PROJECT REPORT

TITLE : SMART LENDER- APPLICANT CREDIBILITY PREDICTION FOR LOAN APPROVAL

Team ID : PNT2022TMID27218

Team Leader

Aakash. S (311019104002)

Team Member 1

Aakash. B (311019104001)

Team Member 2

Ajith. G (311019104007)

Team Member 3

Dharshan. S (311019104019)

Project Report Index

1.INTRODUCTION

1.1 PROJECT OVERVIEW

One of the most important factors which affect our country's economy and financial condition is the credit system governed by the banks. The process of bank credit risk evaluation is recognized at banks across the globe. "As we know credit risk evaluation is very crucial, there is a variety of techniques are used for risk level calculation. In addition, credit risk is one of the main functions of the banking community.

The prediction of credit defaulters is one of the difficult tasks for any bank. But by forecasting the loan defaulters, the banks definitely may reduce their loss by reducing their non-profit assets, so that recovery of approved loans can take place without any loss and it can play as the contributing parameter of the bank statement. This makes the study of this loan approval prediction important. Machine Learning techniques are very crucial and useful in the prediction of these types of data.

We will be using classification algorithms such as Decision tree, Random forest, KNN, and XGboost. We will train and test the data with these algorithms. From this best model is selected and saved in PKL format. We will be doing flask integration and IBM deployment.

1.2. PURPOSE

Taking up loans is one of the important financial decisions to make. That too, to take up loans for big investments such as buying a house, pushes the lender to check the status of the borrower's financial side of life. Lenders can easily be profited by the interest borrowers pay for the loans taken up. But it is only one side of the coin. The lender is also putting his money at stake if the borrower fails to pay at any cost.

Loan underwriting is assessing loan eligibility based on an individual's financial status. This includes :

1. Verifying employment

2. Assessing income and assets

3. Examining credit history.

As conducting a manual assessment takes a long time (approx. 60 days), it makes a lot of sense to automate this process.

2. LITERATURE SURVEY

2.1. EXISTING PROBLEM

The problem statement is to develop a Machine Learning model that can predict and evaluate user's eligibility to obtain loans based on the data fed into the model, for which we have carried out the following Literature Survey :

2.2 REFERENCES

1.Loan Credibility Prediction System using Data Mining Techniques

Authors : Anuja Kadam, Pragati Namde, Sonal Shirke, Siddhesh Nandgaonkar, Dr. D.R. Ingle

Published Month & Year : May 2021

Project Description : This model is implemented using the Logistic Regression algorithm. Whenever program takes the input data it gives the output in the form of binary i.e., either 0 or 1. If the output is 1, it indicates that loan is approved. If the output is 0, then it indicates that loan is not approved. Logistic Regression was the best fit with highest accuracy score 81.12%.

Constraints :

- The major limitation of Logistic Regression is the assumption of linearity between the dependent variable and the independent variables.
- It is tough to obtain complex relationships using logistic regression. More powerful and compact algorithms such as Neural Networks can easily outperform this algorithm.

Possible solution :

Decision tree algorithm could be a possible solution to overcome the limitations for the following reasons -

- While utilizing a decision tree algorithm, it is not essential to standardize or normalize the data that has been collected. It can handle both continuous and categorical variables.

- The execution of a Decision tree algorithm must be possible without having to scale the data as well.
- The idea/ concept that drives the decision tree making model is more familiar and easier for developers/ programmers in comparison to other algorithms.

2.An Approach For Prediction Of Loan Approval Using Machine Learning Algorithm

Authors : Ms. Kathe Rutika Pramod, Ms. Panhale Sakshi Dattatray, Ms. Avhad Pooja Prakash, Ms. Dapse Punam Laxman, Mr. Ghorpade Dinesh B.

Published Month & Year : June, 2021.

Project Description : This model is implemented using Decision Tree algorithm. Decision trees are widely used in the banking industry due to their high accuracy and ability to formulate a statistical model in plain language. In Decision tree each node represents a feature (attribute), each link (branch) represents a decision (rule) and each leaf represents an outcome (categorical or continues value). Using different data analytics tools loan prediction and there severity can be forecasted.

Constraints :

- A small change in the data can cause a large change in the structure of the decision tree causing instability.
- For a Decision tree sometimes calculation can go far more complex compared to other algorithms.
- Decision tree often involves higher time to train the model.

Possible Solution :

Random forests are a strong modeling technique and much more robust than a single decision tree. They aggregate many decision

trees to limit overfitting as well as error due to bias and therefore

yield useful results.

3.Credit Risk Model Based on Central Bank Credit Registry Data.

Authors : Fisnik Doko, Slobodan Kalajdziski, Igor Mishkovski.

Published Month & Year : March, 2021.

Project Description : In this Project, different algorithms and models like Logistic Regression, Decision Tree, Random Forests, Support Vector Machines (SVM), and Neural Networks, are evaluated and compared under different cases of Datasets (Imbalanced Data without scaling, Imbalanced Data with scaling, Balanced Data set without scaling).

Constraints :

- This paper uses only one dataset, and all countries have a similar dataset, which can vary by its requirements, laws and roles.
- There is not any research that uses data from credit risk, and we were unable to carry out such a comparison.
- It comparatively takes a lower execution time. But still it fails to provide results with better accuracy.

Possible Solution :

- Multiple classes of Datasets can be explored to extract more potential Variables in order to lessen the impact of Dataset Bias problem.
- To gain Business insights, various analytical tools can be incorporated and integrated into the model.

4.An Approach for Prediction of Loan Approval using Machine Learning Algorithm

Authors : Mohammad Ahmad Sheikh, Amit Kumar Goel

Published Year & Month : May, 2020.

Project Description : This prediction model takes variables like age, Purpose, Credit history, Credit amount, Credit Purpose etc., apart from traditionally considering only account information. The evaluation is finally done by implementing Logistic Regression algorithm. The output of the predicted model will be either 1 or 0. Predicted value 1 shows that the model is classified the application as accepted and predicted value 0 implies that model classified the application has not been accepted.

Constraints :

This model fails to take certain other potential variables into

consideration like -

- Gender
- Marriage History etc.

Possible Solution :

Exhaustive search for all possible variables can be done in order to bring a more accurate and efficient solution.

5.Predict Loan Approval in Banking System Machine Learning Approach for Cooperative Banks Loan Approval

Authors : Amruta S. Aphale, Dr. Sandeep R. Shinde.

Published Month & Year : August, 2020.

