PROJECT REPORT

PROJECT NAME	: SMART LENDER-APPLICANT CREDIBILITY FOR LOAN APPROVAL
TEAM ID	: PNT2022TMID49425
TEAM SIZE	: 4
TEAM LEADER	: M.POOJA
TEAM MEMBERS	: G.MOHANAPRIYA R.SIVASATHYA M.YUVARANI

CONTENTS:

1.INTRODUCTION

1.1 PROJECT OVERVIEW

A loan is the core business part of banks. The main portion the bank’s profit is directly come from the profit earned from the loans. Though bank approves loan after a regress process of verification and testimonial but still there's no surety whether the chosen hopeful is the right hopeful or not. This process takes fresh time while doing it manually. We can prophesy whether that particular hopeful is safe or not and the whole process of testimonial is automated by machine literacy style. Loan Prognostic is really helpful for retainer of banks as well as for the hopeful also.

1.2 PURPOSE

The purpose of loan is to borrow money. When we fill out a loan application, we might come across a section that asks why we are applying. Some lenders do this to match you with the right product. we can also use our loan purpose to assess risk and assign loan terms.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM :

Bank employees check the details of applicant manually and give the loan to eligible applicant. Checking the details of all applicants takes lot of time. The artificial neural network model for predict the credit risk of a bank. The Feed- forward back propagation neural network is used to forecast the credit default. The method in which two or more classifiers are combined together to produce a ensemble model for the better prediction. They used the bagging and boosting techniques and then used random forest technique. The process of classifiers is to improve the performance of the data and it gives better efficiency. In this work, the authors describe various ensemble techniques for binary classification and also for multi class classification. The new technique that is described by the authors for ensemble is COB which gives effective performance of classification but it also compromised with noise and outlier data of classification.finally they concluded that the ensemble based algorithm improve the results for training data set.

Drawback of existing system:

checking details of all applicants consumed lot of time and effects.there is chance of human error may occur due checking all details manully.there is possibility of assigning loan to ineligible applicant

2.2 REFERENCES :

- Kumar Arun, Garg Ishan, Kaur Sanmeet, May-Jun. 2016. Loan Approval Prediction based on Machine Learning Approach, IOSR Journal of Computer ENGINEERING(IOSR-JCE)
- Wei Li, Shuai Ding, Yi Chen, and Shanlin Yang, Heterogeneous Ensemble for Default Prediction of Peer-to-Peer Lending in China, Key Laboratory of Process Optimization and Intelligent Decision-Making, Ministry of Education, Hefei University of Technology, Hefei 2009, China
- Short-term prediction of Mortgage default using ensembled machine learning models, Jesse C.Sealand on july 20, 2018.
- Clustering Loan Applicants based on Risk Percentage using K-Means

Clustering Techniques, Dr. K. Kavitha, International Journal of Advanced Research in Computer Science and Software Engineering

2.3 PROBLEM STATEMENT DEFINITION :

Understanding the Problem Statement

Dream Housing Finance company deals in all kinds of home loans. They have a presence across all urban, semi-urban and rural areas. The customer first applies for a home loan and after that, the company validates the customer eligibility for the loan.

The company wants to automate the loan eligibility process (real-time) based on customer detail provided while filling out online application forms. These details are Gender, Marital Status, Education, number of Dependents, Income, Loan Amount, Credit History, and others.

To automate this process, they have provided a dataset to identify the customer segments that are eligible for loan amounts so that they can specifically target these customers.

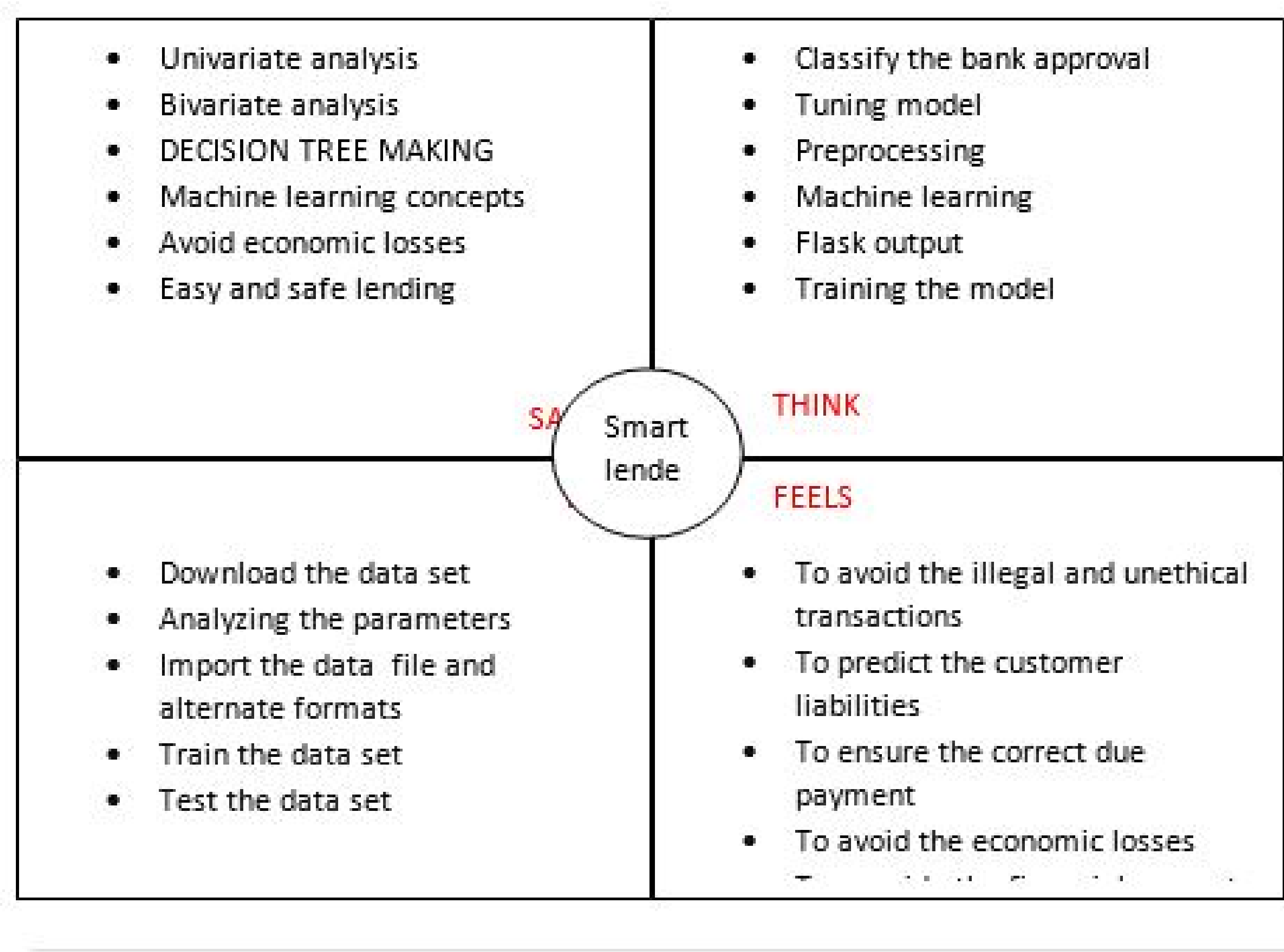
You can find the complete details about the problem statement here and also download the training and test data.

I am	Expecting the lending with proper system without any financial loss	The customer is expecting a smart system and predicting the eligibility of clients
I am trying to	To achieve the loan prediction and to make the loans and raise funds to the common people	To achieve and avoid the persons financial crisis and to make the next step for career (or) development
But	In this prediction the dataset should be trained and tested without any errors and accuracy should be higher	The dataset training and testing should be precise if not the person cannot get loans
Because	This problem can be resolved by using the machine learning	The problem can be resolved by logistic regression, machine learning, concept

Which makes me feel	The person can develop in the society with great entrepreneurship	Sometimes the people expect loan for developing their career (or)talking the career to next step

3.IDEATION&PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS:



- Univariate analysis
- Bivariate analysis
- DECISION TREE MAKING
- Machine learning concepts
- Avoid economic losses
- Easy and safe lending

- Classify the bank approval
- Tuning model
- Preprocessing
- Machine learning
- Flask output
- Training the model

- Download the data set
- Analyzing the parameters
- Import the data file and alternate formats
- Train the data set
- Test the data set

- To avoid the illegal and unethical transactions
- To predict the customer liabilities
- To ensure the correct due payment
- To avoid the economic losses

3.2 IDEATION AND BRAINSTORMING :

PROBLEM STATEMENT

How to make lending easier and simpler without the financial loss ?

