MMF2030 MACHINE LEARNING FOR FINANCE

Home Credit Default Risk Project Individual Component

Xinyi (Cynthia) Shen, 1005778428
xinyi.shen@mail.utoronto.ca

Master of Mathematical Finance

University of Toronto

December 18, 2024

Table of Contents

1 Individual Business Write-up

1.1 Business Insights

1.1.1 Business Value of Findings and Application

The business goal of credit risk modeling is to classify non-default/low credit risk/low payment difficulty and default/high credit risk/high payment difficulty customers.

Summary of Selected Model: XGBoost Tuned on ROC-AUC Score

- Tuned XGBoost ROC AUC on Test Set: 0.75
- Maximum Profit on Test Set: 35,513
- Optimal Threshold for Maximum Profit: 0.16

Class	Precision	Recall	F1-score	Support
non-default	0.94	0.90	0.92	56538
default	0.25	0.38	0.30	4965
Accuracy	0.86 (61503 samples)			
Macro Avg.	0.60	0.64	0.61	61503
Weighted Avg.	0.89	0.86	0.87	61503

Table 1: Classification Report for Tuned XGBoost (ROC AUC)

The XGBoost model is designed to predict the likelihood of loan default, enabling efficient credit risk assessment. The ROC AUC score of 0.75 indicates a good discriminatory power in distinguishing between default and non-default clients. The non-default precision of 94% suggests a high confidence in identifying non-default clients. The default recall of 39% shows a moderate ability to identify default clients. The overall accuracy of 86% with a weighted F1-score of 87% demonstrates a strong overall performance.

Business Value

(1) Enhanced Credit Risk Management: Our findings contributes to optimal decision making. The optimal threshold of 0.16 balances false negatives (missed defaults) and false positives (incorrectly flagged defaults) to maximize profit. The selected model prioritizes minimizing defaults while retaining profitable clients, directly aligning with business profitability goals. Also, our findings turn risk mitigation proactive by identifying high-risk clients, enabling the business to take preemptive actions such as rejecting applications, offering smaller loans, or increasing interest rates to mitigate potential losses.

(2) Improved Financial Inclusion: The model's high precision for non-default clients ensures that low-risk individuals are accurately identified and granted credit access. This contributes to the organization's objective of expanding the customer base while maintaining portfolio quality.

(3) Strategic Insights for Policy Formulation: Our model adjust credit approval strategies dynamically based on economic conditions or business objectives by tuning the threshold. For example, in a strong economy, a slightly higher threshold can increase loan approval rates, while in a downturn, a stricter threshold reduces risk exposure. The model's outputs enable segmentation into risk tiers (e.g., low, medium, high). This segmentation can inform tailored products such as higher interest rates for high-risk clients and promotional offers for low-risk clients to improve market penetration.

(4) Optimized Operational Efficiency: By focusing on high-risk clients, the model reduces the burden on manual underwriting processes, improving efficiency and reducing costs. The model can be integrated into automated decision-making systems for real-time credit scoring, accelerating loan approvals and enhancing customer experience.

Application for the Model

(1) Deployment in Loan Approval Systems: We can integrate the model into the loan underwriting process to provide automated risk scores for applicants. or incorporate the profit-maximizing threshold (0.16) into the decision-making framework for loan approvals.

(2) Portfolio Monitoring: We can use the model for ongoing monitoring of existing loan portfolios to identify clients at higher risk of default or enable dynamic portfolio rebalancing by adjusting thresholds as market conditions or client behaviors evolve.

(3) Tailored Risk-Based Pricing: We can leverage the model's insights for risk-based pricing strategies, offering differentiated interest rates and terms based on risk scores.

(4) Fraud Detection: We can use model outputs to flag anomalous applications that deviate significantly from typical client behavior, complementing existing fraud detection mechanisms.

Business Strategy Implementation

(1) Threshold Calibration: It is suggested to monitor profits and ROC AUC periodically to recalibrate the optimal threshold and maintain profitability.

(2) Model Maintenance: The business needs to continuously retrain the model with fresh data to improve predictive accuracy and ensure relevance.

(3) Employee Training: The company needs to train staff on interpreting model outputs and incorporating them into credit decision workflows.

(4) Communication and Transparency: The employer should clearly communicate the risk-based model decisions to stakeholders and clients to build trust in the decision-making process.

1.1.2 Profit Curve and Optimal Strategy

The profit curve serves as a critical tool for evaluating the financial performance of the credit risk model under varying classification thresholds. The curve identifies the threshold that maximizes profit, providing a quantitative framework to guide strategic decisions.

Business Context and Key Assumptions of Profit Curve

The following assumptions are made for constructing the profit curve.

(1) Correct Classification Benefit (True Negative):

A correctly classified **non-default** customer (low-risk customer) generates a **benefit of 1 unit**. This represents the financial gain from approving loans to reliable borrowers without risk of default.

(2) Misclassification Cost (False Negative):

A misclassified **default** customer (high-risk customer) leads to a **cost of 5 units**. This accounts for financial losses due to unpaid loans or defaults.

(3) Threshold Variation:

The model outputs predicted probabilities of default, which are compared against a classification threshold to determine loan approval or rejection.

Customers with probabilities **below threshold** are classified as non-default, thus **approving** the loan.

Customers with probabilities **above threshold** are classified as default, thus **rejecting** the loan.

