REPORT

ON

Data Science

Submitted by

Utsav Parajuli

TABLE OF CONTENTS

CHAPTER I INTRODUCTION

1.1 Background

This project “Loan Default Prediction” is a model made using various machine learning algorithms. It helps the model’s user to correctly predict whether or not the loan will default by looking at the input data. As the data are increasing daily due to digitization in the banking sector, people want to apply for loans through the internet. Daily there are so many applications that are challenging to manage by the bank employees, and also the chances of some mistakes are high. Most banks earn profit from the loan, but it is risky to choose deserving customers from the number of applications.

1.2 Problem Statement of the Report

The problem is that one mistake can make a massive loss to a bank. Also, dependency on human intervention and delay in results have been the biggest obstacles in this system. Such moderation increases the cost as well as the time required for completion of the objective which can be reduced by automating this process.

1.3 Objectives for the Development

The following are the objectives for which the system was developed;

- To be able to correctly predict defaulters.
- To contribute to the financial sector of the country.

1.4 Literature Review

A prediction is a statement about what someone thinks will happen in the future. Predictive analytics is a branch of advanced analytics that uses many techniques from data mining, statistics, modeling, machine learning, and artificial intelligence to analyze current data to make predictions. [1] For financial institutions and the banking industry, it is very crucial to have predictive models for their financial activities, as they play a major role in risk management. Predicting loan default is one of the critical issues that they focus on, as huge revenue loss could be prevented by predicting customer’s ability to pay back on time. [2]

The use of a federated learning approach for the task in hand is necessary to overcome the issues discussed previously. The Synthetic Minority Oversampling Technique (SMOTE) approach is proposed to overcome the class imbalance issue. [3]

1.5 Proposed Methodology

The different steps taken to predict the bank loan of applicants is shown in the figure below;

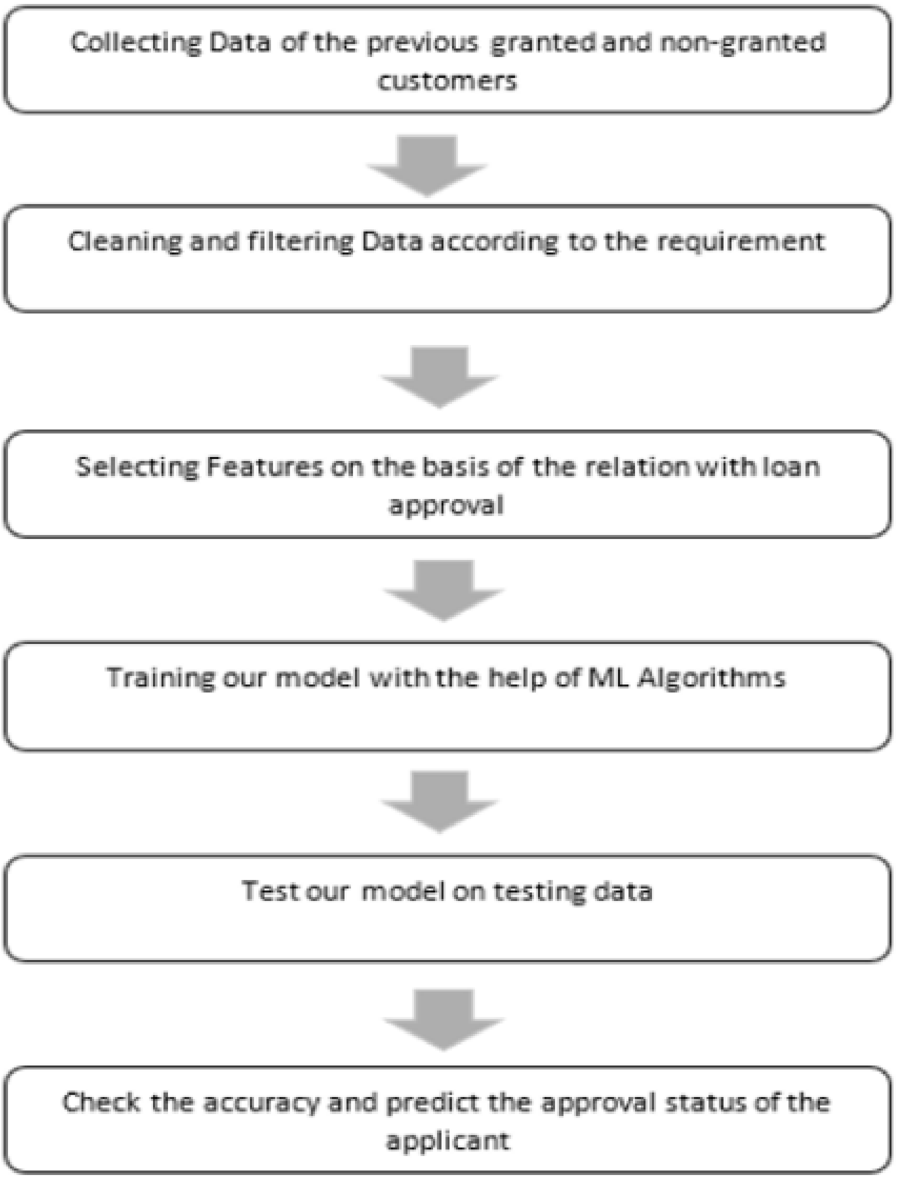
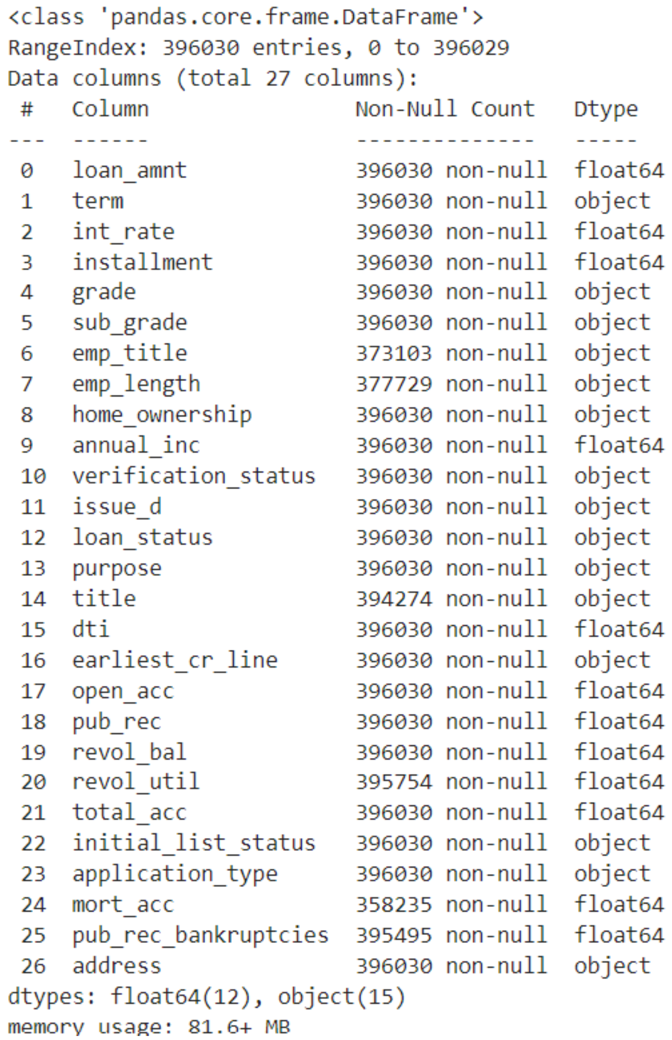

