# Predicting Loan Defaults Based on Customer Behavior

**Abstract** - The primary objective and contribution of this paper is to analyze loan prediction based on customer behavior using two machine learning models, K-Nearest Neighbors and Decision Tree. The dataset consists of 12 features with over 252,000 samples. This study describes Data process and purification, as well as exploratory data analysis of the dataset. K-Nearest Neighbors is a non-parametric method used for classification and regression, while Decision Tree is a statistical model suitable for binary classification. Both algorithms are suitable for predicting potential defaulters for consumer loan products.

## I. Introduction

Loan prediction based on customer behavior is a crucial aspect of risk assessment and management in financial institutions. This study focuses on employing machine learning techniques, specifically K-Nearest Neighbors (KNN) and Decision Tree algorithms, to predict potential defaulters for consumer loan products. With a dataset comprising 13 features and over 252,000 samples, this research aims to provide insights into the prediction accuracy and effectiveness of these models.

Effective data management and purification, along with exploratory data analysis, are essential steps in preparing the dataset for modeling. KNN, a non-parametric method versatile for both classification and regression tasks, is leveraged alongside Decision Tree, a statistical model particularly suitable for binary classification scenarios like loan default prediction.

The significance of this study lies in its practical application for financial institutions seeking to mitigate risks associated with lending by identifying customers more likely to default based on their historical behavior. By analyzing historic customer behavior data, institutions can enhance decision-making processes, allowing for more informed and proactive risk management strategies when acquiring new customers.

This introduction sets the stage for exploring the methodologies and findings of this research, shedding light on the potential of machine learning in improving loan prediction accuracy and aiding financial institutions in making sound lending decisions.

### Objective

- Identify key features for loan default prediction from customer behavior.
- Develop predictive models with KNN and Decision Tree algorithms to detect potential loan defaulters.
- Conduct exploratory data analysis to understand correlations in the dataset.
- Evaluate accuracy of KNN and Decision Tree models in predicting loan defaults.
- Provide practical recommendations for refining risk assessment strategies.

## II. Literature review

Loan prediction based on customer behavior is a critical area of research in the financial

sector, with significant implications for risk assessment and management. In this section, five relevant studies contribute to the understanding of this topic.

Firstly, the report by Zuama et al. (2024) explores using machine learning for loan default prediction based on customer behavior in the banking sector. Results show XGBoost outperforms other algorithms with an 89% accuracy rate, followed closely by Random Forest and Logistic Regression at 88%. This study emphasizes the importance of incorporating consumer behavior variables for a comprehensive understanding of loan projections. [1]

The second study, "Prediction and Analysis of Financial Default Loan Behavior Based on Machine Learning Model" by Chen (2022), addresses the increasing risk of loan defaults amid economic challenges. Chen proposes a method that involves data preprocessing, feature extraction, and the implementation of both a penalized linear regression model and a neural network prediction model. [2]

Thirdly, Ndayisenga (2021) utilizes machine learning to predict bank loan approval, identifying Gradient Boosting as the most effective model. [3]

The next source is a book titled “Intelligent Computing Theories and Application” , in the section of "WT Model & Applications in Loan Platform Customer Default Prediction Based on Decision Tree Algorithms" by Pang and Yuan (2018). This study introduces a WT early warning model for predicting customer defaults in loan platforms. It utilizes decision tree algorithms, including C5.0, CART, and CHAID, along with weighted calculating algorithms.

Lastly, the resource is A Comparative Study of Machine Learning Approaches for Non Performing Loan Prediction by Serengil et al. (2021) compare machine learning approaches for non-performing loan prediction, with LightGBM identified as the best-performing model. [5]

These studies collectively contribute to our understanding of loan prediction based on customer behavior, offering insights into predictive factors, analytical techniques, and risk management strategies in the lending sector.

# III. Data processing

### Data source and description

The data for this study was obtained from Kaggle, accessible through the following reference link [6]. It encompasses information pertinent to loan prediction, comprising a total of 13 features and 25,200 rows. Key features within the dataset include:

```
Id                   int64
Income               int64
Age                  int64
Experience           int64
Married/Single      object
House_Ownership     object
Car_Ownership       object
Profession          object
CITY                object
STATE               object
CURRENT_JOB_YRS      int64
CURRENT_HOUSE_YRS    int64
Risk_Flag            int64
```

## Data Preparation/Pre-processing

Data preprocessing is a crucial step to ensure the dataset is suitable for training machine learning models. In this study, several preprocessing steps were undertaken to prepare the data for analysis:

*1. Removal of the "id" column:*

Unique identifiers, such as the "id" column, do not contribute to the learning process. Therefore, this column was dropped from the dataset using the `df.drop()` function.

*2. Handling missing values:*

To check for missing values in the dataset, the `df.isnull().sum()` function was utilized. The dataset did not contain any missing or empty values, so no filling of missing values was required.