Profit Formula

The profit is calculated using the following formula:

$$\text{Profit} = (\text{True Negative} \times \text{Benefit}) - (\text{False Negative} \times \text{Cost})$$

where

- True Negatives (TN): Correctly identified non-default customers
- False Negatives (FN): Default customers incorrectly classified as non-default.
- Benefit: Fixed at 1 unit per correctly approved loan.
- Cost: Fixed at 5 units per missed default.

Steps to Construct Profit Curve

(1) Model Training and Probability Prediction: Both the tuned XGBoost model and the Random Forest model are trained using the optimal hyperparameters. The models predict probabilities for the test set, representing the likelihood of each customer defaulting on their loan.

(2) Threshold Iteration: The predicted probabilities are compared against varying thresholds, ranging from 0 to 1. For each threshold, customers are classified into:

  - Non-default (Approved): if the probability $\leq$ threshold.
  - Default (Rejected): if the probability $>$ threshold.

(3) Confusion Matrix Calculation: At each threshold, elements of the confusion matrix are computed:

  - True Positives (TP): Default customers correctly identified.
  - False Positives (FP): Non-default customers incorrectly classified.
  - False Negatives (NP): Default customers misclassified as non-default.
  - True Negatives (TN): Non-default customers correctly classified.

(4) Profit Calculation at Each Threshold: Using the confusion matrix, the profit is calculated as

$$\text{Profit} = (TN \times 1) - (FN \times 5)$$

The profit is computed across all thresholds to analyze the model's financial performance.

(5) Optimal Threshold Identification: The threshold corresponding to the maximum profit is identified as the optimal threshold. This threshold serves as the decision boundary for loan approvals to maximize financial gain. The optimal threshold balances True Positive Ratio (TPR) and Negative Positive Ratio (NPR) to maximize the profit. The formula for TPR and NPR are

$$\text{TPR} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

$$\text{FPR} = \frac{\text{False Positives (FP)}}{\text{False Positives (FP)} + \text{True Negative (TN)}}$$

By identifying this threshold, businesses capture most default customers (high TPR) while controlling for financial losses from misclassified non-default customers (low FPR). The profit curve serves as a quantitative tool to guide this trade-off and align the credit risk model with business objectives.

Profit Curve

Now we visualize the profit curve for our selected model, XGBoost model tuned on ROC AUC, and here are some key observations and business interpretation from the profit curve in Figure 1.

(1) Profit Curve Trends: The profit curve peaks at a specific threshold of approximately 0.16 for the testing set, indicating the optimal threshold for classification. Profit decline after the optimal threshold due to increased false positives or reduced true negatives.

(2) Model Performance: The maximum profit achieved on the test set is $35,513. The ROC-AUC score on the test set is 0.75, suggesting good discriminatory power.

(3) Optimal Threshold: The threshold of 0.16 balances the trade-off between identifying defaults and avoiding unnecessary costs on non-default clients. This low threshold highlights that the cost of missing a default is higher than misclassifying a non-default.

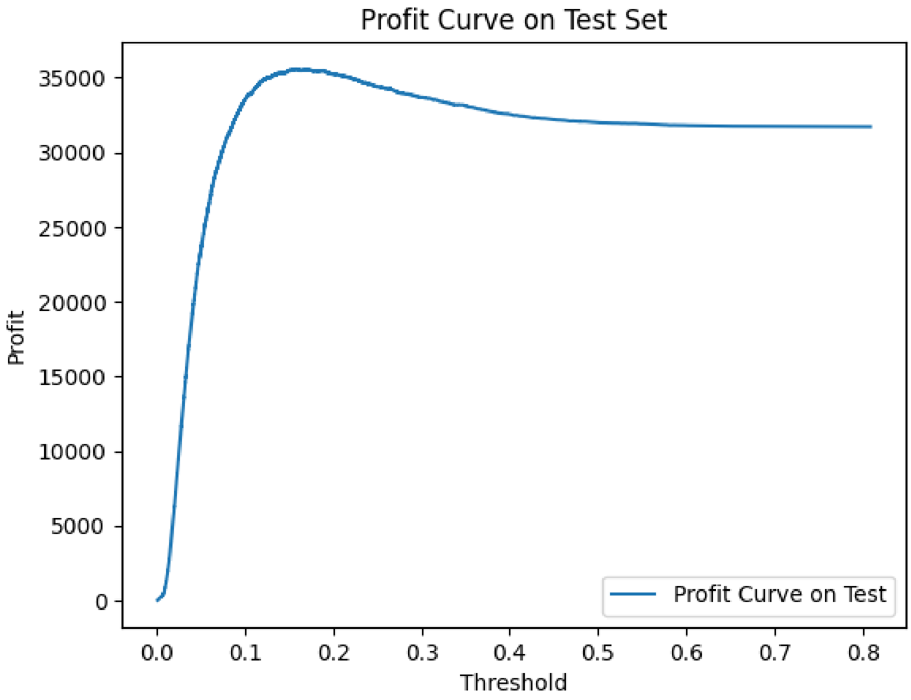

Figure 1: Profit Curve on Test Set Generated by XGBoost Model Tuned on ROC AUC

1.2 Feature Interpretation

1.2.1 Feature Engineering

The feature engineering process involved aggregating and transforming multiple datasets (`bureau`, `credit_card_balance`, `installments_payments`, etc.) to generate domain-specific features relevant to credit risk prediction. These features include repayment status metrics, overdue amounts, credit activity ratios, and delayed payment calculations. By extracting meaningful aggregations such as maximum, average, and total values for key indicators like days past due (DPD), and repayment behavior, the newly constructed features provide a comprehensive understanding of applicant financial behavior, credit usage, and repayment tendencies. This domain knowledge-driven approach ensures the model can distinguish creditworthy applicants from risky borrowers effectively.

