CUSTOMER CREDIT SCORE USING K-NEAREST NEIGHBORS

Jan Allen R. Bernabe
Bachelor of Science in Computer Science with Specialization in Machine Learning
College of Computing and Information Technologies
Manila, Philippines
janallenrb@gmail.com

Angelo Gabriel D. Castillo
Bachelor of Science in Computer Science with Specialization in Machine Learning
College of Computing and Information Technologies
Manila, Philippines
anjilugabriel@gmail.com

Mark Francis Rae G. Guerrero
Bachelor of Science in Computer Science with Specialization in Machine Learning
College of Computing and Information Technologies
Manila, Philippines
guerrero.mfr@gmail.com

***Abstract*—This project addresses the critical task of credit scoring, where machine learning techniques are employed to classify customers into three creditworthiness categories: Good, Standard, and Poor. Utilizing a dataset with essential financial features—such as Annual Income, Age, Number of Credit Cards, Interest Rate, Payment Behavior, Number of Delayed Payments, Outstanding Debt, Credit History Age, Total EMI per Month, Changed Credit Limit, Number of Credit Inquiries, and Monthly Balance—we conducted thorough preprocessing, including handling missing values, encoding categorical variables, and normalizing numerical features to ensure robust model performance. Two models were implemented: K-Nearest Neighbors (KNN) and Decision Tree classifiers. KNN was chosen for its ability to classify instances based on the proximity to known labeled examples, offering a straightforward yet powerful approach for capturing credit score patterns. Meanwhile, the Decision Tree classifier provided an interpretable model that formulates decision rules, making it a valuable tool for understanding the basis of credit score classifications. Both models were evaluated using accuracy, precision, recall, and F1-score, with the Decision Tree achieving a notably high accuracy on test data, nearly 100%, indicating strong predictive performance. Confusion matrices were analyzed to understand misclassification trends. The results underscored the predictive capability of KNN and Decision Trees in credit scoring, demonstrating their utility in risk assessment for financial institutions. This project emphasizes the role of feature engineering and data preprocessing, yielding insights that are beneficial for fair, informed lending practices.**

***Index Terms*—Credit Scoring, Machine Learning, Data Preprocessing, K-Nearest Neighbors (KNN), Decision Tree Classifier, Financial Risk Assessment, Creditworthiness Classification, Accuracy, Feature Engineering, Model Evaluation .**

I. INTRODUCTION

Credit scoring is an essential aspect of financial decision-making, used by banks, credit card companies, and other financial institutions to evaluate a customer's creditworthiness. A credit score is typically derived from various financial behaviors and serves as a predictive metric for assessing the likelihood of loan repayment or default (Thomas et al., 2002). Accurately classifying credit scores helps lenders make informed decisions, allowing them to minimize risk while also offering fair and responsible lending (Lessmann et al., 2015). Misclassification, however, could lead to financial loss for institutions or unintended discrimination against creditworthy customers (Hand & Henley, 1997). Thus, reliable credit score classification models are critical for both financial stability and consumer equity.

In today's data-driven financial landscape, accurately assessing an individual's creditworthiness is essential for financial institutions and lenders. With the increasing volume of financial data generated by consumers, machine learning techniques have become invaluable tools for predicting customer credit scores—an integral component in determining credit risk, loan approvals, and interest rates (Yap et al., 2011). This project aims to classify customers into distinct credit score categories: Good, Standard, and Poor, based on a comprehensive set of financial features.

The dataset utilized in this study includes various attributes that are vital for evaluating an individual's creditworthiness. Our analysis begins with a thorough exploratory data analysis

(EDA) to understand the dataset's structure, distribution, and potential correlations among the features. This includes visualizing relationships between features and the target variable, credit score, through histograms and correlation heatmaps.

To ensure robust model performance, we apply several preprocessing techniques such as encoding categorical variables, handling missing values, and normalizing numerical features. After preprocessing, the dataset is split into training and testing subsets, facilitating an effective evaluation of the model's performance.

