UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Home Credit Default Risk

Gwendoline Hays-Valentin, Hugo Michel, Thomas Rigou, Charaf Zguiouar

Explainable AI for credit risk management

Professor: Etienne Gay

Final Report

-

M2 Finance Technology Data - Sorbonne School Of Economics

Université Paris 1 Panthéon-Sorbonne

Paris, France, Fall 2023

Source code.

Abstract

The objective of this project is to evaluate the feasibility and effectiveness of using Explainable Artificial Intelligence (XAI) in the context of credit risk management. The project involves a comprehensive analysis using the Home Credit Default Risk dataset to develop and validate credit scoring models. Initially, various machine learning algorithms, including Random Forest, LGBM, Support Vector Machine (SVM), XGBoost, and Neural Networks, are employed to create robust scoring models. Following the model development, XAI frameworks such as SHAP (SHapley Additive exPlanations) and Shapash , LIME, DICE are implemented to elucidate the models' predictions and enhance their interpretability.

The primary aim is to assess whether XAI can provide meaningful insights into the factors influencing credit risk predictions, thereby improving the transparency and reliability of credit scoring systems. The findings from the XAI analysis are expected to identify critical features contributing to credit decisions, offer explanations for the models' behavior, and highlight any inconsistencies or biases within the predictions. This project not only aims to deliver a high-performing credit scoring model but also emphasizes the importance of interpretability in AI-driven decision-making processes within the financial sector.

Through this proof of concept, the project provides scientific, methodological, and practical insights into the potential benefits and challenges of integrating XAI into credit risk management. The results are intended to guide future investments in XAI technologies and support the development of more transparent and trustworthy credit scoring systems.

The source code is available here.

Contents

List of Tables

1 Introduction

1.1 Project Context

The rapid advancement of artificial intelligence (AI) has significantly transformed various industries, with the financial sector being one of the most impacted. Among the many applications of AI in finance, credit risk management stands out as a critical area where AI can provide substantial benefits. Accurate credit scoring models are essential for financial institutions to assess the reliability of borrowers and mitigate potential risks. However, the complexity and opacity of many AI models pose challenges in understanding and trusting their predictions, particularly in a domain where transparency is crucial.

Explainable Artificial Intelligence (XAI) has emerged as a promising solution to address these challenges. By making AI models more interpretable, XAI enables stakeholders to understand the underlying factors influencing predictions, thereby enhancing trust and facilitating better decision-making. This project aims to explore the viability of integrating XAI into credit risk management, providing insights into its practical applications and potential benefits.

1.2 Project Objective

The primary objective of this project is to evaluate the feasibility and effectiveness of employing Explainable AI (XAI) in the context of credit risk management. Specifically, we want to develop a robust credit scoring models using various machine learning algorithms, including Random Forest, Support Vector Machine (SVM), XGBoost, and Neural Networks. We want to implement and compare different XAI frameworks, such as SHAP (SHapley Additive exPlanations), Quantus, and Captum, to enhance the interpretability of the credit scoring models. We will also analyzed the insights provided by XAI to identify critical features influencing credit risk predictions and assess the consistency and logic of the models' behavior. Finally, the idea is to evaluate the practical implications of using XAI in credit risk management, including its potential to improve transparency, trust, and decision-making processes within financial institutions. By achieving these objectives, the project seeks to demonstrate the value of XAI in enhancing the reliability and transparency of AI-driven credit scoring systems.

2 Data Exploration

2.1 Dataset selection

Feature	German Credit Data	Credit Card Dataset	Credit Risk Dataset	Selected Dataset
Age	Yes	Yes	Yes	Yes
Income	Yes	Yes	Yes	Yes
Credit Amount	Yes	No	Yes	Yes
Employment Length	No	Yes	Yes	Yes
Home Ownership	Yes	Yes	Yes	Yes
Loan Intent	No	No	Yes	Yes
Loan Status	No	No	Yes	Yes
Target	No	Yes	Yes	Yes
Time Index	No	No	No	Yes
Number of Rows	1000	438,510	32,581	300,000
Number of Columns	10	20	12	122

Table 2.1: Comparison of Different Datasets

We selected the fourth dataset due to its superior features and data richness. Compared to the first three datasets, the chosen dataset offers several distinct advantages:

Completeness of Variables: This dataset includes all key variables such as age, income, credit amount, employment length, home ownership, loan intent, and loan status. The presence of these variables is crucial for a detailed and accurate credit scoring analysis.

Target Variable and Time Index: Unlike the other datasets, this one includes both a target variable and a time index, allowing for more dynamic and longitudinal analyses of borrower behavior over time.

Data Volume: With 300,000 rows, this dataset offers a substantial volume of data, which is essential for training robust and reliable machine learning models. A larger dataset captures greater variability and improves model generalization.

Richness of Variables: This dataset has 122 columns, providing a wealth of explanatory variables. This diversity of variables allows for the creation of more detailed models and a better understanding of the various factors influencing credit scoring.

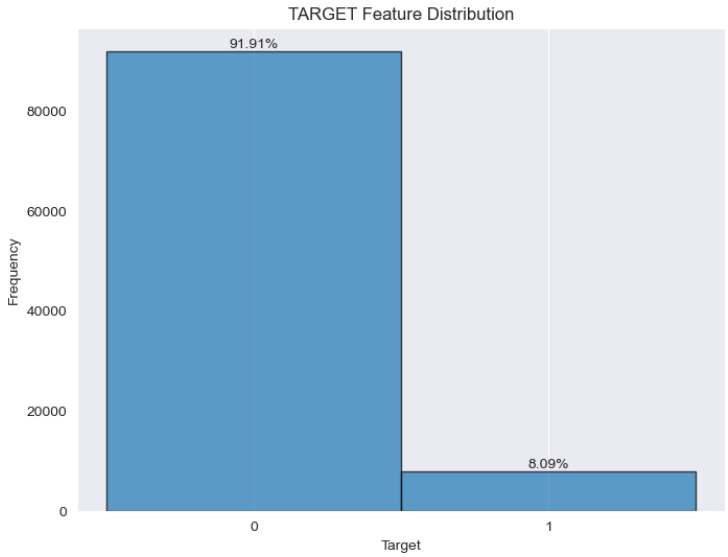
In summary, the fourth dataset was chosen for its completeness, richness, and data volume, enabling the development of more explainable and accurate credit scoring models while fully leveraging XAI capabilities for better interpretability of the results.

2.2 Exploratory Data Analysis (EDA)

2.2.1 Target distribution

It is evident that the distribution of the target variable within our dataset exhibits significant class imbalance. Specifically, instances belonging to class 0, indicating non-default, are notably over-represented compared to instances in class 1, denoting default. This observed imbalance is a common occurrence in credit risk datasets, mirroring the real-world scenario where the prevalence of non-defaulters typically outweighs that of defaulters. As such, dealing with imbalance in credit risk datasets is a standard practice, necessitating the implementation of specialized techniques to address this inherent class imbalance and ensure robust model performance.

Figure 2.1: Target distribution (0: non-default, 1: default)



2.2.2 Categorical Variables

To scrutinize the categorical variables within our dataset, we employ several analytical techniques. Initially, we examine the distribution of each variable through histogram plots to gain a comprehensive understanding of their overall distribution patterns. Subsequently,

we narrow our focus to the distribution of each variable concerning the default class, as it is of particular interest in our analysis. To further elucidate the proportional representation of variable distributions within the default class, we utilize tree maps for visualization purposes.

Figure 2.2: Distribution of the categorical variable *Education Type*

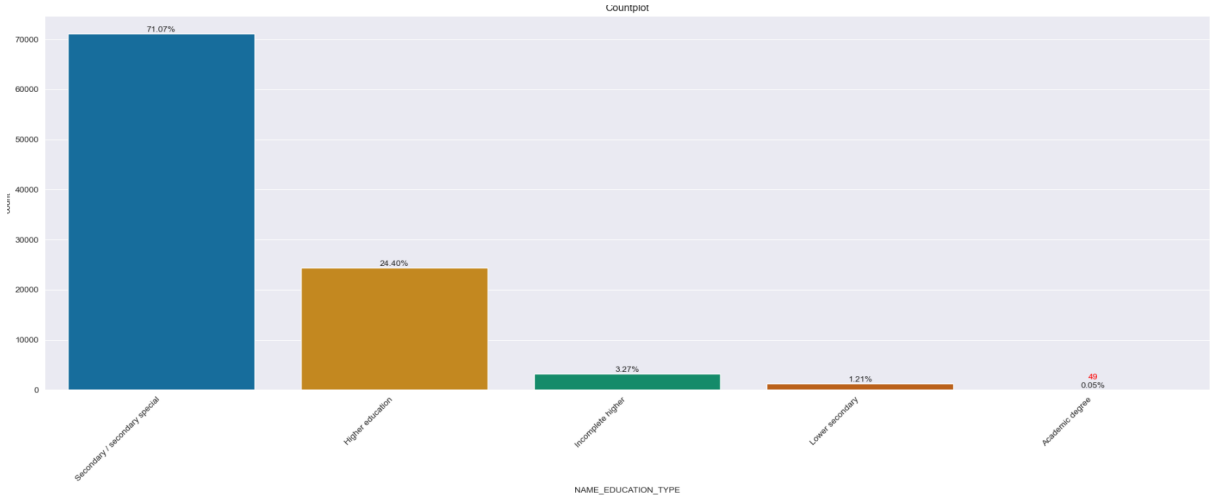


Figure 2.3: Distribution of the categorical variable *Education Type* among the class 1: Default

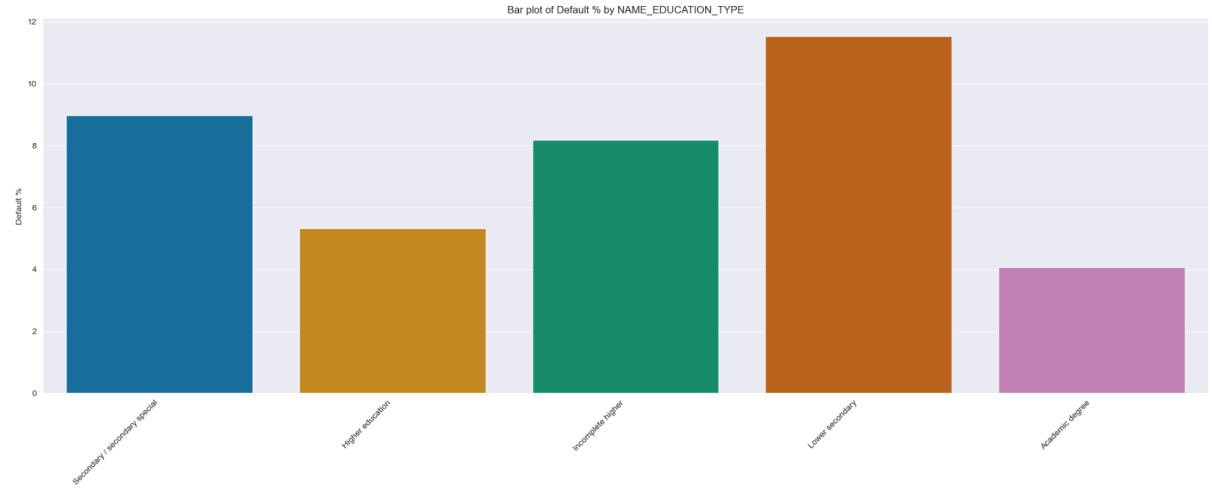


Figure 2.4: Distribution of the categorical variable *Education Type* among the class Non-Default: 0 and Default: 1

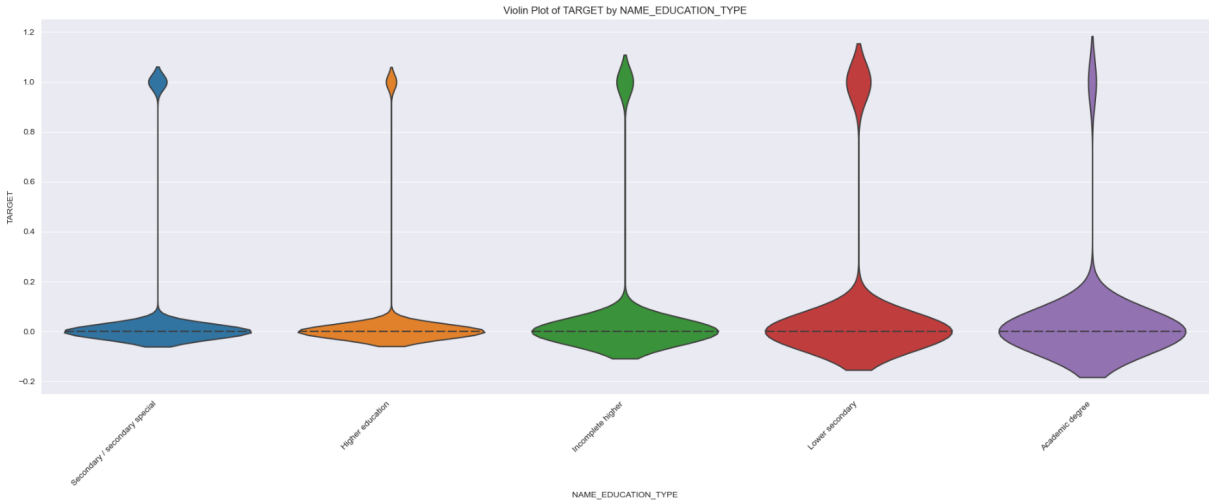
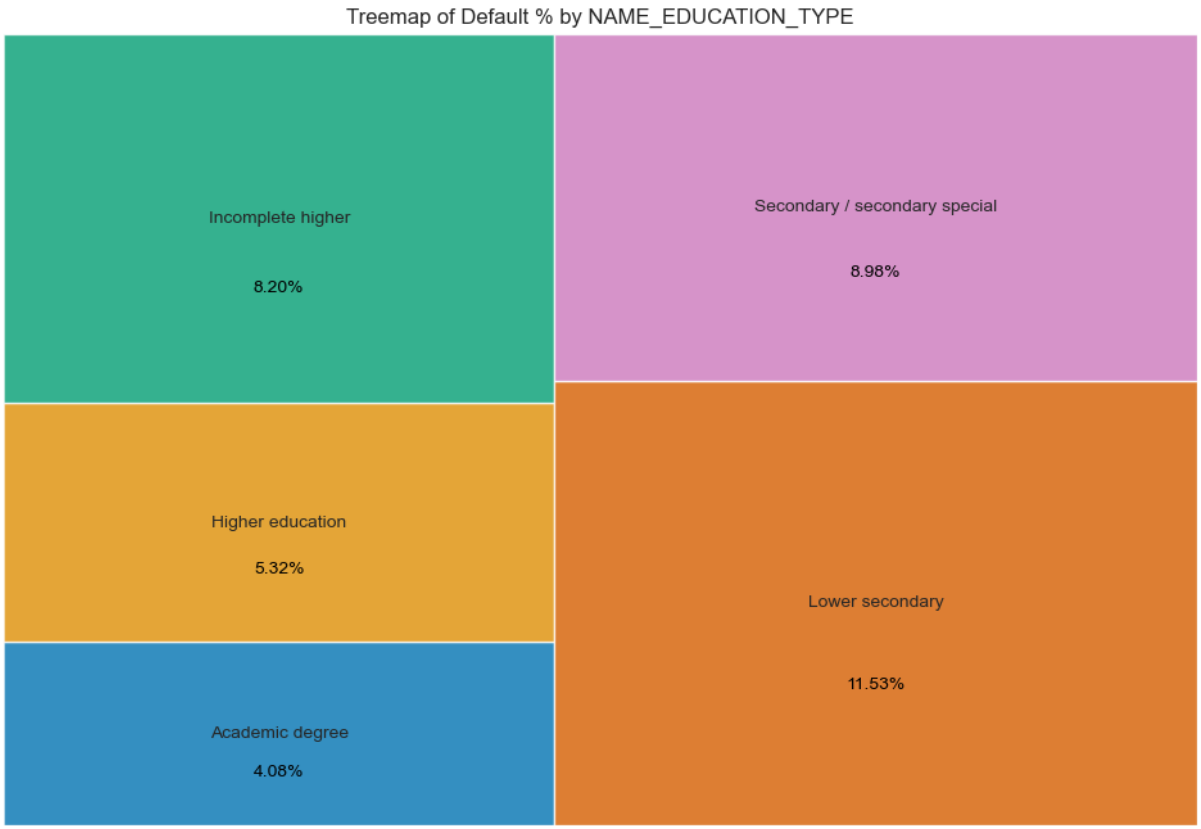


