**Bank Loan Approval Prediction Using Classification Tree and Logistic Regression**

**Project Summary**

The primary objective is to develop a predictive model tailored to aid Banks in assessing loan applications. This model will leverage customer attributes encompassing financial information, credit history, and other pertinent data to determine the appropriate course of action regarding the approval or rejection of a loan. In the current competitive financial landscape, having a robust system to accurately assess creditworthiness is crucial for mitigating risks and enhancing profitability.

Our analysis revolved around implementing and comparing two classification algorithms: Classification Tree and Logistic Regression. We constructed a Classification tree by manually entering parameters in our tree model and optimized the parameters through GridSearchCV.

Logistic Regression, a statistical model, utilizes a logistic function to estimate probabilities of binary outcomes. Despite its statistical underpinnings, it proved to be highly effective for our binary classification task of loan approval prediction. We examined the model coefficients, made predictions on the validation set, and analyzed performance metrics such as accuracy and confusion matrices.

Notably, both algorithms demonstrated high predictive performance, with the tuned Decision Tree model achieving approximately 99% accuracy on the training set and 98% accuracy on the validation set. The Logistic Regression model also performed admirably, with an accuracy of approximately 95% on both training and validation partitions.

While Decision Trees exhibited a slight edge in terms of raw performance metrics, we ultimately recommended the Classification tree model due to its simplicity, interpretability, and marginally lower misclassification rate compared to the Logistic Regression model. Overall, this project provides valuable experience in applying advanced data mining techniques to a real-world financial problem.

**Introduction**

In today's dynamic financial landscape, banks face the challenge of assessing loan applications to minimize risks and maximize profitability. One crucial aspect of the assessment process is determining whether to grant a loan to the customer. To tackle this challenge, predictive modeling techniques offer a solution by leveraging historical data to make informed decisions for the banks.

As the digital age unfolds, the landscape of banking undergoes a robust transformation, characterized by rapid technological advancements and shifting consumer behaviors. In this evolving world, the need for data-driven decision-making becomes increasingly important. By taking advantage of predictive modeling, banks can not only streamline their loan approval processes but also gain a competitive edge in an ever-expanding market. This helps the banks for growth and expansion. By looking into the customer data, we aspire to unravel the hidden patterns and insights that hold the key to unlocking sustainable success in the dynamic landscape of modern banking.

This project aims to develop a predictive model for loan approval using a dataset with diverse customer attributes. Attributes such as age, income, education level, family size, credit history, and financial habits like credit card usage and mortgage status. The dataset also has Personal loan variable whether a personal loan was granted, serving as the target variable. Through the process, the project seeks to provide valuable insights to banks on customer risk assessment and loan approval strategies.

**Main Chapter**

1. **Develop Understanding**

Project Goal: Our project seeks to equip banks with a predictive model to facilitate informed decision-making on loan applications. By taking historical data and predictive analytics, we aim to streamline the loan approval process and enhance efficiency in their decision.

Ultimate Outcome: The outcome of our project is twofold: Firstly, to enable banks to make data-driven decisions regarding loan approvals, thereby minimizing risks and maximizing profitability. Secondly, to optimize the loan approval process by accurately predicting the likelihood of loan repayment based on customer attributes and historical data.

1. **Obtain Data for Analysis**

Obtaining data for analysis is a crucial step in developing a predictive model for loan approval. We took the dataset bankloan.csv from Kaggle <https://www.kaggle.com/datasets/vikramamin/bank-loan-approval-lr-dt-rf-and-auc/code>

Below is the description of these columns and their significance in our analysis

**i. Demographic and Financial Attributes:**

Variables such as Age, Income, Education, Family Size, and zip code, highlighting how these demographic and financial attributes provide essential insights into a customer's financial profile and loan eligibility.

**ii. Credit Behavior and History:**

CCAvg (Credit Card Average Score), Mortgage, Securities Account, CD Account, and Credit Card. This section will discuss how customer credit behavior, existing financial commitments, and investment patterns influence loan approval decisions.

