LOAN ELIGIBILITY PREDICTION

INTRODUCTION:

OVERVIEW:

Loans are the core business of banks. The main profit comes directly from the loan's interest. The loan companies grant a loan after an intensive process of verification and validation. However, they still don't have assurance if the applicant is able to repay the loan with no difficulties.

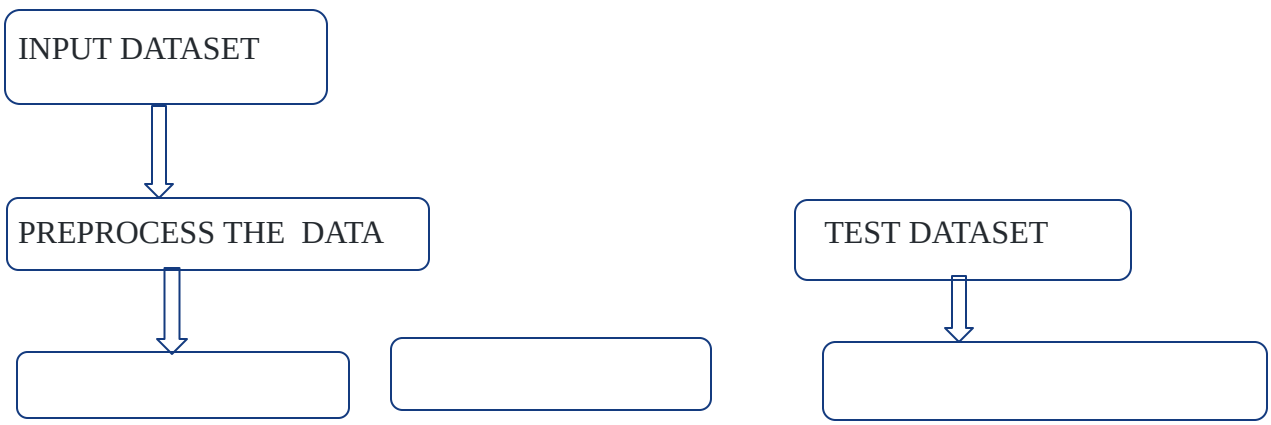
Goal of this project is to automate the loan eligibility process (real time) based on customer detail provided while filling online application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others. To automate this process, a problem has been given to identify the customers segments, those who are eligible for loan amount so that they can be targeted.

PURPOSE:

The main aim of this use-case is to build a predictive model to predict if an applicant is able to repay the lending company or not.

THEORITICAL ANALYSIS:

BLOCK DIAGRAM:



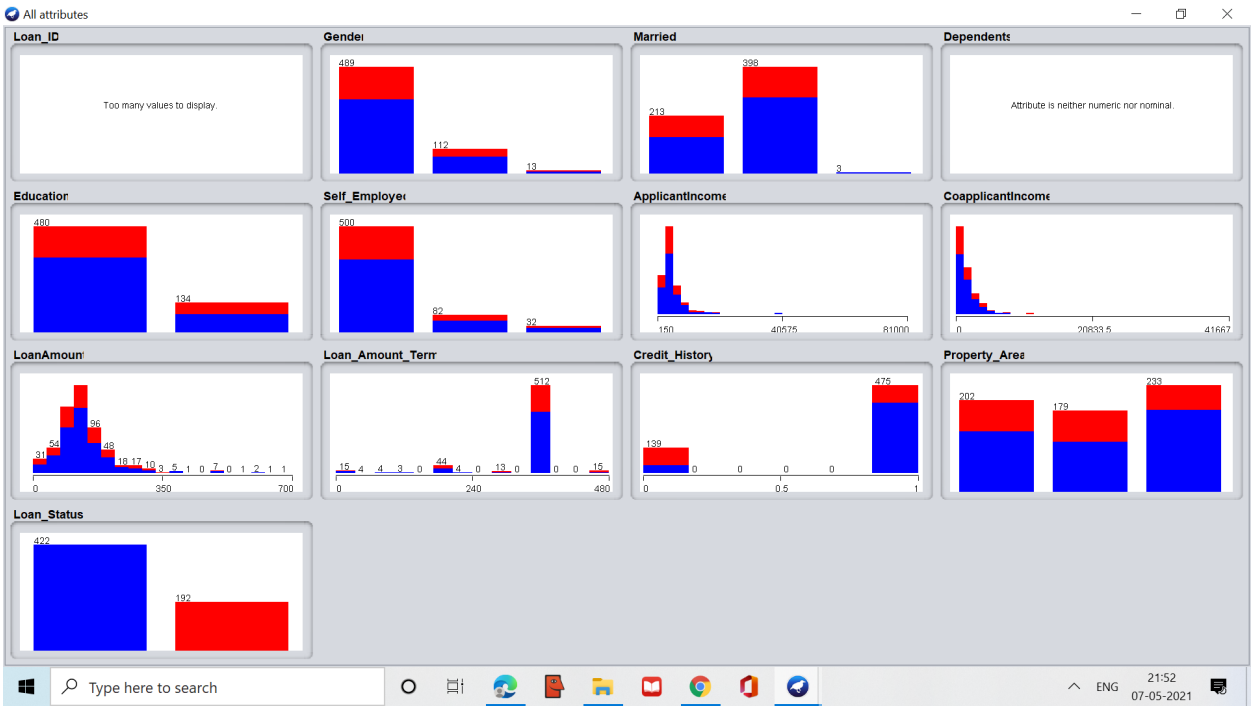
CREATE THE MODEL ⇨ TRAIN THE MODEL ⟹ TEST THE MODEL

