CAPSTONE DEFENSE

PROJECT TITLE

EVALUATING MACHINE LEARNING ALGORITHMS TO BUILD WEB-BASED PREDICTIVE LOAN APPROVAL SYSTEM BASED ON CREDIT SCORE

PRESENTED BY: ROHAN PARKAR

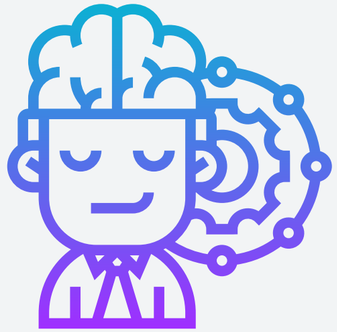
CONTENT

INTRODUCTION

- Loan lending process is **essential** in the **financial sector**, balancing eligibility with **minimizing default risk**.
- Historically, the process relied on **subjective judgments**, **personal interviews**, and the **experiences of credit officers**.
- The **credit scoring** method developed between **1950 and 1960** became the **cornerstone** of modern lending, offering a systematic evaluation of creditworthiness.
- The introduction of credit scores **revolutionized** loan assessments.

MOTIVATION

- Traditional loan approval methods heavily relied on **personal judgment**, which could introduce **biases** and inconsistencies.
- The credit scores alone **cannot capture** the **full complexity** of an applicant's financial situation and risk profile.
- **Machine learning** offers a more **elaborate** and **reliable** approach to assessing loan applications by analyzing complex patterns and relationships in data.
- **Machine learning** has the potential to significantly **improve** the **accuracy** and **efficiency** of loan approval processes, thereby reducing the likelihood of defaults.



PROJECT OBJECTIVES

Evaluate and Compare Algorithms

Assess the **performance** of five machine learning algorithms (***K-Nearest Neighbors, Logistic Regression, Support Vector Machine, Extreme gradient Boost, Multiple Layer Perceptron***) for loan approval prediction.

Develop Web Application

Create a user-friendly **web application** that **integrates** the **best-performing model** for real-time loan approval decisions.

Enhance Loan Approval System

Incorporate machine learning techniques to improve upon conventional credit score-based methods, delivering a more **accurate**, **consistent**, and **scalable** solution for financial institutions.

LITERATURE SURVEY

- More **comprehensive datasets**, including economic indicators, are needed to enhance model accuracy.
- Proper **feature selection** and **hyperparameter** tuning are essential to improve model generalization and prevent overfitting.
- Techniques like **under-sampling, over-sampling** and **class weights balancing** are crucial for creating more balanced models.
- A **broader comparison of algorithms** with larger, more diverse datasets is needed for more reliable loan approval predictions.

DATASET OVERVIEW

- **Source**: Dataset from **Kaggle**, focusing on loan applications and associated financial data.
- **Total Data Points**: **55,106** records.
- **Features**: **16 features**, including **12 numerical** and **4 categorical** variables.
- **Target Variable**: **GoodLoan**, a binary classification indicating whether a loan is **approved** (1) or **rejected** (0).

