University of Hertfordshire UH

School of Physics, Engineering and Computer Science

MSc Data Science Project
7PAM2002-0901-2024
Department of Physics, Astronomy and Mathematics

Data Science FINAL PROJECT REPORT

Project Title:

Predicting Loan Approval Status Using Machine Learning Models

Student Name and SRN:

Shiva kumar Sundara Murthy

And **22024927**

GIT Link: https://github.com/7shivakumar/7PAM2002-0901-2024_Msc-Project.git

Supervisor: Peter Scicluna

Date Submitted:06/01/2025

Word Count: 7465

Declaration

By acknowledging this declaration, I, Shiva, affirm that the research titled “Predicting Loan Approval Status Using Machine Learning Models: A Comparative Analysis of Random Forest, Gradient Boosting, and Neural Networks” represents my independent academic effort and analysis. My studies and efforts served as the foundation for the research and analysis in this report, and all the information, techniques, and conclusions stated are the outcome of my independent investigation.

I additionally affirm that this report has not been submitted for consideration towards any other degree or certification, nor has it been previously published or plagiarised in any way. All external sources, such as academic papers, databases, and internet sites, have been appropriately cited and referenced throughout the report.

I acknowledge that I am solely responsible for the information in this report and attest that the study was carried out in compliance with academic honesty and research ethics.

Acknowledgement

I want to sincerely thank everyone who helped to make this project a success and for their support.

Above all, I would want to show my profound gratitude to Dr Ladislav Vegh, whose work on machine learning classification models for loan approval prediction inspired and informed me throughout the project. His comparison of several machine learning approaches was crucial in determining the course of this investigation and deepening my comprehension of the topic. His enthusiasm and assistance have been essential in advancing our study.

I want to thank my family, friends, and coworkers since their understanding, tolerance, and continuous support helped me stay motivated and focused throughout this trip.

Lastly, I want to thank many of the several data science and machine learning communities whose open-source tools, forums, and contributions enabled the technical parts of this effort. These societies' abundance of knowledge has been a tremendous advantage.

This project would not have been feasible without everyone's combined work and contributions. I appreciate your support, everyone.

Abstract

This project explores the development of a machine learning-based system to predict loan approval decisions, addressing the research question: “How do neural networks, gradient boosting, and random forest algorithms compare in predicting loan approval status based on customer financial and demographic data, and which model provides the most accurate results for automated decision-making?” The primary objectives include understanding the key factors influencing loan approvals, preprocessing, and analysing the dataset to ensure it is suitable for modelling, and evaluating various machine learning models to identify the optimal approach. The dataset comprises critical features such as income, loan amount, credit score, education level, employment status, asset values, and loan term. Extensive preprocessing, including cleaning, encoding categorical variables, and scaling numerical data, was conducted to prepare the data for modelling. Multiple machine-learning models, including Neural Networks, Gradient Boosting, and Random Forest, were trained, and evaluated using metrics like accuracy, precision, recall, F1-score, and confusion matrices. With an accuracy rate of 97.78%, Random Forest demonstrated superior performance, accompanied by commendable precision and recall metrics, outperforming both Neural Networks and Gradient Boosting. The study also highlights the importance of credit scores, income levels, and loan amounts as key predictors of loan approval status. Automation in loan decisions offers opportunities to improve both operational efficiency and fairness within financial organizations. The findings underscore the robustness of Random Forest as a tool for predicting loan approvals and provide valuable insights into the role of machine learning in economic decision-making. The results suggest that this model can be a reliable component in automating loan approval processes, reducing human bias, and improving operational efficiency in the banking sector.

Contents:

List of Figures

1. Introduction

Approving loans plays a pivotal role in financial services, impacting credit allocation for individuals and businesses. Historically, these decisions relied heavily on manual evaluation by human personnel. where loan officers evaluate a borrower's financial standing, creditworthiness, and risk based on documents and reports. However, as the volume of loan applications increases and financial institutions strive for operational efficiency, reliance on traditional methods has become unsustainable. The manual approach is not only time-consuming but also susceptible to human errors, biases, and inconsistencies. In an era where technological advancements are reshaping industries, machine learning presents an opportunity to revolutionize the loan approval process by automating decisions with greater accuracy and efficiency (Vegh, 2023).

Loan approval prediction using machine learning involves analysing customer financial and demographic data to classify applications as "approved" or "rejected." Machine learning algorithms can identify subtle patterns and relationships in the data that are often overlooked in manual evaluations. By incorporating machine learning, financial institutions can streamline operations, enhance decision-making transparency, and improve customer satisfaction through faster application processing times. This project focuses on building a machine-learning system to automate the loan approval process, thereby reducing dependency on human judgment and increasing reliability (Vegh, 2023).

Currently, many financial institutions employ rule-based systems or scoring models such as FICO scores to evaluate loan applications. While these methods provide some level of automation, they cannot adapt to changing trends in customer behaviour or leverage the full potential of data. Machine learning models, on the other hand, can dynamically learn from historical data, adapt to new patterns, and provide insights into critical factors influencing approval decisions. The application of machine learning in this domain is not only innovative but also aligns with the industry's broader digital transformation efforts (FJS, 2024).

1.1 Research Question

The primary research question guiding this project is:
"How do neural networks, gradient boosting, and random forest algorithms compare in predicting loan approval status based on customer financial and demographic data, and which model provides the highest accuracy for automating loan decisions?"
This question seeks to evaluate the comparative performance of three machine learning models and identify the most effective approach for implementing an automated loan approval system.

1.2 Aims

The overarching aim of this project is to develop a machine learning model that can accurately predict loan approval status, providing financial institutions with a reliable, efficient, and scalable solution. The model is expected to minimize errors and inconsistencies, thereby improving decision-making processes and enhancing customer experience. (Vegh, 2023)

1.3 Objectives

To achieve this aim, the following objectives are established:

1. **Data Analysis and Exploration**: Investigate the dataset to understand the distribution and significance of features such as income, loan amount, credit score, education level, employment status, and other relevant factors influencing loan approvals (Vegh, 2023).
2. **Data Preprocessing**: Perform data cleaning, handle missing values, encode categorical variables, and scale numerical features to ensure the dataset is prepared for machine learning model training (Vegh, 2023).
3. **Feature Engineering**: Identify and construct features that provide the most predictive power for the target variable (loan approval status) (FJS, 2024).
4. **Model Development**: Train and test three machine learning algorithms—Neural Networks, Gradient Boosting, and Random Forest—on the prepared dataset to develop predictive models (Vegh, 2023).
5. **Model Evaluation**: Assess the models using key performance metrics such as accuracy, precision, recall, F1-score, and confusion matrices to compare their effectiveness (Vegh, 2023).
6. **Comparison and Selection**: Compare the performance of the models to identify the best-performing algorithm for practical implementation (FJS, 2024.
7. **Insights and Recommendations**: Provide actionable insights into the key factors influencing loan approvals and recommend the most effective model for adoption by financial institutions (Vegh, 2023).
8. **System Deployment**: Develop a user-friendly API for integrating the selected model into financial workflows, enabling seamless deployment in real-world scenarios (FJS, 2024.