REQUIREMENTS:

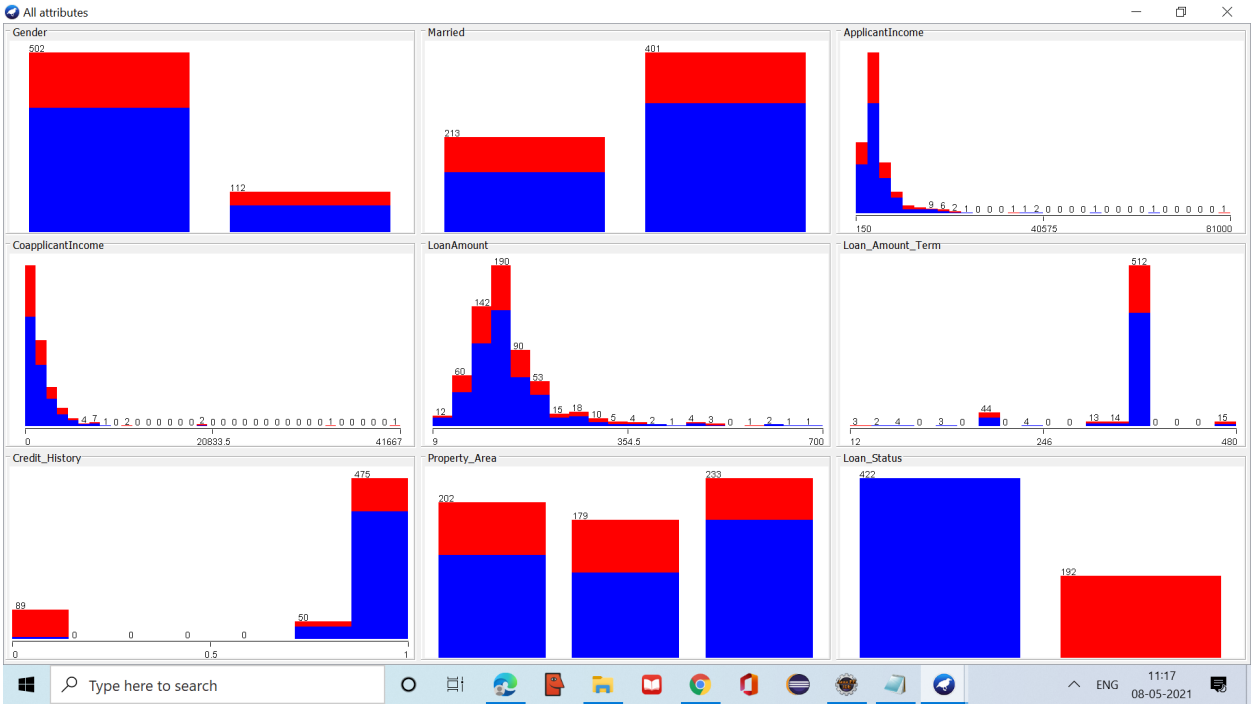
* ECLIPSE
* WEKA
* LANGUAGE:JAVA

- EXPERIMENTAL INVESTIGATIONS:

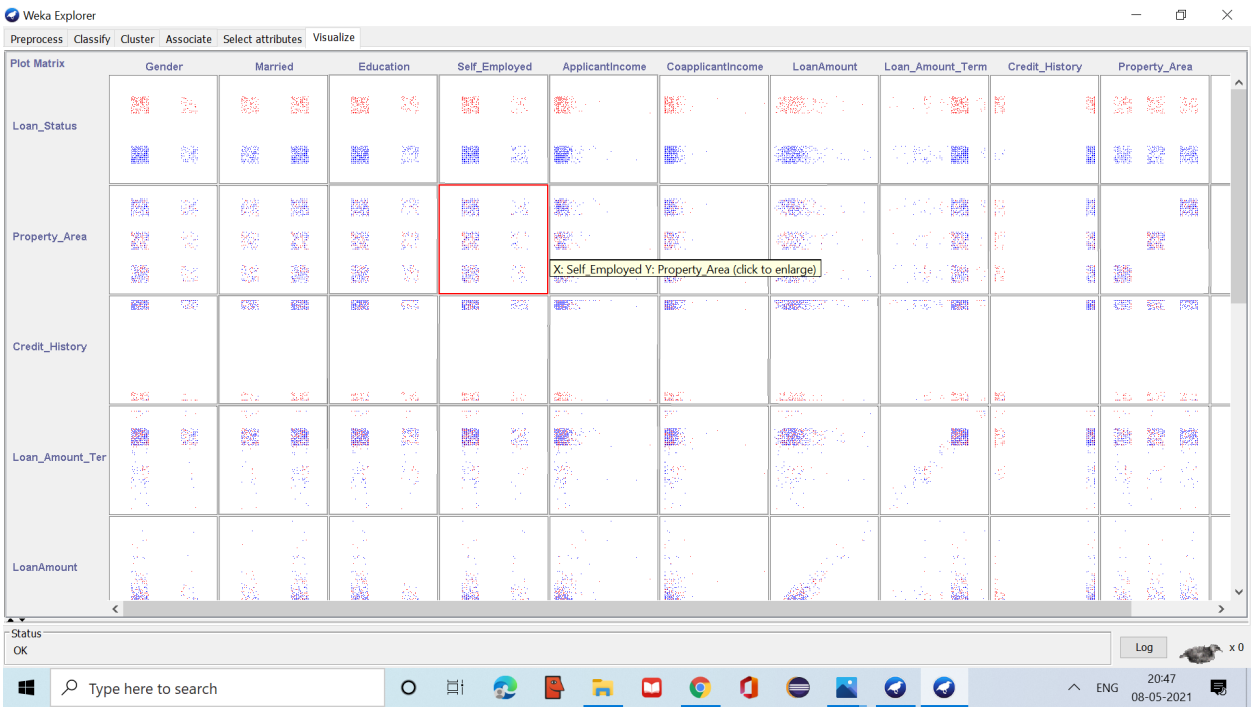
DATA ANALYSIS:

