International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 07, ISSN No. 2455-2143, Pages 46-50
Published Online November 2023 in IJEAST (http://www.ijeast.com)

PREDICTING CREDIT WORTHINESS IN BANK LOAN APPROVAL PROCESS

Dr.S.Gomathi alias Rohini
Department of Artificial Intelligence and Machine Learning
Kongunadu Arts and Science College
Coimbatore, India

Dr.S.Mohanavel
Department of Management Studies
AJK College of Arts and Science
Coimbatore, India

Abstract - **Loans make up significant part in bank profits. In a bank, amongst large number of loan applications, it can be lengthy and challenging to choose genuine and eligible applications to approve the loan, if the process is done manually. The system was developed using Python 3.7 & its libraries and Jupyter Notebook cross-platform Integrated Development Environment (IDE). The developed system uses the machine learning algorithms - Naïve Bayes, Support Vector Machine (SVM) and XGBoost (Extreme Gradient Boosting) individually to classify the loan applications to automatically choose genuine and eligible applications. Based on average bank balance of the period and financial background, the system approves the loan amount to the predicted applicants with good credit history. The system's performance in predicting the credit worthiness of applicants was evaluated. Naive Bayes algorithm predicted the credit worthiness with 79% accuracy, SVM algorithm with 81% and XGBoost algorithm with 84%.**

Keywords – **credit worthiness, classification, ensemble, prediction, pre-processing.**

I. INTRODUCTION

The primary source of income for banks is interest from loans. Majority of banks' profits are money generated from loans issued. Even when the bank approves the loan amount after a lengthy process of applicant's testimonial gathering and verification, there is no guarantee that the chosen applicant will be genuine to the bank or not in repaying the outstanding amount. Bank workers manually verify the applicants' documents before approving loans to eligible candidates. It takes a considerable amount of time to go through every applicant's documents. Due to personally scrutinising every detail, there is a possibility of human error and there is a chance that a loan could be approved to ineligible applicants or rejected to eligible applicants.
Machine learning algorithms shall be used to process the loan applications to automatically choose genuine and eligible applications [1]. The entire process is simplified using machine learning and the system can foretell whether a loan applicant is secure or not and recommend or reject to approve the loan amount. Thus, the processing time will be reduced and more number of applications can be processed. And both bank employees and applicants will be more satisfied. By this way, for bankers as well as potential borrowers, loan prognostic is extremely beneficial.

II. SYSTEM OVERVIEW

The system was developed using Python 3.7 language and Jupyter Notebook cross-platform Integrated Development Environment (IDE). The system consists of reading the raw data, pre-processing the data and classification. The pre-processed data are used for training and testing. The testing data are fed into the model for classification using the three machine learning algorithms - Naïve Bayes, SVM & XGBoost. The developed system divides the loan applicants into two groups based on their credit history as applicants with good credit history and bad credit history. Then based on average bank balance of the period and financial background, the loan amount is approved to applicants with good credit history. Decision tree algorithm is used to classify the applications [4]. The performance of the system is evaluated based on their accuracies.

III. SYSTEM DESCRIPTION

A. Software, Libraries and Dataset Used [3]:

1. Python 3.7: Python is an interpreted, high-level general programming language. Its formatting is visually uncluttered and it uses English keywords. It provides a vast library for data mining and predictions[10].
2. Jupyter Notebook: Jupyter Notebook is an open-source cross-platform IDE for scientific programming in the

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 07, ISSN No. 2455-2143, Pages 46-50
Published Online November 2023 in IJEAST (http://www.ijeast.com)

Python language.

3. NumPy: NumPy is used for building front-end part of the system.
4. Pandas: Pandas is used for data pre-processing and statistical analysis of data.
5. Matplotlib: Matplotlib is used for graphical representation and prediction.
6. Seaborn: Seaborn provides a high level interface for drawing attractive and informative graphs.
7. Kaggle: Kaggle is a data source provider for learning purpose.