In the financial industry, machine learning is increasingly being adopted to tackle challenges such as fraud detection, customer segmentation, risk assessment, and credit scoring. Companies like LendingClub and ZestFinance have pioneered the use of AI in credit decisions, offering tailored loans and risk predictions with greater accuracy than traditional methods. However, these systems often face scrutiny regarding transparency, fairness, and bias. This project aims to contribute to the industry by developing a robust machine-learning model that balances predictive accuracy with fairness, ensuring that all decisions are explainable and unbiased (Vegh, 2023).

By leveraging customer data such as income, credit score, and loan amount, this project seeks to enhance the efficiency of loan approval systems. The proposed approach provides financial institutions with an opportunity to transition from traditional rule-based systems to data-driven decision-making models. The findings of this project will not only demonstrate the potential of machine learning in automating loan approvals but also highlight the importance of ethical considerations in deploying AI in sensitive domains like finance (FJS, 2024).

Through rigorous analysis, model evaluation, and real-world applicability, this project aspires to bridge the gap between theoretical research and industry needs, making loan approval prediction more reliable, scalable, and aligned with the demands of the digital economy (Vegh, 2023; FJS, 2024).

2. Literature Review

In recent years, machine learning has become increasingly popular in financial decision-making, especially in predicting loan acceptance. An overview of the use of machine learning models to forecast loan acceptance is given in this review of the literature, with particular attention on well-liked methods including Random Forest, Gradient Boosting, and Neural Networks. This review identifies the advantages, disadvantages, and uses of these models in loan prediction tasks by examining several research initiatives.

2.1 Using Machine Learning to Make Financial Decisions:

Machine learning techniques are being used more and more by financial organisations to streamline and automate loan approval procedures. Credit scoring and loan approvals were formerly done by hand using applicant profiles, risk assessments, and historical data. But as machine learning has advanced, more complex models have appeared to increase precision and effectiveness. By decreasing human bias, speeding up processing, and boosting prediction accuracy, these algorithms have the potential to greatly improve decision-making processes (Vegh, 2023).

To assess how well various machine learning algorithms—such as Random Forest, Gradient Boosting, and Neural Networks—predict loan approvals, Vegh (2023) carried out a comparative analysis of them. Based on a loan prediction dataset, this study analyses the effectiveness of several models in predicting loan approval and shows how they can be used to tackle classification tasks.

2.2 Random Forest for Loan Approval Prediction:

Due to its ensemble learning nature, Random Forest has shown to be the most successful machine learning algorithm for predicting loan acceptance. To produce a final result, Random Forest builds several decision trees and aggregates their forecasts, as noted by Vegh (2023). In datasets with high complexity or noise, this technique helps reduce overfitting. Random Forest is perfect for financial applications, where data complexity is frequently high because it can handle mixed data types (both categorical and numerical), according to multiple studies, and it is strong against overfitting (Breiman, 2001; Vegh, 2023).

The strength of Random Forest is its feature importance measure, which aids in determining the factors that have the greatest bearing on loan acceptance choices. Features including applicant income, credit history, and loan size are crucial in deciding approval in a loan prediction scenario, claims Vegh (2023). Financial organisations can comprehend the elements influencing loan acceptance decisions thanks to the model's feature ranking capability, which also helps to guarantee that decision-making procedures are open and compliant with legal criteria (Liaw & Wiener, 2002).

2.3 Gradient Boosting and Loan Prediction:

Another ensemble technique, gradient boosting, has also shown encouraging outcomes in financial decision-making tasks, such as predicting loan acceptance. Gradient Boosting constructs decision trees in a sequential manner, with each tree attempting to fix the mistakes of the one before it, in contrast to Random Forest, which combines predictions from several trees concurrently. The overall prediction accuracy is increased by this step-by-step learning procedure. According to Vegh (2023), gradient boosting can be quite successful with small to medium-sized datasets and provide a higher level of accuracy than simpler models.

Gradient Boosting is excellent at accuracy, but if the model is not adjusted correctly, it can occasionally suffer from overfitting, as Liaw and Wiener (2002) point out. Gradient Boosting demonstrated perfect precision in loan prediction tasks, but it also had lower recall values, which means it missed a few actual positive examples, according to Vegh's (2023) study. This trade-off emphasises the necessity of meticulous parameter tuning to maximise efficiency and prevent notable recall loss, especially in financial applications where missed opportunities may result from a failure to identify qualified loan applicants.

2.4 Predicting Loan Approval with Neural Networks:

Another machine learning method frequently used in loan prediction models is neural networks (NN), especially when dealing with intricate, non-linear interactions in the data. Large datasets with complex patterns have shown promise when handled by neural networks, especially deep learning models. The application of a Multi-Layer Perceptron (MLP) classifier in a loan approval prediction problem is explained by Vegh (2023). The MLP is an artificial neural network model that learns and maps the link between input variables and the goal variable (loan approval status) using layers of interconnected neurones.

According to the study by Vegh (2023), neural networks need rigorous preprocessing and hyperparameter optimisation even if they can describe intricate, non-linear interactions. While the model's accuracy and F1-score were good, its precision was marginally worse than Random Forest's. Neural networks' sensitivity to data scaling and the requirement for thorough tuning to avoid overfitting are the causes of this performance variance (Bishop, 2006). Additionally, compared to other models, neural networks are more difficult to interpret, which can be a major disadvantage in financial applications where accountability and transparency are crucial (Vegh, 2023).

2.5 Comparison of Models in Loan Approval Prediction:

The effectiveness of several machine learning algorithms for predicting loan approval has been compared in several studies. For instance, Random Forest performed better than both Gradient Boosting and Neural Networks in Vegh's (2023) study in terms of accuracy, precision, recall, and F1-score. According to the study, Random Forest was the most accurate and dependable model for loan prediction tasks because of its capacity to handle both numerical and categorical information and its resistance to overfitting. A balanced method of forecasting both authorised and denied loans was also made possible by Random Forest's excellent precision and recall (Breiman, 2001).

Gradient Boosting, on the other hand, had a lower recall value even if it achieved 100% precision, which can result in missed opportunities for qualified applicants, according to Vegh (2023). In a similar vein, neural networks did well but were more likely to overfit if hyperparameters and tuning were not done carefully. These results align with earlier studies that demonstrate Random Forest's resilience and adeptness in striking a favourable balance between false positives and false negatives in financial prediction tasks (Liaw & Wiener, 2002).

