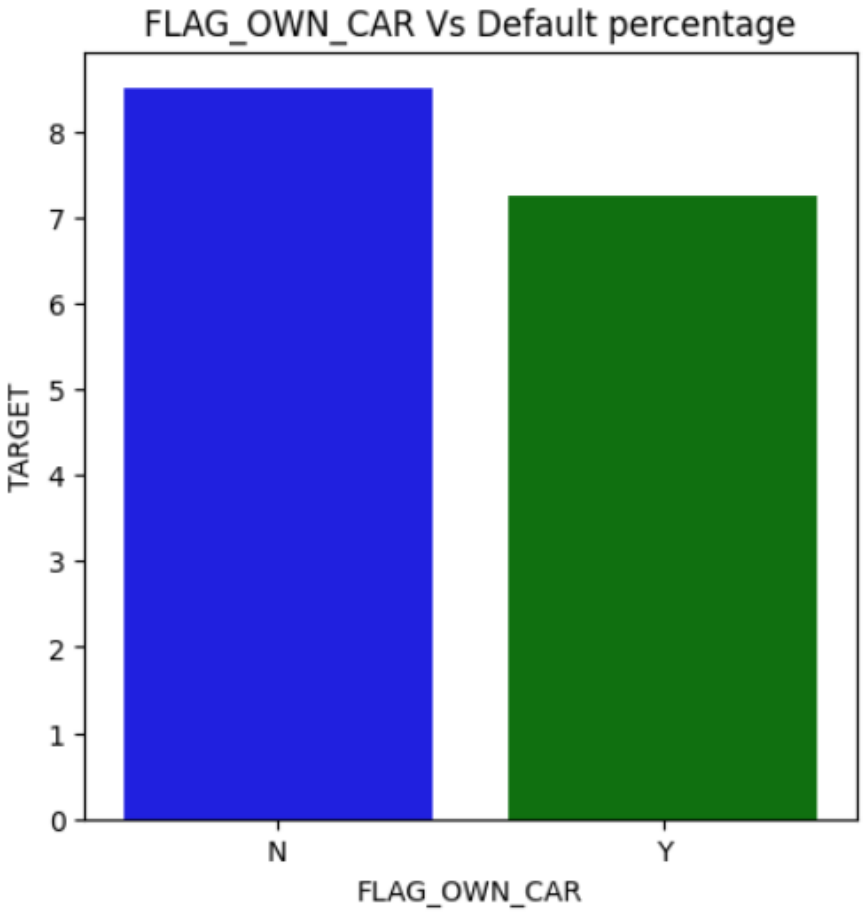


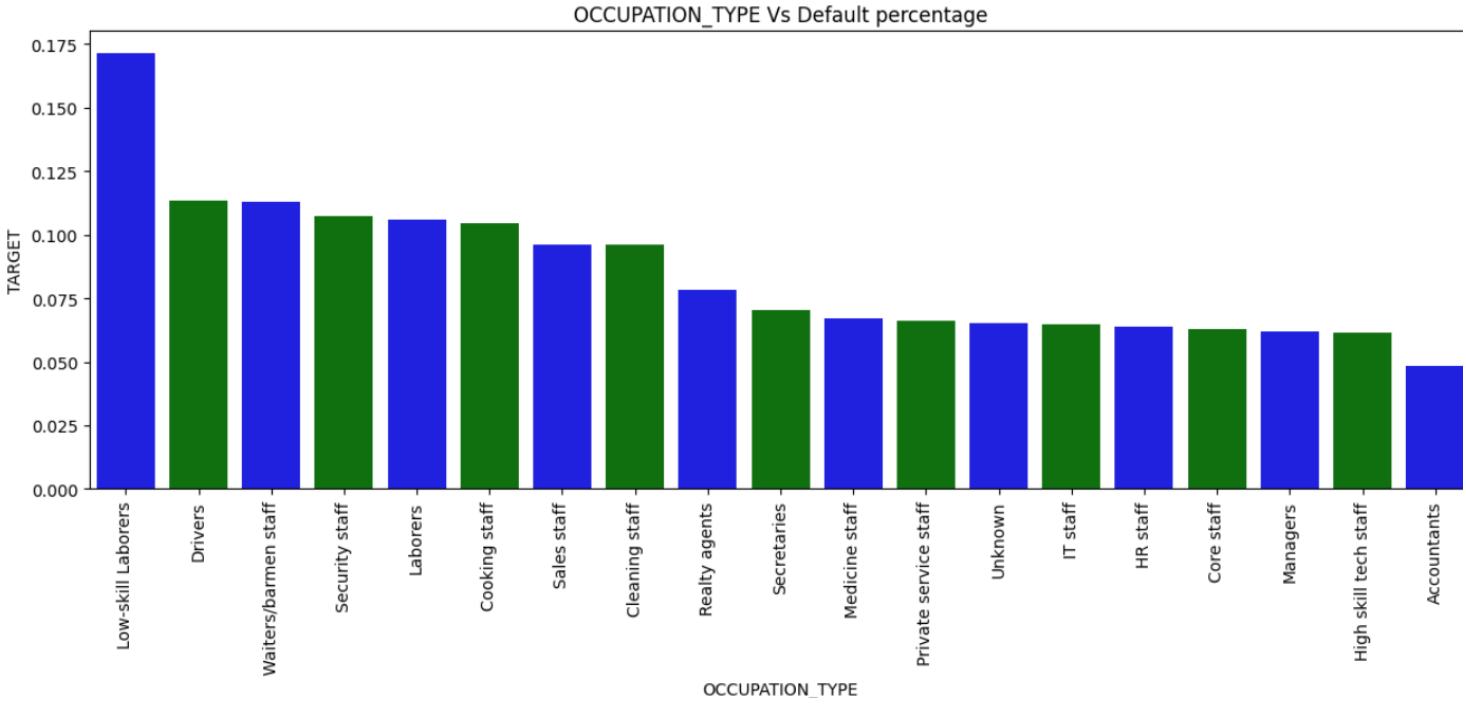


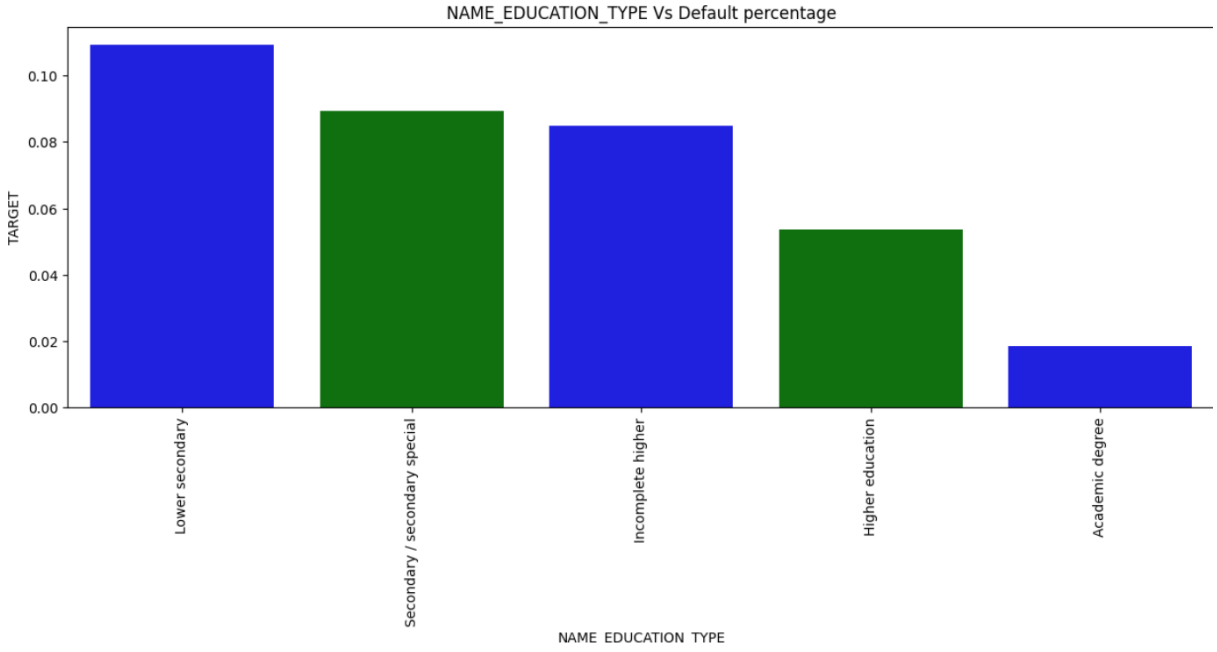


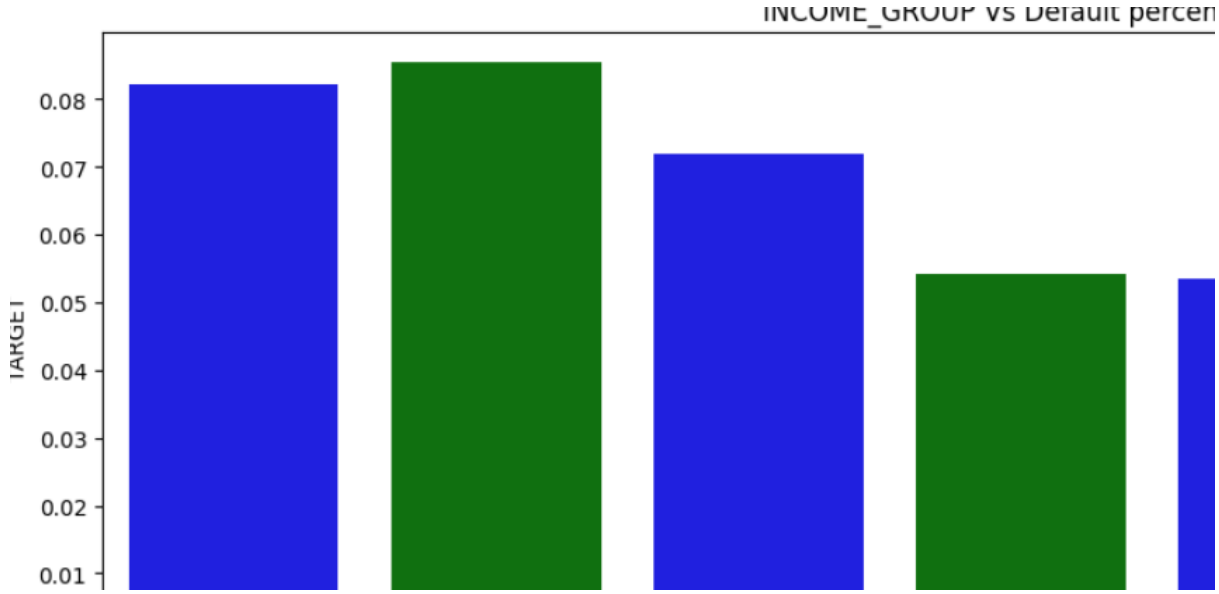