Loan_ID	Gender	Married	Dependent	Education	Self_Emplo	ApplicantIn	Coapplican	LoanAmou	Loan_Amou	Credit_Hist	Property_A	Loan_Status
LP001002	Male	No	0	Graduate	No	5849	0		360	1	Urban	Y
LP001003	Male	Yes	1	Graduate	No	4583	1508	128	360	1	Rural	N
LP001005	Male	Yes	0	Graduate	Yes	3000	0	66	360	1	Urban	Y
LP001006	Male	Yes	0	Not Gradua	No	2583	2358	120	360	1	Urban	Y
LP001008	Male	No	0	Graduate	No	6000	0	141	360	1	Urban	Y
LP001011	Male	Yes	2	Graduate	Yes	5417	4196	267	360	1	Urban	Y
LP001013	Male	Yes	0	Not Gradua	No	2333	1516	95	360	1	Urban	Y

Fig. 1. Sample Data Set Used

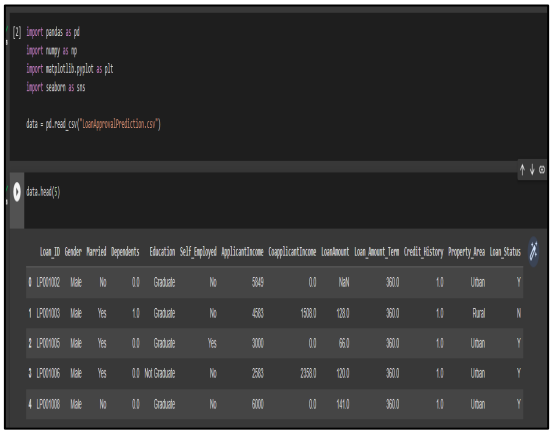

Fig. 2. Importing Libraries and Dataset

B. Data Pre-processing

Data pre-processing is an important step in creation of a machine-learning model [8]. The raw data may not be in the required format for the model, which can cause misleading outcomes. Data pre-processing deals with noises, duplicates and missing values in the dataset. It includes importing datasets, splitting datasets, attribute scaling and cleansing to improve the accuracy of the model [11].

Fig. 3. Data Preprocessing

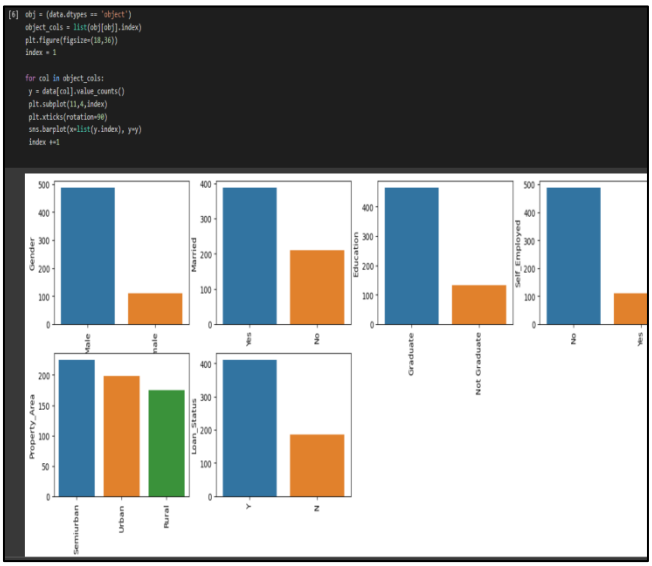

Fig. 4. Data Visualisation

C. Algorithms Used

1. Naive Bayes Algorithm

Naive Bayes is a probabilistic machine learning algorithm, commonly used for classification. Assumption of feature independence, equal importance of all features, representative training data and balanced class distribution assumed and poor performance with rare events are its features.

In loan approval prediction, this algorithm fits a model on the training data using GaussianNB function from scikit learn library, makes predictions on the testing data using the predict() method and accuracy of the model is evaluated using accuracy_score() function.

The dataset is pre-processed before fitting the model by dropping the Loan_ID column and converting categorical variables to numerical using label encoding. The pre-processed data is then split into training and testing sets using the train_test_split() function from the scikit learn library. Overall, Naive Bayes is a powerful and efficient algorithm for classification problems and can be easily implemented using libraries like sklearn in Python.

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 07, ISSN No. 2455-2143, Pages 46-50
Published Online November 2023 in IJEAST (http://www.ijeast.com)

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score

# Load the loan approval prediction dataset
data = pd.read_csv("LoanApprovalPrediction.csv")

# Drop the Loan_ID column and fill missing values with the mean
data.drop(['Loan_ID'], axis=1, inplace=True)
data.fillna(data.mean(), inplace=True)

# Convert categorical variables to numerical using label encoding
for col in data.columns:
    if data[col].dtype == 'object':
        data[col] = pd.Categorical(data[col]).codes