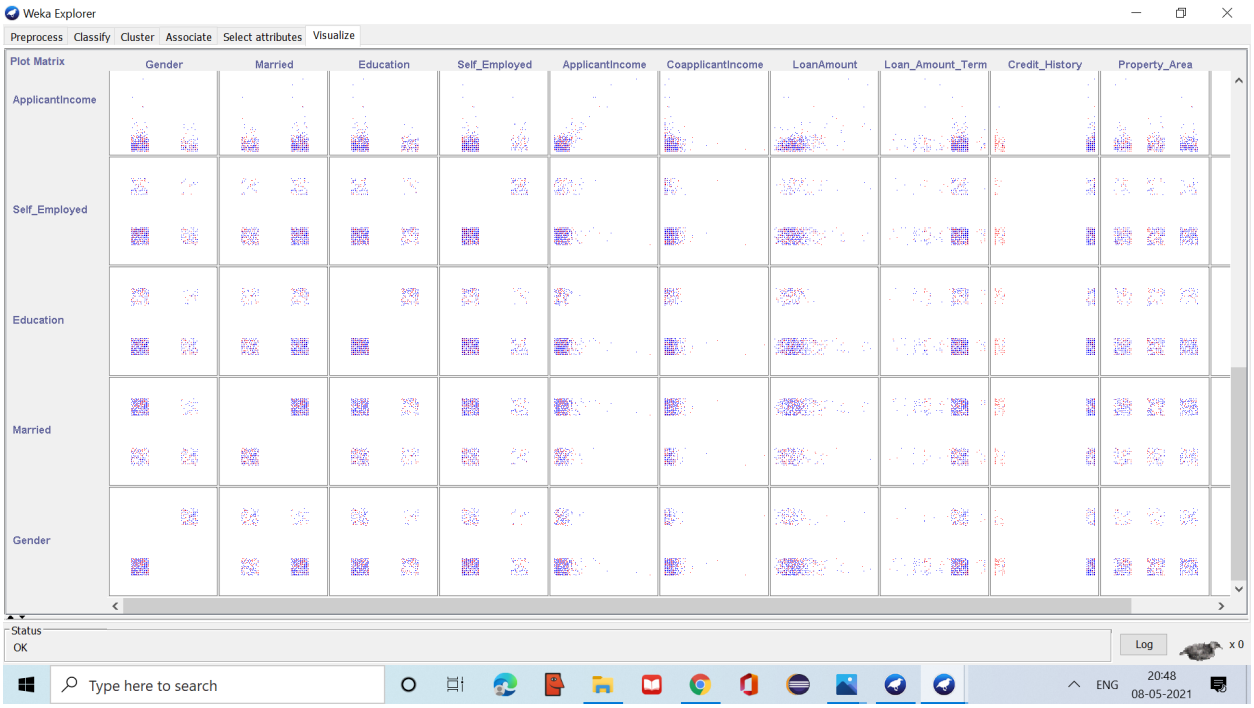


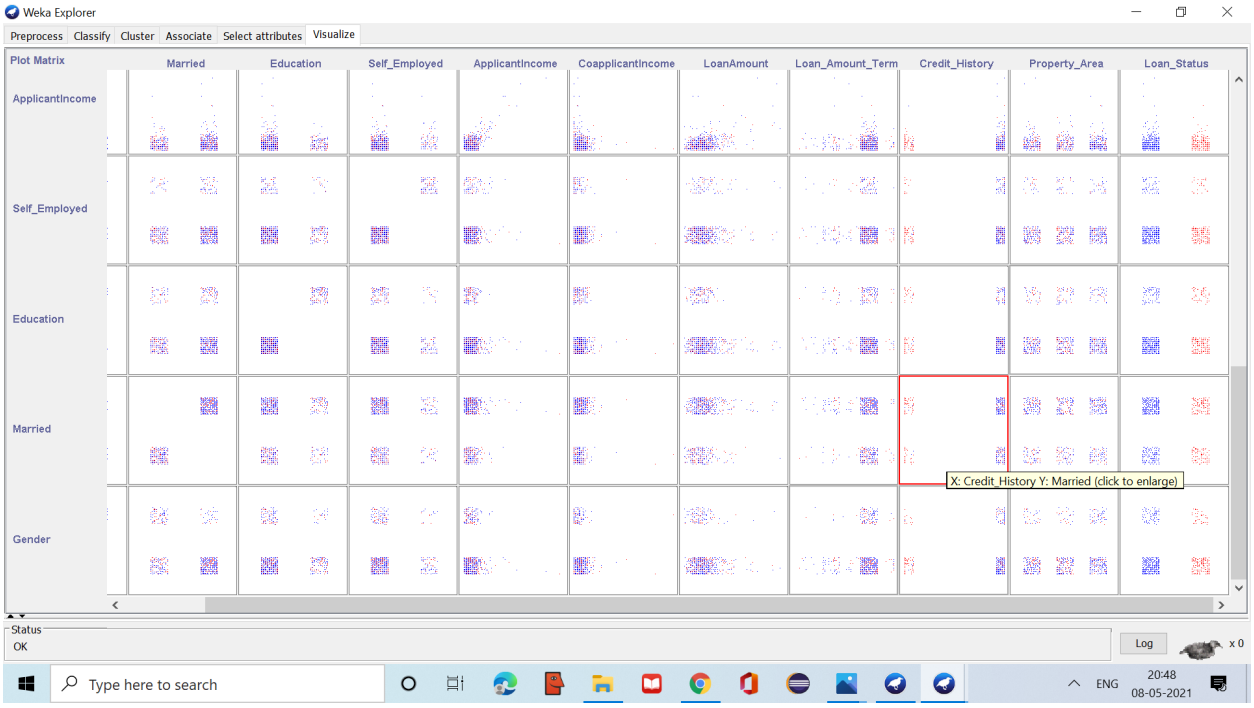
DATA PREPROCESSING:



