PREDICTING BANK LOAN APPROVAL USING MACHINE LEARNING

NAME: OLAWALE FRANCIS ONAOLAPO | **DATE:** MAY 08, 2025

WORD COUNT (EXCLUDING REFERENCES, TITLE PAGE AND ABSTRACT): 2400 words

Abstract

This study develops and evaluates machine learning models to predict personal loan approval, addressing the critical need for accurate, fair, and efficient loan allocation in financial institutions. Using a Kaggle dataset with 5,000 records and 14 features, Logistic Regression, Decision Trees, Random Forests, and Gradient Boosting was applied to predict loan approval based on features like income, credit history, and demographics. Data preprocessing involved handling class imbalance with SMOTENC, mitigating multicollinearity, and normalizing features. Hyperparameter tuning and feature importance analysis enhanced model performance and transparency. Gradient Boosting achieved the best results, with 99.0% accuracy, 100.0% AUC, and 0.9% false positive rate, minimizing financial risk and ensuring fairness. The models address ethical concerns like bias and legal requirements for explainability, with a prototype web application deployed for real-time predictions. Future work includes exploring deep learning and fairness-aware methods to enhance scalability and equity.

Keywords: Loan Approval Prediction, Machine Learning, Gradient Boosting, SMOTENC, Class Imbalance, Feature Importance, Fair Lending, Model Transparency, Financial Risk, Hyperparameter Tuning

1 INTRODUCTION

1.1 PROBLEM DEFINITION AND SIGNIFICANCE

Banks and financial institutions rely on predictive models to assess loan eligibility. This study aims to develop a machine learning model that predicts whether a customer will be approved for a personal loan based on features such as income, credit history, and demographic data.

Accurate prediction is critical for financial institutions to optimize loan allocation, reduce default risks by identifying high-risk applicants, and enhance customer satisfaction by automating loan approval processes, and reducing manual review time. This ensures regulatory compliance by ensuring fair lending practices by avoiding discriminatory biases. Misclassification can lead to financial losses or unfair denials, making this a high-stakes, real-world problem.

Figures 1 and 2 depict the application development process and the specific model development steps, respectively.

