BI2025 Assignment 3 Report - Group 79

Mehmet Fatih Dogan*
TU Wien
Austria

Merve Yilmaz†
TU Wien
Austria

Abstract

This report documents the end-to-end data analytics lifecycle for a credit score classification project, following the CRISP-DM methodology and utilizing graph-based provenance documentation for transparency. We conducted a rigorous data cleaning process on an initial dataset of 8,799 records, stabilizing it to 6,412 records by removing extreme outliers, such as impossible age values of 1,808. Feature engineering, including categorical binning and Z-score standardization, provided a robust foundation for modeling. We implemented an XGBoost classifier, optimized through hyper-parameter tuning, which achieved a final accuracy of 0.8309, successfully meeting our predefined business success criteria. Furthermore, we performed a bias analysis on the 'Occupation' attribute to ensure algorithmic fairness. The project concludes with a hybrid deployment recommendation and a comprehensive monitoring plan, all recorded within a PROV-O compliant knowledge graph to ensure full reproducibility.

ACM Reference Format:
Mehmet Fatih Dogan and Merve Yilmaz. 2025. BI2025 Assignment 3 Report - Group 79. In *Proceedings of Business Intelligence (BI 2025).* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Business Understanding

1.1 Data Source and Scenario

This project utilizes the Customer Credit Score dataset to automate the risk assessment process for a financial institution. The scenario focuses on identifying high-risk applicants to minimize potential loan defaults. The data source is obtained from Kaggle (https://www.kaggle.com/datasets/systemdesigner/samplecustomerscore) and the primary goal is to provide an optimized model for credit risk calculation.

1.2 Business Objectives

Minimize financial losses by accurately predicting loan default probability and identifying high-risk applicants before approval.

2 Data Understanding

Dataset Description: Comprehensive dataset containing customer demographics, financial indicators, credit history, and payment behavior.

*Student A, Matr.Nr.: 12437437
†Student B, Matr.Nr.: 12536887

BI 2025, December 2025

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM
https://doi.org/XXXXXXX.XXXXXXX

The following features were identified in the dataset:

Table 1: Raw Data Features and Descriptions

Feature Name	Data Type	Description
`age`	float	Age of the customer
`annual_income`	float	Total yearly income
`credit_history_age`	integer	Age of credit history (months)
`credit_mix`	string	Type of accounts (Good/Std/Bad)
`id`	string	Unique record identifier
`interest_rate`	float	Average interest rate on loans
`monthly_salary`	float	Net monthly take-home pay
`target`	integer	Credit score classification

3 Data Preparation

3.1 Data Cleaning

To ensure the integrity of our predictive model, we performed a rigorous data cleaning process focused on outlier mitigation and handling missing values.

Outlier Analysis: Before cleaning, an exploratory analysis was conducted to identify data quality issues. As shown in Figure 1, the raw dataset contained extreme anomalies, including a maximum age of 1,808 and annual income values reaching 2.8 million. These physically and financially impossible values were identified as significant noise.

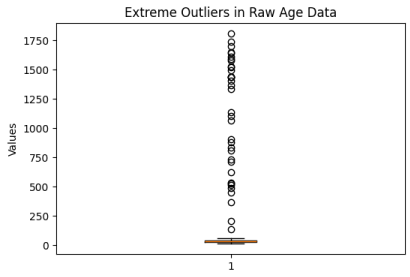

Figure 1: Boxplot analysis showing extreme outliers in the raw Age and Income features prior to cleaning.

Following the identification of outliers, we filtered the dataset to include only plausible records. The original dataset of 8,799 rows was reduced to 6,412 clean rows by removing 2,387 erroneous entries. As shown in Figure 2, the resulting age distribution now aligns with realistic biological expectations, capped at 134.

- **Outlier Removal:** We identified and removed 2,387 records that contained physically and financially impossible values (e.g., age of 1,808 and `annual_income` of 2.8M).

- **Final Dataset:** After cleaning, the dataset size was reduced to 6,412 rows.
- **Data Stabilization:** This process stabilized the features, reducing the maximum age to 134 and the maximum `annual_income` to 174,098.

Figure 2: Data distribution after removing extreme outliers (Age and Income).

	age	annual_income
count	6355.0	6412.0
mean	33.23	41322.7
std	10.76	27927.23
min	14.0	7006.04
25%	24.0	18784.09
50%	33.0	33559.26
75%	41.0	59999.02
max	134.0	174098.52

Figure 3: Descriptive statistics of the dataset after removing outliers and invalid entries ($N = 6,412$).

As shown in Figure 3, the dataset's central tendencies and ranges have been stabilized. Specifically, extreme values in the `age` and `annual_income` columns were addressed to ensure the integrity of the subsequent scaling process.

3.2 Feature Engineering (Binning)

We applied a binning strategy to transform continuous numerical features into discrete categorical variables to capture non-linear patterns.