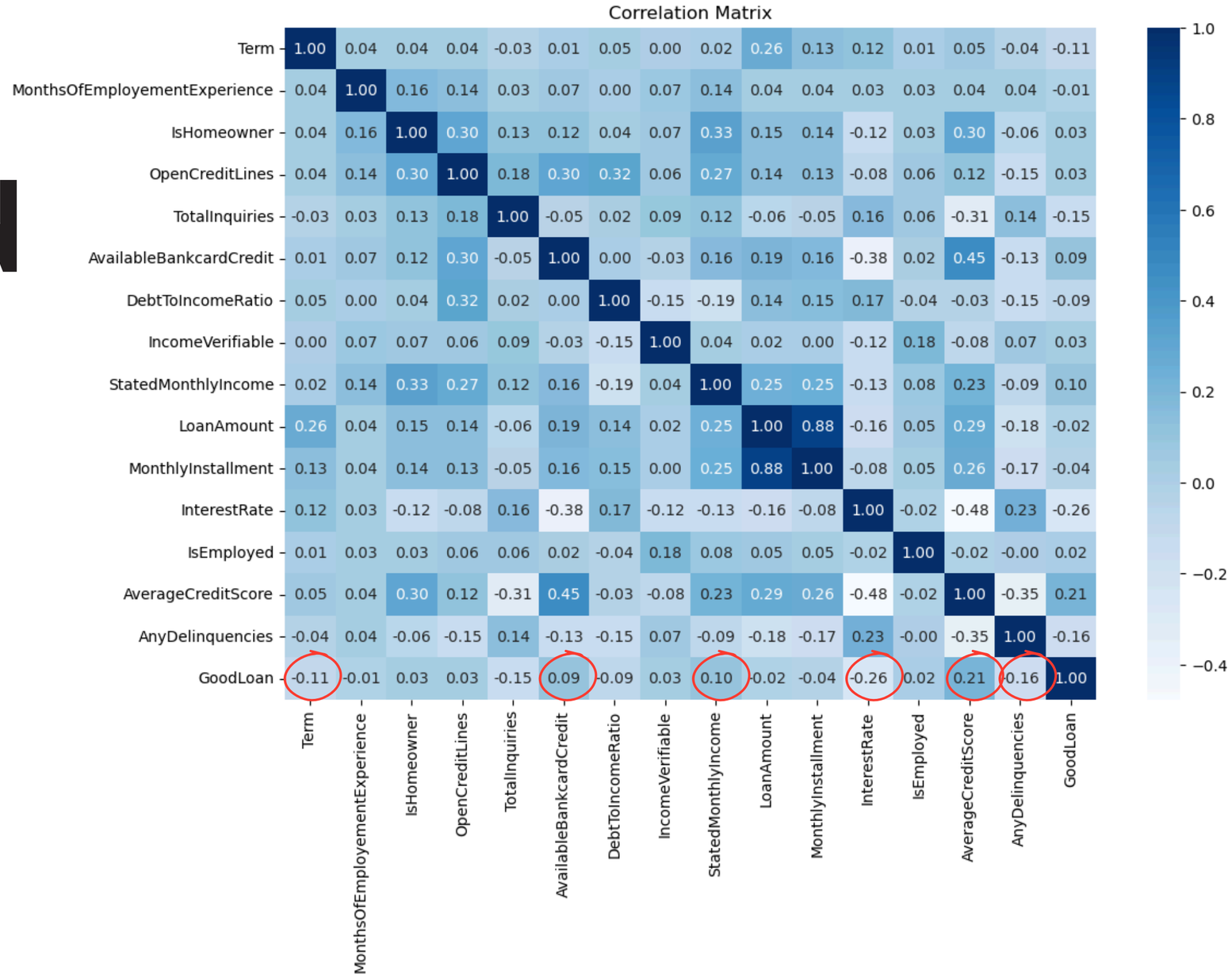
DATASET OVERVIEW:
FEATURE LIST

Feature	Type	Description
Term	Numerical	Loan term in months.
MonthsOfEmployementExperience	Numerical	Number of months of employment experience.
IsHomeowner	Categorical	Whether the applicant is a homeowner. (1: Yes, 0: No)
OpenCreditLines	Numerical	Number of open credit lines.
TotalInquiries	Numerical	Total number of credit inquiries.
AvailableBankcardCredit	Numerical	Available bankcard credit.
DebtToIncomeRatio	Numerical	Debt to income ratio.
IncomeVerifiable	Categorical	Whether the income is verifiable (1: Yes, 0: No).
StatedMonthlyIncome	Numerical	Stated the monthly income of the applicant.
LoanNumber	Numerical	Unique identifier for each loan.
LoanAmount	Numerical	Amount of the loan.
MonthlyInstallment	Numerical	Monthly installment amount.
InterestRate	Numerical	Interest rate of the loan.
IsEmployed	Categorical	Whether the applicant is currently employed. (1: Yes, 0: No)
AverageCreditScore	Numerical	Average credit score of the applicant.
AnyDelinquencies	Categorical	Whether there are any delinquencies. (1: Yes, 0: No)
GoodLoan	Categorical	Whether the loan is considered good. (1: Good, 0: Bad)

DATA EXPLORATION

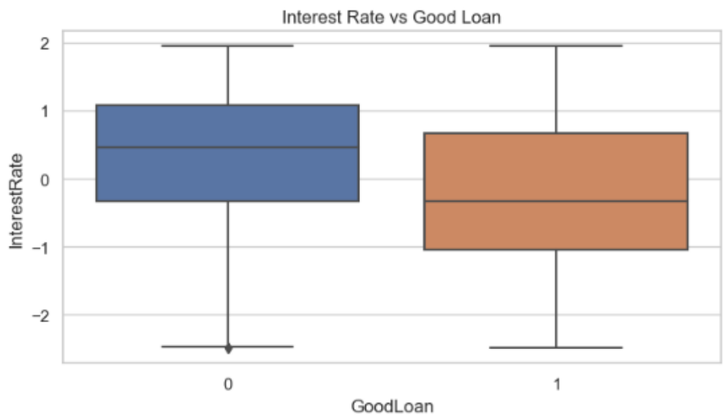
Key Features:

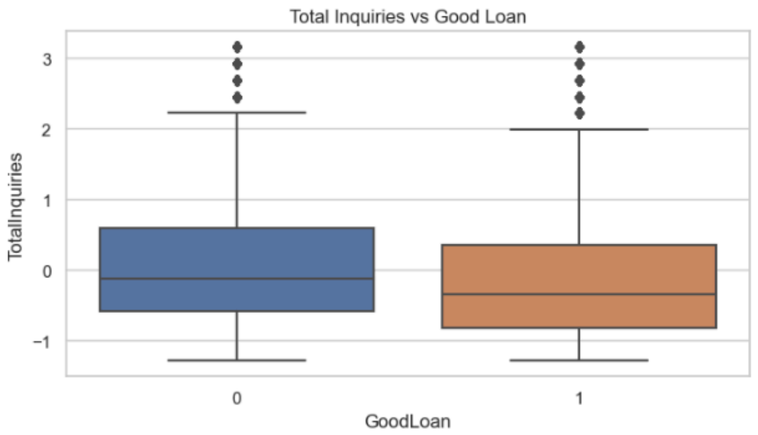
- InterestRate
- AverageCreditScore
- AnyDelinquencies
- TotalInquiries
- Term
- StatedMonthlyIncome

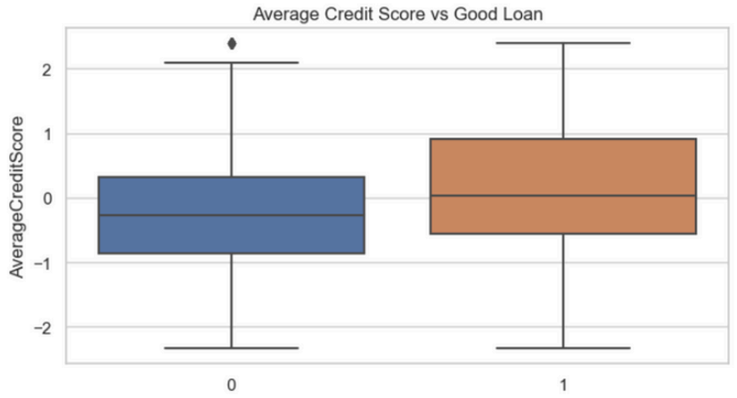


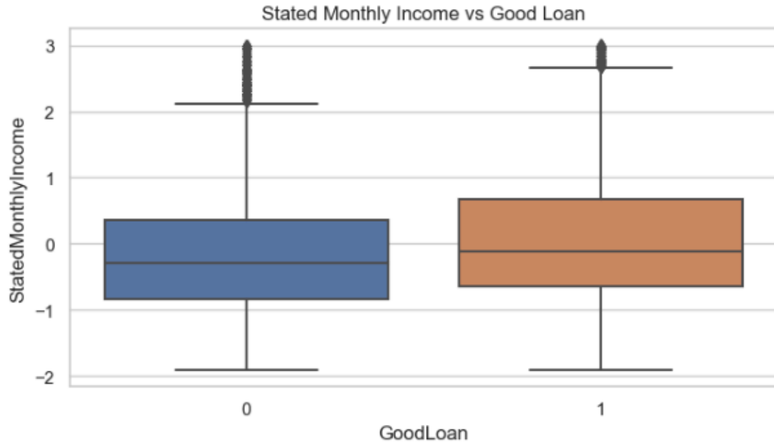
DATA EXPLORATION

During the exploration phase, several variables **showed notable correlation** with GoodLoan (Target Variable).