Figure 2.5: Treemap of the categorical variable *Education Type* among the class Default: 1

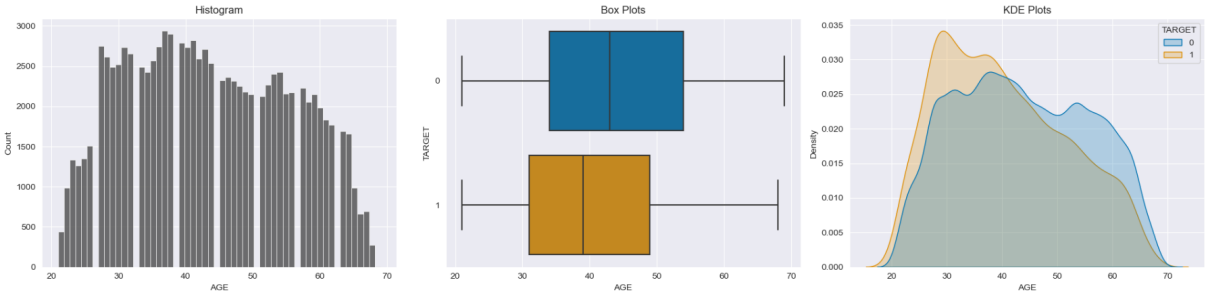


2.2.3 Quantitative Variables

To analyze the quantitative variables in our dataset, we adopt a multi-step approach. Initially, we examine the distribution of each variable individually using histogram plots,

facilitating an understanding of their overall distributional characteristics. Subsequently, we delve deeper into the distribution of each variable, stratifying the analysis according to the two classes: non-default (0) and default (1). This allows us to discern any differential patterns in variable distributions between the two classes, providing valuable insights into potential predictors of default behavior. Finally, we employ box plots to gain an overview of the main statistical descriptors of the quantitative variables. This visualization tool enables us to identify any outliers present in the data and assess their potential impact on our analysis.

Figure 2.6: Descriptive statistic of the variable quantitative *Age*



2.2.4 Corralation for categorical variables with Cramer's Matrix

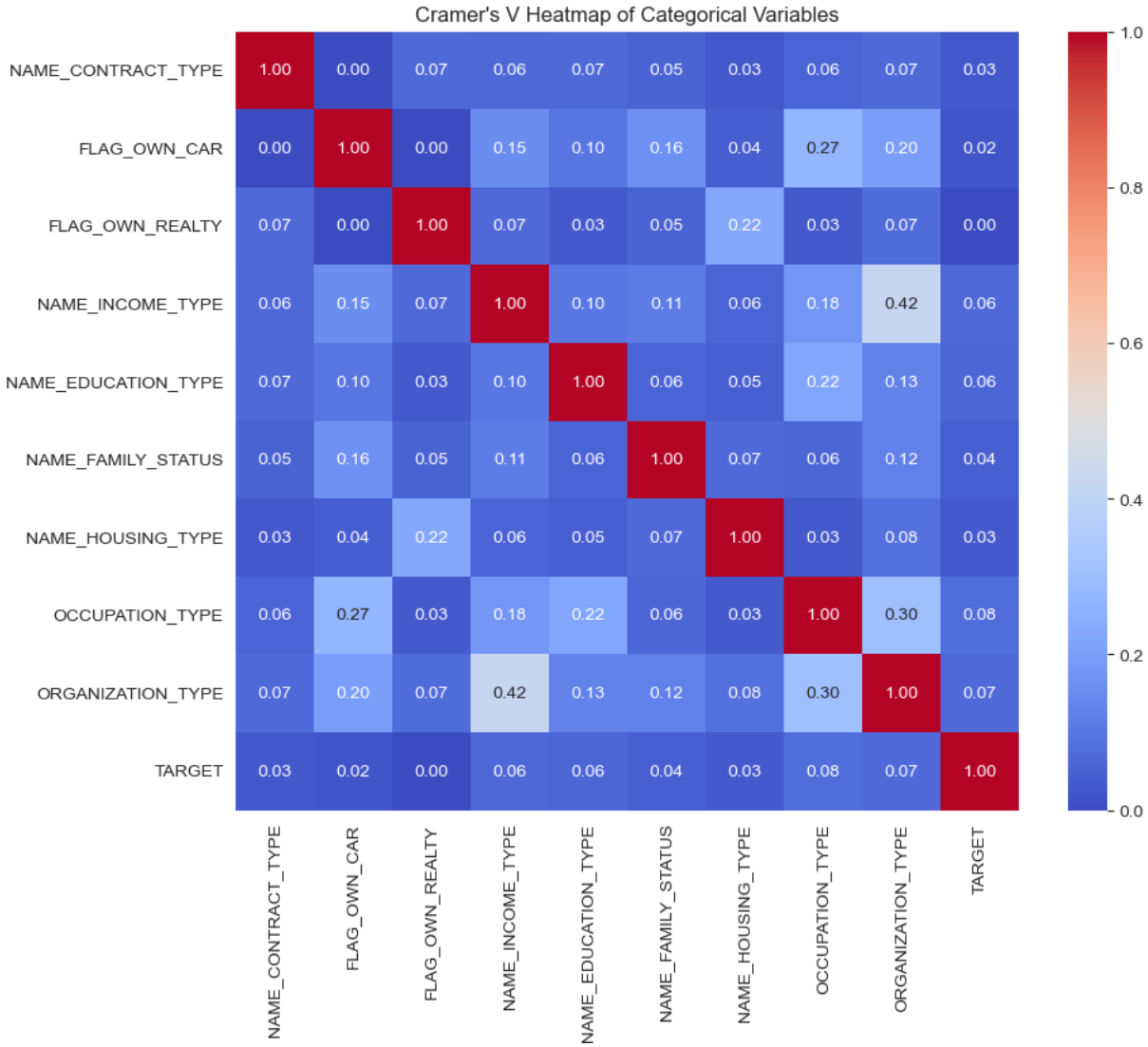
Cramer's V heatmap is a visualization tool used to examine the association between categorical variables in a dataset. It uses Cramer's V statistic, which is a measure of the strength of association between two nominal variables, to generate a heatmap that highlights these relationships. This is particularly useful in exploratory data analysis to identify potential interactions or dependencies between categorical features.

Cramer's V is defined as:

$$V = \sqrt{\frac{\chi^2}{n \cdot \min(k-1, r-1)}}$$

where: - χ^2 is the chi-squared statistic, - n is the total number of observations, - k is the number of categories in one variable, - r is the number of categories in the other variable.

Figure 2.7: Corralation for categorical variables with Cramer's Matrix



3 Data Preparation

3.1 Data Cleaning

Feature	Status	Reason
DAYS_BIRTH	Kept	Client's age in days at the time of application (relative to application)
NAME_CONTRACT_TYPE	Kept	Identification if loan is cash or revolving
CNT_CHILDREN	Kept	Number of children the client has
NAME_EDUCATION_TYPE	Kept	Level of highest education the client achieved (employment security)
FLAG_OWN_CAR	Kept	Indicates if the client owns a car (additional charge information)
FLAG_OWN_REALTY	Kept	Indicates if the client owns a house or flat (additional charge information)
NAME_HOUSING_TYPE	Kept	Client's housing situation (renting, living with parents, etc.)
NAME_FAMILY_STATUS	Kept	Family status of the client
SK_ID_CURR	Kept	ID of the loan in the sample
TARGET	Kept	Target variable indicating payment difficulties
AMT_INCOME_TOTAL	Kept	Total income of the client
AMT_CREDIT	Kept	Credit amount of the loan
AMT_ANNUITY	Kept	Loan annuity
NAME_INCOME_TYPE	Kept	Client's income type (e.g., businessman, working, maternity leave)

Continued on next page

Table 3.1 – *Continued from previous page*

Feature	**Status**	**Reason**
DAYS_EMPLOYED	Kept	Days since the client started current employment (relative to application)
OCCUPATION_TYPE	Kept	Client's occupation
REGION_RATING_CLIENT	Kept	Rating of the client's region (1, 2, 3)
ORGANIZATION_TYPE	Kept	Type of organization where the client works
AMT_REQ_CREDIT_BUREAU_MON	Kept	Number of enquiries to Credit Bureau one month before application
REGION_POPULATION_RELATIVE	Dropped	Drop - information available in other columns
AMT_REQ_CREDIT_BUREAU_QRT	Dropped	Drop - too large temporal range
AMT_REQ_CREDIT_BUREAU_YEAR	Dropped	Drop - too large temporal range
REGION_RATING_CLIENT_W_CITY	Dropped	Drop - too correlated with region rating
WEEKDAY_APPR_PROCESS_START	Dropped	Drop - not significant
AMT_GOODS_PRICE	Dropped	Drop - too correlated with credit amount
NAME_TYPE_SUITE	Dropped	Drop - no interest
DAYS_REGISTRATION	Dropped	Drop - no interest
DAYS_ID_PUBLISH	Dropped	Drop - no interest
OWN_CAR_AGE	Dropped	Drop - no interest
FLAG_MOBIL	Dropped	Drop - no interest
FLAG_EMP_PHONE	Dropped	Drop - no interest
FLAG_WORK_PHONE	Dropped	Drop - no interest
FLAG_CONT_MOBILE	Dropped	Drop - no interest
FLAG_PHONE	Dropped	Drop - no interest

Continued on next page

Table 3.1 – *Continued from previous page*

Feature	**Status**	**Reason**
FLAG_EMAIL	Dropped	Drop - no interest
HOUR_APPR_PROCESS_START	Dropped	Drop - no interest
REG_REGION_NOT_LIVE_REGION	Dropped	Drop - no interest
REG_REGION_NOT_WORK_REGION	Dropped	Drop - no interest
LIVE_REGION_NOT_WORK_REGION	Dropped	Drop - no interest
REG_CITY_NOT_LIVE_CITY	Dropped	Drop - no interest
REG_CITY_NOT_WORK_CITY	Dropped	Drop - no interest
LIVE_CITY_NOT_WORK_CITY	Dropped	Drop - no interest
AMT_REQ_CREDIT_BUREAU_HOUR	Dropped	Drop - no interest
AMT_REQ_CREDIT_BUREAU_DAY	Dropped	Drop - no interest
AMT_REQ_CREDIT_BUREAU_WEEK	Dropped	Drop - no interest
EXT_SOURCE_1	Dropped	Drop - not enough information
EXT_SOURCE_2	Dropped	Drop - not enough information
EXT_SOURCE_3	Dropped	Drop - not enough information
DAYS_LAST_PHONE_CHANGE	Dropped	Drop - not enough information
FLAG_DOCUMENT_2	Dropped	Drop - not enough information
FLAG_DOCUMENT_3	Dropped	Drop - not enough information
FLAG_DOCUMENT_4	Dropped	Drop - not enough information
FLAG_DOCUMENT_5	Dropped	Drop - not enough information
FLAG_DOCUMENT_6	Dropped	Drop - not enough information
FLAG_DOCUMENT_7	Dropped	Drop - not enough information
FLAG_DOCUMENT_8	Dropped	Drop - not enough information
FLAG_DOCUMENT_9	Dropped	Drop - not enough information
FLAG_DOCUMENT_10	Dropped	Drop - not enough information
FLAG_DOCUMENT_11	Dropped	Drop - not enough information
FLAG_DOCUMENT_12	Dropped	Drop - not enough information
FLAG_DOCUMENT_13	Dropped	Drop - not enough information
FLAG_DOCUMENT_14	Dropped	Drop - not enough information
FLAG_DOCUMENT_15	Dropped	Drop - not enough information

Continued on next page

Table 3.1 – *Continued from previous page*

Feature	**Status**	**Reason**
FLAG_DOCUMENT_16	Dropped	Drop - not enough information
FLAG_DOCUMENT_17	Dropped	Drop - not enough information
FLAG_DOCUMENT_18	Dropped	Drop - not enough information
FLAG_DOCUMENT_19	Dropped	Drop - not enough information
FLAG_DOCUMENT_20	Dropped	Drop - not enough information
FLAG_DOCUMENT_21	Dropped	Drop - not enough information
CNT_FAM_MEMBERS	Dropped	Drop - redundant with children count
APARTMENTS_AVG	Dropped	Drop - discriminative
BASEMENTAREA_AVG	Dropped	Drop - discriminative
YEARS_BEGINEXPLUATATION_AVG	Dropped	Drop - discriminative
YEARS_BUILD_AVG	Dropped	Drop - discriminative
COMMONAREA_AVG	Dropped	Drop - discriminative
ELEVATORS_AVG	Dropped	Drop - discriminative
ENTRANCES_AVG	Dropped	Drop - discriminative
FLOORSMAX_AVG	Dropped	Drop - discriminative
FLOORSMIN_AVG	Dropped	Drop - discriminative
LANDAREA_AVG	Dropped	Drop - discriminative
LIVINGAPARTMENTS_AVG	Dropped	Drop - discriminative
LIVINGAREA_AVG	Dropped	Drop - discriminative
NONLIVINGAPARTMENTS_AVG	Dropped	Drop - discriminative
NONLIVINGAREA_AVG	Dropped	Drop - discriminative
APARTMENTS_MODE	Dropped	Drop - discriminative
BASEMENTAREA_MODE	Dropped	Drop - discriminative
YEARS_BEGINEXPLUATATION_MODE	Dropped	Drop - discriminative
YEARS_BUILD_MODE	Dropped	Drop - discriminative
COMMONAREA_MODE	Dropped	Drop - discriminative
ELEVATORS_MODE	Dropped	Drop - discriminative
ENTRANCES_MODE	Dropped	Drop - discriminative

Continued on next page

Table 3.1 – *Continued from previous page*

Feature	**Status**	**Reason**
FLOORSMAX_MODE	Dropped	Drop - discriminative
FLOORSMIN_MODE	Dropped	Drop - discriminative
LANDAREA_MODE	Dropped	Drop - discriminative
LIVINGAPARTMENTS_MODE	Dropped	Drop - discriminative
LIVINGAREA_MODE	Dropped	Drop - discriminative
NONLIVINGAPARTMENTS_MODE	Dropped	Drop - discriminative
NONLIVINGAREA_MODE	Dropped	Drop - discriminative
FONDKAPREMONT_MODE	Dropped	Drop - discriminative
HOUSETYPE_MODE	Dropped	Drop - discriminative
TOTALAREA_MODE	Dropped	Drop - discriminative
WALLSMATERIAL_MODE	Dropped	Drop - discriminative
EMERGENCYSTATE_MODE	Dropped	Drop - discriminative
OBS_30_CNT_SOCIAL_CIRCLE	Dropped	Drop - discriminative
DEF_30_CNT_SOCIAL_CIRCLE	Dropped	Drop - discriminative
OBS_60_CNT_SOCIAL_CIRCLE	Dropped	Drop - discriminative
DEF_60_CNT_SOCIAL_CIRCLE	Dropped	Drop - discriminative
CODE_GENDER	Dropped	Drop - Discriminative

Table 3.1: Feature Selection and Reasons for Keeping or Dropping columns

3.2 Handling Missing Values

We developed a flag NaN function to indicate missing data to the model, rather than imputing these values. This approach is preferred because it prevents the introduction of noise or bias into the dataset. By flagging missing data, we ensure that the model is aware of the absence of information without artificially altering the dataset, which could otherwise lead to inaccurate predictions. For instance, imputing missing income data for a client could distort their financial profile and affect the model's performance. By explicitly marking missing values, we maintain the integrity and transparency of the data, which is crucial for the accuracy and explainability of our credit scoring models.