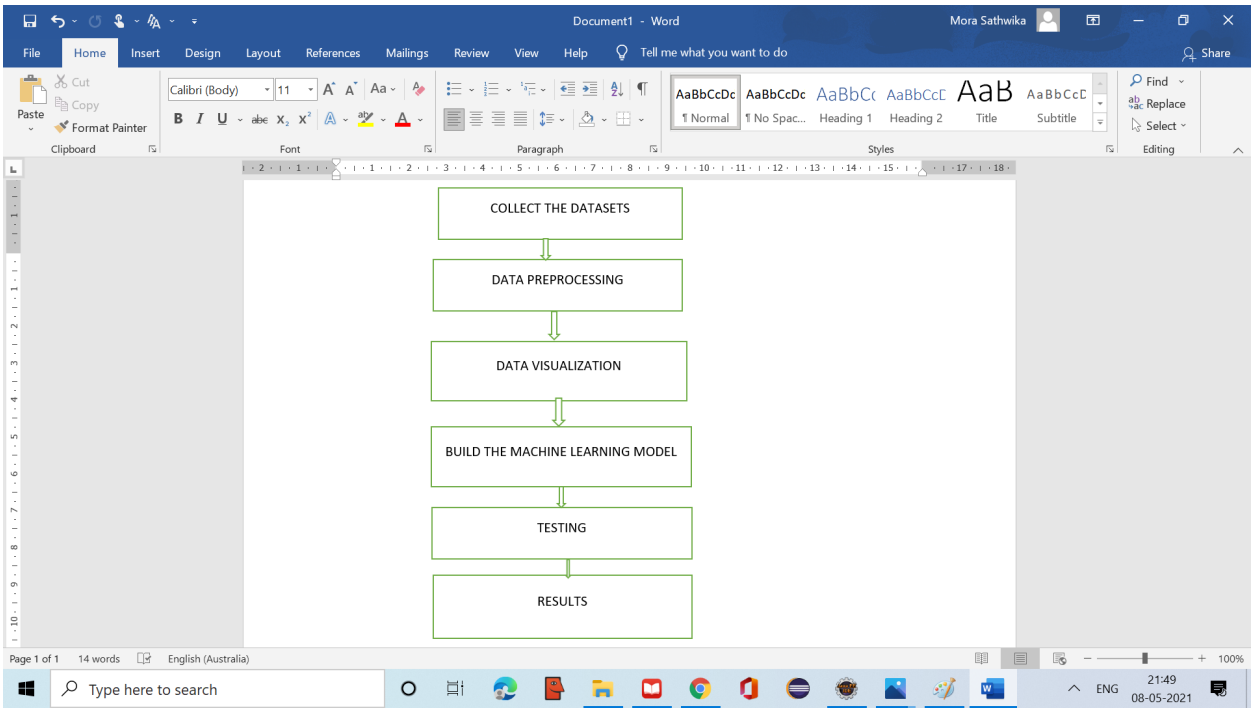
DATA VISUALIZATION:



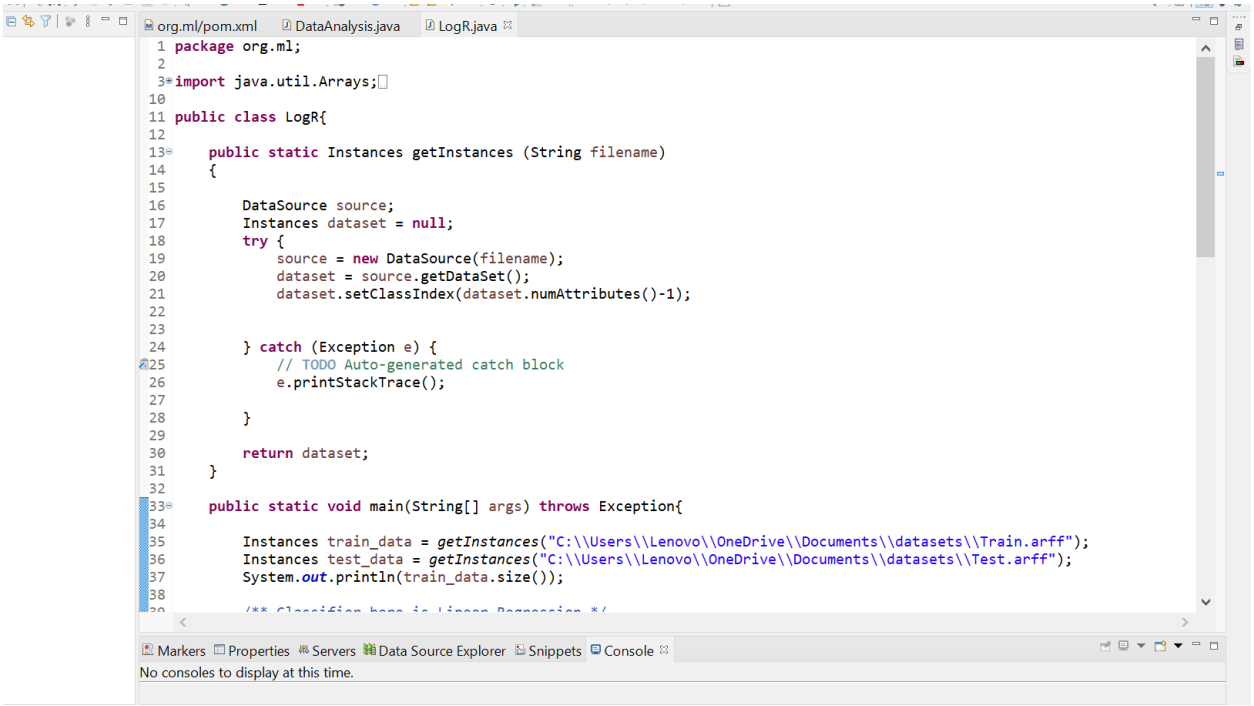

Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Plot Matrix
Gender
Married
Education
Self_Employed
ApplicantIncome
CoapplicantIncome
LoanAmount
Loan_Amount_Term
Credit_History
Property_Area
ApplicantIncome
Self_Employed
Education
Married
Gender
Status
OK
Log
x 0
Type here to search
ENG
20:48
08-05-2021


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Plot Matrix
Married
Education
Self_Employed
ApplicantIncome
CoapplicantIncome
LoanAmount
Loan_Amount_Term
Credit_History
Property_Area
Loan_Status
ApplicantIncome
Self_Employed
Education
Married
Gender
X: Credit_History Y: Married (click to enlarge)
Status
OK
Log
x 0
Type here to search
ENG
20:48
08-05-2021

FLOW START:



PREDICTION MODEL AND TESTING:



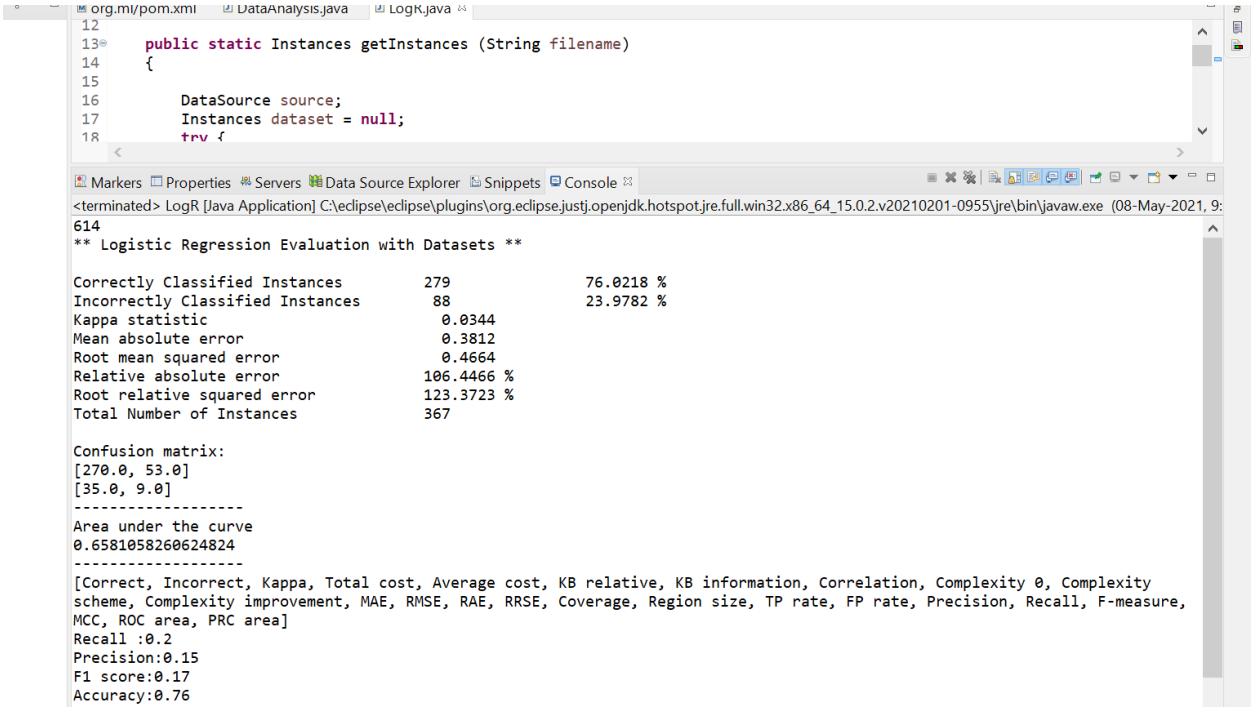
```
        /** Classifier here is Linear Regression */
        Classifier classifier = new weka.classifiers.functions.Logistic();
        /** */
        classifier.buildClassifier(train_data);


        /**
         * train the algorithm with the training data and evaluate the
         * algorithm with testing data
         */
        Evaluation eval = new Evaluation(train_data);
        eval.evaluateModel(classifier, test_data);
        /** Print the algorithm summary */
        System.out.println("** Logistic Regression Evaluation with Datasets **");
        System.out.println(eval.toSummaryString());
//      System.out.print(" the expression for the input data as per algorithm is ");
//      System.out.println(classifier);

        double confusion[][] = eval.confusionMatrix();
        System.out.println("Confusion matrix:");
        for (double[] row : confusion)
            System.out.println(  Arrays.toString(row));
        System.out.println("--------------------");

        System.out.println("Area under the curve");
        System.out.println( eval.areaUnderROC(0));
        System.out.println("--------------------");

        System.out.println(Evaluation.getAllEvaluationMetricNames());

        System.out.print("Recall :");
        System.out.println(Math.round(eval.recall(1)*100.0)/100.0);
```