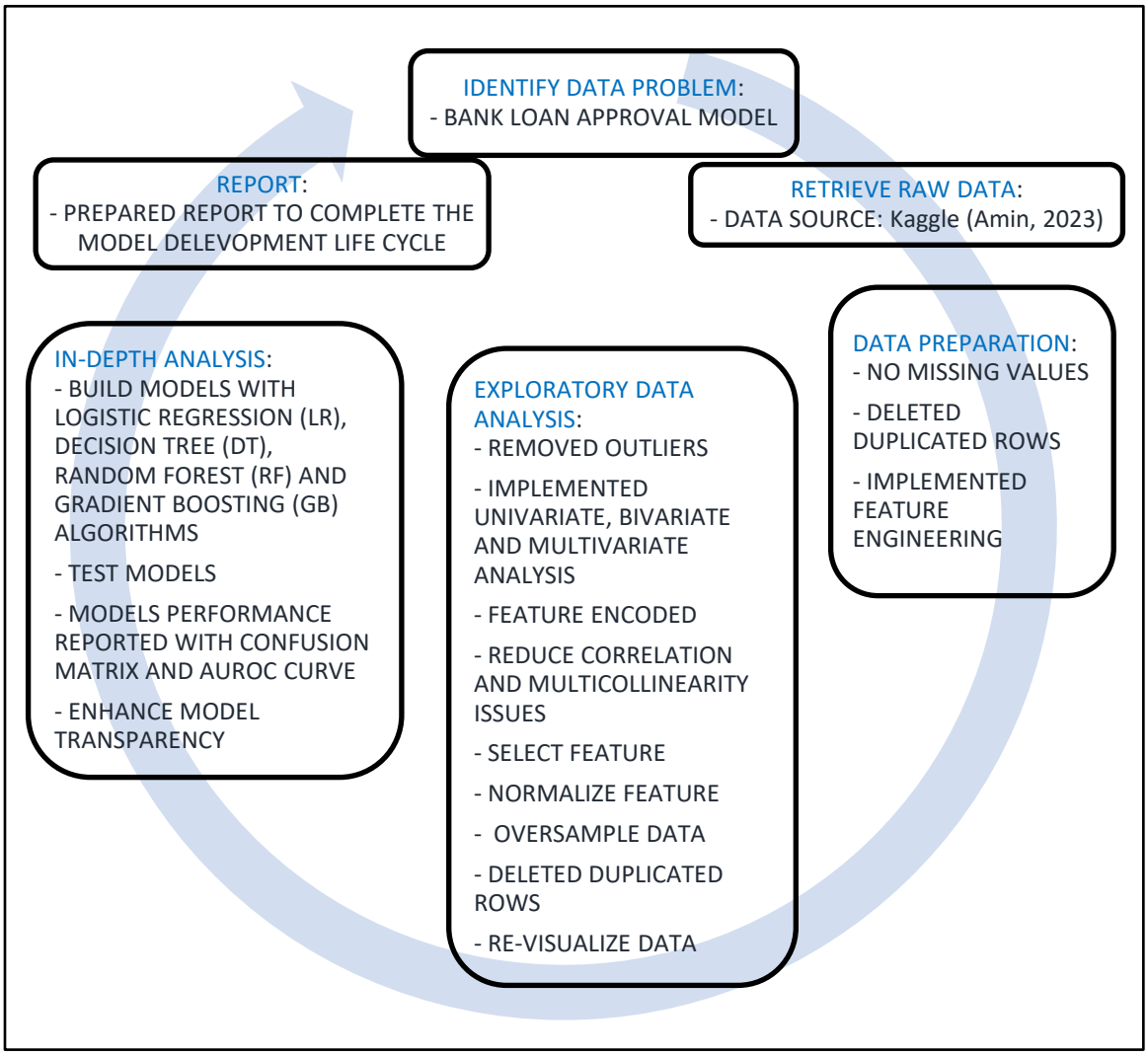

Figure 1

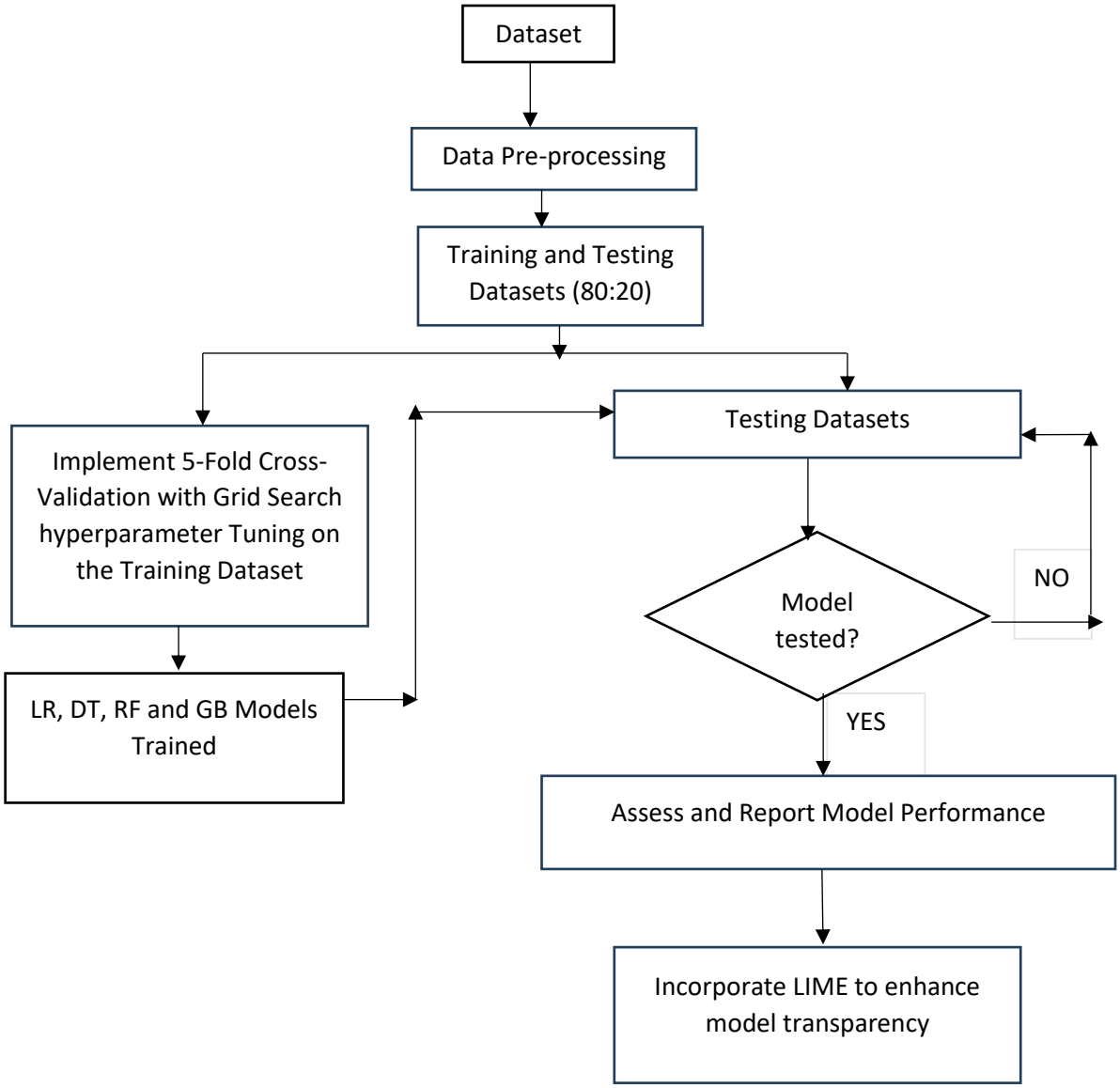

Figure 2

1.2 LITERATURE REVIEW

Previous studies have applied various machine learning algorithms for credit scoring, including logistic regression, decision trees, and ensemble methods like random forests and gradient boosting (Lessmann et al., 2015). Loan approval prediction is a well-explored area in machine learning. Logistic Regression is widely used for its interpretability and effectiveness in binary classification tasks (Hosmer & Lemeshow, 2000). Decision Trees are valued for their simplicity and ability to model non-linear relationships. Random Forests, an ensemble of decision trees, improve predictive performance and robustness by reducing overfitting (Breiman, 2001). Gradient Boosting, another ensemble method, enhances accuracy

by iteratively correcting errors and has shown significant advancements in predictive performance (Chen & Guestrin, 2016).
To address class imbalance, SMOTE technique and its variants can be used to generate synthetic samples to balance datasets (Chawla et al., 2002). Recent studies also highlight the importance of hyperparameter tuning and feature importance analysis to optimize model performance (Bergstra & Bengio, 2012). This study builds on these foundations by applying SMOTENC to handle class imbalance, performing rigorous hyperparameter tuning, and systematically comparing Logistic Regression, Decision Trees, Random Forests, and Gradient Boosting to address data quality issues and improve loan approval prediction.

2 DATA EXPLORATION AND FEATURE SELECTION

2.1 DATASET OVERVIEW

The dataset, sourced from Kaggle (Amin, 2023), includes 5,000 records and 14 features for personal loan approval prediction. Table 1 provides information about the feature attributes.

S/N	Features	Attributes
1	Age	Numeric - Continuous Quantitative Data
2	Income	
3	Experience	
4	CCAvg (Credit Card Average Score)	
5	Mortgage (later categorized to binary categorical feature, named Mortgage Category)	
6	Family (number of family members)	Numeric - Discrete Quantitative Data
7	Education (Ordinal: 1 = High School, 2 = Bachelors, 3 = Masters & Above)	Categorical - Ordinal Qualitative Data Type (Non-Binary Categorical)
8	Securities.Account (Securities Account)	Categorical - Nominal Qualitative Data Type (Binary Categorical)
9	CD.Account (Certificate of Deposit Account)	
10	Online (Online Banking)	
11	CreditCard (Credit Card)	
12	ZIP.Code (Zip Code)	Categorical - Nominal Qualitative Data Type (Non-Binary Categorical)
13	Personal.Loan (renamed Personal Loan Status)	Boolean - Target Variable (binary: 0 = Rejected, 1 = Approved)
14	ID	Identifier—ID (dropped as non-predictive)

Table 1

2.2 DATA EXPLORATION

Initial exploration confirmed no missing values or duplicated rows in the dataset. However, inconsistencies were identified. 52 rows with negative Experience values were removed, as negative work experience is illogical. The zip code contained valid but inconsistent 4- and 5-digit formats, which were standardized to two-digit prefixes to cluster applicants by region, reducing dimensionality while retaining regional information. Post-feature engineering, one duplicated row was found and deleted.

Descriptive statistics showed insights such as, 30% of applicants had credit cards, and 3,421 had zero mortgage values. Univariate, bivariate, and multivariate analyses were conducted. Categorical features were manually mapped from numerical to text descriptions for interpretable visualizations, yielding several insights. Visualizations highlighted a significant class imbalance in the target variable (Figure 3). Income, Credit Card Average Score, and Mortgage displayed skewed distributions with outliers (Figures 4 and 5); **outliers** in Income (Figure 4) and Credit Card Average Score were **removed using robust Z-score**, while Mortgage was categorized into zero (no mortgage) and non-zero (have mortgage) groups (Figure 7). Age and Experience had no outliers (Figure 6).

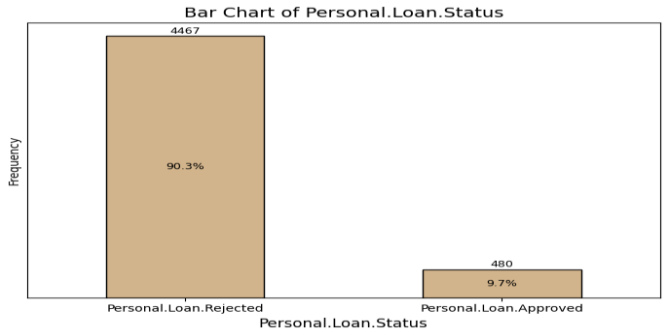

Figure 3

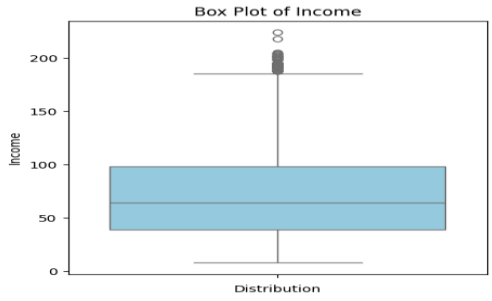

Figure 4

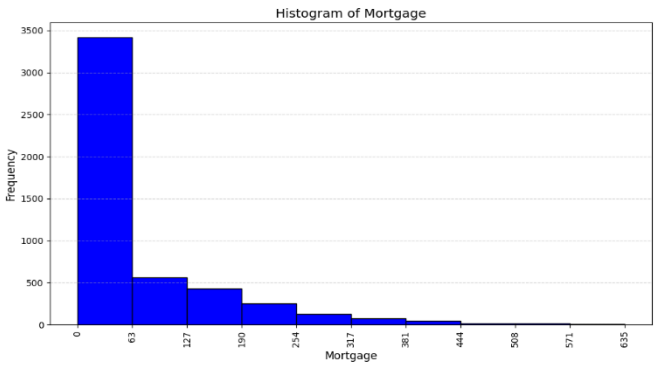

Figure 5

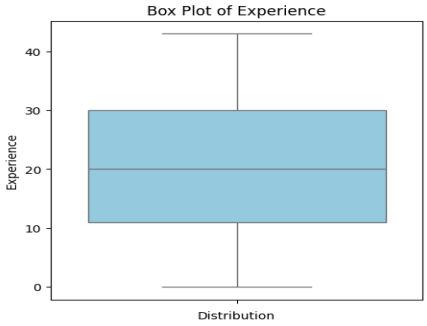

Figure 6

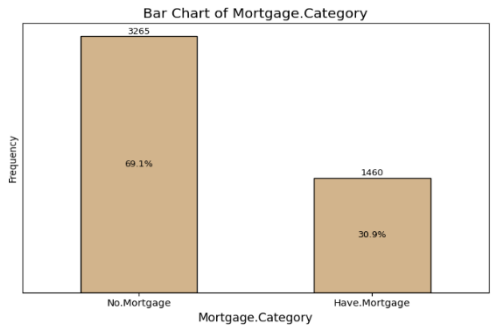

Figure 7

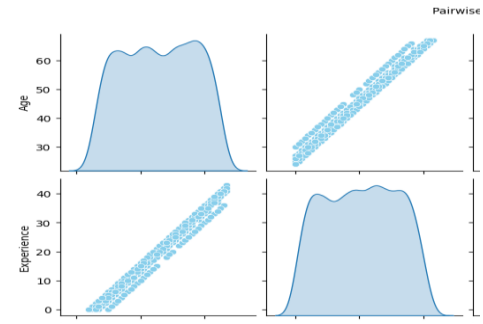

Figure 8

The Scatterplot matrix chart revealed a strong positive correlation between Age and Experience (Figure 8).

Predictive influence analysis indicated Credit Card Average Score, Income, and Education had high predictive power due to varying distributions across target categories (Figures 9–11), while Age, Experience, and CreditCard showed low influence with similar distributions (Figures 12–14).