Collecting Data of the previous granted and non-granted customers
Cleaning and filtering Data according to the requirement
Selecting Features on the basis of the relation with loan approval
Training our model with the help of ML Algorithms
Test our model on testing data
Check the accuracy and predict the approval status of the applicant

Fig 1. Proposed Methodology

CHAPTER II TASKS AND ACTIVITIES PERFORMED

2.1 Dataset Description

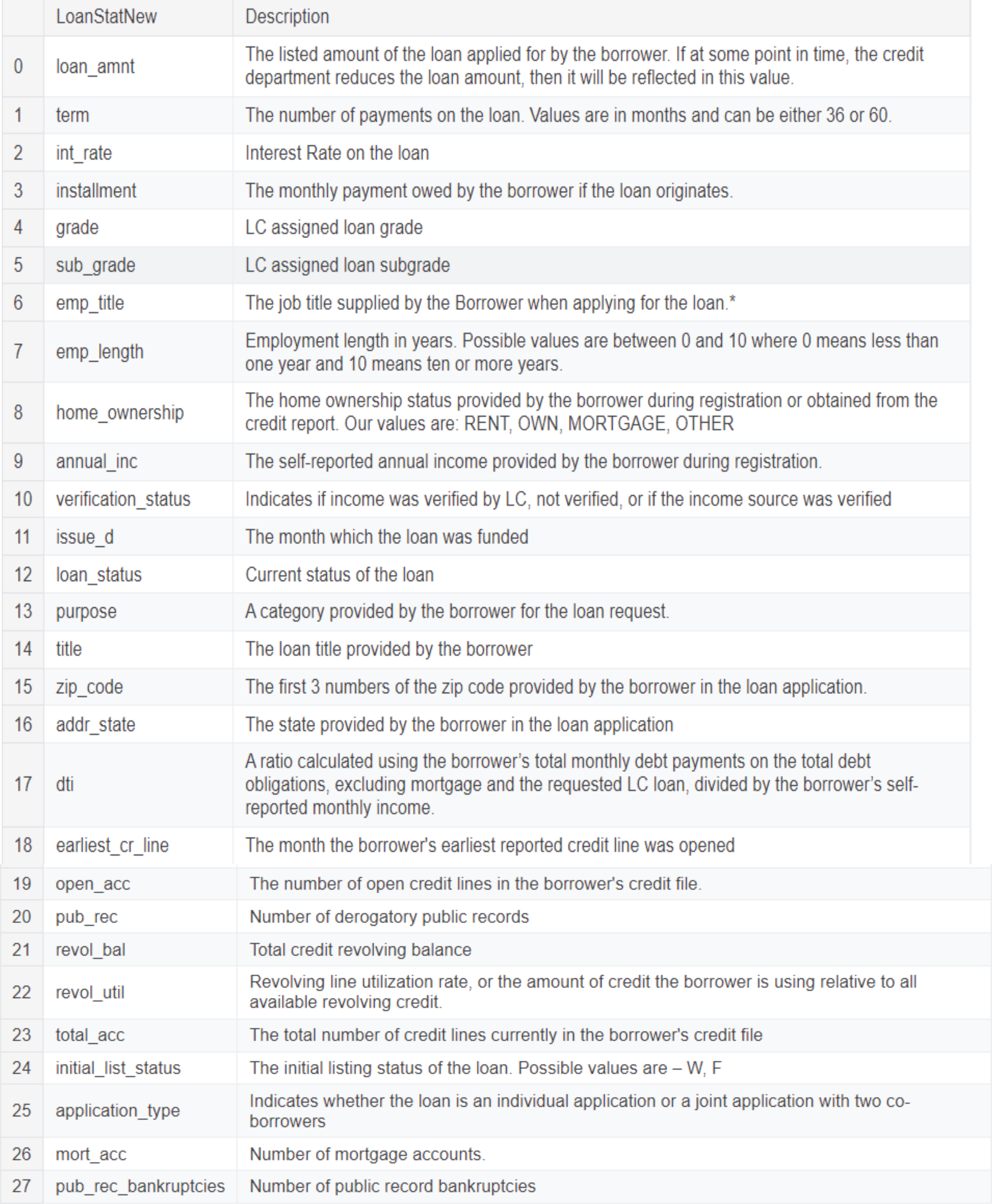
The bank loan prediction system dataset is taken from Kaggle which belongs to Lending Club. The shape of the dataset is (396030, 27). The info of the dataset is shown below;



