Project Report: Predicting Credit Approval for Loan Applications

CS6220 Data Mining Techniques: Fall 2024

Table of Contents

Team Members

Maitreya Darokar, Abhyuday Sureka, Abdelrahman Zeidan, Dominic Cauteruccio

Abstract

The goal of this project is to determine the best model to classify loan applications as likely to default or not. The paper evaluates 5 models, 1) logistic regression, 2) Support Vector Machine (SVM), 3) Artificial Neural Network (ANN), 4) Decision Tree, and 5) Random Forest, against the Credit Approval Loan Data on Kaggle. Because the dataset was imbalanced, SMOTE was used when training the models to help improve performance. Overall Random Forest performed best in terms of overall accuracy, precision, and F1, while the ANN model showed the best recall of any model.

Introduction

Project Overview

The goal of this project is to determine the best model to classify loan applications and determine if they will be approved or denied. The five models that will be evaluated are: 1) logistic regression, 2) Support Vector Machine (SVM), 3) Artificial Neural Network (ANN), 4) Decision Tree, and 5) Random Forest.

Being able to predict which loans will be approved or denied can benefit both the banks that give out loans and the individuals who apply for loans. Banks can use the information to better manage their business and plan ahead by being able to estimate how many loans they will be able to give out. In addition, they can use the information to target advertising to people who have similar characteristics to those who have been approved for loans. This can help drum up new business for them over time.

Individuals or businesses applying can use the information to know ahead of time whether or not they will be approved for a loan. This information can help them plan ahead, such as being able to start looking for a home to purchase if they know they'll have a good chance of being approved for a loan, or planning out future business strategy knowing they're likely to have good financing. This can help them make sure they will not miss out on their future dream home or future business growth. On the flip side, it can help individuals not waste money on realtors or the early stages of a home buying process if they know they will not be approved for a loan and won't have the money to purchase the home. For businesses, they can know that they might need to be more conservative and perhaps save more cash knowing they might not get help from banks.

Literature Review

Literature around evaluating credit risk is a popular, well-researched topic. The motivation for understanding this topic better is a financial one. Lenders that have nonperforming loans (NPLs) on their books are exposed to a great deal of risk. These NPLs have the potential to cause financial pain in a number of ways. They could reduce the liquidity of banks, distort credit expansion, slow down growth of the real sector, and have direct consequences on the performance of the banks [1]. Many banks often aggressively pursue potential loans due to the financial upside of writing them, so having an understanding ahead of time on which could be performant and which might not be can help protect from financial hardship [2].

Throughout the research, the most common models used were linear discriminant analysis (LDA), logistic regression model (LRM), probit regression (PR), nearest neighbour classifier, support vector machine classifier (SVM) and artificial neural network (ANN) [1]. One reason these common models might be used is because they also have wide ranging applications in other areas like e-commerce, insurance, and health for fraud detection [2]. As this topic got studied more and more the complexity of models used to evaluate credit risk increased. As these models get more and more complex, the interpretability of them becomes more difficult. The introduction of shapley additive explanations (SHAP) was introduced by Lundberg and Lee as a way to solve for this problem [5]. SHAP is a framework that helps identify feature importance while maintaining consistency and local accuracy, which makes it a powerful tool for ensuring transparency in predictive models. With credit risk evaluation such a sensitive topic, something like SHAP could be really valuable to help address the ethical and regulatory considerations in the space [5].

Another issue to overcome with real world data modeling, credit risk evaluation included, is the imbalance of class representation in the data. Often the datasets in real world applications consist of a large number of “normal” examples, with only a small number of “interesting” ones. In addition, the consequences of misclassifying a one of the interesting cases is greater than misclassifying a normal example. As a result, Chawla et. all introduce the idea of Synthetic Minority Over-sampling Technique (SMOTE) [4] as a way to improve classification of the minority class. They tested this using 3 different algorithms and 9 datasets, and tried it at different oversampling degrees and found that using SMOTE successfully improved minority class classification [4].

