https://doi.org/10.47392/IRJAEH.2025.0058

Machine Learning Based Loan Eligibility Prediction and Automation

Meenakshi A[1], *Niranjana P*[2], *Pavitra Rao S*[3], *Dhivya G*[4]
[1,2,3,4]*Department of CSE, Kamaraj college of Engineering and Technology, Virudhunagar, India.*
Emails: *meenakshicse@kamarajengg.edu.in*[1], *21ucs034@kamarajengg.edu.in*[2], *21ucs051@kamarajengg.edu.in*[3], *21ucs004@kamarajengg.edu.in*[4]

Abstract

As the demand for bank loans rises, banks receive more loan applications every day. To determine who qualifies, they carefully assess each applicant's credit score and overall financial risk. However, even with these rigorous assessments, some borrowers still fail to repay their loans leading to significant financial losses for banks. To address this, challenge an advanced solution in a web application based on machine learning in the automation of loan evaluation and improves decision-making. It uses an historical model of loan to analyze key financial features such as the customer's credit history, status of income, employment status, and debt to income ratio that will help appropriately understand the qualifications of the applicants with the result of automating much of the processes, the solution will reduce heavy manual loads increase efficiency in deciding speed, consistency, and more transparency. The web application provides real- time insights and instant decision results, hence enabling easier communication with applicants and a more excellent customer experience. The project perfectly aligns with the digital transformation goals of the bank and presents a scalable solution that responds to changes in regulatory and market conditions. The bank is now able to portray itself as an innovator in financial services due to the adoption of machine learning in the automated decision-making process and is better placed to be responsive to customer needs while reducing operational costs. It modernizes loan evaluations, but basically, it is aligned with strategic objectives as it establishes customer trust.

Keywords: *Loan application, Loan Prediction, Machine learning, Classification, Data analysis, Scalability, Customer trust, Innovation in banking, Operational efficiency*

1. Introduction

The most important banking operations, such as loan application screening and approval, can consume too much time. We propose a design for a machine learning- based system linked to a web application that could classify loan applications efficiently and accurately. By leveraging historical data, the system enhances predictive accuracy and minimizes the risk of loan defaults.

1.1 Importance of the work

Automating the loan application evaluation process brings substantial benefits by reducing the time and effort that bank staffs need to spend on manual reviews. An automated system not only minimizes human error but also ensures that each application is evaluated fairly and consistently, free from bias. This approach leads to faster and more accurate decisions, allowing customers to receive quicker responses on their applications, which can be especially valuable during times of financial need. Ultimately, this automation helps the bank operate more efficiently and focus its resources on supporting customers and growing its services. [1-5]

1.2 Objective

The objective of this paper is to develop a machine learning model integrated with a web application that can be used by the bank to classify loan applications efficiently. The primary goal is to create a system that accurately predicts whether a loan applicant should be approved or rejected based on various factors such as credit history, income level, employment status, and other relevant financial indicators. The predicted machine learning model will be trained on historical loan data to learn patterns and relationships between the input features and the target variable. The system is designed to increase operational efficiency by automating the loan application review process, reducing the manual workload, and ensuring consistent decision-making.

1.3 Project Description and Features

Our proposed system Loan Application Recommendation System will use machine learning to streamline the loan evaluation process by analyzing key factors like credit history, income, and debt-to-income ratio. With a user- friendly web application interface, bank staff can quickly input applicant data and receive real-time decisions on approvals or rejections. This approach makes loan processing faster, more accurate, and consistent, helping customers get timely responses while also allowing bank personnel to focus on delivering a better overall service experience. Additionally, the system's scalability ensures it can adapt to growing customer demands and changing regulatory requirements.

1.4 Social Impact

Automating loan evaluations brings fairer decisions by reducing human bias and speeding up the approval process, giving applicants quicker access to funds. It ensures consistent evaluations across all applications, making loan approvals more reliable. For businesses, this means faster funding, while personal banking customers benefit from quicker decisions on loans, credit cards, and mortgages, improving their overall experience.