**iii. Technological Adoption and Banking Preferences:**

Online Banking and Securities Accounts, discuss the role of technological adoption and banking preferences in shaping customer relationships with financial institutions and their impact on loan management.

**iv. Target Variable: Personal Loan Approval:**

Focusing specifically on the Personal Loan, the target variable for our predictive modeling. It would discuss the criteria for personal loan approval and the importance of accurately predicting loan eligibility based on customer attributes and historical data. Personal Loan: 0 = No personal loan given, 1 = personal loan given

**3-4. Explore, Clean and Preprocess Data Reduce Data Dimension**

From the bank.csv file, we have 5000 rows and 14 columns

The 14 columns are

A screenshot of a computer

Description automatically generated

**Checking for Negative values in the dataset:**

**A white background with black dots

Description automatically generated**

Removing the negative values from the dataset is crucial. Maintains data integrity and quality by eliminating errors and inconsistencies. ensures accuracy, preventing misinterpretations. The interpretation of the dataset becomes clearer, focusing on relevant information, and consistency within the dataset is maintained.

After removing the negative values, we have 4948 rows and 14 columns

A screenshot of a computer

Description automatically generated

In our Bankloan dataset, all columns contain numerical values, eliminating the need for categorical data conversion or dummy variable creation.

**Removing unnecessary columns:**

We are removing the "ID" and "ZIP Code" columns from the dataset. The "ID" column is a unique identifier for each record and while the "ZIP Code" column contains geographic data, but it doesn't provide any meaningful information for predictive modeling.

**Checking For Missing Values**

**A white background with black dots

Description automatically generated**

There are no missing values in the dataset

**Checking for Null Values**

A white background with black and white clouds

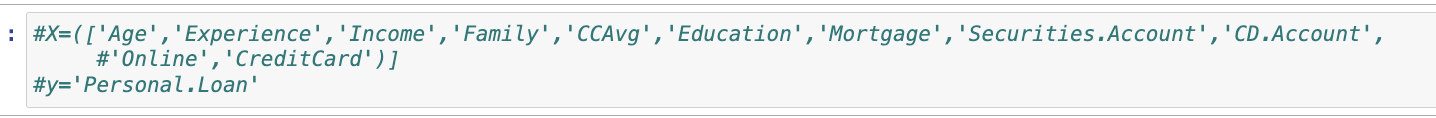
Description automatically generated

There are no Null Values in the dataset

**5. Determine the Data Mining Task**

Personal.Loan will be the outcome variable because it indicates whether a person has taken a personal loan or not. Therefore, Personal.Loan will be the dependent variable.

The predictors and outcome variables are below:



**6. Partition Data**

To prevent overfitting, we split our data into two parts:

1. Training Set: This contains 70% of the data and is used to develop the model. Here, the model learns from the data's patterns.

2. Validation Set: Comprising 30% of the data, this set evaluates the model's performance on unseen data, helping to ensure it can generalize well.

The training and validation data are separated, this prevents overfitting, With this strategy, experiments not only remain impartial but also avoid inconsistencies that may arise when multiple models are tested using the same data.

**7. Techniques**

The outcome variable is classification, So the methods We Considered are Classification Tree and Logistic Regression.

**8. Algorithm and Measures**

Classification Tree

**DecisionTreeClassifier** is a class in the sci-kit-learn library, it is used for building decision tree models for classification tasks. We Split the data as 70 percent data for training data and 30 percent as validation data. The main reason for data partitioning is to prevent overfitting

**A diagram of a diagram

Description automatically generated**

**9. Interpret Results**

**Income Decision Node (Root):**

The decision goes to the CCAvg if income is less than or equal to $113.5.

If income is greater than $113.5, it checks for Education.

**Education Decision Node:**

If the data point for the decision node is less than or equal to 1.5 (True), it checks for Family.

It checks the Income decision node if it's greater than 1.5 (False).

**Income Decision Node (Second Check):**

If income is less than or equal to $116.5, it checks for CCAvg.

If income is greater than $116.5, the loan will be approved.