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 396030 entries, 0 to 396029
Data columns (total 27 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   loan_amnt             396030 non-null  float64
 1   term                  396030 non-null  object 
 2   int_rate              396030 non-null  float64
 3   installment           396030 non-null  float64
 4   grade                 396030 non-null  object 
 5   sub_grade             396030 non-null  object 
 6   emp_title             373103 non-null  object 
 7   emp_length            377729 non-null  object 
 8   home_ownership        396030 non-null  object 
 9   annual_inc            396030 non-null  float64
 10  verification_status   396030 non-null  object 
 11  issue_d               396030 non-null  object 
 12  loan_status           396030 non-null  object 
 13  purpose               396030 non-null  object 
 14  title                 394274 non-null  object 
 15  dti                   396030 non-null  float64
 16  earliest_cr_line      396030 non-null  object 
 17  open_acc              396030 non-null  float64
 18  pub_rec               396030 non-null  float64
 19  revol_bal             396030 non-null  float64
 20  revol_util            395754 non-null  float64
 21  total_acc             396030 non-null  float64
 22  initial_list_status   396030 non-null  object 
 23  application_type      396030 non-null  object 
 24  mort_acc              358235 non-null  float64
 25  pub_rec_bankruptcies  395495 non-null  float64
 26  address               396030 non-null  object 
dtypes: float64(12), object(15)
memory usage: 81.6+ MB
```

Fig 2.1. Data Description

There are 396030 total records of the applicants with the values of their concerning attributes in categorical and numerical data. In the pre-processing and feature engineering of the data, we handle the missing value and also normalize the data so we can further process it into ML algorithm. The dataset is further divided into training and testing. The model is trained on machine learning algorithms and predicts the system on test data which is discussed in the Next section in details. The description of all the column present in the dataset is given below;



	LoanStatNew	Description
0	loan_amnt	The listed amount of the loan applied for by the borrower. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value.
1	term	The number of payments on the loan. Values are in months and can be either 36 or 60.
2	int_rate	Interest Rate on the loan
3	installment	The monthly payment owed by the borrower if the loan originates.
4	grade	LC assigned loan grade
5	sub_grade	LC assigned loan subgrade
6	emp_title	The job title supplied by the Borrower when applying for the loan.*
7	emp_length	Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
8	home_ownership	The home ownership status provided by the borrower during registration or obtained from the credit report. Our values are: RENT, OWN, MORTGAGE, OTHER
9	annual_inc	The self-reported annual income provided by the borrower during registration.
10	verification_status	Indicates if income was verified by LC, not verified, or if the income source was verified
11	issue_d	The month which the loan was funded
12	loan_status	Current status of the loan
13	purpose	A category provided by the borrower for the loan request.
14	title	The loan title provided by the borrower
15	zip_code	The first 3 numbers of the zip code provided by the borrower in the loan application.
16	addr_state	The state provided by the borrower in the loan application
17	dti	A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income.
18	earliest_cr_line	The month the borrower's earliest reported credit line was opened
19	open_acc	The number of open credit lines in the borrower's credit file.
20	pub_rec	Number of derogatory public records
21	revol_bal	Total credit revolving balance
22	revol_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
23	total_acc	The total number of credit lines currently in the borrower's credit file
24	initial_list_status	The initial listing status of the loan. Possible values are – W, F
25	application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers
26	mort_acc	Number of mortgage accounts.
27	pub_rec_bankruptcies	Number of public record bankruptcies

Fig2.2 Dataset Description

