ISSN 2063-5346

MACHINE LEARNING BASED ASSESSMENT OF MODERNIZED LOAN APPROVAL SYSTEM

Faiza Anwar[1] Mrs. Ashwini Bhaskar Gulhane[2]

Article History: Received: 10.05.2023 Revised: 29.05.2023 Accepted: 09.06.2023

Abstract

Humankind's survival and standard of living have been enhanced through technology. The proposal aims to produce new and original content every day. In the banking sector, the candidate receives proofs/backup prior to the loan being approved, therefore technology supports our lives and helps in making them better. The system's use of the individual's or entity's historical data determines whether a loan would be sanctioned or not. Numerous individuals apply for loans in either banks or financial institutions each day, yet the banking sector or financial institution resources are constrained. Using a classes-function algorithm in this situation would be quite advantageous. Several instances include support vector machine (SVM) classification, logistic regression, and random forest classifiers. The quantity of loans a bank makes or loses depends on how much the client or candidate pays back the loan. The most crucial task for commercial banks is loan recovery. The process of improvement is crucial in the banking industry. Utilizing several categorization techniques, a computational model was developed with historical information about the candidates. The major goal of this study is to use models of machine learning trained on historical information to forecast whether a new application will be approved for a loan.

Keywords—*Machine learning, Data, Loan, Training, Testing, Prediction*.

[1]Research Scholar, Dept. of Computer Science and Engineering, Lords Institute of Engineering & Technology, Hyderabad, Telangana

[2]Asst Professor, Dept. of Computer Science and Engineering, Lords Institute of Engineering & Technology, Hyderabad, Telangana

DOI:10.48047/ecb/2023.12.9.33

I. INTRODUCTION

The proposed system can determine whether a loan will be approved or not using the prediction of a modernized loan approval system that is based on a machine learning technique. Information is collected from the user for this system, such as his monthly salary, family status, moneylending, tenure, etc. The bank will then determine if the client will receive the loan based on its criteria. Therefore, there exists a classification scheme. In this framework, a training set is used to create the model, and the classifier is able to place the data objects into the correct class[2]. In order to train the data and provide the correct outcome, which is the client's potential and ability to repay the loan, a testing dataset is constructed. For banks and customers alike, the prediction of a modernized credit approval system is quite helpful[1]. This method evaluates each candidate based on their priority. The client can register his application form to the bank, in which case the bank will handle the entire procedure without interference from a third party or stockholder[3]. Finally, based on its priority system, the bank will determine if the applicant is deserving or not. This study paper's sole goal is to ensure that the worthy applicant receives straightforward answers right away.

II. RESEARCH BACKGROUND

A. Problem Statement

A significant issue is that many individuals are unable to repay bank debts. Additionally, banks are losing money. Every day, banks receive many loan applications, but not all of them are approved. The majority of banks or financial institutions use their own risk estimation & assessment methods and credit scoring systems to determine whether to sanction the loan. In a matter of minutes, the predicament of why there is a loan difficulty will be answered. The primary purpose of acquiring a loan would be to meet a need[4]. A loan is necessary for a businessman if they want to grow their company or if they need to recover from a loss. People in the middle class want to meet their demands so people would like to apply for credit leading. Therefore, the fundamental goal of this is to satisfy someone's or something's wants. Once more, the issue of what issues are developing in credit provision is raised. The response to this inquiry is that not everyone qualifies for loans because if the borrower is unable to repay the loan, either they themselves or the business or bank that provided it would suffer a loss[5]. Therefore, the person granting the loan must first confirm or establish some criteria as to whether the person receiving it is capable of paying back it or not. Like with banks, we do have a Credit lending bank card option, but not everyone qualifies for one. To determine eligibility for just that, a credit history is available. To be eligible for a loan, a person needs to have a decent credit score. A source of earnings and other requirements should be present to obtain a credit card. Banks offer loans, but the borrower must submit documentation and undergo verification. For example, when a company is unable to offer loans, banks suffer losses, and they are referred to as NBFCs.

B. Aim Of The Project

In this project, I aim to come up with an effective and efficient machine-learning model that can predict whether an individual can repay a loan. This system will help banks or financial institutions to make appropriate decisions and help in the process of approving loans faster. As every individual applicant’s information would contain multiple patterns, machine learning models are the best way to predict their eligibility and their capacity to repay loan amounts specific to their capacity and tenure. This would be of immense help as banks or other financial institutions can process loans faster and have a better percentage of the recovery.