**Confusion Matrix:**

When analyzing the performance of the model, we got the confusion matrix values for training and validation datasets, as follows:

A white background with black text

Description automatically generated

**Training Partition:**

Confusion Matrix (Accuracy 0.9882)

Accuracy: 0.9882

Misclassification Rate: 1 - 0.9882 = 0.0118

In this case, the model correctly predicted 3,112 instances of class 0 and 310 instances of class 1

There were 16 instances incorrectly classified as class 1 when they were actually class 0, and 25 instances incorrectly classified as class 0 when they were actually class 1.

The overall accuracy of the model on the training partition is 98.82%

**Validation Partition:**

Confusion Matrix (Accuracy 0.9838):

Misclassification Rate: 1 - 0.9838 = 0.0162

The model correctly predicted 1,331 instances of class 0 and 130 instances of class 1 in the validation dataset.

There were 9 instances incorrectly classified as class 1 when they were actually class 0, and 15 instances incorrectly classified as class 0 when they were actually class 1.

The overall accuracy of the model on the validation partition is 98.38%, indicating its ability to generalize well to new, unseen data.

**Interpretation:**

Both the training and validation partitions demonstrate high accuracy, with minimal misclassifications.

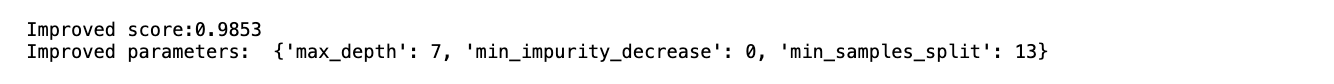
The model's performance on the validation partition is consistent with its performance on the training partition, indicating that it does not suffer from overfitting.

These results suggest that the smaller decision tree model has meaningful patterns from the training data and can effectively generalize to new data, making it a reliable predictor for loan acceptance in this context.

**GridSearchCV**

Initially, for the classification tree, we set the parameters as follows: max\_depth = 5 and min\_samples\_split = 10. This configuration was chosen to balance model complexity and generalization performance.

To optimize the model further, we employed GridSearch, a technique for parameter tuning. After running GridSearch, the optimal parameters obtained were max\_depth = 7 and min\_samples\_split = 14. These updated values represent improved parameter settings that enhance the model's ability.



The below Classification Tree with Grid Search comprises 49 nodes and involves 48 splits and maximum depth is 7 levels.

A diagram of a tree

Description automatically generated

**Confusion Matrix for GridSearch**

A white screen with black text

Description automatically generated

**Training Partition for Best Classification Tree with Grid Search:**

**Confusion Matrix (Accuracy 0.9910)**

**Misclassification Rate: 1 - 0.9910 = 0.0090**

In this case, the model accurately predicted 3,119 instances of class 0 and 313 instances of class 1. However, there were 22 instances incorrectly classified as class 1 when they were actually class 0, and 9 instances incorrectly classified as class 0 when they were actually class 1. The overall accuracy of the model on the training partition is 99.10%, indicating its high level of performance in correctly classifying instances.

**Validation Partition for Best Classification Tree with Grid Search:**

**Confusion Matrix (Accuracy 0.9832)**

**Misclassification Rate: 1 - 0.9832 = 0.0168**

For the validation dataset, the model correctly predicted 1,332 instances of class 0 and 128 instances of class 1. However, there were 17 instances incorrectly classified as class 1 when they were actually class 0, and 8 instances incorrectly classified as class 0 when they were actually class 1. The overall accuracy of the model on the validation partition is 98.32%, indicating its ability to generalize well to new, unseen data.

**Interpretation Result:**

The model demonstrates high accuracy on both the training and validation partitions, with minimal misclassifications.

Consistency between the model's performance on the training and validation partitions suggests that it does not suffer from overfitting.

These results imply that the best classification tree model derived from the grid search process has meaningful patterns from the training data and can effectively generalize to new data. Thus, it represents a reliable predictor for loan acceptance.