Predictive analytics is used to predict the data about future events. It includes many techniques such as data mining, machine learning and modeling. Machine learning is a type of artificial intelligence that allows a software application to learn from the data & become more accurate in predicting outcomes without human intervention. Machine learning and deep learning help to design and develop such a machine that automatically learns and predicts your data and situation. Machine learning is often divided into different subcategories according to the type of problems being comes. Some ML type is as follows:

- Supervised Learning

  Supervised learning is the point at which the model is getting prepared on a labelled dataset. In this kind of learning both training and testing, datasets are labelled. The output of prediction is always coming either 1 (yes) or 0 (No).

- Unsupervised Learning

  In unsupervised learning, the input data are not labelled and also do not have any prior information about the data. Here the task of the machine is to find the hidden pattern from the data by using cluster analysis. The dataset is labelled so that here we used supervised learning approach.

Our Dataset is labeled, so we use supervised machine learning algorithm.

2.2 Algorithms used for Prediction

The algorithms used for prediction are:

2.2.1 Logistic Regression

It is a classification set of rules used to assign observations to a discrete set of instructions. Logistic regression is also a predictive analysis, like other regression analyses methods. Logistic regression is basically used for define the relationship between dependent binary variable and nominal or other independent variable. Now a day’s logistic regression is used in many research areas like medical science, machine learning and social science. It also used by many e-commerce applications to predict the mind set of customer to buy the product.

2.2.2 Random Forest

Random Forest is a robust system learning algorithm that is used for a ramification of responsibilities along with classification and regression. Random forests method overcome the over fitting issue of decision trees during training. It is an ensemble method made up of a large number of small decision trees [5,7] called estimators where each tree produces the prediction. The random forest model combines the predictions of the estimators to produce a more accurate prediction.

2.3 Exploratory Data Analysis

For EDA, various visualization tools like bar graph, line chart, scatter plot, box plot, heatmap, etc. were used. EDA helped us to identify relationships between various variables and properly use it in the time of model formulation.

2.4 Data Cleaning and Preprocessing

The following steps were performed for data cleaning and preprocessing.

- Firstly, the dataset was checked for duplicates.
- The dataset was checked for null values.
- The dataset contained null values so the null values were handled accordingly by removing or filling the missing data with mean, median or mode.
- Unnecessary or repetitive features were removed.
- Outliers were detected and handled accordingly.
- Categorical columns were encoded for better performance.
- Scaling of data using Standard Scaler was done as well to handle outliers and for proper functioning of the algorithm.
- Unnecessary columns were removed for better prediction.
- For certain columns like address, only the relevant information like city was stored.
- The preprocessed data was stored in another csv file for model formulation.

2.5 Training the Dataset

In this stage, the preprocessed dataset was finally split into train data and test data. The train data was trained using various algorithms like; Random Forest Classifier, Logistic Regression and so on. For training purpose, SMOTE was used to reduce the class imbalance. The dataset was divided into input and target variable and afterwards various algorithms were used in the split data.

2.6 Testing the dataset

Various metrics like accuracy score, f1 score, recall, precision were used in the process of testing the dataset. Classification report and confusion matrix were imported and used to verify the performance of the model.

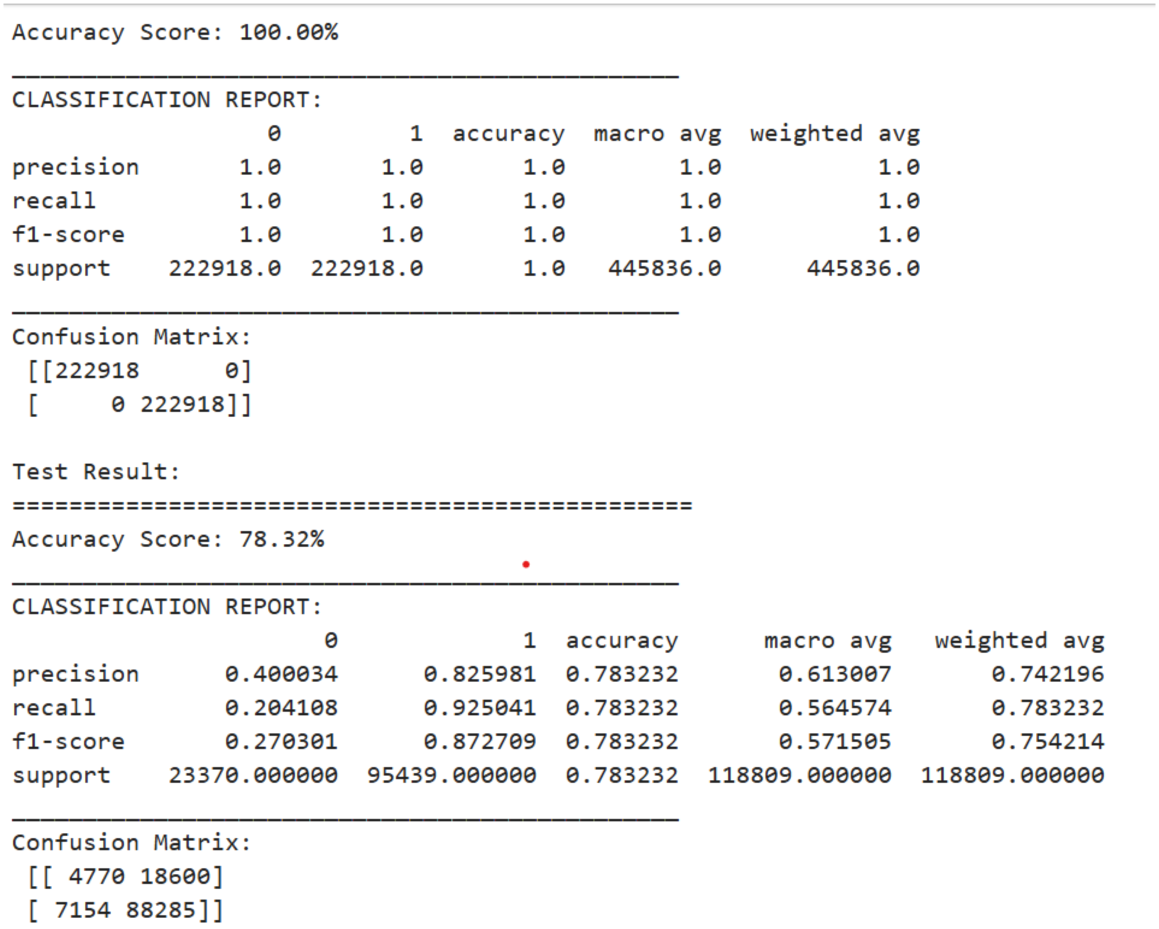
2.7 UI Formation and Model Deployment

After the model was tested, appropriate UI was formed for user interaction. The model formed was saved into another file and template was formed for the homepage. The model was deployed with the help of Flask.

CHAPTER III DISCUSSION AND CONCLUSION

3.1 Result

The result of the model is as follows;

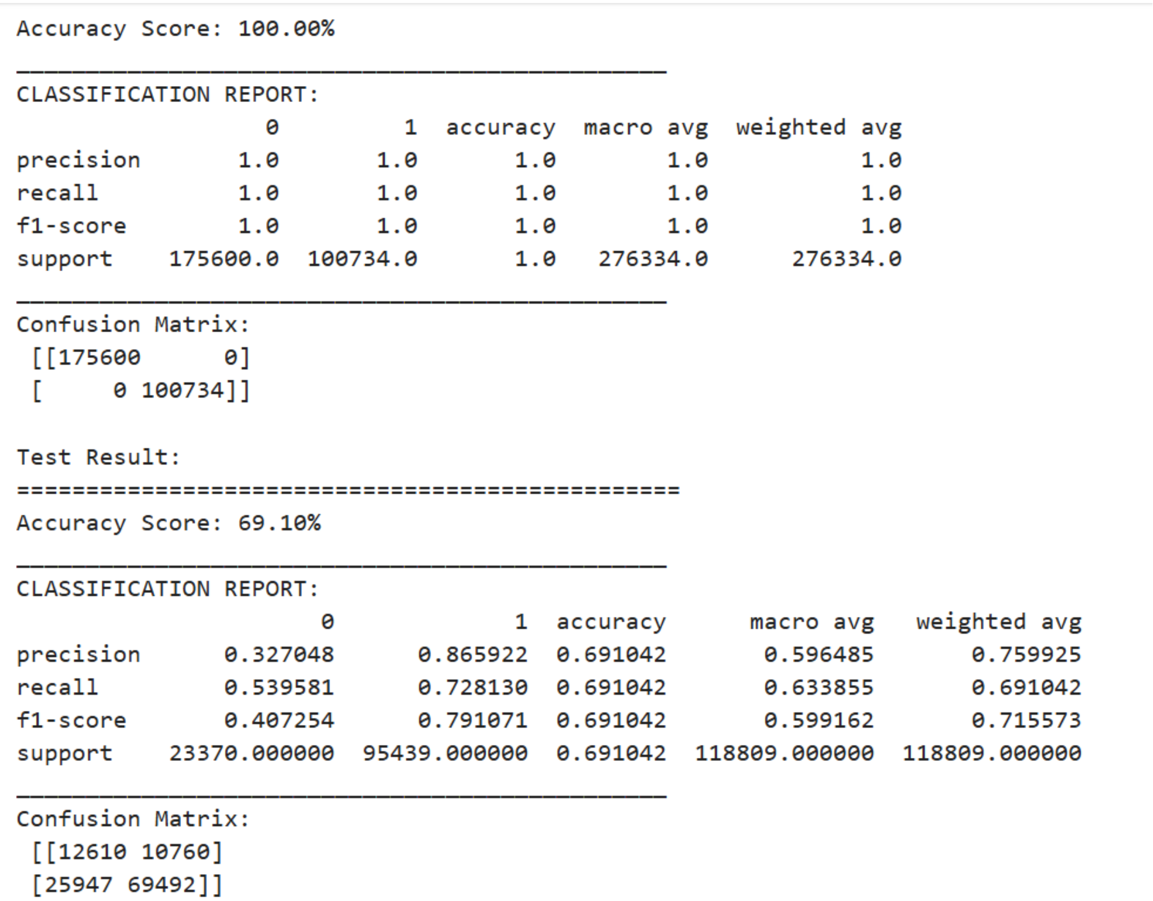