Project Description : In this project, different classification algorithms are applied over testing datasets such as Neural Networks, Discriminant Analysis, Naïve Bayes, K-Nearest Neighbor, Linear Regression, Ensemble Learning/Method, Decision Trees. The experiment revealed that, apart from the Nearest Centroid and Gaussian Naive Bayes, the rest of the algorithms perform credibly well in term of their accuracy and other performance evaluation metrics. Each of these algorithms achieved an accuracy rate between 76% to over 80%.

Constraints :

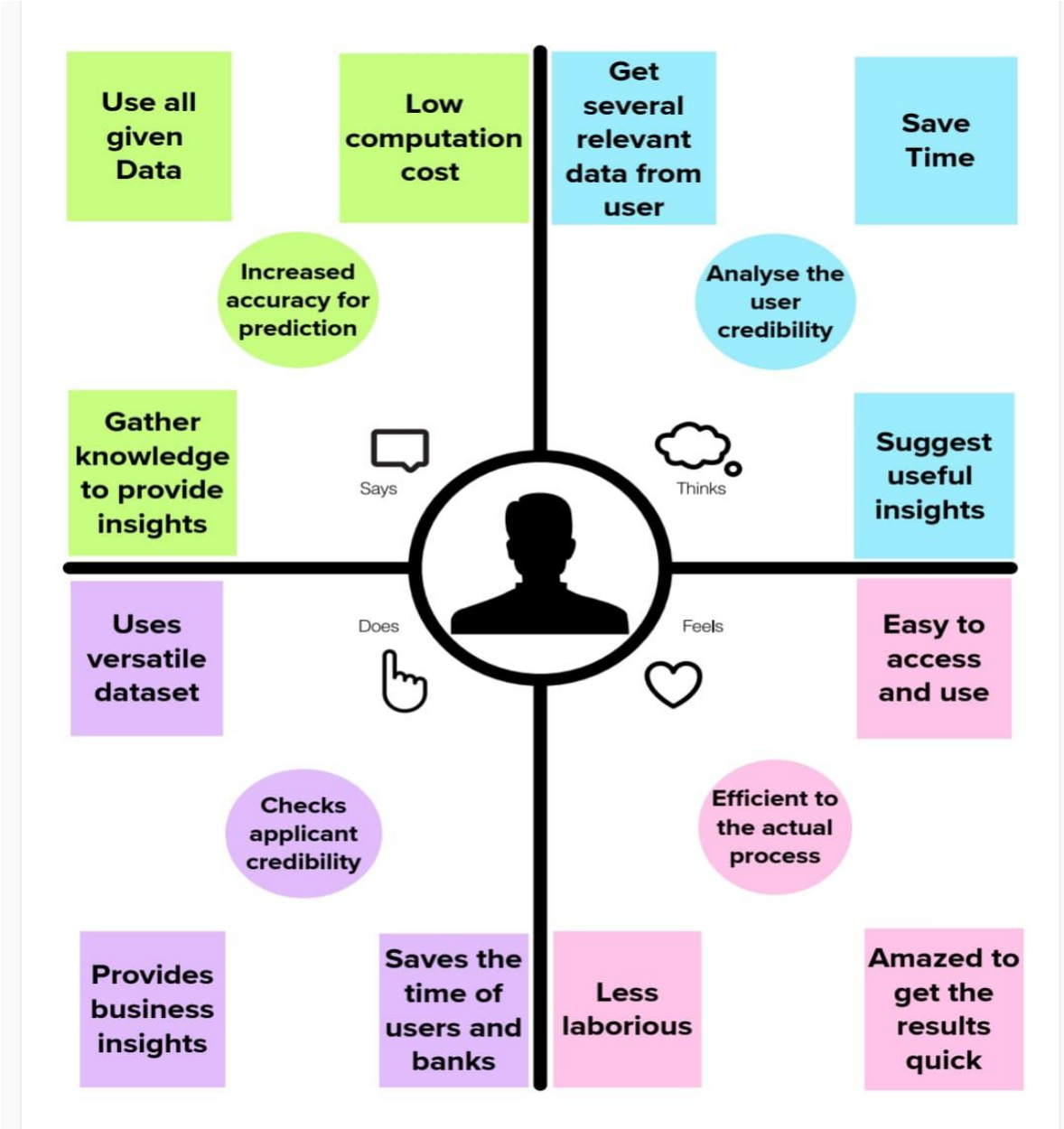
- Even if using Neural Networks is deemed to be more accurate through this experiment, they are more computationally expensive than traditional Algorithms.
- Ensemble learning is less interpretable, the output of the ensembled model is hard to predict and explain. Hence the idea with ensemble is hard to sell and get useful business insights.

Possible solution :

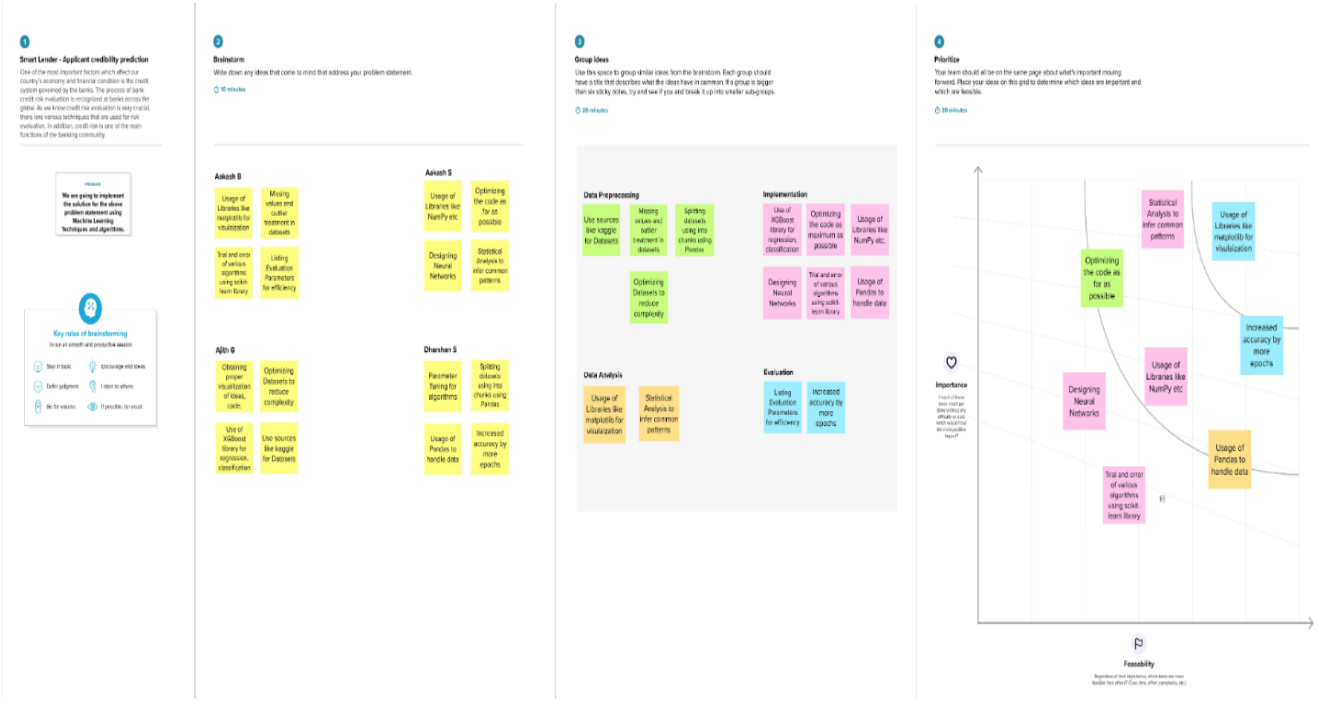
Various Datasets can be used to train the model in the used algorithms to gain much more refined and definite output. This can be further used for gathering Accurate insights on the pre-established factors.