1.2.2 Feature Selection

(1) Forward and Backward Selection based on p-value: Forward selection starts with no features and iteratively adds features with p-values below 0.01. Backward elimination removes features with p-values exceeding 0.05. In this step, we identified 35 key features that significantly contribute to the target variable.

(2) Forward and Backward Selection based on AIC/BIC: Akaike Information Criterion (AIC) balances model fit and complexity, selecting 50 features that minimize the trade-off. Bayesian Information Criterion (BIC) favors simpler models, leading to 27 features by imposing a stronger penalty for complexity.

(3) LASSO Regression: We applied LASSO regression with a regularization parameter $\alpha = 0.005$. This method inherently penalizes irrelevant features, shrinking some coefficients to zero. In this step, we retained 5 key features.

(4) PCA Analysis: Individual PCA loading selected 25 features with contribution significance greater than 0.1 to each specific principal component. Total PCA loading retained 18 features with cumulative contributions exceeding 70% of the variance. PCA reduced dimensionality while preserving the most important variance-explaining features.

(5) High Correlation Removal: Pairwise correlation analysis identified features with absolute correlation greater than 0.9, leading to a removal of 3 redundant features. This step reduces multicollinearity, ensuring that remaining features are non-redundant and contribute unique predictive value.

By combining statistical feature selection techniques (p-value, AIC, BIC), LASSO regularization, and dimensionality reduction through PCA, and final set of features is both interpretable and predictive. These methods collectively ensure that only the most significant, non-redundant features are retained, enhancing the model's accuracy, robustness, and efficiency while mitigating risks like overfitting and multicollinearity.

1.2.3 Feature Importance

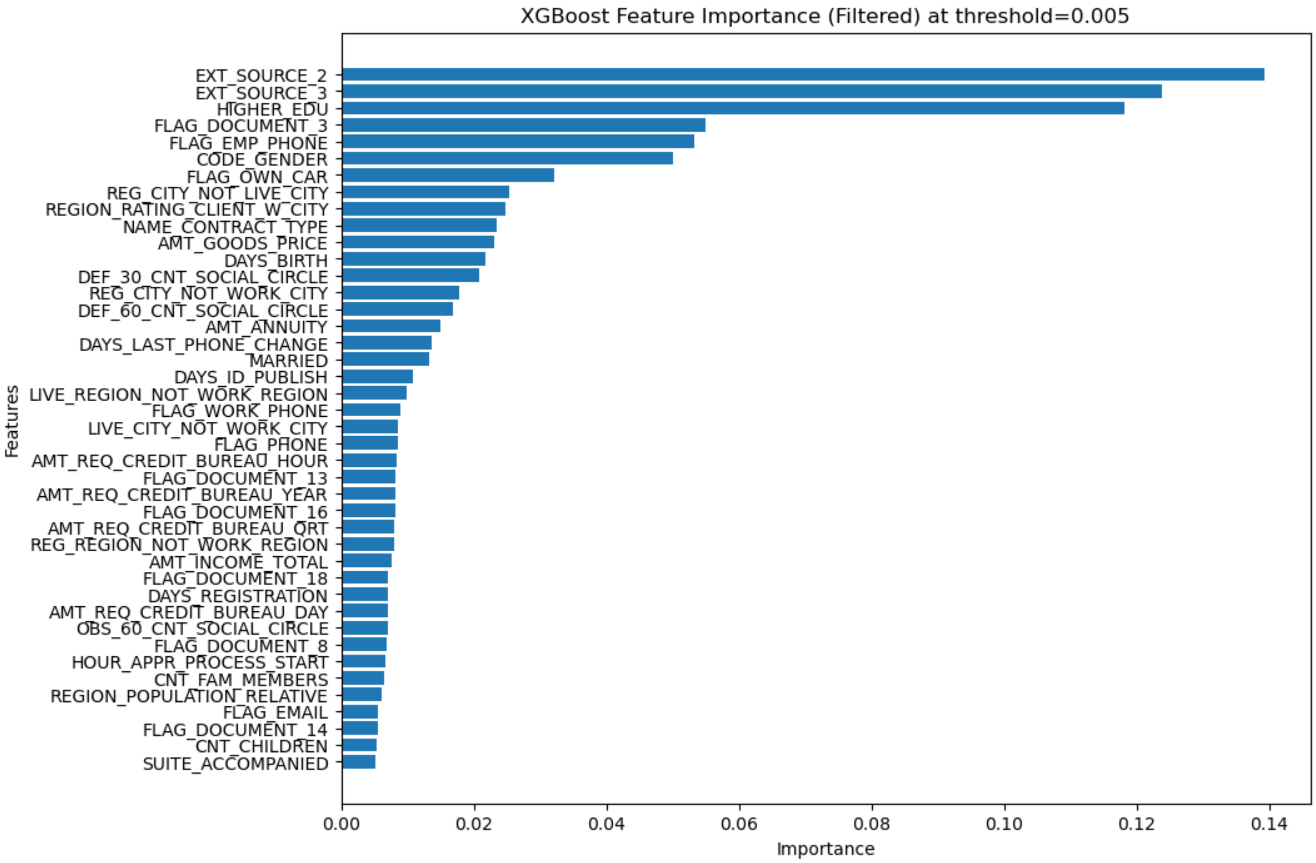

Figure 2: Feature Importance Selected by XGBoost Model Tuned on ROC AUC with Threshold 0.005

According to the important features selected by our chosen XGBoost model tuned on ROC AUC, filtered with a threshold of feature importance greather than 0.005, presented in Figure 2, we can give some intuitive explanations for the most important features.