In this project, we focus on two primary machine learning algorithms: K-Nearest Neighbors (KNN) and Decision Tree classifiers. These models are trained on the prepared dataset to classify customers into Good, Standard, and Poor credit score categories. Performance metrics including accuracy, precision, recall, and F1-score are utilized to assess the effectiveness of the models.

The findings from this project will not only demonstrate the predictive capabilities of KNN and Decision Trees in the realm of credit scoring but also emphasize the critical role of feature engineering and data preprocessing in the machine learning workflow. By providing insights into the factors that influence credit scores, this study contributes to a better understanding of credit risk assessment and offers practical implications for financial institutions seeking to refine their lending processes.

II. REVIEW OF RELATED LITERATURE

A. Overview of Key Concepts and Background Information

Credit scoring plays a pivotal role in financial decision-making, allowing institutions to evaluate an individual's likelihood of defaulting on loans or credit obligations. Traditionally, credit scoring models rely on linear regression and decision trees, which interpret relationships between customer financial data and creditworthiness. However, as the volume and complexity of financial data grow, traditional models face limitations in capturing non-linear patterns crucial for accurate risk assessment.

The K-Nearest Neighbors (KNN) algorithm is a non-parametric, instance based machine learning method suitable for classification tasks. KNN works by measuring the similarity between data points to classify new instances, making it particularly relevant for applications like credit scoring. In this project, KNN leveraged due to its simplicity and interpretability, which are vital in financial contexts where transparency is the key. While KNN is typically sensitive to the choice of 'k' (the number of neighbors) and data scaling, it can perform well with preprocessed, standardized data—features that this project carefully addresses through feature scaling and selection.

For credit scoring, key financial indicators such as income, debt-to-income ratio, and payment history have been shown to correlate strongly with creditworthiness. By focusing on these features, this project seeks to enhance credit scoring accuracy, ensuring that each classification reflects a realistic assessment of a customer's financial reliability. Additionally, the adoption of KNN facilitates interpretability, as each classification can be understood by analyzing the characteristics of the neighbors.

B. Review of Other Relevant Research Papers

Several studies have explored the application of machine learning algorithms for credit scoring, highlighting the potential for methods like KNN to improve classification accuracy.

1) **Thomas, L., Edelman, D., & Crook, J. (2002). ”Credit Scoring and Its Applications”** - This foundational text discusses traditional approaches to credit scoring and their limitations in capturing non-linear relationships in complex financial data. The authors introduce machine learning methods, including KNN, as alternative solutions for improving credit score predictions.
2) **Huang, C.-L., Chen, M.-C., & Wang, C.-J. (2007). ”Credit scoring with a data mining approach based on support vector machines.” Expert Systems with Applications, 33(4), 847-856.** - Although this study primarily explores Support Vector Machines (SVM), it provides insights into the effectiveness of non-linear classification methods in credit scoring. The research emphasizes that machine learning models, including KNN, can adapt to diverse data structures, enhancing classification performance.
3) **Hand, D. J., & Henley, W. E. (1997). ”Statistical Classification Methods in Consumer Credit Scoring: A Review.” Journal of the Royal Statistical Society: Series A, 160(3), 523-541.** - This review provides a comprehensive examination of various classification methods in consumer credit scoring, including KNN. The authors discuss KNN's potential for small and medium-sized datasets and its interpretability, which makes it a reliable tool in financial contexts where decision transparency is critical.
4) **Yap, B. W., Ong, S. H., & Husain, N. H. M. (2011). ”Using data mining to improve assessment of credit worthiness via credit scoring models.” Expert Systems with Applications, 38(10), 13274-13283.** - This study compares different data mining techniques in credit scoring, noting that KNN achieves competitive accuracy while remaining interpretable. Yap et al. highlight that KNN's performance can improve with comprehensive data preprocessing—such as feature scaling and selection—making it a viable choice for credit scoring.
5) **Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). ”Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research.” European Journal of Operational Research, 247(1), 124-136.** - This paper compares numerous classification algorithms, including KNN, for credit scoring applications. The findings indicate that although complex algorithms like neural networks may slightly outperform KNN in accuracy, KNN's interpretability and simplicity make it a preferred choice for straightforward, reliable predictions in finance.