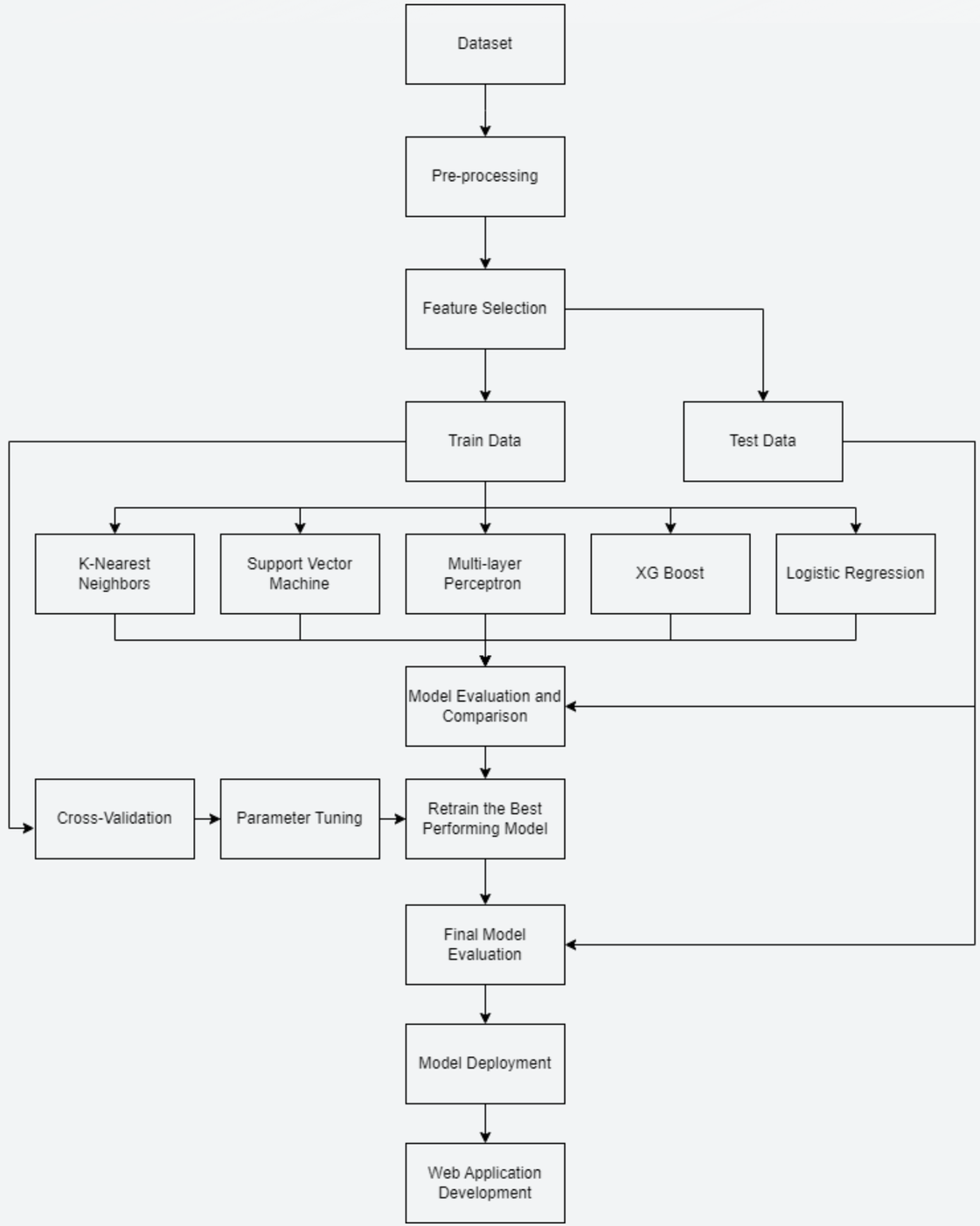






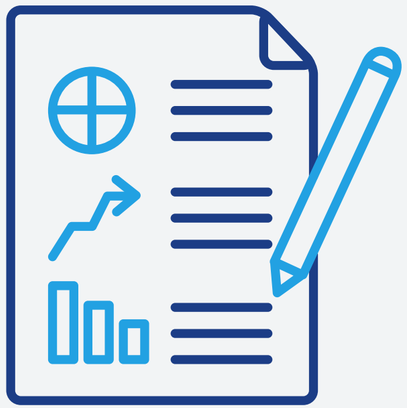
PROJECT WORKFLOW

- **Data Preprocessing:** Cleaning and preparing the dataset.
- **Feature Selection:** Selecting key features using ANOVA F-value scoring.
- **Model Implementation:** Training and tuning models.
- **Model Evaluation:** Assessing performance with metrics like accuracy and ROC-AUC.
- **Deployment:** Integrating the best model into a web app for real-time predictions.