3.IDEATION & PROPOSED SOLUTION

3.1. Empathy Map Canvas



3.2. IDEATION & BRAINSTORMING

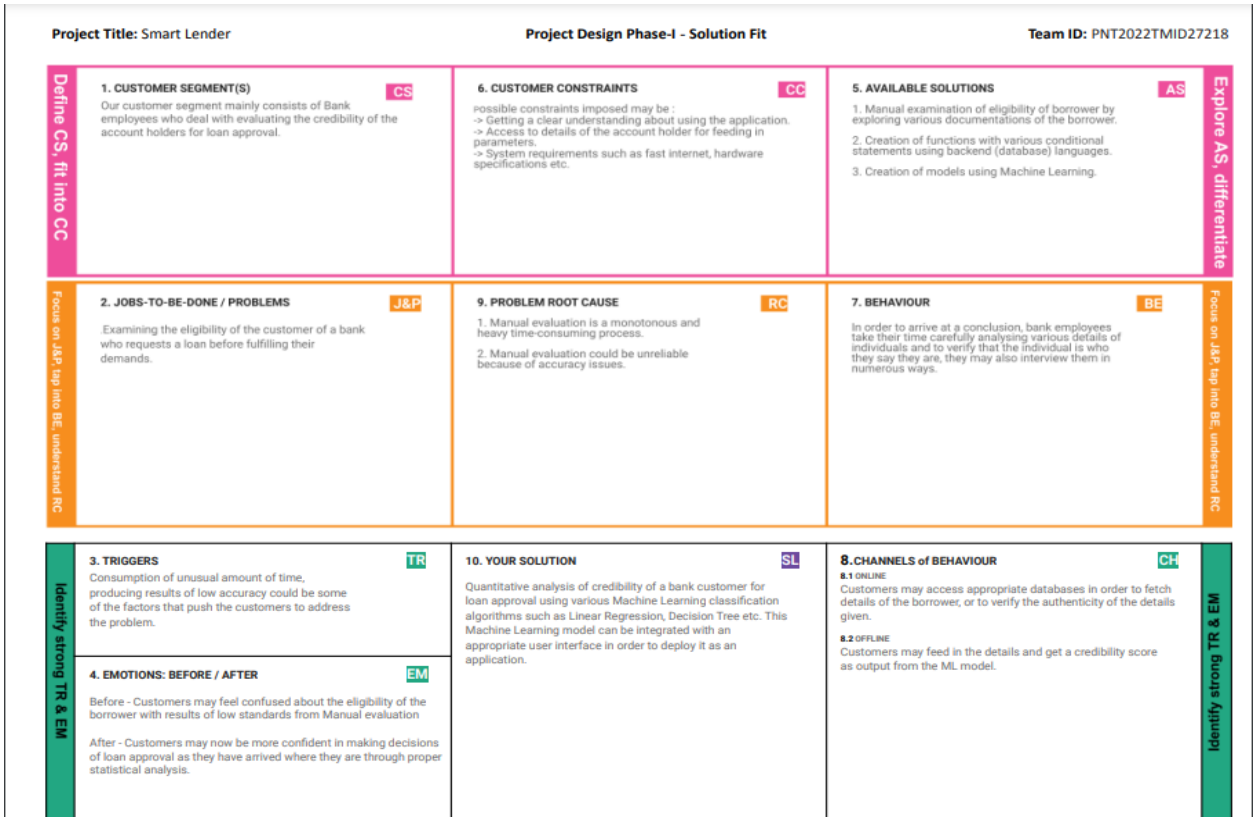


3.3. PROPOSED SOLUTION

S.No	PARAMETER	DESCRIPTION
1.	Problem statement	Smart Lender – A Machine Leaning model for applicant credibility prediction

2.	Idea/Solution Description	By using a machine learning algorithm known as linear regression, we get an idea of how the process is about to be modelled in the following ways: 1. The various liability checks, employment status, credit history and other such aspects that affect process are rated on a scale of 1 to X. 2. These numbers are introduced as variables and they are graphed against each other in a regressive manner, with all the relations between each of the variables 3. The final graph that contains all the variables and their relations plotted against each other gives a straight line which depicts a final value. 4. We then find the maximum or almost maximum range of the final value by giving the variables various use case test values. 5. The Range is then measured on a scale of 1 to 10 for easier understanding.
3.	Novelty/Uniqueness	Predicts the eligibility of the user in an efficient, orderly, and timely manner.
4.	Social Impact/ Customer Satisfaction	Stakeholders need not worry about the Monotonous process of manual credibility assessment.
5.	Business Model/ Revenue Model	The royalties and network traffic by integrating the model to existing services will act as revenue.
6.	Scalability of the solution	As the app is based on ML it is scalable.

