DONE BY : SUNKARI LAKSHMIPRIYA
COLLEGE : GITAM UNIVERSITY

Loan Status Prediction using IBM Watson Machine Learning

Decision Tree Regression Algorithm

1.Introduction:

Customer segmentation and profitability, high-risk loan applicants, predicting payment default, marketing, credit analysis, ranking investments, fraudulent transactions, optimising stock portfolios, cash management and forecasting operations, most profitable Credit Card Customers, and cross-selling are some of the areas where Machine Learning can be used in the financial sector. When it comes to borrowing money, there are many different sorts of loans to choose, and it's vital to be aware of your alternatives. The practice of evaluating loan collections and assigning loans to groups or grades based on perceived danger and other related loan features is known as loan classification. The process of continuous assessment and classification of loans allows you to keep track of the credit quality of your loan portfolios and take action to prevent them from deteriorating. Banks are required to utilise more difficult internal classification schemes than the more conventional schemes that bank managers use for reporting purposes and are designed to facilitate observation and inter bank evaluation.

There are numerous sorts of loans, including: Open-ended loans are those in which you can take out a loan for an indefinite period of time. The most well-known open-ended loans are credit cards and lines of credit. With both of these forms of loans, you have a credit limit that you can purchase. Your available credit will decrease at any time you can make a transaction.As you spend, your cash on hand grows, allowing you to use the credit card more frequently. Closed-ended loans are those that can't be taken out again once they've been repaid. When you use closed-ended loans to make purchases, the loan balance decreases. You do not, however, have any current credit that you can use for closed-ended loans.

If you want to lend more money, you have the option of applying for another loan. Auto loans, mortgage loans, and student loans are all examples of closed-ended loans. For prediction and description, there are two primary objectives. Prediction is the process of using some variables from a data collection to forecast unknown values for other variables. The goal of description is to uncover patterns in data that can be interpreted by humans. The generated knowledge must be novel, not obvious, relevant, and applicable to the field in which it was attained. It's also the process of extracting valuable data from unstructured data.

Machine Learning approaches assist in identifying borrowers who pay back loans on time and those who do not. It also aids in predicting when a borrower will default and whether or not giving a loan to a specific consumer would result in bad loans.

1.2 Purpose:

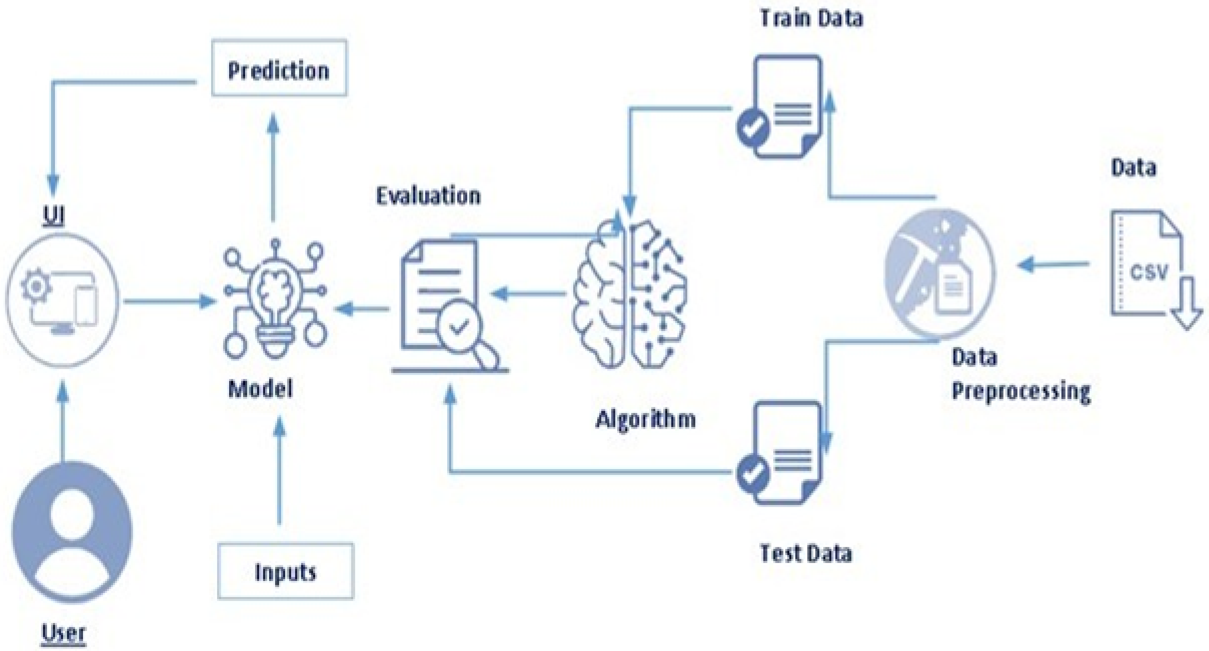
The project's goal is to extract the libraries for machine learning for loan prediction using Python's pandas, matplotlib, and seaborn libraries. Second, for the Logistic Regression machine learning algorithm, learn how to hyper tune the parameters using grid search cross validation.