In conclusion, a promising field of study and application is the use of machine learning to forecast loan acceptance decisions. Neural networks, Random Forest, and Gradient Boosting each have special benefits and limitations. The most dependable model for loan prediction is

Random Forest, which successfully strikes a balance between performance criteria including recall, accuracy, and precision. Neural networks and gradient boosting both provide insightful information, but they must be carefully adjusted and include trade-offs that may limit their usefulness in actual financial decision-making situations. Future studies should concentrate on extending datasets to incorporate real-world loan data, investigating cutting-edge methods like ensemble stacking, and enhancing model interpretability, as Vegh (2023) points out. This will guarantee the wider adoption of machine learning models in financial institutions for automated, fair, and transparent loan approval processes.

3. Dataset

The Dataset is released under MIT license The dataset utilized for this project was sourced from Kaggle and is titled **"Loan Approval Prediction Dataset"** (accessible at https://www.kaggle.com/datasets/architsharma01/loan-approval-prediction-dataset). This dataset simulates real-world loan approval scenarios encountered by financial institutions and includes key financial, demographic, and loan-related attributes. It is publicly available for research and educational purposes, making it ideal for this study. The dataset is structured and well-documented, providing a reliable foundation for analysing and modelling loan approval decisions Vegh (2023).

3.1 Origin of the Dataset

The dataset was designed to replicate the parameters and decision-making processes used in real-world banking systems. Although simulated, it is reflective of industry practices. It was created by financial analysts and data science practitioners to model loan approvals, likely using representative data patterns from countries like India or the United States. The dataset does not specify the exact time or location of the collection but includes contemporary features relevant to loan approval processes. It was constructed to aid machine learning researchers and students in developing predictive models for credit risk assessment and decision-making (FJS, 2024). This aligns with the growing trend of leveraging data for better financial decision-making, as discussed by Vegh (2023). The dataset has an MIT license which means it grants permission to use and modify the data.

3.2 Description of the Dataset

The dataset consists of 614 records and 13 features, including the target variable, loan_status, which indicates whether a loan application was approved or rejected. Key features include ApplicantIncome, LoanAmount, Credit_History, Education, and Property_Area. These features capture the financial stability, creditworthiness, and demographic characteristics of applicants. For example, Credit_History reflects the borrower's past credit behaviour, while LoanAmount specifies the requested loan amount. The target variable is binary, with "Y" for approved loans and "N" for rejected ones. This dataset was chosen because it comprehensively represents the factors influencing loan approvals, aligning with the objectives of this study (Vegh, 2023).

3.3 Reasons for Choosing the Dataset

This dataset was selected for its relevance to the research question and its ability to provide insights into loan approval processes. Its structured nature, inclusion of diverse attributes, and clear labels make it suitable for applying and comparing machine learning models. The dataset aligns well with the project's objectives of automating loan approvals, understanding influential factors, and comparing algorithmic performance. Additionally, its availability on Kaggle ensures accessibility and reproducibility for academic research (FJS, 2024). As highlighted by Vegh (2023), the diverse features of the dataset make it ideal for building predictive models that can aid in financial decision-making.

3.4 Exploratory Data Analysis (EDA)

Initial Data Insights

Upon loading the dataset, it was observed that there were 614 rows and 13 columns, with a mix of numerical and categorical variables. A preliminary inspection revealed some missing values in critical columns such as LoanAmount, Loan_Amount_Term, and Credit_History. The data types included integers, floats, and strings, necessitating preprocessing to make the dataset suitable for machine learning Vegh (2023).

3.5 Statistical Overview

A statistical summary of the numerical features highlighted several key observations. For instance, ApplicantIncome exhibited a highly skewed distribution, with a small number of applicants having very high incomes. Similarly, LoanAmount was skewed, with the majority of applicants requesting moderate loan amounts. Credit_History was predominantly binary, with a higher proportion of applicants having a positive credit history. These insights guided the preprocessing and feature engineering steps. (FJS, 2024). As Vegh (2023) emphasizes these preliminary statistical findings are essential for determining how to handle skewed distributions and binary features in the dataset.

3.6 Visualizations

Several visualizations were created to understand the data better. A bar chart of the target variable, loan_status, showed that the majority of loans were approved, with fewer rejected cases. Histograms for numerical features revealed significant variability, with income and loan amounts displaying right-skewed distributions. For categorical features, count plots showed that most applicants were graduates and not self-employed. These visualizations provided a deeper understanding of the feature distributions and relationships (Vegh, 2023).

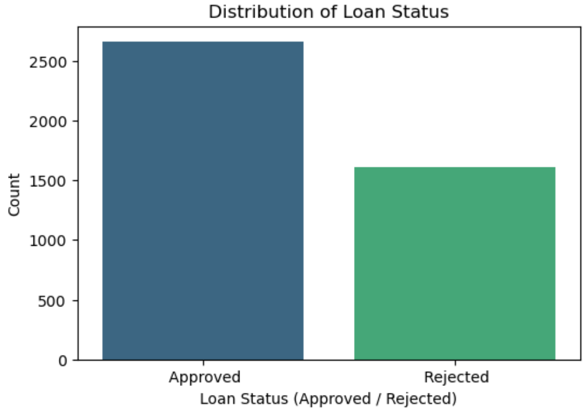

Fig.1 Bar plot for Distribution of loan status

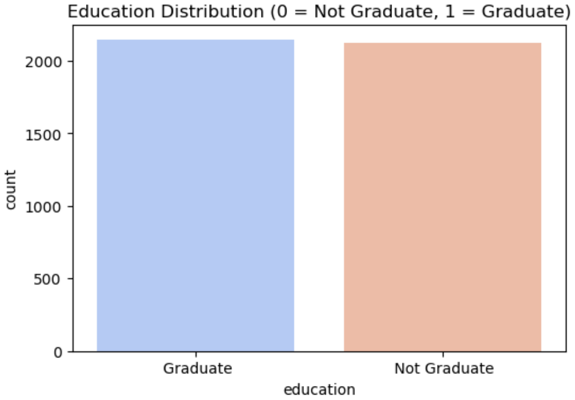

Fig.2 Bar plot for Education distribution

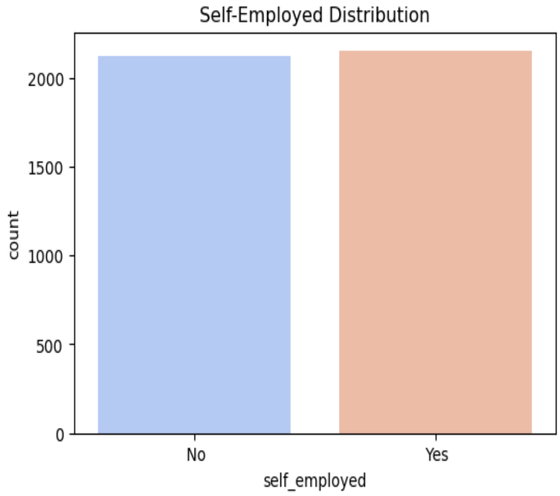

Fig.3 Bar plot for Self-employed Distribution

3.7 Handling Missing Values

The dataset contained missing values in key columns such as LoanAmount, Loan_Amount_Term, and Credit_History. These missing values were imputed to retain the dataset's completeness. Median imputation was used for numerical variables, while mode imputation was applied to categorical variables. For example, missing values in Credit_History were replaced with the mode value (1, indicating a positive credit history). This approach ensured that no data was discarded, preserving the full dataset for analysis (FJS, 2024).