The lending can be simpler if the device is smart and easy to acces the and discussion of the team as follows

Team member 1 idea:

The idea of the team member 1 is making the project with the univariate analysis with the salary proof should be considered .The client's basic details should be considered as the eligibility of the customer .The efficient transaction will be the lender satisfaction and the receiver repayment.

Team member 2 idea:

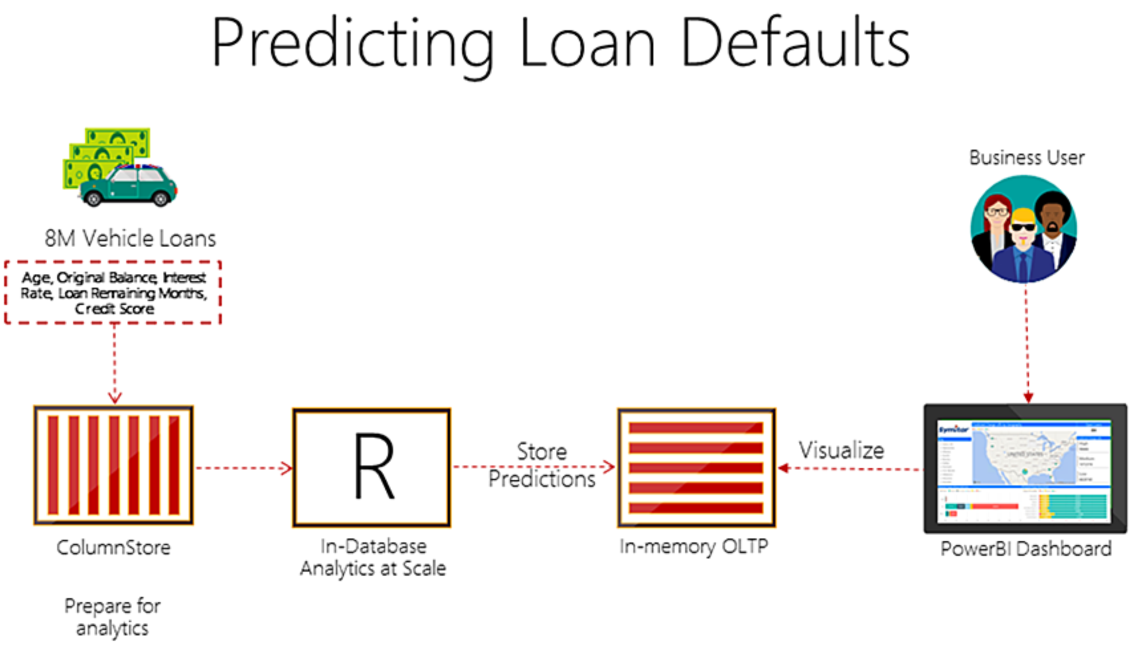
The idea of the team member 2 is using the bivariate analysis with the regression concept will be easier to complete this project and this lending system should be quite complex to crack with hacking and should be even and sharp in the data file entered and much better prediction.

Team member 3 idea:

*T*he idea of the team member 3 is using the decision tree will be easier to predict than the other schemes but the concept is using the python will be platform independent language and the software prediction is must in the lending process .The concept should be consider every details and the age consideration including the monthly expenses of the family.

Team member 4 idea:

The idea of the team member 4 is using the machine learning concept and using the variate analysis and the basic details should be consider[redacted]ed and the age and the salary wage and the details should be considered .The regression can be used and machine learning data science and analytic will be easier for prediction.



PRIORITIZED IDEA:

The prioritized idea will be using the univariate bivariate analysis and the basic details should be considered with the machine learning and decision making and lending should be proper .Data science and analysis is used in the smart lending system .The parameters should be considered are age, gender,salary details and several important parameters.

3.3 PROPOSED SOLUTION:

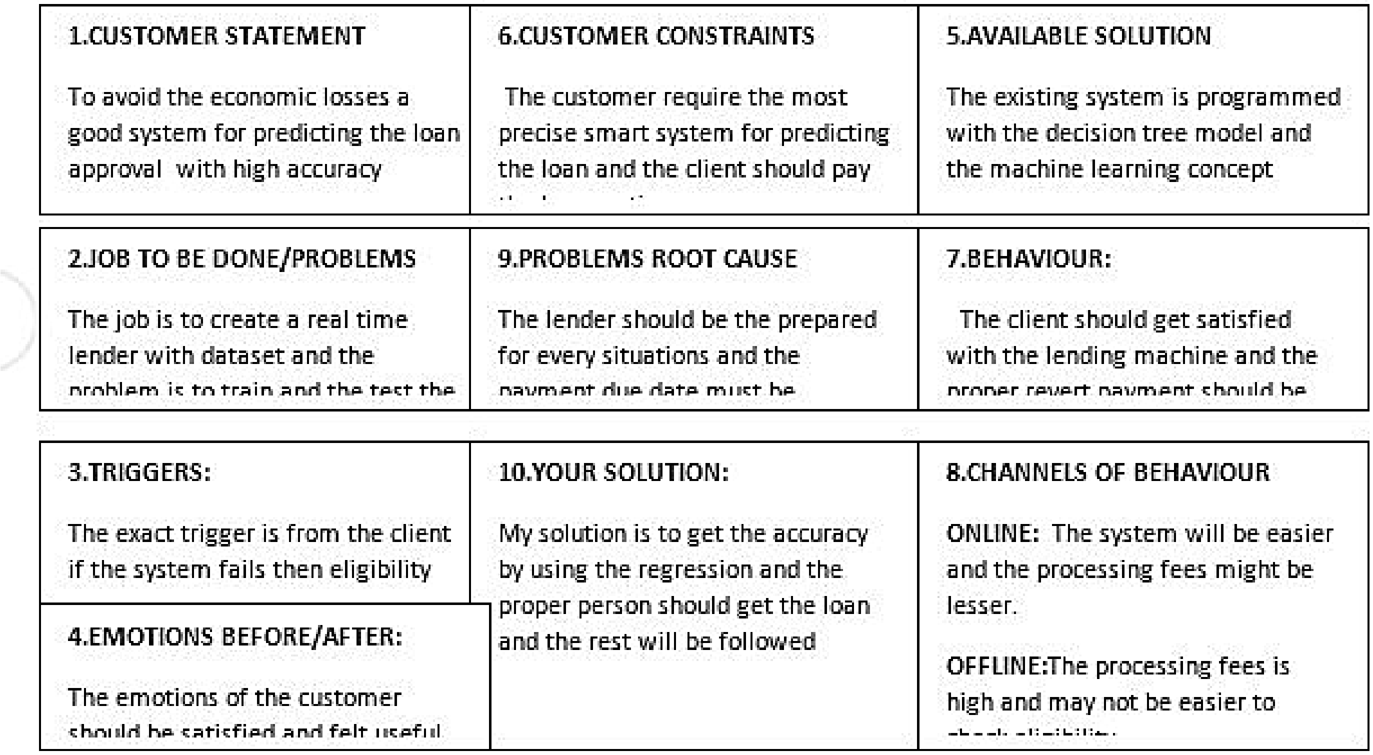
Proposed solution Template:

Project team shall fill the following information in Proposed solution Template.

S.No.	Parameter	Description
1.	Problem statement(Problem to be solved)	The dataset is to be tested&trained with more accuracy
2.	Idea/Solution description	The dataset can be coded with the decision making, logistic regression& machine learning concept
3.	Novelty/Uniqueness	The unique format is the using the logistic regression,variate analysis
4.	Social Impect/Customer satisfaction	Customer satisfy the service

5.	Business model(revenue model)	The existing system of the loan prediction using decision tree model is available
6.	Scalability of the solution	This model is 95% efficient and more accurate

3.4 PROBLEM SOLUTION FIT :



1.CUSTOMER STATEMENT

To avoid the economic losses a good system for predicting the loan approval with high accuracy

6.CUSTOMER CONSTRAINTS

The customer require the most precise smart system for predicting the loan and the client should pay

5.AVAILABLE SOLUTION

The existing system is programmed with the decision tree model and the machine learning concept

2.JOB TO BE DONE/PROBLEMS

The job is to create a real time lender with dataset and the problem is to train and the test the

9.PROBLEMS ROOT CAUSE

The lender should be the prepared for every situations and the payment due date must be

7.BEHAVIOUR:

The client should get satisfied with the lending machine and the proper revert payment should be

3.TRIGGERS:

The exact trigger is from the client if the system fails then eligibility

4.EMOTIONS BEFORE/AFTER:

The emotions of the customer should be satisfied and felt useful

10.YOUR SOLUTION:

My solution is to get the accuracy by using the regression and the proper person should get the loan and the rest will be followed

8.CHANNELS OF BEHAVIOUR

ONLINE: The system will be easier and the processing fees might be lesser.