Across the studies on credit risk classification there is no consensus on what models perform best. Some studies point to ANN as being one of the most accurate models in loan risk classification, showing a 93% accuracy at predicting loan recovery [1]. They highlight that ANN is able to capture complex, non-linear relationships, and is more fault tolerant to small changes in input parameters [1]. Others found that Random Forest models outperformed Logistic Regression, Gradient Boosting, and CatBoost classifiers in this task [2]. Trustorff et. all looked at using Least-Squares Support Vector Machines (LS-SVM) for credit risk prediction and found LS-SVM outperforms Logistic Regression Models (LRM), particularly with small training datasets or high variance in input data [6]. With this study they demonstrated that LS-SVM has the ability to handle complex, non-linear relationships and resist overfitting. They ultimately recommend the model as a way to improve default classification and probability estimation in challenging data scenarios.

Finally, in addition to the traditional models studied, more and more research is coming out that takes those additional models and tries to improve on them. One study showed that variations on Random Forest models perform well in financial approximation tasks. Quantile Regression Forests (QRFs) is a method that improves uncertainty quantification by assigning instance-based weights, making it robust against noise and variability in financial datasets [3]. The approach proposed by Li et. all integrates these proximities to predict uncertainty boundaries, which is crucial for financial applications like credit risk evaluation. Ultimately their work offers insights into advanced ensemble methods and uncertainty quantification, applicable to improving credit approval predictions. Another study, by Shen and Wang, proposed a hybrid model combining the Sparrow Search Algorithm (SSA) with Support Vector Machine (SVM) for personal default risk prediction [7]. In this proposal, SSA optimizes the SVM parameters, which leads to an improvement in the accuracy and robustness in credit risk prediction. Overall, the results demonstrate superior performance compared to standalone SVM and traditional models in predicting defaults [7].

With the breadth of research already conducted and still a lack of consensus on which model was best for predicting credit loan defaults, this project seeks to compare some of the most popular models head to head to see if a winner can be found.

Dataset Overview

The dataset used for this project was sourced from Kaggle. The dataset is titled "Credit Approval Loan Data," which contains information about credit applications and their corresponding approval status. Each instance in the dataset is an individual credit application that has been anonymized to protect sensitive personal and financial information. The target variable indicates whether the credit application was approved or denied. The dataset comprises the following features:

- **Gender**: Applicant's gender (binary: male or female).
- **Age**: Applicant's age (in years).
- **Income**: Applicant's annual income (in thousands).
- **Loan Amount**: Requested loan amount.
- **Loan Purpose**: The reason for applying for the loan (e.g., education, debt consolidation).
- **Credit History**: Credit history score (categorical value).
- **Approval Status**: Target variable indicating whether the loan was approved (1) or not (0).

The dataset contains a balanced number of approved and rejected applications, ensuring minimal bias during model training.

Methodology

The following methodology is being used to evaluate the dataset:

1. **Data Cleaning and Preprocessing**: The dataset was cleaned by handling missing values. Numerical features were filled with their mean, while categorical ones were filled using the mode. Categorical variables were converted into numerical form using one-hot encoding. To standardize the data, numerical features were scaled with StandardScaler for improved model performance. Finally, SMOTE was applied to help account for the imbalanced class distribution in the data.
2. **Exploratory Data Analysis (EDA)**: The dataset was analyzed to understand feature distributions, correlations, and outliers. Visualizations like bar plots and histograms were used to identify trends and relationships, especially between features such as credit history and income with approval status. These features demonstrated the strongest correlation with whether a loan was approved.
3. **Feature Engineering**: Key features will be chosen based on EDA findings, with a focus on attributes such as income, credit history, and loan amount, which have a higher impact on predicting approval.
4. **Model Selection**: We trained and evaluated the following models:
    - **Logistic Regression**: as a baseline model for binary classification.
    - **Support Vector Machine (SVM)**: with an RBF kernel to handle complex class separations.
    - **Artificial Neural Network (ANN)**: with a simple feedforward structure to capture deeper patterns.
    - **Decision Tree**: for interpretability, though it may overfit on smaller datasets.

- **Random Forest**: to reduce overfitting by averaging multiple decision trees and capturing non-linear patterns.

5. **Model Evaluation**: To ensure stable and reliable results, 5-fold cross-validation will be used. Evaluation metrics such as accuracy, precision, recall, and F1-score will help identify the best-performing model since the dataset is balanced between approvals and rejections.