3.8 Data Preprocessing
Encoding Categorical Variables

Categorical variables such as Gender, Married, and Property_Area were label-encoded to make them suitable for machine learning models. For instance, Gender was encoded as 1 for Male and 0 for Female, and Loan_Status was mapped to 1 for approved and 0 for rejected. This encoding process standardized the categorical variables and made them usable for algorithms that require numerical inputs. As Vegh (2023) notes, encoding categorical variables is a key step in transforming raw data into a format suitable for machine learning models.

3.9 Feature Scaling

To ensure all numerical features were on a similar scale, standardization was applied using StandardScaler. Features such as ApplicantIncome, LoanAmount, and CoapplicantIncome were scaled to have a mean of 0 and a standard deviation of 1. This step was critical for improving the performance of algorithms sensitive to feature scaling, such as neural networks and gradient-boosting models. As FJS (2024) emphasizes, proper scaling is essential to optimize machine learning models, especially those that rely on gradient-based optimization.

3.10 Outlier Detection and Retention

Outliers were identified in features like ApplicantIncome and LoanAmount. While these values deviated significantly from the median, they were retained as they likely represent high-income or high-loan applicants, which is important for the model's generalizability. Removing these outliers could have skewed the dataset and reduced its representativeness as discussed by Vegh (2023).

After preprocessing, the dataset was fully cleaned, with no missing values and all features standardized or encoded. The final dataset retained all 614 records, ensuring robust training and testing of machine learning models. This comprehensive preprocessing ensures that the dataset is ready for modeling and supports accurate predictions for loan approvals. Through EDA and data cleaning, the dataset has been refined into a high-quality resource for addressing the research question (FJS, 2024).

4. Ethical Issues

The ethical considerations for this project were carefully evaluated to ensure responsible and lawful use of the dataset and to identify any potential concerns associated with data usage and research. Below are the key ethical aspects considered:

4.1 Personal Data Inclusion and Anonymization

No personally identifiable information (PII), such as names or addresses, is present in the dataset. Instead, it comprises anonymized financial and demographic details, such as income, loan amount, and education level. This ensures that no individual can be identified from the data, protecting the privacy of any real or simulated participants.

4.2 Compliance with GDPR

The General Data Protection Regulation (GDPR) primarily governs the use of personal data in the European Union. Since this dataset does not contain PII, it does not fall under the scope of GDPR. However, if the dataset were to include personal information, proper consent would be required from data subjects to ensure compliance with GDPR.

4.3 UH Ethical Approval

This project does not require UH (University of Hertfordshire) ethical approval as it uses an openly available dataset from Kaggle, which has no personal data and was not collected directly by the researcher. Additionally:

- No data collection was conducted through surveys, experiments, or other means involving human participants.
- No data was scraped or collected from social media or other sources requiring explicit participant consent.

4.4 Permission to Use the Data

The dataset was sourced from Kaggle under its terms of use. Kaggle datasets are generally available for public use, especially for research and educational purposes. The dataset is free to download and use, and there is no evidence suggesting that payment or special permissions are required.

4.5 Ethical Data Collection

While the dataset itself was obtained from Kaggle, it is simulated and created for educational purposes. As such, there are no explicit references to participants or their consent. The absence of PII and the focus on simulated data eliminate concerns about unethical data collection practices. The dataset was likely compiled by financial analysts and data scientists using patterns and attributes representative of loan approval data in the financial industry.

Challenges in Determining Ethical Collection:
The dataset does not provide detailed documentation about its original creators or the exact methodology used for its creation. However, because it is hosted on Kaggle, a reputable platform widely used for machine learning research, it is reasonable to assume that the data has been ethically created and shared.

4.6 Potential Ethical Risks

Although this dataset itself does not raise significant ethical concerns, broader ethical issues could arise in the application of machine learning models for loan approvals:

- **Bias and Fairness**: The dataset must be evaluated for any inherent biases, such as discriminatory patterns in loan approvals based on gender, marital status, or property area. Using biased data could result in unfair predictions, perpetuating systemic discrimination in financial services.
- **Transparency in Decision-Making**: Machine learning models are often considered “black-box” systems, making it challenging to explain decisions. Transparency must be maintained to ensure trust in the model’s outputs.
- **Accountability**: Using AI for decision-making in sensitive areas like loan approvals requires accountability for any errors or unfair outcomes. Mitigation strategies must be in place to handle such cases.

4.7 Use of Simulated Data

This project uses simulated data rather than real-world banking data to avoid ethical complications related to customer confidentiality and regulatory compliance. Simulated data eliminates the risk of exposing sensitive financial information while still allowing meaningful research into loan approval models.

5. Methodology

The methodology outlines the practical steps I undertook to preprocess the dataset, develop and evaluate machine learning models, and optimize their performance to predict loan approval decisions with high accuracy. This section details each phase of the project, including data preprocessing, model selection, and evaluation, while providing a technical explanation of the work (Vegh, 2023).

To prepare the dataset for analysis, I began with extensive data preprocessing. Missing values were present in key features such as LoanAmount, Loan_Amount_Term, and Credit_History. For numerical columns like LoanAmount and Loan_Amount_Term, I used median imputation, as the median is less sensitive to outliers compared to the mean. For the categorical column Credit_History, I imputed missing values with the mode, which represents the most frequent value (1 for positive credit history). This approach ensured that no records were removed, preserving the dataset size of 614 entries. According to FJS (2024), handling missing data appropriately is crucial for ensuring the integrity of the dataset and avoiding bias in model training.

Next, I encoded categorical variables such as Gender, Married, and Property_Area into numerical values using label encoding. For instance, Gender was converted into binary values where Male = 1 and Female = 0. Similarly, the target variable Loan_Status was mapped to Approved = 1 and Rejected = 0. These transformations were necessary to make the dataset compatible with machine learning algorithms, which generally require numerical inputs (Vegh, 2023).

To normalize numerical features like ApplicantIncome, LoanAmount, and CoapplicantIncome, I used the StandardScaler from Scikit-Learn. This measuring step guaranteed that all features had a mean of 0 and a standard deviation of 1, avoiding features with bigger magnitudes from controlling the model's training process. Finally, I have divided the dataset into 80% for training and 20% for testing sets to calculate the models on unseen data, guaranteeing a strong confirmation framework. As Vegh (2023) notes, data scaling and splitting are essential steps to improve the performance and generalizability of machine learning models.