OFFLINE:The processing fees is high and may not be easier to check eligibility

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT :

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through Internet
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Preparing the dataset	Preparing the dataset from collected information via client
FR-4	Machine learning using regression	Using the machine learning concept the regression is used
	Training the dataset	The training can be done by the train the dataset
	Testing the dataset	The dataset is tested

Non-functional Requirements:

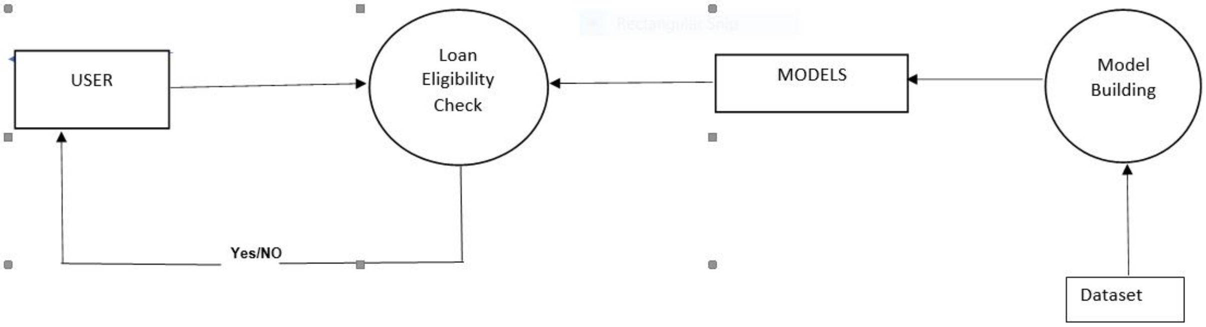
Following are the non-functional requirements of the proposed solution.

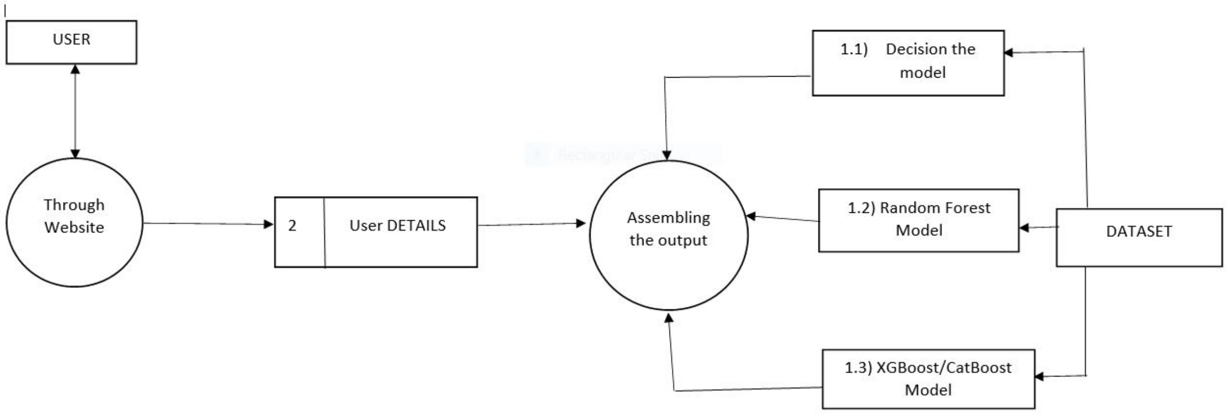
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The usability is high preferably
NFR-2	**Security**	The security information is high
NFR-3	**Reliability**	The system is reliable and more easy to access
NFR-4	**Performance**	The performance should be high
NFR-5	**Availability**	The system should be always available
NFR-6	**Scalability**	The performance should be higher

5.PRODUCT DESIGN

Data Flow Diagrams:

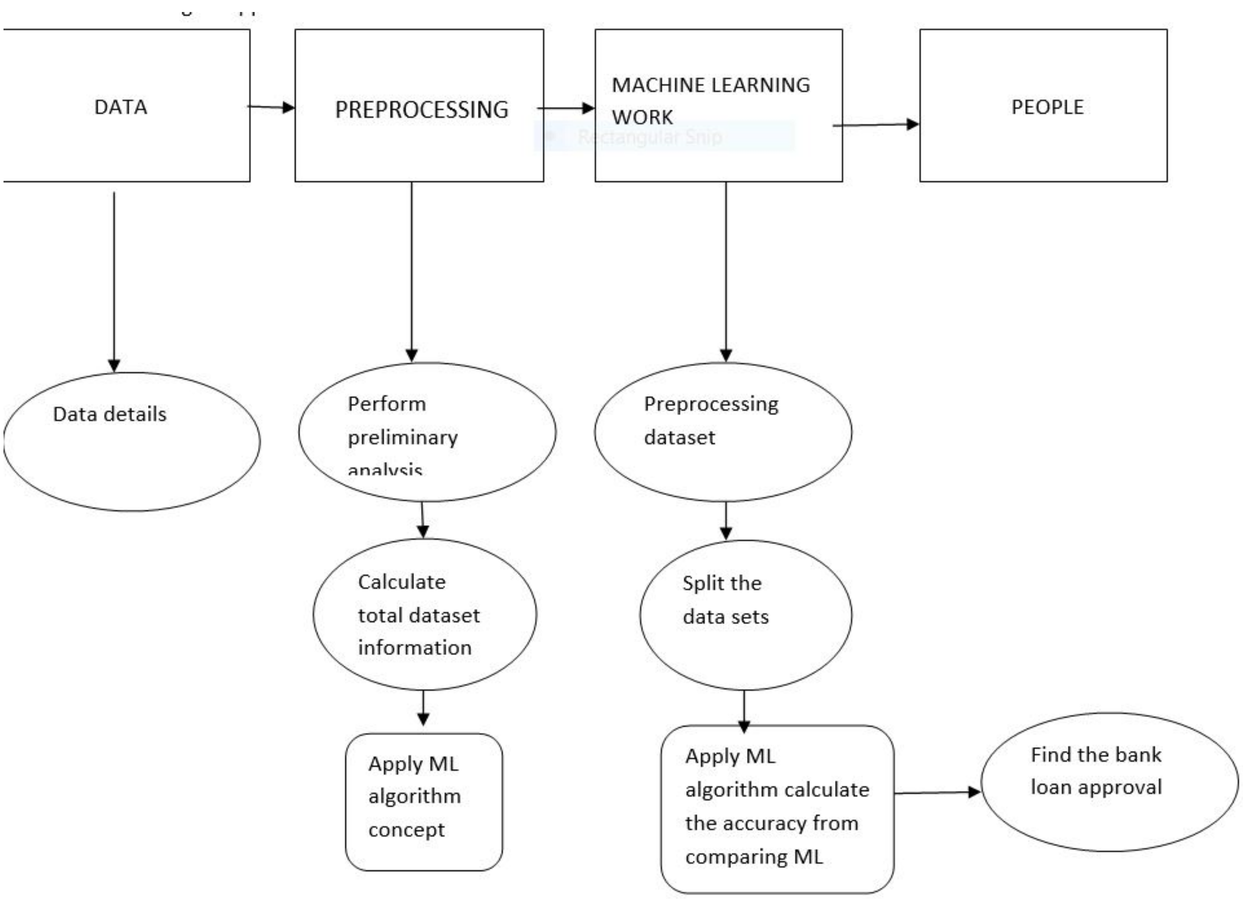
A Data Flow Diagram (DFD) is a traditional visual representatio ofthe information flows within a system. A neat and clear DFD can depict the rightamount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.