3.4. PROBLEM SOLUTION FIT



4. REQUIREMENT ANALYSIS

4.1. FUNCTIONAL REQUIREMENTS

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	New users should be able to register by giving their details such as Email ID, Mobile No. , Password etc. OTPs shall be sent for verification.
FR-2	User Confirmation	The users who have just registered should be able to receive a confirmation/acknowledgement message through E-mail or SMS services.
FR-3	Login process	Existing users must be able to login with their Mail ID/Mobile No. and password. OTPs shall be sent to verify login in special cases. (Forgot password etc.)
FR-4	Credibility assessment	Users should be able to submit all their details required for credibility evaluation through the credibility assessor

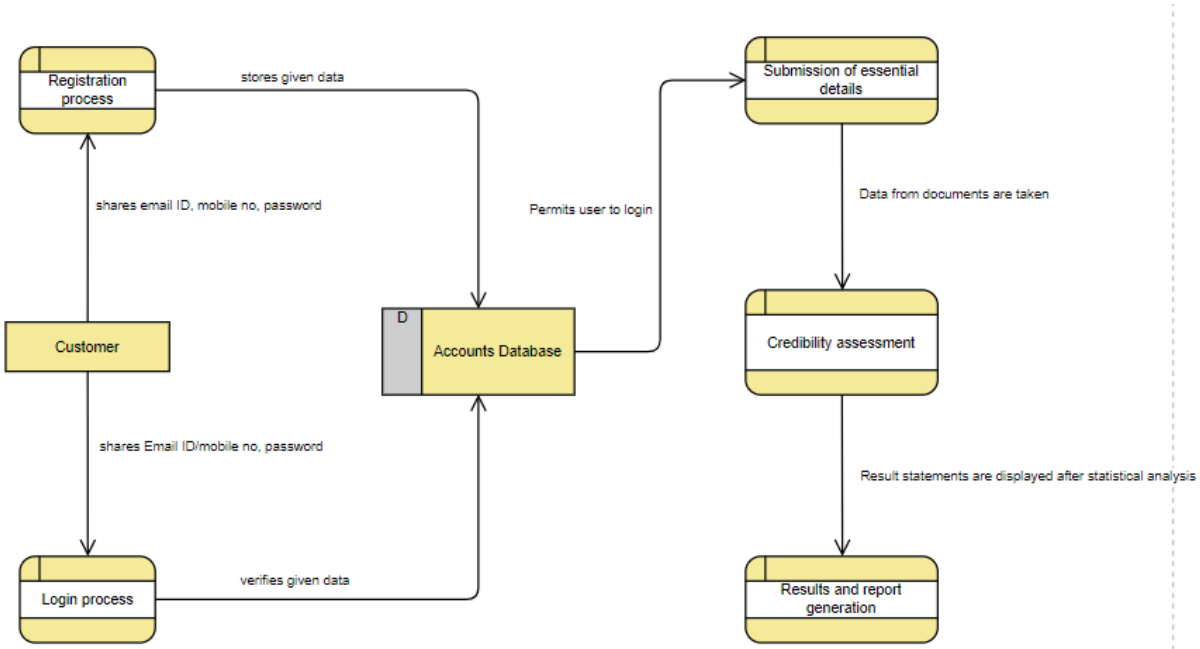
		incorporated as a primary component of the application.
FR-5	Result statements	Users of the application should be able to receive result statements as to what aspects contribute positively in case of YES, and what aspects contribute negatively in case of NO.
FR-6	Report generation	The users should be able to receive comprehensive and understandable reports of results through Mail services etc.

4.2. NON FUNCTIONAL REQUIREMENTS

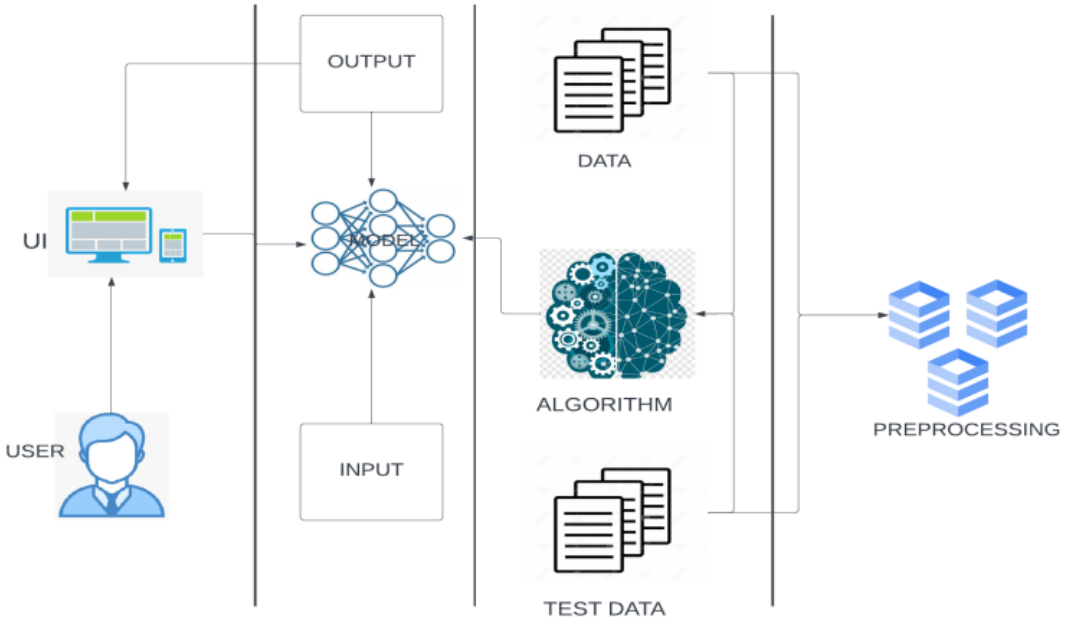
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application should be user-friendly in all aspects, throughout from Login/Registration to Report generation.
NFR-2	**Security**	The basic details given for the access of users to the application such as login details and the details given for the main evaluation process must be strongly secured in order to prevent data theft etc.
NFR-3	**Reliability**	The evaluation results from the application should be accurate in such a way that maximum reliability is achieved.
NFR-4	**Performance**	The users should be able to receive results as quick as possible.
NFR-5	**Availability**	The application should remain operational under all possible fair circumstances of usage.
NFR-6	**Scalability**	The application should be scalable and flexible in order to serve the demands of users over periods of time.

5. PROJECT DESIGN

5.1. DATA FLOW DIAGRAMS



5.2. SOLUTION AND TECHNICAL ARCHITECTURE



5.3. USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration (Case 1)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
Customer (Mobile user)	Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Customer (Mobile user)	Registration (Case 2)	USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
Customer (Mobile user)	Registration (Case 3)	USN-4	As a user, I can register for the application through Gmail	I can register & access the dashboard with Gmail Login	Medium	Sprint-1
Customer (Mobile user)	Login	USN-5	As a user, I can log into the application by entering email & password	I can access the dashboard with the conventional way of login	High	Sprint-1
Customer (Mobile user)	Credibility assessment	USN-6	As a user, I can check if my credibility score matches the eligibility criteria for loan application	I can get an appropriate credibility score	High	Sprint-2
Customer (Mobile user)	Result statements	USN-7	As a user, I can get a readable result statement for both positive and negative cases of results.	I can receive a result statement and understand why the result is positive/negative	High	Sprint-3
Customer (Mobile user)	Report generation	USN-8	As a user, I can get a detailed report of the statistical analysis carried out by the Machine Learning model.	I can receive an email consisting of the report	Medium	Sprint-4