*3. Convert categorical variables to integer:*

The categorical variables such as "Married/Single," "House_Ownership ," "Car_Ownership," "Profession," "CITY," and "STATE" will be converted to integer type using the method `df.astype({*'colname'*: int, ...})`, as most machine learning algorithms require numeric input for analysis.

With these preprocessing steps completed, the dataset is now ready for further analysis and model training.

## Exploratory Data Analysis and feature selection

*1. Distributions of continuous variables:*

The analysis includes two distribution boxplots for the 'Age' and 'Income' variables. These visualizations examine if the distributions display skewness, long tails, or outliers. Both box plots suggest normal distributions, with no significant signs of skewness, long tails, or outliers present in the data.

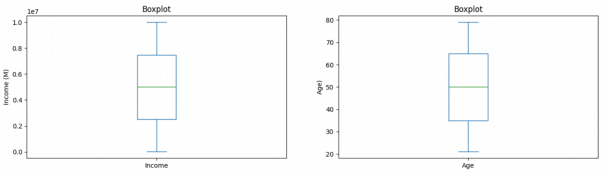


*Figure1: Box plot of Income and Age*

*2. Bivariate Analysis:*

The analysis aims to assess whether a categorical variable is predictive of loan_status. The analysis includes an assessment of "Married/Single", "House_Ownership", and "Car_Ownership". The findings point out that within the "houseowner" category, individuals who rent their homes exhibit a higher proportion of red flags, indicating a potentially higher risk associated with loan defaults. This trend may be indicative of less stable financial or credit conditions among renters compared to homeowners. Further investigation is warranted to understand the underlying factors contributing to this phenomenon.

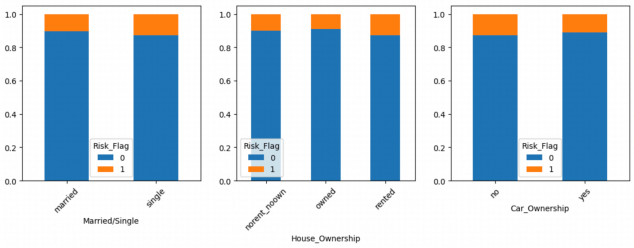


*Figure2: Bivariate Analysis*

*3. Correlations analysis:*

By analyzing the correlation heatmap and correlation table, it becomes evident which variables exhibit stronger correlations and whether these correlations are positive or negative. Upon examination, a notable correlation between "CURRENT_JOB_YRS" and "Experience" is observed. Given that

"Experience" better encapsulates overall work experience, a decision has been made to drop "CURRENT_JOB_YRS" from further analysis.

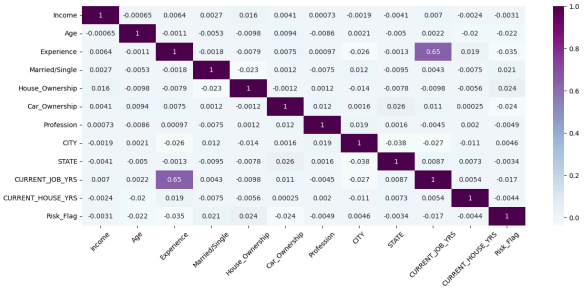


*Figure3: Correlation heatmap*

*4. Strength of association:*

The strength of association between feature variables and the target variable is assessed using four different methods. These include Chi-squared score, F-Test score, ExtraTreesClassifier, and Mutual Information (MI) Test score. Notably, across all these methods, Income consistently obtains high scores, suggesting that it is a highly predictive feature.

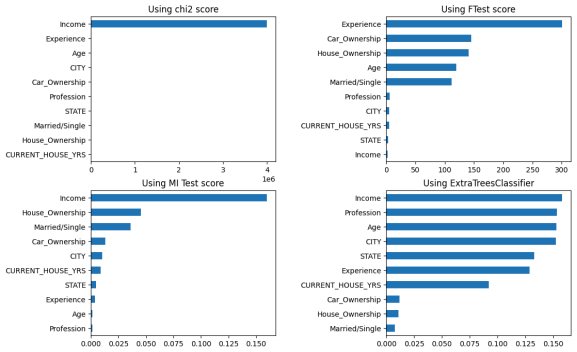


*Figure 4: Strength of Association between Feature Variables and Target Variable*

# IV. METHODOLOGY

To forecast potential defaulters for consumer loan products, we utilize two supervised learning models: K-Nearest Neighbors (KNN) and Decision Tree. Given that our dataset contains labeled data, supervised machine learning algorithms are chosen. This selection ensures that the models can learn from the provided labels to make accurate predictions.