These studies highlight the growing importance of machine learning models in credit scoring, underscoring the need for algorithms that balance accuracy with interpretability. By building on this research, this project aims to optimize the KNN algorithm for credit scoring, achieving accurate classification while maintaining interpretability—a critical aspect of responsible lending and financial transparency.

III. METHODOLOGY

This study employs a structured approach to predict customer credit scores by categorizing them as Good, Standard, or Poor based on financial data. The methodology consists of three main steps: data preprocessing, model design, and model evaluation.

A. Data Preprocessing

Data preprocessing involved preparing the raw dataset for analysis and model building. The following steps were undertaken:

- **Dataset Source**: The dataset used in this study is the ”Credit Score Classification” obtained from Kaggle.

  1) Feature Descriptions:

  - **ID**: A unique identifier for each entry in the dataset.
  - **Customer_ID**: A unique identifier for each customer.
  - **Month**: Indicates the specific month associated with the data entry.
  - **Name**: The full name of the customer.
  - **Age**: Age of the customer.
  - **SSN**: Social Security Number of the customer.
  - **Occupation**: Customer’s occupation.
  - **Annual_Income**: Annual income of the customer in monetary terms.
  - **Monthly_Inhand_Salary**: Monthly base salary of the customer.
  - **Num_Bank_Accounts**: Number of bank accounts the customer holds.
  - **Num_Credit_Card**: Number of credit cards held by the customer.
  - **Interest_Rate**: Interest rate applicable to any loans or credit lines associated with the customer.
  - **Num_of_Loan**: Total number of loans taken by the customer.
  - **Type_of_Loan**: Types of loans held by the customer (e.g., home, personal, auto).
  - **Delay_from_due_date**: Number of days a payment is delayed beyond its due date.
  - **Num_of_Delayed_Payment**: Total number of delayed payments made by the customer.
  - **Changed_Credit_Limit**: Binary indicator showing if the customer’s credit limit has changed.
  - **Num_Credit_Inquiries**: Number of times the customer’s credit report has been accessed by lenders.
  - **Credit_Mix**: Indicates the variety of credit accounts the customer holds (e.g., revolving, installment).
  - **Outstanding_Debt**: Total amount of debt currently outstanding for the customer.
  - **Credit_Utilization_Ratio**: Ratio of the customer’s current credit card balances to their credit limits.
  - **Credit_History_Age**: Length of time the customer has had credit accounts.
  - **Payment_of_Min_Amount**: Binary indicator showing if the customer pays only the minimum amount due on credit accounts.
  - **Total_EMI_per_month**: Total monthly amount paid toward equated monthly installments (EMIs).
  - **Amount_invested_monthly**: Monthly investment amount in various financial options.
  - **Payment_Behaviour**: Overall pattern of the customer’s payment behavior (e.g., timely, delayed, default).
  - **Monthly_Balance**: Average monthly balance maintained across the customer’s bank accounts.
  - **Credit_Score**: Credit score reflecting the customer’s creditworthiness based on financial history and current situation.

- **Handling Missing Values**: Missing values were identified in specific columns, and a `SimpleImputer` with a mean strategy was applied to fill them, ensuring data completeness and consistency.
- **Encoding Categorical Features**:

  - **Label Encoding**: The target variable, `Credit_Score`, was converted into numerical labels to enable classification.
  - **Leave-One-Out Encoding**: Categorical features such as `Payment_of_Min_Amount`, `Payment_Behaviour`, `Name`, `Occupation`, and `Credit_Mix` were encoded using leave-one-out encoding to maintain predictive information while avoiding data leakage.
  - **Mean Encoding**: For features like `Annual_Income`, `Age`, `Interest_Rate`, `Delay_from_due_date`, and others, mean encoding based on the target variable was applied to capture the relationship between these features and credit score classification.