1.5 Challenges

Accessing high-quality, well-structured financial data is vital for training effective machine learning models in loan evaluation. However, gathering large and diverse datasets can be difficult due to privacy concerns and the varying formats of financial records. Additionally, models trained on one set of data may not perform well with new applicants or different financial situations. It's essential to ensure that the system is robust and can accurately assess applications from various demographics and financial backgrounds to work effectively in real-world banking scenarios.

1.6 Limitations

Machine learning has several limitations that affect its real- world effectiveness. It heavily depends on high-quality data, and poor or biased data can lead to inaccurate predictions. Many models struggle with over-fitting, performing well on training data but failing on new data. Feature engineering often requires domain expertise, making model development time-consuming. Additionally, scalability issues make real-time adaptation challenging.

1.7 Organization of the Report

The report gives a clear overview of the project's goals and key outcomes. It starts with an introduction explaining the challenges in loan processing and why automation is needed. The literature review looks at existing methods and their limitations. The methodology section describes how data is collected, cleaned, and used to build a machine learning model. The system design explains how the model is integrated into a web application for real-time loan classification. The implementation part covers model training, testing, and deployment. The results section evaluates the model's accuracy and efficiency. The conclusion summarizes the project's impact on banking operations. [6-10]

2. Literature Survey

A strong and efficient software method that classifies based on 13 properties of the data from Kaggle was developed in the paper [1] by Sharayu Dosalwar et al. The following models were examined: SVM, Decision Tree, Random Forest, Logistic Regression, XGBoost, KNN, and Naive Bayes. Given the lucidity and adaptability of its mathematics, it is discovered that Logistic Regression provides greater accuracy, yielding an accuracy of 78.5%. Vaidya [2] talks about logistic regression and how to represent it mathematically. His study employs logistic regression as a machine learning technique to actualize the predictive and probabilistic methods to a particular problem of loan approval prediction. This study employs logistic regression to determine if a loan for a set of records belonging to an applicant will be authorized. It also covers some of the Machine Learning mode's other real- world uses.

2.1 Methodology Used

The methodology for automating the evaluation of loan applications involves a systematic, multi-step process that integrates advanced technologies and established techniques. Key components of this methodology include:

2.1.1 Data Collection

Gather a diverse dataset of loan applications, including credit scores, income levels, and other relevant criteria. The dataset should represent various demographics and financial situations to improve the model's ability to generalize.t

2.1.2 Exploratory Data Analysis

Analyze the dataset to understand its structure and patterns. By visualizing key features like credit history and income levels, we can identify trends that guide our feature selection for accurate loan evaluations.

2.1.3 Feature Selection

Feature selection involves identifying the most important criteria that influence loan approval decisions. We will evaluate variables such as credit scores, income levels, and debt-to-income ratios to determine which features significantly impact eligibility. By narrowing down to these key features, we can improve the model's efficiency and effectiveness.

2.1.4 Model Selection and Training

Choose suitable machine learning algorithms, like decision trees or logistic regression, and train the model on the prepared dataset to distinguish between approved and declined applications. Additionally, fine-tune the model's hyper-parameters to optimize performance and improve its ability to generalize on unseen data.

2.1.5 Model Evaluation and Risk Scoring

Evaluate model performance using accuracy, precision, and recall. Implement a risk scoring mechanism to assess the likelihood of loan defaults based on applicant features.

2.1.6 Prediction and Decision Making

Once the model is trained and evaluated, it will be used to make predictions on new loan applications. The system will automatically analyze incoming data against the established criteria, determining which applications meet the bank's approval requirements. This automation streamlines the decision-making process, reducing delays and human error. customer demands and changing regulatory.