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(data.drop(['Loan_Status'], axis=1),
                                                    data['Loan_Status'],
                                                    test_size=0.4,
                                                    random_state=1)
```

Fig. 5. Naive Bayes Algorithm - Coding

2. SVM Algorithm

The SVM algorithm is a popular choice for solving classification problems in machine learning [5]. Its features are high variance & low bias, does not execute well when the data set has more noise, can only be applied to two-dimensional data and computational complexity is high and time-consuming.

It is used to predict the loan approval status of applicants [2] based on the attributes - Gender, Married, Education, Applicant_Income, Coapplicant_Income, Loan_Amount, Loan_Amount_Term, Credit_History and Property_Area. Before training, the dataset is pre-processed by converting categorical variables into numerical variables using label encoding or one-hot encoding techniques. The model can be deployed for loan approval prediction. The applicants' data can be fed into the model and it will output their loan approval status.

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score

# Load the dataset
data = pd.read_csv("LoanApprovalPrediction.csv")

# Drop the Loan_ID column
data.drop(['Loan_ID'], axis=1, inplace=True)

# Encode categorical variables
for col in data.columns:
    if data[col].dtype == 'object':
        data[col] = pd.Categorical(data[col]).codes

# Fill missing values with the mean of the column
data = data.fillna(data.mean())

# Split the data into training and testing sets
X = data.drop(['Loan_Status'], axis=1)
Y = data['Loan_Status']
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4, random_state=1)
```

Fig. 6. SVM Algorithm - Coding

```
sns.catplot(x="Gender", y="Married",
            hue="Loan_Status",
            kind="bar",
            data=data)
```

```
<seaborn.axisgrid.FacetGrid at 0x7eff6f0e7fd0>
```

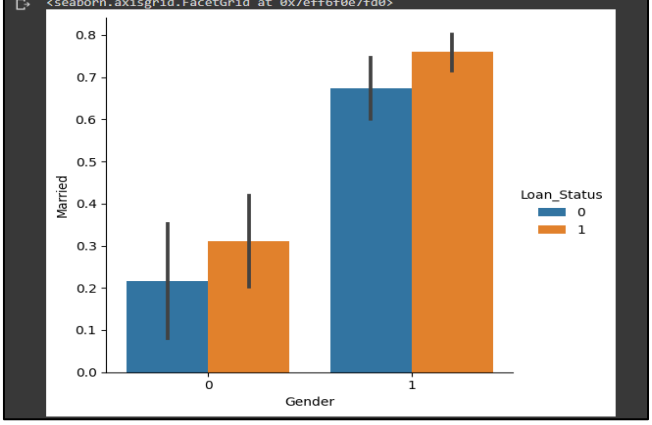

Fig. 7. Output of SVM Algorithm

3. XGBoost Algorithm

XGBoost is a popular machine learning algorithm used for classification and regression tasks. Built-in regularisation to prevent over-fitting, ensemble learning and several hyper parameters to optimise model performance are its features. It is used in loan approval prediction, because it is effective in handling complex, non-linear relationships in data and can handle missing values in data.

Boosting is a technique that combines multiple weaker models into a stronger one. Boosting is used to improve the SVM algorithm's performance. The training set is used to train the XGBoost model by fit method and the testing by predict method is used to predict the loan approval status of the applicants.

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score

# load data
data = pd.read_csv("/content/LoanApprovalPrediction.csv")

# remove null values
data = data.dropna()

# encode target variable
le = LabelEncoder()
data['Loan_Status'] = le.fit_transform(data['Loan_Status'])

# encode categorical variables
data_encoded = pd.get_dummies(data, columns=['Gender', 'Married', 'Education', 'Self_Employed', 'Property_Area'])

# split into train and test sets
X = data_encoded.drop(['Loan_ID', 'Loan_Status'], axis=1)
y = data_encoded['Loan_Status']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Fig.8. XGBoost Algorithm – Coding

The purpose of classifier is to productively apply the system in processing the applications [6]. The system used

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 07, ISSN No. 2455-2143, Pages 46-50
Published Online November 2023 in IJEAST (http://www.ijeast.com)

numerous ensemble strategies for multi-class classification as well as binary classification [7]. The system used the above three machine learning algorithms individually to determine the credit worthiness [9] of loan applicants. The algorithms automatically predict the credit worthiness of applicants by using their credit histories and financial background [12]. To get much better results ensemble learning techniques like Boosting (XGBoost algorithm) is used in the system.