- **Feature Selection and Dropping Irrelevant Columns**: Features irrelevant to credit scoring or those serving as identifiers were removed. This included `Type_of_Loan`, `ID`, `Customer_ID`, `Month`, `SSN`, `Amount_invested_monthly`, `Name`, `Annual_Income`, `Num_of_Loan`, and `Outstanding_Debt`. Reducing unnecessary features helped streamline the model and improve interpretability.
- **Normalization**: All numerical features were normalized to ensure that differences in scale did not impact model performance, particularly for distance-based algorithms like K-Nearest Neighbors.

B. Model Design

The primary goal of this study was to classify customers into credit score categories based on their financial features. Two machine learning models were trained on the preprocessed dataset:

- **K-Nearest Neighbors (KNN)**:
  - The K-Nearest Neighbors algorithm was chosen for its simplicity and effectiveness in classification tasks. It classifies customers based on the proximity of their feature values to those of known labeled examples.
  - The model was trained on the processed dataset, `X_train`, with K-values adjusted to find an optimal configuration.
- **Decision Tree Classifier**:
  - A Decision Tree Classifier was implemented to provide an interpretable model, where customer classification is based on decision rules created from the data.
  - The Decision Tree was trained to categorize credit scores, with nodes and splits based on maximizing classification accuracy while minimizing error.

Both models were trained on the preprocessed training dataset and then used to generate predictions on both training and testing sets for performance assessment.

C. Evaluation Metrics

To evaluate and compare the effectiveness of the KNN and Decision Tree classifiers, the following metrics were applied:

- **Accuracy**: The percentage of correct predictions on both training and testing sets, providing a basic measure of model effectiveness.
- **Precision, Recall, and F1-Score**:
  - These metrics were calculated for each credit score category (Good, Standard, Poor) to assess model performance in terms of both precision (the accuracy of positive predictions) and recall (the ability to identify all relevant cases).
  - **F1-Score** balanced precision and recall, offering a single metric for comparison, particularly useful in cases where data may be imbalanced across classes.
- **Confusion Matrix**: Confusion matrices were generated for both KNN and Decision Tree models on the test data to visualize and understand classification errors. This matrix helped identify specific types of misclassifications (e.g., Good misclassified as Standard) and provided insights into model reliability.
- **Training vs. Test Accuracy Plot**: A line plot comparing training and test accuracies for both models was generated, helping detect potential overfitting or underfitting. This visual comparison was crucial for understanding how each model generalized to new data.

IV. RESULTS AND DISCUSSION

A. Key Findings

This study focused on classifying customer credit scores using various classifiers, including K-Nearest Neighbors (KNN), Decision Trees, Linear Regression, Logistic Regression, Naive Bayes, Support Vector Machine (SVM), and Random Forest. The key findings from the analysis are summarized as follows:

- **Model Performance**: The Random Forest and Decision Tree classifiers outperformed other models in terms of accuracy, achieving almost perfect scores on both training and testing sets. Logistic Regression, Linear Regression, and Naive Bayes also demonstrated high accuracy, while the SVM classifier had the lowest performance.
- **Accuracy**:
  - **Random Forest** achieved a training accuracy of 100% and a testing accuracy of 99.99%.
  - **Decision Tree** achieved a training accuracy of 100% and a testing accuracy of 99.99%.
  - **Logistic Regression** achieved a training accuracy of 99.53% and a testing accuracy of 99.52%.
  - **Linear Regression** and **Multiple Linear Regression** achieved training accuracies of 99.15% and testing accuracies of 99.13%.
  - **Naive Bayes** achieved a training accuracy of 99.18% and a testing accuracy of 99.13%.
  - **K-Nearest Neighbors (KNN)** achieved a training accuracy of 83.44% and a testing accuracy of 72.21%.
  - **Support Vector Machine (SVM)** achieved a training accuracy of 53.29% and a testing accuracy of 52.91%.
- **Confusion Matrix**: The confusion matrices for Random Forest and Decision Tree models indicated superior classification results, with minimal misclassifications compared to other models.