3.3 Encoding Categorical Variables

We transformed the categorical variables into numerical values to facilitate their processing within the model. This transformation is essential because most machine learning algorithms require numerical input to perform calculations and make predictions. By converting categorical data into numerical form, we can ensure that the model effectively interprets and utilizes these variables. For example, variables such as "employment type" or "loan intent" are encoded as integers or through one-hot encoding. This approach not only improves the model's performance but also enhances its ability to handle a diverse set of input features, thereby contributing to more accurate and reliable credit scoring predictions.

3.4 Quantile scale

To comprehensively test all models, we created a minimally preprocessed version for decision trees and a fully preprocessed version for validating SVM and neural network models, which are highly sensitive to scaling. Decision trees are robust to the unscaled and unprocessed data, making them suitable for a minimal preprocessing approach. Conversely, SVM and neural networks require full preprocessing, including scaling and normalization, to perform optimally. These models are sensitive to the magnitude of the data, and scaling ensures that all features contribute equally to the model's learning process. This dual approach allowed us to leverage the strengths of each model type while ensuring accurate and reliable credit scoring predictions.

3.5 Creation of Residual Columns

We have engineered additional residual columns to make them more explainable. For each additional residual column we have trained a decision tree to predict the true value of our column, based on column that could cause it, and then take the error residual. For example, the variable that counts the number active credit our clients has contracted is being predicted based on whether he owns a house or a car. This helps in better assessing the causality of our variables by removing the impact of comfounders and correlated variables.

4 Development of a Scoring Model

4.1 Selection of Machine Learning Algorithms

In the context of credit risk classification, selecting the appropriate machine learning algorithm involves balancing two crucial aspects: accuracy and explainability. To do so, we consider two types of algorithms.

1. **Black-box model:** Black-box model utilizes complex decision criteria that make it difficult to understand how input features are being transformed into outputs. Typically, these models offer superior predictive accuracy as they can capture intricate patterns and relationships within the data. Their lack of transparency makes it hard to trust their predictions, audit their decisions, or comply with regulatory requirements in sensitive applications like credit scoring.

2. **Transparent model:** Transparent models, also known as interpretable models, have decision-making processes that are grounded in logic and can be easily interpreted by humans. These models provide clear insights into how inputs are transformed into outputs. These models are easy to understand and explain, making them suitable for situations where interpretability is crucial, such as regulatory compliance and stakeholder communication. They often lack the predictive power of more complex models because they cannot capture non-linear relationships and interactions between features as effectively.

Having established the characteristics of black box and transparent models, LightGBM emerges as a compelling choice because it offers a good compromise between performance prediction and explainability.

We can now see how LightGBM balances the strengths of both to meet the demands of credit risk classification.

- Accuracy refers to the model's ability to correctly predict whether a borrower will default on a loan or not.

- Explainability refers to how easily a human can understand and interpret the model's decision-making process.

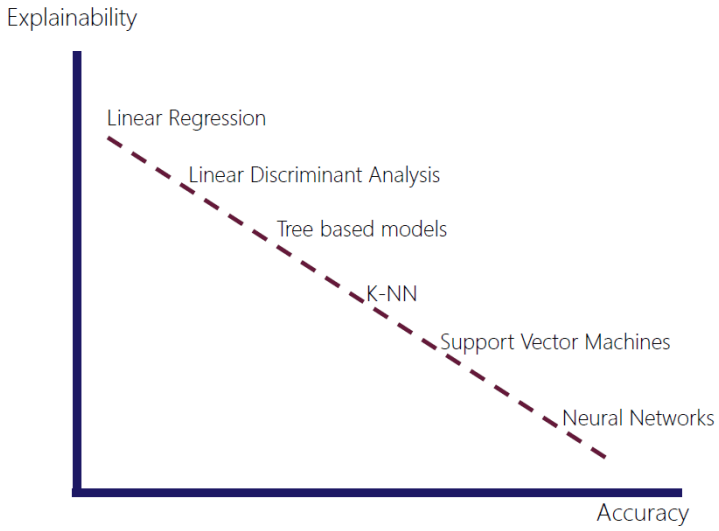
Regarding the model selection, we try to find the best trade-off between the accuracy and the explainability of the model. That's why we opt for LightGBM because it offers a good compromise between performance prediction and explainability.

- **Performance Prediction:** LightGBM is highly efficient and can handle large datasets with high dimensionality. It uses gradient boosting, which is an ensemble technique that builds multiple decision trees in a sequential manner to improve prediction accuracy. This makes LightGBM highly effective in capturing complex patterns and interactions in the data.
- **Explainability:** While LightGBM is more complex than simple linear models or decision trees, it is more interpretable than some other advanced models like deep neural networks. LightGBM provides feature importance scores, which help in understanding the influence of each feature on the model's predictions.

In credit risk classification, using LightGBM allows us to leverage a powerful predictive model while retaining a level of interpretability that is essential for trust and regulatory compliance. This balance ensures that the model not only performs well but also provides insights that can be understood and acted upon by stakeholders.

By understanding the trade-offs between accuracy and explainability, and how LightGBM effectively balances these factors, we can appreciate why it is an ideal choice for this application.

Figure 4.1: Eplainability vs. Accuracy



4.2 Modelling Approach

4.2.1 Classification Model

As explained in the previous section, for our classification task, we have chosen LightGBM model due to its efficiency and balance between accuracy and interpretability.

LightGBM is based on the gradient boosting algorithm, an ensemble learning method. Ensemble methods combine the predictions from multiple machine learning models to produce a single, superior output.

The originality, of the LightGBM is that it constructs an ensemble of decision trees sequentially. Each subsequent tree is built to correct the errors made by the previous trees. Unlike traditional decision trees, which grow level-wise, LightGBM grows trees leaf-wise. This means it expands the leaf with the maximum loss reduction, leading to better accuracy and faster training.

4.2.2 Loss Function

As the goal of our classification task is to predict whether a borrower will default (class: 1) or not (class: 0), we use the Binary Cross Entropy loss function. This loss function is well-suited for binary classification tasks.

During the training process, the model tries to minimize the ***Binary Cross Entropy Loss***. This loss gives useful feedback to the model during the training process to adjust the predictions of the model by measuring the performance of a classification model whose output is a probability value between 0 and 1. It compares the predicted probabilities with the actual class labels and penalizes the model based on the difference between them. In fact, during the training process the model tries to minimize the loss function by adding new trees that predict the residual errors (gradients) of the previous trees.

Given that this loss function is convex, it is designed to be minimized thanks to a optimization algorithm such Descent Gradient (find local minimum). Lower values indicate that the predicted probabilities are closer to the actual class labels, meaning the model is performing well. It effectively penalizes large deviations between the predicted probabilities and the actual outcomes, guiding the model to improve its predictions.

$$L = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(p_i) + (1 - y_i)\log(1 - p_i)\right] \tag{4.1}$$

where:

- N is the number of sample
- y_i is the actual label of sample i (1 for default, 0 for non-default)
- p_i is the predicted probability that sample i belongs to the positive class (default)

4.2.3 Fine-tuning model

To ensure that our LightGBM model performs optimally in predicting credit risk, we rely on the Optuna framework for hyperparameter optimization. Optuna is a flexible framework that uses Bayesian optimization to search for the best hyperparameters. The objective of this optimization process is to maximize the accuracy of our model.

To do so, Optuna uses Bayesian optimization, a method that builds a probabilistic model of the objective function and uses it to select the most promising hyperparameters to evaluate in the next iteration.

- **Define the Search Space:** The range of possible values for each hyperparameter is defined. For example, this could include the learning rate, number of leaves, and maximum depth for LightGBM.
- **Objective Function:** The objective function is defined to maximize accuracy. This function trains the LightGBM model with a given set of hyperparameters and evaluates its accuracy on a validation set.
- **Bayesian Optimization:** Optuna uses past evaluation results to build a surrogate model of the objective function. It selects hyperparameters to evaluate by balancing exploration (trying new areas of the search space) and exploitation (focusing on areas known to perform well).
- **Evaluation and Iteration:** The selected hyperparameters are evaluated by training the model and calculating the accuracy. This process is iterated many times, continually updating the surrogate model and selecting new hyperparameters to

evaluate.

- **Best Hyperparameters:** After a predefined number of iterations or when the improvement in accuracy plateaus, Optuna identifies the set of hyperparameters that yielded the highest accuracy.

By leveraging Optuna's Bayesian optimization approach, we can efficiently navigate the complex hyperparameter space and identify the optimal settings that maximize our model's accuracy. This ensures that our LightGBM model is not only accurate but also robust in predicting credit risk, thereby improving the reliability of our credit risk classification system.

The best hyperpameters found by Optuna are the following:

Hyperparameter	**Value**
n_estimators	86
max_depth	410
learning_rate	0.134
subsample	0.467
num_leaves	42
feature_fraction	0.757
sub_bin	84053

Table 4.1: Optimal Hyperparameters for LightGBM Model

4.3 Post Processing: Adjusting the decision frontier

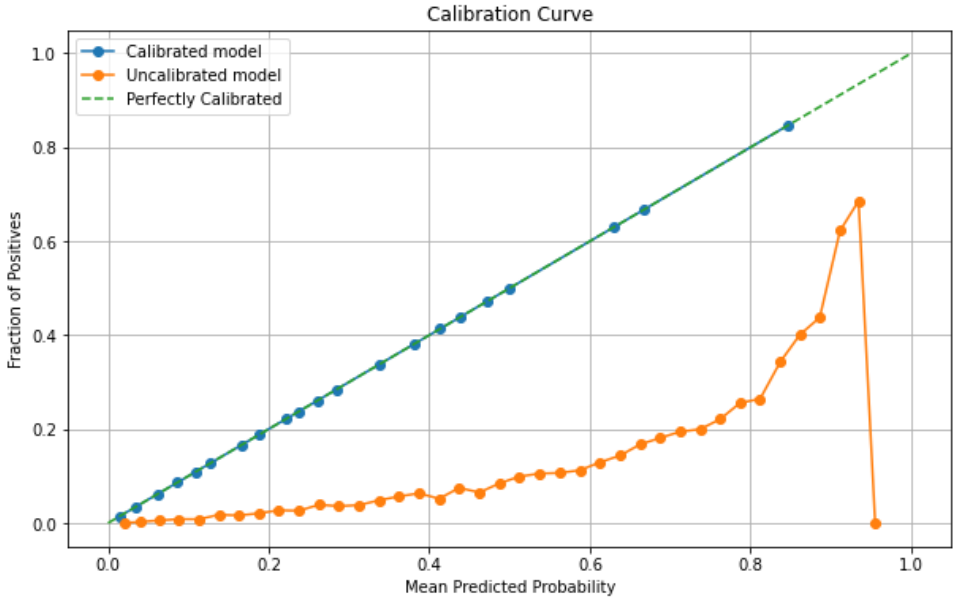
4.3.1 Model Calibration

In the context of credit risk classification, it is crucial to ensure that the model's predicted probabilities are well-calibrated. Calibration is the process of aligning the predicted probabilities with the actual likelihood of the predicted classes. A well-calibrated model ensures that the predicted probabilities reflect reality and are more reliable. For instance, if a model predicts a probability of 0.8 for a borrower defaulting, it should be the case that 80% of the time, the borrower actually defaults.

To achieve this level of reliability, we apply isotonic regression to calibrate the model. Isotonic regression is a non-parametric method that fits a piecewise constant, non-decreasing function to the predicted probabilities. The methodology involves the following steps:

- **Fit the Model:** Initially, the LightGBM model is trained and makes predictions, producing raw predicted probabilities.
- **Create Bins:** The predicted probabilities are divided into bins. For each bin, we calculate the observed frequency of the positive class (e.g., default).
- **Apply Isotonic Regression:** Isotonic regression is then applied to these bins. It adjusts the predicted probabilities to ensure they are monotonically increasing. This means that higher predicted probabilities correspond to higher actual frequencies of the positive class.
- **Transform Predictions:** The raw predicted probabilities are transformed using the fitted isotonic regression function. This step aligns the predicted probabilities with the actual likelihood observed in the training data.

Figure 4.2: Threshold



By applying isotonic regression, we enhance the reliability of the predicted probabilities. The calibrated probabilities provide a more accurate representation of the true risk, which is essential for making informed decisions in credit risk management. This postprocessing step ensures that the model's outputs are not only accurate but also trustworthy and actionable for stakeholders.

4.3.2 Choose the decision frontier

After obtaining a calibrated model, the next step is to adjust the decision threshold for classification. Adjusting the threshold helps align the model's predictions with the bank's risk management strategy. Specifically, in the context of credit risk, the worst-case scenario is granting a loan to a person who will default. To mitigate this risk, we adjust the thresholds to maximize the recall of the model, aligning with the bank's risk tolerance and liquidity considerations.

The bank's primary concern is to minimize the occurrence of defaults. By adjusting the decision threshold, we can control the balance between precision and recall. In this case, maximizing recall (the ratio of true positives to the sum of true positives and false negatives) is crucial because it ensures that most borrowers who are likely to default are correctly identified. This reduces the risk of granting loans to high-risk individuals.

In this way, setting a lower threshold, the model becomes more sensitive to identifying defaults (positive class). This means that even if there is a slight indication of default, the model will classify the borrower as a default risk. Although this might increase false positives, the bank's strategy prioritizes avoiding defaults over misclassifying safe borrowers as risky.

The threshold is adjusted based on the bank's risk tolerance and liquidity position. If the bank can afford to be conservative and avoid defaults at all costs, a lower threshold is appropriate. Conversely, if the bank can tolerate some defaults in exchange for higher loan approval rates, the threshold might be set higher.