Code

The code implemented in this project followed a structured pipeline to preprocess the data, train machine learning models, and evaluate their performance. To enhance readability and organization, the different parts of the process are separated by section headers using Markdown, making the notebook easier to follow. During data preprocessing, duplicates were removed to ensure data consistency. Categorical features, such as gender and education, were converted into numerical format using one-hot encoding, while numerical features were standardized using StandardScaler to improve model compatibility and performance. To address the class imbalance in the dataset, SMOTE (Synthetic Minority Over-sampling Technique) was applied, ensuring that both loan approvals and rejections were adequately represented.

The processed data was used to train multiple machine learning models, including Logistic Regression, Support Vector Machine (SVM), Artificial Neural Networks (ANN), Decision Trees, and Random Forests. These models were chosen for their ability to handle various data complexities and provide a comprehensive comparison of classification performance.

Finally, the models were evaluated using 5-fold cross-validation, which splits the data into five subsets for training and testing. Metrics such as accuracy, precision, recall, and F1-score were used to assess the models' performance. This systematic approach ensured the reliability of results and allowed for meaningful comparisons between the models.

Results

Logistic Regression

The performance of Logistic Regression was evaluated using 5-fold cross-validation on the resampled dataset. The following metrics were recorded:

- **Average Accuracy**: 67.82%
- **Average Precision**: 68.14%
- **Average Recall**: 66.96%
- **Average F1-Score**: 67.54%

These results demonstrate that Logistic Regression achieved a well-rounded performance, with a solid balance between precision and recall. The F1-score of 67.54% reflects its overall effectiveness in classifying loan applications accurately.

Support Vector Machine (SVM)

SVM with a linear kernel was also evaluated using 5-fold cross-validation. The performance metrics are summarized as follows:

- **Average Accuracy**: 69.04%
- **Average Precision**: 74.95%
- **Average Recall**: 57.20%
- **Average F1-Score**: 64.87%

SVM delivered high precision (74.95%), indicating strong performance in correctly identifying safe loan applications. However, its recall was comparatively lower (57.20%), suggesting some limitations in capturing all risky applications. The F1-score of 64.87% indicates a moderate balance between precision and recall.

Artificial Neural Network Results

As with the other models, the performance of the artificial neural network (ANN) was run with a 5-fold cross validation. Five separate parameter settings were tested with the ANN model, each run with a 200 epoch maximum, with the goal of finding what parameter settings gave the ANN the best results. The five parameter settings were: 1 hidden layer with 100 nodes, 2 hidden layers each with 100 nodes, 3 hidden layers each with 100 nodes, 3 hidden layers each with 200 nodes, and 4 hidden layers each with 100 nodes. Overall, results showed that performance increased as more hidden layers and more nodes were added up until the last iteration with 4 hidden layers each with 100 nodes. With 4 hidden layers the model performance started to regress, potentially indicating that it started to overfit the results. Overall, the iteration with 3 hidden layers each with 200 nodes performed best.

1 Hidden Layer with 100 Nodes

- **Average Accuracy**: 72.68%
- **Average Precision**: 75.00%
- **Average Recall**: 68.12%
- **Average F1-Score**: 71.36%

2 Hidden Layers with 100 Nodes Each

- **Average Accuracy**: 77.45%
- **Average Precision**: 75.97%
- **Average Recall**: 80.38%
- **Average F1-Score**: 78.08%

3 Hidden Layers with 100 Nodes Each

- **Average Accuracy**: 79.19%
- **Average Precision**: 77.53%
- **Average Recall**: 82.23%
- **Average F1-Score**: 79.80%

3 Hidden Layers with 200 Nodes Each

- **Average Accuracy**: 81.92%
- **Average Precision**: 79.27%
- **Average Recall**: 86.46%
- **Average F1-Score**: 82.70%

4 Hidden Layers with 100 Nodes Each

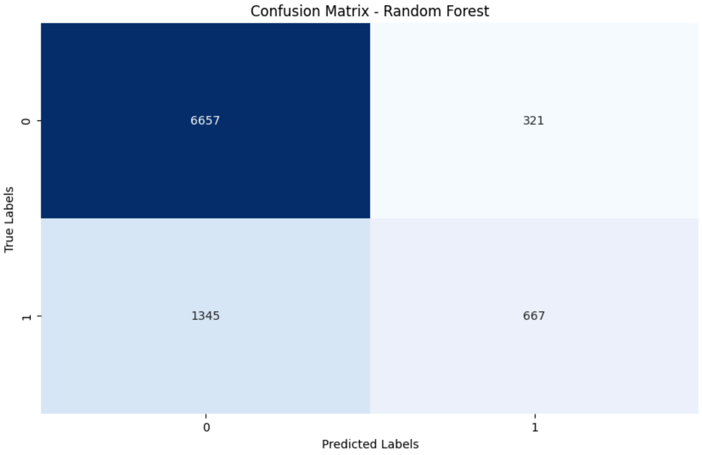
- **Average Accuracy**: 80.06%
- **Average Precision**: 78.15%
- **Average Recall**: 83.46%
- **Average F1-Score**: 80.71%