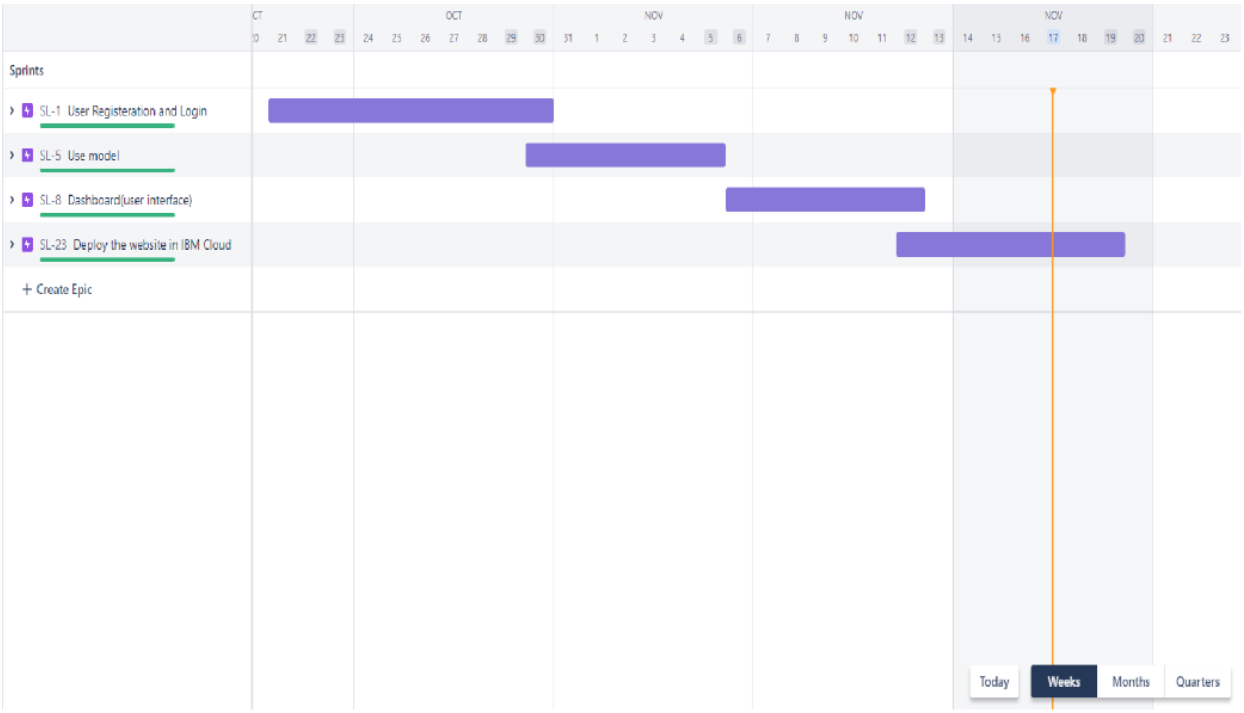
6. PROJECT PLANNING & SCHEDULING

6.1. SPRINT PLANNING, SCHEDULE & ESTIMATION

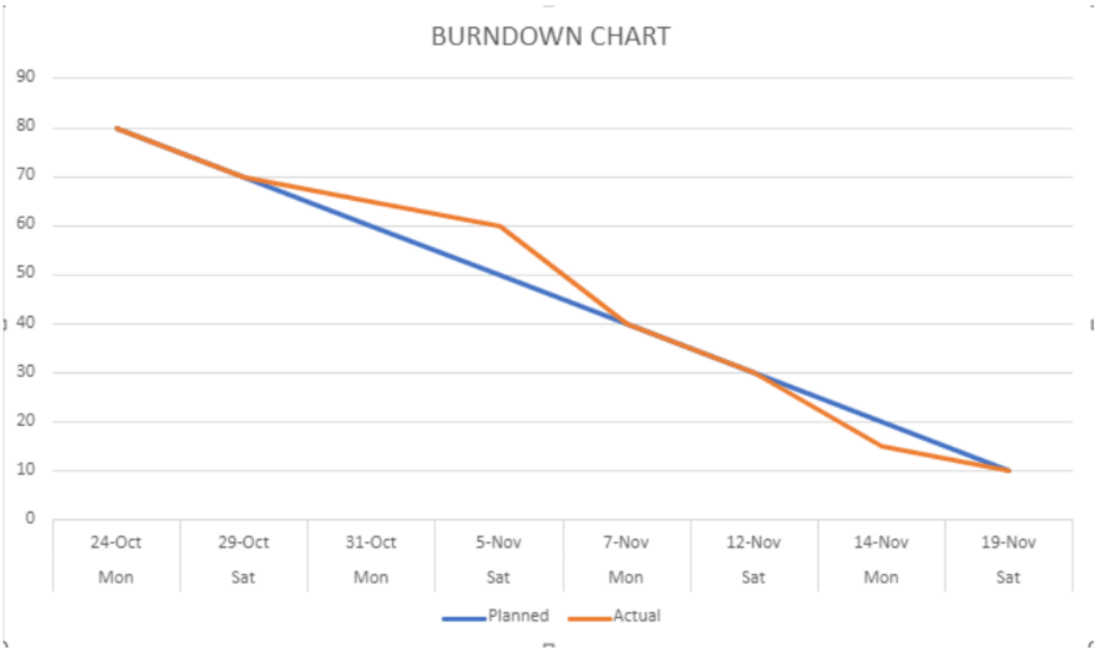
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration (Case 1)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	10	High	Aakash. B, Aakash. S,
Sprint-1	Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	4	High	Ajith. G, Dharshan. S
Sprint-2	Registration (Case 2)	USN-3	As a user, I can register for the application through Facebook	2	Low	Aakash. S, Darshan. S
Sprint-2	Registration (Case 3)	USN-4	As a user, I can register for the application through Gmail	3	Medium	Aakash. S, Dharshan. S
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	6	High	Aakash. S, Dharshan. S
Sprint-2	Information window	USN-6	The interface where a user can submit all their data used for loan eligibility prediction such as credit score, amount demanded, income details etc.	15	High	Aakash. B, Ajith. G
Sprint-3	ML model - development	USN-7	As a user, I can get the results from an ML model to be developed, which can evaluate and assess the loan eligibility of a person by the data given.	10	High	Aakash. B, Aakash. S
Sprint-3	ML model - training	USN-8	As a user, I can get the results from an ML model to be trained by subjecting it to different training datasets.	5	High	Ajith. G, Dharshan. S
Sprint-3	ML model - testing	USN-9	As a user, I can get the results from a trained ML model to be tested by subjecting it to different testing datasets.	5	High	Aakash. S, Ajith. G
Sprint-4	ML model - integration	USN-10	As a user, I can use the model by integrating it with a proper and comprehensible User Interface, thus as a web application	6	High	Aakash. B, Dharshan. S
Sprint-4	Application - testing	USN-11	As a user, I can use a full fledged web application for loan eligibility prediction to be tested for functioning	8	High	Aakash. B, Aakash. S
Sprint-4	Application - Deployment	USN-12	As a user, I can use a full fledged web application for loan eligibility prediction to be deployed in IBM cloud for functioning	6	High	Aakash. B, Aakash. S, Ajith. G, Dharshan. S

6.2. REPORT GENERATION USING JIRA

Jira is a software application used for issue tracking and project management. The tool, developed by the Australian software company Atlassian, has become widely used by agile development teams to track bugs, stories, epics, and other tasks.