(1) `EXT_SOURCE_2` (Highest Importance):

Description: Normalized score from an external data source.

Intuitive Reason: External credit scores or normalized scores from trusted credit agencies provide significant predictive power. These scores typically summarize a client's past credit behavior, risk profile, and default probability, making them critical inputs for credit risk modeling.

Reasonable for Credit Risk Modeling: Highly trusted external data sources encapsulate historical repayment patterns, making it a robust predictor for default probability.

(2) `EXT_SOURCE_3` (Second Highest Importance):

Description: Normalized score from an external data source.

Intuitive Reason: Similar to `EXT_SOURCE_2`, another external credit rating provides additional, non-overlapping information about the client's risk level. Multiple external sources help cross-validate creditworthiness.

Reasonable for Credit Risk Modeling: These scores are derived from years of credit behavior and financial activity, providing crucial insights for assessing long-term repayment abilities.

(3) HIGHER_EDU:

Description: Binary flag indicating whether the client has achieved higher education.

Intuitive Reason: Education level is often correlated with income stability and career progression. Clients with higher education tend to have better employment opportunities and, therefore, a lower likelihood of defaulting.

Reasonable for Credit Risk Modeling: Higher education suggests a stronger socioeconomic background and better financial literacy, which can reduce credit risk.

(4) FLAG_DOCUMENT_3:

Description: Flag indicating whether the client provided document 3.

Intuitive Reason: Providing critical identification documents during the loan application process increases the transparency and reliability of the borrower. Missing documents might signal a riskier client.

Reasonable for Credit Risk Modeling: Missing or incomplete documentation is often correlated with higher default risk, as it may reflect issues with identity verification or intent.

(5) FLAG_EMP_PHONE:

Description: Flag indicating if the client provided a work phone number.

Intuitive Reason: Providing a work phone suggests stable employment and verifiability. Clients with stable jobs are generally less likely to default.

Reasonable for Credit Risk Modeling: Employment verification adds credibility and reduces risk uncertainty.

(6) CODE_GENDER:

Description: Gender of the client.

Intuitive Reason: While gender alone does not determine creditworthiness, it can serve as a proxy for financial behavior patterns observed in the data. For example, statistically, one gender might demonstrate lower default rates.

Reasonable for Credit Risk Modeling: Used carefully, demographic variables like gender help capture historical trends in repayment behavior while adhering to ethical and legal standards.

(7) FLAG_OWN_CAR:

Description: Flag indicating if the client owns a car.

Intuitive Reason: Car ownership can indicate financial stability, disposable income, and the ability to manage long-term commitments like loans. It may also correlate with employment status (e.g., having a car facilitates commuting).

Reasonable for Credit Risk Modeling: Owning a car signifies asset ownership, which often translates to higher financial stability.

(8) REGION_RATING_CLIENT_W_CITY:

Description: Rating of the client's region, including city-level information.

Intuitive Reason: Regional ratings capture local economic conditions. Clients from better-rated regions tend to have better economic opportunities and lower unemployment rates, reducing the risk of default.

Reasonable for Credit Risk Modeling: Geographic variables like region rating reflect the likelihood of financial stability and access to opportunities.

(9) **REG_CITY_NOT_LIVE_CITY:**

Description: Flag indicating if the client's registered city differs from the city of residence.

Intuitive Reason: Discrepancies between registered and current cities might indicate instability, frequent relocations, or financial distress. Clients with stable residence histories often demonstrate better repayment behavior.

Reasonable for Credit Risk Modeling: Geographic mobility can sometimes signal higher financial uncertainty or economic distress.

(10) **DEF_30_CNT_SOCIAL_CIRCLE:**

Description: Number of client's social surroundings who defaulted within 30 days past due.

Intuitive Reason: A higher number of defaults in the client's social circle may suggest a riskier environment or community, indirectly impacting the client's ability to manage debt.

Reasonable for Credit Risk Modeling: Peer default rates are strong proxies for external financial influences and behaviors that may affect the applicant.

The selected features are intuitively aligned with credit risk models. They capture a combination of external credit scores, socioeconomic indicators (e.g., education, car ownership), behavioral patterns (e.g., peer defaults), and loan-specific attributes (e.g., goods price). These features are reasonable as they collectively reflect the client's creditworthiness, financial stability, and risk exposure.

2 Individual Technical Write-up

2.1 Model Validation and Selection

In the context of a binary classification problem—predicting whether a customer will default or not—it is paramount to ensure that the chosen model and its hyperparameters are not just empirically good, but also robust and reliable. To achieve this, a systematic model validation process was undertaken, involving advanced techniques such as Grid Search with Cross-Validation (GridSearchCV), multiple evaluation metrics, and custom profit curves. This process aimed not only to maximize predictive performance (in terms of metrics like ROC AUC and F1-score) but also to align the model's predictions with the business objective of maximizing profit or minimizing cost.

2.1.1 Model Validation Approaches

Train-Test Split: The data was initially split into training and test sets. The training set was used for hyperparameter tuning and model fitting, while the test set was held out for final performance evaluation. This ensures that the test set offers an unbiased estimate of the model's generalization capabilities.

Cross-Validation (CV): Within the training set, k-fold cross-validation (with `k=5` for XGBoost and `k=3` for Random Forest in this particular run) was applied. Cross-validation partitions the training data into `k` subsets ("folds"). For each parameter configuration, the model was trained on `k-1` folds and validated on the remaining fold, cycling through all folds. This process reduces variance in the performance estimates and ensures the model's robustness across different subsets of the data.