IV. SYSTEM EVALUATION

To evaluate the system's performance in predicting the credit worthiness of applicants and know how efficiently and accurately it works, it was tested on a dataset [13]. Two datasets - one for training and another for testing were collected from Kaggle, a data source provider for learning purpose. Naive Bayes algorithm predicted the credit worthiness with 79% accuracy, SVM algorithm with 81% accuracy and XGBoost algorithm with 84% accuracy.

```
# Fit a Naive Bayes model on the training data
nb = GaussianNB()
nb.fit(X_train, y_train)

# Make predictions on the testing data
y_pred = nb.predict(X_test)

# Evaluate the model's accuracy
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy*100)
```

```
Accuracy: 81.66666666666667
<ipython-input-16-5c7d9293107c>:11: FutureWarning: The default value of numeric_only in DataFrame.mean is deprecated.
  data.fillna(data.mean(), inplace=True)
```

Fig. 9. Accuracy of Naïve Bayes Algorithm

```
# Scale the data
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Train the SVM model
svm = SVC(kernel='rbf', gamma='auto')
svm.fit(X_train, Y_train)

# Predict on the testing set
Y_pred = svm.predict(X_test)

# Evaluate the model's accuracy
accuracy = accuracy_score(Y_test, Y_pred)
print("Accuracy:", accuracy*100)
```

```
Accuracy: 81.25
```

Fig. 10. Accuracy of SVM Algorithm

```
# fit XGBoost model
xgb_clf = XGBClassifier()
xgb_clf.fit(X_train, y_train)

# make predictions and evaluate accuracy
y_pred_xgb = xgb_clf.predict(X_test)
accuracy_xgb = accuracy_score(y_test, y_pred_xgb)
print("Accuracy using XGBoost:", accuracy_xgb*100)
```

```
Accuracy using XGBoost: 84.15841584158416
```

Fig. 11. Accuracy of XGBoost Algorithm

Table 1. Accuracy Results

No.	Algorithm	Accuracy
1	Naive Bayes	79%
2	SVM	81%
3	XGBoost	84%

V. CONCLUSION

The developed system to predict credit worthiness in bank loan approval process satisfies bankers' requirements. By boosting with SVM, it is possible to improve the accuracy and robustness of loan approval prediction models, thus reducing the risk of over-fitting. This system can be integrated with other modules of a banking management information system. Mobile apps can also be developed for the prediction purpose. The system can be tested with many other datasets. The system need to be maintained periodically to incorporate new data or changes in the loan approval criteria.

VI. REFERENCES

[1]. Ashwini S. Kadam, Shraddha R. Nikam, Ankita A. Aher, Gayatri V. Shelke and Amar S. Chandgude. (2021). Prediction for Loan Approval using Machine Learning Algorithm, International Research Journal of Engineering and Technology,Vol.08, No. 04, (pp.4089-4092).

[2]. E. Kadam, A. Gupta, S. Jagtap, I. Dubey and G. Tawde. (2023). Loan Approval Prediction System using Logistic Regression and CIBIL Score, In Proc. of International Conference on Electronics and Sustainable Communication Systems, (pp. 1317-1321).

[3]. Gomathi alias Rohini. S and Mohanavel.S. (2023). Credit Card Fraud Detection with Machine Learning Algorithms, International Journal of Research and Analytical Reviews, Vol.10, No.4, (pp.818-824).

[4]. https://towardsdatascience.com/decision-trees-in-

machine-learning-641b9c4e8052

[5]. https://towardsdatascience.com/predict-loan-eligibility-using-machine-learning-models-7a14ef904057

[6]. https://www.analyticsvidhya.com/blog/2022/01/decision-tree-machine-learning-algorithm

[7]. https://www.seldon.io/decision-trees-in-machine-learning

[8]. https://www.xoriant.com/blog/decision-trees-for-classification-a-machine-learning-algorithm

[9]. Hussein A. Abdou, Marc D. Dongmo Tsafack, Collins G. Ntim and Rose D. Baker. (2016), Predicting Credit Worthiness in Retail Banking with Limited Scoring Data, Knowledge-Based Systems, Vol. 103,(pp.89-103).

[10]. Mehul Madaan, Aniket Kumar, Chirag Keshri, Rachna Jain and Preeti Nagrath, (2021). Loan Default Prediction Using Decision Trees and Random Forest: A Comparative Study. In IOP Conference Series: Materials Science and Engineering 1022.

[11]. Prateek Dutta. (2021). A Study on Machine Learning Algorithm for Enhancement of Loan Prediction, International Research Journal of Modernization in Engineering Technology and Science, Vol.03, No.01, (pp.160-165).

[12]. Regina Esi Turkson, Edward Yellakuor Baagyere and Gideon Evans Wenya. (2016). A Machine Learning Approach for Predicting Bank Credit Worthiness, in Proc. of International Conference on Artificial Intelligence and Pattern Recognition, (pp. 1-7).

[13]. Tejaswini. J, Mohana Kavya. T and Devi Naga Ramya. R. (2020). Accurate Loan Approval Prediction based on Machine Learning Approach, Journal of Engineering Sciences, Vol. 11, No. 4, (pp. 523-532).