```
Accuracy Score: 100.00%
_______________________________________________
CLASSIFICATION REPORT:
                 0         1  accuracy  macro avg  weighted avg
precision      1.0       1.0       1.0        1.0           1.0
recall         1.0       1.0       1.0        1.0           1.0
f1-score       1.0       1.0       1.0        1.0           1.0
support   222918.0  222918.0       1.0   445836.0      445836.0
_______________________________________________
Confusion Matrix:
 [[222918      0]
 [     0 222918]]

Test Result:
================================================
Accuracy Score: 78.32%
_______________________________________________
CLASSIFICATION REPORT:
                     0             1  accuracy      macro avg   weighted avg
precision     0.400034      0.825981  0.783232       0.613007       0.742196
recall        0.204108      0.925041  0.783232       0.564574       0.783232
f1-score      0.270301      0.872709  0.783232       0.571505       0.754214
support   23370.000000  95439.000000  0.783232  118809.000000  118809.000000
_______________________________________________
Confusion Matrix:
 [[ 4770 18600]
 [ 7154 88285]]
```

Random Forest Classifier with SMOTE

SMOTE, which stands for **Synthetic Minority Oversampling Technique**, is a popular method for handling imbalanced datasets in machine learning. It aims to balance the dataset by generating synthetic samples for the minority class. For each sample in the minority class, SMOTE identifies its k-nearest neighbors within the same class (based on Euclidean distance).

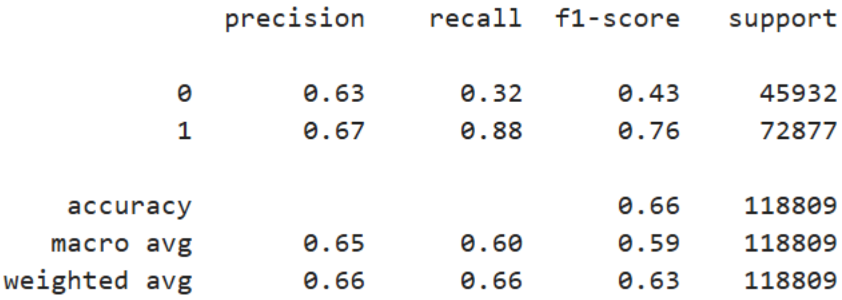


```
Accuracy Score: 100.00%
_______________________________________________
CLASSIFICATION REPORT:
                  0         1  accuracy  macro avg  weighted avg
precision       1.0       1.0       1.0        1.0           1.0
recall          1.0       1.0       1.0        1.0           1.0
f1-score        1.0       1.0       1.0        1.0           1.0
support    175600.0  100734.0       1.0   276334.0      276334.0
_______________________________________________
Confusion Matrix: 
 [[175600      0]
 [     0 100734]]