Finally, utilising voting ensemble techniques of pooling predictions from many machine learning algorithms and withdrawing conclusions, predict whether the loan applicant can repeat the loan or not.

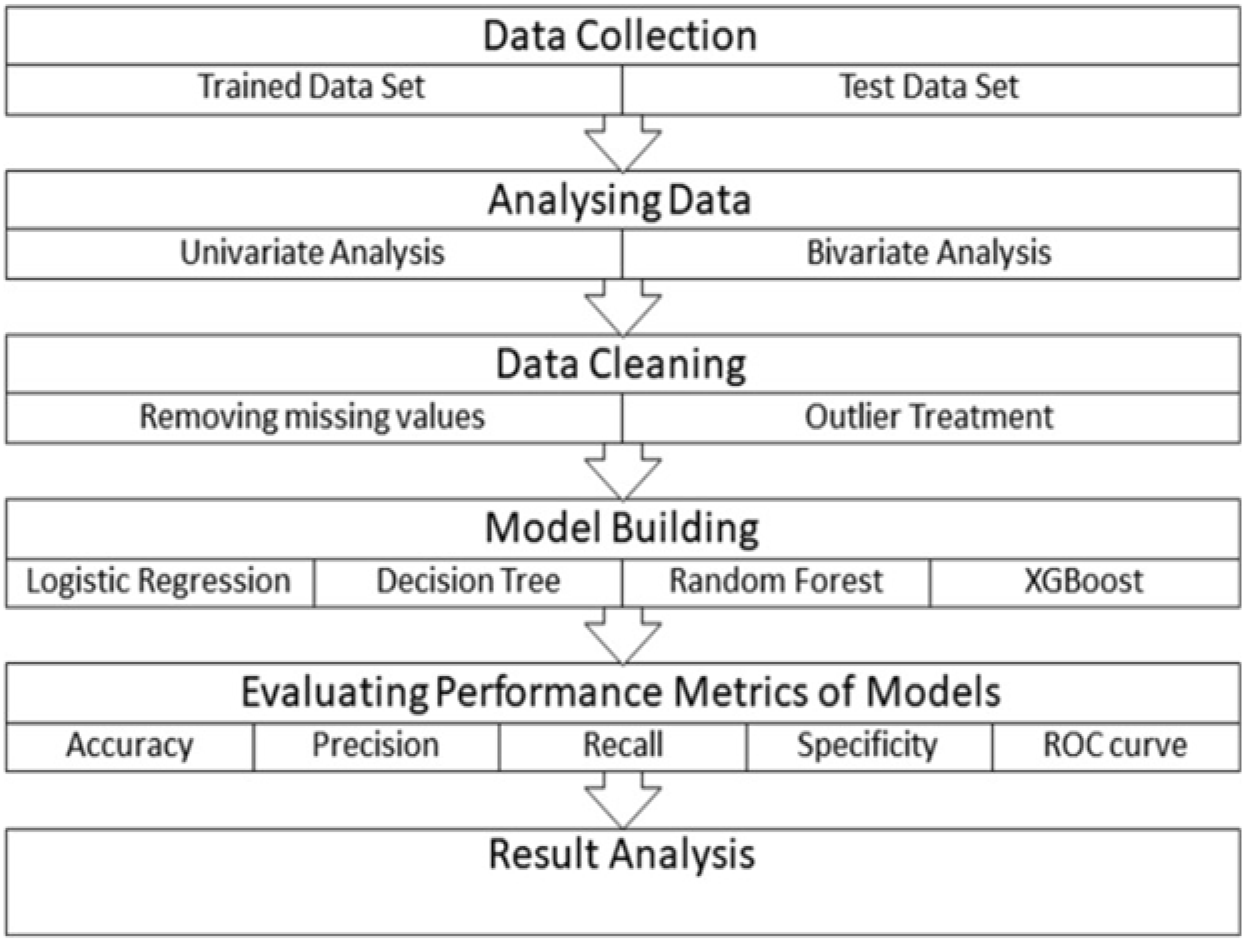
2.Literature Review:

The practise of evaluating data from many viewpoints and extracting meaningful knowledge from it is known as data mining. It is at the heart of the process of knowledge discovery. As indicated, the many procedures involved in extracting knowledge from raw data. Classification, clustering, association rule mining, prediction and sequential patterns, neural networks, regression, and other data mining techniques are examples. The most widely used data mining technique is classification, which uses a group of pre-classified samples to create a model that can categorise the entire population of information. The categorization technique is particularly well suited to fraud detection and credit risk applications. The most widely used data mining technique is classification, which uses a set of pre-classified examples to create a model that can classify a huge number of records. The categorization technique is especially well suited for fraud detection and credit risk applications.

3.Architecture:



4.Methodology:



Software Designing:

- Jupyter Notebook Environment
- Machine Learning Algorithms
- Python (pandas, numpy, matplotlib, seaborn, sklearn)
- HTML
- Flask

The models are implemented using Python 3.7 with listed libraries:

Pandas

Pandas is a Python package to work with structured and time series data. The data from various file formats such as csv, json, SQL etc can be imported using Pandas. It is a powerful open-source tool used for data analysis and data manipulation operations such as data cleaning, merging, selecting as well wrangling.

Seaborn

Seaborn is a python library for building graphs to visualize data. It provides integration with pandas. This open-source tool helps in defining the data by mapping the data on the informative and interactive plots. Each element of the plots gives meaningful information about the data.

Sklearn

This python library is helpful for building machine learning and statistical models such as clustering, classification, regression etc. Though it can be used for reading, manipulating and summarizing the data as well, better libraries are there to perform these functions.

Importing The Libraries:

```
import pandas as pd
import numpy as np
from collections import Counter as c
import matplotlib.pyplot as plt
from sklearn import preprocessing
import seaborn as sns
from sklearn.model_selection import train_test_split
```

Loading the dataset:

The machine learning model is trained using the training data set. Every new applicant details filled at the time of application form acts as a test data set. On the basis of the training data sets, the model will predict whether a loan would be approved or not. Train file will be used for training the model, i.e. our model will learn from this file. It contains all the independent variables and the target variable.

For this problem, we have three CSV files: Credit_train.csv file.

```
dataset = pd.read_csv('credit_train.csv')
dataset.head()
```

Datainformation:

```
data.shape
```

```
(100514, 19)
```

```
type(data)
```

Handling Null Values:

```
 data.isnull().any()
```

```
Loan ID                          True
Customer ID                       True
Loan Status                       True
Current Loan Amount                True
Term                             True
Credit Score                      True
Annual Income                      True
Years in current job               True
Home Ownership                      True
Purpose                            True
Monthly Debt                        True
```

```
Years of Credit History              True
Months since last delinquent        True
Number of Open Accounts             True
Number of Credit Problems          True
Current Credit Balance             True
Maximum Open Credit                True
Bankruptcies                       True
Tax Liens                          True
dtype: bool
```

