e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:01/January-2021 Impact Factor- 5.354 www.irjmets.com

A STUDY ON MACHINE LEARNING ALGORITHM FOR ENHANCEMENT OF LOAN PREDICTION

Prateek Dutta[*1]

[*1]Student, B.tech Artificial Intelligence, G.H. Raisoni College of Engineering, India.

ABSTRACT

In the lending industry, investors offer loans to lenders for the purpose of repaying interest. If the borrower pays the loan, then the lender will make a profit on the interest. However, if the borrower fails to repay the loan, the lender loses the loan. Therefore, lenders face the problem of predicting the risk of the borrower not being able to repay the loan. The main purpose of this project is to predict which of the customers will be repaid with their loans or not.

Keyword: Loan, Machine Learning, Kaggle, Regression.

I. INTRODUCTION

1.1 What is Machine Learning?

Machine learning is a subset of Artificial Intelligence that allows a computer program to automatically learn from a previous task. It works by analysing data, identifying patterns, and incorporating minimal human interventions. Almost any work that can be completed with a data-defined pattern or set of rules can be done with machine learning machines. This allows companies to change processes that were previously only possible for people to make assumptions that respond to customer service calls, bookkeeping, and reviews.

1.2 Types of Machine Learning

Machine Learning has been further classified into four types. They are categories as follow:

- Supervised Machine Learning
- Unsupervised Machine Learning
- Semi-supervised Machine Learning
- Reinforcement Machine Learning

1.3 Importance

Machine learning algorithms enable the construction of a new model using previously unknown historical data that can be used to train the model to make better predictions not only for credit risks, but also for other risks such as early payment opportunities leading to loss of income from interest, existing withdrawal risks etc. With a good model, financial institutions can predict the likelihood of a customer repaying a loan before the maturity date and then have procedures with pre-defined prevention measures to be taken, prior to this occurrence.[1]

Dataset

It is the collection of data arranged in rows and columns. It can be of any form but mostly it is found in csv format. A single dataset can contain numerous number of data. The dataset for this project on which this paper deals with, has been taken from Kaggle. It contain two dataset, one of them is training and another is testing. It contains 13 columns labelled as: Loan_ID, Gender, Married, Dependents, Education, Self_Employed, Applicant Income, Co-applicant Income, Loan Amount, Loan_Amount_tearm, Credit_History, Property_Area, Loan_Status.

II. ALGORITHM USED IN LOAN PREDICTION

In this paper supervised classification problem to be trained with algorithms like:

1. Logistic Regression
2. Decision Tree
3. Random Forest

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:01/January-2021 Impact Factor- 5.354 www.irjmets.com

In this project, default hyperparameter values are employed. More visualization can be done beyond what's executed in this post.

In this model, mainly five libraries has been used as, pandas, NumPy, matplotlib, seaborn and, sklearn. These libraries has been widely used in order to get the well defined and categorized result.

The identifiable machine learning separator is not limited to the above. Other models such as XGBoost, CatBoost and likes can be used in model training. The choice of these three algorithms in a row over the desire to keep the model defining itself again, the database is small.

Correlating Attributes

Based on the combination between the symbols it has been observed that they are more likely to repay their loans. Independent and important qualities can include Property location, education, loan amount, and last in credit History, i.e. from intuition it is considered important. Meeting in between attributes can be seen using corplot and boxplot in the Python platform.[2]

Steps Involved

⇨ So firstly you need to import the libraries (fig.-1)