2.2 Merits

Automating the loan application evaluation process has several important benefits. First, it helps reduce bias by ensuring that every application is treated fairly, leading to better outcomes for all applicants. Second, it speeds up the loan processing time, allowing people to get access to funds more quickly, which can be very important during emergencies. Third, automation provides consistency in how applications are evaluated, making sure everyone is judged by the same standards and reducing mistakes. Additionally, businesses can benefit from a faster evaluation process, helping them secure funding more easily for their growth. Finally, this system simplifies the approval process for personal loans, credit cards, and mortgages, making it easier for customers and improving their overall experience with the bank.

3. Requirements

3.1 Software Requirements

Google Colab: Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

3.2 Python Packages

Web frame Work: Flask is a micro web framework for Python, designed specifically for building web applications and APIs. It is commonly used to create simple web applications and RESTful APIs, serving as a backend for various web-based projects. One of its key advantages is its lightweight nature, making it easy to use and allowing developers to quickly set up a web server. Flask supports essential features like routing, templating, and request handling, making it an ideal choice for building and deploying web services and applications efficiently, without the need for extensive setup. Figure 1 shows Design of Dataset, Figure 2 shows Heat Map. scalability ensures it can adapt to growing customer demands and changing regulatory requirements.

4. System Design

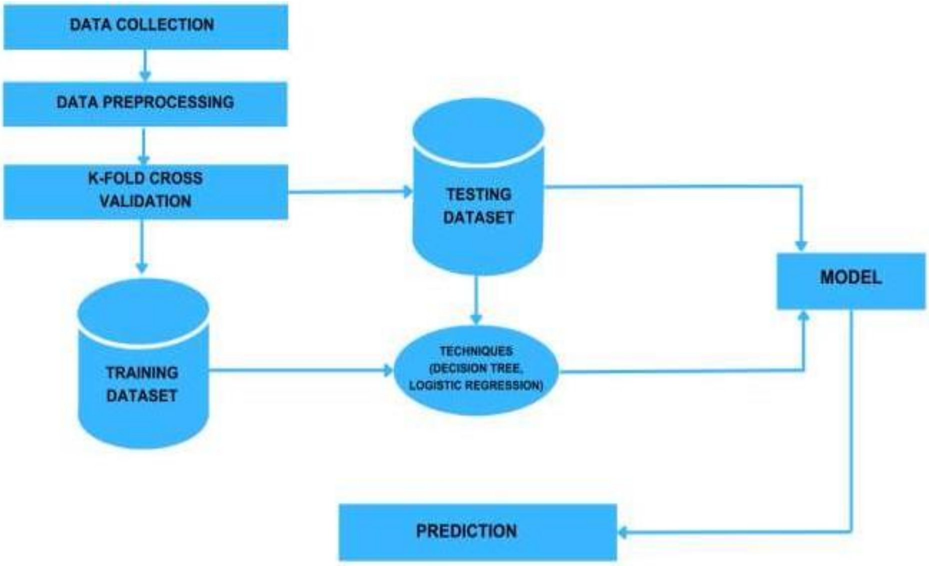


Figure 1 **Design of Dataset**

5. Implementation

Implemented a predictive model for loan statuses using Logistic Regression and enhanced it with Stratified K-Folds Cross Validation. The initial model was trained on a dataset containing various features relevant to loan approval. I utilized the Scikit-learn library to preprocess the data, which included creating dummy variables for categorical features and splitting the dataset into training and validation sets. The model's performance was evaluated using accuracy and F1 scores, and I visualized the results using ROC curves and AUC scores to assess its effectiveness. To further enhance the model's robustness, I employed Stratified K-Folds Cross Validation, ensuring a balanced representation of classes in each fold. This approach allowed me to validate the model's predictions and improve its generalization to unseen data.