After removing the null values:

```
data.isnull().any()
```

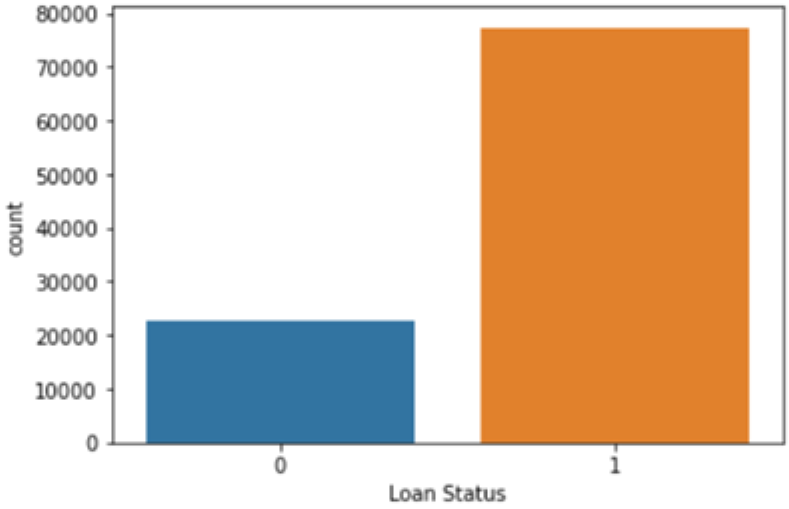
```
Loan ID                            False
Customer ID                        False
Loan Status                        False
Current Loan Amount                 False
Term                               False
Credit Score                       False
Annual Income                       False
Years in current job               False
Home Ownership                      False
Purpose                            False
Monthly Debt                        False
Years  of  Credit  History    False
Months   since   last   delinquent
False Number of Open Accounts
False      Number      of      Credit
Problems   False   Current   Credit
Balance
```

```
False
Maximum Open Credit              False
Bankruptcies                    False
Tax Liens                       False
dtype: bool
```

Data Visualization:

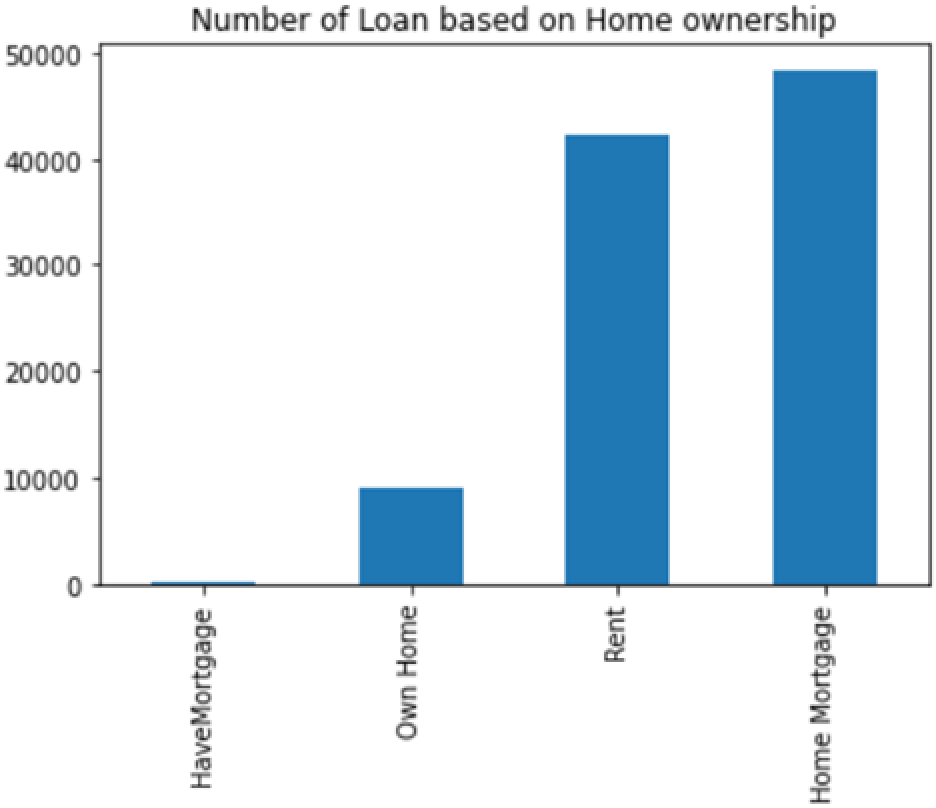
Status of the loan:

```
<AxesSubplot:title={'center':'Status of the Loan'}>
sns.countplot(data['Loan Status'])
```



```
sns.countplot(data['Home Ownership'])
```

```
<AxesSubplot:xlabel='Home Ownership', ylabel='count'>
```



Categorical Independent Variable vs Target Variable:

First, we'll figure out how the target variable and categorical independent variables are related. Now let's take a look at the stacked bar plot, which shows the percentage of granted and unapproved loans.