**Comparison of Classification Tree with and without GridSearch:**

**A white background with black text

Description automatically generated**

|  |  |  |  |
| --- | --- | --- | --- |
|  | SmallerTree(withoutGrid Search) | | Best Tree with Grid Search |
| Accuracy | 0.9838 | | 0.9832 |
| True Positives (TP) | | 1331 (class 0), 130 (class 1) | 1332 (class 0), 128 (class 1) |
| False Positives (FP) | | 9 (class 0), 15 (class 1) | 8 (class 0), 17 (class 1) |
| True Negatives (TN) | | 130 (class 1), 1331 (class 0) | 128 (class 1), 1332 (class 0) |
| False Negatives (FN) | | 15 (class 0), 9 (class 1) | 17 (class 0), 8 (class 1) |

**Smaller Tree:**

Correctly predicted instances of actual class 0: 1,331

Correctly predicted instances of actual class 1: 130

**Best Classification Tree with Grid Search:**

Correctly predicted instances of actual class 0: 1,332

Correctly predicted instances of actual class 1: 128

**Interpretation:**

Both models exhibit very high accuracy, with the smaller decision tree edging slightly ahead in overall performance. However, the grid search model, despite having a marginally lower accuracy, demonstrates a slightly better ability to correctly identify Actual 0 while a bit less in terms of classifying Actual 1’s. Overall, the difference in performance between the two models is minimal.

For an accuracy of 98.38%, the misclassification rate for smaller tree

Misclassification Rate = 1 - Accuracy = 1 - 0.9838 = 0.0162

for an accuracy of 98.32%, the misclassification rate for bestclasstree

Misclassification Rate = 1 - Accuracy = 1 - 0.9832 = 0.0168

The performance difference between the two models is marginal, with the smaller tree having a slightly better balance between false positives and false negatives. However, the differences are quite closer. Higher accuracy and lower misclassification is considered for better prediction, Therefore small tree (classification tree without grid search) is considered.

**Logistic Regression:**

After we ran the Logistic Regression model the intercept value and coefficients for the predictors are as follows:

**A white background with a black border

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

The majority of the validation records are correctly classified as 0 (no personal loan given). While

few indexes are classified as 1. The index of 3543 is misclassified. The actual status is no personal loan given but the logistic regression model misclassified them as 1 (personal loan given) because the probability of 1 in record 3543 is 57 percent.

A screenshot of a computer

Description automatically generated

From the above image, the record 1768 is wrongly classified. The Actual is 1 but classified as 0

. The index of 3543 is misclassified. The actual status is personal loan given but the logistic regression model misclassified them as 0 (personal loan not given) because the probability of the index 1768 is 74.4 percent

**Confusion Matrix**

A screenshot of a computer

Description automatically generated

**Training Partition:**

The training partition reveals an overall accuracy of 95.18%. The model accurately predicted 3082 instances of class 0 and 214 instances of class 1. However, there were 46 instances incorrectly classified as class 1 when they were actually class 0, and 121 instances incorrectly classified as class 0 when they were actually class 1.

Misclassification Rate: 1 - 0.9518 = 0.0482

**Validation Partition:**

Similarly, on the validation partition, the logistic regression model achieved an accuracy of 95.42%. It correctly predicted 1323 instances of class 0 and 94 instances of class 1. However, there were 17 instances incorrectly classified as class 1 when they were actually class 0, and 51 instances incorrectly classified as class 0 when they were actually class 1.

Misclassification Rate: 1 - 0.9542 = 0.0458

**Interpretation:**

Both the training and validation partitions demonstrate relatively high accuracy, with minimal misclassifications. The model's performance on the validation partition closely aligns with its performance on the training partition, suggesting that it generalizes well to new data without overfitting. These results indicate that the logistic regression model effectively captures meaningful patterns from the training data and can make reliable predictions for loan acceptance.

**Lift Chart**

A graph of a bar chart

Description automatically generated

The lift chart for the ‘Personal loan not approved’ status shows the ratio of the proportion of classifications as 0 (‘loan not given’) using the model vs. the proportion of the ‘loan not given’ flight status taken randomly for different percentiles in the validation partition. For the top 10% of the data most probable to be 0, the logistic model provides 1.1 times higher chance of 0 than the proportion of 0’s taken randomly

**Using Backward Elimination in Logistic Regression:**

We tried the logistic regression model based on Backward Elimination which can be a potentially a good choice for the classification of Personal Loan.