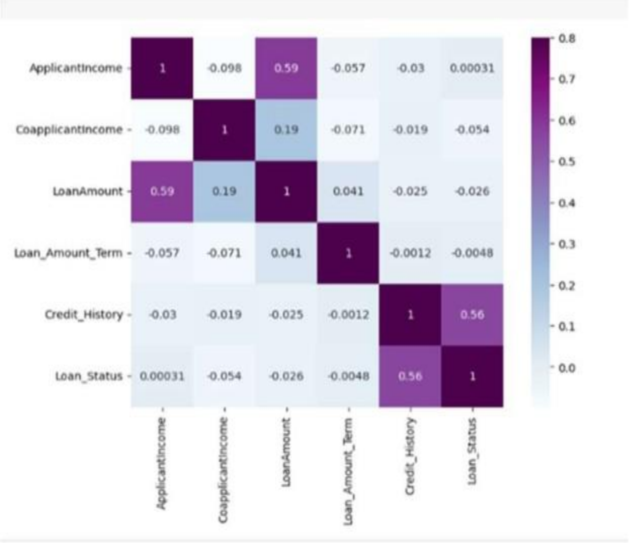
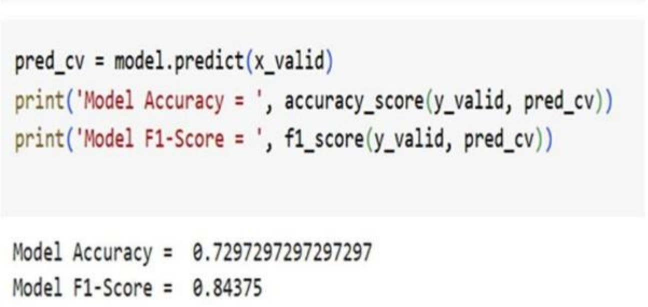


Figure 2 **Heat Map**

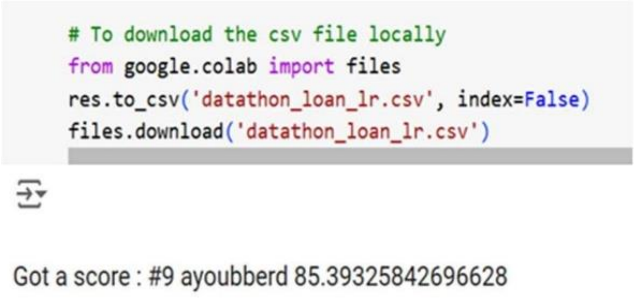
6. Results

The system design explains how the model is integrated into a web application for real-time loan classification. The implementation part covers model training, testing, and deployment. The results section evaluates the model's accuracy and efficiency. The conclusion summarizes the project's impact on banking operations.



```
pred_cv = model.predict(x_valid)
print('Model Accuracy = ', accuracy_score(y_valid, pred_cv))
print('Model F1-Score = ', f1_score(y_valid, pred_cv))
```

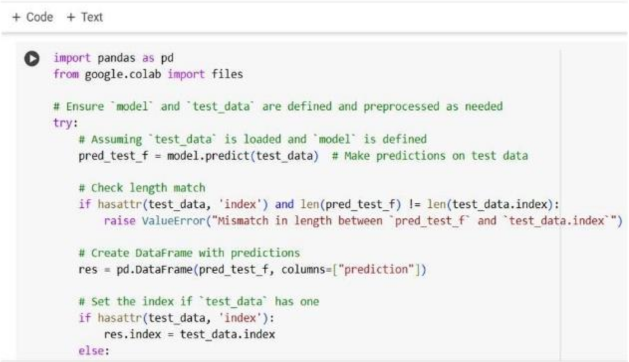
```
Model Accuracy =  0.7297297297297297
Model F1-Score =  0.84375
```



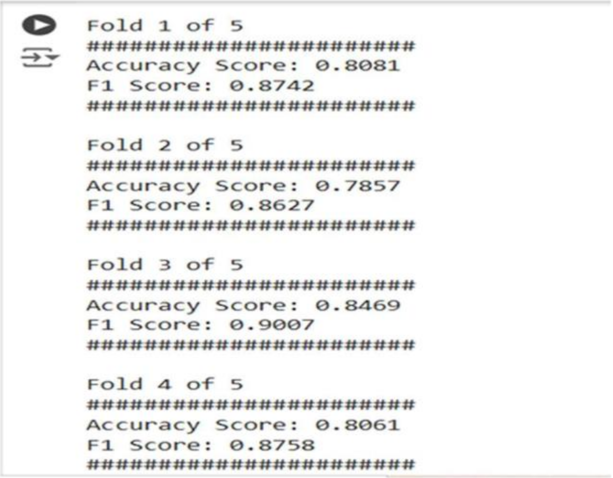
```
# To download the csv file locally
from google.colab import files
res.to_csv('datathon_loan_lr.csv', index=False)
files.download('datathon_loan_lr.csv')
```