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        %matplotlib inline
```

Fig.-1

⇨ Then insert the dataset into the environment and read it.(fig.-2)

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0

Fig.-2

⇨ Now check the missing values and fix them.(fig.-3)

	Total	Percent
Credit_History	50	0.081433
Self_Employed	32	0.052117
LoanAmount	22	0.035831
Dependents	15	0.024430
Loan_Amount_Term	14	0.022801
Gender	13	0.021173
Married	3	0.004886
Loan_Status	0	0.000000
Property_Area	0	0.000000
CoapplicantIncome	0	0.000000
ApplicantIncome	0	0.000000
Education	0	0.000000
Loan_ID	0	0.000000

Fig.-3

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:01/January-2021 Impact Factor- 5.354 www.irjmets.com

⇨ Now analyse the data and visualise it with respect to each column. It will results in graphical format(fig.- 4)

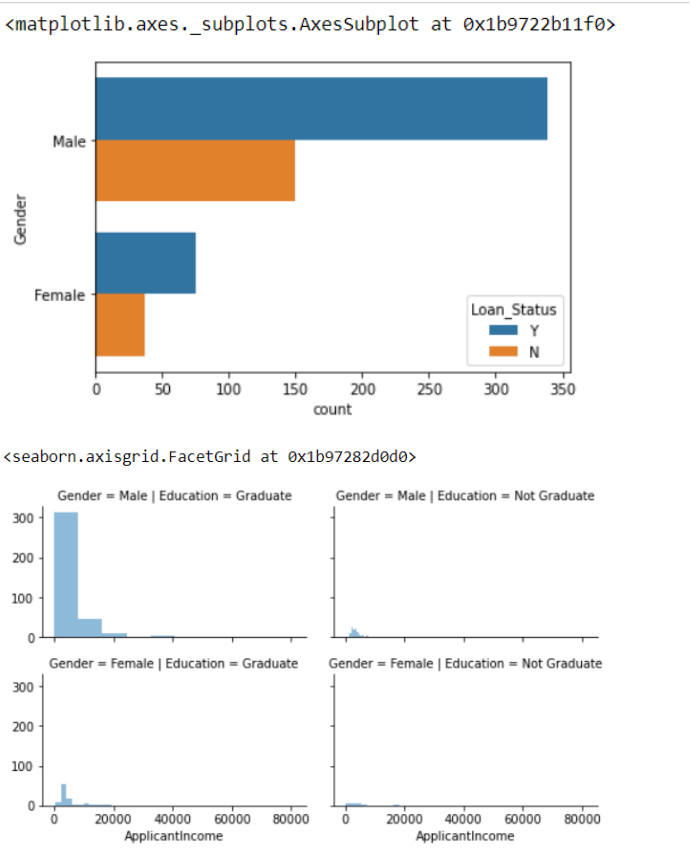

Fig.-4

⇨ Now next step is to encoding the numeric data and get it ready for training. (fig.-5)

```
data_train['Dependents'].value_counts()
```

```
0     345
1     102
2     101
3+     51
Name: Dependents, dtype: int64
```

```
data_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Loan_ID            614 non-null    object
 1   Gender             601 non-null    object
 2   Married            611 non-null    object
 3   Dependents         599 non-null    object
 4   Education          614 non-null    object
 5   Self_Employed      582 non-null    object
 6   ApplicantIncome    614 non-null    int64
 7   CoapplicantIncome  614 non-null    float64
 8   LoanAmount         592 non-null    float64
 9   Loan_Amount_Term   600 non-null    float64
 10  Credit_History     564 non-null    float64
 11  Property_Area      614 non-null    object
 12  Loan_Status        614 non-null    object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
```

Fig.-5

⇨ Then project the heatmap Showing the correlations of features with the target. (fig.-6)

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:01/January-2021 Impact Factor- 5.354 www.irjmets.com

```
<matplotlib.axes._subplots.AxesSubplot at 0x1b975b5b250>
```

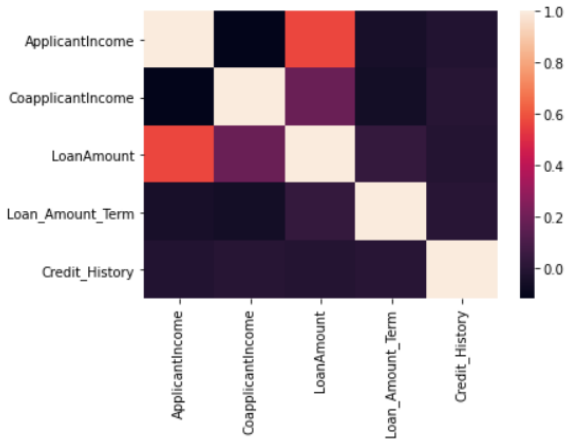

Fig.-6

III. RESULT

⇨ Now you can check the evaluation of the model using any of the three algorithm.

⇨ Logistic Regression (fig.-7)

```
model = LogisticRegression()
```

```
model.fit(X_train, y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='ovr', n_jobs=1,
          penalty='l2', random_state=None, solver='liblinear', tol=0.0001,
          verbose=0, warm_start=False)
```

```
ypred = model.predict(X_test)
print(ypred)
```

```
[1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 1
 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 0 0 1 1 1 1 1 0 1]
```

```
evaluation = f1_score(y_test, ypred)
evaluation
```

```
0.8979591836734695
```

Fig.-7

⇨ Decision Tree (fig.-8)

```
tree = DecisionTreeClassifier()
tree.fit(X_train, y_train)
```

```
DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
            max_features=None, max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=1, min_samples_split=2,
            min_weight_fraction_leaf=0.0, presort=False, random_state=None,
            splitter='best')
```