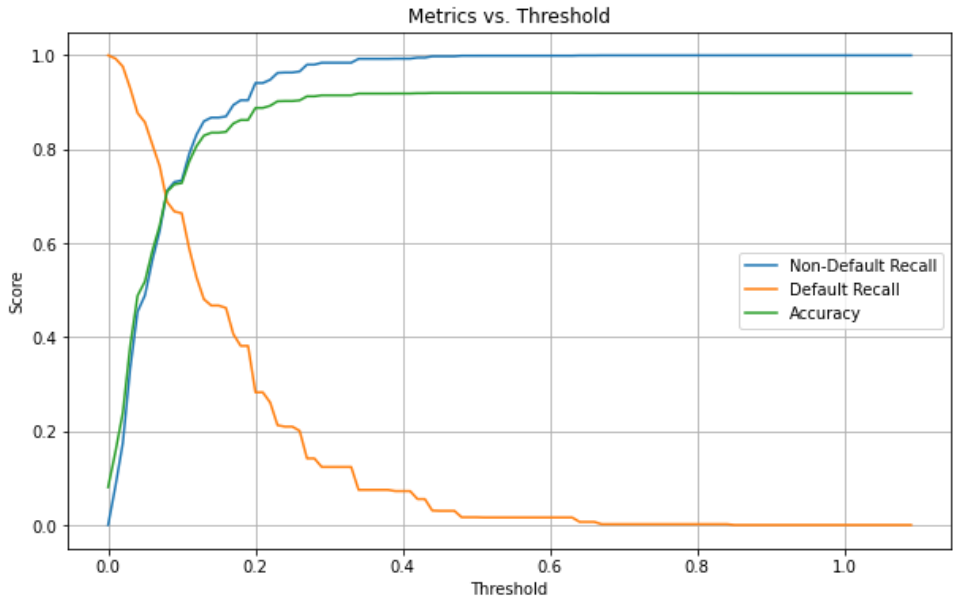
Therefore, a well-calibrated model provides predicted probabilities that accurately reflect the likelihood of default. This makes the model's predictions easier to interpret. For example, if the model predicts an 0.8 probability of default, stakeholders can trust that approximately 80% of similar cases historically resulted in defaults. This transparency helps in making informed decisions.

Knowing the exact decision threshold used by the model adds another layer of transparency. It clarifies the criteria under which loans are approved or denied. This transparency is crucial for regulatory compliance and for explaining decisions to stakeholders, including customers and auditors. Adjusting the threshold according to the bank's strategic goals

ensures that the model's decisions are not only data-driven but also aligned with business objectives.

There is a trade-off between measures such as default recall, non-default recall and model accuracy. An easy way to approximate a good starting threshold value is to examine a graph of these three measures. With this graph, we can see how each of these measures looks when we change the threshold values, and find the point at which the performance of all three is good enough to be used for credit data.

Figure 4.3: Threshold



4.4 Model Performance Evaluation

4.4.1 Classification report

The classification report provides a comprehensive overview of the model's performance metrics for each class. It includes precision, recall, F1-score, and support (the number of actual occurrences of the class in the dataset). This report helps visualize the trade-off between precision and recall for each class, allowing for a balanced assessment of model performance. It provides a quick summary of how well the model performs on each class, making it easier to identify strengths and weaknesses.

Class	Precision	Recall	F1-Score	Support
Non-Default	0.96	0.70	0.81	70687
Default	0.17	0.70	0.27	6191
Accuracy	0.70			

Table 4.2: Classification Report

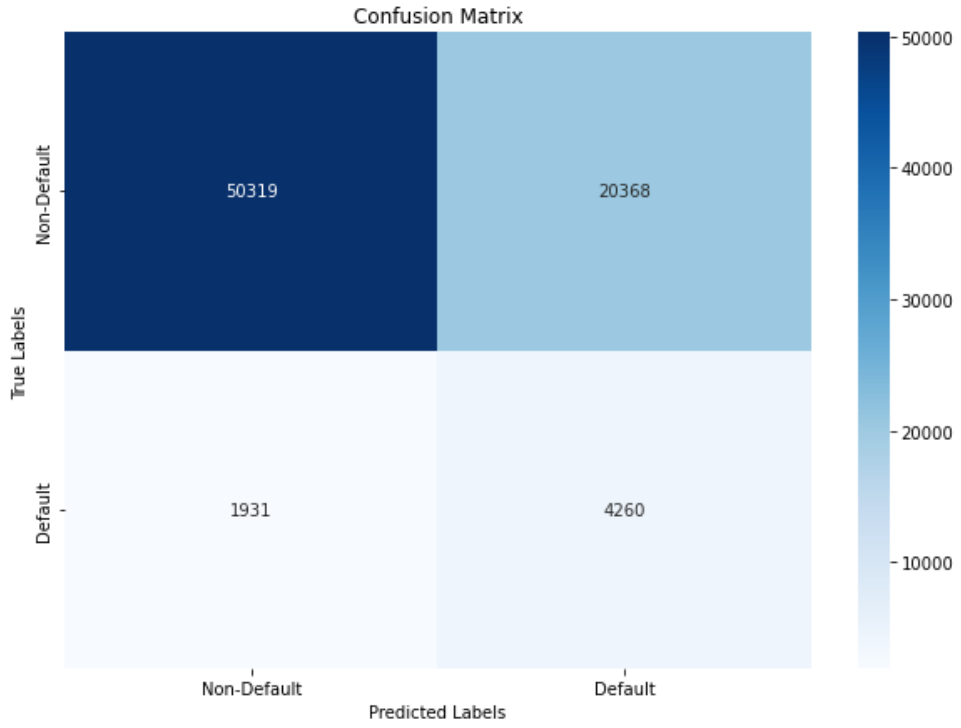
Overall, the classification report highlights that while the model performs well in identifying non-default cases with high precision and moderate recall, its performance in identifying default cases is less satisfactory. The model has a tendency to misclassify some non-default cases as default, resulting in a relatively low precision for the default class. However, the low false negative rate suggests that the model effectively identifies the majority of default cases, which is crucial in credit risk management. Overall, there is room for improvement, particularly in enhancing the precision of default predictions, to ensure a more balanced and reliable credit risk classification system.

4.4.2 Confusion Matrix

Explanation: The confusion matrix is a table that summarizes the performance of a classification algorithm. It displays the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions.

The confusion matrix helps visualize the types and frequencies of classification errors. It is particularly useful for understanding how the model performs across different classes and identifying where it tends to make mistakes.

Figure 4.4: Threshold



By choosing the optimal threshold, the classification matrix shows that the false negative rate is quite low, this means that the model accurately captures the defaulter among the population.

While the model has a good recall for the Default class, its precision is quite low, leading to many false positives. This might be acceptable in scenarios where the cost of missing a default is very high compared to the cost of incorrectly predicting a default. However, improving precision would be important to reduce the number of Non-Default instances incorrectly labeled as Default, thereby improving customer experience and reducing unnecessary credit denials.

4.4.3 ROC-AUC curve

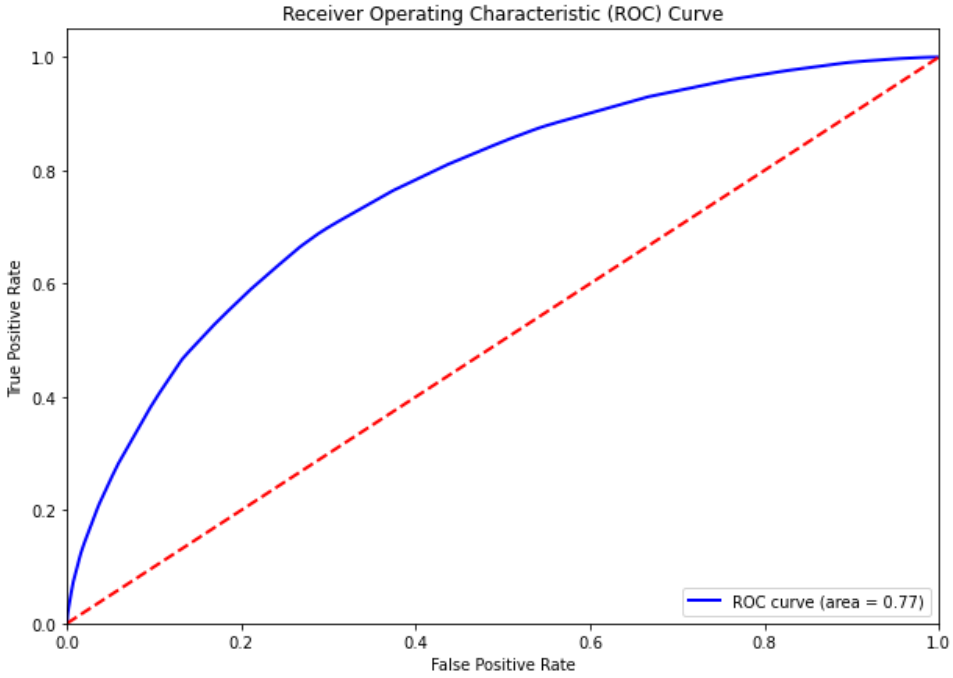
The ROC (Receiver Operating Characteristic) curve plots the true positive rate (recall) against the false positive rate (1-specificity) at various threshold settings. The AUC (Area Under the Curve) represents the degree of separability between the classes.

- True Positive Rate (Recall): Proportion of actual positives correctly identified by the model.

- False Positive Rate: Proportion of actual negatives incorrectly identified as positive by the model

The ROC curve helps visualize the trade-off between the true positive rate and false positive rate across different thresholds. The AUC provides a single value to compare the overall performance of models; a higher AUC indicates better performance. It is particularly useful for evaluating binary classifiers and comparing multiple models.

Figure 4.5: ROC-AUC curve



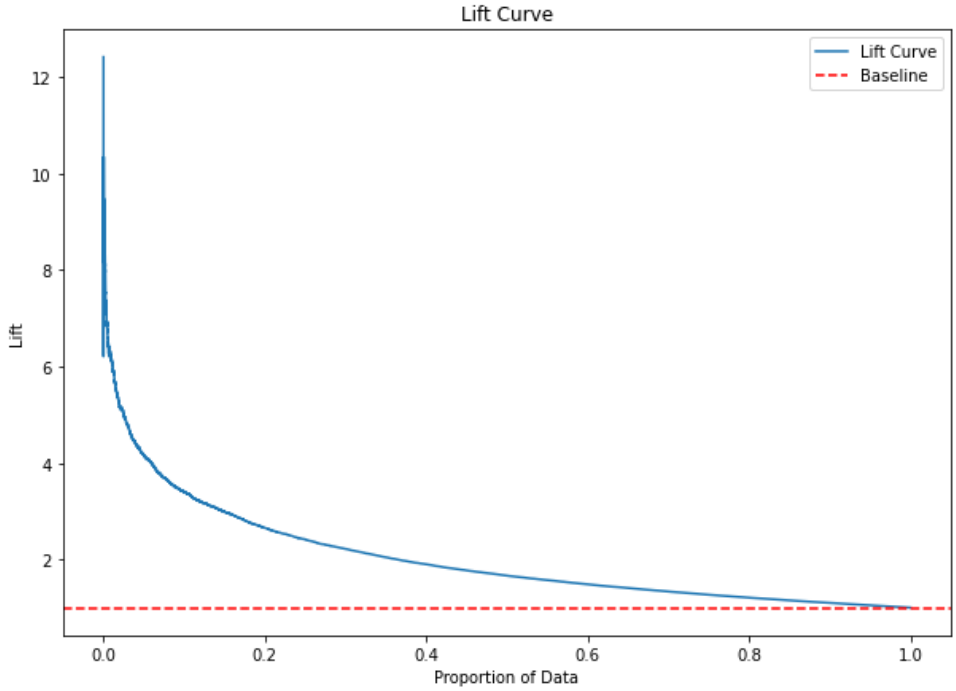
The ROC curve shows that the model has a reasonably good performance with an AUC of 0.77. This indicates that the model is fairly effective at distinguishing between defaults and non-defaults. However, efforts can be made to further improve the model's accuracy, particularly in reducing the false positive rate while maintaining or improving the true positive rate.

4.4.4 Lift Curve

The lift curve measures the effectiveness of a predictive model by comparing the model's performance to random guessing. It plots the lift, which is the ratio of the results obtained with and without the model, against the cumulative percentage of the dataset.

In the context of credit risk classification helps to understand how well the model is identifying borrowers who are likely to default compared to a random selection.

Figure 4.6: Lift curve



At the beginning of the curve (left part), the lift is pretty high. This indicates that the model is very effective at identifying defaulters in the top-ranked predictions.The curve gradually declines as more of the population is considered. This decline occurs because as we move to the right, we are including more of the population, which will include more non-defaulters.

Overall, we can see that the lift curve remains above 1. This indicates that the model is consistently better than random guessing.

By focusing on the top-ranked predictions with the lift curve we can identify borrowers who are much more likely to default. This helps in making informed lending decisions, such as declining high-risk loans or offering them at higher interest rates to compensate for the risk.

4.5 Credit Risk optimization strategy

So far, we've used simple assumptions and checks to set threshold values to determine loan status based on the predicted probability of default. With these values, we used

code like this to define a new loan status based on probability and threshold. These new loan status values have an impact on our model's performance measures, as well as on the estimated financial impact on the portfolio. If we have three loans with these default probabilities, those above the threshold are considered defaults and those below are considered non-defaults.

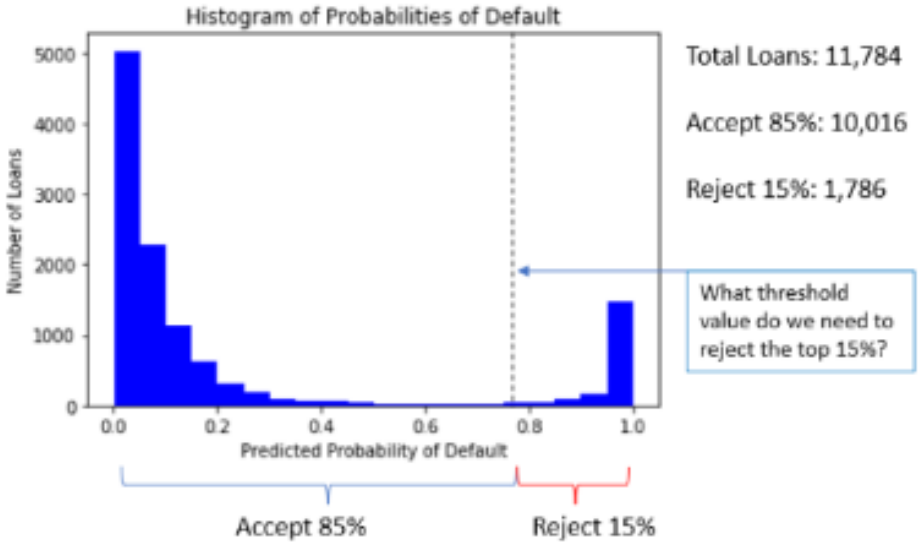
We have seen before that our models have already predicted default probabilities, and we can use these probabilities to calculate the threshold. Since the threshold is used to determine what constitutes a default or non-default, it can also be used to approve or reject new loans as they come in. As an example, let's assume that our test set is a new batch of new loans. Before calculating the threshold, we need to understand a concept known as the acceptance rate. This is the percentage of new loans we accept in order to keep the number of defaults in a portfolio below a certain number.