Got a score : #9 ayoubberd 85.39325842696628

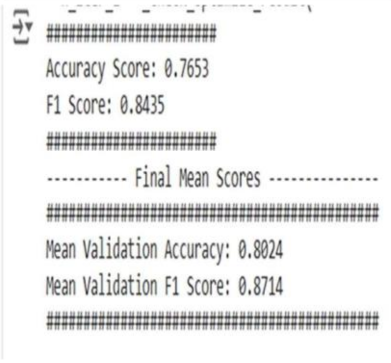
6.1 Logistic Regression Using Stratified K-Fold Cross Validation



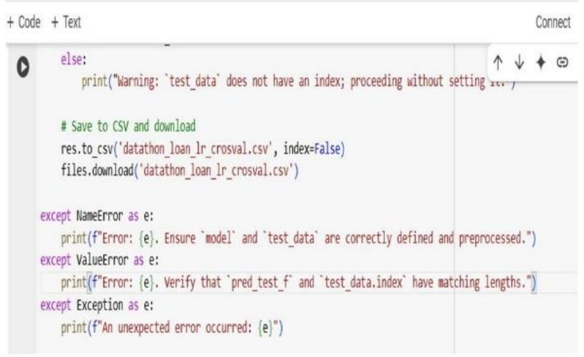
```
import pandas as pd
from google.colab import files

# Ensure `model` and `test_data` are defined and preprocessed as needed
try:
    # Assuming `test_data` is loaded and `model` is defined
    pred_test_f = model.predict(test_data)  # Make predictions on test data

    # Check length match
    if hasattr(test_data, 'index') and len(pred_test_f) != len(test_data.index):
        raise ValueError("Mismatch in length between `pred_test_f` and `test_data.index`")

    # Create DataFrame with predictions
    res = pd.DataFrame(pred_test_f, columns=["prediction"])

    # Set the index if `test_data` has one
    if hasattr(test_data, 'index'):
        res.index = test_data.index
    else:
```



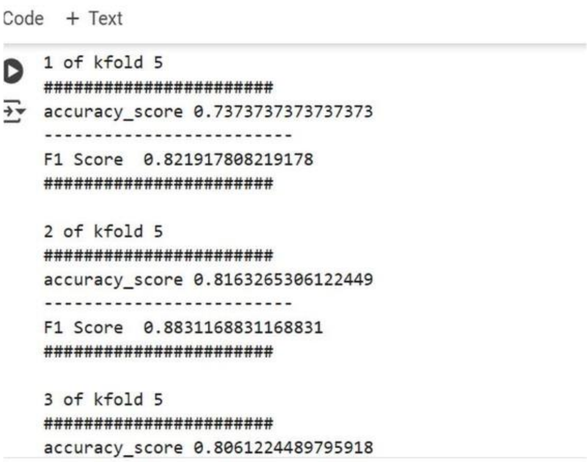
```
Fold 1 of 5
##########################
Accuracy Score: 0.8081
F1 Score: 0.8742
##########################

Fold 2 of 5
##########################
Accuracy Score: 0.7857
F1 Score: 0.8627
##########################

Fold 3 of 5
##########################
Accuracy Score: 0.8469
F1 Score: 0.9007
##########################

Fold 4 of 5
##########################
Accuracy Score: 0.8061
F1 Score: 0.8758
##########################
```