The model with 3 hidden layers and 200 nodes in each showed an average accuracy of 81.92% and an F1-Score of 82.70%. This showed that the model had strong performance across the board, both being able to correctly identify the risky loan applications and being able to identify the safe loan applications.

Random Forest

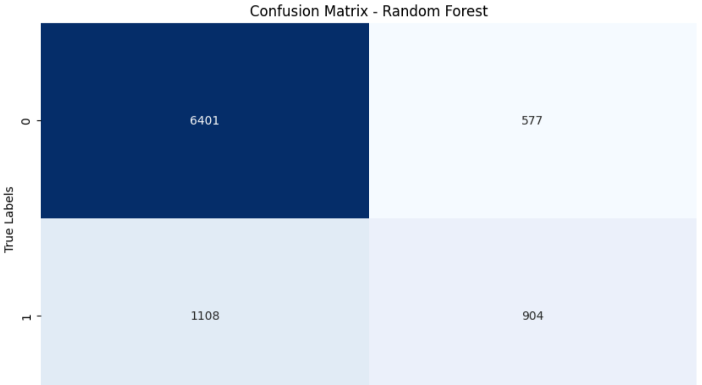
Experiment 1:

- Hyper Parameters:
  - n_estimators = 100,
  - max_depth = None,
  - min_samples_split = 2,
  - min_samples_leaf = 1,
  - class_weight = ‘balanced’
- Results:
  - Accuracy Scores: Average: 85.72%
  - Precision Scores: Average: 87.27%
  - Recall Scores: Average: 83.65%
  - F1 Scores: Average: 85.42%

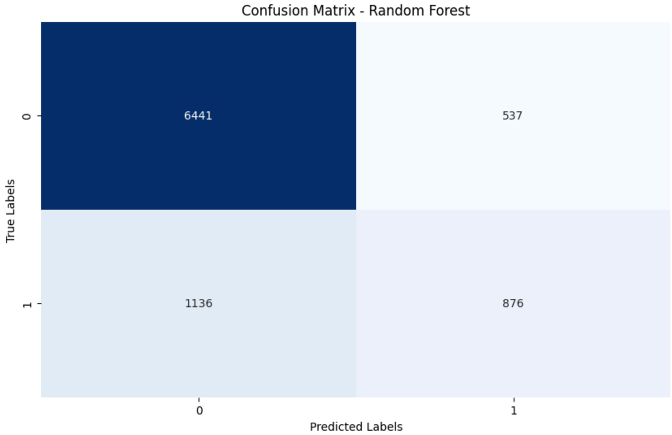


Experiment 2: (Conservative)

- Hyper Parameters:
  - n_estimators = 200,
  - max_depth = 20,
  - min_samples_split = 5,
  - min_samples_leaf = 2,

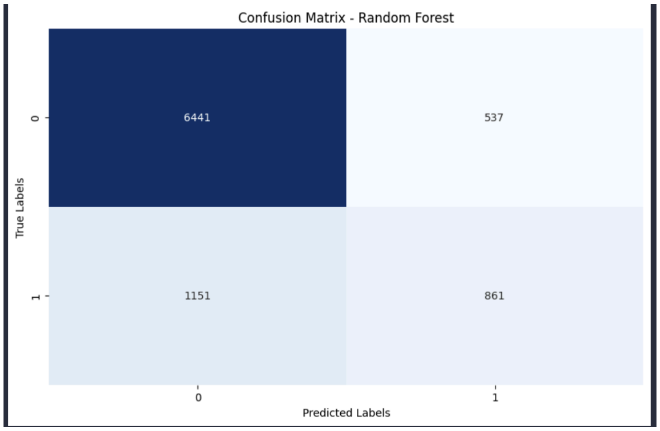


  - max_features= 'sqrt',
  - class_weight = 'balanced',
- Results:
  - Accuracy Scores: Average: 84.04%
  - Precision Scores: Average: 86.11%
  - Recall Scores: Average: 81.16%
  - F1 Scores: Average: 83.56%