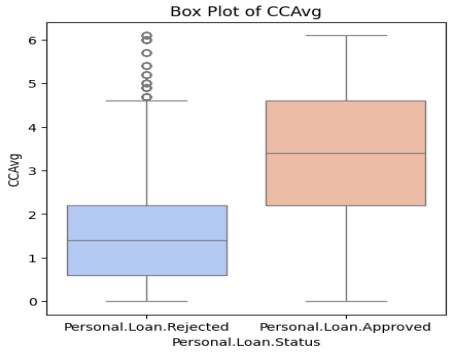

Figure 9

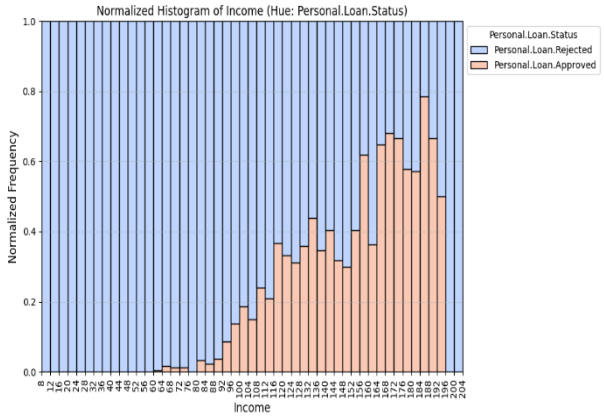

Figure 10

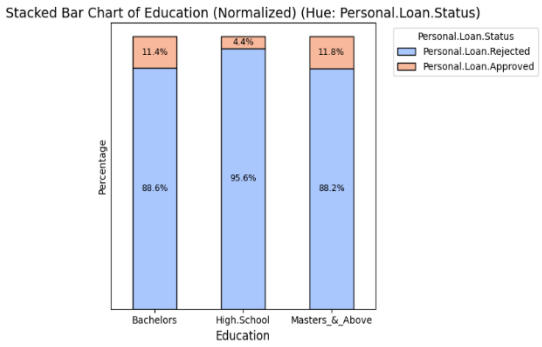

Figure 11

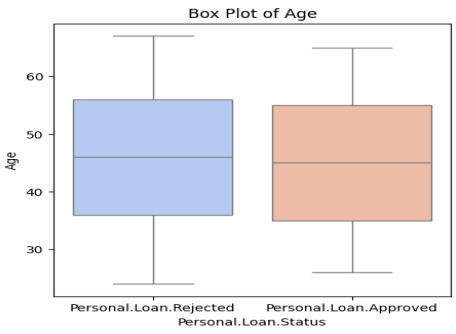

Figure 12

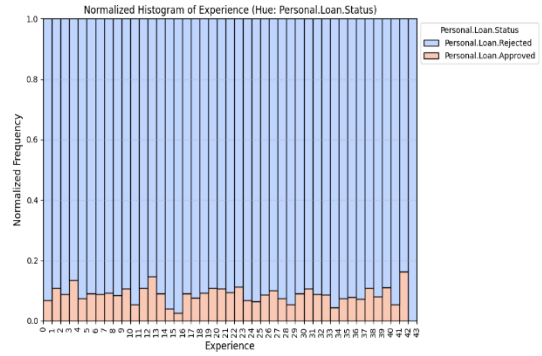

Figure 13

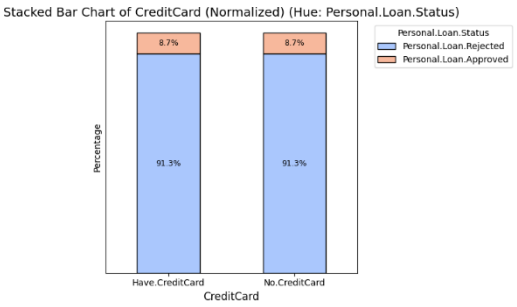

Figure 14

Bar charts with frequencies color-coded by the target variable (Figures 15 and 16) were generated to evaluate the statistical significance of features' predictive influence, identifying potential biases from attributes like applicants with 42–43 years of Experience, which had limited data representation.

The scatter plot matrix with points color-coded by the target variable (Figure 17) indicated that tree-based models would be effective, as most target category clusters were not linearly separable.

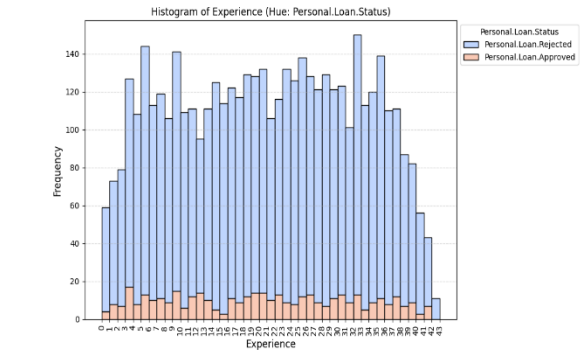

Figure 15

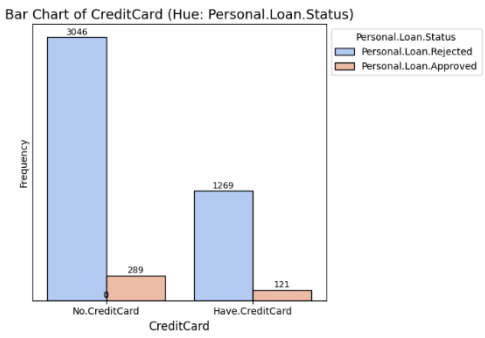

Figure 16

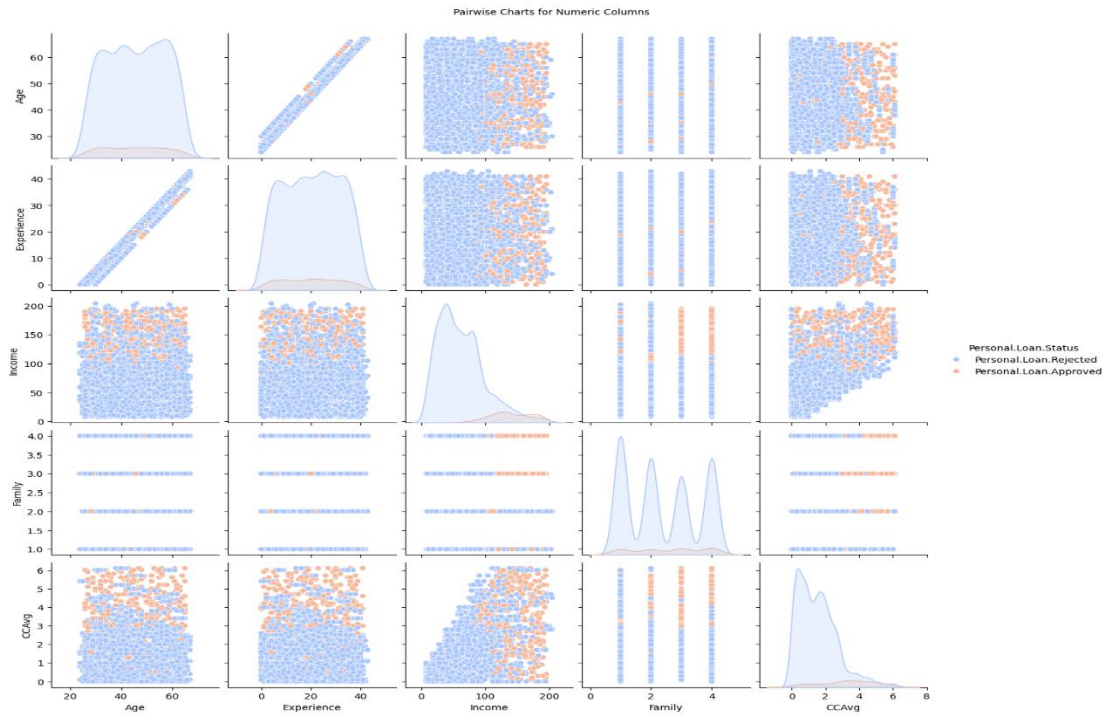

Figure 17

2.3 DATA PREPROCESSING

2.3.1 CATEGORICAL ENCODING

Non-zip code categorical variables were manually re-encoded to numerical values. The ZIP.Code feature was one-hot encoded, and the resulting dummy variable ZIP_96 was removed to avoid multicollinearity, as its variance was below 0.05 according to sklearn.feature_selection.VarianceThreshold, indicating minimal predictive influence. The retained dummy variables are ZIP_90 to ZIP_95.

2.3.2 FEATURE CORRELATION, MULTICOLLINEARITY ANALYSIS, AND FEATURE SELECTION

Figure 18 reveals a strong correlation between Age and Experience and a moderate correlation between Income and Credit Card Average Score. The initial condition index is 19, with VIFs up to 405 (Figure 19), indicating strong multicollinearity. After removing Age, the condition index decreases to 14 and VIFs drop below 7 (Figure 20), lowering multicollinearity to a moderate level, with VIFs below 10 (Dormann et al., 2013).