To do so, we implement a strategy which consists of accepting or rejecting a loan according to an acceptance rate. The acceptance rate, is the percentage of new loans we accept in order to keep the number of defaults in a portfolio below a certain number. By adjusting this acceptance rate, we find the best thresholds that minimize the bad rate which is the percentage of accepted loans that are defaults. Hence, minimizing the bad rate leads to maximize the portfolio estimated net value and allow us to better manage the risk according to the economic conditions.

4.5.1 Acceptance rate

If we want to accept 85% of all loans with the lowest probability of default, our acceptance rate is 85%. This means that we reject 15% of all loans with the highest probability of default. Instead of setting a threshold value, we want to calculate it to separate the loans we accept using our acceptance rate from the loans we reject. This value will not be the same as the 85% we used as our acceptance rate.

Figure 4.7: Default probabilities default



Here we can see where the threshold lies within the range of predicted probabilities. We can see not only how many loans will be accepted (left-hand side), but also how many loans will be rejected (right-hand side). I recommend that you rerun this code with different threshold values to better understand how this affects the acceptance rate.

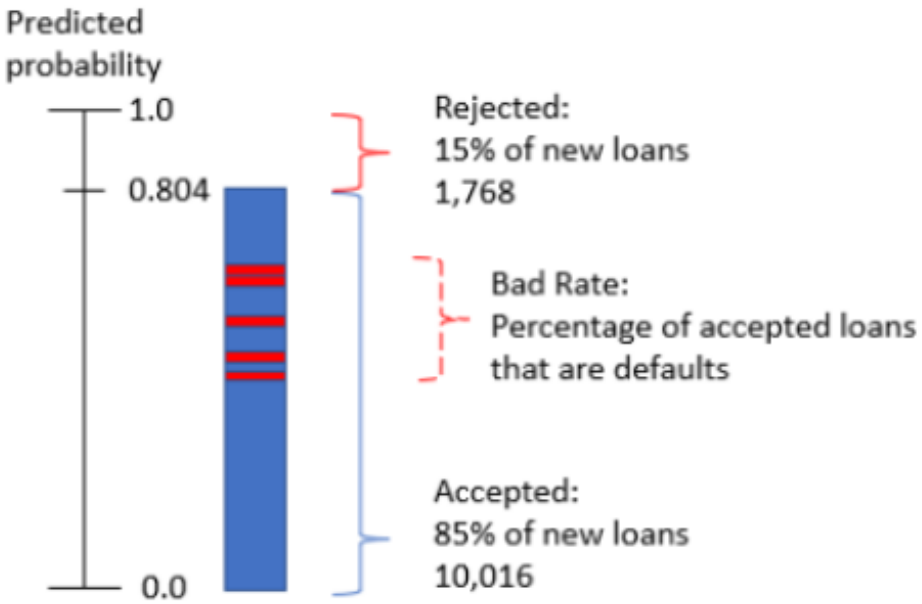
In our example, the threshold is 0.804. This means that all our new loans with a probability of default of less than 80% are accepted, and all higher probabilities are rejected.

4.5.2 Bad Rate Calculation

The calculation of the bad rate is the number of defaults in our accepted loans divided by the total number of accepted loans.

$$\frac{\text{Accepted Defaults}}{\text{Total Accepted Loans}}$$

Figure 4.8: Bad rate



Even if we have calculated an acceptance rate for our loans and set a threshold, there will always be defaults in our accepted loans. These are often in probability ranges where our model has not been properly calibrated. In our example, we have accepted 85% of the loans, but not all of them are non-defaulting loans as we would like. The bad rate is the percentage of accepted loans that are actually defaults. Thus, our bad rate is a percentage of the 10,016 loans accepted.

Then we look at the amount of each loan to understand the impact of acceptance rates on the portfolio. To do so, we use cross-tabulations with calculated values, such as the average loan amount, of the new set of loans. To do this, multiply the number of each loan by the average loan amount.

Figure 4.9: Cross table for acceptance of 15%

```
Acceptance rate:  0.15
Bad Rate: 0.011
Threshold:  0.018
nb accepted loan: 11335
pred_loans_status              0             1
true_loan_status
0                   2.096807e+07  1.112741e+08
1                   2.375932e+05  1.134461e+07
```

The next step is to compute the bad rate for several acceptance rate. Hence we can

obtain a strategy table. To build this strategy table, for each acceptance rate we define, we calculate the threshold, store it for later, apply it to the loans to separate our set of loans into two subsets: accepted loans and rejected loans. Then we create a subset called accepted loans and we calculate and store the bad rate. According each bad rates, we compute the estimated portfolio value by calculating the difference between the average value of non-defaulting loans accepted and the average value of defaulting loans accepted.

Figure 4.10: Strategy Table

Strategy Table

	Acceptance Rate	Bad Rate	...	Avg Loan Amount	Estimated Value
0	1.00	0.081	...	1870.81	1.205246e+08
1	0.95	0.065	...	1870.81	1.251270e+08
2	0.90	0.062	...	1870.81	1.259899e+08
3	0.85	0.052	...	1870.81	1.288664e+08
4	0.80	0.048	...	1870.81	1.300170e+08
5	0.75	0.042	...	1870.81	1.317429e+08
6	0.70	0.039	...	1870.81	1.326058e+08
7	0.65	0.035	...	1870.81	1.337564e+08
8	0.60	0.035	...	1870.81	1.337564e+08
9	0.55	0.029	...	1870.81	1.354823e+08
10	0.50	0.025	...	1870.81	1.366329e+08
11	0.45	0.023	...	1870.81	1.372082e+08
12	0.40	0.022	...	1870.81	1.374959e+08
13	0.35	0.022	...	1870.81	1.374959e+08
14	0.30	0.019	...	1870.81	1.383588e+08
15	0.25	0.013	...	1870.81	1.400847e+08
16	0.20	0.013	...	1870.81	1.400847e+08
17	0.15	0.011	...	1870.81	1.406600e+08
18	0.10	0.007	...	1870.81	1.418106e+08
19	0.05	0.004	...	1870.81	1.426735e+08

Then we visualize the results of our portfolio optimization strategy with several plots

Figure 4.11: Portfolio Estimated Net Value over the Acceptance Rate

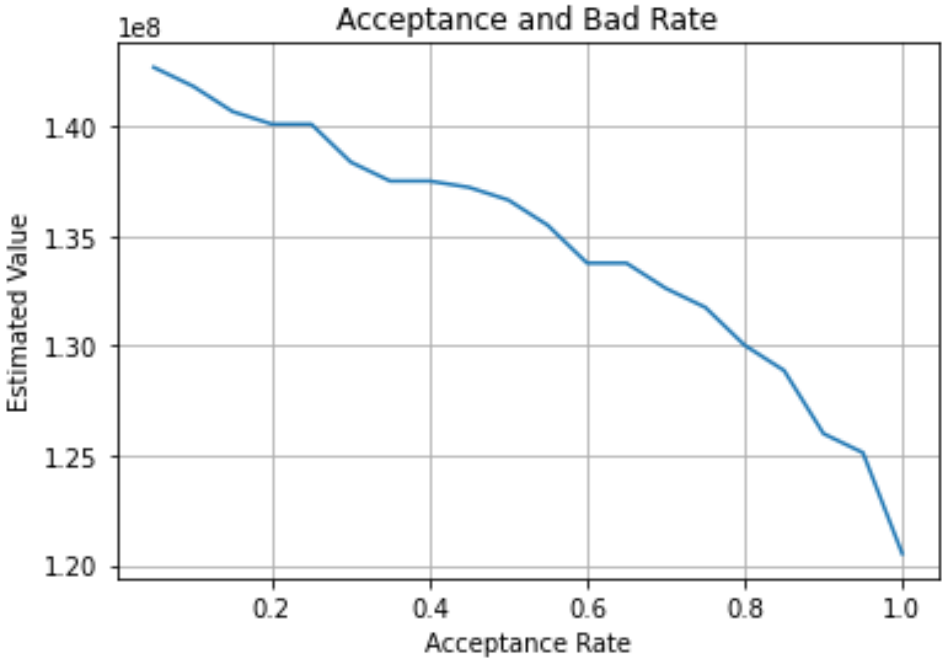


Figure 4.12: Portfolio Bad Rate over the Acceptance Rate

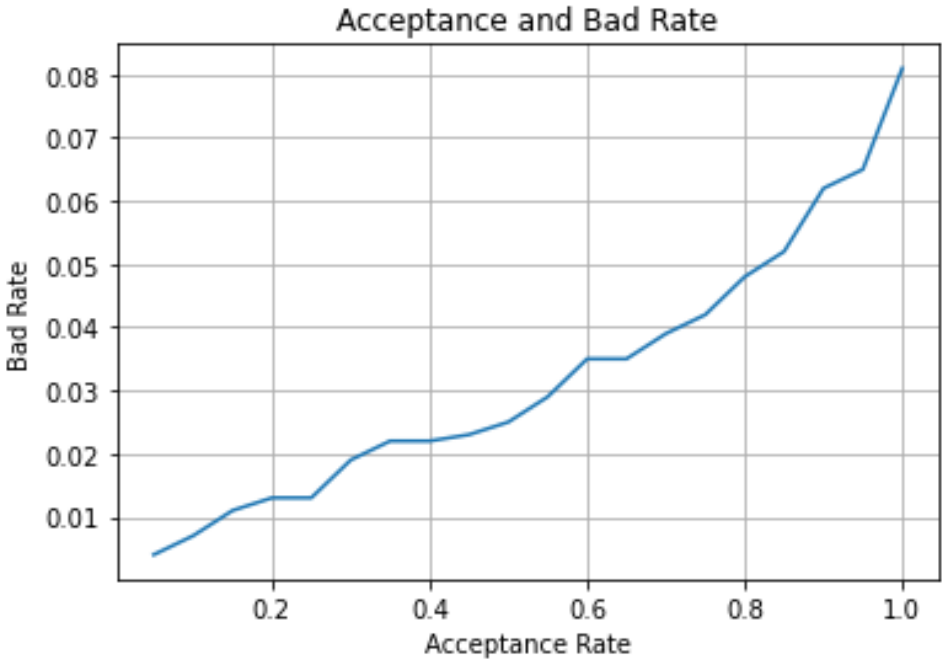


Figure 4.13: Box plot of the portfolio optimization strategy thresholds

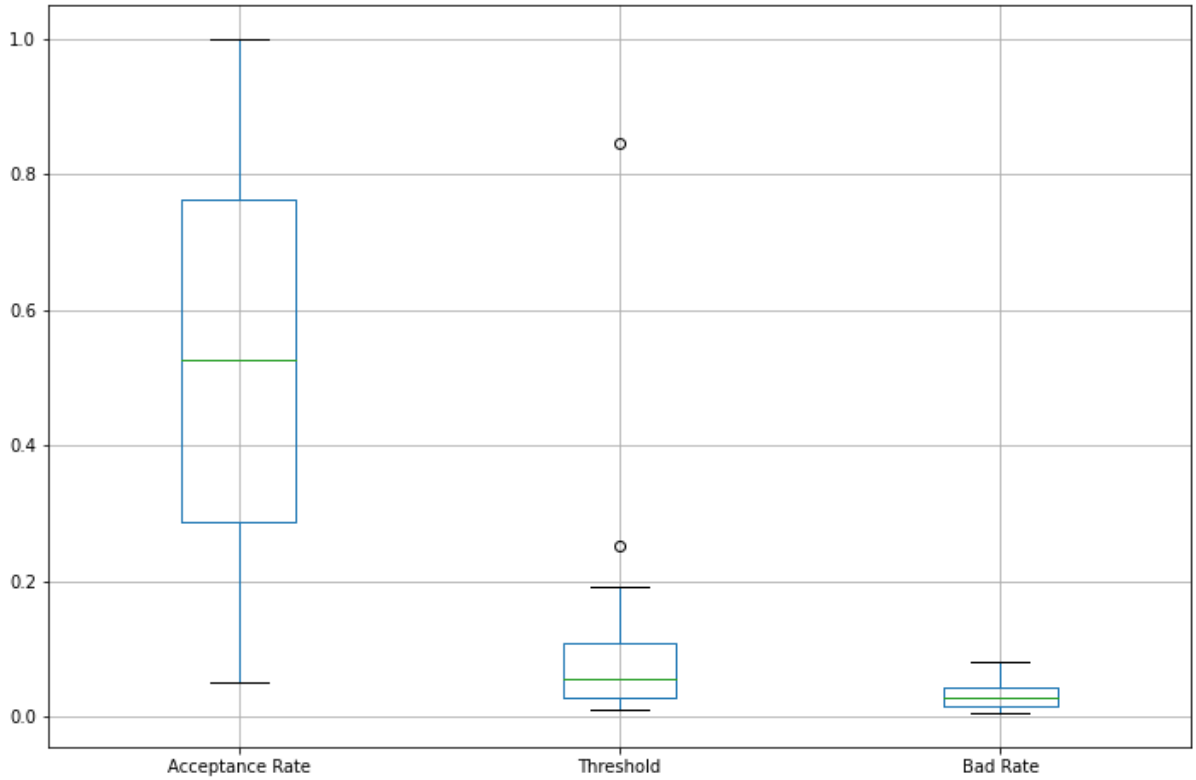
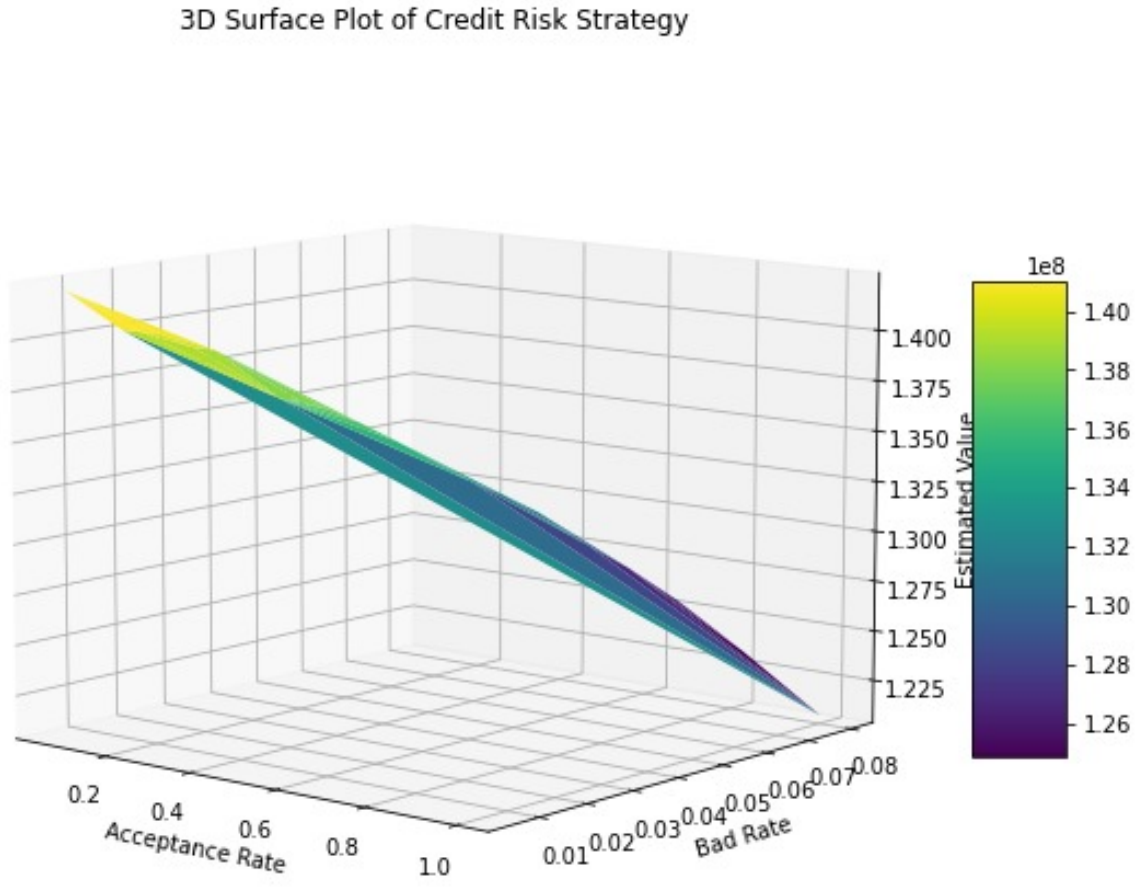


Figure 4.14: Portfolio credit risk optimization strategy



The final way of measuring the financial impact of our predictions is the total expected loss. This represents the amount we expect to lose if a loan defaults, given its probability of default. We take the product of the probability of default, the loss given default and the exposure at default for each loan, and add them together. In our predictive data framework, we'll use the probability of default, the exposure will be assumed to be the total value of the loan, and the loss given default will be equal to 1 for a total loss on the loan.

$$EL = PD \times LGD \times EAD$$

where:

- PD is the Probability of Default.
- LGD is the Loss Given Default (expressed as a fraction or percentage).
- EAD is the Exposure at Default (the amount of money the lender is exposed to at the time of default).