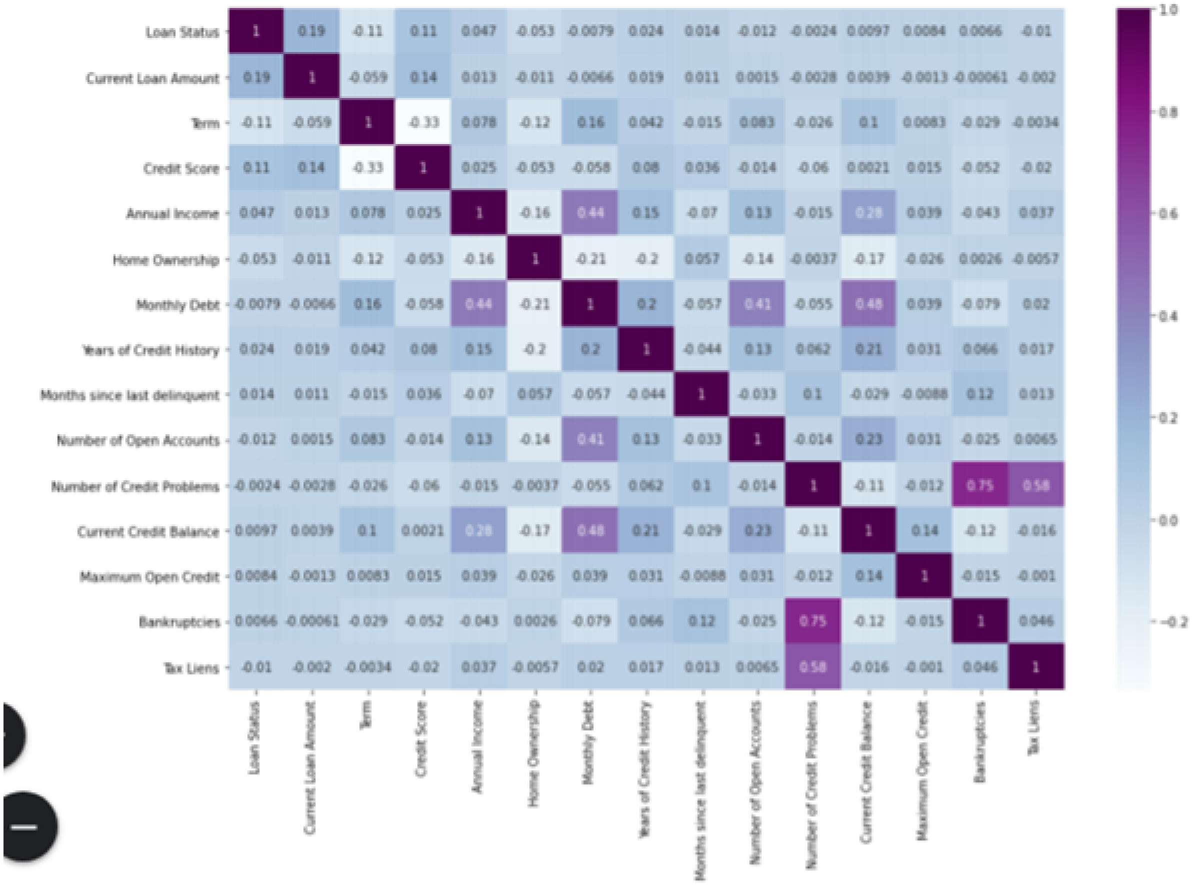
It can be assumed that the proportion of Fully Paid and Charged Off applicants for approved and unapproved loans is around the same.

Let's see how the other categorical variables compare to the target variable.
People with a credit score of 1 appear to have a higher chance of getting their loans authorized.