BURNDOWN CHART



7.

7. CODING AND SOLUTIONING

7.1. RETRIEVAL OF DATA FOR PREDICTION AND FEEDING THE DATA INTO THE MODEL

```
def predict():
    if request.method == "POST":
        gender = request.form['genderBox']
        married = request.form['maritalBox']
        dependents = request.form['dependents']
        education = request.form['educationBox']
        employment = request.form['employmentBackground']
        applicant_income = request.form['applicantIncomeBox']
        coapplicant_income = request.form['coApplicantIncomeBox']
        loan_amount = request.form['laonAmtBox']
        loan_amount_term = request.form['laonAmtTermBox']
        credit_history = request.form['CHBox']
        prop_area = request.form['propertyAreaBox']


        if gender == 'Male':
            gender = 1
        else:
            gender = 0
        if married == 'Yes':
            married = 1
```

```
else:
    married = 0

if dependents == '0':
   dependents = 0
elif dependents == '1':
   dependents = 1
elif dependents == '2':
   dependents = 2
else:
   dependents = 3

if education == 'Graduate':
   education = 0
else:
   education = 1

if employment == 'Yes':
   employment = 1
else:
   employment = 0
if credit_history=='Yes':
   credit_history=1
```

```
    else:
        credit_history=0
    if prop_area == 'Rural':
        prop_area = 0
    elif prop_area == 'Semiurban':
        prop_area = 1
    else:
        prop_area = 2
x=[[gender,married,dependents,education,employment,applicant_income,coapplicant_income,loan_amount, loan_amount_term,credit_history,prop_area]]

payload_scoring = {"input_data": [{"fields": [[gender,married,dependents, education, employment,applicant_income,coapplicant_income,loan_amount, loan_amount_term,credit_history,prop_area]], "values":x}]}

response_scoring=requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/4222a434-d5ed-4618-8641-96c021c213e3/predictions?version=2022-11-17',json=payload_scoring, headers={'Authorization': 'Bearer ' + mltoken})
print("Scoring response")
prediction=response_scoring.json()
print(response_scoring.json())
if(prediction=="Y"):
        prediction="Yes"
```

```
            return render_template("predict.html", prediction_text="Congratulations Your Loan Status is {}".format(prediction))
else:
        prediction="No"
        return render_template("predict.html", prediction_text=" Your Loan Status is Rejected")
else:
       return render_template("predict.html")
```

In the above flask function, the data for the loan approval prediction process is retrieved from the Loan Approval Prediction form and fed into the model for credibility examination. The code implementation of the core algorithm of the ML model is given below :

7.2. RANDOM FOREST ALGORITHM

```
def randomForest(x_train,x_test,y_train,y_test):
        Rmodel = RandomForestClassifier()
        Rmodel.fit(x_train,y_train)
        pred_test = Rmodel.predict(x_test)
        print('Confusion Matrix')
        print(confusion_matrix(y_test,pred_test))
        print('Classification Report')
        print(classification_report(y_test,pred_test))
        print('Score')
        print(Rmodel.score(x_test,y_test))
```

8. TESTING

8.1. USER ACCEPTANCE TESTING

Purpose Of Document: The purpose of this document is to briefly explain the test coverage and open issues of the [Smart Lender - Applicant Credibility Prediction for Loan Approval] project at the time of the release to User AcceptanceTesting(UAT).

Defect Analysis: This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	0	4	2	0	6
Duplicate	1	0	3	0	4
External	18	10	0	1	29
Fixed	19	14	5	20	58
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	1	2
Won't Fix	0	0	0	0	0
Totals	38	28	12	22	99

Test Analysis : This report shows the number of test cases that have passed, failed, untested.

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	10	0	0	10
Client Application	10	0	0	10
Security	10	0	0	10

Outsource Shipping	10	0	0	10
Exception Reporting	10	0	0	10
Final Report Output	10	0	0	10
Version Control	10	0	0	10

9. RESULTS

9.1. PERFORMANCE METRICS

DECISION TREE

Decision trees may be used to forecast numerical values (regression) as well as categorize data.

Confusion Matrix

[[49 14]

[20 58]]

Classification Report

	precision	recall	f1-score	support
0	0.71	0.78	0.74	63
1	0.81	0.74	0.77	78
accuracy			0.76	141
macroavg	0.76	0.76	0.76	141
weightedavg	0.76	0.76	0.76	141

Score:0.7588652482269503

RANDOM FOREST

In a random forest, the machine learning algorithm predicts a value or category by combining the results from a number of decision trees.The random forest algorithm is a bagging technique extension that use both bagging and feature randomization to produce an uncorrelated forest of decision trees.

Confusion Matrix :

[[52 0]
 [20 113]]

Classification Report :

	precision	recall	f1-score	support
0	0.72	1.00	0.84	52
1	1.00	0.85	0.92	133
accuracy			0.89	185
macro avg	0.86	0.92	0.88	185
weighted avg	0.92	0.89	0.90	185

Score : 0.8974358974358975

K-NEAREST NEIGHBORS ALGORITHM

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point.

Confusion Matrix :
[[40 27]
 [26 50]]

Classification report :

	precision	recall	f1-score	support
0	0.61	0.60	0.60	67
1	1.00	0.85	0.92	133
accuracy			0.63	143
macro avg	0.63	0.63	0.63	143
weighted avg	0.63	0.63	0.63	143

Score : 0.6293706293706294

XGBOOST

XGBoost, or Extreme Gradient Boost, is a machine learning technique used to create gradient boosting decision trees. When it comes to unstructured data, such as photos and unstructured text data, ANN models (Artificial neural network) appear to be at the top of the list when it comes to prediction.