Grid Search Hyperparameters and Metrics: `GridSearchCV` was employed to systematically explore hyperparameter combinations. For XGBoost, parameters like `n_estimators`, `learning_rate`, `max_depth`, `colsample_bytree`, `subsample`, and `scale_pos_weight` were tuned. For Random Forest, parameters such as `n_estimators`, `max_depth`, `min_samples_split`, `min_samples_leaf`, and `max_features` were varied.

XGBoost Hyperparameters:

- **`n_estimators`:** The number of boosting rounds or trees the model builds. Increasing this generally allows the model to fit more complex functions, but may increase the risk of overfitting if set too large.
- **`learning_rate`:** A shrinkage parameter that scales the contribution of each newly added tree. A smaller learning rate typically leads to better generalization but may require more estimators.
- **`max_depth`:** The maximum depth of each decision tree. Deeper trees can model more complex relationships but may overfit if set too high.
- **`colsample_bytree`:** The fraction of features (columns) randomly selected for each tree. This helps reduce overfitting by introducing randomness in the feature space.
- **`subsample`:** The fraction of the training data (rows) sampled for each tree. Similar to `colsample_bytree`, this adds randomness and can help prevent overfitting.
- **`scale_pos_weight`:** A parameter to balance the positive and negative classes. This is particularly useful for imbalanced classification tasks, as it adjusts the weight of the positive class to improve the model's sensitivity to minority classes.

Random Forest Hyperparameters:

- **`n_estimators`:** The number of decision trees in the forest. More trees generally improve performance but increase computation time. A sufficiently large number usually leads to stable estimates.
- **`max_depth`:** The maximum depth of each tree. Limiting the depth can help prevent overfitting, ensuring that each tree does not become too complex.
- **`min_samples_split`:** The minimum number of samples required to split an internal node. Larger values lead to less complex trees, as splits must be made on larger subsets, potentially reducing overfitting.
- **`min_samples_leaf`:** The minimum number of samples required to be at a leaf node. Setting this to a larger value creates smoother decision boundaries and can help reduce overfitting.
- **`max_features`:** The number of features to consider when looking for the best split at each node. Restricting this introduces randomness and helps break correlations among features, improving the model's ability to generalize. Common options include using all features, the square root of the total number of features, or the logarithm of the total number of features.

By exhaustively searching through predefined grids, we compared models trained under different configurations. Two primary metrics were used for model selection:

- **ROC AUC (Receiver Operating Characteristic Area Under the Curve):** This metric measures the model's ability to distinguish between classes across various thresholds. A higher ROC AUC indicates better separability and is robust to class imbalance.
- **F1-score:** The harmonic mean of precision and recall, suitable when we care about both identifying the positive class correctly and minimizing false positives. Useful in more imbalanced or cost-sensitive scenarios.

The models were tuned once using ROC AUC as the target metric and then again using F1-score to see which metric led to better alignment with the business goals.

Profit Curve Analysis and Threshold Optimization: Beyond ROC AUC and F1, a profit curve was constructed to integrate business rules into the model evaluation. This curve uses the model's predicted probabilities and assigns a monetary value (benefit or cost) to correct and incorrect classifications. By varying the threshold at which we classify an instance as "default," we identified the threshold that maximized profit. The logic:

- Benefit for correctly identifying a non-default (true negative).
- Cost for missing a default (false negative).

2.1.2 Best Model and Hyperparameter Selection

Four key configurations were evaluated: XGBoost and Random Forest, each tuned separately for ROC AUC and F1-score. By examining the ROC AUC, maximum profit, and the detailed classification reports on the test set, we aimed to choose the model that best balanced strong discrimination, practical financial gain, and robust handling of the positive (default) class.

Evaluation Metrics

(1) ROC AUC Score: Quantifies a binary classifier's ability to distinguish between positive and negative classes at all possible thresholds, with higher values indicating better discrimination and 1.0 representing a perfect classifier.

(2) Maximum Profit: Quantifies the net financial gain or loss derived from applying the model's predictions at an optimal threshold, ensuring alignment with business objectives.

(3) Classification Report: Provides precision, recall, F1-score, and support for non-default and default classes.

Model Outputs

(1) Tuned XGBoost (ROC AUC):

- Tuned XGBoost ROC AUC on Test Set: 0.75
- Maximum Profit on Test Set: 35,513

Class	Precision	Recall	F1-score	Support
non-default	0.94	0.90	0.92	56538
default	0.25	0.38	0.30	4965
Accuracy	0.86 (61503 samples)			
Macro Avg.	0.60	0.64	0.61	61503
Weighted Avg.	0.89	0.86	0.87	61503

Table 2: Classification Report for Tuned XGBoost (ROC AUC)

(2) Tuned XGBoost (F1-score):

- Tuned XGBoost ROC AUC on Test Set: 0.75
- Maximum Profit on Test Set: 35,421

Class	Precision	Recall	F1-score	Support
non-default	0.94	0.90	0.92	56538
default	0.25	0.37	0.30	4965
Accuracy	0.86 (61503 samples)			
Macro Avg.	0.60	0.64	0.61	61503
Weighted Avg.	0.89	0.86	0.87	61503

Table 3: Classification Report for Tuned XGBoost (F1-score)

(3) Tuned Random Forest (ROC AUC):

- Tuned Random Forest ROC AUC on Test Set: 0.74
- Maximum Profit on Test Set: 34,666