In comparison to rural and urban areas, the proportion of loans authorized in semi-urban areas is higher.
Let's look at numerical independent variables in relation to the target variable now

numerical variable. We will also convert the target variable’s categories into 0 and 1 so that we can find its correlation with numerical variables. One more reason to do so is few models like logistic regression takes only numeric values as input. We will replace N with 0 and Ywith 1.

Let's have a look at the relationship between all of the numerical variables now. The correlation will be visualised using a heat map. Heatmaps use colour variations to illustrate data. The factors with a darker colour have a higher correlation



	Loan Status	Current Loan Amount	Term	Credit Score	Annual Income	Home Ownership	Monthly Debt	Years of Credit History	Months since last delinquent	Number of Open Accounts	Number of Credit Problems	Current Credit Balance	Maximum Open Credit	Bankruptcies	Tax Liens
Loan Status	1	0.19	-0.11	0.11	0.047	-0.053	-0.0079	0.024	0.014	-0.012	-0.0024	0.0097	0.0084	0.0066	-0.01
Current Loan Amount	0.19	1	-0.059	0.14	0.013	-0.011	-0.0066	0.019	0.011	0.0015	-0.0028	0.0039	-0.0013	-0.00061	-0.002
Term	-0.11	-0.059	1	-0.33	0.078	-0.12	0.16	0.042	-0.015	0.083	-0.026	0.1	0.0083	-0.029	-0.0034
Credit Score	0.11	0.14	-0.33	1	0.025	-0.053	-0.058	0.08	0.036	-0.014	-0.06	0.0021	0.015	-0.052	-0.02
Annual Income	0.047	0.013	0.078	0.025	1	-0.16	0.44	0.15	-0.07	0.13	-0.015	0.28	0.039	-0.043	0.037
Home Ownership	-0.053	-0.011	-0.12	-0.053	-0.16	1	-0.21	-0.2	0.057	-0.14	-0.0037	-0.17	-0.026	0.0026	-0.0057
Monthly Debt	-0.0079	-0.0066	0.16	-0.058	0.44	-0.21	1	0.2	-0.057	0.41	-0.055	0.48	0.039	-0.079	0.02
Years of Credit History	0.024	0.019	0.042	0.08	0.15	-0.2	0.2	1	-0.044	0.13	0.062	0.21	0.031	0.066	0.017
Months since last delinquent	0.014	0.011	-0.015	0.036	-0.07	0.057	-0.057	-0.044	1	-0.033	0.1	-0.029	-0.0088	0.12	0.013
Number of Open Accounts	-0.012	0.0015	0.083	-0.014	0.13	-0.14	0.41	0.13	-0.033	1	-0.014	0.23	0.031	-0.025	0.0065
Number of Credit Problems	-0.0024	-0.0028	-0.026	-0.06	-0.015	-0.0037	-0.055	0.062	0.1	-0.014	1	-0.11	-0.012	0.75	0.58
Current Credit Balance	0.0097	0.0039	0.1	0.0021	0.28	-0.17	0.48	0.21	-0.029	0.23	-0.11	1	0.14	-0.12	-0.016
Maximum Open Credit	0.0084	-0.0013	0.0083	0.015	0.039	-0.026	0.039	0.031	-0.0088	0.031	-0.012	0.14	1	-0.015	-0.001
Bankruptcies	0.0066	-0.00061	-0.029	-0.052	-0.043	0.0026	-0.079	0.066	0.12	-0.025	0.75	-0.12	-0.015	1	0.046
Tax Liens	-0.01	-0.002	-0.0034	-0.02	0.037	-0.0057	0.02	0.017	0.013	0.0065	0.58	-0.016	-0.001	0.046	1

Building Model

Once the pre-processing of data is done next, we apply the train data to the algorithm.

There are several Machine learning algorithms to be used depending on the data you are goingto process such as images,sound, text, and numerical values.The algorithms that you can choose according to the objective that you might have it may be Classification algorithms are Regression algorithms.

Example: 1. Linear Regression.

2. Logistic Regression.

3. Random Forest Regression / Classification.

4. Decision Tree Regression / Classification.

You will need to train the datasets to run smoothly and see an incremental improvement in the prediction rate.