Confusion Matrix :
[[53 16]
 [25 44]]

Classification Report :

	precision	recall	f1-score	support
0	0.68	0.77	0.72	69
1	0.73	0.74	0.68	69
accuracy			0.70	138
macro avg	0.86	0.92	0.88	185
weighted avg	0.92	0.89	0.90	185

Score : 0.7028985507246377

EVALUATION OF PERFORMANCE

When all of the above algorithms are evaluated and compared, it can be seen that the Random Forest algorithm has the highest rate of accuracy i.e., 0.8974358974358975. So we preferably use this algorithm for loan approval prediction.

10. ADVANTAGES & DISADVANTAGES

Random Forest is a powerful algorithm in Machine Learning. It is based on the Ensemble Learning technique (bagging). Following are the advantages and disadvantages of Random Forest algorithm.

Advantages :

1. Random Forest is based on the bagging algorithm and uses Ensemble Learning technique. It creates as many trees on the subset of the data and combines the output of all the trees. In this way it reduces overfitting problem in decision trees and also reduces the variance and therefore improves the accuracy.

2. Random Forest can be used to solve both classification as well as regression problems.

3. Random Forest works well with both categorical and continuous variables.

4. Random Forest can automatically handle missing values.

5. No feature scaling required: No feature scaling (standardization and normalization) required in case of Random Forest as it uses rule based approach instead of distance calculation.

Disadvantages :

1. Random Forest creates a lot of trees (unlike only one tree in case of decision tree) and combines their outputs. By default, it creates 100 trees in Python sklearn library. To do so, this algorithm requires much more computational power and resources. On the other hand decision tree is simple and does not require so much computational resources.

2. Random Forest require much more time to train as compared to decision trees as it generates a lot of trees (instead of one tree in case of decision tree) and makes decision on the majority of votes.

11. CONCLUSION

The developmental phase hence begins with collection and preprocessing of datasets, followed by different kinds of visualizations and analysis of data, model development, training, testing and evaluation. The model is said to have the best accuracy only when it visibly overcomes several barriers such as overfitting, underfitting etc. through subjecting the model to different testing datasets. This detailed project report can assist as to give a basic idea of the application from a user's point of view, explore the technicalities of the application and also give a clear walkthrough of the development of this application from the beginning.

12. FUTURE SCOPE

This elementary web application can be commercially scaled up in many ways such as :

- This application can be professionally used by public and private banks to speed up the process of Loan underwriting.

- This application can be completely deployed and released as a full fledged web application for public use, in which different ways of automated assistance such as

ChatBots etc. can be integrated for assisting the users to navigate through the application and also for credibility increment consultation.

13. APPENDIX

13.1. SOURCE CODE - FRONTEND

Home.html :

```
<html>
 <head>
    <meta charset="UTF-8">
    <h1><center>Welcome to Loan Approval Prediction</center></h1>
    <style>
       body{
                                                                    background-image:
url("https://w0.peakpx.com/wallpaper/34/887/HD-wallpaper-loan-3d-icon-white-background-3d-symbols-loan-finance-icons-3d-icons-loan-sign-business-3d-icons.jpg");
          background-size: 1600px;
          background-repeat: no-repeat;
          background-position: center;
       }
    </style>
  </head>
  <body>
    <p style="font-size: 21px;">
       <center>
            Obtaining Loans is a tedious process, as it involves a lot of background check and verification of credibility of the borrower.<br>
        We made it simpler in a way that you can get your loan approval prediction using our Machine Learning model, for which we need some of your information.<br>
       Please click the "Predict" button below in order to predict your credibility status for loan approval<br><br>
    <a href="predict.html"><button>PREDICT</button></a>
       </center>
  </body>
```

```
</html>
```

Predict.html :

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Loan Predictor</title>
    <style>
      .inputfield {
        width: 100%
      }

      body {
        background-color: black;
        font-family: sans-serif;
        color: white;
      }

      form {
        background-color: #15172b;
        border-radius: 20px;
        box-sizing: border-box;
        height: 820px;
        padding: 20px;
        width: 100%;
        color: white;
        font-family: sans-serif;
      }
```

```
  </style>
</head>

<body>
    <h2>LOAN PREDICTOR FORM</h2>
    <p>Enter your details for loan approval prediction</p>
    <h2><b>{{prediction_text}}</b></h2>
    <form action="/predict" method='POST'>

     <div class="form">

      <div class="inputfield">
        <label>Gender</label>
        <div class="custom_select">
          <select name="genderBox">
            <option value="">Select</option>
            <option value="Male">Male</option>
            <option value="Female">Female</option>
          </select>
        </div>
      </div><br>

      <div class="inputfield">
        <label>Married</label>
        <div class="custom_select">
          <select name="maritalBox">
            <option value="">Select</option>
            <option value="Yes">Yes</option>
            <option value="No">No</option>
          </select>
        </div>
      </div><br>
```

```
 <div class="inputfield">
   <label>Dependents</label>
   <div class="custom_select">
    <select name="dependents">
     <option selected disabled hidden>Select</option>
     <option value="0">0</option>
     <option value="1">1</option>
     <option value="2">2</option>
     <option value="3+">3</option>
 </select>
</div>
</div><br>

 <div class="inputfield">
   <label>Education</label>
   <div class="custom_select">
     <select name="educationBox">
       <option value="">Select</option>
       <option value="Graduate">Graduate</option>
       <option value="NonGraduate">Non Graduate</option>
     </select>
   </div>
 </div><br>

<div class="inputfield">
   <label>Self Employed</label>
   <div class="custom_select">
     <select name = "employmentBackground">
       <option value="">Select</option>
       <option value="Yes">Yes</option>
       <option value="No">No</option>
     </select>
   </div>
```

```
</div><br>

<div class="inputfield">
  <label>Applicant Income (Monthly)</label><br>
  <input type="number" class="input" name="applicantIncomeBox" min="0">
</div> <br>

<div class="inputfield">
  <label>Co Applicant Income</label><br>
  <input type="number" class="input" name="coApplicantIncomeBox" min="0">
</div> <br>

<div class="inputfield">
  <label>Loan  Amount</label><br>
  <input type="number" class="input" name="laonAmtBox" min="0">
</div> <br>

<div class="inputfield">
    <label>Loan Amount Term</label><br>
    <input type="number" class="input" name="laonAmtTermBox" min="0">
</div> <br>

<div class="inputfield">
    <label>Credit History</label>
    <div class="custom_select">
      <select name = "CHBox">
        <option value="">Select</option>
        <option value="Yes">Yes</option>
        <option value="No">No</option>
      </select>
    </div>
</div> <br>
```

```
        <div class="inputfield">
           <label>Property Area</label>
           <div class="custom_select">
             <select name = "propertyAreaBox">
               <option value="">Select</option>
               <option value="Rural">Rural</option>
               <option value="SemiUrban">Semi Urban</option>
               <option value="Urban">Urban</option>
             </select>
           </div>
        </div> <br>