Class	Precision	Recall	F1-score	Support
non-default	0.94	0.89	0.91	56538
default	0.23	0.38	0.28	4965
Accuracy	0.85 (61503 samples)			
Macro Avg.	0.58	0.63	0.60	61503
Weighted Avg.	0.88	0.85	0.86	61503

Table 4: Classification Report for Tuned Random Forest (ROC AUC)

(4) Tuned Random Forest (F1-score):

- Tuned Random Forest ROC AUC on Test Set: 0.73
- Maximum Profit on Test Set: 34,459

Class	Precision	Recall	F1-score	Support
non-default	0.94	0.88	0.91	56538
default	0.22	0.37	0.28	4965
Accuracy	0.84 (61503 samples)			
Macro Avg.	0.58	0.63	0.59	61503
Weighted Avg.	0.88	0.84	0.86	61503

Table 5: Classification Report for Tuned Random Forest (F1-score)

Model Comparison

Table 6 presents a summary of the tuned model outputs for the convenience of comparison.

Model	ROC AUC	Max Profit	Accuracy	Non-Default (Majority)			Default (Minority)		
				Prec.	Rec.	F1	Prec.	Rec.	F1
XGBoost (ROC AUC)	0.75	35,513	0.86	0.94	0.90	0.92	0.25	0.39	0.30
XGBoost (F1)	0.75	35,421	0.86	0.94	0.90	0.92	0.25	0.37	0.30
RF (ROC AUC)	0.74	34,666	0.85	0.94	0.89	0.91	0.23	0.38	0.28
RF (F1)	0.73	34,459	0.84	0.94	0.88	0.91	0.22	0.37	0.28

Table 6: Comparison of tuned XGBoost and Random Forest models, optimized under different criteria. The *Score* column reflects the metric used for tuning (ROC AUC or F1). Metrics are reported from the test set.

(1) ROC AUC Score: The two XGBoost models both achieve a ROC AUC score of 0.75, outperforming the Random Forest models' ROC AUC scores.

(2) Maximum Profit: The XGBoost model optimized for ROC AUC reaches the highest maximum profit among all four models, which is 35,513.

(3) Accuracy: The two XGBoost models both have a high overall accuracy of 0.86, outperforming the Random Forest models' overall accuracy.

(4) Default Class Precision: For the XGBoost models, the default class achieve a precision of 0.25. Although this indicates that only a quarter of the instances predicted as default are actually default, it still outperforms the Random Forest models.

(5) Default Class Recall: The recall of the XGBoost model tuned based on ROC AUC stands at 0.39, which is the highest among the four models.

(6) Default Class F1-Score: Combining precision and recall yields an F1-score of 0.30, reflecting a competitive balance between precision and recall for the default class among the four models.

Among the four configurations, XGBoost (tuned for ROC AUC) maintains a high ROC AUC (0.75), achieves the highest maximum profit (35,513), high accuracy (0.86), and shows competitive precision and recall for the default class. These combined strengths suggest that the XGBoost model, with parameters selected to maximize ROC AUC, offers the best overall balance of predictive performance and financial return.

2.1.3 Performance Testing on ROC Curve

The Receiver Operating Characteristic (ROC) curve for the test set demonstrates the trade-off between the True Positive Rate (TPR) and False Positive Rate (FPR). The area under the curve (AUC) is 0.75, suggesting the model has a good ability to distinguish between default and non-default clients. A diagonal baseline representing a random guess (AUC = 0.50) further highlights the model's superior performance.

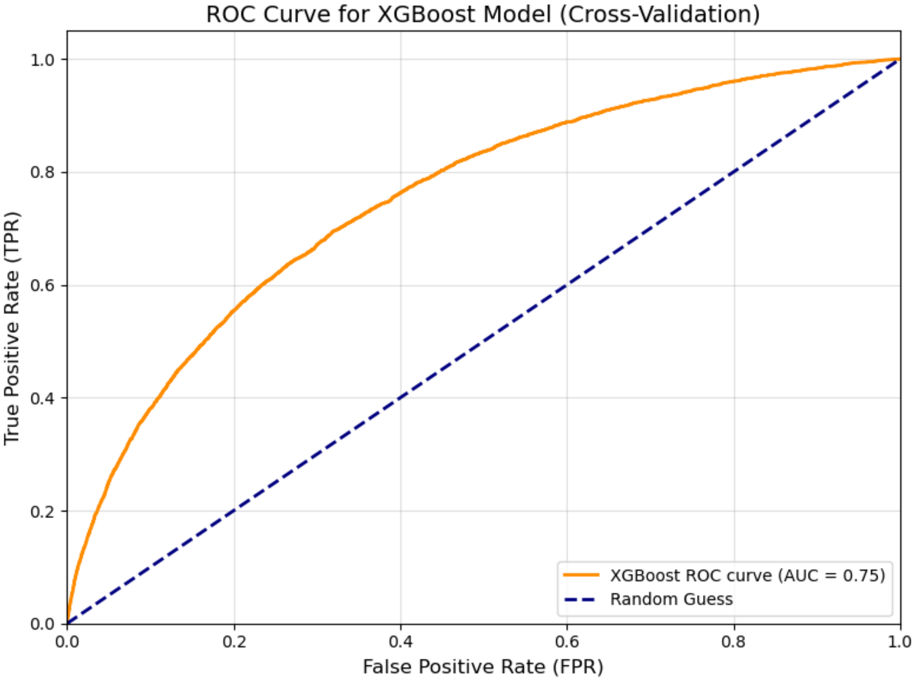

Figure 3: ROC Curve for XGBoost Model Tuned on ROC AUC

2.1.4 Reliability in Production

The XGBoost model tuned on ROC AUC has demonstrated strong reliability and robustness during the model validation and testing phases. The following points outline why this model can be trusted in a production environment:

(1) **Robust Performance Across Metrics:** The XGBoost model achieves a ROC AUC of 0.75 on the test set, indicating good separability between default and non-default clients. Its accuracy of 86% confirms that the model maintains a high overall prediction quality. Furthermore, the maximum profit of $35,513 reinforces the model's alignment with business objectives, ensuring profitability when deployed.