Test Result:
================================================
Accuracy Score: 69.10%
_______________________________________________
CLASSIFICATION REPORT:
                      0             1  accuracy      macro avg  weighted avg
precision      0.327048      0.865922  0.691042       0.596485      0.759925
recall         0.539581      0.728130  0.691042       0.633855      0.691042
f1-score       0.407254      0.791071  0.691042       0.599162      0.715573
support    23370.000000  95439.000000  0.691042  118809.000000 118809.000000
_______________________________________________
Confusion Matrix: 
 [[12610 10760]
 [25947 69492]]
```

Random Forest Classifier with SMOTEENN

SMOTEENN is a hybrid technique that combines SMOTE with ENN (**Edited Nearest Neighbors**), which oversamples the minority class using SMOTE and then removes noisy or ambiguous samples using ENN.



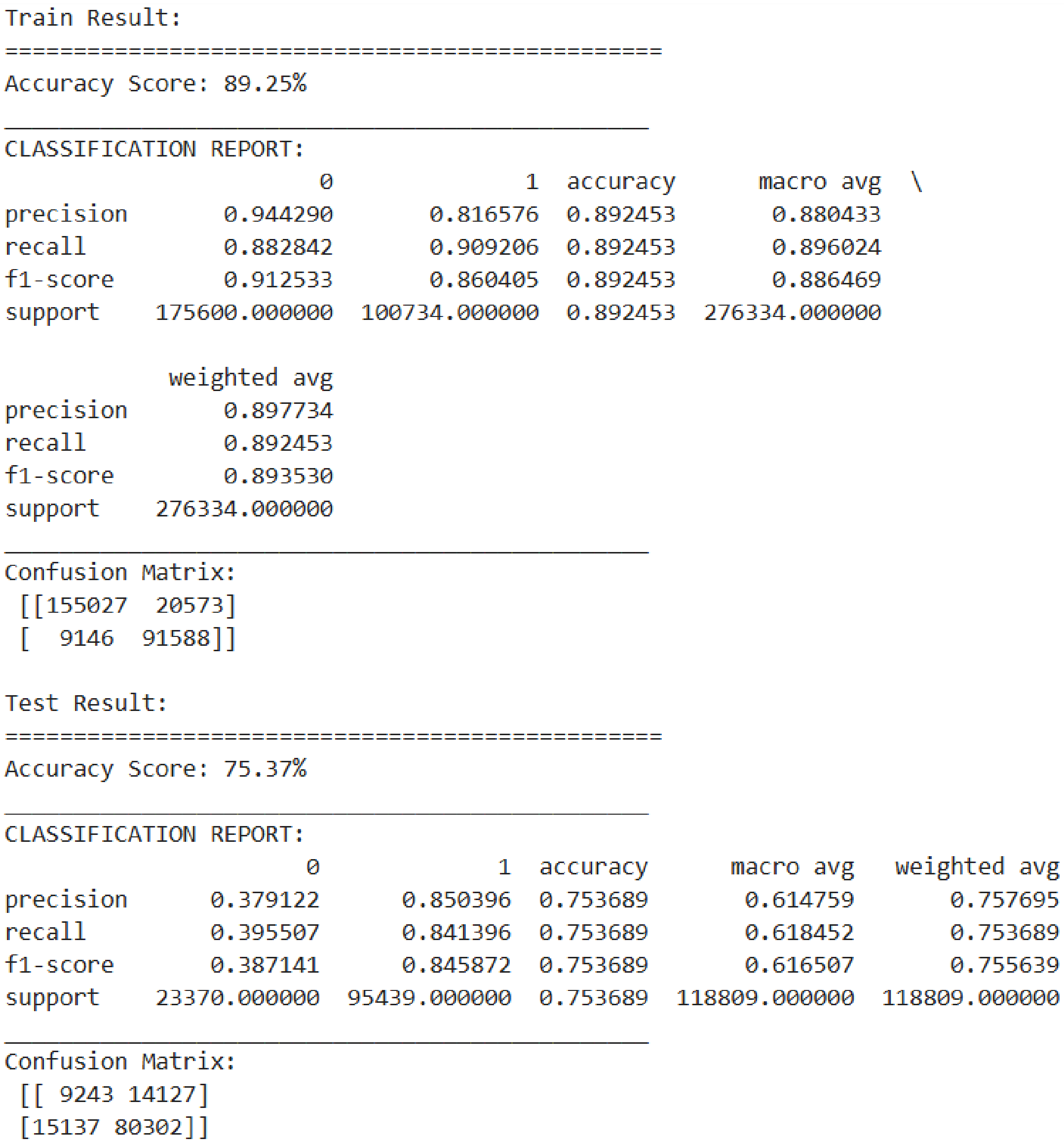
```
              precision    recall  f1-score   support

           0       0.63      0.32      0.43     45932
           1       0.67      0.88      0.76     72877