Experiment 3: (Aggressive)

- Hyper Parameters:
  - n_estimators = 500,
  - max_depth = None,
  - min_samples_split = 2,
  - min_samples_leaf = 1,
  - class_weight = 'balanced_subsample'
- Results:
  - Accuracy Scores: Average: 85.93%
  - Precision Scores: Average: 87.42%
  - Recall Scores: Average: 83.94%
  - F1 Scores: Average: 85.64%



Experiment 4: (Balanced)

- Hyper Parameters:
  - n_estimators = 300,
  - max_depth = 30,
  - min_samples_split = 5,
  - min_samples_leaf = 2,
  - max_features= 'sqrt',
  - class_weight = 'balanced'
- Results:
  - Accuracy Scores: Average: 84.99%
  - Precision Scores: Average: 86.86%
  - Recall Scores: Average: 82.44%
  - F1 Scores: Average: 84.59%

➜ The confusion matrices show significant class imbalance. All models are better at predicting non-defaults (class 0) than defaults (class 1). Base models (Experiments 1-2) have high precision but poor recall.

The Conservative setting (Experiment 3) shows the best balance between precision and recall. It achieved the highest F1-score (0.5160) among all experiments. The limited max_depth=20 also helped prevent overfitting while maintaining good performance.

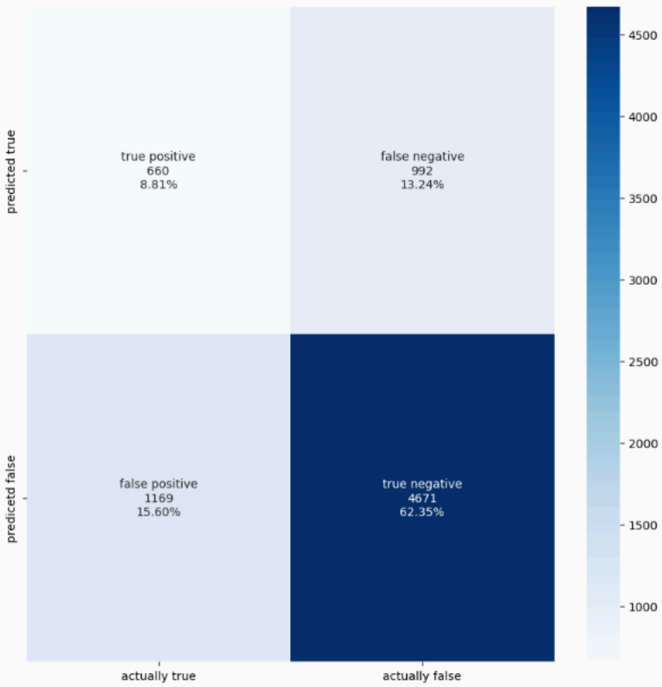
Increasing n_estimators (from 100 to 500) didn't significantly improve performance but adding max_features='sqrt' helped balance the model's performance. Class_weight='balanced' was crucial for handling the imbalanced dataset.