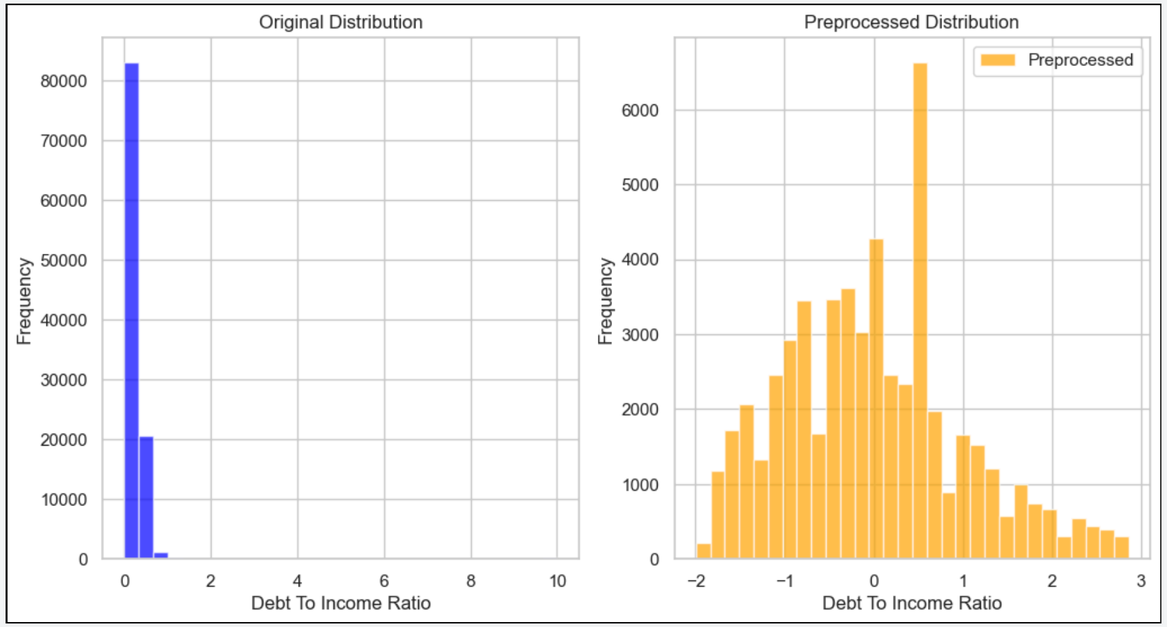
PRE-PROCESSING

- **Renaming** Columns.
- **Dropping** Unnecessary Columns.
- Handling **Outliers**.
- Handling **Missing** Values.
- **Standardizing** Numerical Features.
- **Encoding** categorical values using Label Encoding
- Converting to appropriate **data types**.



PRE-PROCESSING EFFECTS

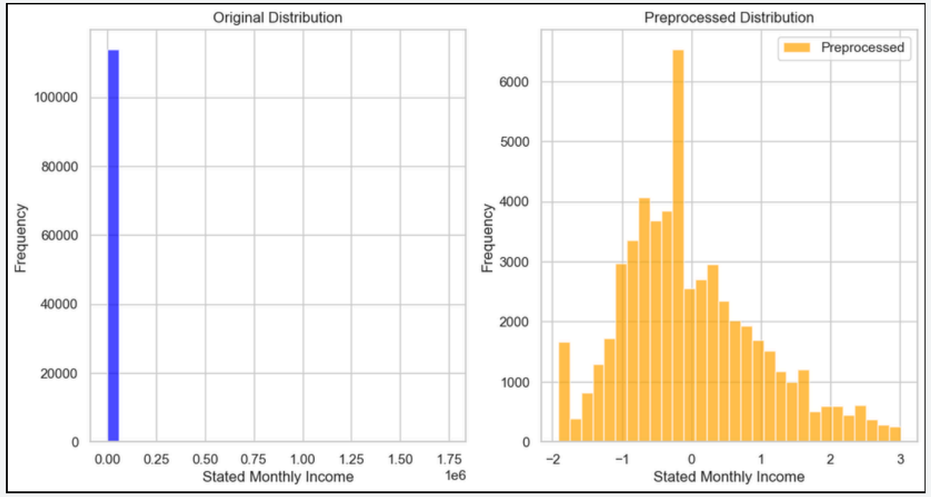
- **Before Pre-processing:** Highly **right-skewed**, indicating a concentration of lower debt-to-income ratios with some outliers.
- **After Pre-processing**: Less skewed, with a more **even distribution** (Normal Distribution).

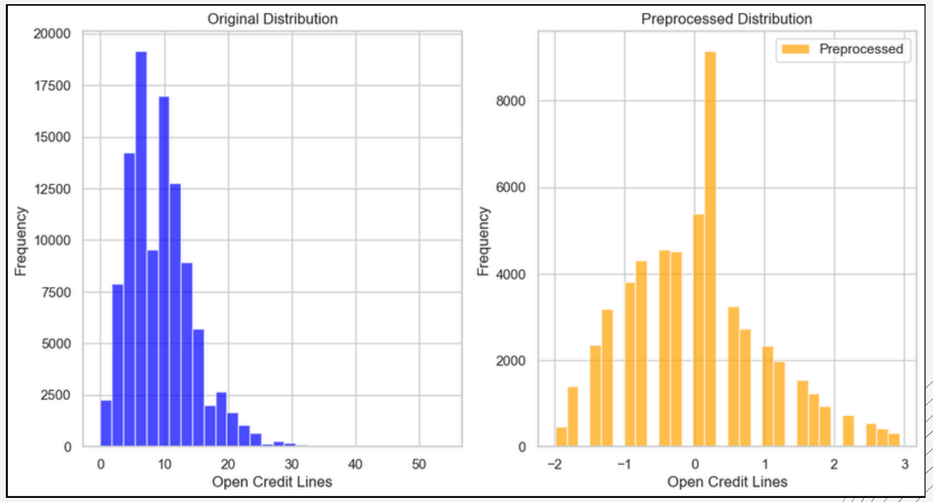


PRE-PROCESSING EFFECTS

- Similarly, other features with **skewed distributions** also improved using the pre-processing techniques.

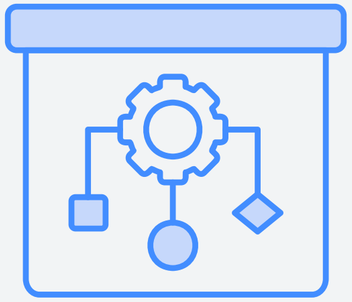
- Pre-processing **reduced skewness** and standardized the scale, enhancing model accuracy and training efficiency.