Income and Credit Card Average Score were retained for their critical role in loan approval prediction, and all kept features surpassed a variance threshold of 0.05, as determined by sklearn.feature_selection.VarianceThreshold, for further analysis.

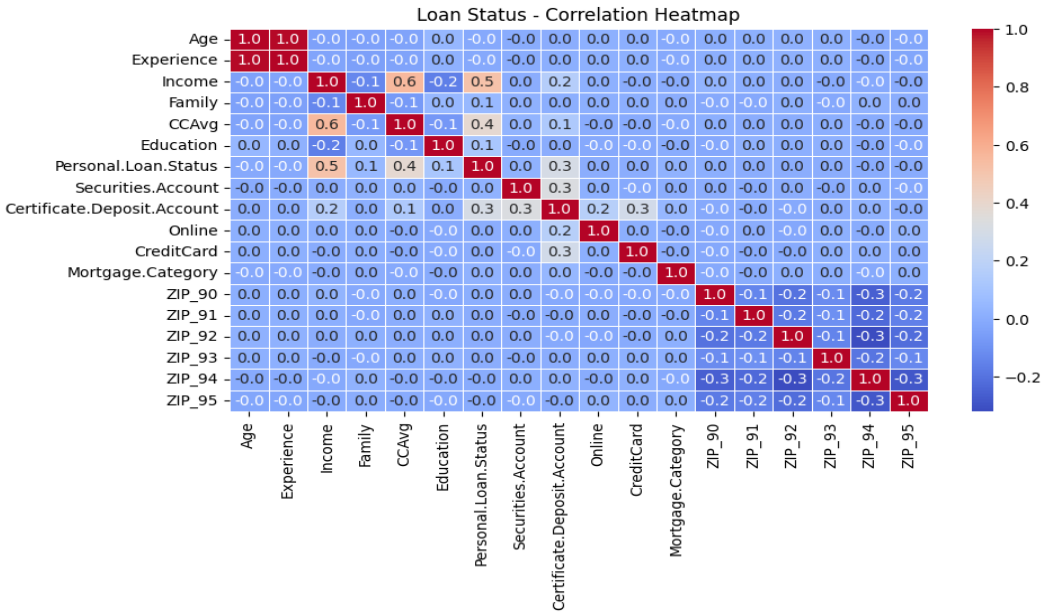

Figure 18

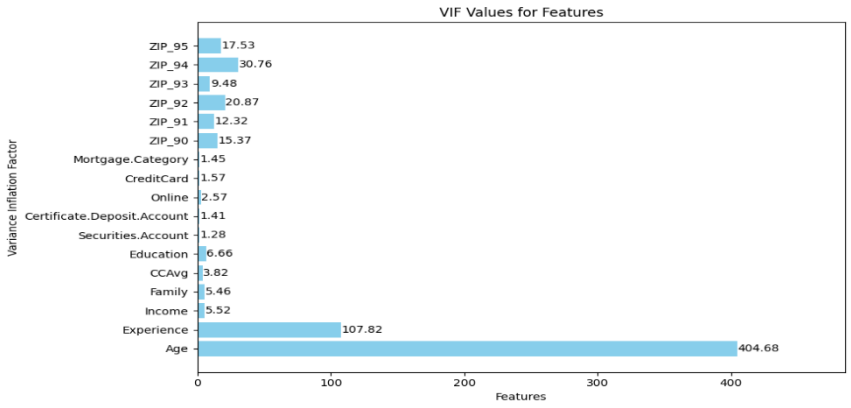

Figure 19

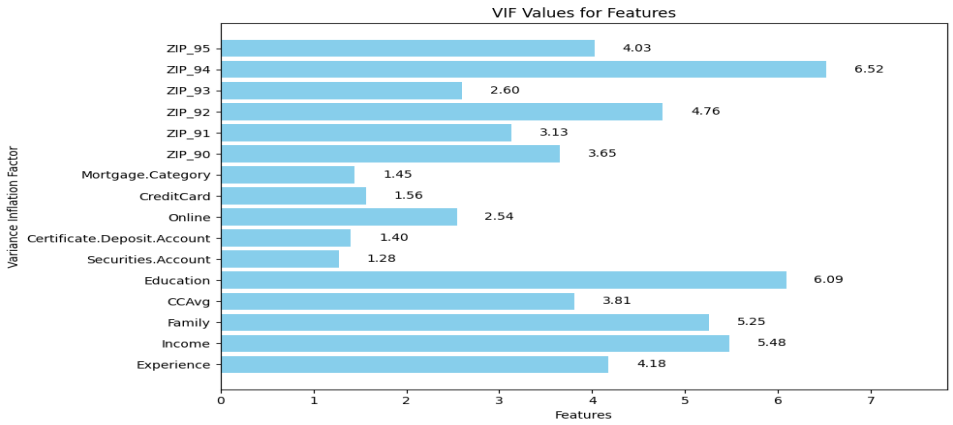

Figure 20

2.3.3 NORMALIZATION

Numerical features were normalized using Robust Scaler, to reduce the impact of extreme values.

2.3.4 CLASS IMBALANCE MITIGATION

SMOTENC was applied to address the 9:1 rejection-to-approval ratio, generating synthetic samples for the minority class (approved loans) to achieve a 50:50 distribution. In SMOTENC, numerical continuous features were processed as numerical, while categorical and numerical discrete features were processed as categorical (Imbalanced-learn, 2023). A single duplicate row was removed from the oversampled data. Figures 21 and 22 illustrate the target distribution before and after oversampling, respectively.

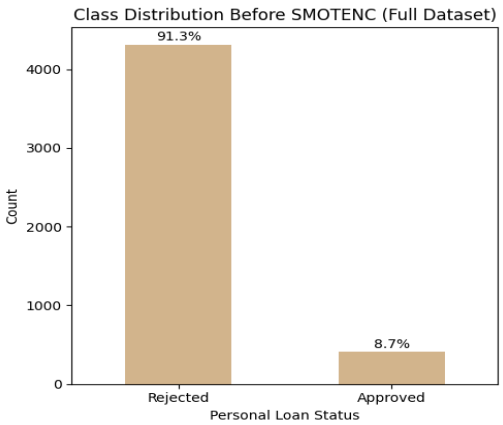

Figure 21

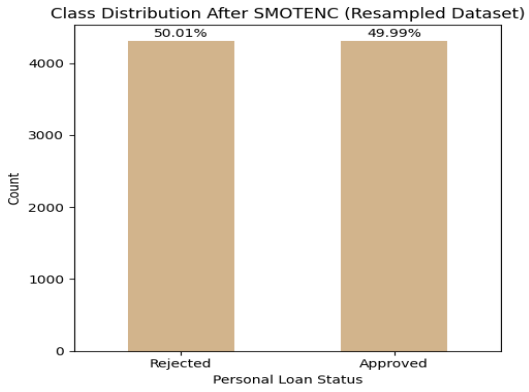

Figure 22

3 EXPERIMENTS

3.1 PROPOSED SOLUTIONS AND JUSTIFICATION

For personal loan approval prediction in a binary classification task, the tree-based algorithms were chosen for their robustness to correlation and multicollinearity. Regularization was applied to logistic regression (LR) to mitigate the impact of correlation and multicollinearity. Regularization stabilizes coefficients, enhances interpretability, and improves generalization.

The four algorithms were also selected for their complementary strengths in managing bias, variance, and data patterns: Logistic Regression (LR), an interpretable, low-variance baseline suited for linear relationships but limited by high bias for non-linear patterns; Decision Tree (DT), which captures complex patterns with low bias but risks overfitting, mitigated by pruning; Random Forest (RF), an ensemble method with low bias and variance when tuned, resilient to noise and providing feature importance; and Gradient Boosting (GB), which achieves high accuracy by iteratively boosting and modeling subtle interactions, though requiring tuning to prevent overfitting. This selection ensures a comprehensive evaluation to determine the optimal model.

3.2 DESIGN AND IMPLEMENTATION

3.2.1 MODEL TRAINING AND HYPERPARAMETER TUNING

The preprocessed dataset was split into 80% training and 20% test sets. The four selected algorithms were initially trained with default parameters on both the imbalanced and balanced datasets to establish baseline performance. Subsequently, the algorithms were trained on the balanced dataset, with hyperparameter tuning performed using grid search and 5-fold cross-validation. The random state was set to 42 for the data stratification and other processes that involve randomization. Table 2 displays the optimal parameters from the grid search cross-validation.

Models	Best parameters
LR	'C': 0.1, 'penalty': 'l2', 'solver': 'liblinear'. The maximum iteration was set to 1000
DT	'ccp_alpha': 0.0, 'criterion': 'entropy', 'max_depth': 9, 'max_features': None, 'min_samples_leaf': 1, 'min_samples_split': 10
RF	'max_depth': 10, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100
GB	'learning_rate': 0.2, 'max_depth': 5, 'n_estimators': 200

Table 2

3.2.2 BIAS AND VARIANCE

Figures 23-26 show charts of model performance across tuned hyperparameters, indicating low bias and low variance for the best hyperparameters.