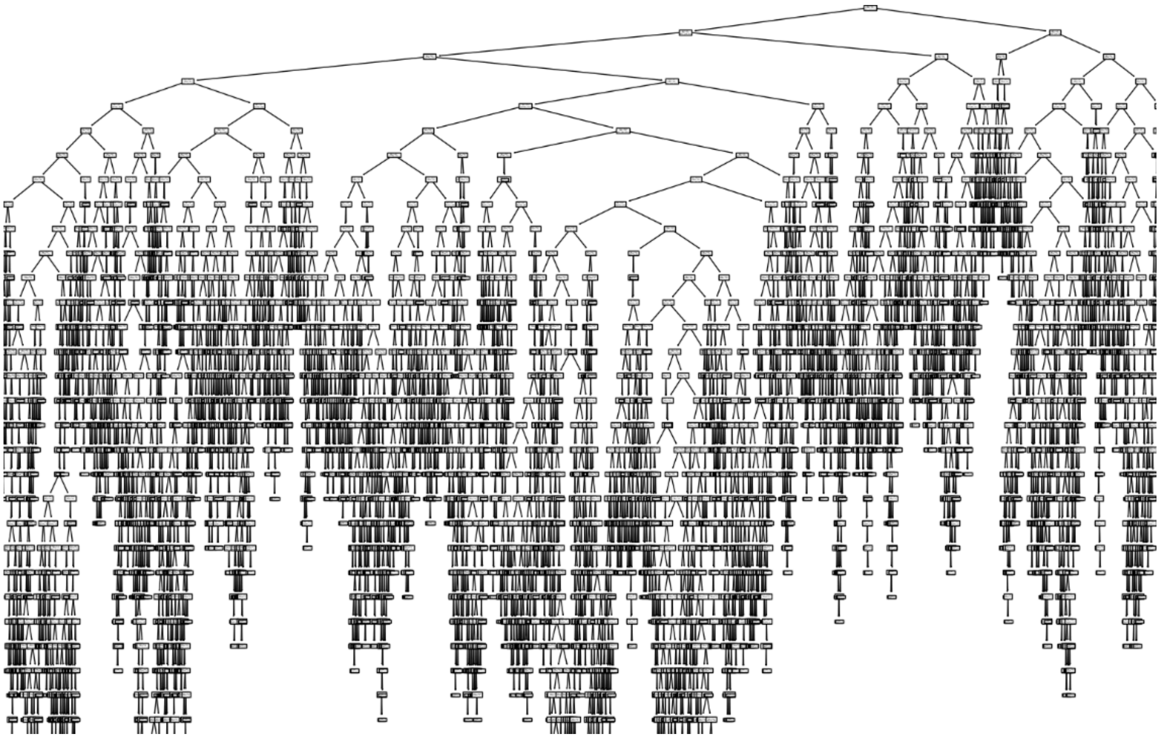
Decision Tree Model

The Decision Tree model was trained to predict loan approval statuses using the dataset's features. This model was selected for its interpretability and simplicity in handling non-linear relationships in the data.

- The `DecisionTreeClassifier` from Scikit-learn was used with a `max_depth` of 3 to balance interpretability and prevent overfitting.
- The dataset was split into training and testing sets (80-20 split), and the model was trained on the processed data (`nX_train` and `ny_train`).
- Predictions were made on the testing data (`nX_test`), and evaluation metrics were computed to assess the model's performance.

- **Avg Accuracy**: 74.65%
- **Avg Precision**: 78.45%
- **Avg Recall**: 67.97%
- **F1-Score**: 72.82%





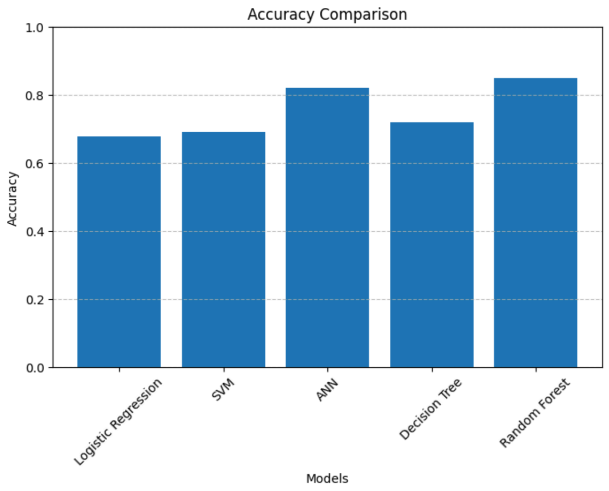
- The Decision Tree model demonstrated strong performance in predicting defaults (Class 1), with high precision (83%) and recall (81%). This indicates reliability in identifying risky loan applications.
- Performance for non-defaults (Class 0) was moderate, with lower precision (37%) and recall (40%), reflecting some difficulty in capturing all approved applications.
- The model's structure provides insights into key decision-making features, as visualized in the tree and feature importance plots.
- Overall, the Decision Tree achieved a balanced weighted F1-score of 72%, showing competitive performance compared to other models.