## K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a versatile machine learning algorithm commonly used for classification and regression tasks. In KNN, the prediction for a new data point is based on the majority class (for classification) or the average value (for regression) of its nearest neighbors in the training dataset. The parameter "k" in KNN signifies the number of neighbors considered, which greatly influences the model's performance. KNN operates on the principle that similar data points tend to have similar outcomes, making it particularly effective for datasets with clear patterns or clusters. One of the main strengths of KNN is its simplicity and ease of implementation, as it doesn't require extensive model training. However, selecting an appropriate value for "k" and determining the optimal distance metric are crucial steps in maximizing KNN's predictive accuracy. [7]

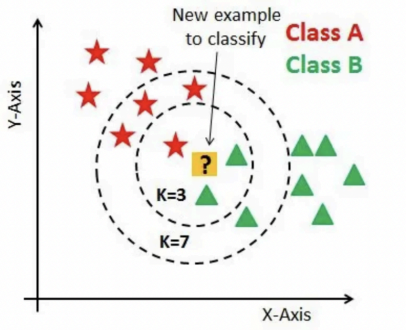


*Figure 5: Illustration of a K-NN Algorithm*

## Decision Tree

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It builds a tree-like structure by recursively partitioning

the data into subsets based on the values of input features. At each node of the tree, a decision is made based on a feature's value, aiming to maximize the homogeneity of the resulting subsets in terms of the target variable. This process continues until certain stopping criteria are met, such as reaching a maximum tree depth or no further improvement in homogeneity. Decision Trees are interpretable, allowing for easy visualization and understanding of decision rules, making them particularly useful for tasks where interpretability is essential. [8]

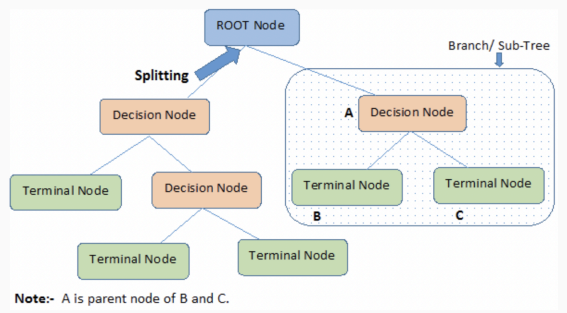


*Figure 6: Illustration of Decision Tree Algorithm*

# V. Testing and Results

The dataset is split into training and testing sets using the train_test_split function from the sklearn library, with a test size of 20% and a random state of 42. This indicates that 20% of the data is allocated for testing, while the remaining 80% is used for training the model. This approach ensures that a portion of the data is reserved for final accuracy assessment without influencing the model training process, adhering to the best practice to prevent the leakage of test data characteristics into model training.

**Cross-validation**

Employing the cross_val_score function from sklearn, the dataset is partitioned into training and validation sets with a ratio of 80:20. This process is repeated iteratively, with the model being trained and evaluated on different combinations of these sets. Consequently, it furnishes an approximation of the model's performance across various data subsets. The mean cross-validation accuracy, alongside its standard deviation, is computed to evaluate the consistency in the model's performance and provide a more robust assessment of its accuracy.

**Training**

After cross-validation, the model is trained using all the data in the training set, and its performance is evaluated on both the training and testing datasets. The accuracy on the training data is computed to evaluate the model's performance on the data it was trained on, providing insights into potential overfitting. Subsequently, the model is applied to the testing data to predict target values, and various evaluation metrics, including accuracy, F1-score, precision, and recall, are calculated to assess the model's predictive performance on unseen data.

Additionally, a confusion matrix is generated to visualize the model's classification performance on the testing data [see Figure 7 & 8]. The confusion matrix provides a tabular representation of the model's predictions compared to the actual outcomes, enabling a detailed analysis of the model's classification accuracy and any misclassifications.

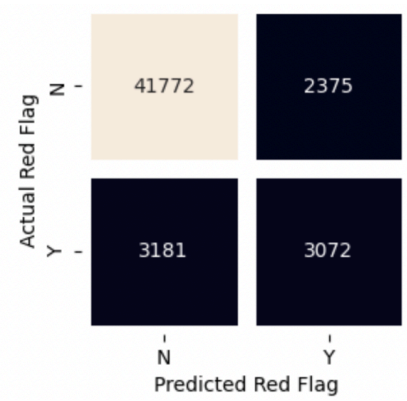


*Figure 7. Confusion matrix - KNN*


*Figure 8. Confusion matrix - Decision Tree*

## Results

*K-Nearest Neighbors (KNN) :*

The KNeighborsClassifier model achieved a training accuracy of 0.899, indicating its ability to predict the target variable accurately on the training dataset. Through cross-validation, the model demonstrated a mean accuracy of 0.889 with a standard deviation of 0.001, suggesting consistent performance across different validation folds.