The Total expecred loss of our portfolio is Total expected loss: $10,737,417.19

5 Implementation of Explainable AI (XAI)

5.1 Introduction to Explainable AI

5.1.1 What is Explainable AI?

Explainable Artificial Intelligence (XAI) encompasses a range of processes and methods designed to help users understand and trust the outcomes produced by machine learning algorithms. XAI is essential for interpreting AI models, identifying potential biases, and ensuring that AI systems operate with transparency, fairness, and accountability.

5.1.2 Why Explainable AI Matters

The significance of XAI can be seen in several areas. First, it builds trust and confidence by enabling users to understand AI decision-making processes. Second, it helps in evaluating model accuracy and fairness, ensuring that AI systems do not perpetuate biases related to race, gender, age, or location. XAI is also crucial for meeting regulatory requirements, providing transparency, and allowing individuals affected by AI decisions to challenge or comprehend those decisions. Finally, XAI supports operational accountability, enabling organizations to maintain auditability and manage risks associated with AI deployment.

5.1.3 How Explainable AI Works

XAI uses various techniques to provide insights into AI decision-making processes. These techniques can be grouped into three main areas: prediction accuracy, traceability, and decision understanding. Methods like Local Interpretable Model-Agnostic Explanations (LIME) improve prediction accuracy by explaining individual predictions. Traceability techniques, such as DeepLIFT (Deep Learning Important Features), show dependencies and activation pathways within neural networks. Decision understanding focuses on educating users about AI decision-making processes to build trust and facilitate human-AI collaboration.

5.2 What are the Benefits of Explainable AI?

Explainable AI offers several benefits. It helps build trust and confidence in AI models by ensuring their decisions are interpretable and explainable. It also enhances business outcomes by allowing systematic monitoring and management of models, thus speeding up the time to achieve results. Additionally, XAI helps mitigate risks and costs associated with AI model governance by reducing the need for manual inspections and preventing costly errors. Overall, XAI supports the responsible and effective deployment of AI systems across various industries.

5.2.1 Permutation Feature Importances

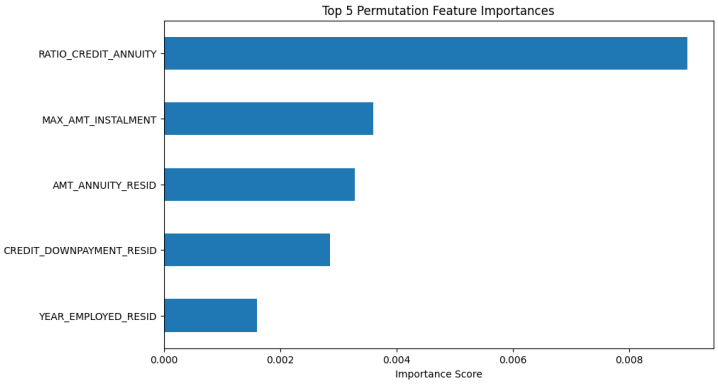


Figure 5.1: Top 5 Permutation Feature Importances

Permutation feature importance is a technique used to assess the impact of each feature on a machine learning model's predictions. It involves randomly shuffling the values of a feature and measuring the resulting change in the model's performance. The greater the decrease in performance after shuffling a feature, the more important that feature is considered to be. This process helps identify which features are most influential in making accurate predictions.

How Permutation Feature Importance Works

The importance of a feature X_j is calculated as the difference between the model's performance on the original data and the performance after shuffling the values of feature X_j. Mathematically, it can be expressed as:

$$\text{Importance}(X_j) = \text{Perf}_{\text{baseline}} - \text{Perf}_{\text{shuffled}(X_j)}$$

where: - $\text{Perf}_{\text{baseline}}$ is the performance metric (e.g., accuracy) of the model on the original data. - $\text{Perf}_{\text{shuffled}(X_j)}$ is the performance metric after shuffling the values of feature X_j.

This equation quantifies the importance of each feature in influencing the model's predictions.

5.2.2 Partial Dependence Plots

Partial dependence plots (PDPs) are a visualization technique used to understand the relationship between a feature and the predictions made by a machine learning model while marginalizing the effects of all other features. PDPs show how the predicted outcome changes as a single feature varies, while averaging out the effects of all other features.

How Partial Dependence Plots Work

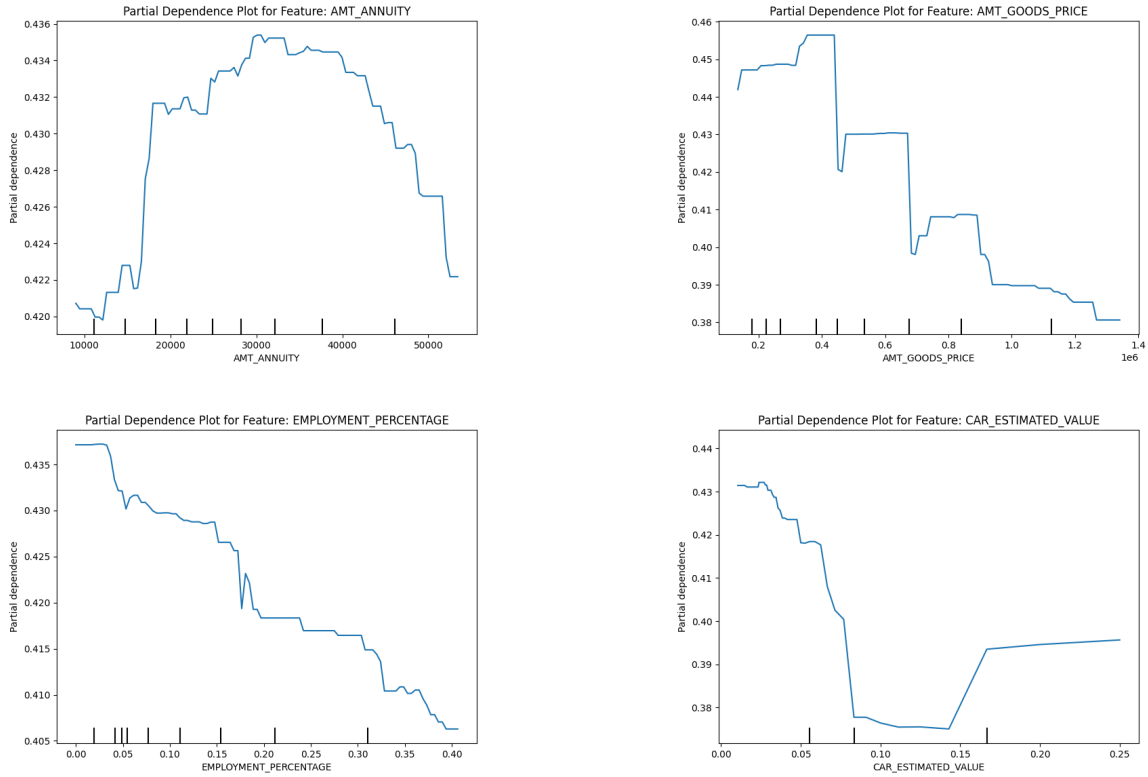


Figure 5.2: Partial Dependence Plots for Four Features

The partial dependence of the predicted outcome $\hat{y}$ on a feature X_j is calculated by averaging the predictions of the model over all possible values of X_j, while keeping the values of all other features fixed. Mathematically, it can be expressed as:

$$\text{Partial Dependence}(X_j) = \frac{1}{N}\sum_{i=1}^{N} \hat{y}_{\text{partial}(X_j)}(X_{\text{other},i})$$

where: - N is the number of observations in the dataset. - $\hat{y}_{\text{partial}(X_j)}(X_{\text{other},i})$ is the predicted outcome when feature X_j is varied across all possible values, while the values of all other features $X_{\text{other},i}$ are fixed. - The sum is taken over all observations in the dataset, and the average is computed.

This equation represents the partial dependence of the predicted outcome on a single feature, providing insights into how changes in that feature affect the model's predictions. **Results:** While we ran partial dependence plots for all the features in our dataset, we thought it would be more appropriate to show them for some of the features.

We can clearly notice that the higher the price of the good the power the probability of the dependence, the higher the yearly instalment, the higher the probability of default up to the threshold of 35K USD. We also notice that the higher the estimated value of the car the lower the probability of default.

5.2.3 SHAP (SHapley Additive exPlanations)

SHAP (SHapley Additive exPlanations) values are a technique used to explain the output of machine learning models by attributing the contribution of each feature to the model's predictions. They provide a unified measure of feature importance and explainability.

How SHAP Values Work

SHAP values are based on Shapley values from cooperative game theory, which allocate the contribution of each feature to the prediction by considering all possible combinations of features. They quantify the impact of each feature by comparing the model's prediction when including the feature with its actual value against a baseline prediction.

The SHAP value ϕ_j for a feature X_j can be calculated as:

$$\phi_j = \sum_{S \subseteq N \setminus \{j\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!}[\text{val}(S \cup \{j\}) - \text{val}(S)]$$

where: - N is the set of all features. - S represents a subset of features excluding X_j. -

val($S \cup \{j\}$) is the model's prediction when including feature X_j along with the features in subset S. - val(S) is the model's prediction when considering only the features in subset S. - $|S|$ and $|N|$ denote the cardinality of sets S and N, respectively.

This equation computes the SHAP value for each feature, indicating its contribution to the model's prediction. SHAP values provide insights into the importance and impact of individual features on the model's output. **Results:**

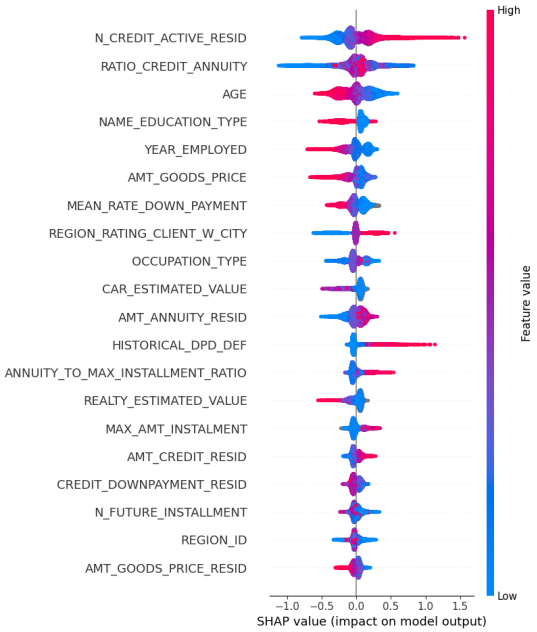


Figure 5.3: Shap Values by feature and impact on the model prediction

- Features like **N_CREDIT_ACTIVE_RESID**, and **YEAR_EMPLOYED** have clear, strong impacts on the prediction, indicating their importance in the model.
- **NAME_EDUCATION_TYPE**
- The spread of SHAP values for features like **RATIO_CREDIT_ANNUITY** and **MEAN_RATE_DOWN_PAYMENT** suggests they have a more nuanced influence on the prediction, depending on their specific values.
- **AGE** Being older is associated with a lower impact towards predicting default.
- Having a higher than expected number of active loans contributes significantly towards a default prediction by the model.
- A higher duration of employment tends towards a non-default prediction.

- Previous behavior of late payments, contributes towards a default prediction.

5.2.4 Conformal learning

Conformal learning is a statistical framework that provides a method for constructing predictive models with reliable measures of uncertainty. It is particularly useful in situations where understanding the confidence of predictions is crucial. Unlike traditional machine learning models, which output point predictions or probabilities without a clear indication of their reliability, conformal learning aims to produce prediction sets that are valid with a specified probability.

The core idea behind conformal learning is to use past data to define a conformity measure, which quantifies how typical or atypical a new example is compared to the training data. This measure is then used to determine the prediction interval or set for new examples, ensuring that the true label is contained within this interval with a specified confidence level.

- **Conformity Score:** A measure of how typical or atypical a new credit applicant's data is compared to the historical data used to train the model. Lower scores indicate higher conformity and lower risk.
- **Calibration:** The process of adjusting conformity scores using a separate calibration set to achieve the desired coverage probability.
- **Prediction Set:** A range of possible outcomes for a new applicant that is guaranteed to include the true outcome with a specified confidence level (e.g., 95%).

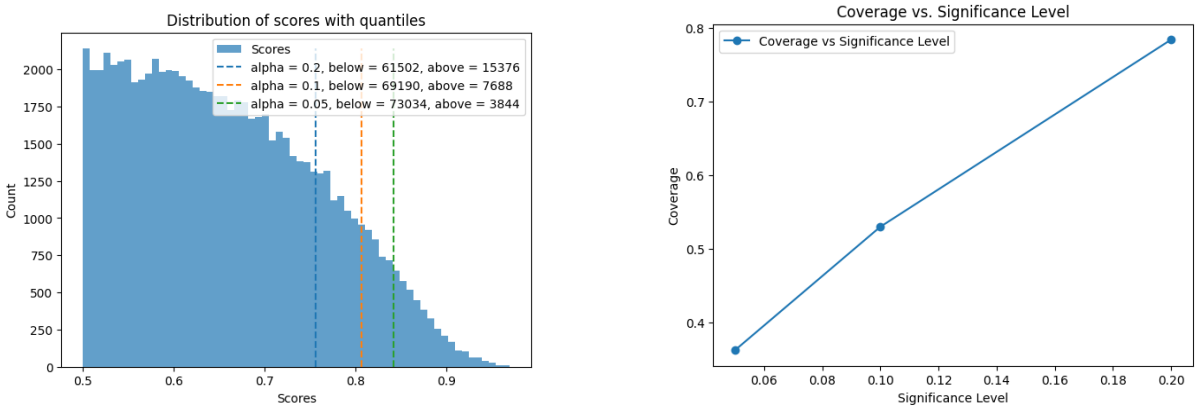


Figure 5.4: Distribution of conformal scores and coverage versus significance levels.