(2) **Balanced Handling of Imbalanced Classes:** The model incorporates the `scale_pos_weight` hyperparameter, which adjusts the weight of the minority (default) class. This ensures that the model does not overlook high-risk clients despite the imbalance in class distribution. The default class recall of 0.39 is particularly noteworthy, as identifying more defaults is critical for minimizing financial risk.

(3) **Cross-Validation and Generalization:** The use of 5-fold cross-validation during hyperparameter tuning ensures that the model is robust across different subsets of the training data. This process minimizes the risk of overfitting and provides confidence that the model's performance will generalize well to unseen data.

(4) **Optimal Threshold for Business Objectives:** Profit curve analysis identified the optimal threshold for classification, directly integrating business priorities into the modeling process. By balancing the cost of missed defaults (false negatives) with the benefit of correctly identified non-defaults (true negatives), the model maximizes financial returns and minimizes potential losses.

(5) **Feature Importance Analysis:** The model relies on intuitive and meaningful features, such as external credit scores (`EXT_SOURCE_2`, `EXT_SOURCE_3`), education level (`HIGHER_EDU`), and financial behavior indicators (e.g., `FLAG_EMP_PHONE`, `FLAG_DOCUMENT_3`). These features are strongly tied to credit risk and demonstrate that the model makes decisions based on economically and socially relevant attributes.

(6) **Interpretability and Auditability:** The feature importance rankings and intuitive explanations of the top features ensure that the model's predictions are interpretable and transparent. This is critical for maintaining trust with stakeholders, regulatory compliance, and identifying opportunities for model improvement.

(7) **Scalability and Efficiency:** XGBoost is highly efficient and scalable, making it suitable for processing large volumes of data in production environments. Its ability to handle missing values and work effectively with sparse datasets ensures reliability under real-world conditions.

(8) **Consistent Validation Across Models:** The comparison with other models, such as Random Forest, highlights the consistent superiority of the XGBoost model in terms of ROC AUC, profit, and accuracy. This further confirms its robustness and reliability for production deployment.

(9) **Business Alignment:** The model's direct optimization for profit ensures that its predictions align with the organization's financial goals. By reducing defaults while maintaining high approval rates for non-default clients, the model supports both risk management and revenue generation.

(10) **Monitoring and Retraining Strategy:** A post-deployment monitoring and retraining strategy ensures that the model remains reliable over time. This involves regular evaluation of model performance on new data, retraining when necessary, and addressing any drift in data patterns or feature distributions.

Overall, the XGBoost model offers a robust, interpretable, and efficient solution for credit risk assessment. Its alignment with business objectives and proven performance across various evaluation metrics make it a trustworthy candidate for production deployment.

2.2 Possible Improvements Provided Given More Time

Given additional time and resources, the model's performance could be further improved through several enhancements to data preprocessing, feature engineering, and model training. These improvements include more advanced techniques for handling missing values, better treatments for class imbalance, and refinements in model evaluation and deployment strategies.

(1) **Advanced Missing Value Treatment:**

- **Imputation Based on Data Relationships:** Instead of relying on simple imputation techniques (e.g., mean, median, or mode), advanced methods such as K-Nearest Neighbors (KNN) imputation or multivariate imputation using chained equations (MICE) could be employed. These methods consider the relationships between variables, leading to more accurate estimates of missing values.
- **Domain-Specific Imputation:** For certain variables (e.g., financial or demographic attributes), domain-specific rules could guide the imputation. For example, missing values in income-related features could be estimated based on employment type or education level.
- **Indicator Variables for Missingness:** Introducing binary indicator variables to flag missing values could allow the model to capture patterns in missingness itself, which might be predictive of default behavior (e.g., clients with missing income might have higher default risk).

(2) **Better Treatment of Class Imbalance:**

- **Over-Sampling the Minority Class:** Techniques like SMOTE (Synthetic Minority Over-sampling Technique) could be used to generate synthetic samples for the minority (default) class, improving the model's ability to learn patterns in this underrepresented group.
- **Under-Sampling the Majority Class:** Randomly under-sampling the majority (non-default) class could balance the dataset. To mitigate information loss, this could be combined with ensemble techniques like EasyEnsemble, where multiple under-sampled datasets are trained on.
- **Cost-Sensitive Learning:** Incorporating cost-sensitive learning directly into the model by assigning higher penalties to misclassified defaults (false negatives) could improve the model's focus on the minority class.
- **Adaptive Boosting (AdaBoost):** Implementing adaptive boosting methods could help the model focus on misclassified samples, including those from the minority class.
- **Class-Balanced Loss Functions:** Using advanced loss functions like focal loss or class-balanced loss can dynamically adjust the weights of the minority class during training, making the model more sensitive to minority-class patterns.