Comparison of **TARGET** with **CODE_GENDER** revealed that males had a higher propensity for defaulting compared to female

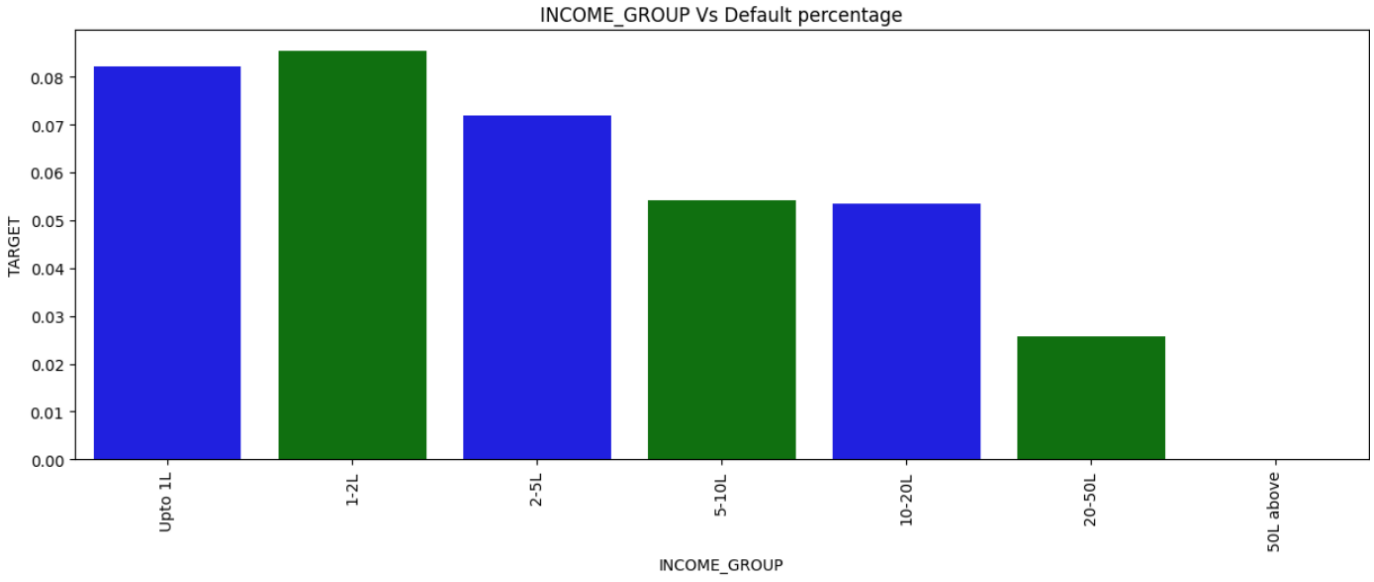

TARGET
0
1
250000
200000
150000
100000
50000
0
Co-op apartment
House / apartment
Municipal apartment
Office apartment
Rented apartment
With parents
NAME_HOUSING_TYPE