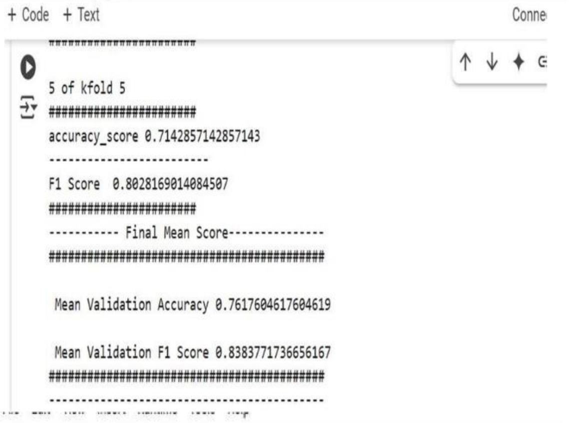
```
####################
Accuracy Score: 0.7653
F1 Score: 0.8435
####################
----------- Final Mean Scores --------------
########################################
Mean Validation Accuracy: 0.8024
Mean Validation F1 Score: 0.8714
########################################
```



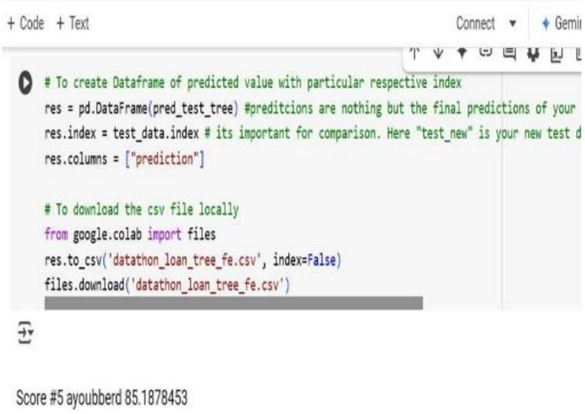
```
    else:
        print("Warning: `test_data` does not have an index; proceeding without setting it.")

    # Save to CSV and download
    res.to_csv('datathon_loan_lr_crosval.csv', index=False)
    files.download('datathon_loan_lr_crosval.csv')

except NameError as e:
    print(f"Error: {e}. Ensure `model` and `test_data` are correctly defined and preprocessed.")
except ValueError as e:
    print(f"Error: {e}. Verify that `pred_test_f` and `test_data.index` have matching lengths.")
except Exception as e:
    print(f"An unexpected error occurred: {e}")
```



```
1 of kfold 5
########################
accuracy_score 0.7373737373737373
-------------------------
F1 Score  0.821917808219178
########################

2 of kfold 5
########################
accuracy_score 0.8163265306122449
-------------------------
F1 Score  0.8831168831168831
########################

3 of kfold 5
########################
accuracy_score 0.8061224489795918
```



```
5 of kfold 5
######################
accuracy_score 0.7142857142857143
------------------------
F1 Score  0.8028169014084507
######################
----------- Final Mean Score--------------
########################################

 Mean Validation Accuracy 0.7617604617604619

 Mean Validation F1 Score 0.8383771736656167
########################################
----------------------------------------
```



```
# To create Dataframe of predicted value with particular respective index
res = pd.DataFrame(pred_test_tree) #preditcions are nothing but the final predictions of your
res.index = test_data.index # its important for comparison. Here "test_new" is your new test d
res.columns = ["prediction"]

# To download the csv file locally
from google.colab import files
res.to_csv('datathon_loan_tree_fe.csv', index=False)
files.download('datathon_loan_tree_fe.csv')
```

Score #5 ayoubberd 85.1878453

6.2 Model Accuracy

Table 1 Accuracy

Model	Accuracy(%)
Logistic Regression	72.97
Logistic Regression (Stratified K-Fold Cross Validation)	80.24
Decision Tree	76.17

Logistic Regression with Stratified K-Fold Cross Validation is the best as it gives the highest accuracy of 80.24%.