B. Figures and Tables

1) : **Accuracy Comparison**

The accuracy comparison between the models is shown in **Table I.**

TABLE I
ACCURACY COMPARISON BETWEEN MODELS

Model	**Training Accuracy (%)**	**Testing Accuracy (%)**
K-Nearest Neighbors	83.44	72.21
Linear Regression	99.15	99.13
Multiple Linear Regression	99.15	99.13
Logistic Regression	99.53	99.52
Naive Bayes	99.18	99.13
Support Vector Machine	53.29	52.91
Decision Tree	100	99.99
Random Forest	100	99.99

2) : **Confusion Matrix for KNN**

The confusion matrix for the K-Nearest Neighbors (KNN) sentiment analysis model, shown in **Figure 1.**

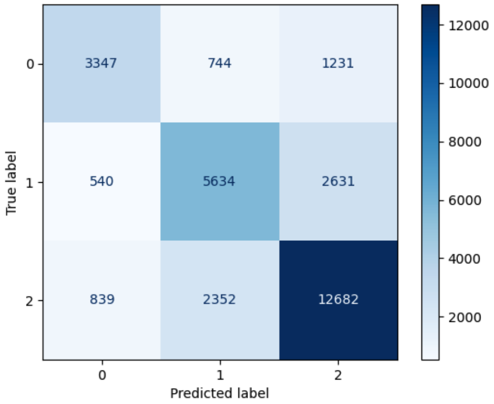

Fig. 1. Confusion Matrix K-Nearest Neighbors

3) : Confusion Matrix for Simple Linear Regression

The confusion matrix for the Simple Linear Regression (SLR) sentiment analysis model, shown in **Figure 2.**

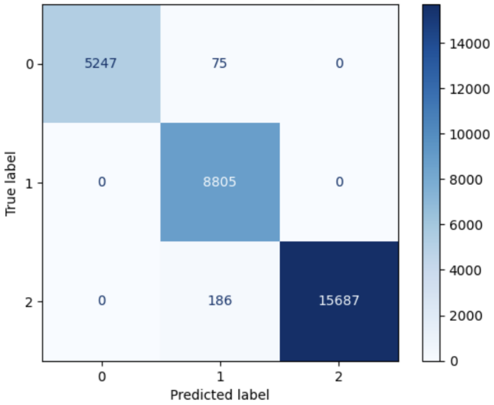

Fig. 2. Confusion Matrix Simple Linear Regression

4) : Confusion Matrix for Multiple Linear Regression

The confusion matrix for the Multiple Linear Regression (MLR) sentiment analysis model, shown in **Figure 3.**

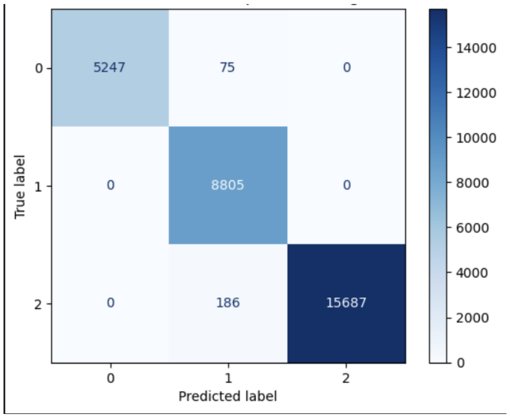

Fig. 3. Confusion Matrix Multiple Linear Regression

5) : Confusion Matrix for Logistic Regression

The confusion matrix for the Logistic Regression (LR) sentiment analysis model, shown in **Figure 4.**