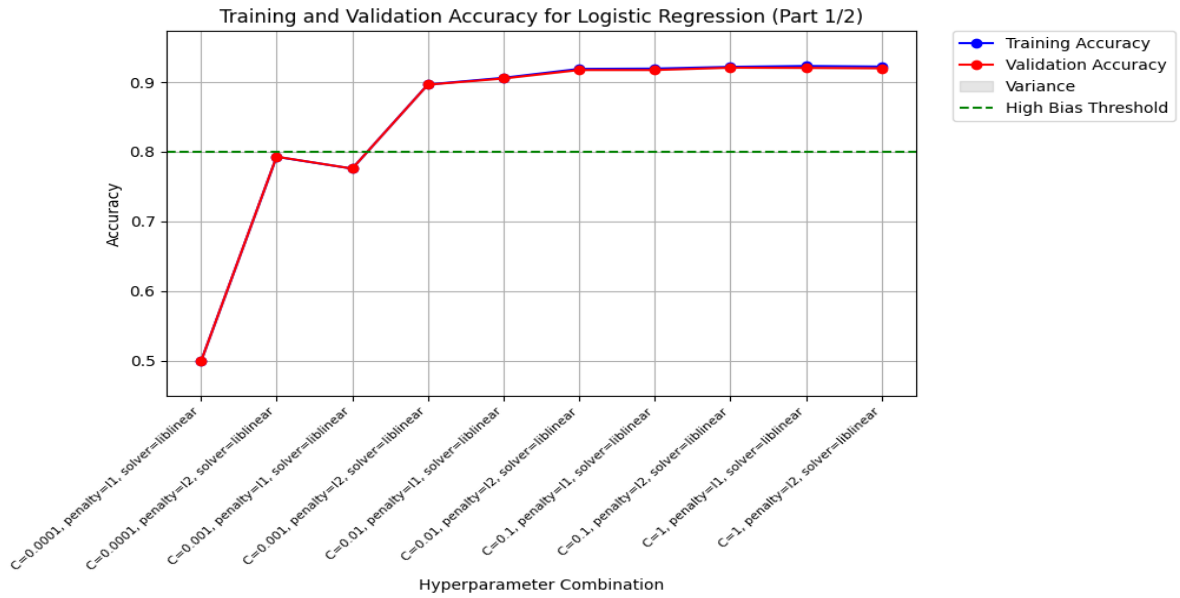

Figure 23

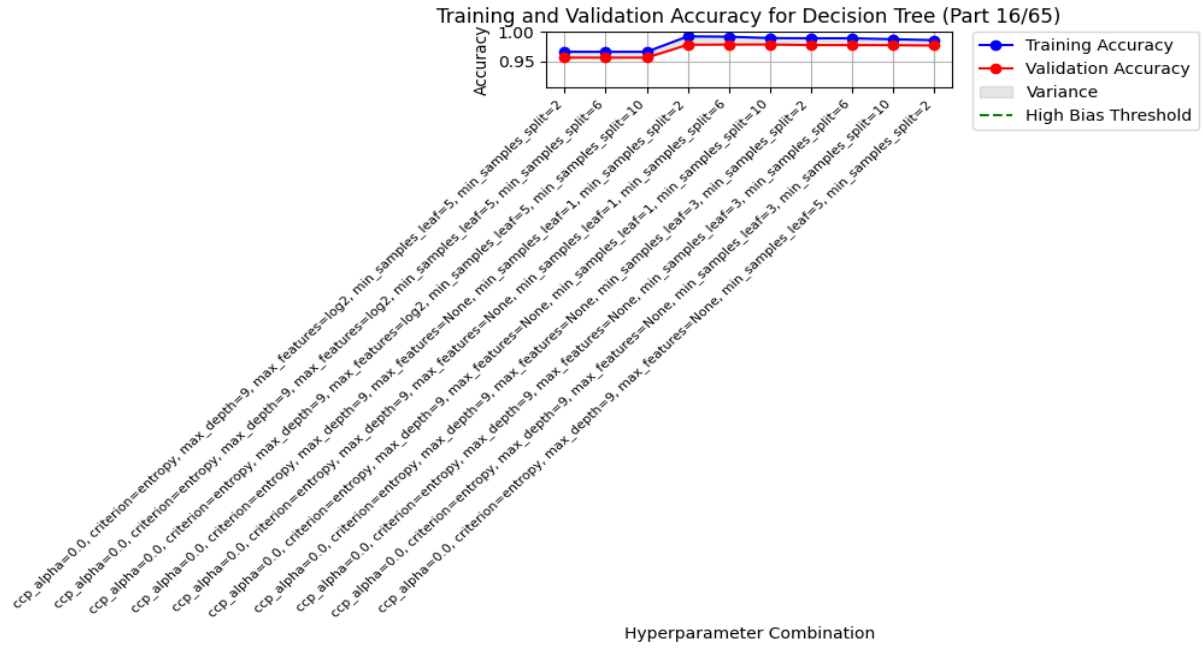

Figure 24

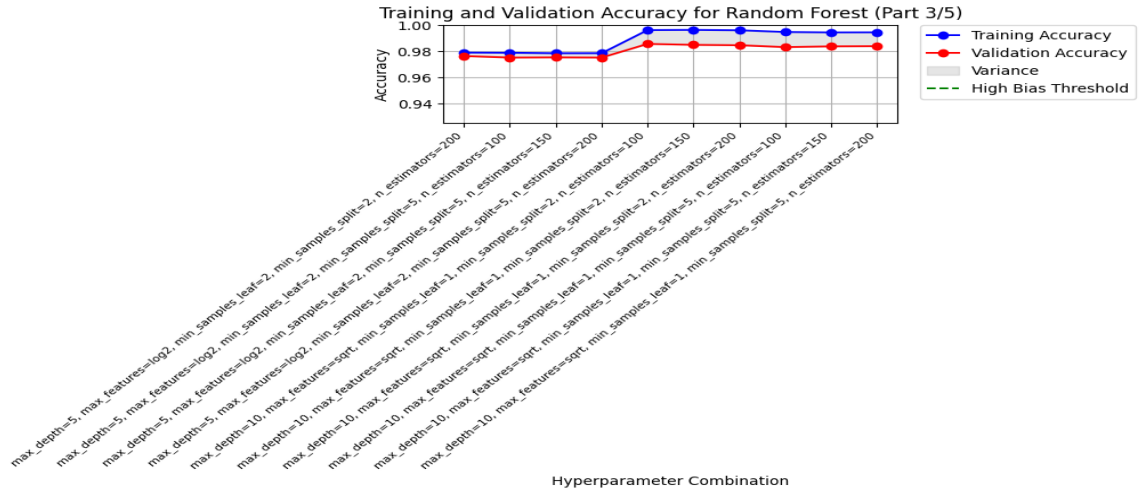

Figure 25

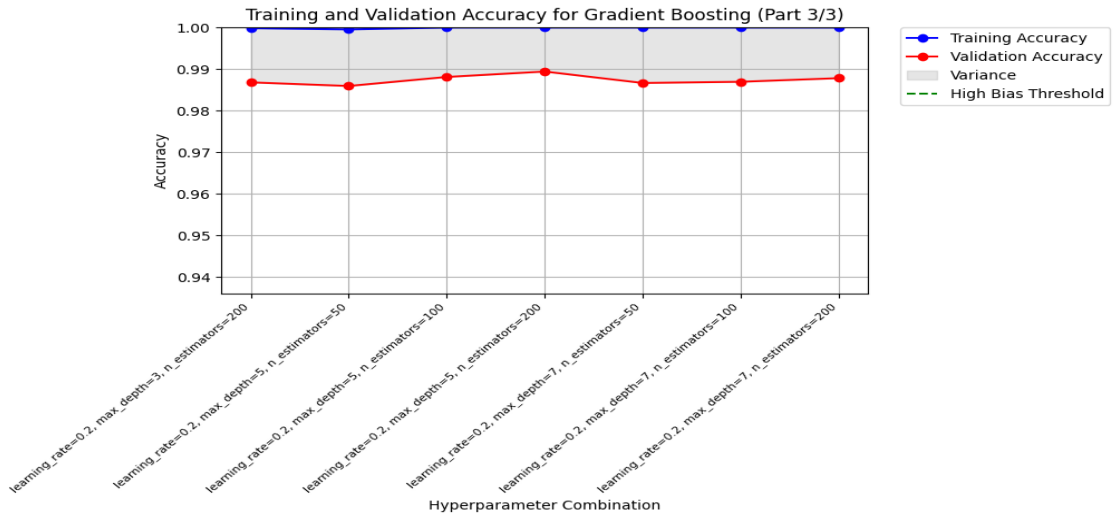

Figure 26

Figures 27 and 28 illustrate low bias and low variance across all models. Figure 27 depicts high, stable training and validation accuracies in 5-fold cross-validation for the optimal hyperparameters, whereas Figure 28 underscores high, stable training, validation, and test accuracies, showcasing robust model generalization.