C. Scope Of The Project

Background verification should be high so that we can expect a return of the loan at the perfect time. So, we analyze several factors and these are called our input variables.

This proposed model will characterize the behavior of customers based on their records. These records are taken from the customers to create a data set. With the help of these data sets and training machine learning models, we predict whether the customer's loan will be approved or not. This Machine algorithms predict the possibility of whether a customer would be able to repay the loan or not

Technical Approach

The technological strategy to solve the issue is listed below:

1. Dataset identification
2. Analysis of Exploratory Data
3. Dataset preparation
4. Running the dataset through many algorithms to see which one best fits the situation.
5. Developing a final classifier model and training the final classifier
6. Validating the ultimate classifier and recording the outcomes.

III. SYSTEM ANALYSIS

A. Research Gap

In the banking industry, the candidate receives proofs/backup prior to acceptance of the loan amount. Folks utilize gadgets to assist our existence and make us more or less complete. The system's historical data about the candidate determines whether or not the application is granted. Numerous people ask for credit lending every day in the banking industry, yet the bank's resources are constrained. The people who truly deposit in a bank would be impacted if the credit lending's not retrieved, which could result in losses for the institutions. The current conventional processes are unable to determine whether or not the sanctioned credit lending may be properly retrieved.

B. Proposed System

The suggested model focuses on forecasting client repayment reliability for loans by examining their behavior. The client behavior that was gathered serves as the model's input. One can decide whether one should accept or decline the customer request based on the classifier's output. Loan prediction and severity can be predicted using various data analytics technologies. In order to anticipate the type of loan, it is essential to train this data using various algorithms before comparing it to user data. to identify commonalities in a dataset of often authorized loans, and then to create a predictive model focusing on these identified patterns. The test data is now sent to the machine learning model, and the model is built using this data set. Each new applicant's application form information serves as just a test data set.

The question of how we evaluate whether or not to grant the loan emerges. We supply the credit to our consumers based on two goal criteria. We must verify all the requirements, including evidence of income, address, and identification. The applicant is then given the loan, whether they are eligible to return it or not. The middle class has a significant need for loans because parents need them for their children's education as well as for their businesses. Some people have abrupt financial crises, while others attempt to defraud banks of their money. As a result, we must double-check everything just because lenders are not experiencing NPA loans. Higher possibilities of loan repayment are associated with better customers. Background checks should be thorough therefore that we can anticipate receiving the loan back at the ideal moment. We analyze data on a variety of bases, and these are referred to as our target variables.

This suggested approach will evaluate consumer behavior based on their past conduct. These client records are collected to form a data set. We make predictions about whether or not the customer's loan will be approved using these data sets and a machine learning model that has been trained. These computer algorithms forecast the likelihood that a consumer will be able to pay back the lending credit or not.

After testing, the model determines whether the new application is a good candidate for loan approval or not based on the inference it draws from the training sets of data to determine if a client would indeed be capable of repaying his loan or not

Advantages Of Proposed System

- High precision
- Extendable to real-time settings.

IV. ALGORITHMIC PROCESS

A. Creating Model

The algorithm used: Decision Tree algorithm

Below is the technical approach used for the Loan recommendation system using the Loan review dataset

1. Data cleaning and visualization
2. Feature Extraction
3. Data Preparation
4. Splitting the data set into test data and training data
5. Data modeling using Sequential and Decision Tree algorithm classifier
6. Data Evaluation and prediction
7. Building the prediction of a modernized loan approval system

1) Feature extraction:

Below are the features present in the dataset:

Feature Name	Type
Loan_ID	String
Gender	String
Married	String
Dependents	Int
Education	String
Self_Employed	String
ApplicantIncome	Int
CoapplicantIncome	Int
LoanAmount	Int
Loan_Amount_Term	Int
Credit_History	Int
Property_Area	String

Feature Extraction Details:

A. Imputing Missing Values

When no information is given for one or more elements, a whole unit, or both, this is known as missing data. Missing data is a major issue in real-world situations. In pandas, missing data can also refer to NA (Not Available) values. Many datasets in DataFrame occasionally arrive with missing data, either because the data was never collected or because it was present but was not captured. We can use the fillna(), replace(), and interpolate() functions to fill in any null values in a dataset by replacing NaN values with one of their own. Each of these functions aids in filling in null values in a data frame's datasets. Interpolate() function is basically used to fill NA values in the data frame but it uses various interpolation techniques to fill the missing values rather than hard-coding the value. Although it uses a variety of interpolation algorithms rather than hard-coding the value, the Interpolate() function is mostly used to fill NA values in data frames.

```
for the column in ['Gender', 'Dependents', 'Married', 'Education',
'Self_Employed', 'Loan_Amount_Term',
                  'Credit_History']:
    dataset[column].fillna(dataset[column].mode()[0], inplace=True)

dataset['LoanAmount']                                          =
dataset['LoanAmount'].fillna(np.nanmedian(dataset['LoanAmount']))
```

B. Label Encoding

In machine learning, we usually deal with datasets that contain multiple labels in one or more than one column. These labels can be in the form of words or numbers. To make the data understandable or in human-readable form, the training data is often labeled in words.

Label Encoding refers to converting the labels into a numeric form so as to convert them into a machine-readable form. Machine learning algorithms can then decide in a better way how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.

```
for column in ['Gender', 'Dependents', 'Married', 'Education',
'Self_Employed', 'Credit_History']:S
    encoder = LabelEncoder()
    dataset[column] = encoder.fit_transform(dataset[column])

encoder = LabelEncoder()
dataset['Loan_Status'] =
encoder.fit_transform(dataset['Loan_Status'])
```

C. One Hot Encoding

Most Machine Learning algorithms cannot work with categorical data and needs to be converted into numerical data. Sometimes in datasets, we encounter columns that contain categorical features (string values) for example parameter Gender will have categorical parameters like Male, Female. These labels have no specific order of preference and also since the data is string labels, machine learning models misinterpreted that there is some sort of hierarchy in them.

One approach to solve this problem can be label encoding where we will assign a numerical value to these labels for example Male and Female mapped to 0 and 1. But this can add bias in our model as it will start giving higher preference to the Female parameter as 1>0 and ideally both labels are equally important in the dataset. To deal with this issue we will use the One Hot Encoding technique.

One Hot Encoding:

In this technique, the categorical parameters will prepare separate columns for both Male and Female labels. So, wherever there is a Male, the value will be 1 in the Male column and 0 in the Female column, and vice-versa.

```
dummy_data = pd.get_dummies(dataset['Property_Area'])
dataset = pd.concat([dataset, dummy_data], axis=1)
dataset.drop(['Property_Area'], axis=1, inplace=True)
dataset.drop(['Loan_ID'], axis=1, inplace=True)

X = dataset.drop(['Loan_Status'], axis=1)
y = dataset['Loan_Status']
```

D. Defining features and Labels

```
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=300)
```

E. Generating Synthetic Samples

Imbalanced Data Distribution is a phrase used frequently in machine learning and data science and refers to situations where observations in one class are significantly greater or lower than those in the other classes. Machine learning algorithms do not take the class distribution into account since they prefer to improve accuracy by decreasing the error.

SMOTE (Synthetic Minority Oversampling Technique) – Oversampling

One of the most popular oversampling techniques to address the imbalance issue is SMOTE (synthetic minority oversampling technique).

By increasing minority class samples at random and duplicating them, it seeks to balance the distribution of classes.

SMOTE combines already existing minority instances to create new minority instances. For the minority class, it creates virtual training records using linear interpolation. For each example in the minority class, one or more of the k-nearest neighbours are randomly chosen to serve as these synthetic training records. Following the oversampling procedure, the data is

rebuilt and can be subjected to several categorization models.

```
sm = SMOTE(random_state=300)
X_train, y_train = sm.fit_resample(X_train, y_train)
```

F. Scaling the data

Data Scaling is a data preprocessing step for numerical features. Many machine learning algorithms like Gradient descent methods, KNN algorithm, linear and logistic regression, etc. require data scaling to produce good results.