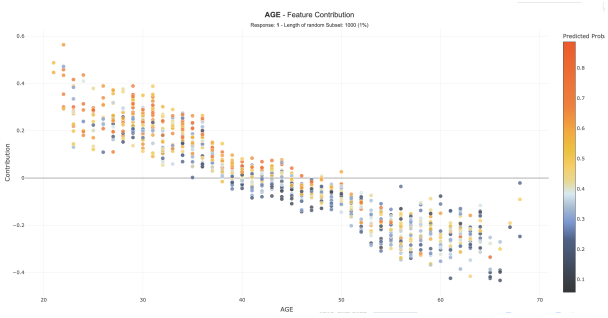


Figure 5.5: Enter Caption

Coverage Values

The coverage values for different significance levels (alphas) provide insights into the performance of the conformal prediction model. Here's the interpretation of the coverage values:

- At $\alpha = 0.2$, the coverage is 78.43%, indicating that approximately 78.43% of the predictions fall within the corresponding prediction sets.
- At $\alpha = 0.1$, the coverage is 52.99%, suggesting that around 52.99% of the predictions are within the prediction sets for this significance level.
- At $\alpha = 0.05$, the coverage is 36.21%, signifying that roughly 36.21% of the predictions fall within the prediction sets at this significance level.

These coverage values help evaluate the reliability and accuracy of the conformal prediction model at different confidence levels.

Distribution of conformity scores

The statistics for each alpha value provide insights into the distribution of conformity scores relative to the specified confidence intervals. Let's interpret these results:

- **Alpha: 0.2**
  - Below: 61,502
  - Above: 15,376

  Interpretation: For $\alpha = 0.2$, the threshold is set such that 80% of the conformity scores in the calibration set fall below it. Therefore, 80% of the predictions for the

test set are within the specified confidence interval, indicating moderate confidence.

- **Alpha: 0.1**
  - Below: 69,190
  - Above: 7,688

  Interpretation: For $\alpha = 0.1$, the threshold is set to include 90% of the conformity scores. Thus, 90% of the test set predictions fall within the confidence interval, providing higher confidence compared to $\alpha = 0.2$.

- **Alpha: 0.05**
  - Below: 73,034
  - Above: 3,844

  Interpretation: For $\alpha = 0.05$, the threshold includes 95% of the conformity scores, resulting in 95% of the test set predictions being within the confidence interval, indicating the highest level of confidence.

5.2.5 DICE

Counterfactual Analysis with DICE

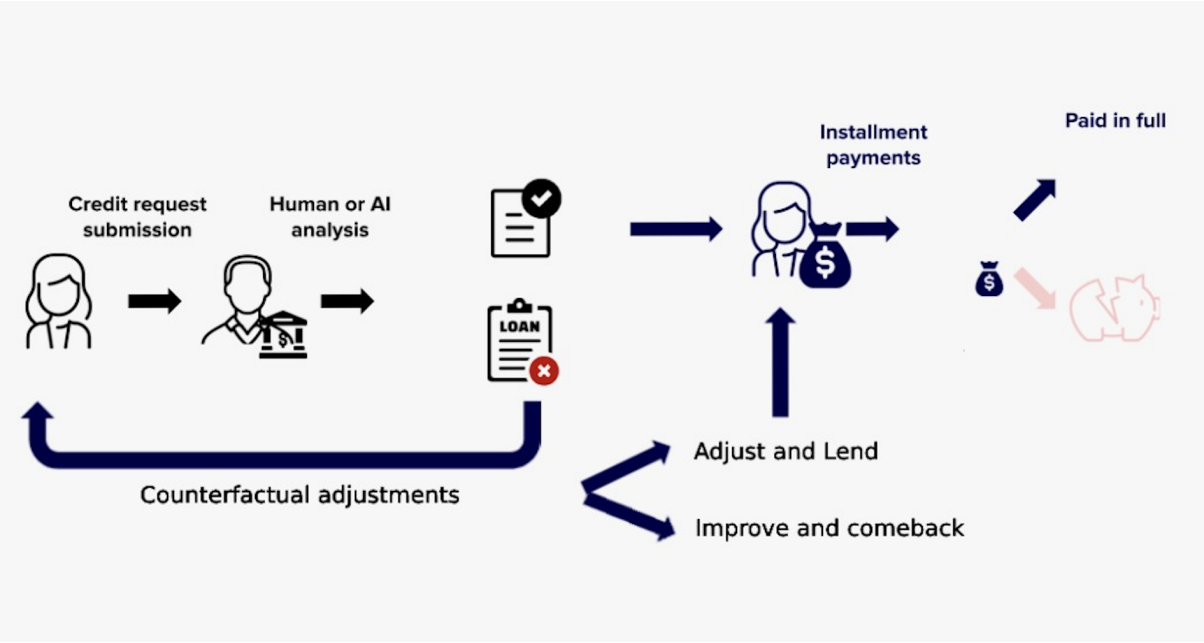


Figure 5.6: DICE explanation

Counterfactual analysis is a crucial tool for enhancing decision-making processes in credit scoring. The DICE (Diverse Counterfactual Explanations) framework operates by generating alternative scenarios that could lead to different outcomes. For example, if a customer's loan application is rejected, DICE can suggest specific changes the customer can make to their financial profile to improve their chances of approval in the future.

For the bank, DICE helps identify risk boundaries, inform decision-making, and ensure compliance with regulations. By understanding the counterfactual scenarios, banks can delineate clear guidelines on what changes customers need to make to qualify for loans. This not only streamlines the approval process but also aligns with regulatory requirements by providing transparent and justifiable decisions.

For the customer, DICE offers personalized advice, enhances trust, and provides valuable data insights. By showing customers what specific actions they need to take, such as reducing debt or increasing savings, DICE empowers them with actionable steps to improve their financial standing and reapply with a higher likelihood of success. This transparency and guidance foster a stronger, trust-based relationship between the customer and the bank.

The graph depicts the flow of credit request submission, analysis by human or AI, and the outcomes based on counterfactual adjustments. Customers receive installment payments if approved, and if not, they get insights on how to improve and reapply. This cyclical process of adjustment and reapplication underscores the continuous improvement facilitated by DICE, ultimately leading to better financial health for customers and more informed lending decisions for banks.

How to use it : an example ?

SK_ID_CURR	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	RATIO_CREDIT_ANNUITY	N_CREDIT_ACTIVE_RESID	CAR_ESTIMATED_VALUE
105455	835380.0	40320.0	675000.0	20.71875	0.007282	NaN

Figure 5.7: Situation of the profile with the ID number 105455

Counterfactual analysis with DICE can be illustrated through the following example.

In the provided tabs, we see a customer's credit application data and we decided to work on two scenarios. These scenarios suggest changes that could lead to different credit approval outcomes.

1. Initial Data: The customer's original data includes all the variables. We decided to constraint the model to only leave AMT_CREDIT, AMT_ANNUITY, AMT_GOODS_PRICE, RATIO_CREDIT_ANNUITY, N_CREDIT_ACTIVE_RESID, and CAR_ESTIMATED_VALUE to vary. The initial values indicate that the customer's loan request might not be approved due to certain financial metrics and we want to give him some recommendations.

2. Scenario 1: DICE proposes a counterfactual scenario where the AMT_ANNUITY is significantly reduced to 5259.4. This adjustment increases the RATIO_CREDIT_ANNUITY to 44.8, which might improve the customer's chances of loan approval by indicating better affordability.

AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	RATIO_CREDIT_ANNUITY	N_CREDIT_ACTIVE_RESID	CAR_ESTIMATED_VALUE
835380.0	5259.4	675000.0	44.8	0.007282	0.000000

Figure 5.8: Scenario 1

3. Scenario 2: Another counterfactual scenario suggests a different adjustment, with AMT_ANNUITY remaining high at 40320.0 but modifying CAR_ESTIMATED_VALUE to 0.063068. This scenario also modifies the RATIO_CREDIT_ANNUITY to 39.3, which could also be favorable for the customer's creditworthiness.

AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	RATIO_CREDIT_ANNUITY	N_CREDIT_ACTIVE_RESID	CAR_ESTIMATED_VALUE
835380.0	40320.0	675000.0	39.3	0.007282	0.063068

Figure 5.9: Scenario 2

These examples demonstrate how DICE generates alternative scenarios by adjusting key financial variables. By exploring these counterfactuals, customers can understand the specific changes needed to improve their credit scores and increase their chances of loan approval. For the bank, this process helps in providing personalized advice and making informed lending decisions.

5.3 Analysis and Interpretation of Results

5.3.1 SHAPASH Monitor

Shapash is a user friedly extension built on top of the Shap framework. It's main advantage is the interactive report that allows users to analyse the most influential features, with a built-in parameter that inverse transforms the preprocessing steps for easier interpretability.

5.3.2 Consistency and Logic of Predictions

As we proceed to evaluate the predictive efficacy of our models, it is imperative to consider three foundational pillars that are critical for credit assessment. These pillars not only guide our expectations but also influence how we interpret the model's performance.

1. Stability The first pillar is the stability of our clients. Given the inherent risks in lending, it is crucial to minimize uncertainties. A primary source of uncertainty is the potential for significant changes in a client's circumstances, such as employment shifts or relocation. Such changes could render the initial credit assessment data obsolete, thereby increasing the financial risks over the loan's duration.

2. Finances The financial soundness of the client's project is the second pillar. Questions to consider include whether the client is financially overextending themselves, whether the project is viable, and if the client can maintain their standard of living while repaying the debt on time. These considerations are vital for a thorough and effective credit assessment.

3. Trust The third pillar focuses on trust. Credit granting is fraught with risks, including unpredictable risks affecting that emanates from unexpected events and accidents, and risks stemming from information asymmetries due to clients concealing relevant information. To foster a sound financial system, efforts must be made to reduce these asymmetries and cultivate trustworthy relationships with clients, thus allowing more flexibility on our part.

For each of these pillars, we have identified a few features that are directly linked to these dimensions and have looked at their impact on credit risks. We believe Age and the Number of years a client has been employed for to have a negative impact on the probability of default. Younger clients are more likely to experience impactful changes in their lives as they have more flexible lives. We may also expect this relationship to reverse at latter points in life as health becomes a growing concern. The same reasoning holds for the number of years a client has been employed for, the longer you have been employed the more likely you are to keep being employed.

Following the same thought process, we acknowledge that certain professions are more exposed to instabilities potentially due to high turnover in their workplace. We are mostly thinking about low-skill labors and unsecure industries with high turnover such as restaurants, drivers... As we can see in the following graph the model's predictions align

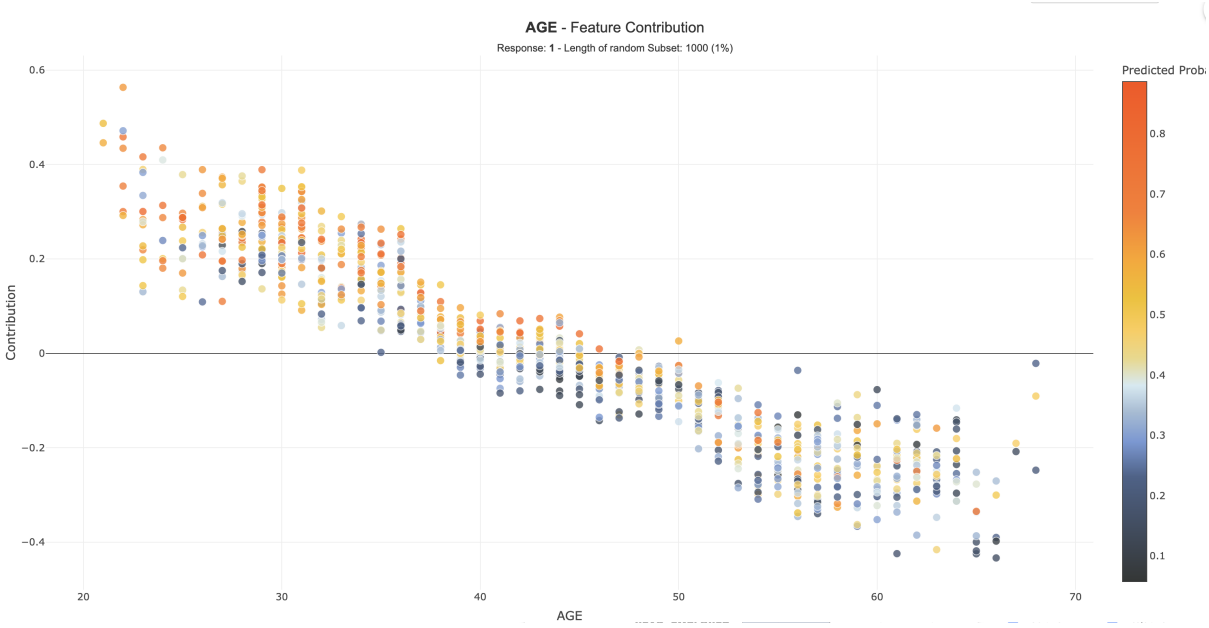


Figure 5.10: Age - Feature Contribution

with our hypotheses. Low-skill laborers (drivers, cleaning staff, cooking staff, low-skill laborers, laborers) are expected to have a higher probability of default by the model whereas high-skill labor (accountants, medicine staff, core staff) have a negative impact on a clients' probability of default.

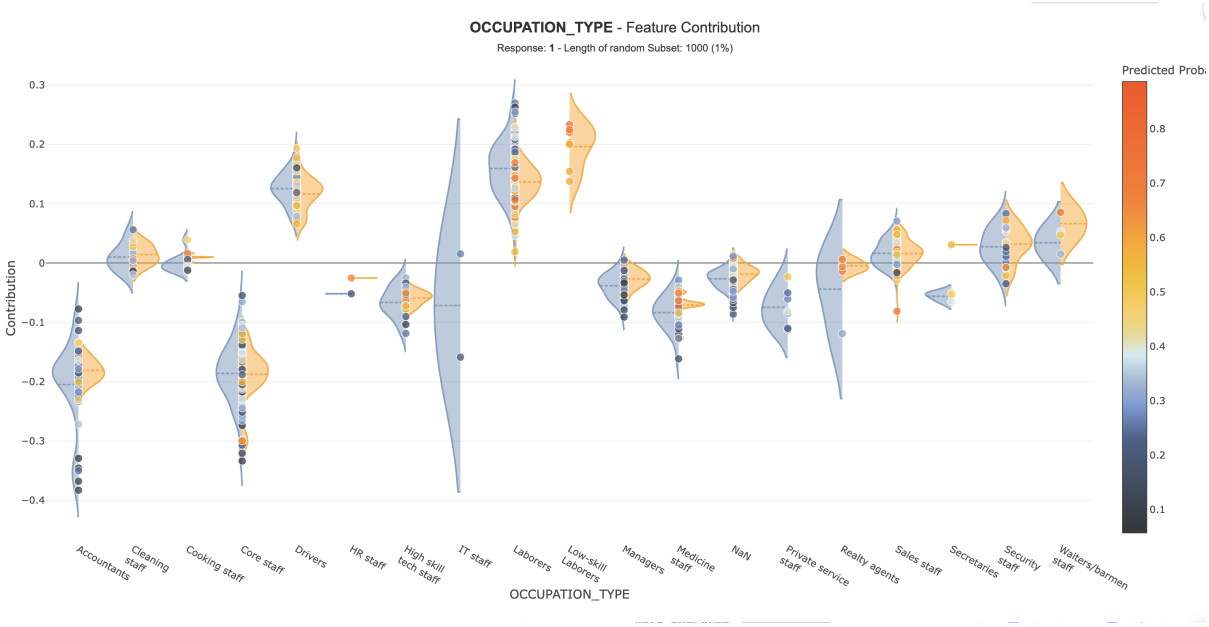


Figure 5.11: Occupation type - Feature Contribution

The number of excess credit a client has is expected to be indicative of over-indebtedness and we expect it to be positively correlated with one's probability of default. As we can see from the model's results, this feature is the most influential feature for predicting

credit default, thus aligning with our thought process.