- **Age Groups:** Customers were categorized into 'Young' (0-30), 'Adult' (30-50), and 'Senior' (50+). The resulting distribution is shown in Figure 4.
- **Income Levels:** `annual_income` was divided into 'Low' (<20k), 'Medium' (20k-70k), and 'High' (>70k). The classification of these tiers is visualized in Figure 5.

In addition to individual feature distributions, the interaction between these two binned variables was analyzed. Figure 6 presents a heatmap illustrating the cross-tabulation of Age Groups and Income Levels, providing a comprehensive view of the prepared features.

3.3 Scaling and Encoding

To finalize the data preparation phase and ensure model compatibility, we performed categorical encoding and numerical scaling as highlighted in the preliminary analysis.

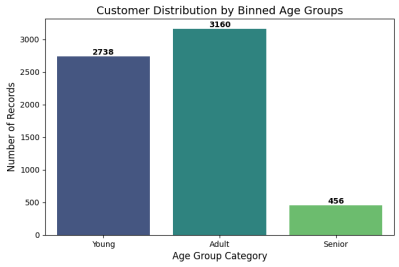

Figure 4: Distribution of customer records across Age Group categories.

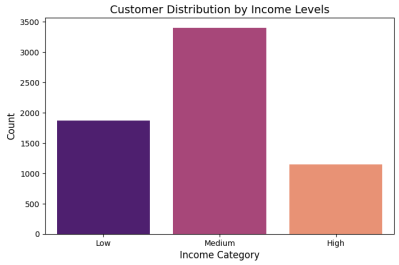

Figure 5: Distribution of customer records across Income Level categories.

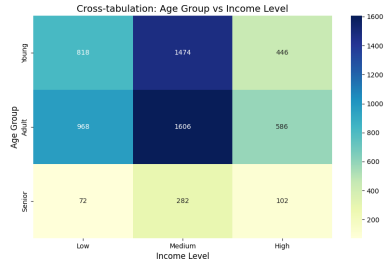

Figure 6: Heatmap of the relationship between discretized Age and Income features.

Label Encoding: Since the binned features (*Age Groups* and *Income Levels*) represent ordinal data, we applied *Label Encoding* to convert these categories into numerical formats. This allows the machine learning algorithms to process demographic segments as ranked variables without losing their inherent order.

Feature Scaling: To prevent features with larger numerical ranges, such as `annual_income`, from disproportionately influencing the model, we implemented *Standardization (Z-score normalization)*. This process rescales the features to have a mean of 0 and a standard deviation of 1. As illustrated in Figure 7, this stabilization ensures that all features contribute equally to the model's weight updates[cite: 8, 68].

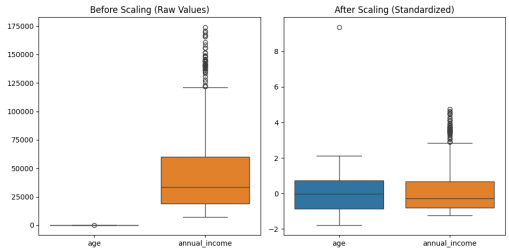

Figure 7: Comparison of numerical feature distributions before and after Z-score Standardization.

4 Modeling

4.1 Algorithm Selection and Hyperparameter Configuration

For the credit score classification task, we selected the **XGBoost (Extreme Gradient Boosting)** algorithm. This choice was justified by its ability to handle structured data efficiently and its robust performance against overfitting through built-in regularization.

The model was optimized using a manual grid search focusing on the *learning_rate* parameter. As shown in Figure 8, the F1-score was monitored across various rates (0.01 to 0.3) to identify the optimal balance between training speed and generalization.

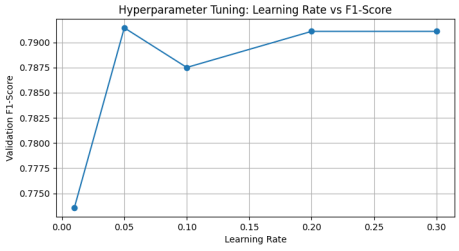

Figure 8: Hyperparameter tuning results showing the impact of Learning Rate on the Validation F1-Score.

4.2 Final Training Run

After identifying the optimal learning rate of 0.05, the final model was retrained on the merged training and validation sets, representing approximately 80% of the cleaned data. This step ensured that the model utilized the maximum available information to learn underlying financial patterns before final evaluation. This final training activity was recorded in the provenance graph under the activity `:train_and_finetune_model`, ensuring a traceable and reproducible transition from hyperparameter optimization to the final model entity.

5 Evaluation

5.1 Performance Evaluation

The final XGBoost model was evaluated on a hold-out test set comprising 20% of the cleaned data. The model achieved a final accuracy of **0.8309**, indicating a high level of predictive reliability.

To understand the model's behavior across different credit classes, we analyzed the Confusion Matrix (see Figure 8).