IMPLEMENTATION OF ALGORITHMS

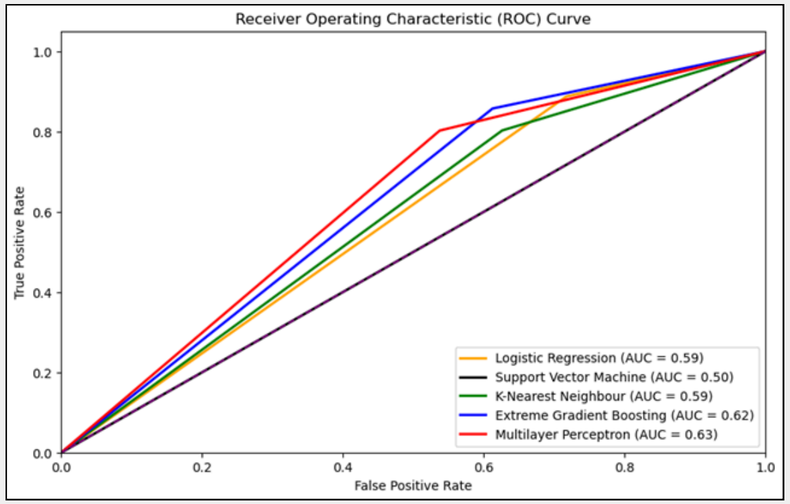
- The dataset was initially split into a **training** set (70%) and a **testing** set (30%).
- The training set was divided into **training** (75%) and **validation** (25%) subsets.
- **ANOVA F-test** identified the top **12 relevant features** for model training.
- Algorithms (Logistic Regression, SVM, KNN, XGBoost, MLP) were **trained** using these features.

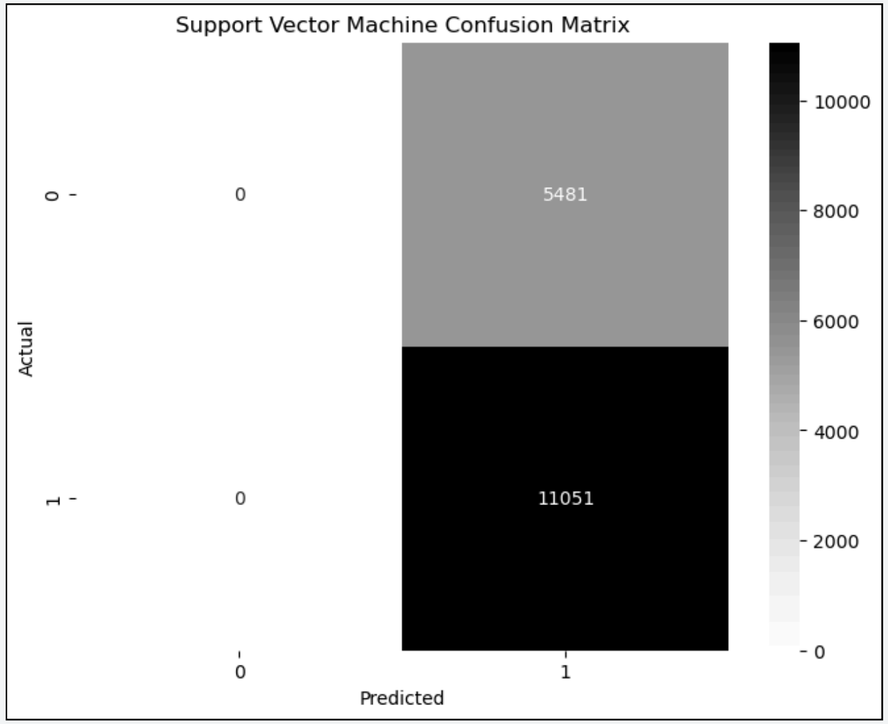


MODEL DEVELOPMENT ITERATION 1

- Models were trained with **default** settings.
- Base models served as a **baseline** for **comparison**.
- **Strong accuracy** for **Extreme Gradient Boosting.**
- Difficulty in classifying the **minority class.**

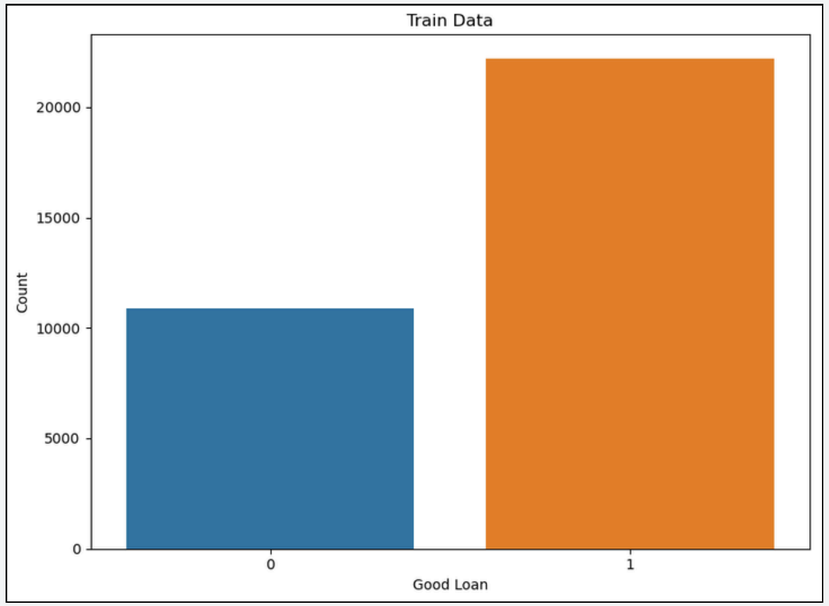
Model	Accuracy	ROC-AUC	F1-Score
Logistic Regression	0.686910	0.585068	0.653066
Support Vector Machine	0.668461	0.500000	0.535632
K-Nearest Neighbour	0.660477	0.588286	0.647727
Extreme Gradient Boosting	0.701549	0.622455	0.683769
Multilayer Perceptron	0.689693	0.632485	0.683292

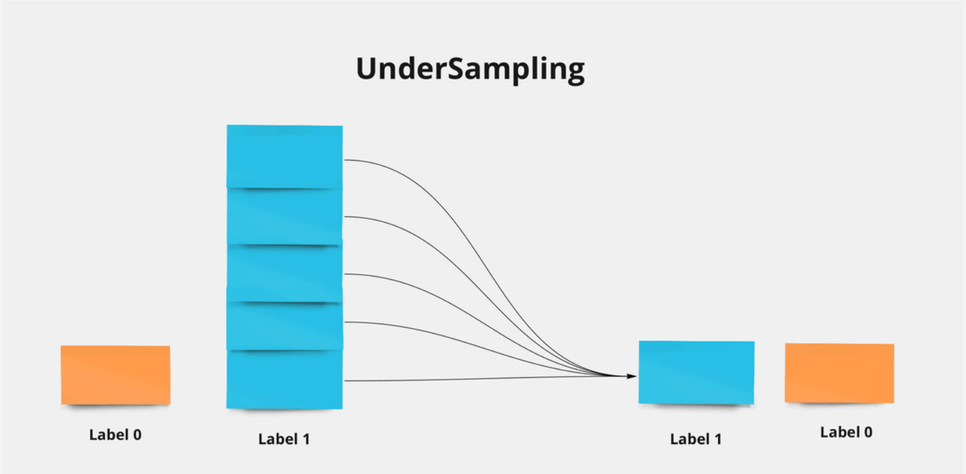




MODEL DEVELOPMENT ITERATION 2

- **Issue:** Poor model sensitivity towards the minority class (**class imbalance**).
- **Techniques:**
  - **Class weighting:** Assigns higher penalties for misclassifying minority class instances.
  - **Under-sampling:** Reduces the number of samples in the majority class.