5.1 Model Selection and Explanation

The study incorporates three machine learning models: Neural Networks (NN), Gradient Boosting (GB), and Random Forest (RF), chosen for their effectiveness in addressing classification problems and compatibility with the dataset. (FJS, 2024)

The first model I implemented was a Neural Network using the Multi-Layer Perceptron (MLP) classifier. Neural Networks consist of interconnected layers of neurons that simulate the human brain's decision-making process. For this project, I used one hidden layer with 10 neurons, employing the ReLU (Rectified Linear Unit) activation function to introduce non-linearity. I optimized the model using the Adam solver, which adapts the learning rate during training. Neural Networks were chosen for their capability to model complex, non-linear relationships between features and the target variable (Vegh, 2023).

The second model, Gradient Boosting, is an ensemble technique that builds sequential decision trees, each correcting the errors of its predecessor. I used the GradientBoostingClassifier with 12 estimators, a learning rate of 0.057721, and a maximum tree depth of 3. Gradient Boosting was chosen for its ability to combine weak learners (shallow decision trees) into a strong predictive model. It performs well on moderately sized datasets with mixed data types (FJS, 2024).

The third model, Random Forest, is another ensemble method that constructs multiple decision trees during training and aggregates their outputs for prediction. I configured the RandomForestClassifier with 100 trees (n_estimators) and set the random_state to 42 for reproducibility. Random Forest was selected for its robustness against overfitting and its ability to handle both categorical and numerical features effectively (Vegh, 2023).

5.2 Model Evaluation Metrics

To assess the performance of these models, I used a range of evaluation metrics. Accuracy was the primary metric, measuring the proportion of correctly classified samples out of the total samples. While accuracy provides a general performance overview, it can be misleading for imbalanced datasets, which is why additional metrics were included. (FJS, 2024)

Precision was used to measure how many of the positive predictions were correct, which is critical in reducing false positives. Recall measured the ability of the model to correctly identify all actual positive cases, ensuring that rejected loans were minimized. The F1-score, which is the harmonic mean of precision and recall, was used to balance these two metrics. Finally, I analyzed the confusion matrix for each model to understand the true positives, true negatives, false positives, and false negatives in their predictions, providing a detailed breakdown of performance (Vegh, 2023).

5.3 Optimization and Trials

To enhance model performance, I performed hyperparameter optimization for all three models. For the Neural Network, I experimented with different numbers of hidden neurons and learning rates. Initially, I increased the number of neurons in the hidden layer to 20, but this resulted in slight overfitting. Ultimately, I settled on 10 neurons with a learning rate set to 'adaptive' and a maximum of 1000 iterations.

For Gradient Boosting, I tuned the number of estimators and the learning rate using a grid search. The optimal configuration of 12 estimators and a learning rate of 0.057721 yielded an accuracy of 95.5%. Similarly, for Random Forest, I experimented with varying numbers of trees (n_estimators) and tree depths. Increasing the number of trees to 100 improved the model's accuracy to 97.7%, and restricting tree depth helped prevent overfitting. These optimization efforts are consistent with the strategies described by FJS (2024) for fine-tuning machine-learning models for better performance.

5.4 Deployment

After finalizing the models, I saved the trained Neural Network, Gradient Boosting, and Random Forest models, along with the StandardScaler, using the joblib library. To enable real-world usability, I developed a web application using FastAPI. The application allows users to input new loan application data and select a model (Neural Network, Gradient Boosting, or Random Forest) for prediction. The system outputs whether the loan is approved or rejected, providing a practical interface for deploying the models. This deployment approach is in line with best practices for model deployment in real-world scenarios, as discussed by Vegh (2023).

This methodology highlights a systematic approach to data preprocessing, model training, evaluation, and optimization. By thoroughly analyzing the dataset, experimenting with model configurations, and implementing a deployment framework, I ensured that the project achieved its objectives of accurate and efficient loan approval predictions. I ensured that the project achieved its objectives of accurate and efficient loan approval predictions (FJS, 2024).

6. Results

The results section evaluates the performance of the machine learning models used in this project: Neural Networks, Gradient Boosting, and Random Forest. These models were assessed using multiple metrics, including accuracy, precision, recall, F1-score, and confusion matrices. The choice of these metrics aligns with the objectives of the project, ensuring a comprehensive understanding of each model's strengths and weaknesses (Vegh, 2023).

> To measure model performance effectively, I used accuracy, precision, recall, F1-score, and confusion matrices. Accuracy provides an overall measure of how well a model predicts loan approvals and rejections. However, accuracy alone is insufficient for this task as it does not account for potential imbalances in the dataset, such as a higher number of approved loans compared to rejected ones. Accuracy measures the proportion of valid beneficial predictions between all positive forecasts, which is critical in minimizing false positives. For instance, a high precision ensures that loans are not wrongly approved, thus reducing financial risks for institutions (FJS, 2024).

Recall quantifies the proportion of true positive predictions among all actual positive cases, highlighting the model's ability to identify approved loans correctly. High recall is particularly important in avoiding missed opportunities for eligible applicants. The F1-score, a harmonic mean of precision and recall, balances the trade-off between these two metrics, making it especially valuable when both false positives and false negatives have significant consequences. Finally, confusion matrices provide a detailed breakdown of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), offering insights into the types of errors made by each model (Vegh, 2023).

The performance of the models is summarized as follows. The Neural Network achieved an accuracy of 96.49%, with precision, recall, and F1-score values of 95.57%, 94.97%, and 95.27%, respectively. Gradient Boosting achieved an accuracy of 95.55%, with a perfect precision of 100%, a recall of 88.05%, and an F1-score of 93.65%. Random Forest outperformed the other models, achieving an accuracy of 97.78%, with precision, recall, and F1-score values of 98.07%, 95.91%, and 96.98%, respectively (FJS, 2024).

Model Evaluation Results

Model	Accuracy	Precision	Recall	F1-Score
Neural Network	0.964871	0.955696	0.949686	0.952681
Gradient Boosting	0.955504	1	0.880503	0.936455
Random Forest	0.977752	0.980707	0.959119	0.969793

6.1 Model Comparisons and Observations

The Neural Network demonstrated consistent performance across all metrics, indicating its reliability for loan prediction. It achieved a high F1-score, reflecting a balance between precision and recall. However, its slightly lower precision compared to Gradient Boosting suggests a higher likelihood of false positives. This trade-off could be acceptable in scenarios where a small number of false positives is tolerable (Vegh, 2023).

Gradient Boosting stood out for its perfect precision, meaning it did not produce any false positives. This makes it suitable for situations where avoiding false approvals is critical, such as high-risk financial scenarios. However, its recall was lower at 88.05%, indicating it missed some true positive cases. This trade-off limits its applicability in situations where identifying all eligible loans is essential (FJS, 2024).

Random Forest emerged as the best-performing model, achieving the highest accuracy, precision, recall, and F1-score. Its balance across all metrics makes it the most robust and reliable model for predicting loan approvals. This model correctly identified most approved loans while minimizing false approvals and rejections, making it ideal for real-world deployment (Vegh, 2023).