    accuracy                           0.66    118809
   macro avg       0.65      0.60      0.59    118809
weighted avg       0.66      0.66      0.63    118809
```

Logistic Regression

Logistic regression is a statistical method used for binary classification problems. It predicts the probability of an outcome that can belong to one of two categories, such as yes/no, true/false, or 0/1.



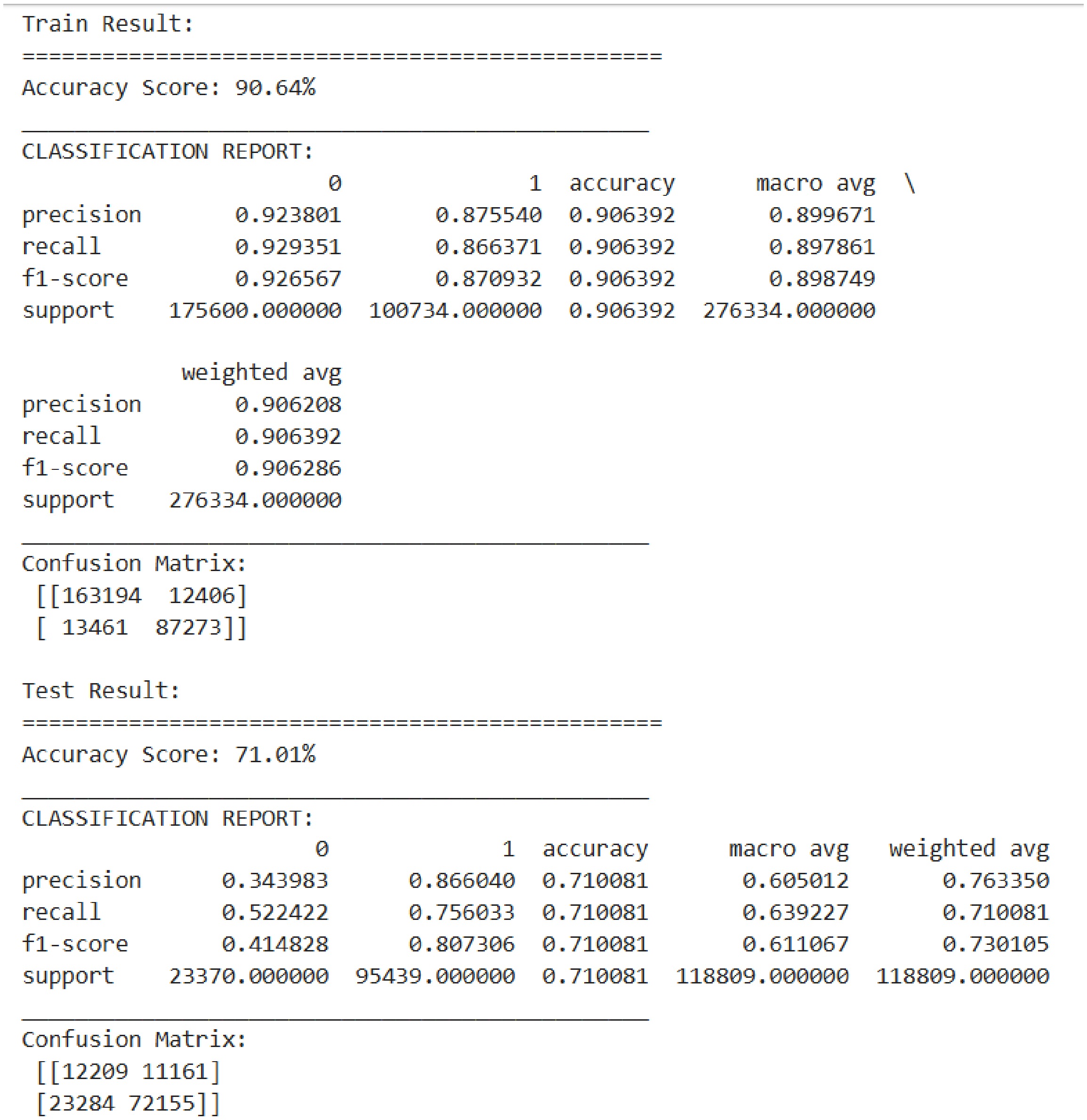
```
Train Result:
================================================
Accuracy Score: 89.25%
_______________________________________________
CLASSIFICATION REPORT:
                       0              1  accuracy      macro avg  \
precision       0.944290       0.816576  0.892453       0.880433   
recall          0.882842       0.909206  0.892453       0.896024   
f1-score        0.912533       0.860405  0.892453       0.886469   
support    175600.000000  100734.000000  0.892453  276334.000000   

            weighted avg  
precision       0.897734  
recall          0.892453  
f1-score        0.893530  
support    276334.000000  
_______________________________________________
Confusion Matrix: 
 [[155027  20573]
 [  9146  91588]]

Test Result:
================================================
Accuracy Score: 75.37%
_______________________________________________
CLASSIFICATION REPORT:
                      0             1  accuracy      macro avg   weighted avg
precision      0.379122      0.850396  0.753689       0.614759       0.757695
recall         0.395507      0.841396  0.753689       0.618452       0.753689
f1-score       0.387141      0.845872  0.753689       0.616507       0.755639
support    23370.000000  95439.000000  0.753689  118809.000000  118809.000000
_______________________________________________
Confusion Matrix: 
 [[ 9243 14127]
 [15137 80302]]
```

XGBoost Classifier with SMOTEENN

The XGBoost Classifier is an implementation of the gradient boosting algorithm designed for high performance and efficiency. It is widely used for both classification and regression tasks. XGBoost (short for *Extreme Gradient Boosting*) is known for its speed, accuracy, and scalability to large datasets.