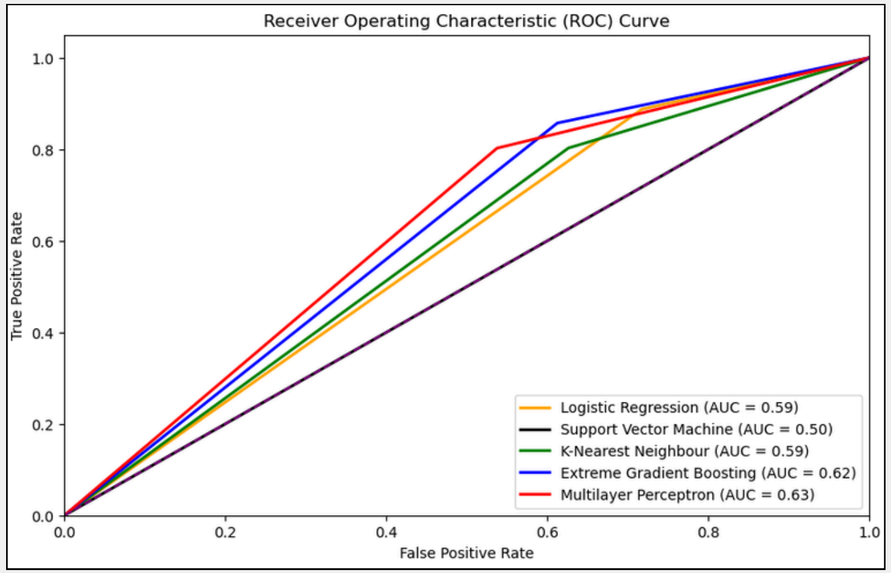




MODEL DEVELOPMENT ITERATION 2

Impact:

- **Increased ROC-AUC** scores for the models.
- **Improved** handling of the **minority** class.
- **Decreased** overall **accuracy** (expected trade-off).



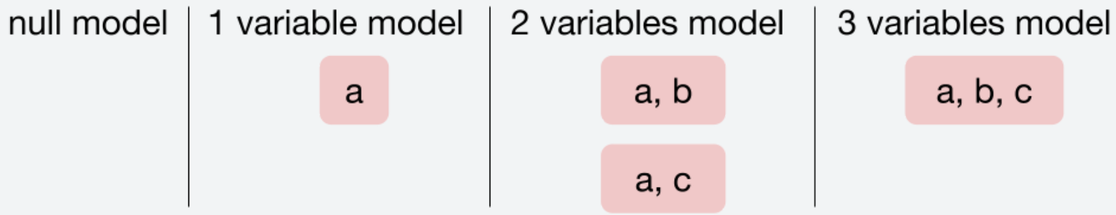
Model	Accuracy	ROC-AUC	F1-Score
Logistic Regression	0.686910	0.585068	0.653066
Support Vector Machine	0.668461	0.500000	0.535632
K-Nearest Neighbour	0.660477	0.588286	0.647727
Extreme Gradient Boosting	0.701549	0.622455	0.683769
Multilayer Perceptron	0.689693	0.632485	0.683292

Model	Accuracy	ROC-AUC	F1-Score
Logistic Regression	0.577486	0.641112	0.581043
Support Vector Machine	0.609418	0.640421	0.619262
K-Nearest Neighbour	0.604609	0.606686	0.615418
Extreme Gradient Boosting	0.658541	0.655485	0.667411
Multilayer Perceptron	0.616585	0.651983	0.625839

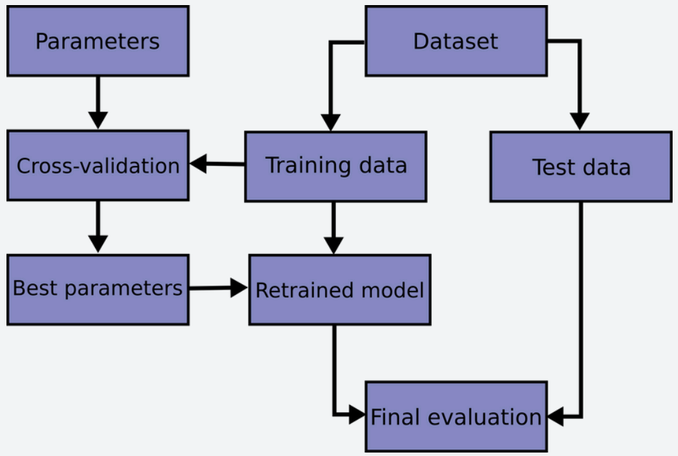
MODEL DEVELOPMENT ITERATION 3

- **Issue:** Improve model performance.
- **Techniques:**
  - **Feature selection**: Employed **forward stepwise selection** to identify features that maximized the ROC AUC score, optimizing model efficacy.
  - **Hyperparameter tuning**: Conducted hyperparameter tuning using **Grid Search CV** to refine model parameters.