7. Result

In this loan application recommendation system, we collected and preprocessed data, performed exploratory data analysis, and selected key features influencing loan approval. We then trained multiple machine learning models, including logistic regression, logistic regression with stratified k-fold cross-validation, and decision tree models. Among these, logistic regression with stratified k-fold cross-validation provided the highest accuracy of 80.24% by handling data imbalance effectively. Model evaluation using accuracy, precision, recall, and F1-score confirmed its reliability. A user-friendly front-end was developed where users can input details like income, credit history, and loan amount to receive an instant loan approval prediction along with a risk score. This approach ensures an efficient, accurate, and accessible system for loan recommendation.

8. Discussion

This project aims to streamline loan application evaluations by automating the assessment process, which is currently time-consuming and labor-intensive. By using a system that analyzes criteria like credit history and income, the bank can achieve faster, more accurate, and consistent loan decisions. This automation minimizes human error, reduces processing time, and allows bank staff to focus on more complex tasks, ultimately enhancing efficiency and customer satisfaction. Challenges like data quality and model transparency remain, but the system holds strong potential to improve the bank's operational workflow and decision accuracy.

Conclusion

In conclusion, automating the evaluation of loan applications presents a significant opportunity for the bank to enhance operational efficiency and accuracy. By implementing a system that analyzes critical factors like credit history and income level, the bank can streamline its decision-making process, significantly reducing the time and effort involved in manual reviews. This not only ensures quicker access to funds for applicants but also minimizes the risk of human error, leading to fairer and more consistent loan approvals. Ultimately, this automation initiative will position the bank to better serve its customers while optimizing its internal operations.

References

[1]. S. Sobana and P. J. L. Ebenezer, "A comparative study on machine learning algorithms for loan approval prediction analysis," Int. Res. J. Mod. Eng. Technol. Sci., vol. 4, no. 12, pp. 565-569, Dec. 2022.

[2]. Bhattad S, S. Bawane, S. Agrawal, U. Ramteke, and P. B. Ambhore, "Loan prediction using machine learning algorithms," Int. J. Comput. Sci. Trends Technol., vol. 9, no. 3, pp. 143-146, 2023.

[3]. A. Kadam, S. Nikam, A. Aher, G. Shelke, and A. Chandgude, "Prediction for loan approval using machine learning algorithm," Int. Res. J. Eng. Technol., vol. 8, no. 4, pp. 4089-4092, 2023.

[4]. A. M. Miraz, A. Farjana, and M. Mamun, "Predicting bank loan eligibility using machine learning models and comparison analysis," Int. Res. J. Eng. Technol. (IRJET), vol. 8, no. 5, 2021.

[5]. A. Sarkar, "Machine learning techniques for recognizing the loan eligibility," in IRJMETS, vol. 3, no. 12, 2022.

[6]. P. Dutta, "A study on machine learning algorithm for enhancement of loan prediction," in *Int. Res. J. Modernization Eng. Technol. Sci. (IRJMETS)*, vol. 3, no. 1, 2021.

[7]. S. Dosalwar, K. Kinkar, R. Sannat, and N. Pise, "Analysis of loan availability using

machine learning techniques," in *Int. J. Adv. Res. Sci. Commun. Technol. (IJARSCT)*, vol. 9, no. 1, pp. 15-20, 2024.

[8]. N. Pandey, R. Gupta, S. Uniyal, and V. Kumar, "Loan approval prediction using machine learning algorithms approach," in *Int. J. Innov. Res. Technol. (IJIRT)*, vol. 8, no. 1, 2022.

[9]. Zhang H, Li Z, Shahriar H, Tao L, Bhattacharya P, and Qian Y, "Improving prediction accuracy for logistic regression on imbalanced datasets," in 2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC), vol. 1, pp. 918-919, Jul. 15, 2019.

[10]. J. Tejaswini, T. Mohana Kavya, R. Devi Naga Ramya, P. Sai Triveni, and Venkata Rao Maddumala, "Accurate loan approval prediction based on machine learning approach," *JES Publication*, vol. 11, no. 4, pp. 523, Apr. 2020, ISSN: 0377-9254.