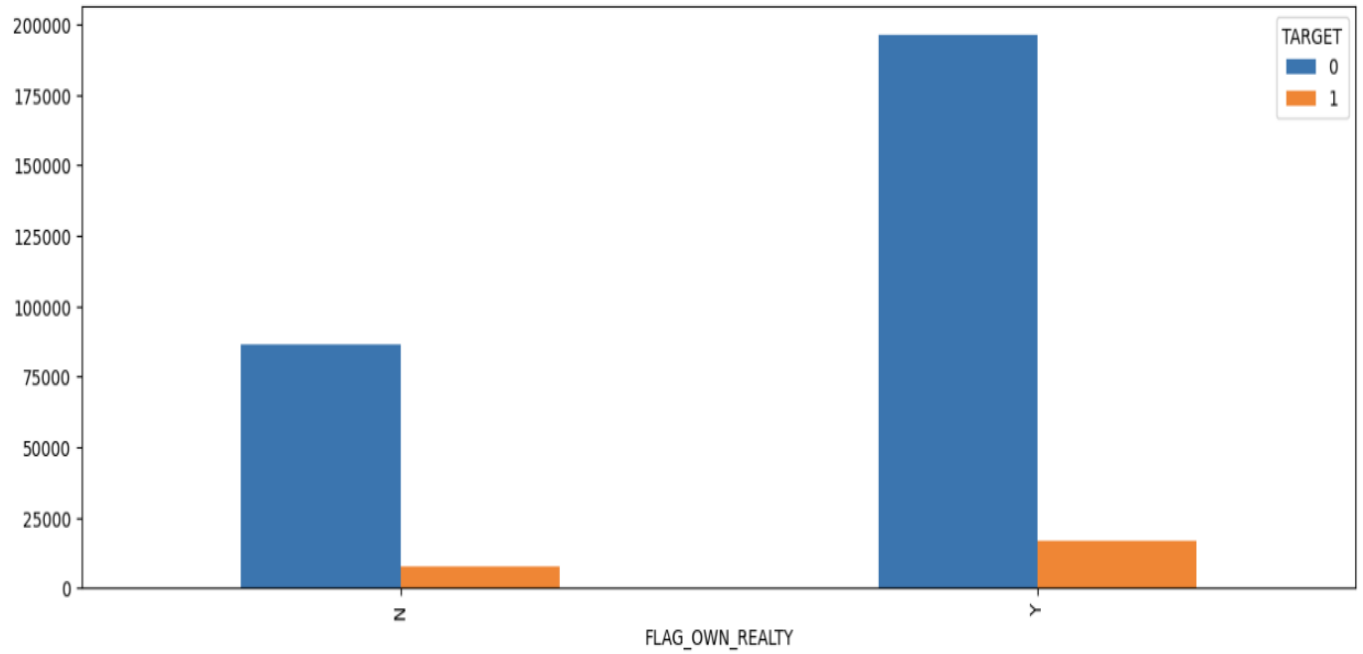

FLAG_OWN_CAR Vs Default percentage
TARGET
8
7
6
5
4
3
2
1
0
N
Y
FLAG_OWN_CAR


OCCUPATION_TYPE Vs Default percentage
TARGET
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
Low-skill Laborers
Drivers
Waiters/barmen staff
Security staff
Laborers
Cooking staff
Sales staff
Cleaning staff
Realty agents
Secretaries
Medicine staff
Private service staff
Unknown
IT staff
HR staff
Core staff
Managers
High skill tech staff
Accountants
OCCUPATION_TYPE