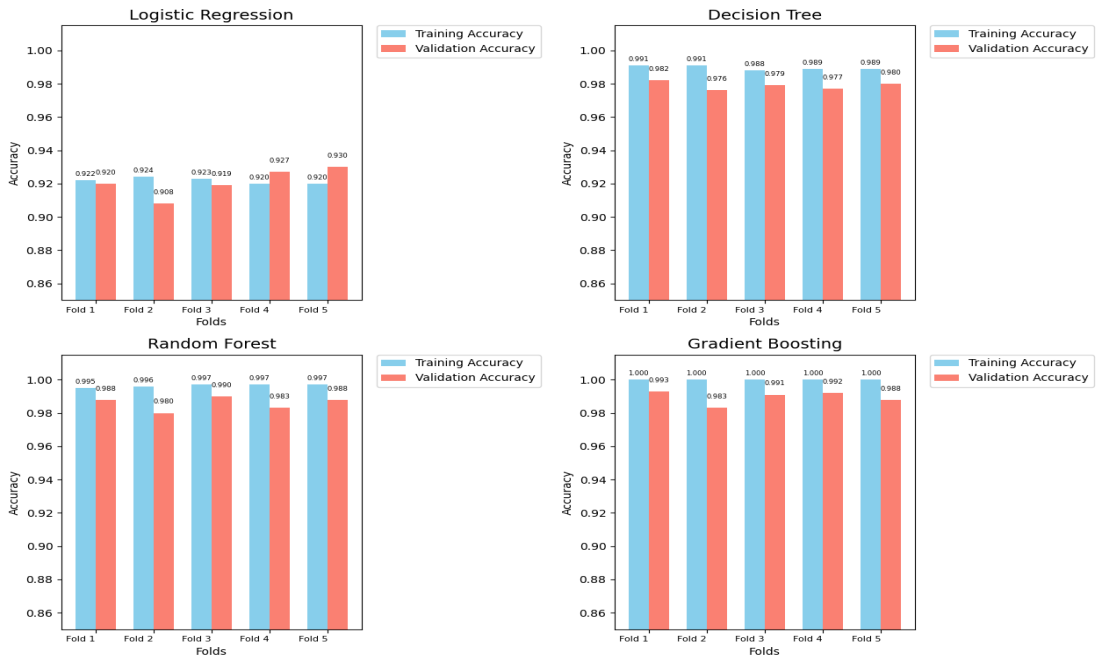

Figure 27

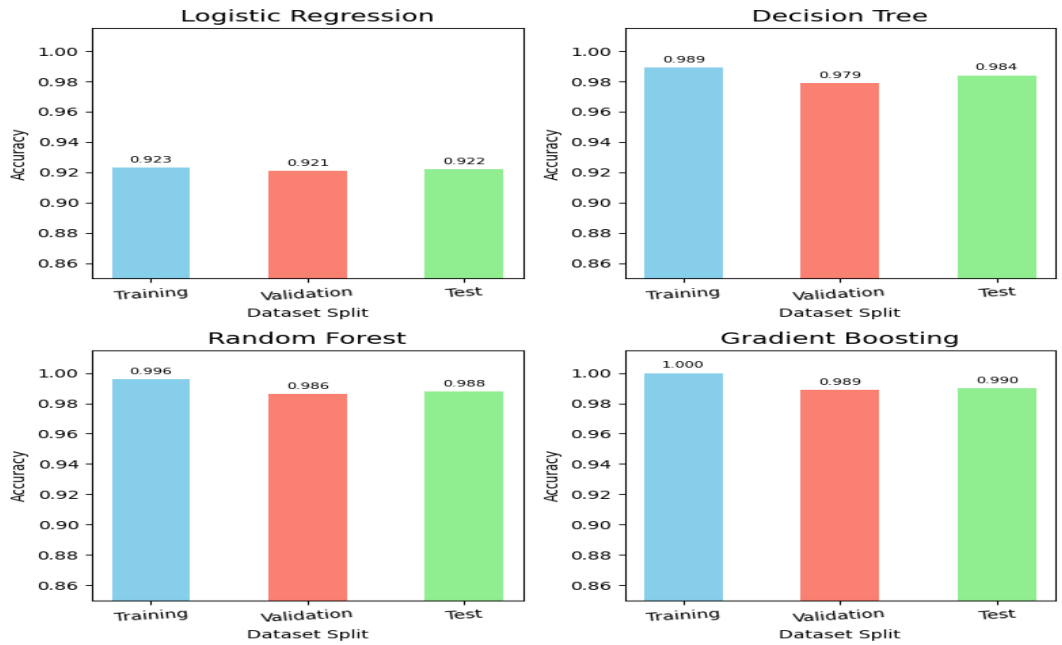

Figure 28

3.2.3 MODEL TRANSPARENCY

Feature importance and LIME were used to enhance model explainability.

3.2.3.1 FEATURE IMPORTANCE

Feature importance reveals global contributions of features to the models' decisions. Figures 29–32 for LR, DT, RF, and GB confirm Income and Credit Card Average Score as highly predictive, while Credit Card and Experience show low influence, consistent with visualization findings.

The feature importance of LR is inconsistent with DT, RF, and GB, likely due to the moderate multicollinearity among features. Although LR predictions remain reliable, its feature importance is unreliable for explaining model decisions, as multicollinearity causes unstable coefficients (Dormann et al., 2013; Gujarati, 2003).

Although certain features contribute minimally to the model's predictions, none were eliminated because the model maintains high performance without adverse effects from these features, and the feature set is already small.

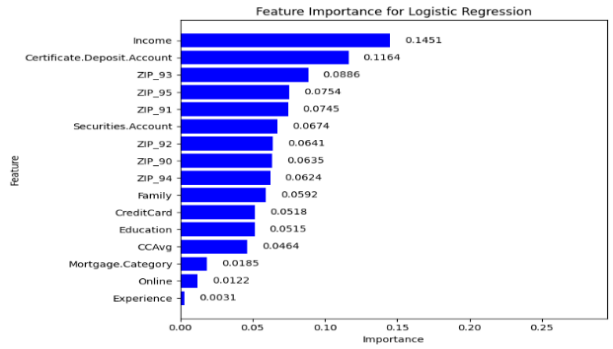

Figure 29

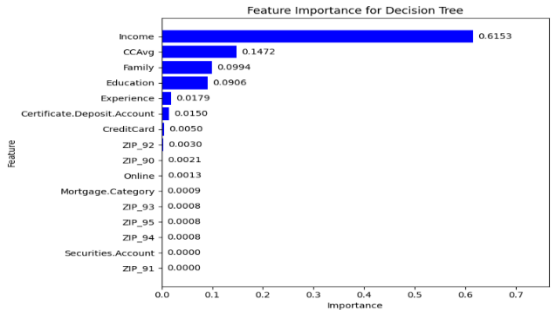

Figure 30

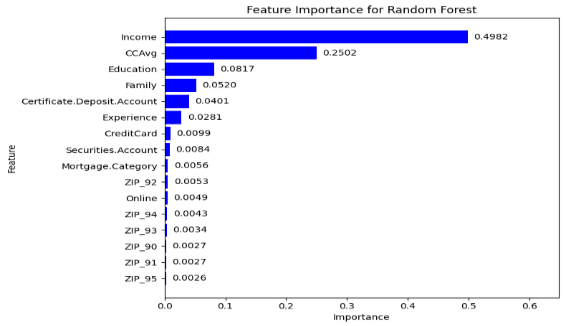

Figure 31

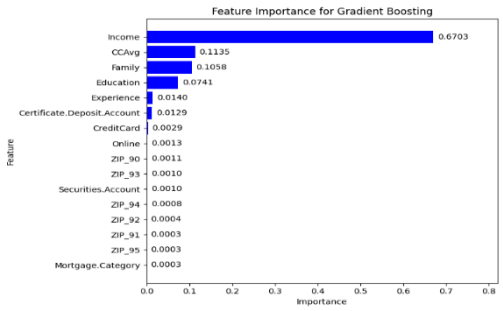

Figure 32

3.2.3.2 LIME EXPLANATION

LIME enhances feature importance by revealing local feature contributions for individual predictions, including positive and negative influences on the prediction. Figures 33–34 for the GB model confirm visualization findings, with Income and Credit Card Average Score showing high predictive influence and Credit Card and Experience being less influential.