         <div class="inputfield">
               <style>
                  button{
                     background-color: #08d;
                     border-radius: 12px;
                     color: white;
                     padding: 6px 20px;
                  }
               </style>

               <button type="submit" class="btn">
                  SUBMIT
               </button>
         </div>
      </form>
</body>
</html>
```

13.2. SOURCE CODE - BACKEND

Importing required packages :

```
import numpy as np
import pandas as pd
import pickle
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
import sklearn
from sklearn.preprocessing import LabelEncoder
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import GradientBoostingClassifier,RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import RandomizedSearchCV
from xgboost import XGBClassifier
from sklearn.ensemble import RandomForestClassifier
import imblearn
from imblearn.under_sampling import RandomUnderSampler
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import scale
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score,classification_report,confusion_matrix,f1_score
```

Loading the dataset :

```
df=pd.read_csv('/content/drive/MyDrive/IBM/loan_prediction.csv')
df
```

To remove warnings :

```
import warnings
warnings.filterwarnings('ignore')
```

Checking for null values :

```
df.isnull().any()
df['LoanAmount']=df['LoanAmount'].fillna(df['LoanAmount'].mean())
df['Loan_Amount_Term']=df['Loan_Amount_Term'].fillna(df['Loan_Amount_Term'].mean())
df['Credit_History']=df['Credit_History'].fillna(df['Credit_History'].mean())
df['Gender']=df['Gender'].fillna(df['Gender'].mode()[0])
df['Married']=df['Married'].fillna(df['Married'].mode()[0])
df['Dependents']=df['Dependents'].fillna(df['Dependents'].mode()[0])
df['Self_Employed']=df['Self_Employed'].fillna(df['Self_Employed'].mode()[0])
df.isnull().any()
df.isnull().sum()
```

Handling categorical values :

```
df.head()
le=LabelEncoder()
df.Gender=le.fit_transform(df.Gender)
df.Married=le.fit_transform(df.Married)
df.Education=le.fit_transform(df.Education)
df.Self_Employed=le.fit_transform(df.Self_Employed)
df.Property_Area=le.fit_transform(df.Property_Area)
df.Loan_Status=le.fit_transform(df.Loan_Status)
df.Dependents=le.fit_transform(df.Dependents)
df.head()
```

Splitting into dependent and independent data :

```
df=df.drop(columns=["Loan_ID"],axis=1)
df.head()
x=df.iloc[:,:-1]
```

```
y=df.Loan_Status
x.head()
y.head()
```

Scaling the data :

```
x_scale=pd.DataFrame(scale(x),columns=x.columns)
x_scale.head()
```

Balancing the dataset :

```
sns.countplot(df.Loan_Status)
rus=RandomUnderSampler(sampling_strategy=1)
x_res,y_res=rus.fit_resample(x,y)
ax=y_res.value_counts().plot.pie(autopct='%.2f')
_=ax.set_title("under-sampling")
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.3,random_state=10)
xtrain.head()
xtest.head()
ytrain.head()
ytest.head()
xtrain.shape
xtest.shape
ytrain.shape
ytest.shape
```

Model Building :

Decision Tree :

```
dmodel=DecisionTreeClassifier(random_state=100)
dmodel.fit(x_res,y_res)
DecisionTreeClassifier(random_state=100)
ypredd=dmodel.predict(xtest)
ypred2d=dmodel.predict(xtrain)
print("Decision Tree Model Testing Accuracy")
```

```
print(accuracy_score(ytest,ypredd))
print("Decision Tree Model Training Accuracy")
print(accuracy_score(ytrain,ypred2d))
```

Random Forest :

```
Rmodel=RandomForestClassifier(n_estimators=100)
Rmodel.fit(x_res,y_res)
RandomForestClassifier()
ypredR=Rmodel.predict(xtest)
ypred2R=Rmodel.predict(xtrain)
print("Random Forest Model Testing Accuracy")
print(accuracy_score(ytest,ypredR))
print("Random Forest Model Training Accuracy")
print(accuracy_score(ytrain,ypred2R))
```

KNN :

```
kmodel=KNeighborsClassifier()
kmodel.fit(x_res,y_res)
KNeighborsClassifier()
ypredk=kmodel.predict(xtest)
ypred2k=kmodel.predict(xtrain)
print("KNN Model Testing Accuracy")
print(accuracy_score(ytest,ypredk))
print("KNN Model Training Accuracy")
print(accuracy_score(ytrain,ypred2k))
```

XGBoost :

```
xmodel=XGBClassifier(eval_metric='mlogloss',n_estimators=100,random_state=100)
xmodel.fit(x_res,y_res)
XGBClassifier(eval_metric='mlogloss', random_state=100)
ypredx=xmodel.predict(xtest)
ypred2x=xmodel.predict(xtrain)
print("Xgboost Model Testing Accuracy")
```

```
print(accuracy_score(ytest,ypredx))
print("Xgboost Model Training Accuracy")
print(accuracy_score(ytrain,ypred2x))
```

Comparison of models :

```
print("Decision Tree Model Testing Accuracy")
print(accuracy_score(ytest,ypredd))
print("Decision Tree Model Training Accuracy")
print(accuracy_score(ytrain,ypred2d))

print("Random Forest Model Testing Accuracy")
print(accuracy_score(ytest,ypredR))
print("Random Forest Model Training Accuracy")
print(accuracy_score(ytrain,ypred2R))

print("KNN Model Testing Accuracy")
print(accuracy_score(ytest,ypredk))
print("KNN Model Training Accuracy")
print(accuracy_score(ytrain,ypred2k))

print("Xgboost Model Testing Accuracy")
print(accuracy_score(ytest,ypredx))
print("Xgboost Model Training Accuracy")
print(accuracy_score(ytrain,ypred2x))
```

Evaluating the performance and saving the model :

```
print("Random Forest Model Testing Accuracy")
print(accuracy_score(ytest,ypredR))
print("Random Forest Model Training Accuracy")
print(accuracy_score(ytrain,ypred2R))

f1_score(ypredR,ytest,average='weighted')
```

```
pd.crosstab(ytest,ypredR)

print(confusion_matrix(ytest,ypredR))

print(classification_report(ytest,ypredR))
```

Saving the model :

```
pickle.dump(Rmodel,open('Rmodel.pkl','wb'))
pickle.dump(x_scale,open('scale.pkl','wb'))
```

GITHUB REPOSITORY LINK :

https://github.com/IBM-EPBL/IBM-Project-10980-1659249168

PROJECT DEMONSTRATION VIDEO LINK :

https://share.vidyard.com/watch/YCdzzsP2bEtobd1C6Htw5X