Forward Stepwise Selection



Hyper Parameter Tuning



MODEL DEVELOPMENT ITERATION 3

- **Tuning Example:** XG Boost Model
- **Parameters Tuned:**
  - **n_estimators:** Number of trees (boosting rounds).
  - **learning_rate:** Step size to update weights during each boosting round.
  - **max_depth:** Controls the depth of each tree.
- **Best Result:**
  - learning_rate: **0.1**
  - max_dept: **4**
  - n_estimators: **100**

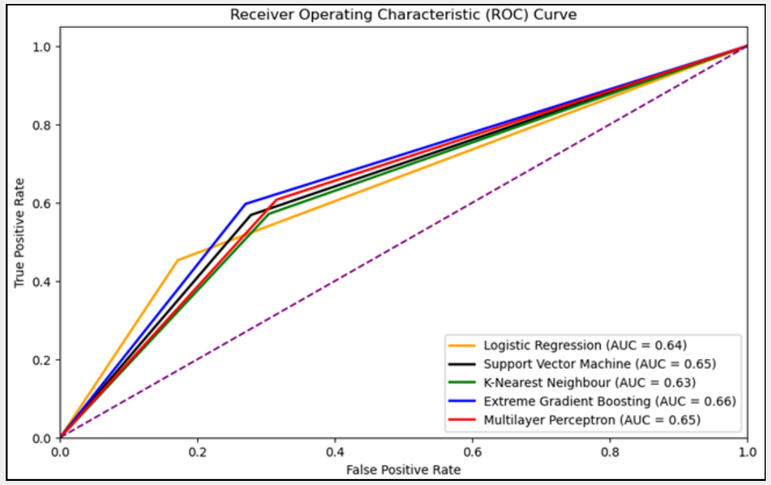
XG Boost Parameter Grid

```
param_grid = {
'n_estimators': [100, 200, 300],
'learning_rate': [0.05, 0.1, 0.2],
 'max_depth': [3, 4, 5],
}
```

MODEL DEVELOPMENT ITERATION 3

Impact:

- Improved model **stability** and **performance**.
- Increased **F1 scores** for most models.
- Reduced model complexity, potentially **lowering** the risk of **overfitting**.
- More **balanced model** performance, as reflected in ROC-AUC scores.



Model	Accuracy	ROC-AUC	F1-Score
Logistic Regression	0.577486	0.641112	0.581043
Support Vector Machine	0.609418	0.640421	0.619262
K-Nearest Neighbour	0.604609	0.606686	0.615418
Extreme Gradient Boosting	0.658541	0.655485	0.667411
Multilayer Perceptron	0.616585	0.651983	0.625839

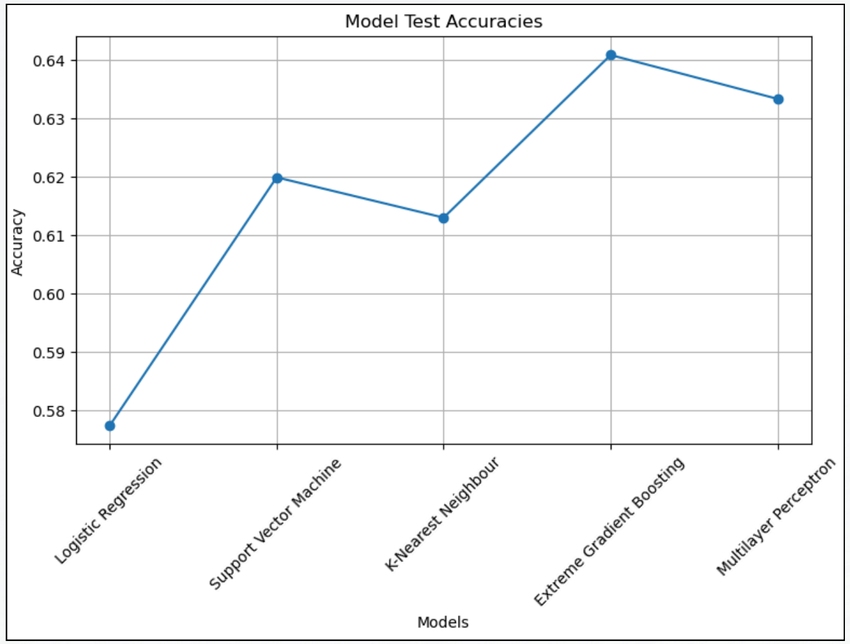
Model	Accuracy	ROC-AUC	F1-Score
Logistic Regression	0.577426	0.640791	0.581063
Support Vector Machine	0.619942	0.645385	0.630119
K-Nearest Neighbour	0.613047	0.633693	0.623644
Extreme Gradient Boosting	0.640878	0.663424	0.651274
Multilayer Perceptron	0.633370	0.646228	0.643601

FINAL MODEL EVALUATION

- **Best Model:** Extreme Gradient Boosting

- **Key Metrics:**
  - ROC-AUC score: **0.6634**.
  - Accuracy: **64.09%**.
  - F1-score: **0.6512**.

- **XG Boost's Advantage:**
  - Superior performance in **handling imbalanced datasets**.
  - **Robustness** in learning complex patterns.
  - **Balanced performance** in terms of precision and recall.



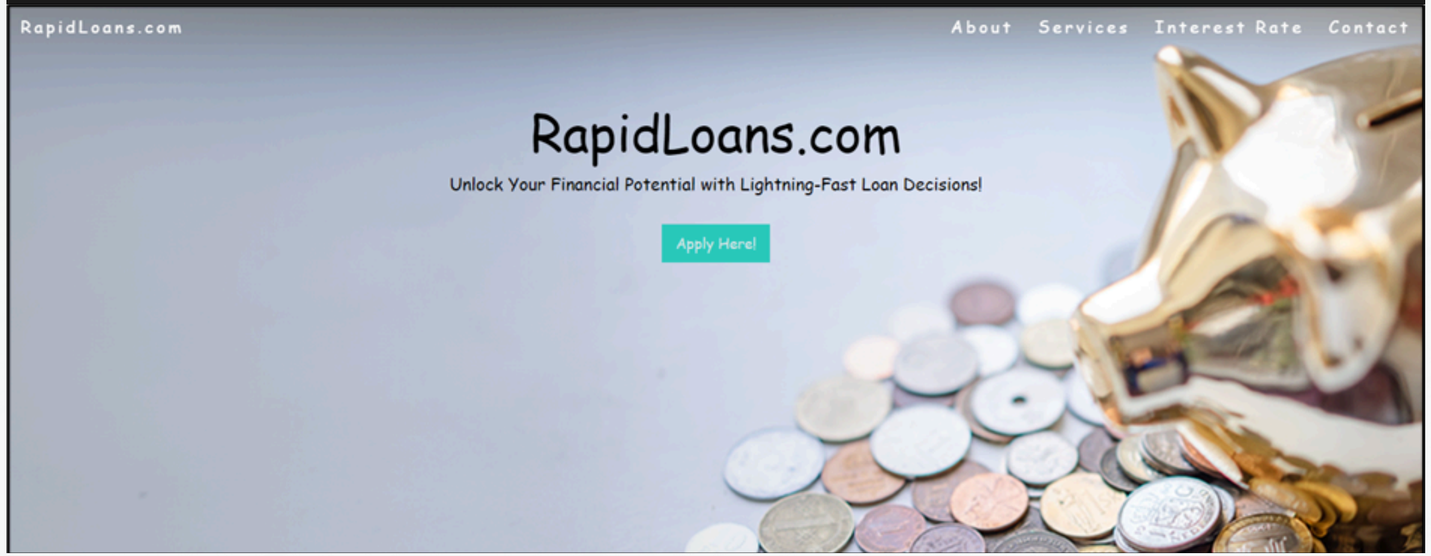
Model	Accuracy	ROC-AUC	F1-Score
Logistic Regression	0.577426	0.640791	0.581063
Support Vector Machine	0.619942	0.645385	0.630119
K-Nearest Neighbour	0.613047	0.633693	0.623644
Extreme Gradient Boosting	0.640878	0.663424	0.651274
Multilayer Perceptron	0.633370	0.646228	0.643601