A white background with black text

Description automatically generated

When we ran the model after with the best variables from Backward Algorithm, The Intercept value and **Coefficients for Predictions** as follows

A white background with black text

Description automatically generated

**Confusion Matrix:**

A screen shot of a computer

Description automatically generated

**Training Partition**

The confusion matrix for the logistic regression model trained on the training partition using backward elimination reveals an overall accuracy of 95.06%. The model correctly predicted 30871 instances of class 0 and 211 instances of class 1. However, there were 47 instances incorrectly classified as class 1 when they were actually class 0, and 124 instances incorrectly classified as class 0 when they were actually class 1.

**Misclassification Rate**: 1 - 0.9506 = 0.0494

**Validation Partition**

Similarly, on the validation partition, the logistic regression model achieved an accuracy of 95.35%. It correctly predicted 1322 instances of class 0 and 94 instances of class 1. However, there were 18 instances incorrectly classified as class 1 when they were actually class 0, and 51 instances incorrectly classified as class 0 when they were actually class 1.

**Misclassification Rate**: 1 - 0.9535 = 0.0465

**Comparison of the Logistic Regression Model and Logistic Regression Model using backward Elimination:**

A screenshot of a computer

Description automatically generated

Both models exhibit comparable performance on the validation partition, with accuracies of approximately 95%. However, upon closer inspection, the logistic regression model has one fewer misclassification (17) compared to the model based on backward elimination (18).

**For Logistic Regression**

**Misclassification Rate** = 1 - Accuracy = 1 - 0.9542 = 0.0458

**For the Logistic Regression Model using backward Elimination**

**Misclassification Rate**: 1 - 0.9535 = 0.0465

Despite their similar performance, the Logistic Regression Model emerges as the preferred choice due to its superior accuracy of 95.42% compared to the 95.35% accuracy of the Logistic Regression Model using backward elimination. This higher accuracy with a lower misclassification rate, is suitable for making accurate predictions, aligning well with the criteria. Therefore, the Logistic Regression Model stands out as a optimal selection for prediction purposes.

**Conclusion:**

In conclusion, our project developed predictive models to assist banks in making informed decisions regarding loan approvals. Leveraging the bank loan dataset from Kaggle, we explored various financial, and behavioral attributes to understand their impact on loan approval decisions. Through data exploration, cleaning, preprocessing, and dimensionality reduction techniques, we prepared the data for predictive modeling.

With primary algorithms, Classification Tree, and Logistic Regression, to develop predictive models for loan approval. We initially constructed a model without parameter tuning for the Classification Tree and then optimized it using GridSearchCV. Similarly, Logistic Regression models were built both with and without backward elimination.

Our analysis revealed that the Classification Tree and Logistic Regression models achieved high accuracy rates on training and validation datasets, indicating their ability to generalize well to new, unseen data. However, upon comparison, the **Classification model** emerged as the preferred choice due to its **higher accuracy** and lower misclassification rate.

**Recommendations:**

Our analysis recommends implementing the Classification Tree model for loan approval prediction due to its superior accuracy and performance. This model can effectively assist banks in assessing loan applications, thereby minimizing risks, and helping to achieve maximum profitability.

**Benefits and Limitations:**

The use of data mining methods and techniques in our project offers several benefits,

Enhanced decision-making: Predictive models enable banks to make data-driven decisions, leading to more accurate and efficient loan approval processes.

Risk mitigation: By accurately assessing creditworthiness, banks can minimize the risk of default and financial losses.

Improved customer experience: Streamlined loan approval processes result in faster responses to customer applications, enhancing overall satisfaction.

**Bibliography:**

Dataset: [https://www.kaggle.com/datasets/vikramamin/bank-loan-approval-lr-dt-rf-and-](https://www.kaggle.com/datasets/vikramamin/bank-loan-approval-lr-dt-rf-and-%20%20auc/code)

[auc/code](https://www.kaggle.com/datasets/vikramamin/bank-loan-approval-lr-dt-rf-and-%20%20auc/code)

<https://www.kaggle.com/code/mohamedsamyy2/personal-loan-prediction/input>