5.2SOLUTION AND TECHNICAL ARCHITECTURE:

In the smart lender project, the dataset is imported and the univariate, bivariate analysis is done and the ,logistic regression and the linear regression can be imported with the code and the libraries and these dataset is trained and tested and finally the eligibility is checked and the loan gets approved


DATA
PREPROCESSING
MACHINE LEARNING WORK
PEOPLE
Data details
Perform preliminary analysis
Preprocessing dataset
Calculate total dataset information
Split the data sets
Apply ML algorithm concept
Apply ML algorithm calculate the accuracy from comparing ML
Find the bank loan approval

6.PROJECT PLANNING AND SCHEDULING:

Project Tracker, Velocity & Burndown Chart:

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date(Actual)**
Sprint-1	20	10 Days	24 Oct 2022	04 Nov 2022	20	04 Nov 2022
Sprint-2	20	7 Days	30 Oct 2022	06 Nov 2022		
Sprint-3	20	4 Days	06 Nov 2022	10 Nov 2022		
Sprin	20	2 Days	10 Nov	12 Nov 2022		

t-4			2022			

7.CODING AND SOLUTIONING

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv("loan_prediction (1).csv
df.head(5)
obj=(df.dtypes=='object')
print("Categorical variables :",len(list(obj[obj].index)))
df.drop(['Loan_ID'],axis=1,inplace=True
obj=(df.dtypes=='object')
object_cols=list(obj[obj].index)
plt.figure(figsize=(18,36))
index=1
for col in object_cols:
    y=df[col].value_counts()
    plt.subplot(11,8,index)
    plt.xticks(rotation=0)
    sns.barplot(x=list(y.index), y=y)
    index+=1
from sklearn import preprocessing
label_encoder=preprocessing.LabelEncoder()
obj=(df.dtypes=='object')
for col in list(obj[obj].index):
  df[col]=label_encoder.fit_transform(df[col])
```

```
obj=(df.dtypes=='object')
print("Categorical variables:",len(list(obj[obj].index)))
plt.figure(figsize=(12,6))
sns.heatmap(df.corr(),cmap='BrBG',fmt='.2f',linewidths=2,annot=True)
sns.catplot(x="Gender", y="Married", hue="Loan_Status", kind="bar", data=df
for col in df.columns:
   df[col]=df[col].fillna(df[col].mean())
df.isna() .sum
from sklearn.model_selection import train_test_split
X=df.drop(['Loan_Status'],axis=1)
Y=df['Loan_Status']
X.shape,Y.shape
X_train, X_test, Y_train, Y_test=train_test_split(X,Y,test_size=0.4, random_state=1)
X_train.shape,X_test.shape,Y_train.shape,Y_test.shape
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
 sklearn.svm import SVC
from ssklearn.linear_model import LogisticRegression
from klefromarn import metrics
knn=LogisticRegreslassifier(n_neighbors=3)
rfc=mForeKNeighborsCstClassifier(n_estimators=7,criterion='entropy',random_state=7)
svc=SVC()Randolc=sion()
for clf in(rfc,,svc,lc
clf.fit(X_train, _train)
   Y_pred=clf.n,Y_train)redict(X_train)
   print(" score of", clf.__class__.__name__,"=",100*metrics.accuracy_score(Y_train,Y_pred))
for clf in(rfc,Yknpnknn,svc,lc):
   clf.fit(X_trai


    Y_pred=clf.(X_test)
    print(" score of", clf.__class__.__name__,"=",100*metrics.accuracy_score(Y_test,Y_pred))
```

HTML :

INDEX.HTML

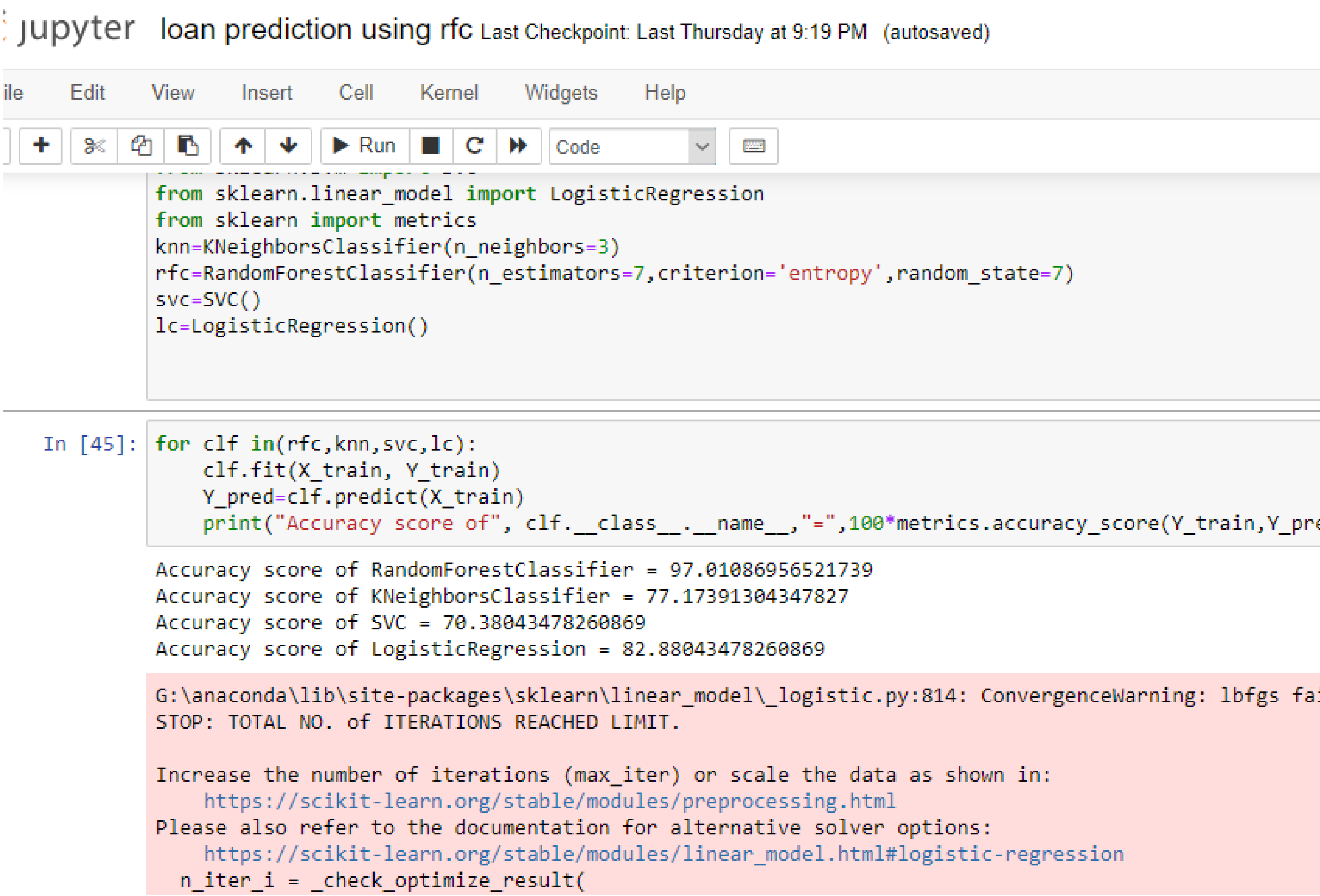
```
<!DOCTYPE  html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta
 http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="" content="width=device-width, initial-scale=1.0">
   <title>LOAN PREDICTION</title>
</head>
<body>
   <h1>LOAN PREDICTION </h1>
   <h2>  UsingRandom Forest Classifier</h2>
<form method ="POST"action="/predict">
    <p>LOAN ID</p>
    <input name="li" required>
    <p>NAME</p>
    <input name="na" required>
    <p>GENDER</p>
    <input name="ge" required>
    <p>MARRIED</p>
    <input name="md" required>
    <p>DEPENDENTS</p>
    <input name="de" required>
    <p>EDUCATION</p>
    <input name="ed" required>
    <p>SELF EMPLOYED</p>
    <input name="se" required>
    <p>APPLICANT INCOME</p>
    <input name="ai" required>
    <p>CO APPLICANT INCOME</p>
    <input name="ca" required>
    <p>LOAN AMOUNT </p>
    <input name="la" required>
    <p>LOAN AMOUNT TERM</p>
    <input name="lat" required>
    <p>CREDIT HISTORY</p>
    <input name="ch" required>
    <p>PROPERTY AREA</p>
    <input name="pa" required>
    <p>LOAN STATUS</p>
    <input name=""ls" required>
</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 PREDICTION</title>
</head>
<body>
   <h1>LOAN PREDICTION </h1>
   <h2>Using Random Forest Classifier</h2>
<form method="POST"action="/predict">
    <p>LOAN ID</p>
    <input name="li" required>
    <p>NAME</p>
    <input name="na" required>
    <p>GENDER</p>
    <input name="ge" required>
    <p>MARRIED</p>
    <input name="md" required>
    <p>DEPENDENTS</p>
    <input name="de" required>
    <p>EDUCATION</p>
    <input name="ed" required>
    <p>SELF EMPLOYED</p>
    <input name="se" required>
    <p>APPLICANT INCOME</p>
    <input name="ai" required>
    <p>CO APPLICANT INCOME</p>
    <input name="ca" required>
    <p>LOAN AMOUNT </p>
    <input name="la" required>
    <p>LOAN AMOUNT TERM</p>
    <input name="lat" required>
    <p>CREDIT HISTORY</p>
    <input name="ch" required>
    <p>PROPERTY AREA</p>
    <input name="pa" required>
    <p>LOAN STATUS</p>
    <input name=""ls" required>
</body>
</html>
```

8.TESTING

8.1 TEST CASES:

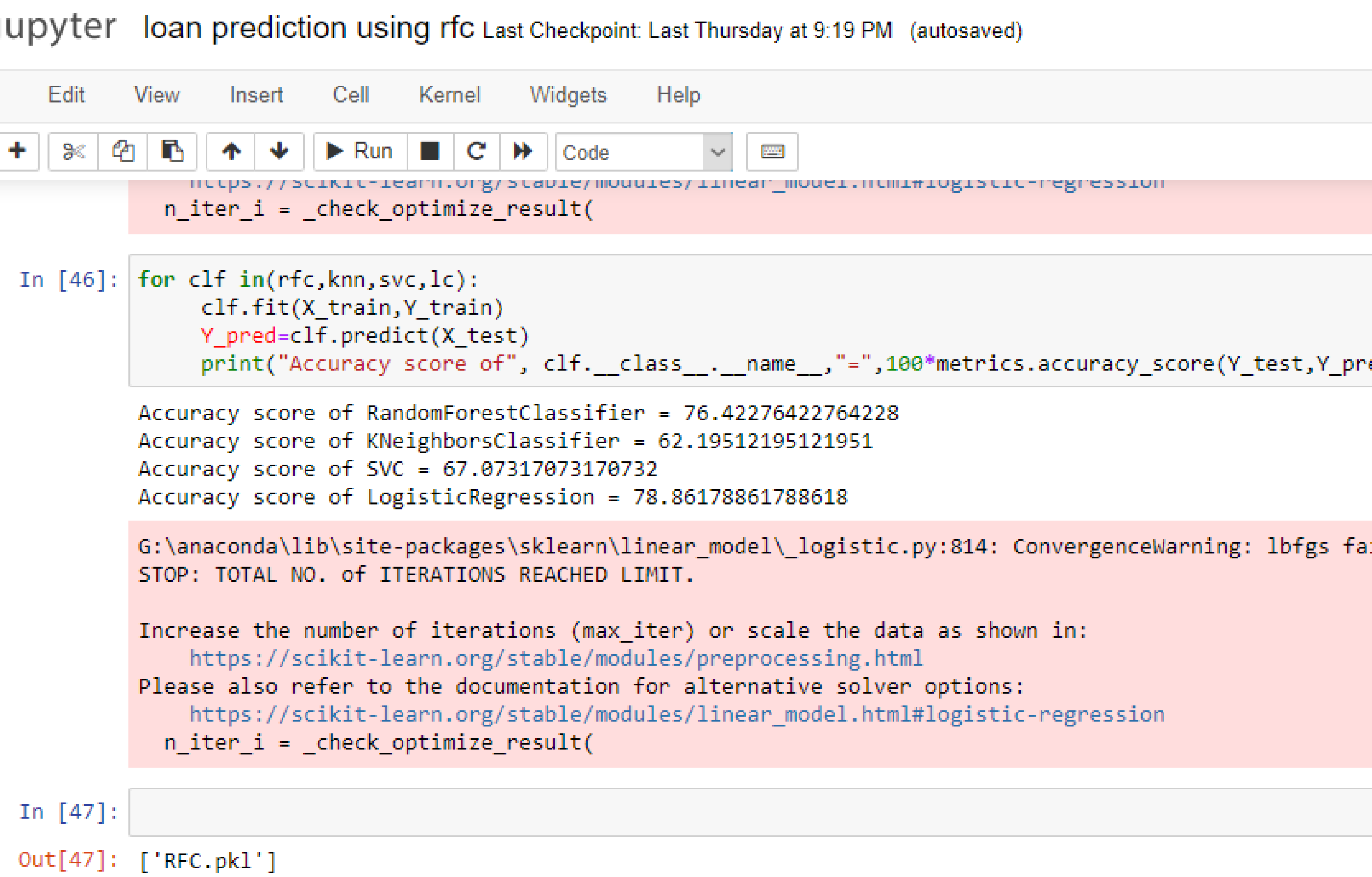

jupyter
loan prediction using rfc Last Checkpoint: Last Thursday at 9:19 PM (autosaved)
ile
Edit
View
Insert
Cell
Kernel
Widgets
Help
Run
Code
In [45]:

```
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
knn=KNeighborsClassifier(n_neighbors=3)
rfc=RandomForestClassifier(n_estimators=7,criterion='entropy',random_state=7)
svc=SVC()
lc=LogisticRegression()
```

```
for clf in(rfc,knn,svc,lc):
    clf.fit(X_train, Y_train)
    Y_pred=clf.predict(X_train)
    print("Accuracy score of", clf.__class__.__name__,"=",100*metrics.accuracy_score(Y_train,Y_pre
```

```
Accuracy score of RandomForestClassifier = 97.01086956521739
Accuracy score of KNeighborsClassifier = 77.17391304347827
Accuracy score of SVC = 70.38043478260869
Accuracy score of LogisticRegression = 82.88043478260869
```

```
G:\anaconda\lib\site-packages\sklearn\linear_model\_logistic.py:814: ConvergenceWarning: lbfgs fai
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
```


upyter loan prediction using rfc Last Checkpoint: Last Thursday at 9:19 PM (autosaved)
Edit
View
Insert
Cell
Kernel
Widgets
Help
Run
Code

```
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
```

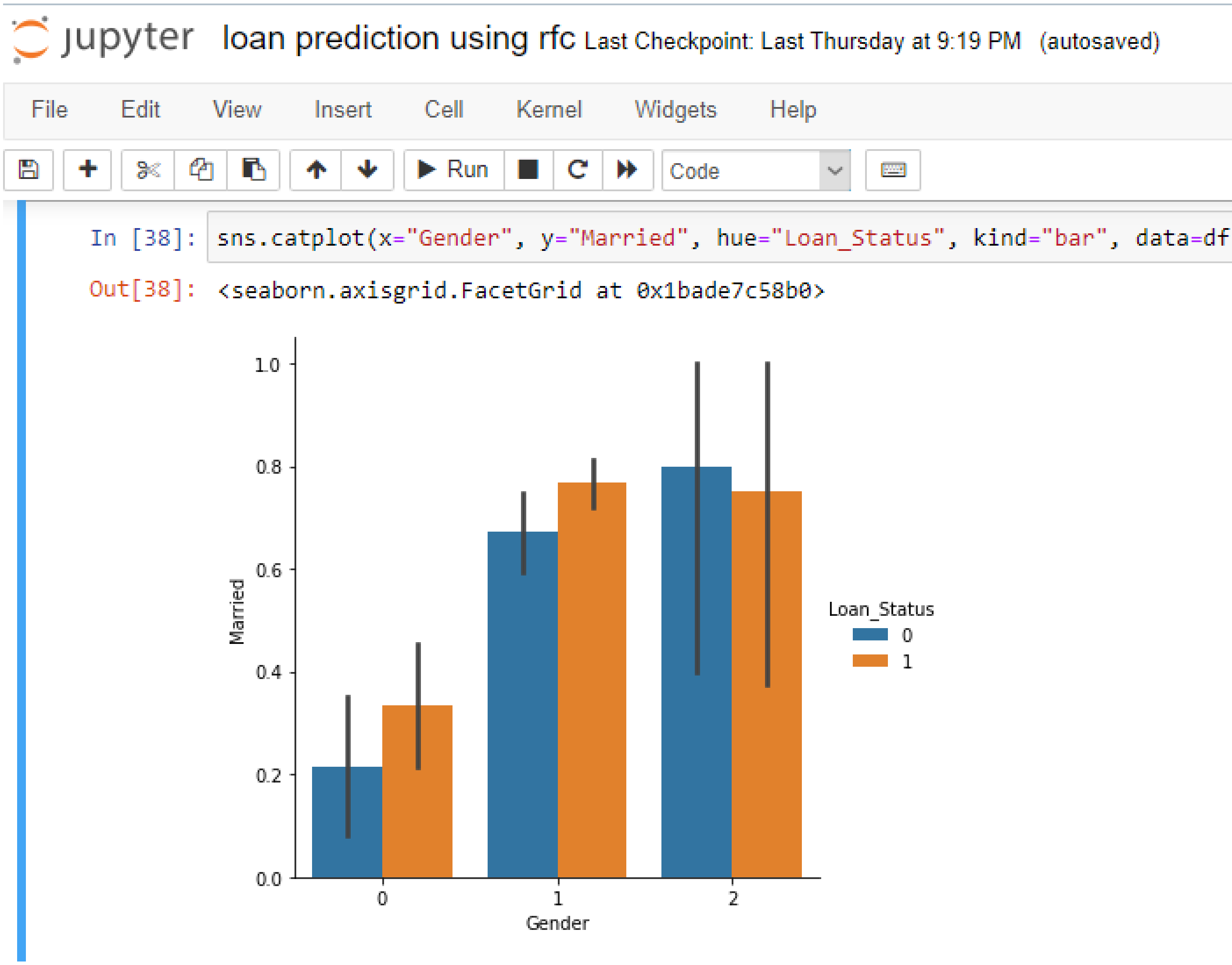
In [46]:
```
for clf in(rfc,knn,svc,lc):
     clf.fit(X_train,Y_train)
     Y_pred=clf.predict(X_test)
     print("Accuracy score of", clf.__class__.__name__,"=",100*metrics.accuracy_score(Y_test,Y_pre
```