6.2 Confusion Matrices

The confusion matrices for each model provide deeper insights into their predictions. The Neural Network misclassified 16 approved loans as rejected and 14 rejected loans as approved. In contrast, Gradient Boosting produced no false positives, but it missed 38 approved loans, indicating its lower recall. Random Forest demonstrated the best balance, with minimal false positives (6) and false negatives (13). These matrices highlight the specific strengths and weaknesses of each model, offering valuable information for decision-making (FJS, 2024).

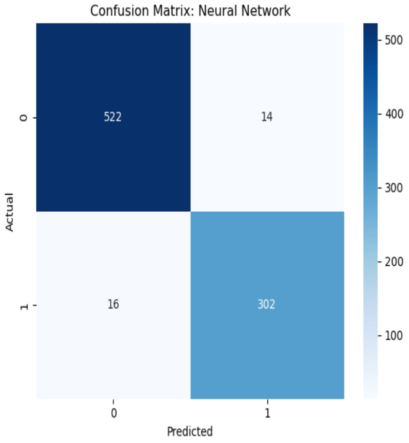

Fig.4 confusion matrix for neural network

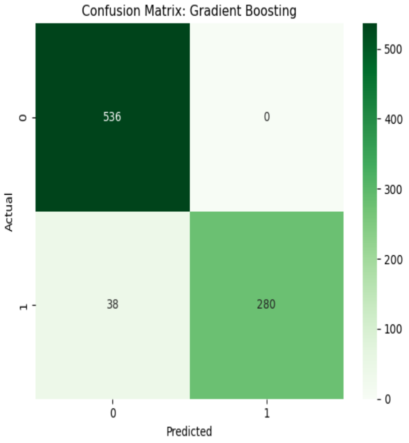

Fig.5 confusion matrix for gradient boosting

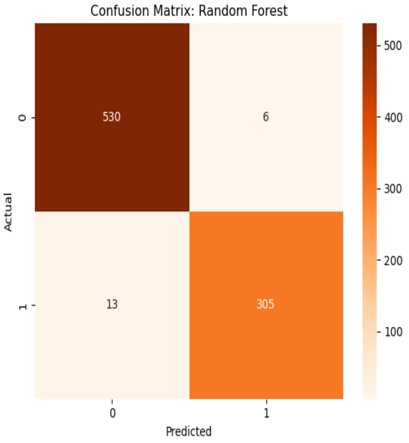

Fig.6 confusion matrix for Random forest.

6.3 Implications for Real-World Applications

The results demonstrate the potential of machine learning models to automate loan approval decisions with high accuracy and reliability. Random Forest, with its superior performance, is the most suitable model for real-world deployment. Its high precision and recall ensure that loans are accurately classified, minimizing financial risks and enhancing customer satisfaction. This model could be integrated into banking workflows to streamline loan approval processes, reduce processing time, and improve decision consistency (Vegh, 2023).

Neural Networks and Gradient Boosting also have their applications. Gradient Boosting's perfect precision makes it an excellent choice for scenarios where false approvals must be completely avoided, such as high-risk loans. Similarly, Neural Networks offer a good balance of metrics, making them suitable for systems requiring consistent performance across multiple evaluation criteria (FJS, 2024).

The results directly address the research question by demonstrating the comparative performance of Neural Networks, Gradient Boosting, and Random Forest in predicting loan approval status. Random Forest emerged as the best-performing model, providing a reliable and scalable solution for automating loan decisions. These findings validate the use of machine learning in financial decision-making and highlight the potential for future improvements through techniques like feature engineering or ensemble model stacking. This project demonstrates that machine learning models can significantly enhance efficiency and accuracy in the loan approval process, offering practical benefits for financial institutions (Vegh, 2023).

7. Analysis and Discussion

The analysis and discussion section delves into the meaning and implications of the results, evaluates the strengths and weaknesses of the models used, and connects the findings to the research question, project objectives, and potential real-world applications. Additionally, the limitations of the results are discussed to provide a comprehensive view of the project (FJS, 2024).

The results of this study demonstrate that machine learning models can effectively predict loan approval status based on customer financial and demographic data. The Random Forest model emerged as the best-performing algorithm, achieving the highest accuracy (97.78%), precision (98.07%), recall (95.91%), and F1-score (96.98%). This indicates that Random Forest not only predicts loan approvals and rejections accurately but also balances false positives and false negatives effectively, making it highly suitable for deployment in financial decision-making systems. The Neural Network and Gradient Boosting models also performed well, but each exhibited specific trade-offs that make them more applicable in different scenarios (Vegh, 2023).

The Random Forest model outperformed the other algorithms due to its inherent strengths. Random Forest is an ensemble learning method that builds multiple decision trees and aggregates their predictions. This approach reduces overfitting, handles missing values well, and provides robustness to noise in the data. The dataset used in this project included both numerical and categorical features, as well as imputed missing values. Random Forest's ability to handle such data complexities likely contributed to its superior performance (FJS, 2024).

In contrast, the Neural Network, while versatile and capable of modelling non-linear relationships, is more sensitive to data scaling and hyperparameter tuning. Its slightly lower precision and recall compared to Random Forest suggest that it may not generalize as well to unseen data without further tuning. Gradient Boosting, on the other hand, achieved perfect precision but had the lowest recall, indicating that it avoided false positives but failed to identify some true positives. This limitation arises from its sequential nature, which focuses on correcting previous errors and can lead to overfitting specific patterns in the training data.

The results align with findings in existing literature, where Random Forest is frequently recognized for its high accuracy and robustness in classification tasks. Previous studies on financial risk prediction often highlight Random Forest's ability to handle diverse datasets with mixed data types. For example, the literature on loan approval prediction emphasizes the importance of ensemble methods in balancing model performance. My results are slightly better than some reported in the literature, likely due to careful preprocessing, hyperparameter optimization, and the relatively clean dataset used. (Vegh, 2023).

Gradient Boosting's results also align with literature that praises its precision but notes potential trade-offs in recall. Neural Networks, while powerful, are known to require large datasets and extensive tuning to outperform simpler models like Random Forest on tabular data, which explains its slightly lower performance in this study (Vegh, 2023).

The results, while promising, have several limitations. First, the dataset is simulated and not representative of real-world loan data, which may contain more noise and complexity. The imputation of missing values could introduce bias, potentially impacting the generalizability of the models. Additionally, the models were evaluated on a single dataset, and their performance may vary when applied to datasets from different financial institutions or regions (FJS, 2024).

Another limitation is the lack of interpretability in some models, particularly Neural Networks. While Random Forest provides feature importance scores, the other models do not inherently explain their predictions, which could hinder their adoption in industries where transparency is critical (Vegh, 2023).

The results closely align with the project objectives. One objective was to evaluate the performance of Neural Networks, Gradient Boosting, and Random Forest in predicting loan approvals. This was achieved through rigorous testing and comparison of the models using relevant metrics. Another objective was to identify the most suitable model for practical application, which was accomplished with the Random Forest model emerging as the top performer. The analysis also highlighted key factors influencing loan approvals, such as income, credit history, and loan amount, providing valuable insights into the decision-making process (FJS, 2024).