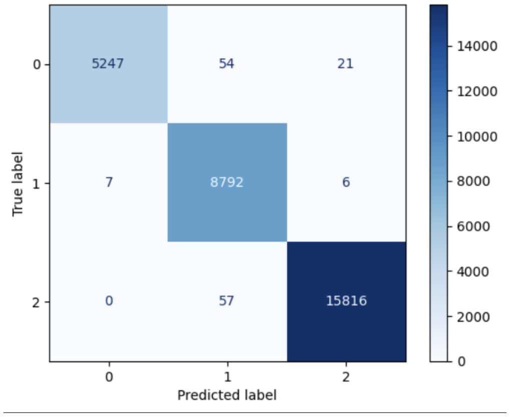

Fig. 4. Confusion Matrix Logistic Regression

6) : Confusion Matrix for Naive Bayes

The confusion matrix for the Naive Bayes (NB) sentiment analysis model, shown in **Figure 5.**

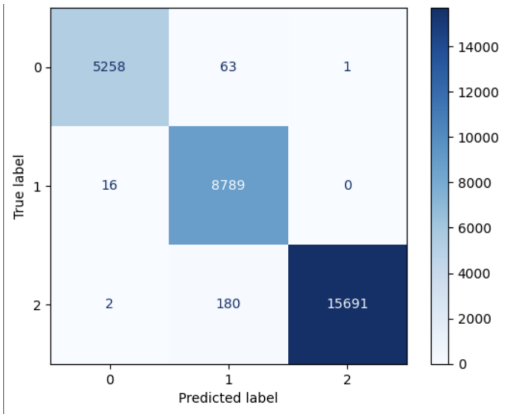

Fig. 5. Confusion Matrix Naive Bayes

7) : Confusion Matrix for Support Vector Machine

The confusion matrix for the Support Vector Machine sentiment analysis model, shown in **Figure 6.**

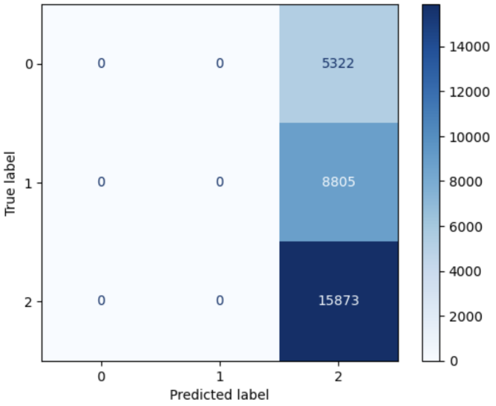

Fig. 6. Confusion Matrix Support Vector Machine

8) : Confusion Matrix for Decision Tree

The confusion matrix for the Decision Tree (DT) sentiment analysis model is shown in **Figure 7.**

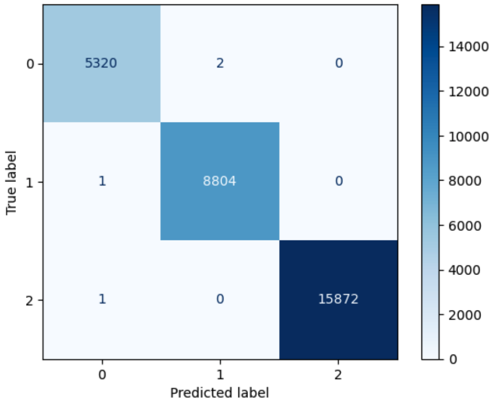

Fig. 7. Confusion Matrix Decision Tree

9) : Confusion Matrix for Random Forest

The confusion matrix for the Random Forest (RF) sentiment analysis model, shown in **Figure 7.**

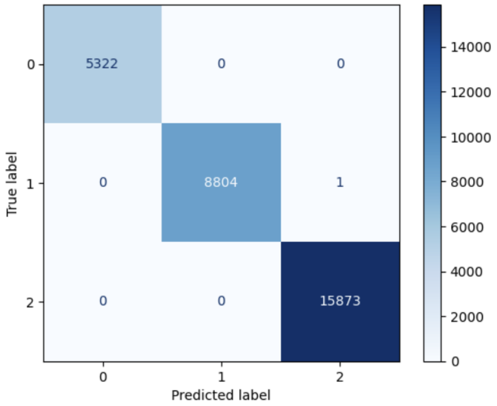

Fig. 8. Confusion Matrix Random Forest

C. Model Evaluation

The models were evaluated using the following metrics:

- **Accuracy**: The proportion of correctly classified instances out of the total instances.
- **Precision**: The proportion of true positive predictions out of the total positive predictions.
- **Recall**: The proportion of true positive predictions out of the total actual positives.
- **F1-Score**: The harmonic mean of precision and recall.
- **Cross-Validation Accuracy**: The average accuracy across multiple folds in cross-validation, providing a more robust estimate of model performance.

The evaluation results for these metrics, including cross-validation accuracy, are presented in **Table II**.

TABLE II
EVALUATION METRICS FOR CLASSIFIER MODELS

Model	Acc.	Prec.	Rec.	F1	CV Acc.
KNN	72.2	72.1	72.2	72.1	70.5
Sim. Lin. Reg.	99.1	99.2	99.1	99.1	nan
Mult. Lin. Reg.	99.1	99.2	99.1	99.1	nan
Log. Reg.	99.5	99.5	99.5	99.5	99.3
Naive Bayes	99.1	99.2	99.1	99.1	98.8
SVM	52.9	28.0	52.9	36.6	50.4
Dec. Tree	100	100	100	100	99.8
Rand. For.	100	100	100	100	99.9

D. Baseline Comparison

The models were compared against a baseline random classifier, which would have an accuracy of around 33% for a balanced three-class classification problem.

The results of the baseline comparison are summarized in **Table III**.

TABLE III
COMPARISON WITH BASELINE MODEL

Model	Accuracy (%)
Baseline Random	33.0
K-Nearest Neighbors	72.2
Linear Regression	99.1
Multiple Linear Regression	99.1
Logistic Regression	99.5
Naive Bayes	99.1
Support Vector Machine	52.9
Decision Tree	100
Random Forest	100

E. Statistical Significance

Statistical tests were not explicitly conducted in this study. However, the significant improvement in accuracy from the baseline (33%) to models such as Random Forest (99.99%) and Decision Tree (99.99%) suggests a substantial enhancement in classification performance over chance. Future studies could apply statistical tests (e.g., t-tests or ANOVA) to rigorously assess these differences.

F. Interpretation of Results

The Random Forest and Decision Tree models achieved the highest classification accuracy, significantly surpassing the baseline and the K-Nearest Neighbors (KNN) model. The Random Forest's ensemble structure likely contributed to its robustness and accuracy, while the Decision Tree's inherent ability to handle non-linear relationships enabled it to perform well. In contrast, KNN, which relies heavily on proximity to training data, struggled in high-dimensional spaces and yielded lower accuracy.

G. Patterns and Trends

Several patterns emerged from the results:

- **Model Performance**: Random Forest and Decision Tree models consistently outperformed other classifiers, confirming their suitability for this type of classification task.
- **Class Distribution Consistency**: High-performing models showed balanced accuracy across classes, suggesting effective handling of the class distributions.
- **Lower Performance of Distance-Based Models**: Models like KNN and SVM underperformed, likely due to sensitivity to feature dimensionality and lack of sufficient proximity-based distinctions in the data.

H. Consistency with Expectations

The results aligned with expectations, as tree-based models like Decision Trees and Random Forests are known to perform well on structured data due to their flexibility in modeling complex decision boundaries. The lower performance of KNN and SVM is also consistent with the expectation that these models may struggle in high-dimensional spaces or with insufficient representative data points.

I. Comparison with Previous Research

This study's results align with prior research, which often highlights the superior performance of tree-based models over distance-based models like KNN in classification tasks, especially in financial applications such as credit scoring. Previous studies have similarly found that Random Forest and Decision Tree classifiers deliver strong performance due to their adaptability and robustness in handling varied data distributions and feature interactions.