```
Accuracy score of RandomForestClassifier = 76.42276422764228
Accuracy score of KNeighborsClassifier = 62.19512195121951
Accuracy score of SVC = 67.07317073170732
Accuracy score of LogisticRegression = 78.86178861788618
```

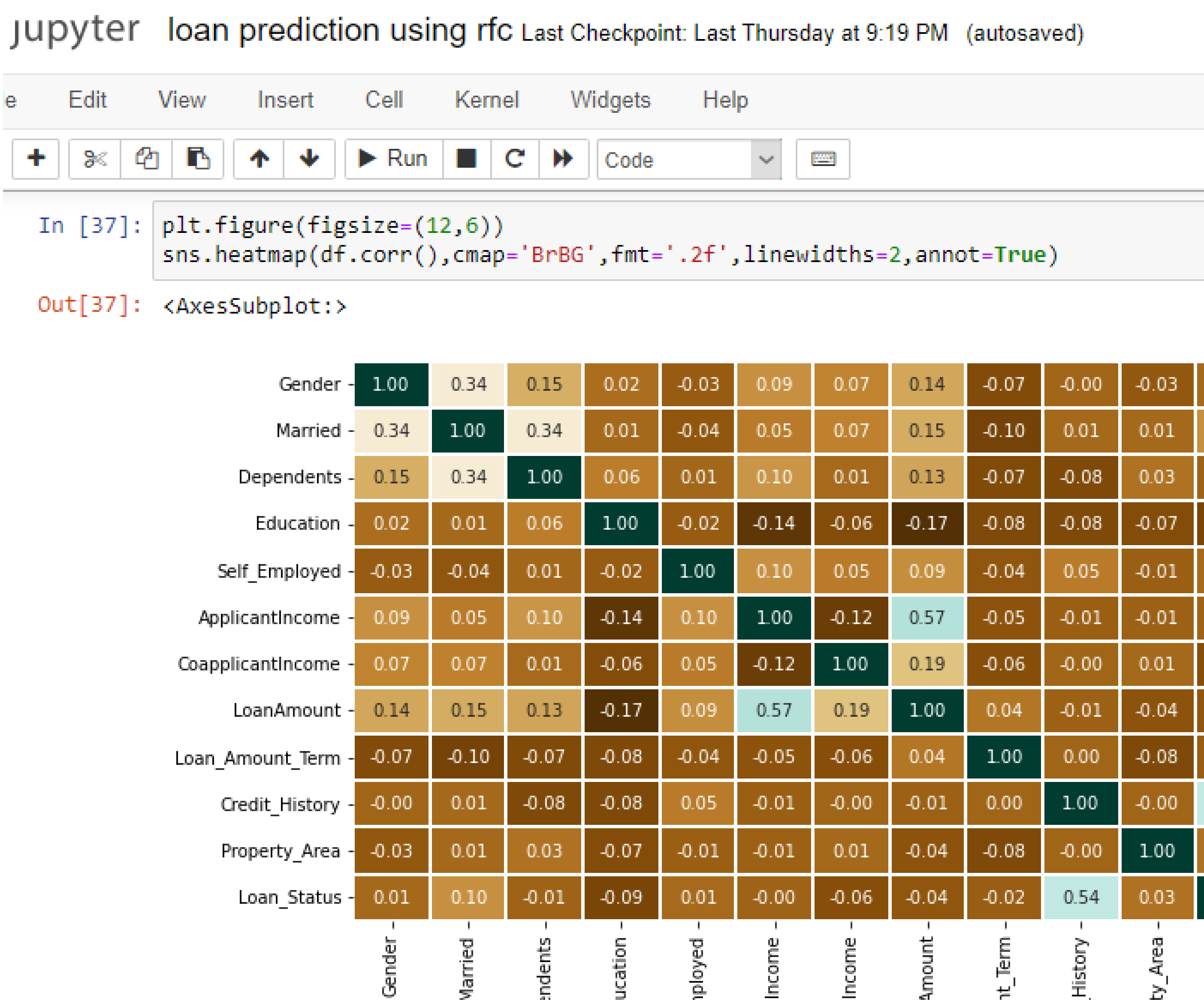
```
G:\anaconda\lib\site-packages\sklearn\linear_model\_logistic.py:814: ConvergenceWarning: lbfgs fa
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
```

In [47]:

Out[47]:
```
['RFC.pkl']
```