```
ypred_tree = tree.predict(X_test)
print(ypred_tree)
```

```
[0 1 1 0 1 1 1 1 0 1 0 1 1 1 1 1 1 1 1 0 0 0 1 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1
 1 1 1 0 0 1 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 0 1 1 0 0 1 0 0 1 1 1 1 1 1 1 1 0
 1 1 0 1 1 0 1 1 1 1 1 0 1 0 1 1 1 1 1 0 1 0 1 1 1 1 0 1 0 1 1 1 0 1 1 1 1 0
 1 1 1 0 0 0 1 1 0 1 0 0]
```

```
evaluation_tree = f1_score(y_test, ypred_tree)
evaluation_tree
```

```
0.7745664739884394
```

Fig.-8

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:01/January-2021 Impact Factor- 5.354 www.irjmets.com

⇨ Random Forest classifier (fig.-9)

```
]: forest = RandomForestClassifier()
   forest.fit(X_train, y_train)

]: RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
                          max_depth=None, max_features='auto', max_leaf_nodes=None,
                          min_impurity_decrease=0.0, min_impurity_split=None,
                          min_samples_leaf=1, min_samples_split=2,
                          min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=1,
                          oob_score=False, random_state=None, verbose=0,
                          warm_start=False)

]: ypred_forest = forest.predict(X_test)
   print(ypred_forest)

   [1 1 1 1 1 0 1 0 0 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 0 0 1 0 1 1 1 0 1
    1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 0 1
    1 1 0 1 1 0 0 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 0 1 1 1 0 1 1 0 1 0 1
    1 1 1 0 0 1 1 1 1 1 1 0 1]

]: evaluation_forest = f1_score(y_test, ypred_forest)
   evaluation_forest

]: 0.8540540540540541
```

Fig.-9

IV. CONCLUSION

From Test Data Analysis, we can generate insights from data. It can be seen in the trial of three models that Logistic Regression did better with evaluation of 89.7059% than others, Random Forest(77.4566%), Decision Tree(85.4054%). And last Logistic regression can be considered as best algorithm for Loan prediction using Machine Learning. Applicants with a low credit history fail to be accepted, probably because they have a chance of not paying. Most of the time, Applicants with a high fee are likely to be eligible for a rebate, which is likely to repay their loan. A particular sexual orientation and marital status seems beyond the reach of the company.

Conflict of Interest Statement

I Prateek Dutta, authors, state that there is no conflict of interest in my manuscript.

V. REFERENCE

[1] Zoran Ereiz. Predicting Default Loan using Machine Learning (OptiML), https://www.researchgate.net/publication/338948751_Predicting_Default_Loans_Using_Machine_Learning_OptiML

[2] Cowell, R.G., A.P., Lauritez, S.L., and Spiegelhalter, D.J.(1999). Graphical models and Expert Systems. Berlin: Springer.

[3] S. Shanmugan, S. Ravichandran, A Fuzzy Point approach the Solar Still Performances an Experimental Investigation, 2020 6(1)681-689. DOI Member: 10.6084/m9.jetir.JETIRDY06108

[4] K. Manikandan, S. Shanmugan, R. Ashish kumar, Vadde Venkat Harish, T. Jansi Rani, T. Nishanthi, Trans membranous fetal movement and pressure sensing. Materials Today: Proceedings, Available online 13 May 2020. https://doi.org/10.1016/j.matpr.2020.04.497

[5] J. Jennifer, Monica Nathasha Marrison, J. Seetha, S. Sivakumar and P. Sathish Saravanan, "DMMRA: Dynamic Medical Machine for Rural Areas", IEEE 2017 International Conference On Power And Embedded Drive Control (ICPEDC), pp. 461-471, 16th - 18th March 2017. DOI: 10.1109/ICPEDC.2017.8081135

[6] P Arokianathan, V Dinesh, B Elamaran, M Veluchamy and S Sivakumar, "Automated Toll Booth and Theft Detection System", IEEE 2017 Technological Innovations in ICT for Agriculture and Rural Development (TIAR), pp. 84-88, 07th - 08th April 2017. DOI: 10.1109/TIAR.2017.8273691

[7] Videla, Lakshmi Sarvani, et al. "Modified Feature Extraction Using Viola Jones Algorithm". Journal of Advanced Research in Dynamical and Control Systems. Volume 10, Issue 3 Special Issue, 2018, Pages 528-538

[8] Sreedevi E., PremaLatha V., Prasanth Y. and Sivakumar S., "A Novel Ensemble Learning for Defect Detection Method With Uncertain Data", Applications of Artificial Intelligence for Smart Technology, pp. 1-13, 2021. doi : 10.4018/978-1-7998-3335-2.ch005

[9] Premalatha V., Vineesha K. and Srinivasarao M., "International Journal of Scientific and Technology Research", 2020, 9(1), pp. 1005-1008.