MinMax Scaler shrinks the data within the given range, usually from 0 to 1. It transforms data by scaling features to a given range. It scales the values to a specific value range without changing the shape of the original distribution.

```
scaler = MinMaxScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

V. PROJECT IMPLIMENTATION

A. Proposed Modular Implementation

Below is the proposed modular implementation of the project. It consists of modules:

1. Admin
2. User

1. Admin Module:

The admin of the system is responsible for the activities like:

a) Uploading the dataset
b) The dataset's data analysis
c) Divvying up the dataset into training and test halves
d) Developing the model with several algorithms
e) Examine how well the algorithms performed on the provided dataset.
f) Use the Decision Tree approach to create the model.

2. User Module:

The system's user may take advantage of the following available machine learning services:
logging in and entering fresh applicant loan information to forecast future trends

B. SYSTEM DESIGN

1. Data Flow Diagram: Admin

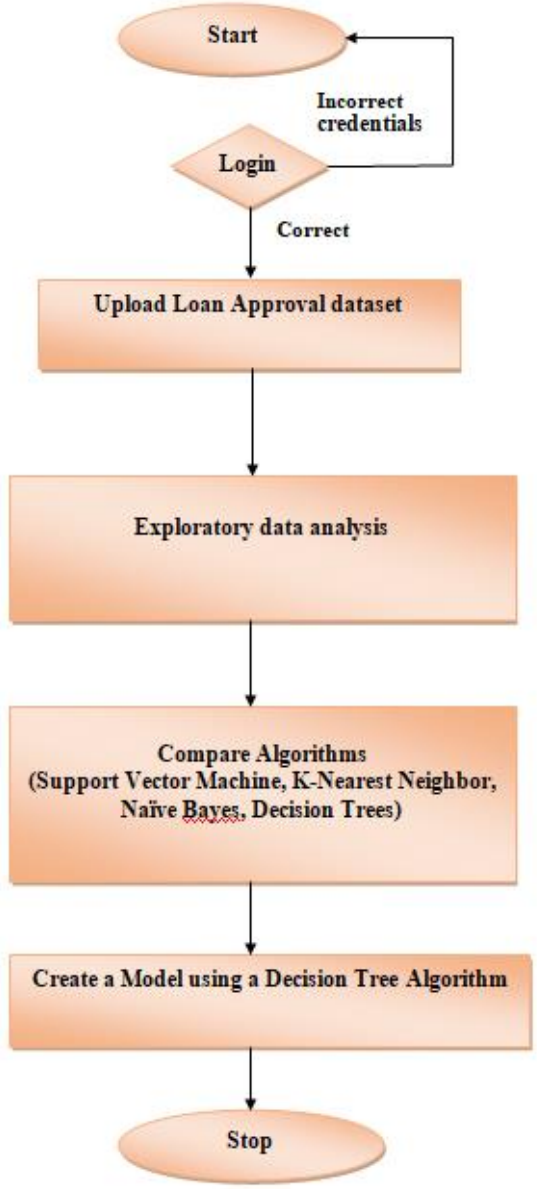

Figure 1: A Data Flow Diagram for Admin

2. Data Flow Diagram: User

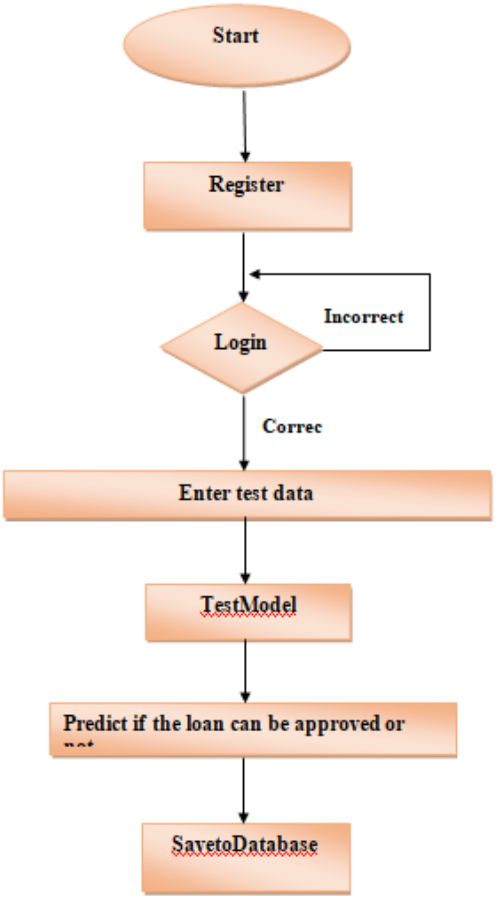

Figure 2: A Data Flow Diagram for Admin

VI. IMPLEMENTATION AND RESULT ANALYSIS

A. Project execution process:

1. Upload Dataset

On this page, the administrator of the system can upload datasets that are used for training the machine learning models. The admin has to select the file by clicking on the Choose file button and clicking on the upload button to upload the file to the server. Once the upload is complete, a success message would be displayed that the file is successfully uploaded. For this project, we are using Loan_Train.csv as a dataset.

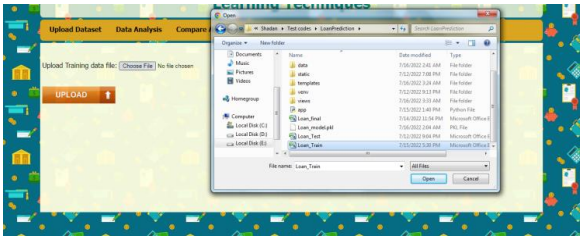

Figure 3: Upload Dataset

Figure 4: File uploaded

2. Data Analysis

Exploratory Data Analysis is performed on the dataset in order to clean the dataset for any missing data, identify patterns, and identify the relationships of various parameters of the outputs with the help of graphs, statistics, etc. so that Data Analysis can be performed.

a) Education Analysis:

The below graph shows the Education Analysis of an individual from the Training dataset Loan_Train.csv File.

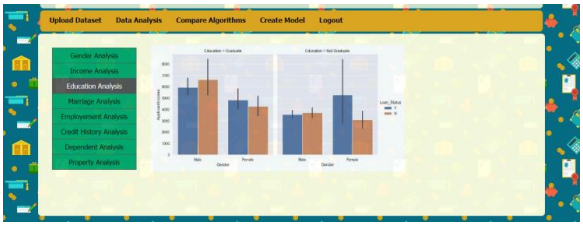

Figure 5: Education Analysis

3. Compare Algorithms

On this page, the admin can feed the dataset to various Algorithms to train them and get the test accuracy for each algorithm.

a) K-Nearest Neighbour:

When the dataset is fed to K-Nearest Neighbour algorithm we observe that the test accuracy is 85.36%

Figure 7: KNN

b) Decision Trees

The test accuracy is 86.99% when the dataset is fed to the decision tree method.

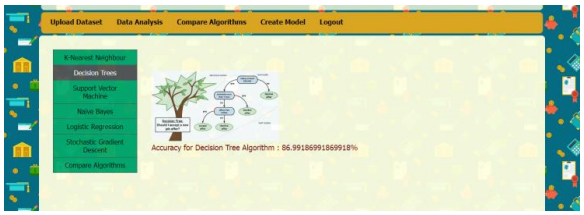

Figure 8: Comment Analysis

a) Support Vector Machine

The test accuracy is 86.99% when the dataset is fed into the support vector machine method, as we can see..

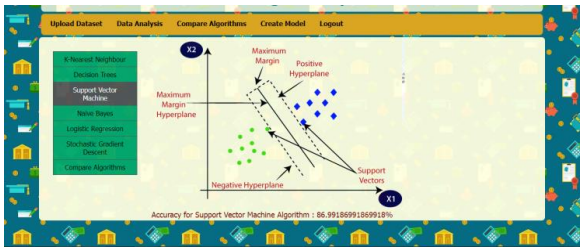

Figure 9: Support Vector Machine Algorithm

4. Create Model