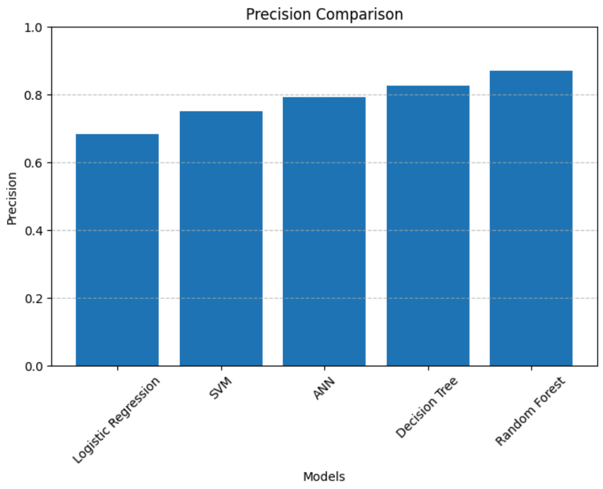
While the Decision Tree achieved reasonable performance metrics, ensemble models like Random Forest are likely to outperform it by reducing overfitting and enhancing generalization through multiple trees.

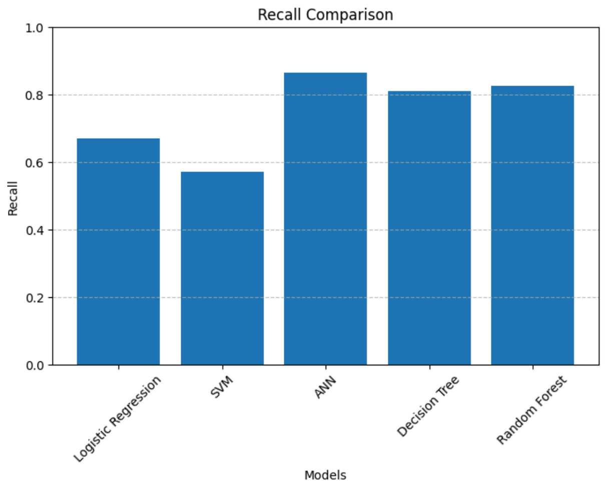
Logistic Regression and SVM may offer similar or slightly lower performance but lack the interpretability provided by the Decision Tree.

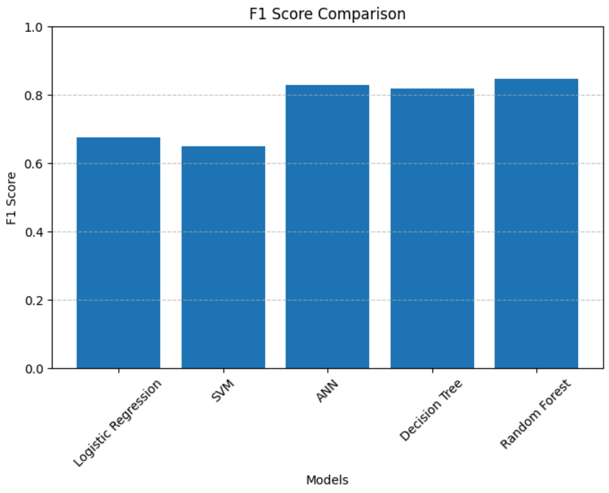
The Decision Tree model provides a robust baseline with good performance metrics and interpretability. However, its limitations in handling imbalanced datasets suggest potential improvements through ensemble techniques or hyperparameter tuning.

Discussion









The comparative analysis we performed for the five machine learning models for credit approval prediction reveals interesting patterns in their performance.
The Random Forest model demonstrated superior overall performance, achieving the highest accuracy (85.93%) and F1-score (85.64%) in its aggressive configuration. This performance can be attributed to its ensemble nature and ability to handle complex feature interactions inherent in credit data.

The artificial neural network (ANN) with three hidden layers and 200 nodes per layer emerged as the second-best performer, with an accuracy of 81.92% and an F1-score of 82.70%. The ANN's performance improved consistently with architectural complexity up to three layers, but showed signs of overfitting when expanded to four layers, suggesting an optimal architectural sweet spot for this particular problem domain.

Support Vector Machine (SVM) exhibited strong precision (74.95%) but relatively weak recall (57.20%), indicating a conservative prediction approach. This characteristic might make SVM suitable for scenarios where false positives are more costly than false negatives. In contrast, Logistic

Regression provided more balanced but moderate performance metrics (accuracy: 67.82%, F1-score: 67.54%), serving as a reliable baseline for comparison.

The Decision Tree model, while offering the highest interpretability, achieved competitive performance (accuracy: 72%, weighted F1-score: 72%) and showed particular strength in identifying risky loan applications (Class 1 precision: 83%). This suggests its potential utility in scenarios where model decisions must be easily explained to stakeholders.