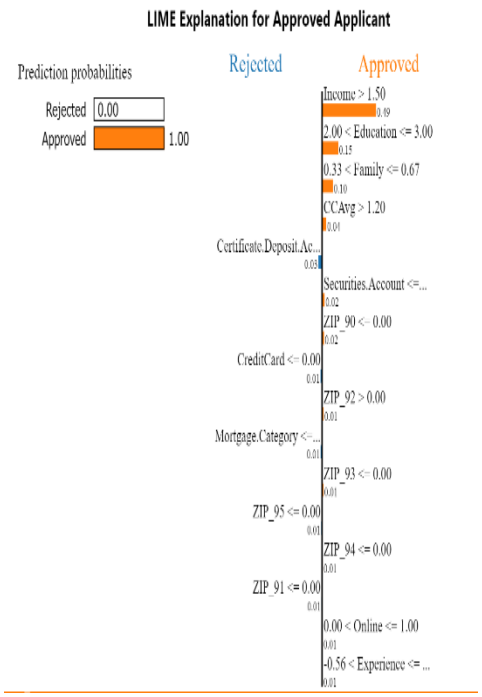

Figure 33

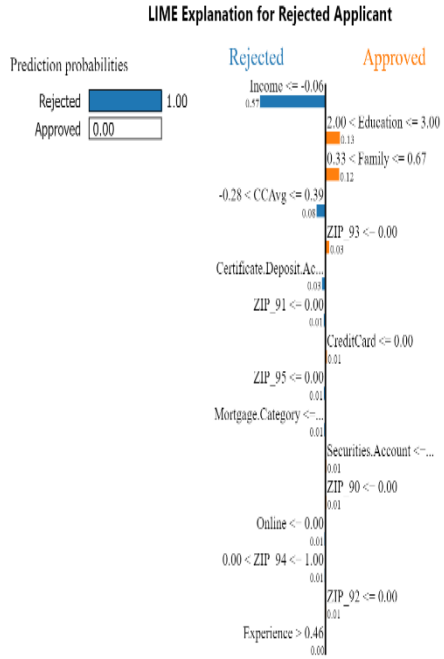

Figure 34

4 RESULTS

The results of the models were assessed with confusion matrix and AUROC curve.
The confusion matrices provide information about the True Positive, True Negative, False Positive and False Negative.
The AUROC (Area Under Receiver Operating Characteristic) curve demonstrates the trade-off between the true positive rate (TPR) and false positive rate (FPR) at various thresholds.

4.1 CONFUSION MATRIX

Figures 35-38 show the confusion matrices for the four models.

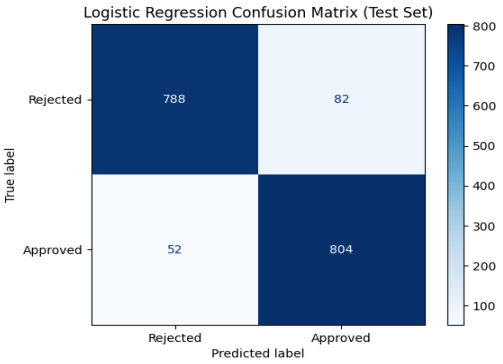

Figure 35

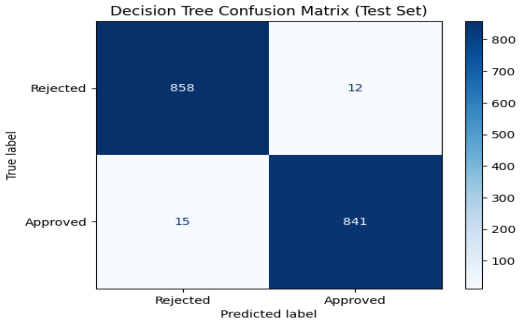

Figure 36

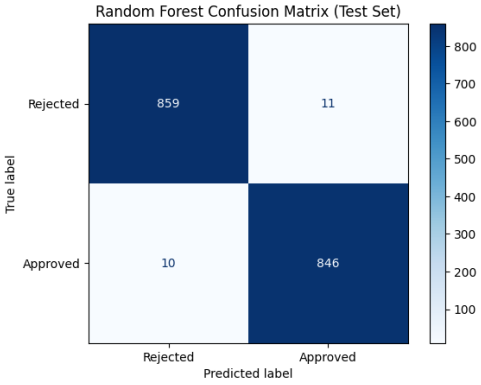

Figure 37

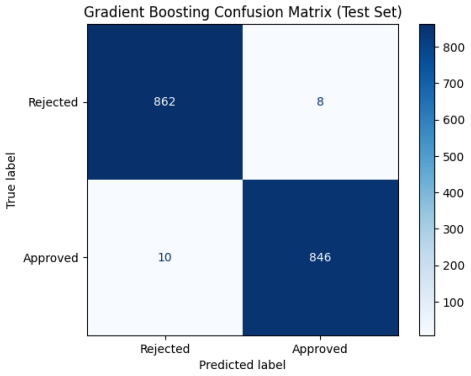

Figure 38

4.2 AUROC CURVE

Figure 39 presents the AUROC curves for the models.

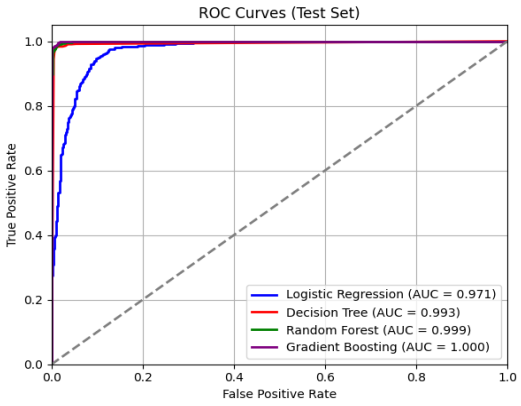

Figure 39

4.3 MODEL ASSESSMENT METRICS

Figure 40 shows the models' accuracy, precision, recall, F1-score, and False Positive Rate from the confusion matrices, along with AUC values obtained from the AUROC curves.

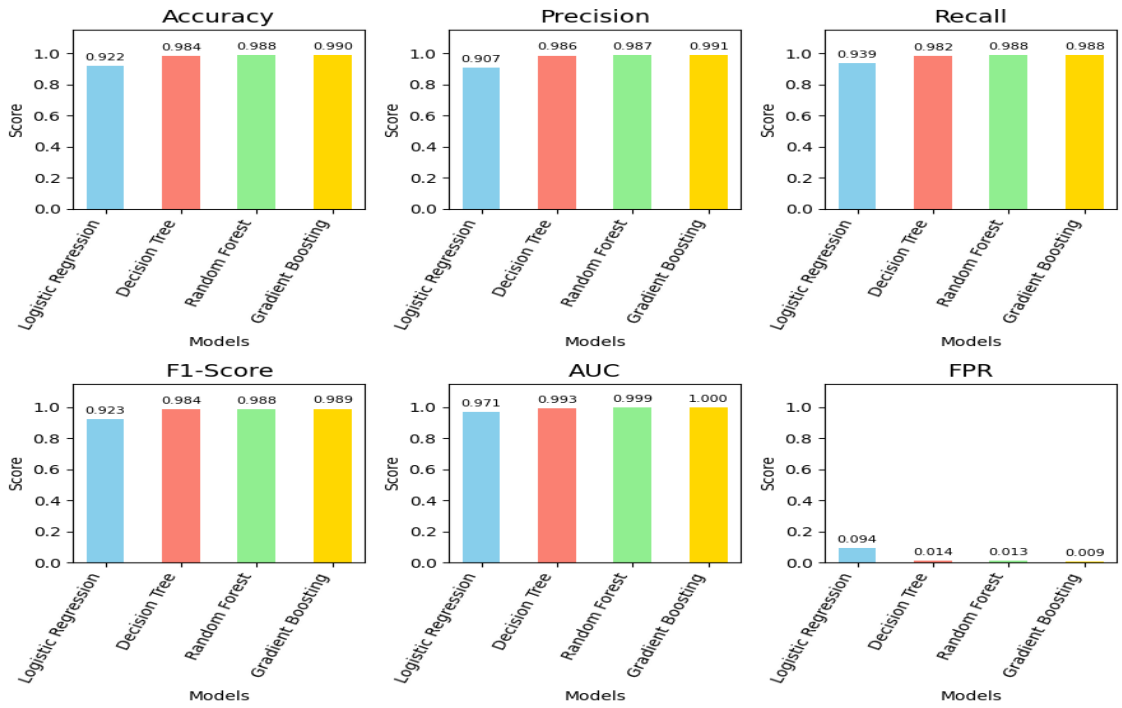

Figure 40

5 DISCUSSION, CONCLUSIONS, AND FUTURE WORK

5.1 DISCUSSION

The four models— LR, DT, RF, and GB—demonstrated strong performance, with metrics (accuracy, precision, recall, F1-score, AUC) above 90% and False Positive Rate (FPR) below 10% (Figure 40). GB out-performed the other models with 99.0% accuracy, 99.1% precision, 98.8% recall, 98.9% F1-score, 100.0% AUC, and 0.9% FPR (Figure 41). While accuracy suggests 99.0% of predictions are correct, it can be misleading as it doesn't account for Type I (false positives) and Type II (false negatives) errors. GB's 98.8% recall ensures most qualified loan applicants are approved, promoting fairness, though potential dataset biases (e.g., lack of diversity) remain a concern. GB 100.0% AUC reflects excellent discriminative ability, and 99.1% precision with 0.9% FPR ensures minimal number of approved loan applicants are wrongly approved, making GB ideal for reducing financial risk to both the company and the applicants (with potential risk to lose their collateral).