Markers Properties Servers Data Source Explorer Snippets Console

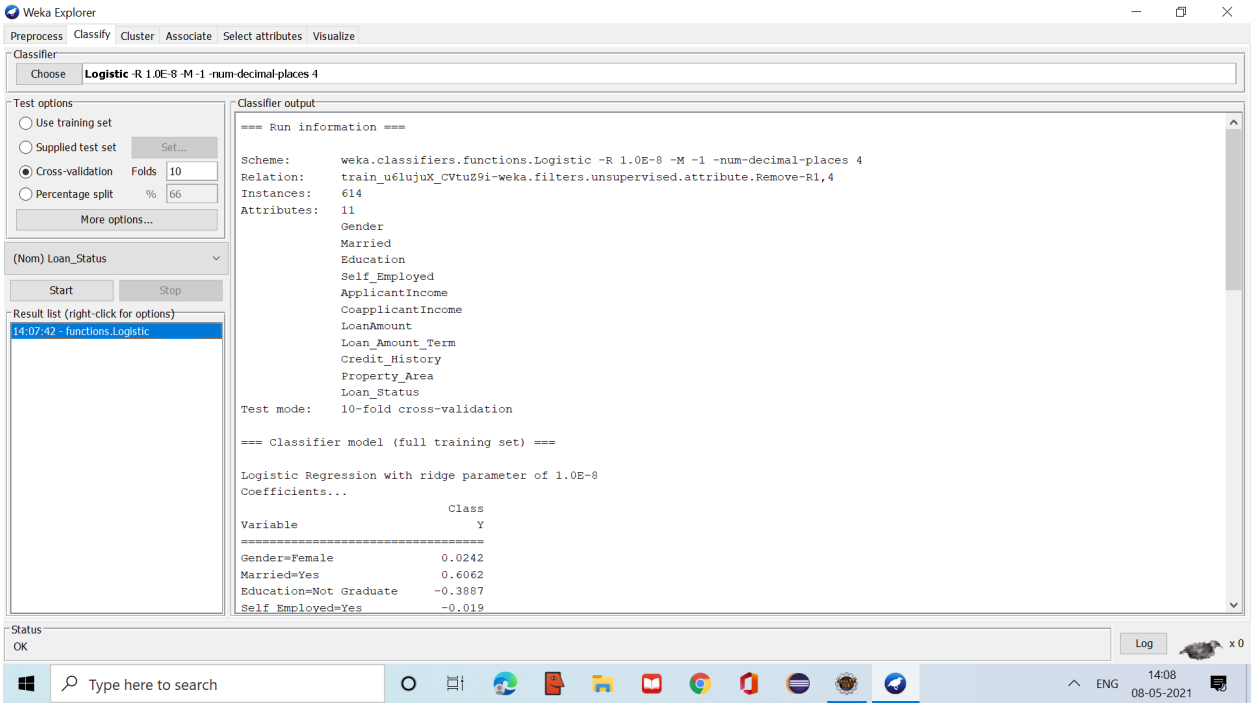
```
            System.out.println(  Arrays.toString(row));
        System.out.println("--------------------");

        System.out.println("Area under the curve");
        System.out.println( eval.areaUnderROC(0));
        System.out.println("--------------------");

        System.out.println(Evaluation.getAllEvaluationMetricNames());

        System.out.print("Recall :");
        System.out.println(Math.round(eval.recall(1)*100.0)/100.0);

        System.out.print("Precision:");
        System.out.println(Math.round(eval.precision(1)*100.0)/100.0);
        System.out.print("F1 score:");
        System.out.println(Math.round(eval.fMeasure(1)*100.0)/100.0);

        System.out.print("Accuracy:");
        double acc = eval.correct()/(eval.correct()+ eval.incorrect());
        System.out.println(Math.round(acc*100.0)/100.0);


        System.out.println("--------------------");
        Instance predicationDataSet = test_data.get(2);
        double value = classifier.classifyInstance(predicationDataSet);
        /** Prediction Output */
        System.out.println("Predicted label:");
        System.out.print(value);


    }
}
```