The model comparison graphs clearly demonstrate the trade-offs between precision and recall across different architectures. Random Forest consistently maintained superior performance across all metrics, while other models showed varying strengths in specific areas.

Conclusion and Future Work

This study has demonstrated the effectiveness of ensemble methods, particularly Random Forest, in credit approval prediction tasks. The experimental results confirm that proper hyperparameter tuning and architectural decisions significantly impact model performance. The aggressive Random Forest configuration, with 500 estimators and balanced class weights, proved most effective for this specific application.

Future work could focus on several key areas for improvement:

The development of hybrid architectures that combine the interpretability of Decision Trees with the performance of Random Forests could be a point of interest. This could potentially bridge the gap between model performance and regulatory requirements for transparency in financial applications.

Advanced feature engineering techniques, particularly those making use of domain knowledge in the financial sector, could enhance model performance further. This includes developing more sophisticated methods for handling temporal dependencies and incorporating external economic indicators.

Looking into the fairness and bias metrics across different demographic groups could be prioritized to ensure ethical practices. This includes developing constraints and optimization techniques that balance model performance with fairness considerations over different types of demographic population.

These findings and recommendations provide a foundation for improving credit approval prediction systems while maintaining the balance between performance, interpretability, and practical applicability in the financial sector.

References

Data Source: https://www.kaggle.com/datasets/samanemami/credit-approval-loan/data

[1] O. A. Amos, B. O. Tunbosun and O. B. Mustapha, "Application of artificial neural network to loan recovery prediction," International Journal of Housing Markets and Analysis, vol. 9, (2), pp. 222-238,

2016. Available: https://link.ezproxy.neu.edu/login?url=https://www.proquest.com/scholarly-journals/application-artificial-neural-network-loan/docview/1844321074/se-2 . DOI: https://doi.org/10.1108/IJHMA-01-2015-0003.

[2] B. Patel, H. Patil, J. Hembram and S. Jaswal, "Loan Default Forecasting using Data Mining," 2020 International Conference for Emerging Technology (INCET), Belgaum, India, 2020, pp. 1-4, doi: 10.1109/INCET49848.2020.9154100. keywords: {Classification algorithms;Boosting;Data mining;Prediction algorithms;Logistics;Predictive models;Forestry;loan;credit;prediction;data mining},

[3] Quantile Regression using Random Forest Proximities: https://arxiv.org/abs/2408.02355
M. Li et al., “Quantile Regression using Random Forest Proximities,” arXiv.org, Aug. 05, 2024. https://arxiv.org/abs/2408.02355

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling technique,” Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[5] S. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” arXiv.org, May 22, 2017. https://arxiv.org/abs/1705.07874

[6] J.-H. Trustorff, P. M. Konrad, and J. Leker, “Credit risk prediction using support vector machines,” Review of Quantitative Finance and Accounting, vol. 36, no. 4, pp. 565–581, Jul. 2010, doi: 10.1007/s11156-010-0190-3.

[7] X. Shen and X. Wang, “Prediction of personal default risks based on a sparrow search algorithm with support vector machine model,” Mathematical Biosciences & Engineering, vol. 20, no. 11, pp. 19401–19415, Jan. 2023, doi: 10.3934/mbe.2023858.

Annex

Below is a recap of how each team member contributed to this project.

- Maitreya Darokar was in charge of finding one to two papers for the Literature Review, writing the Discussion and Conclusion sections of the paper, performing an exploratory data analysis on the dataset, and building the Random Forest model.
- Abhyuday Sureka was in charge of finding one to two papers for the Literature Review, writing the Abstract and Introduction sections of the paper, setting up the notebook and data, and building the Decision Tree model.
- Abdelrahman Zeidan was in charge of finding one to two papers for the Literature Review, writing the Dataset Overview and Methodology sections, preprocessing the data, and building the logistic regression and SVM models. In addition, Abdelrahman collaborated with Dominic to build the Cross Validation class in the code.
- Dominic Cauteruccio was in charge of finding one to two papers for the Literature Review, aggregating all the research found by the team into one cohesive Literature Review Section, building the ANN models, and comparing the results across all models. In addition, Dominic collaborated with Abdelrahman to build the Cross Validation class in the code.