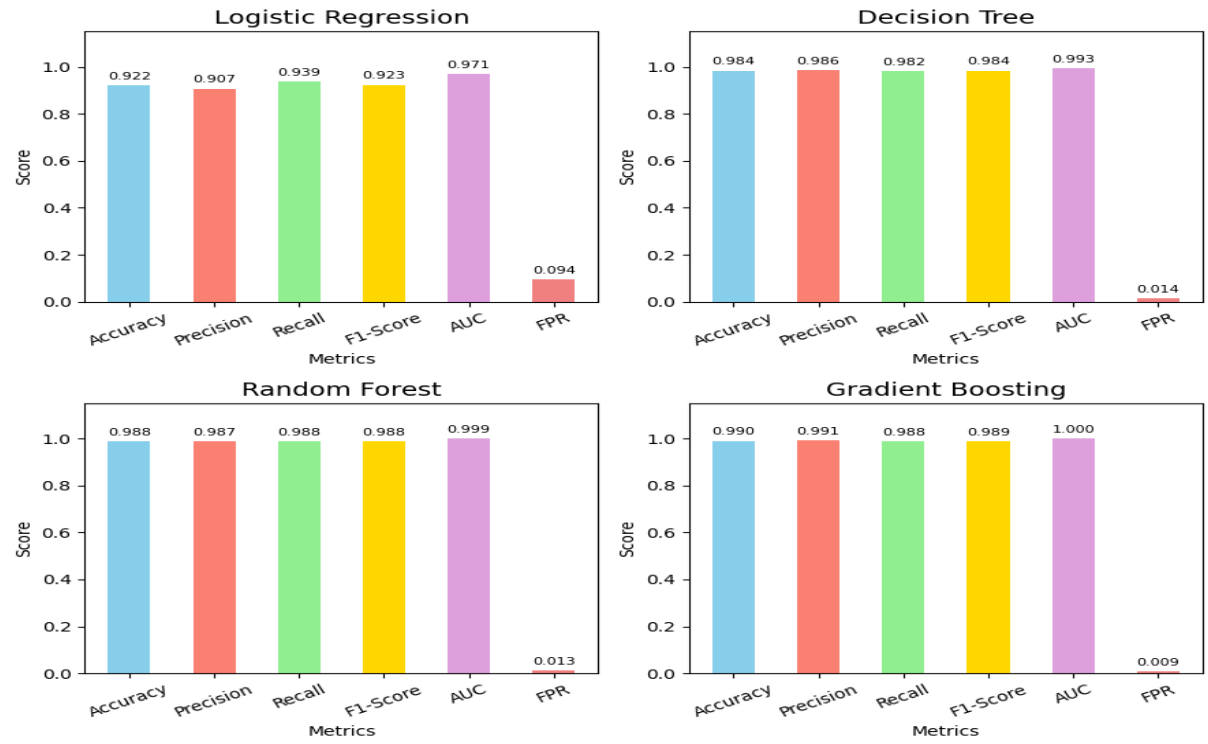

Figure 41

LR's interpretable coefficients and LIME explanations (Figures 33 and 34) help meet regulatory requirements for explainability, such as justifying loan denials.

The implemented preprocessing ensured data quality, a professional concern, by removing inconsistencies (e.g., negative Experience, duplicates), outliers, and applying categorical encoding, ZIP Code standardization, and Robust Scaler normalization (Bergstra & Bengio, 2012). Ethically, features like Income and ZIP Code risked socioeconomic or regional bias; removing Age and encoding ZIP Code as prefixes mitigated some risks, but fairness audits (e.g., demographic parity) are needed. ZIP Code anonymization ensures privacy of customer data, which is a legal requirement.

Limitations include the small dataset size, which may pose deployment challenges.

5.2 CONCLUSIONS

This study successfully developed machine learning models for personal loan approval prediction, with the Gradient Boosting model achieving the best performance. Using SMOTENC for class imbalance, removing Age to mitigate multicollinearity, and applying feature importance and LIME for transparency, the models identified Income and Credit Card Average Score as key predictors. These results help financial institutions optimize loan allocation, reduce default risks by minimizing wrong approvals, and ensure fair lending by minimizing bias through model explainability, while addressing professional (data quality), ethical (fairness), and legal (privacy, explainability) considerations.

5.3 FUTURE WORK

Future work could explore deep learning models for complex patterns in larger datasets, incorporate features like real-time credit scores or economic indicators, and use more diverse datasets to address potential bias and improve generalizability.

Comparing SMOTENC with techniques like ADASYN, optimizing GB's computational complexity for real-time deployment, ensuring fairness with fairness-aware methods, and incorporating time-series data (e.g., credit score trends) could further enhance accuracy, fairness, and scalability of loan approval solutions.

5.4 PROTOTYPE DEPLOYMENT

A prototype web application for loan approval prediction, powered by DT, GB, LR, and RF models, is hosted at https://bankloanapp.onrender.com. Developed using FastAPI and Python 3.12, and hosted on Render's free tier, it includes a homepage (bankloanindex.html) and a /predict endpoint for real-time inference. The free tier causes the app to sleep after 15 minutes of inactivity, leading to a 30–60 seconds delay when reactivated.

The homepage will be enhanced with input guidelines (e.g., Experience: 0–40, Credit Card Average Score: 0–10) to improve usability. Model refinements will follow recommendations from the Future Work section, and LIME explanations will be integrated.

A desktop installable version compatible with Windows Operating Systems can be acquired by reaching out to Olawale Onaolapo at olawaleonaolapo@yhahoo.com.

5.5 SECURITY MEASURES

To secure the bank loan approval prediction model, a robust security framework is vital, emphasizing Python library integrity and confidentiality. Key measures include verifying dependencies with pip-audit, sourcing libraries from PyPI, and preventing supply chain attacks via cryptographic checks. Secure coding involves code reviews, static analysis with Bandit, and environment isolation using venv or Docker. Data protection entails encrypting inputs with cryptography and applying differential privacy via diffprivlib. Runtime security includes library monitoring and sandboxing, while FastAPI's /predict endpoint requires OAuth2 and rate limiting. Logging, audit trails, and a vulnerability response plan ensure quick recovery. Regular penetration testing is essential, especially on Render's free tier.

REFERENCES

- Amin, V. (2023) Bank Loan Approval Dataset. Available at: https://www.kaggle.com/datasets/vikramamin/bank-loan-approval-lr-dt-rf-and-auc (Accessed: 25 February 2025).
- Bergstra, J. and Bengio, Y. (2012) 'Random search for hyper-parameter optimization', Journal of Machine Learning Research, 13, pp. 281–305. Available at: https://jmlr.csail.mit.edu/papers/v13/bergstra12a.html (Accessed: 25 April 2025).

- Breiman, L. (2001) 'Random forests', *Machine Learning*, 45(1), pp. 5–32. doi: 10.1023/A:1010933404324.
- Chawla, N.V. *et al.* (2002) 'SMOTE: Synthetic Minority Over-sampling Technique', *Journal of Artificial Intelligence Research*, 16, pp. 321–357. Available at: https://doi.org/10.1613/jair.953.
- Chen, T. and Guestrin, C. (2016) 'XGBoost: A Scalable Tree Boosting System', in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794. Available at: https://doi.org/10.1145/2939672.2939785.
- Dormann, C.F. *et al.* (2013) 'Collinearity: a review of methods to deal with it and a simulation study evaluating their performance', *Ecography*, 36(1), pp. 27–46. Available at: https://doi.org/10.1111/j.1600-0587.2012.07348.x.
- Gujarati, D. N. (2003) *Basic econometrics*. 4th edn. New York: McGraw-Hill.
- Hosmer, D. W. and Lemeshow, S. (2000) *Applied logistic regression*. 2nd edn. New York: Wiley.
- Imbalanced-learn (2023) 'SMOTENC', imbalanced-learn documentation. Available at: https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.SMOTENC.html (Accessed: 25 February 2025).
- Lessmann, S. *et al.* (2015) 'Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research', *European Journal of Operational Research*, 247(1), pp. 124–136. Available at: https://doi.org/10.1016/j.ejor.2015.05.030.