WEB APPLICATION

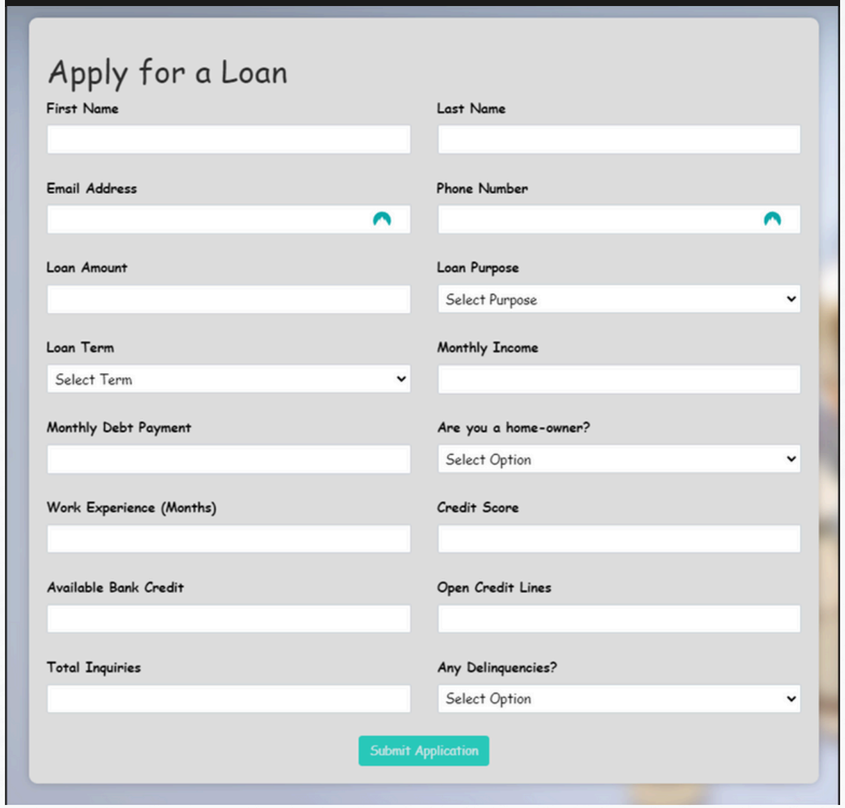
- **Integration:** The XGBoost model was integrated into a web application.
- **Front-end:** HTML, CSS, and JavaScript for a user-friendly interface.
- **Back-end:** Flask framework for simplicity, flexibility, and Python compatibility.
- **Loan Application Form**: Users input financial and personal information.
- **Initial Validation**: Checks for completeness and basic criteria.
- **Predictive Model Integration:** Passes valid inputs to XGBoost for analysis.
- **Decision Output**: Real-time loan approval or rejection.

WEB APPLICATION

Home Page



Application Form



WEB APPLICATION

Loan Approved

Loan Application Result

Thank you, Clark . Your application has been processed.

Congratulations! Your loan has been approved.

Monthly Installment: $664.29

Go to Home Page

Loan Rejected

Loan Application Result

Thank you, Peter . Your application has been processed.

Unfortunately, your loan application was not approved.

Go to Home Page

LOAN APPROVAL EXAMPLE

- Loan Amount: **$20000**
- Interest Rate: **12%**
- Monthly Installment: **$664.29**
- Average Credit Score: **750**
- Term: **36 months**
- Any Delinquencies: **0**
- Is Homeowner: **Yes**
- Debt-to-Income Ratio: **0.10**
- Stated Monthly Income: **$5000**
- Open Credit Lines: **6**
- Decision: **Approved**

Reasoning:

- Strong credit profile (low interest rate, high credit score)
- Reasonable monthly payment compared to income
- Low debt-to-income ratio
- No missed payments
- Homeowner status

LOAN DECLINED EXAMPLE

- Loan Amount: **$80000**
- Interest Rate: **15%**
- Monthly Installment: **$2,773.23**
- Average Credit Score: **650**
- Term: **36 months**
- Any Delinquencies: **0**
- Is Homeowner: **No**
- Debt-to-Income Ratio: **0%**
- Stated Monthly Income: **$1000**
- Open Credit Lines: **3**
- Decision: **Denied**

Reasoning:

- Higher interest rate and lower credit score.
- High monthly installment compared to income raises payment concerns.
- Lack of homeownership indicates less financial stability.
- Low income and fewer open credit lines contribute to the denial decision.

CONCLUSION

- Successfully developed a **robust machine learning model** for predicting loan approvals.
- Implemented **extensive preprocessing** steps and addressed **class imbalance** to ensure data quality.
- **Integrated** the **top-performing model** into a user-friendly **web application**, enabling real-time loan approval predictions and enhancing decision-making.
- The project lays the **groundwork** for **automating loan approvals** and can be further improved to add significant value to financial institutions.

THANK YOU

ANY QUESTION?