```
In [38]: sns.catplot(x="Gender", y="Married", hue="Loan_Status", kind="bar", data=df
Out[38]: <seaborn.axisgrid.FacetGrid at 0x1bade7c58b0>
```

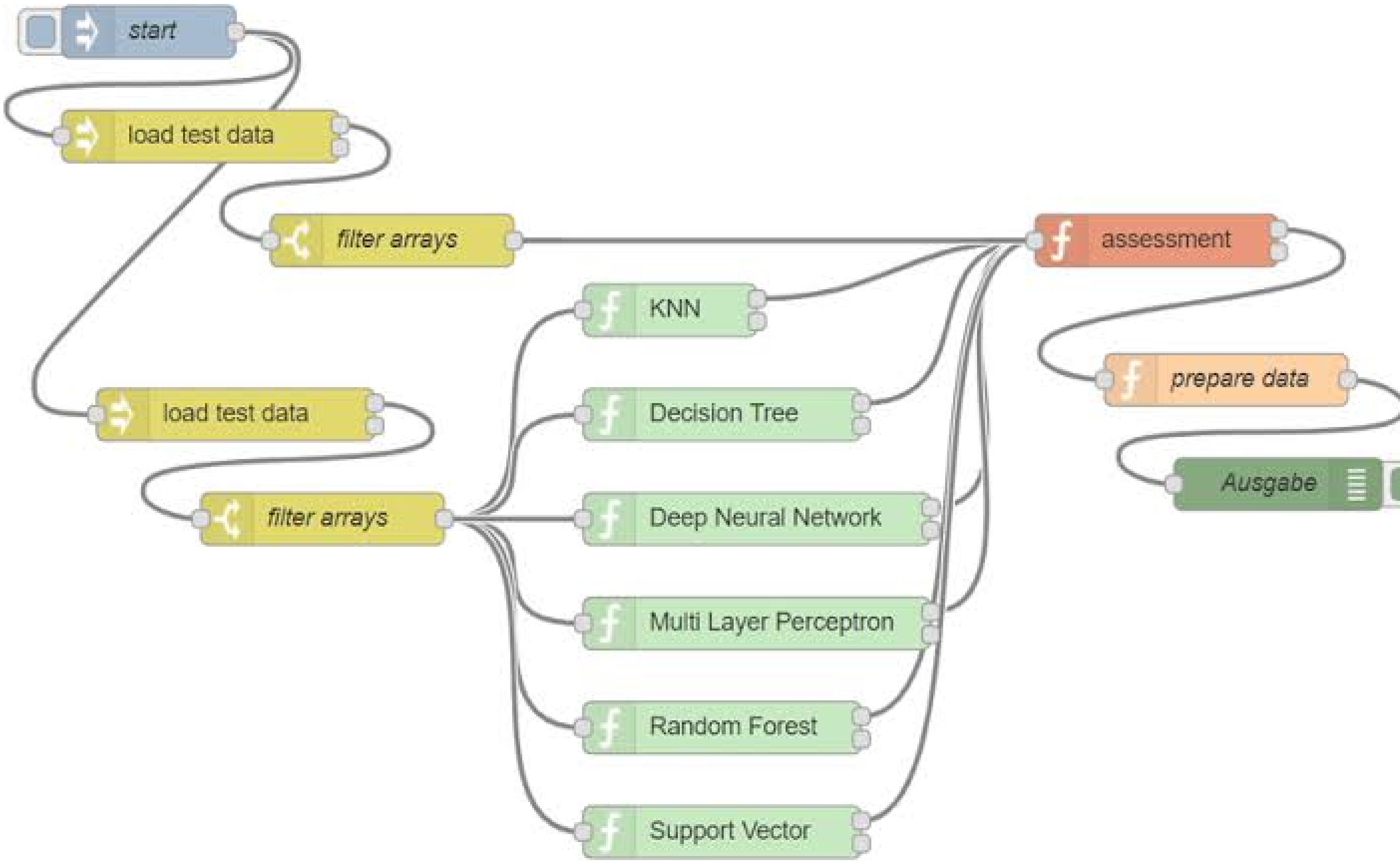


loan prediction using rfc Last Checkpoint: Last Thursday at 9:19 PM (autosaved)

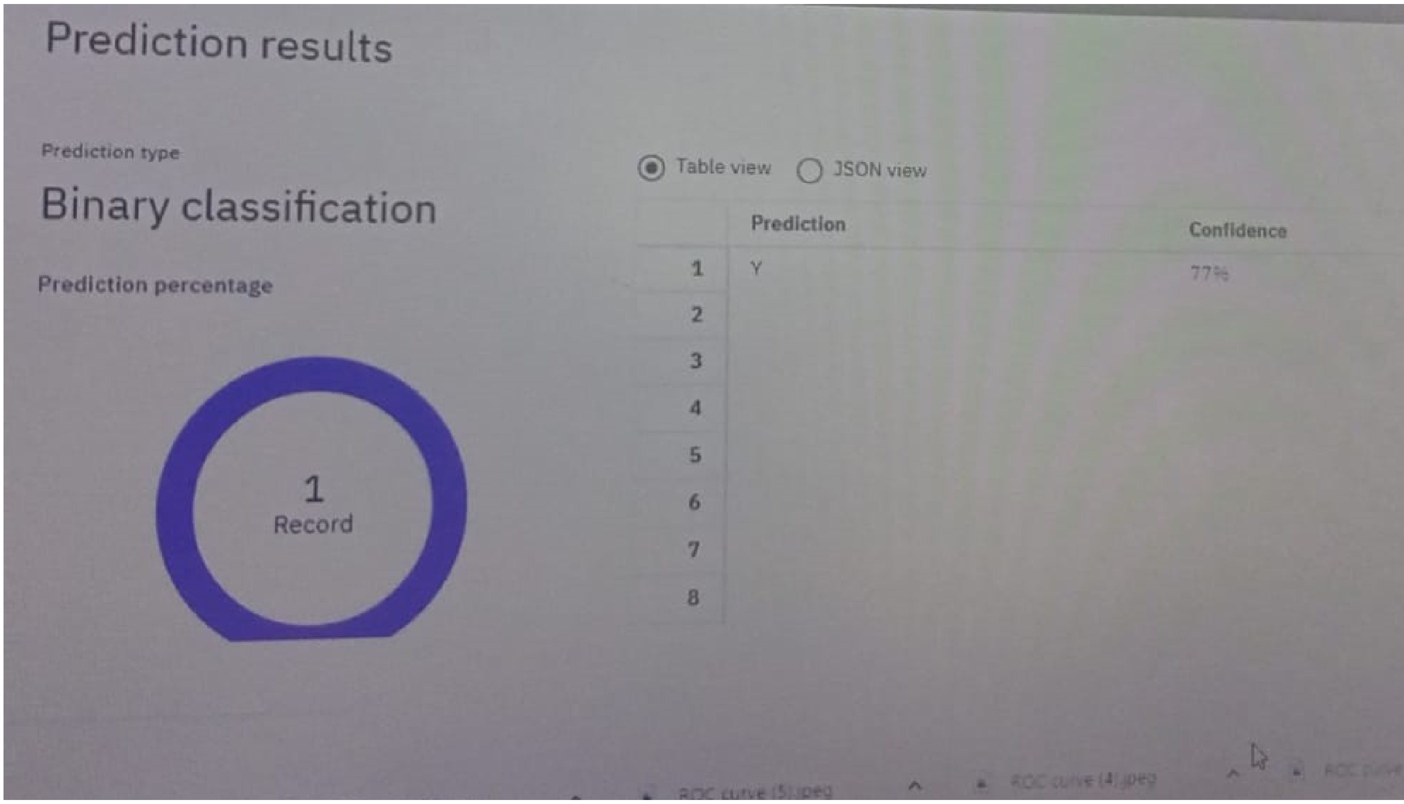
```
In [37]: plt.figure(figsize=(12,6))
         sns.heatmap(df.corr(),cmap='BrBG',fmt='.2f',linewidths=2,annot=True)