Furthermore, when evaluated on the test dataset, the model yielded an accuracy of 0.890. Additional performance metrics include an F1 score of 0.525, precision of 0.564, and recall of 0.491. These metrics provide insights into the model's overall performance and its ability to correctly classify instances from the test dataset.

```
Results from algorithm KNeighborsClassifier():
Mean cross-validation accuracy is 0.889 with SD 0.001

Accuracy on training data is 0.899

Test data metrics: accuracy=0.890, f1=0.525, precision=0.564, recall=0.491
```

*Decision Tree :*

The Decision Tree classifier attained a training accuracy of 0.936, indicating its ability to accurately predict the target variable for the training data. Moreover, during cross-validation, the model demonstrated a mean accuracy of 0.881 with a standard deviation of 0.001, suggesting consistent performance across different validation folds. Additional metrics include an F1 score of 0.542, precision of 0.516, and recall of 0.572.

```
Results from algorithm DecisionTreeClassifier():
Mean cross-validation accuracy is 0.881 with SD 0.001

Accuracy on training data is 0.936

Test data metrics: accuracy=0.880, f1=0.542, precision=0.516, recall=0.572
```

The plot_tree function is also applied to provide a visual representation of the decision tree model. It displays the hierarchical structure of the tree, illustrating the decision-making process at each node based on the features. This visualization allows for the interpretation of how the model partitions the feature space and makes predictions.

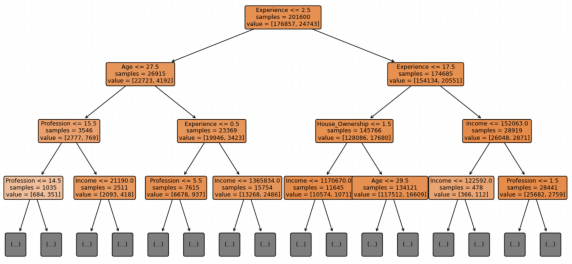

*Figure 9. Visualize a Decision Tree*

# VI. Conclusion

In this study, machine learning techniques, specifically K-Nearest Neighbors (KNN) and Decision Tree algorithms, were employed to predict potential defaulters for consumer loan products based on customer behavior.

The analysis revealed promising results, with both models demonstrating strong predictive performance across various evaluation metrics. The KNN model achieved a mean cross-validation accuracy of 0.889 and performed consistently well on the test dataset, yielding an accuracy of 0.890. Similarly, the Decision Tree model exhibited a mean cross-validation accuracy of 0.881 and achieved an accuracy of 0.880 on the test dataset.

Furthermore, both models demonstrated high training accuracies, indicating their ability to learn from the provided data and make accurate predictions. The Decision Tree model, in particular, showcased superior training accuracy of 0.936.

These findings underscore the potential of machine learning in improving loan prediction accuracy and aiding financial institutions in making informed lending decisions. By leveraging historical customer behavior data, these models can assist institutions in identifying customers more likely to default, thereby enabling proactive risk management strategies and enhancing decision-making processes.

Moving forward, further research could explore the integration of additional features or more advanced machine learning techniques to enhance predictive performance and address the evolving challenges in loan prediction and risk assessment within the financial industry. Additionally, validating the models' performance on diverse datasets and real-world scenarios would contribute to their robustness and applicability in practical settings.

# VII. References


[1] Zuama, R.A., Ichsan, N., Pohan, A.B., Azis, M.S. and Lase, M. (2024). An implementation of machine learning on loan default prediction based on customer behavior. https://ejournal.seaninstitute.or.id/index.php/InfoSains/article/view/3593

[2] Chen, H. (2022). Prediction and Analysis of Financial Default Loan Behavior Based on Machine Learning Model. https://doi.org/10.1155/2022/7907210

[3] Ndayisenga, T. (2021). Bank Loan Approval Prediction Using Machine Learning Techniques. https://dr.ur.ac.rw/handle/123456789/1437

[4] Pang, S. and Yuan, J. (2018). WT Model & Applications in Loan Platform Customer Default Prediction Based on Decision Tree Algorithms. Intelligent Computing Theories and Application, pp.359–371. https://doi.org/10.1007/978-3-319-95930-6_33.

[5] Serengil, S.I., Imece, S., Tosun, U.G., Buyukbas, E.B. and Koroglu, B. (2021). A Comparative Study of Machine Learning Approaches for Non Performing Loan Prediction. https://doi.org/10.1109/UBMK52708.2021.9558894

[6] www.kaggle.com. (n.d.). Loan Prediction Based on Customer Behavior. https://www.kaggle.com/datasets/subhamjain/loan-prediction-based-on-customer-behavior/data

[7] Christopher, A. (2021). K-Nearest Neighbor. K-Nearest Neighbor Algorithm https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4

[8] Nagesh Singh Chauhan (2020). Decision Tree Algorithm, Explained - KDnuggets. https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html