The Generate Model button can be used to create the Model. After pressing the button, a success message is presented and the model is built. Our model's precision is 86.99%.

Figure 10: Create Model

5. Test Model:

This is in User Home Page for the user module. The user need to login into the system with his credentials in order to facilitate prediction over a of new applicant Loan data over dataset.

Figure 11: Test model

B. Metrics Evaluation :

Accuracy- One parameter for assessing classification models is accuracy. Informally, accuracy is the percentage of accurate predictions made by our model.

Macro avg - The macro average, also known as the precision, memory, and f1 score, is the arithmetic mean of each individual class. When all classes must be treated equally, macro average scores are used to assess the classifier's overall performance in comparison to the most popular class labels.

Weighted avg- A calculation that accounts for the varied levels of significance of the numbers in a data set is known as a weighted average.

Metrics for Algorithms in CompAlg.py

Classification report contains the complete metric information of the evaluated

algorithm. They are Precision, Recall, F1-Score, Support

Precision – What percent of your predictions were correct?

Precision is the capacity of a classifier to avoid classifying as positive anything that is in fact negative. It is described for each class as the proportion of true positives to the total of true and false positives.
TP – True Positives
FP – False Positives

Precision – Accuracy of positive predictions.
Precision = TP/(TP + FP)

1) Recall – What percent of the positive cases did you catch?

The capacity of a classifier to locate every successful instance is known as recall. It is described as the proportion of true positives to the total of true positives and false negatives for each class.

FN – False Negatives

Recall: Fraction of positives that were correctly identified.

Recall = TP/(TP+FN)

2) F1 score – What percent of positive predictions were correct?

The F1 score is a weighted harmonic mean of recall and precision, with 1.0 representing the best result and 0.0 the lowest. F1 scores typically perform worse than accuracy measures because they incorporate precision and recall into their computation. It is often recommended to compare classifier models using the weighted average of F1, rather than overall accuracy.

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

The amount of real instances of the class in the given dataset is known as support. The requirement for stratified sampling or rebalancing may be indicated by unbalanced support in the training data, which may point to structural flaws in the classifier's reported scores.

1) Metrics using Decision Tree Algorithm:
   Decision Trees accuracy:
   0.8699186991869918

   Decision Trees Classification Report:

	precision	recall	f1-score	support
0	0.90	0.56	0.69	32
1	0.86	0.98	0.92	91
accuracy			0.87	123
macro avg	0.88	0.77	0.80	123
weighted avg	0.87	0.87	0.86	123

2) Metrics using K-Nearest Neighbour Algorithm:
   KNN accuracy: 0.8536585365853658

   KNN Classification Report:

	precision	recall	f1-score	support
0	0.89	0.50	0.64	32
1	0.85	0.98	0.91	91
accuracy			0.85	123
macro avg	0.87	0.74	0.77	123
weighted avg	0.86	0.85	0.84	123

3) Metrics using Logistic Regression Algorithm:

Logistic Regression accuracy:

0.8536585365853658

Logistic Regression Classification Report:

	precision	recall	f1-score	support
0	0.85	0.53	0.65	32
1	0.85	0.97	0.91	91
accuracy			0.85	123
macro avg	0.85	0.75	0.78	123
weighted avg	0.85	0.85	0.84	123

4) Metrics using Support Vector Machine Algorithm:

SVM accuracy: 0.8699186991869918

SVM Classification Report:

	precision	recall	f1-score	support
0	0.94	0.53	0.68	32
1	0.86	0.99	0.92	91
accuracy			0.87	123
macro avg	0.90	0.76	0.80	123
weighted avg	0.88	0.87	0.86	123

VII. CONCLUSION AND FUTURE SCOPE

Conclusion : This project can determine whether the customer is likely to return the loan, as well as the accuracy is decent. Age, income, loan length, and loan amount are the most crucial variables when determining (whether the applicant would have been). Zip code and credit history are the two most crucial variables in determining the loan applicant's category. The standard of living for humans has improved thanks to technology. We intend to provide something fresh and unique every day. In the banking industry, the candidate receives proofs/backup prior to acceptance of the loan amount. Humans need gadgets to support our life and to make us somewhat complete.The system's historical data about the candidate determines whether or not the application is granted. Numerous people ask for loans every day in the banking industry, yet the bank's resources are constrained. A classes-function algorithm would be very helpful in this situation if the proper prediction could be made. Examples include support vector machine, logistic regression, and random forest classifiers. The quantity of loans a bank makes or loses depends upon whether the customer or the client pays back the loan. The most crucial task for the financial system is loan recovery. The process of improvement is crucial in the banking industry.Utilizing several categorization techniques, a computational model was constructed using the available information of the candidates. The major goal of this study is to use machine learning algorithms trained on historical information to determine whether a new application will be approved for a loan or not.

REFERENCES

[1] Amruta S. Aphale and R. Prof. Dr. Sandeep. R Shinde, “Predict Loan Approval in Banking System Machine Learning Approach for Cooperative Banks Loan Approval”, International Journal of Engineering Trends and Applications (IJETA), vol. 9, issue 8, 2020)

[2] Loan Prediction Using Ensemble Technique, International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 3, March 2016

[3] Exploratory data analysis https://en.wikipedia.org/wiki/Exploratory_data_analysis

[4] Pandas Library https://pandas.pydata.org/pandas-docs/stable/

[5] MeanDecreaseAccuracy

https://dinsdalelab.sdsu.edu/metag.stats/code/randomforest.hml

.