Out[37]: <AxesSubplot:>
```

	Gender	Married	ndents	ucation	ployed	Income	Income	Amount	nt_Term	History	ty_Area
Gender	1.00	0.34	0.15	0.02	-0.03	0.09	0.07	0.14	-0.07	-0.00	-0.03
Married	0.34	1.00	0.34	0.01	-0.04	0.05	0.07	0.15	-0.10	0.01	0.01
Dependents	0.15	0.34	1.00	0.06	0.01	0.10	0.01	0.13	-0.07	-0.08	0.03
Education	0.02	0.01	0.06	1.00	-0.02	-0.14	-0.06	-0.17	-0.08	-0.08	-0.07
Self_Employed	-0.03	-0.04	0.01	-0.02	1.00	0.10	0.05	0.09	-0.04	0.05	-0.01
ApplicantIncome	0.09	0.05	0.10	-0.14	0.10	1.00	-0.12	0.57	-0.05	-0.01	-0.01
CoapplicantIncome	0.07	0.07	0.01	-0.06	0.05	-0.12	1.00	0.19	-0.06	-0.00	0.01
LoanAmount	0.14	0.15	0.13	-0.17	0.09	0.57	0.19	1.00	0.04	-0.01	-0.04
Loan_Amount_Term	-0.07	-0.10	-0.07	-0.08	-0.04	-0.05	-0.06	0.04	1.00	0.00	-0.08
Credit_History	-0.00	0.01	-0.08	-0.08	0.05	-0.01	-0.00	-0.01	0.00	1.00	-0.00
Property_Area	-0.03	0.01	0.03	-0.07	-0.01	-0.01	0.01	-0.04	-0.08	-0.00	1.00
Loan_Status	0.01	0.10	-0.01	-0.09	0.01	-0.00	-0.06	-0.04	-0.02	0.54	0.03

NODERED:



8.2 USER ACCEPTANCE TESTING:



Prediction results

Prediction type

Binary classification

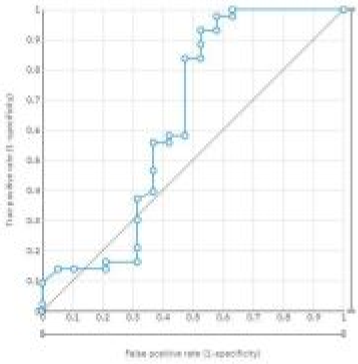
Prediction percentage

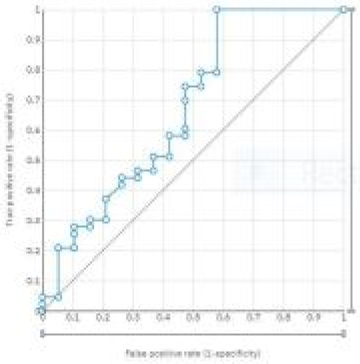
● Table view ○ JSON view

	Prediction	Confidence
1	Y	77%
2		
3		
4		
5		
6		
7		
8		

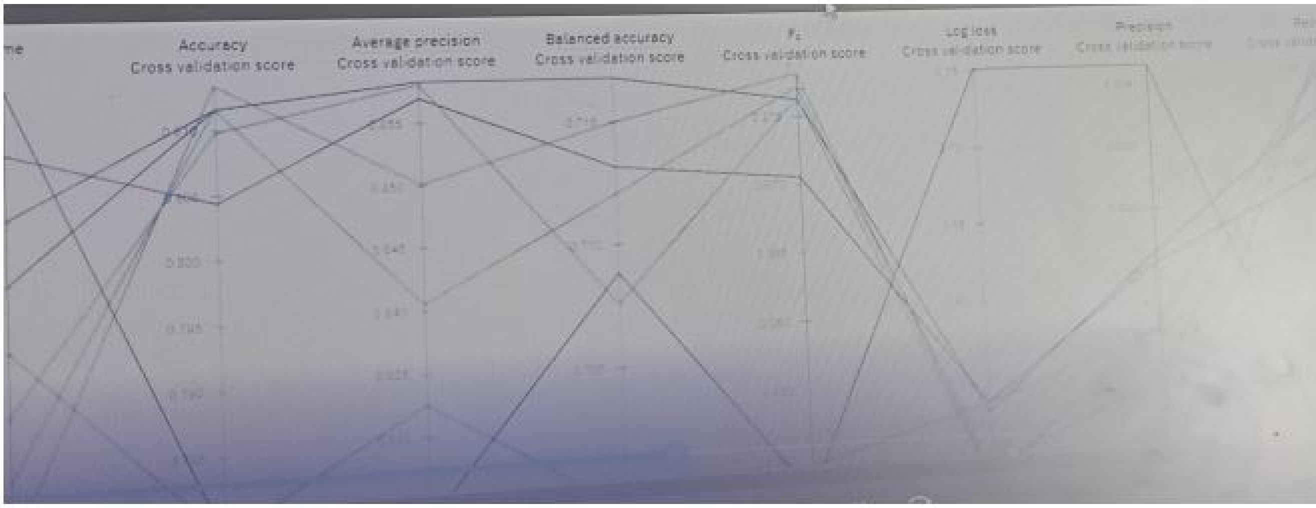
9.RESULT:

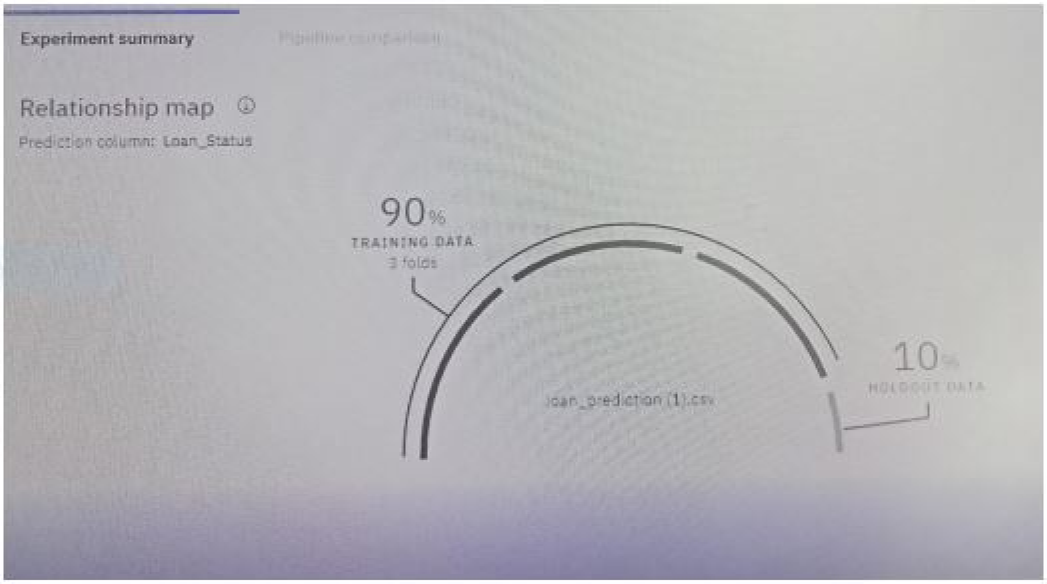
9.1PERFORMANCE METRICS:



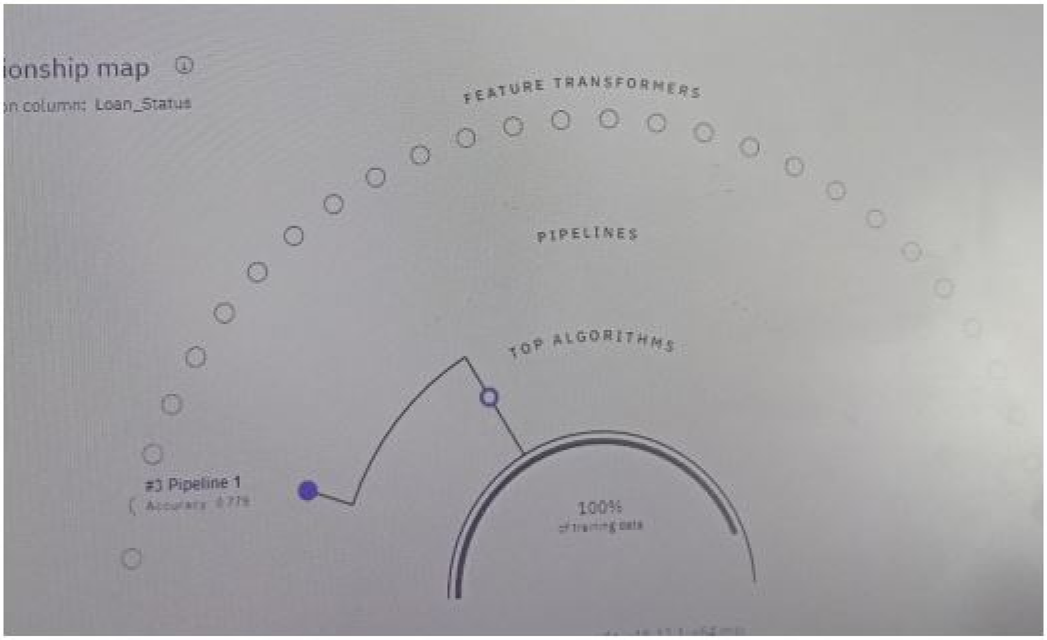


Rectangular Snip





Experiment summary
Pipeline comparison
Relationship map
Prediction column: Loan_Status
90%
TRAINING DATA
3 folds
loan_prediction (1).csv
10%
HOLDOUT DATA



ionship map
on column: Loan_Status
FEATURE TRANSFORMERS
PIPELINES
TOP ALGORITHMS
#3 Pipeline 1
Accuracy 0.779
100%
of training data

10.ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

- Simple and user friendly interface,which is comfortable for naive user like bank employee to of applied customer.
- logistic regression give the accurate result of the prediction up to 83% which is the algorithm
- it is composed using the JSON and python for the web usage in real time.
- as per the data available,the models give prediction in real time.
- no need for feature normalization or scaling.
- easy to measure the relative importance of each feature.can handle categorical and numeric

DISADVANTAGES :

- Hacking is possible
- economic losses may be occurs
- Honesty sometimes fails

11.CONCLUTION

From a proper analysis of positive points and constraints on the member, it can be safely concluded that the product is a considerably productive member. This use is working duly a meeting to all Banker requisites. This member can be freely plugged in numerous other syste There have been mathematics cases of computer glitches, violations in content and most impor weight of features is fixed in automated prophecy system, so in the near future the so – ca software could be made more secure, trustworthy and dynamic weight conformation. In near future this module of prophecy can be integrated with the module of automated processing system. T system is trained on old training dataset in future software can be made resembling that new testin date should also take part in training data after some fix time.

12.FUTURE SCOPE

The purpose of our research review is to automate the banking process of selecting the loan applica for their bank or financial institution. In future, we can develop full-fledged early warning systems ba Learning and that will help a bank or any other financial institution to reduce their losses and increa also trying to reduce the time required to do the predictions so the users can get results in real time productivity of the users.

13.APPENDIX

SOURCE CODE:

```
import flask
from flask import
request,render_template
from flask_cors import cors
Import joblib
app=flask.Flak(_name_,static_url_path='')
```

```
CORS(app)
@app.route('/',methods=['GET'])
def sendHomePage():
    return render_template(index.html)
    @app.route('predict',methods=['POST'])
    def predictSpecies():
        li= float(request.form['li'])
        na= float(request.form['na'])
        ge= float(request.form['ge'])
        ma= float(request.form['ma'])
        de= float(request.form['de'])
        ed= float(request.form['ed'])
        se= float(request.form['se'])
        ai= float(request.form['ai'])
        ca= float(request.form['ca'])
        la= float(request.form['la'])
        lat= float(request.form['lat'])
        ch= float(request.form['ls'])
        X=[[li,na,ge,ma,de,ed,se,ai,ca,la,lat,ch,pa,ls]]
        cfl=joblib.load(RFC.pkl)
        species=clf.predict(X)[0]
        return render_template('predict.html',predict=species)
 if _name=='main_':
        app.run()
```

GITHUB LINK :

https://github.com/IBM-EPBL/IBM-Project-9529-1659016608

PROJECT DEMO LINK :

https://drive.google.com/folderview?id=1ml64FUT6ipa0RsH4--edtFz_NrdPY0J2