(3) **Enhanced Feature Engineering:**

- **Interaction Features:** Creating interaction terms between key features (e.g., income and education level or loan amount and region rating) could improve the model's ability to capture non-linear relationships.
- **Temporal Features:** Additional temporal features, such as the time elapsed between loan application and approval or recent changes in financial behavior, could improve predictions.
- **External Data Integration:** Incorporating external data sources, such as macroeconomic indicators or regional employment rates, could provide additional context for assessing credit risk.

(4) **Alternative Models and Ensemble Techniques:**

- **Deep Learning Models:** Neural networks, particularly those designed for tabular data (e.g., TabNet or TensorFlow Decision Forests), could be explored for capturing more complex relationships.
- **Model Ensembling:** Combining XGBoost with other models like LightGBM, Random Forest, or logistic regression through stacking or blending could yield improved predictive performance.

(5) **Improved Evaluation Strategies:**

- **Long-Term Validation:** Evaluating the model's performance on data from different time periods (e.g., out-of-sample data from later years) could assess its robustness to changes in economic conditions or client profiles.
- **Profit-Driven Metrics:** Additional business-focused metrics, such as lifetime customer value or cost of false negatives, could provide better alignment with business goals.

(6) **Automated Hyperparameter Tuning:**

- Instead of grid search, Bayesian optimization or genetic algorithms could be used to find the best hyperparameter configurations more efficiently and effectively.

(7) **Post-Deployment Monitoring and Retraining:**

- **Data Drift Monitoring:** Monitoring for shifts in data distributions over time (data drift) could ensure the model remains effective as client profiles and economic conditions change.

- **Dynamic Retraining:** Regularly retraining the model with the latest data, incorporating feedback from its performance in production, could sustain high performance.

These improvements, while requiring additional development time and computational resources, would enhance the robustness, interpretability, and predictive power of the credit risk model, ensuring that it aligns closely with both operational needs and business objectives.

3 Next Steps

3.1 2-Year Plan

3.1.1 Technical Side

(1) Data Collection and Enrichment:

- Expand data sources by integrating macroeconomic data, credit bureau updates, and behavioral data (e.g., transaction history, payment patterns).
- Collect data on rejected applications to improve the model's ability to understand default risk across the entire applicant pool, not just approved loans.
- Incorporate real-time data streams, such as employment verification or regional economic indicators, to enhance model timeliness and relevance.

(2) Advanced Techniques:

- Implement deep learning methods like TabNet or neural networks tailored for tabular data to capture complex non-linear relationships in credit behavior.
- Explore transfer learning by leveraging pre-trained models on similar datasets to reduce training time and improve model accuracy.
- Introduce reinforcement learning frameworks to optimize credit policies dynamically, based on historical repayment behaviors and evolving customer profiles.

(3) Explainability and Fairness:

- Deploy tools like SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) to make the model interpretable for stakeholders and regulators.
- Conduct fairness audits to ensure the model is unbiased toward protected attributes (e.g., gender, race, age), ensuring compliance with regulatory standards.

(4) Infrastructure Upgrades:

- Build an automated machine learning pipeline for data preprocessing, feature engineering, hyperparameter tuning, and retraining, reducing manual intervention.
- Implement cloud-based solutions for scalable model deployment and real-time inference.
- Develop an API framework to integrate the model seamlessly with business systems, enabling faster decision-making.

3.1.2 Business Side

(1) Product Development:

- Introduce personalized credit offerings by integrating model predictions with marketing strategies, such as pre-approved loan limits for low-risk customers.
- Develop a risk-adjusted pricing framework where interest rates are dynamically assigned based on predicted default risk.
- Launch real-time credit scoring products for retail partners to enhance decision-making at the point of sale.

(2) Customer Segmentation:

- Use model insights to segment customers into risk tiers, targeting low-risk customers with premium offerings and high-risk customers with debt restructuring programs.
- Develop early intervention programs for at-risk customers based on predicted repayment difficulties, improving retention and reducing default rates.

(3) Regulatory and Compliance Strategy:

- Collaborate with regulatory bodies to ensure the model adheres to compliance standards, especially in data privacy and algorithmic fairness.
- Maintain transparent documentation for all model decisions, providing audit trails to regulators and business stakeholders.

3.2 Model Monitoring

3.2.1 Change in Input Features

- Regularly monitor the distributions of input features for signs of data drift (e.g., shifts in customer demographics or economic conditions).
- Set up automated alerts for significant changes in critical features like `EXT_SOURCE_2` or `AMT_GOODS_PRICE`, which are highly predictive in the model.
- Update and retrain the model periodically to incorporate new patterns in the data or respond to changes in customer behavior and macroeconomic conditions.

3.2.2 Statistical Model Performance

- Track key performance metrics such as ROC AUC, F1-score, and precision/recall for the default class on a rolling basis to detect any degradation in model accuracy.
- Use back-testing techniques to evaluate model predictions against actual outcomes for loans issued in the past quarter.
- Conduct stress tests by simulating adverse scenarios (e.g., economic downturns) to evaluate model robustness under extreme conditions.

3.2.3 Business KPI

- Monitor the impact of model predictions on financial KPIs such as default rates, loan approval rates, and net profit.
- Track customer retention and satisfaction metrics, ensuring that risk-based decisions do not adversely affect customer experience.
- Evaluate the profitability of credit portfolios segmented by risk tier, ensuring that the model aligns with overall business strategy and financial goals.
- Develop dashboards for senior management to provide real-time insights into the model's business impact and operational performance.

By combining technical excellence, business strategy, and rigorous monitoring, this two-year plan ensures the credit risk model remains a valuable and reliable tool for decision-making, driving both operational efficiency and financial growth.