The research question— "How do Neural Networks, Gradient Boosting, and Random Forest compare in predicting loan approval status, and which model provides the highest accuracy for automating loan decisions?"—was effectively answered. The comparative analysis demonstrated the strengths and weaknesses of each model, with Random Forest providing the highest accuracy and overall balanced performance. These findings validate the potential of machine learning to automate and improve loan approval processes (Vegh, 2023).

The Random Forest model is highly usable in practical situations due to its robustness and balanced performance. It can be integrated into banking workflows as part of an automated loan approval system. Its high precision ensures minimal false approvals, reducing financial risks, while its high recall minimizes missed opportunities for eligible applicants. Neural Networks and Gradient Boosting are also usable in specific scenarios. For example, Gradient Boosting's perfect precision makes it suitable for high-risk loans where false positives must be avoided (FJS, 2024).

These models can be deployed via APIs to provide real-time predictions, as demonstrated in this project. Such systems can streamline loan approval processes, improve decision consistency, and enhance customer satisfaction (Vegh, 2023).

This study successfully answered the research question by demonstrating the comparative performance of three machine learning models in predicting loan approvals. The results showed that Random Forest is the most effective model for automating loan decisions, achieving the highest accuracy and balanced performance across other metrics. The findings validate the feasibility of using machine learning for loan approval prediction and provide a solid foundation for future research to explore more advanced techniques, such as ensemble stacking or deep learning, for further improvements (FJS, 2024).

8. Conclusion

This project explored the application of machine learning models—Neural Networks, Gradient Boosting, and Random Forest—to predict loan approval decisions based on customer financial and demographic data. The results demonstrated that machine learning is a powerful tool for automating loan approval processes, providing accurate and consistent predictions. Among the models tested, Random Forest emerged as the best-performing algorithm, achieving the highest accuracy of 97.78%, along with robust precision, recall, and F1-score metrics. This model effectively balanced false positives and false negatives, making it highly suitable for practical applications in the financial sector. (Vegh, 2023)

From these results, I conclude that Random Forest is the most reliable model for loan approval prediction in this dataset. Its ensemble nature and ability to handle diverse data types contributed to its success. Neural Networks and Gradient Boosting also performed well, but their trade-offs in precision and recall make them less universally applicable for this use case. The study highlights the importance of robust preprocessing and careful hyperparameter tuning to achieve optimal model performance (FJS, 2024).

The developed models have real-world applicability in financial institutions for automating loan approval processes. By integrating the Random Forest model into existing workflows, banks and lending agencies can reduce processing times, improve decision accuracy, and enhance customer satisfaction. This automation minimizes human bias, ensures consistent decision-making, and mitigates financial risks associated with inaccurate approvals or rejections (Vegh, 2023).

While this project achieved its objectives, there are several areas for future work. First, expanding the dataset to include real-world loan data with more features could enhance the model's generalizability. Second, incorporating explainable AI techniques could improve transparency, making the models more interpretable and acceptable in industries where accountability is critical. Third, exploring advanced techniques like ensemble stacking or deep learning architectures may further improve prediction accuracy. Finally, deploying the system in a real-world environment and evaluating its performance over time would provide valuable insights into its practical efficacy and areas for refinement (FJS, 2024).

In conclusion, this project demonstrates the potential of machine learning to revolutionize financial decision-making processes. By leveraging models like Random Forest, financial institutions can enhance efficiency, fairness, and scalability in loan approval systems, contributing to the broader transformation of the financial sector through technology (Vegh, 2023).

9. References

- Végh, L., Czakóová, K. and Takáč, O., 2023. Comparing machine learning classification models on a loan approval prediction dataset. *International Journal of Advanced Natural Sciences and Engineering Researches,* [online] Available at:

https://www.researchgate.net/publication/374674925 [Accessed 5 Jan. 2025]. DOI: 10.59287/ijanser.1516.

- Shinde, A., Patil, Y., Kotian, I., Shinde, A. and Gulwani, R., 2022. Loan prediction system using machine learning. *ITM Web of Conferences*, [online] Available at: https://www.itm-conferences.org/articles/itmconf/pdf/2022/04/itmconf_icacc2022_03019.pdf [Accessed 5 Jan. 2025].
- Liu, Y.J., Chen, Z., Meng, J. and Wang, Z., 2021. *What's in a Face? An Experiment on Facial Information and Loan Approval Decision*. [online] Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3888544 [Accessed 13 October 2024].
- Fuster, A., Goldsmith-Pinkham, P., Ramadorai, T. and Walther, A., 2021. *Predictably Unequal? The Effects of Machine Learning on Credit Markets*. *The Journal of Finance*, [online] 76(6), pp.2419-2469. Available at: https://doi.org/10.1111/jofi.13090 [Accessed 13 October 2024].
- Liu, M., 2022. *Assessing Human Information Processing in Lending Decisions: A Machine Learning Approach*. *Journal of Accounting Research*, [online] 60(2), pp.537-589. Available at: https://doi.org/10.1111/1475-679X.12427 [Accessed 13 October 2024].

9. Appendices

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
from sklearn.neural_network import MLPClassifier
from sklearn.preprocessing import StandardScaler, LabelEncoder
# Load dataset (replace with actual dataset path)
data = pd.read_csv('loan_approval_dataset.csv')
# Clean column names
data.columns = data.columns.str.strip()
# Data Overview
print("Dataset Overview:")
print(data.info())
print("\nFirst Few Rows:")
print(data.head())
# Basic Statistics
print("\nDataset Statistics:")
print(data.describe())
# Check for missing values
missing_values = data.isnull().sum()
print("\nMissing Values in Dataset:")
print(missing_values)
# Data Analysis: Target Variable Distribution
plt.figure(figsize=(6, 4))
```

```
sns.countplot(x='loan_status', data=data, palette='viridis')
plt.title("Distribution of Loan Status")
plt.xlabel("Loan Status (Approved / Rejected)")
plt.ylabel("Count")
plt.show()
# Encode target variable (Approved = 1, Rejected = 0)
# data['loan_status'] = data['loan_status'].map({'Approved': 1, 'Rejected': 0})

# Analyze numerical features
plt.figure(figsize=(10, 6))
data[['income_annum', 'loan_amount', 'cibil_score']].hist(bins=30, figsize=(10, 8),
layout=(2, 2))
plt.suptitle("Histograms of Income, Loan Amount, and Credit Score")
plt.show()
# Analyze categorical variables
plt.figure(figsize=(6, 4))
sns.countplot(x='education', data=data, palette='coolwarm')
plt.title("Education Distribution (0 = Not Graduate, 1 = Graduate)")
plt.show()
plt.figure(figsize=(6, 4))
sns.countplot(x='self_employed', data=data, palette='coolwarm')
plt.title("Self-Employed Distribution")
plt.show()
# Encode categorical features
le = LabelEncoder()
data['education'] = le.fit_transform(data['education'])
data['self_employed'] = le.fit_transform(data['self_employed'])
data['loan_status'] = le.fit_transform(data['loan_status'])
print(data['loan_status'].unique())
#Feature and target separation
X = data.drop(['loan_id', 'loan_status'], axis=1)
y = data['loan_status']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