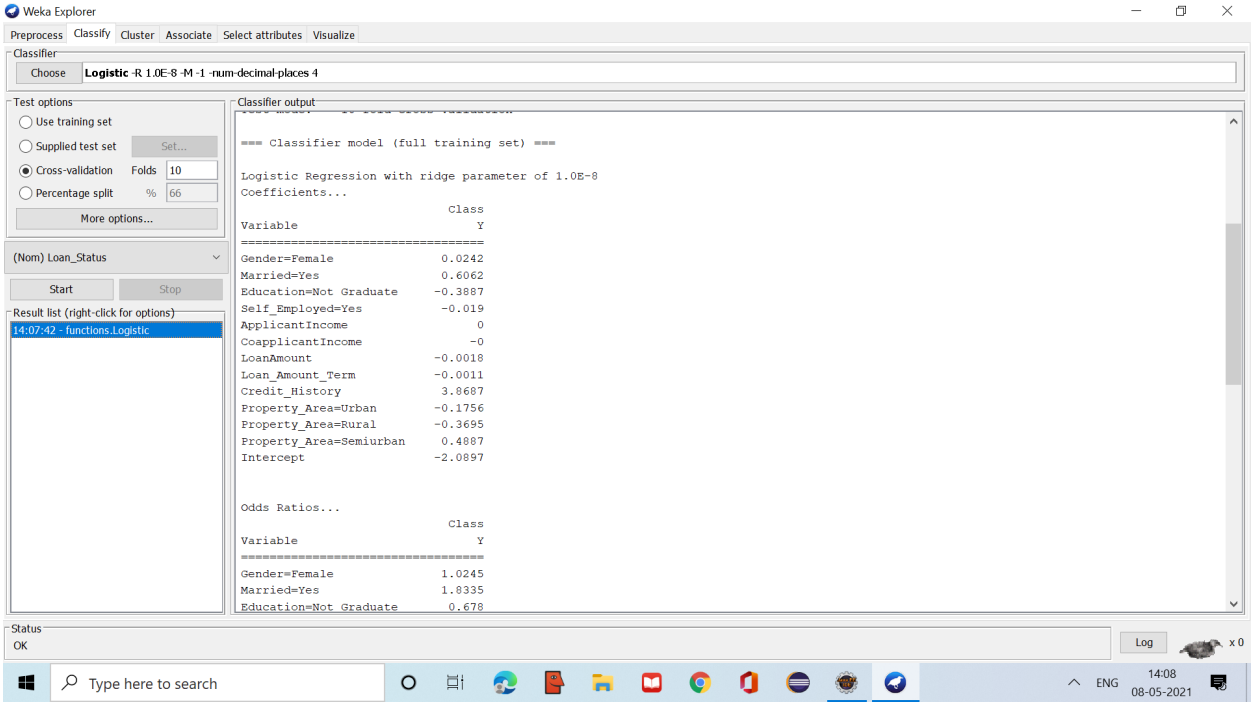
Markers Properties Servers Data Source Explorer Snippets Console

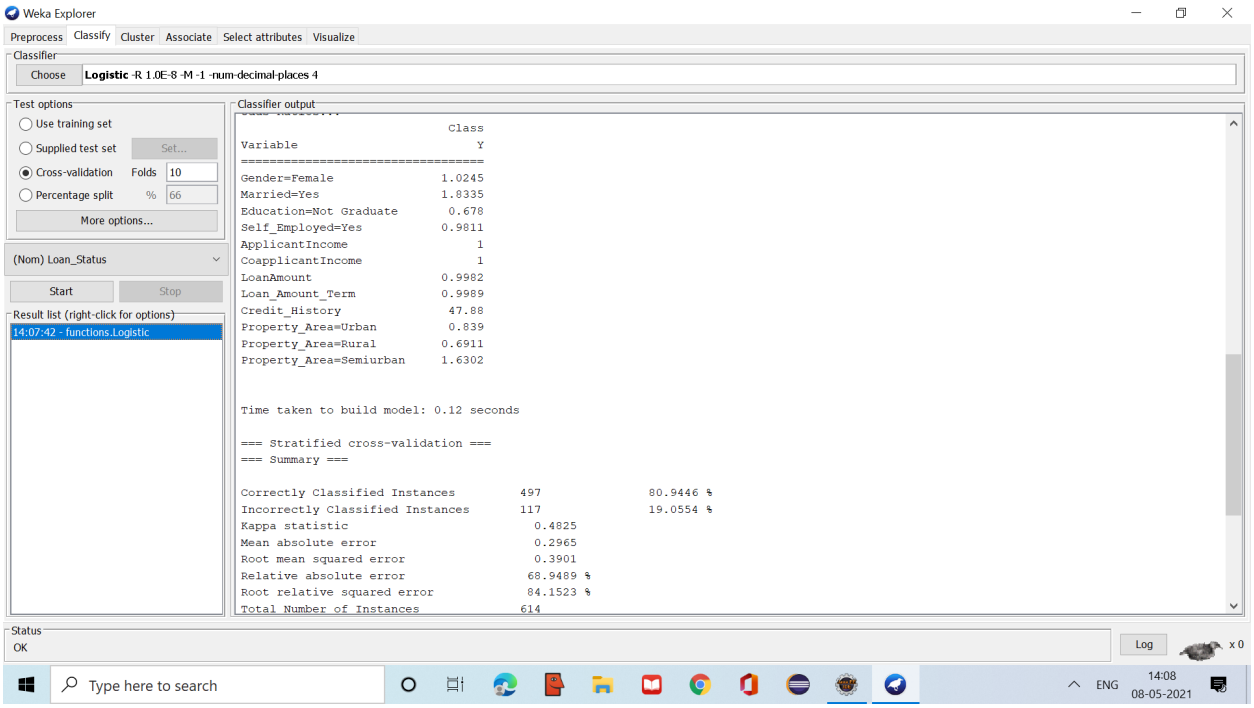
RESULTS:

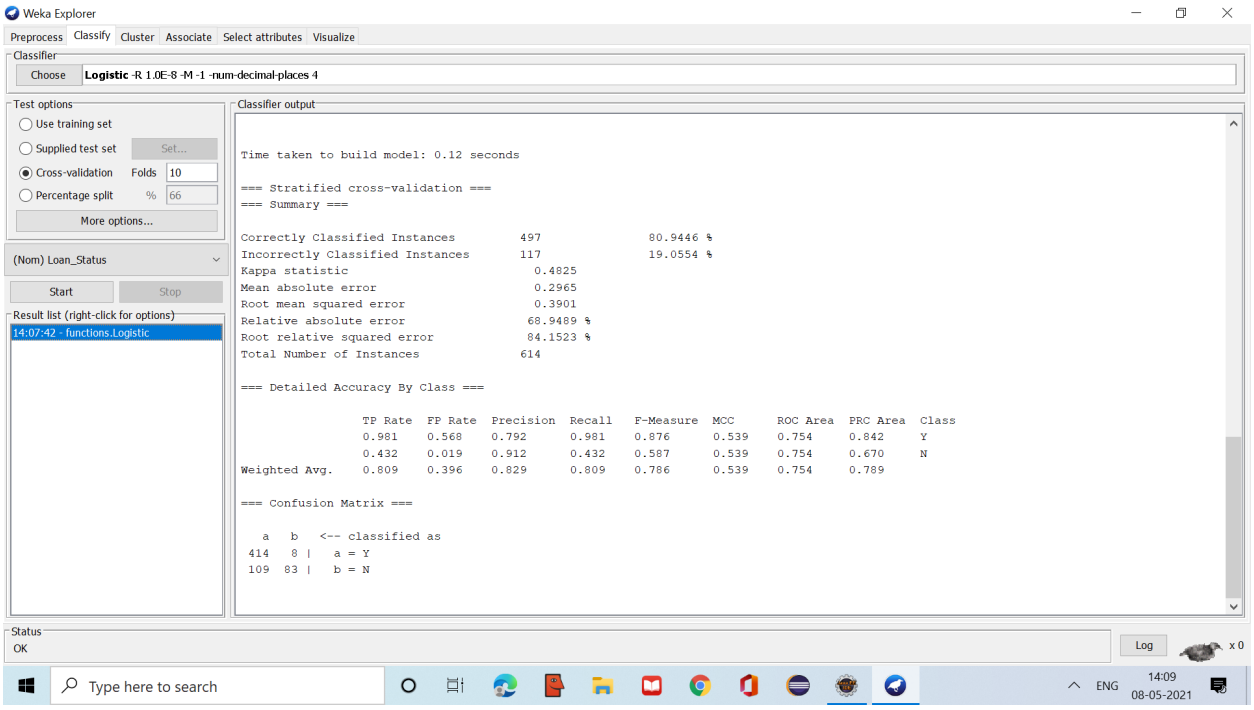


CROSS VALIDATION:




Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
Logistic -R 1.0E-8 -M -1 -num-decimal-places 4
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) Loan_Status
Start
Stop
Result list (right-click for options)
14:07:42 - functions.Logistic
Classifier output
=== Classifier model (full training set) ===
Logistic Regression with ridge parameter of 1.0E-8
Coefficients...
Class
Variable
Y
Gender=Female 0.0242
Married=Yes 0.6062
Education=Not Graduate -0.3887
Self_Employed=Yes -0.019
ApplicantIncome 0
CoapplicantIncome -0
LoanAmount -0.0018
Loan_Amount_Term -0.0011
Credit_History 3.8687
Property_Area=Urban -0.1756
Property_Area=Rural -0.3695
Property_Area=Semiurban 0.4887
Intercept -2.0897
Odds Ratios...
Class
Variable
Y
Gender=Female 1.0245
Married=Yes 1.8335
Education=Not Graduate 0.678
Status
OK
Log
x 0
Type here to search
ENG
14:08
08-05-2021


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
Choose
Logistic -R 1.0E-8 -M -1 -num-decimal-places 4
Test options
Use training set
Supplied test set
Set...
Cross-validation
Folds
10
Percentage split
%
66
More options...
(Nom) Loan_Status
Start
Stop
Result list (right-click for options)
14:07:42 - functions.Logistic
Classifier output
Class
Variable
Y
Gender=Female 1.0245
Married=Yes 1.8335
Education=Not Graduate 0.678
Self_Employed=Yes 0.9811
ApplicantIncome 1
CoapplicantIncome 1
LoanAmount 0.9982
Loan_Amount_Term 0.9989
Credit_History 47.88
Property_Area=Urban 0.839
Property_Area=Rural 0.6911
Property_Area=Semiurban 1.6302
Time taken to build model: 0.12 seconds
=== Stratified cross-validation ===
=== Summary ===
Correctly Classified Instances 497 80.9446 %
Incorrectly Classified Instances 117 19.0554 %
Kappa statistic 0.4825
Mean absolute error 0.2965
Root mean squared error 0.3901
Relative absolute error 68.9489 %
Root relative squared error 84.1523 %
Total Number of Instances 614
Status
OK
Log
x 0
Type here to search
ENG
14:08
08-05-2021



APPLICATIONS:

- Applicable to find whether a customer of bank can get loan or not based on details of the Applicant.
- Helpful to bank for validating whether the applicant will repay the amount .

CONCLUSION:

- Goal of this project is to automate the loan eligibility process (real time) based on customer detail provided while filling online application form.
- The main aim of this use-case is to build a predictive model to predict if an applicant is able to repay the lending company or not.
- Here using logistic regression we built the prediction model and from we have evaluated the testing data with good accuracy.