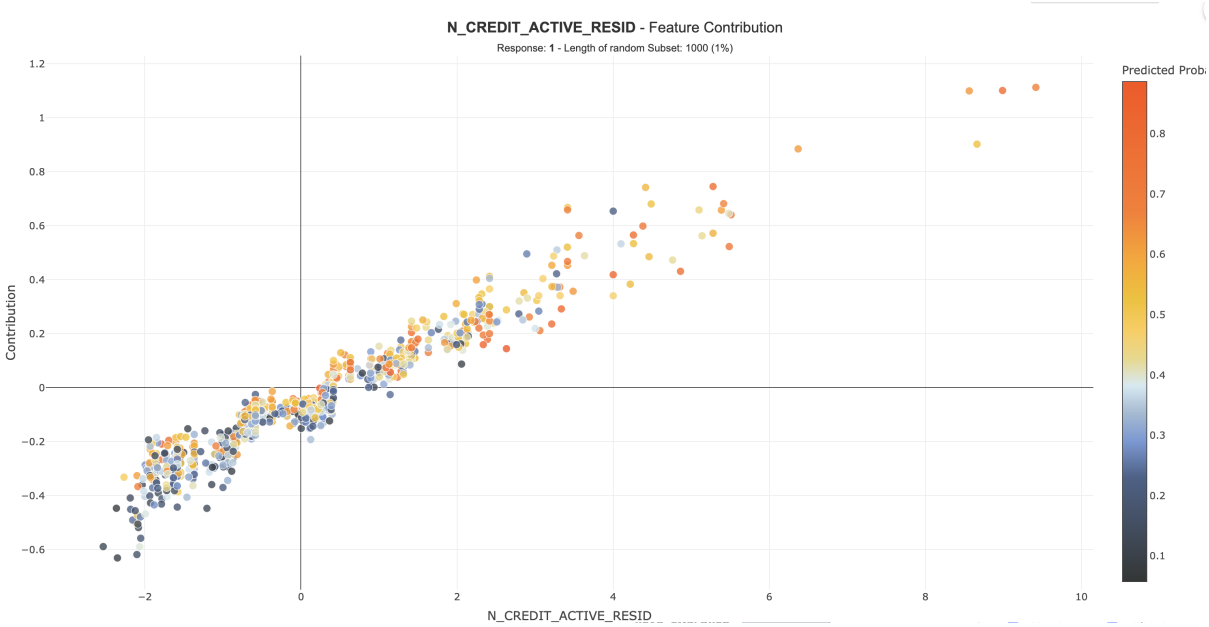


Figure 5.12: Credit active residuals - Feature Contribution

The excess amount annuity of a client could be indicative of one's overconfidence in its ability to repay its loan rapidly and thus we expect it to be positively linked to the probability of default. Once again, the model aligns with our expectations with clients having excessive annuity payments being more prone to defaulting.

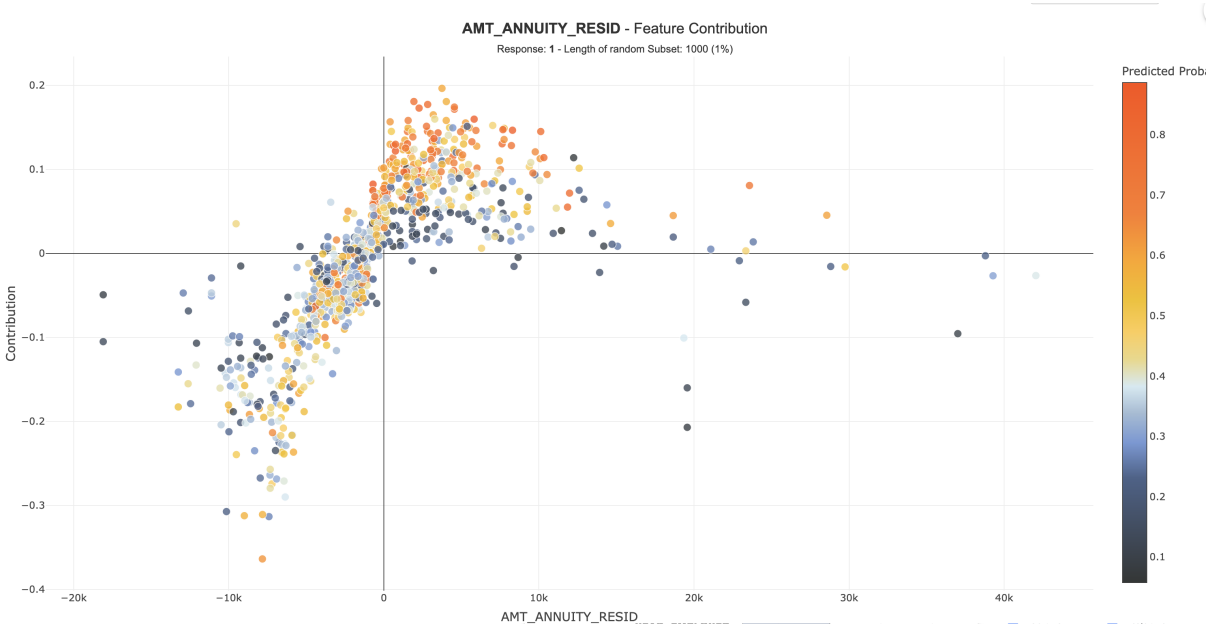


Figure 5.13: Amount Annuity residuals - Feature Contribution

We believe that the number of days our clients misses due payments of loans that meet certain value thresholds is either linked to financial distress, or would indicate a clients lack

of commitments towards his contractual obligations. There seems to be a clear positive decreasing relationships between our target and the number of days past due.

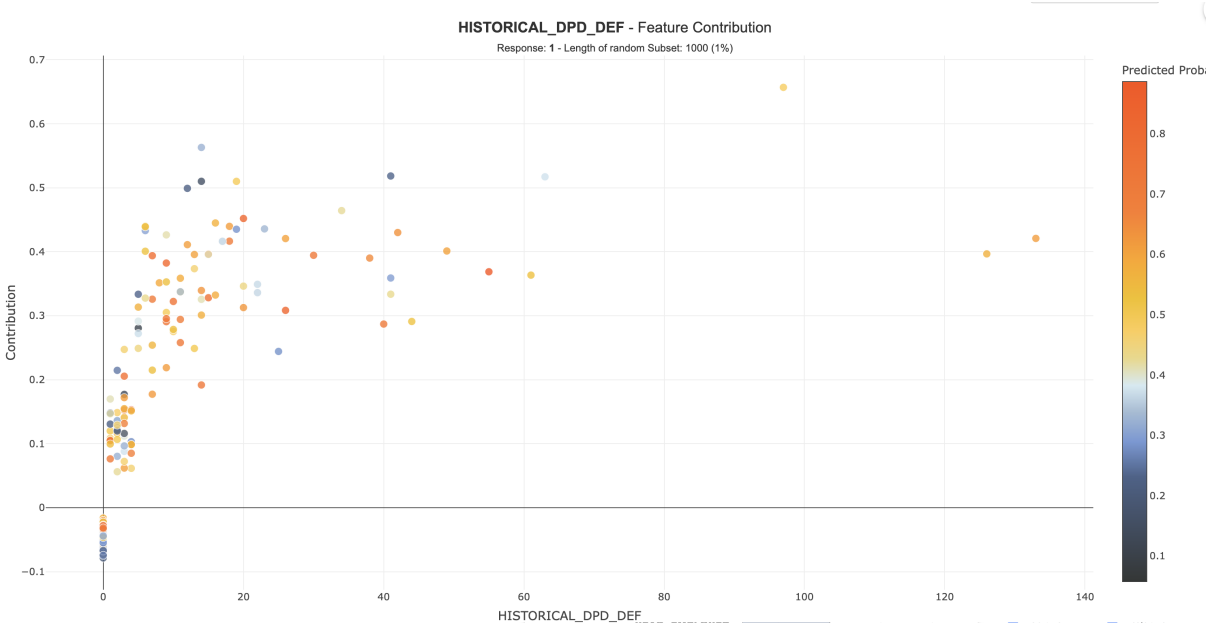


Figure 5.14: Historical Past due - Feature Contribution

On the other hand, we expect excess down payment to be a signal of our client's commitment towards his contractual obligations by having "skin in the game" and sharing part of the risk with the bank. Furthermore it also indicates that our client's is likely to have cash on hands which could serve as a safety cushion if he were to face financial difficulties. Thus we would expect it to be negatively correlated to credit default. As we can see from the following graph the relation seems clear and aligns with our thought process.

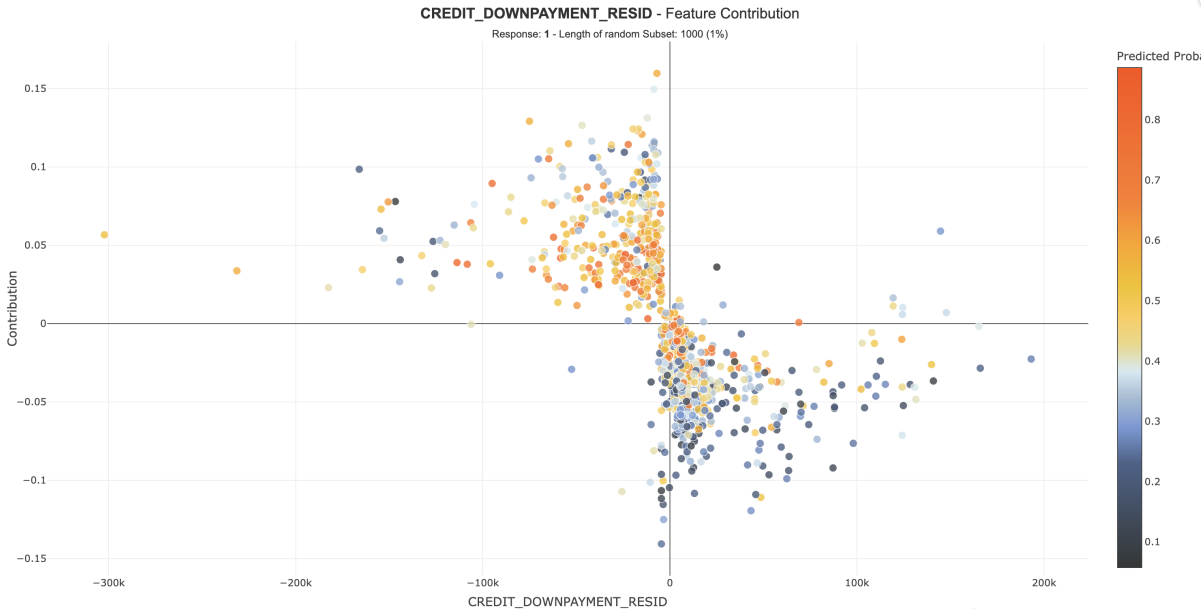


Figure 5.15: Mean rate downpayment - Feature Contribution

6 Conclusion

The objective of this project was to evaluate the feasibility and effectiveness of integrating Explainable Artificial Intelligence (XAI) into credit risk management. We developed and validated credit scoring models using various machine learning algorithms, including Random Forest, Support Vector Machine (SVM), XGBoost, and Neural Networks, leveraging the Home Credit Default Risk dataset. We decided to keep the LGBM in the rest of our study. Subsequently, we implemented XAI frameworks such as SHAP, LIME and DICE to elucidate model predictions and enhance interpretability.

Our analysis demonstrated that XAI provides meaningful insights into the factors influencing credit risk predictions, thereby improving the transparency and reliability of credit scoring systems. Key findings include the identification of critical features such as age, income, credit amount, employment length, and home ownership, which significantly impact credit decisions. The XAI techniques also highlighted potential biases and inconsistencies within the models, enabling us to address these issues and improve the models' fairness and accuracy.

We employed a dual preprocessing strategy to test different models: minimal preprocessing for decision trees and full preprocessing for SVM and neural networks. This approach allowed us to leverage the strengths of each model type while ensuring accurate and reliable credit scoring predictions.

Our evaluation of the models' performance through metrics such as accuracy, precision, recall, F1-score, ROC-AUC curve, and lift curve indicated that while the models performed well overall, there is room for improvement, particularly in enhancing the precision of default predictions. The confusion matrix and ROC curve analyses provided additional insights into the models' strengths and weaknesses, guiding further refinements.

In conclusion, this project underscores the value of integrating XAI into credit risk management. By making AI models more interpretable, XAI enhances trust and facilitates better decision-making processes within financial institutions. The insights gained from this project can guide future investments in XAI technologies and support the development of more transparent and trustworthy credit scoring systems.

Furthermore, the implementation of XAI frameworks revealed several key benefits and considerations:

1. **Model Agnostic** : XAI frameworks are straightforward to use and can be implemented with any model of your choice, offering flexibility in model selection.

2. **Easy to Use** : These tools can be used in conjunction with a credit officer to provide insight into the model's decision-making process, facilitating better understanding and collaboration.

3. **Computation** : XAI tools are computationally intensive, necessitating both precomputation and real-time computation. This requires robust computational resources for efficient implementation.

4. **Human Evaluation** : It is recommended to have human supervision to ensure that the recommendations made by the model are logical and actionable. Human evaluation adds a layer of trust and verification, crucial for decision-making in sensitive areas like credit risk.

In conclusion, this project underscores the value of integrating XAI into credit risk management. By making AI models more interpretable, XAI enhances trust and facilitates better decision-making processes within financial institutions. The insights gained from this project can guide future investments in XAI technologies and support the development of more transparent and trustworthy credit scoring systems.

References

Angelopoulos, A. N., Bates, S. (2021). A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. Journal of Machine Learning Research, 22, 1-41. URL: https://jmlr.org/papers/volume22/20-1367/20-1367.pdf

Shafer, G., Vovk, V. (2008). A Tutorial on Conformal Prediction. Journal of Machine Learning Research, 9, 371-421. URL: https://jmlr.csail.mit.edu/papers/volume9/shafer08a/shafer08a.pdf

Zeng, G. (2022). Metric Divergence Measures and Information Value in Credit Scoring. *International Journal of Financial Studies*, 10(1), 15.