```
Train Result:
================================================
Accuracy Score: 90.64%
_______________________________________________
CLASSIFICATION REPORT:
                       0              1  accuracy      macro avg  \
precision       0.923801       0.875540  0.906392       0.899671   
recall          0.929351       0.866371  0.906392       0.897861   
f1-score        0.926567       0.870932  0.906392       0.898749   
support    175600.000000  100734.000000  0.906392  276334.000000   

            weighted avg  
precision       0.906208  
recall          0.906392  
f1-score        0.906286  
support    276334.000000  
_______________________________________________
Confusion Matrix: 
 [[163194  12406]
 [ 13461  87273]]

Test Result:
================================================
Accuracy Score: 71.01%
_______________________________________________
CLASSIFICATION REPORT:
                      0             1  accuracy      macro avg   weighted avg
precision      0.343983      0.866040  0.710081       0.605012       0.763350
recall         0.522422      0.756033  0.710081       0.639227       0.710081
f1-score       0.414828      0.807306  0.710081       0.611067       0.730105
support    23370.000000  95439.000000  0.710081  118809.000000  118809.000000
_______________________________________________
Confusion Matrix: 
 [[12209 11161]
 [23284 72155]]
```

XGBoost Classifier with Grid Search

Grid search is a systematic method to tune hyperparameters by exhaustively searching over a specified parameter grid. This helps identify the best combination of hyperparameters that optimize model performance.

3.2 Discussion

Loan Default Prediction is developed as per the problems identified in the previous sections. This model solves the problem that occurs when only human intervention is used in the process of verification of loan.

3.3 Conclusion

The model has been successfully completed with the set of defined objectives in time all while learning and gaining new insights to how the model actually works. It can help users to automate the loan verification process saving both time and resources.

3.4 Future Enhancement

Currently, the model's performance is not up to par. The accuracy is not that good either. In the future, this problem will be addressed and the performance will be further improved. The model's parameter will be further tweaked to find the optimal model for loan default prediction.

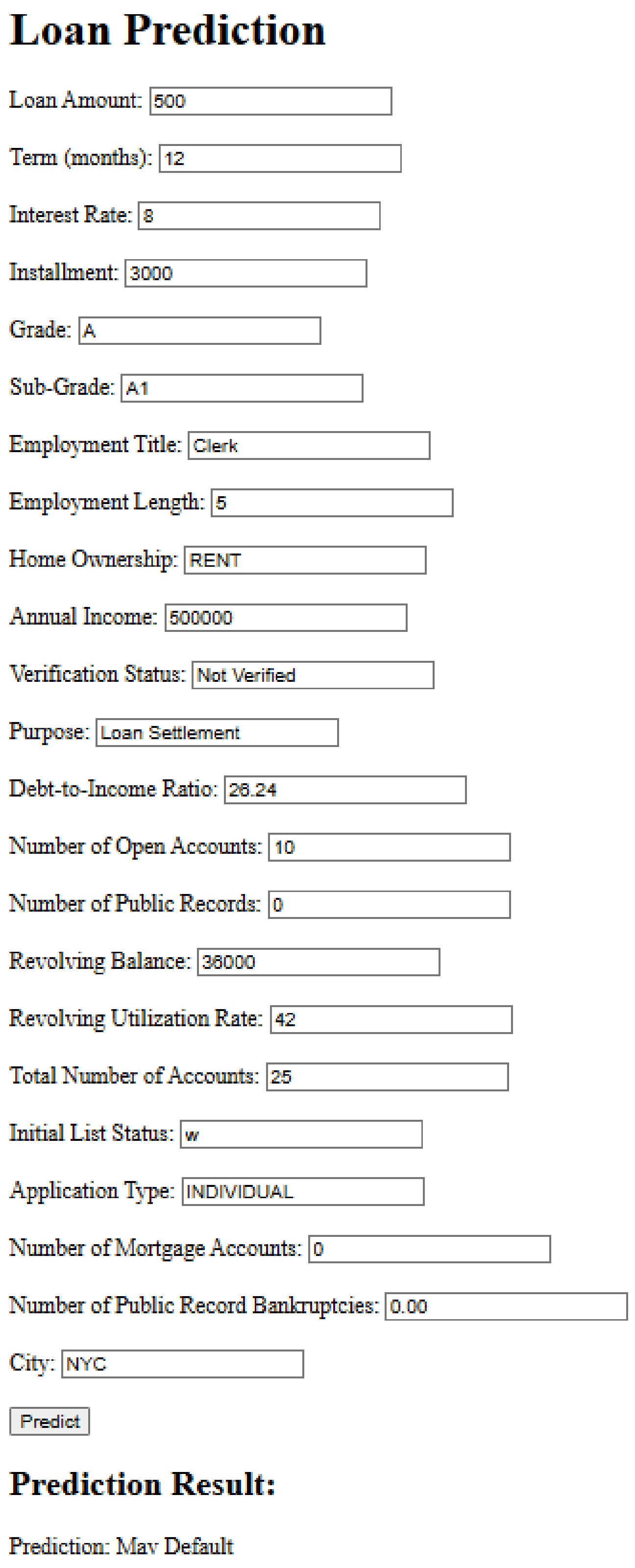
REFERENCES

[1] V. P. S. K. P. K. B. Anshika Gupta, "Bank Loan Prediction System using Machine Learning," *9th International Conference on System Modeling & Advancement in Research Trends, 4th–5th December, 2020 Faculty of Engineering & Computing Sciences, Teerthanker Mahaveer University, Moradabad, India,* 2020.

[2] M. Alhasan, "Loan Default Prediction Model Improvement through Comprehensive Preprocessing and Features Selection," *2019 International Arab Conference on Information Technology (ACIT),* 2019.

[3] G. Shingi, "A federated learning based approach for loan," *2020 International Conference on Data Mining Workshops (ICDMW),* 2020.

APPENDIX



Loan Prediction

Loan Amount: 500

Term (months): 12

Interest Rate: 8

Installment: 3000

Grade: A

Sub-Grade: A1

Employment Title: Clerk

Employment Length: 5

Home Ownership: RENT

Annual Income: 500000

Verification Status: Not Verified

Purpose: Loan Settlement

Debt-to-Income Ratio: 26.24

Number of Open Accounts: 10

Number of Public Records: 0

Revolving Balance: 36000

Revolving Utilization Rate: 42

Total Number of Accounts: 25

Initial List Status: w

Application Type: INDIVIDUAL

Number of Mortgage Accounts: 0

Number of Public Record Bankruptcies: 0.00

City: NYC

Prediction Result:

Prediction: May Default