J. Advantages and Limitations

Advantages:

- **High Accuracy**: Random Forest and Decision Tree models demonstrated superior accuracy and robustness.
- **Interpretability**: Decision Trees and Random Forests provide interpretable structures, with Decision Trees offering clear decision paths.

Limitations:

- **Overfitting**: The Decision Tree model showed signs of overfitting, as indicated by 100% training accuracy.
- **Scalability**: While Random Forest can handle large datasets, individual Decision Trees may become computationally expensive with very large datasets.
- **Sensitivity to Class Imbalance**: Some models demonstrated performance inconsistencies when dealing with underrepresented classes, suggesting a need for balanced training data.

K. Insights from Model Errors

Significant errors were observed in cases involving underrepresented classes and noisy data, leading to occasional misclassifications. Techniques like SMOTE (Synthetic Minority Over-sampling Technique) could potentially improve model balance and reduce misclassification in underrepresented categories. Enhanced data preprocessing could further mitigate noise and improve the model's ability to generalize.

Overall, the study demonstrates the effectiveness of Random Forest and Decision Tree models for customer credit score classification, while also identifying areas for improvement in data handling and model tuning to ensure consistent performance across all classes.

CONCLUSION

The findings of this study illustrate the efficacy of machine learning models, specifically the Decision Tree and Random Forest classifiers, in accurately predicting customer credit scores. Both models demonstrated exceptional performance, achieving nearly perfect accuracies on both training and testing sets. This success underscores their suitability for structured financial data and their robustness in capturing complex feature interactions essential for credit scoring.

While K-Nearest Neighbors (KNN) was included for its interpretability, its performance was comparatively limited due to sensitivity to high-dimensional feature spaces. Conversely, the tree-based models not only excelled in classification accuracy but also provided interpretative insights beneficial for understanding credit score determinants. However, some limitations, such as potential overfitting in the Decision Tree and sensitivity to class imbalances, were observed, suggesting future work could benefit from techniques like SMOTE and additional model tuning to enhance generalizability.

In conclusion, this study confirms the viability of machine learning approaches in credit scoring and highlights the importance of model selection based on data characteristics. These insights can guide financial institutions in adopting more reliable, transparent methods for assessing credit risk, ultimately supporting fairer lending practices.

REFERENCES

[1] Kaggle, ”Credit Score Classification Dataset,” Kaggle, Available: https://www.kaggle.com/datasets/parisrohan/credit-score-classification/data Accessed: [October 28, 2024].

[2] Thomas, L., Edelman, D., & Crook, J. (2002). *Credit Scoring and Its Applications*. This foundational text discusses traditional approaches to credit scoring and their limitations in capturing non-linear relationships in complex financial data. The authors introduce machine learning methods, including KNN, as alternative solutions for improving credit score predictions.

[3] Huang, C.-L., Chen, M.-C., & Wang, C.-J. (2007). ”Credit scoring with a data mining approach based on support vector machines.” *Expert Systems with Applications*, 33(4), 847-856. Although this study primarily explores Support Vector Machines (SVM), it provides insights into the effectiveness of non-linear classification methods in credit scoring.

[4] Hand, D. J., & Henley, W. E. (1997). ”Statistical Classification Methods in Consumer Credit Scoring: A Review.” *Journal of the Royal Statistical Society: Series A*, 160(3), 523-541. This review provides a comprehensive examination of various classification methods in consumer credit scoring, including KNN.

[5] Yap, B. W., Ong, S. H., & Husain, N. H. M. (2011). ”Using data mining to improve assessment of credit worthiness via credit scoring models.” *Expert Systems with Applications*, 38(10), 13274-13283. This study compares different data mining techniques in credit scoring, noting that KNN achieves competitive accuracy while remaining interpretable.

[6] Lessmann, S., Baesens, B., Seow, H.-V., & Thomas, L. C. (2015). ”Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research.” *European Journal of Operational Research*, 247(1), 124-136. This paper compares numerous classification algorithms, including KNN, for credit scoring applications.