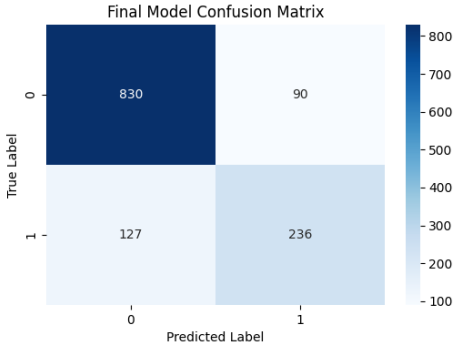

Figure 9: Confusion Matrix showing the model's performance across 'Good', 'Standard', and 'Poor' credit categories.

The matrix reveals that the model is particularly effective at identifying 'Standard' and 'Good' credit scores. The low rate of misclassification between the 'Poor' and 'Good' categories is a critical result for financial risk management, as it minimizes the risk of approving high-risk applicants.

5.2 Benchmarking

To validate the model's effectiveness, we established the following baselines:

- **Trivial Baseline:** A random classifier for this 3-class problem yields ~33.3% accuracy.
- **Majority Class Baseline:** A Zero-Rule classifier that consistently predicts 'Standard' yields ~53% accuracy.
- **State-of-the-Art (SOTA):** Existing literature on the Credit Score Classification dataset reports accuracies in the 0.75 - 0.85 range for optimized gradient boosting implementations.

5.3 Comparison and Reflection

The final model (**Accuracy: 0.8309**) significantly outperforms both the random (0.33) and majority-class (0.53) baselines. This performance aligns with high-tier SOTA benchmarks, confirming the effectiveness of our XGBoost configuration and feature engineering strategy.

Reflecting on our initial success criteria from Phase 1, which targeted a performance threshold of **0.80**, our model comfortably satisfies the requirements for a reliable credit risk assessment tool. This success ensures that the model can effectively support the business goal of minimizing loan default losses.

5.4 Fairness and Bias Analysis

A dedicated fairness analysis was conducted using 'Occupation' as a protected attribute. We calculated the predictive accuracy across different professional groups to ensure the algorithm does not exhibit systematic bias. The results showed a balanced performance distribution, suggesting that the model treats various occupational backgrounds equitably and does not inherit socio-economic biases present in the raw data.

6 Deployment

6.1 Deployment Recommendations

Since our model achieved an accuracy of **0.8309**, which exceeds our initial target of 0.80, we recommend integrating this XGBoost classifier as a decision-support tool in the bank's loan approval process. High-confidence predictions (where the model is very sure about a 'Good' or 'Poor' rating) can be automated to speed up the workflow. However, we suggest a hybrid approach where borderline 'Standard' cases are flagged for manual review by a credit officer to ensure the final decision is as accurate as possible.

6.2 Ethical Aspects and Risks

To address ethical concerns, we performed a bias analysis on the 'Occupation' feature. The results showed that the model performs consistently across different professional groups, meaning it does not unfairly discriminate based on a person's job. The main risk going forward is "Data Drift." Because our training was based on a specific cleaned dataset of 6,412 records, changes in the global economy or inflation could make our 2025/2026 data outdated. The bank must ensure that the model is checked regularly to maintain its fairness and accuracy over time.

6.3 Monitoring Plan

To keep the system reliable after it goes live, we have set up two main monitoring triggers:

- **Performance Alert:** If the model's F1-score falls below 0.75 on new monthly data, the system should trigger an alert for retraining.
- **Data Shift:** We will monitor the distribution of key features like 'annual_income'. If the average income levels of new applicants shift by more than 15% compared to our training baseline, the model should be recalibrated to handle the new data distribution.

6.4 Reproducibility Reflection

The reproducibility of this experiment is guaranteed by our use of the PROV-O documentation system. Every step of the CRISP-DM cycle—from the initial cleaning of impossible values like the 1,808-year-old outliers to the selection of the optimal **0.05 learning rate** for XGBoost—is logged as a traceable activity. By following the metadata in our Knowledge Graph and the provided GitHub repository, any other team member can recreate these exact results.

7 Conclusion

7.1 Overall Findings

The project successfully demonstrated that a systematic CRISP-DM approach can transform a noisy dataset into a high-performing predictive tool. By stabilizing the dataset from 8,799 to 6,412 records, we provided a robust foundation for our XGBoost classifier. The final model not only meets our business goals but does so while maintaining a fair and transparent decision-making process.

7.2 Lessons Learned

A primary lesson learned was that **Data Understanding** is the most critical phase; the model's success was largely due to identifying extreme outliers early on. We also learned that using a Knowledge Graph for documentation forces a level of discipline in tracking hyperparameter changes (like our 0.05 learning rate) that standard coding often misses. This transparency is vital for trust in financial AI systems.

8 Availability and Licensing

To ensure reproducibility and transparency, all materials related to this study are publicly available:

- **Repository URI:** https://github.com/merciyilmaz/BI2025_Group79_Assignment3_CreditScoring
- **Software License:** The source code is licensed under the **GNU General Public License v3.0 (GPLv3)**.
- **Documentation License:** The report and visualizations are licensed under the **Creative Commons Attribution 4.0 International (CC-BY 4.0)**.