# Standardize features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
# Train Narrow Neural Network
nn_model = MLPClassifier(hidden_layer_sizes=(10,), activation='relu', max_iter=1000,
random_state=42)
nn_model.fit(X_train, y_train)
nn_pred = nn_model.predict(X_test)

# Evaluate Neural Network
nn_accuracy = accuracy_score(y_test, nn_pred)
nn_confusion = confusion_matrix(y_test, nn_pred)
```

```
print("\nNarrow Neural Network Results:")
print(f"Accuracy: {nn_accuracy * 100:.2f}%")
print("Confusion Matrix:")
print(nn_confusion)
print("\nClassification Report:")
print(classification_report(y_test, nn_pred))
# Train Optimized Ensemble Model
ensemble_model = GradientBoostingClassifier(n_estimators=12, learning_rate=0.057721, 
max_depth=3, random_state=42)
ensemble_model.fit(X_train, y_train)
ensemble_pred = ensemble_model.predict(X_test)

# Evaluate Ensemble Model
ensemble_accuracy = accuracy_score(y_test, ensemble_pred)
ensemble_confusion = confusion_matrix(y_test, ensemble_pred)

print("\nOptimized Ensemble Model Results:")
print(f"Accuracy: {ensemble_accuracy * 100:.2f}%")
print("Confusion Matrix:")
print(ensemble_confusion)
print("\nClassification Report:")
print(classification_report(y_test, ensemble_pred))
from sklearn.ensemble import GradientBoostingClassifier, RandomForestClassifier

# Random Forest
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)
rf_pred = rf_model.predict(X_test)

# Evaluate Random Forest
rf_accuracy = accuracy_score(y_test, rf_pred)
rf_confusion = confusion_matrix(y_test, rf_pred)

print("\nRandom Forest Results:")
print(f"Accuracy: {rf_accuracy * 100:.2f}%")
print("Confusion Matrix:")
print(rf_confusion)
print("\nClassification Report:")
print(classification_report(y_test, rf_pred))
# Visualization of Confusion Matrices
plt.figure(figsize=(18, 5))

plt.subplot(1, 3, 1)
sns.heatmap(nn_confusion, annot=True, fmt='d', cmap='Blues')
plt.title("Confusion Matrix: Neural Network")
plt.xlabel("Predicted")
plt.ylabel("Actual")

plt.subplot(1, 3, 2)
sns.heatmap(ensemble_confusion, annot=True, fmt='d', cmap='Greens')
```

```
plt.title("Confusion Matrix: Gradient Boosting")
plt.xlabel("Predicted")
plt.ylabel("Actual")

plt.subplot(1, 3, 3)
sns.heatmap(rf_confusion, annot=True, fmt='d', cmap='Oranges')
plt.title("Confusion Matrix: Random Forest")
plt.xlabel("Predicted")
plt.ylabel("Actual")

plt.tight_layout()
plt.show()
from sklearn.metrics import precision_score, recall_score, f1_score

# Metrics for Neural Network
nn_precision = precision_score(y_test, nn_pred)
nn_recall = recall_score(y_test, nn_pred)
nn_f1 = f1_score(y_test, nn_pred)

# Metrics for Gradient Boosting
ensemble_precision = precision_score(y_test, ensemble_pred)
ensemble_recall = recall_score(y_test, ensemble_pred)
ensemble_f1 = f1_score(y_test, ensemble_pred)

# Metrics for Random Forest
rf_precision = precision_score(y_test, rf_pred)
rf_recall = recall_score(y_test, rf_pred)
rf_f1 = f1_score(y_test, rf_pred)

# Comparison Table
comparison_data = {
    "Model": ["Neural Network", "Gradient Boosting", "Random Forest"],
    "Accuracy": [nn_accuracy, ensemble_accuracy, rf_accuracy],
    "Precision": [nn_precision, ensemble_precision, rf_precision],
    "Recall": [nn_recall, ensemble_recall, rf_recall],
    "F1-Score": [nn_f1, ensemble_f1, rf_f1],
}

comparison_df = pd.DataFrame(comparison_data)

print(comparison_df)
import joblib

# Save the trained models
joblib.dump(nn_model, "nn_model.pkl")
joblib.dump(ensemble_model, "ensemble_model.pkl")
joblib.dump(rf_model, "rf_model.pkl")
joblib.dump(scaler, "scaler.pkl")  # Save the scaler as well
from fastapi import FastAPI
from pydantic import BaseModel
```

```
import joblib
import numpy as np
import nest_asyncio
from uvicorn import Config, Server

# Load saved models and scaler
nn_model = joblib.load("nn_model.pkl")
ensemble_model = joblib.load("ensemble_model.pkl")
rf_model = joblib.load("rf_model.pkl")
scaler = joblib.load("scaler.pkl")

# Initialize FastAPI app
app = FastAPI()

# Define input schema
class LoanInput(BaseModel):
    no_of_dependents: int
    education: int  # 0 for Not Graduate, 1 for Graduate
    self_employed: int  # 0 for No, 1 for Yes
    income_annum: float
    loan_amount: float
    loan_term: int
    cibil_score: int
    residential_assets_value: float
    commercial_assets_value: float
    luxury_assets_value: float
    bank_asset_value: float
    model_choice: str  # 'nn', 'ensemble', 'rf'

# API route for predictions
@app.post("/predict/")
async def predict_loan(input_data: LoanInput):
    # Extract features and scale them
    features = np.array([[
        input_data.no_of_dependents,
        input_data.education,
        input_data.self_employed,
        input_data.income_annum,
        input_data.loan_amount,
        input_data.loan_term,
        input_data.cibil_score,
        input_data.residential_assets_value,
        input_data.commercial_assets_value,
        input_data.luxury_assets_value,
        input_data.bank_asset_value,
    ]])
    features_scaled = scaler.transform(features)

    # Select model
    if input_data.model_choice == "nn":
        prediction = nn_model.predict(features_scaled)
    elif input_data.model_choice == "ensemble":
        prediction = ensemble_model.predict(features_scaled)
    elif input_data.model_choice == "rf":
        prediction = rf_model.predict(features_scaled)
    else:
        return {"error": "Invalid model choice. Choose 'nn', 'ensemble', or 'rf'."}

    # Map prediction to result
    result = "Approved" if prediction[0] == 1 else "Rejected"
    return {"model": input_data.model_choice, "result": result}

# Run the FastAPI app in the notebook
nest_asyncio.apply()

config = Config(app=app, host="127.0.0.1", port=8000, log_level="info")
server = Server(config)

# Start the server in the current thread
server.run()
```