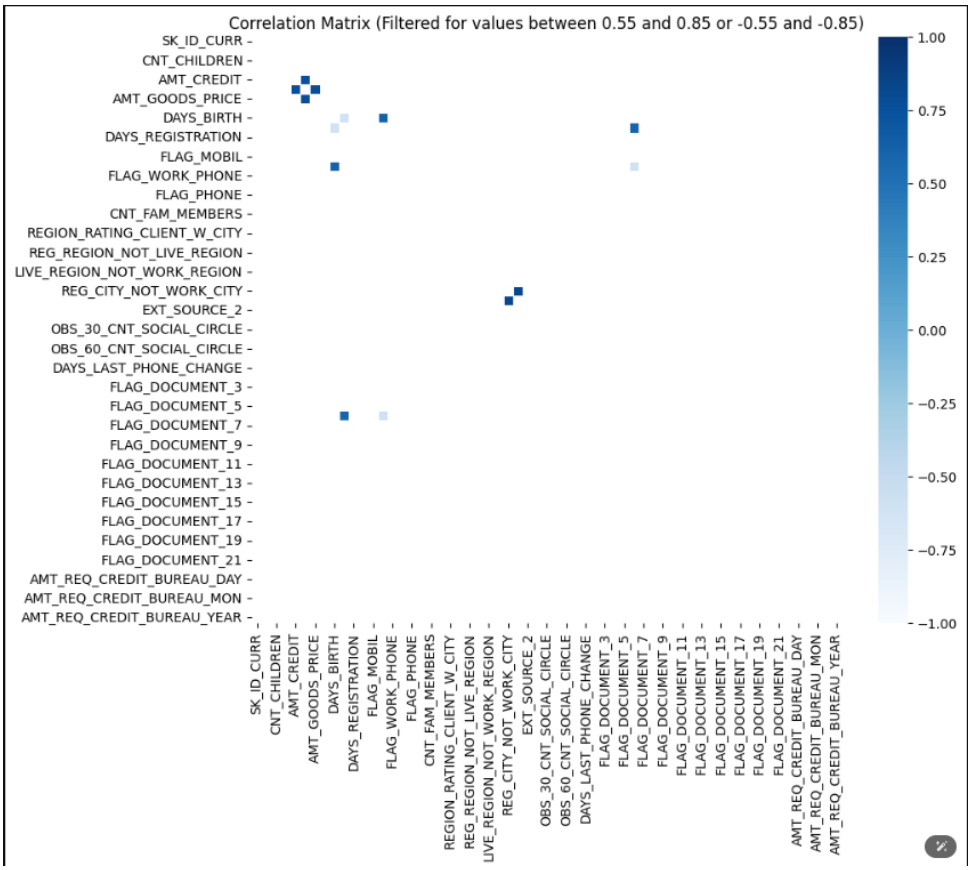

NAME_EDUCATION_TYPE Vs Default percentage
TARGET
0.10
0.08
0.06
0.04
0.02
0.00
Lower secondary
Secondary / secondary special
Incomplete higher
Higher education
Academic degree
NAME_EDUCATION_TYPE


0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01


INCOME_GROUP Vs Default percentage
TARGET
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Upto 1L
1-2L
2-5L
5-10L
10-20L
20-50L
50L above
INCOME_GROUP



- **Multivariate Analysis**:



Features with strong correlations were treated by Feature engineering by combining those features.

7. Implications

The model developed in this project has direct business implications for credit risk management in the lending industry. By accurately predicting which customers are likely to default, lenders can:

- **Reduce financial risk** by targeting high-risk customers with stricter loan terms or denying high-risk loans.
- **Tailor products** to low-risk customers by offering them better loan terms, thereby increasing customer satisfaction and retention.
- **Automation of loan decisions**: The model can be integrated into existing systems to automatically assess risk, speeding up the loan approval process while maintaining a high level of accuracy.

8. Limitations

- Missing Data: Significant missing values, particularly in OCCUPATION_TYPE (31%) . Columns with over 40% missing data were dropped, but this led to information loss.
- Imbalanced Target: The dataset was heavily imbalanced, with ~92% non-defaultersThis made it challenging to train models without overfitting.
- Skewed Data: Features like AMT_INCOME_TOTAL and AMT_CREDIT were highly skewed due to outliers, complicating model performance.
- Negative Values: Time-related columns (e.g., DAYS_EMPLOYED) had negative values, which had to be converted to absolute values for clarity
- Categorical Variables: Variables like ORGANIZATION_TYPE had many levels, increasing complexity in encoding.
- Multicollinearity: High correlation between variables (e.g., AMT_CREDIT and AMT_GOODS_PRICE) needed careful handling.
- Outliers: Extreme values in columns like DAYS_EMPLOYED (e.g., 365,243 days) distorted model predictions.
- Feature Engineering: Creating new features like CREDIT_INCOME_PERCENT was complex but necessary to improve model performance.

9. Closing Reflections

This project provided valuable insights into how machine learning can be leveraged to improve credit risk assessment. Key learnings include the importance of feature engineering and the handling of imbalanced datasets. In future iterations of this project, we could explore deep learning models, which might capture more complex relationships between variables, or use external economic indicators to further improve the model's predictive power.