Now we apply the Decision Tree algorithm on our dataset.

Machine Learning and Concepts:

Four machine learning models have been used for the prediction of loan approvals. Below are the description of the models used:

Decision Tree Regression Algorithm

This is a supervised machine learning algorithm that is primarily used for classification tasks. In this model, all features should be discretized such that the population may be divided into two or more homogenous groups or subsets. This model divides a node into two or more sub-nodes using a separate algorithm. The homogeneity and purity of the nodes grows in relation to the dependent variable as additional sub-nodes are created.

```
from sklearn.tree import DecisionTreeClassifier
dt = DecisionTreeClassifier()
dt.fit(X_train, y_train)

y_pred_dt =dt.predict(X_test)  #prediction
c(y_pred_dt)
```

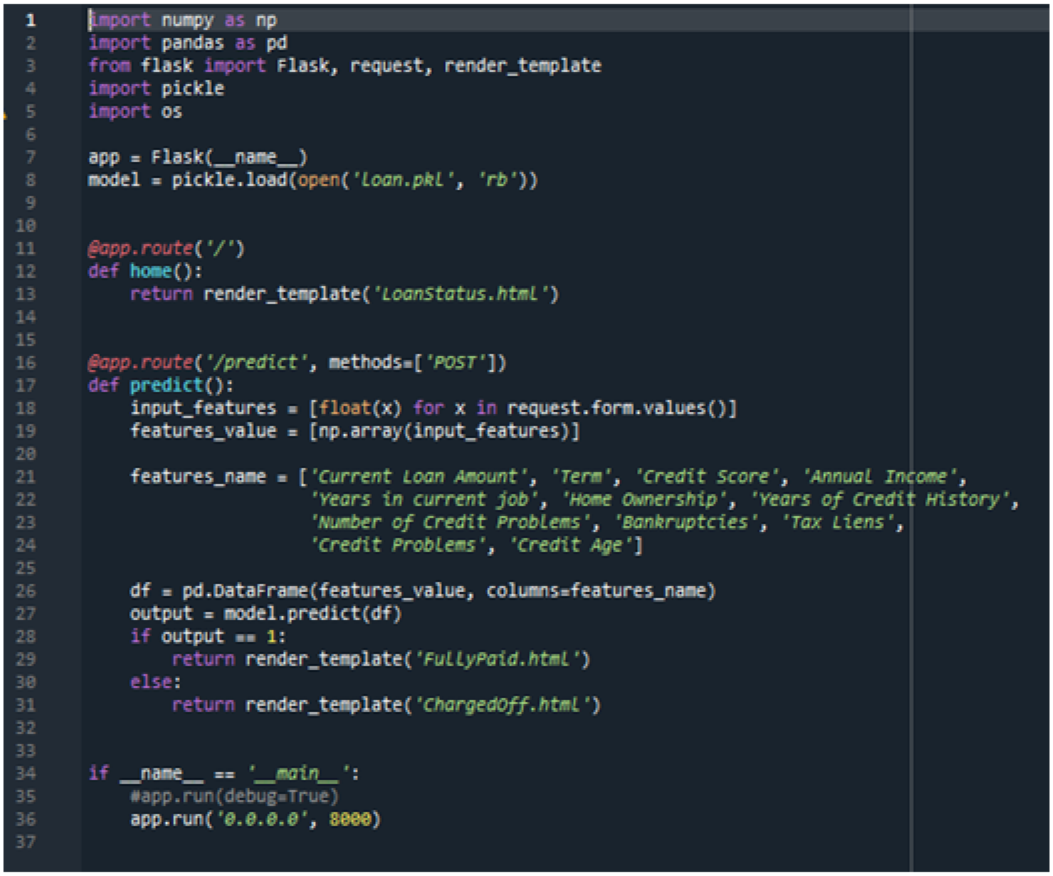
```
Counter({0: 6716, 1: 26284})
```

Creating a pickle for dumping the model:

```
#importing the pickle file
import pickle
#Dumping the model into the pickle file
pickle.dump(dt,open('loan.pkl','wb'))
```

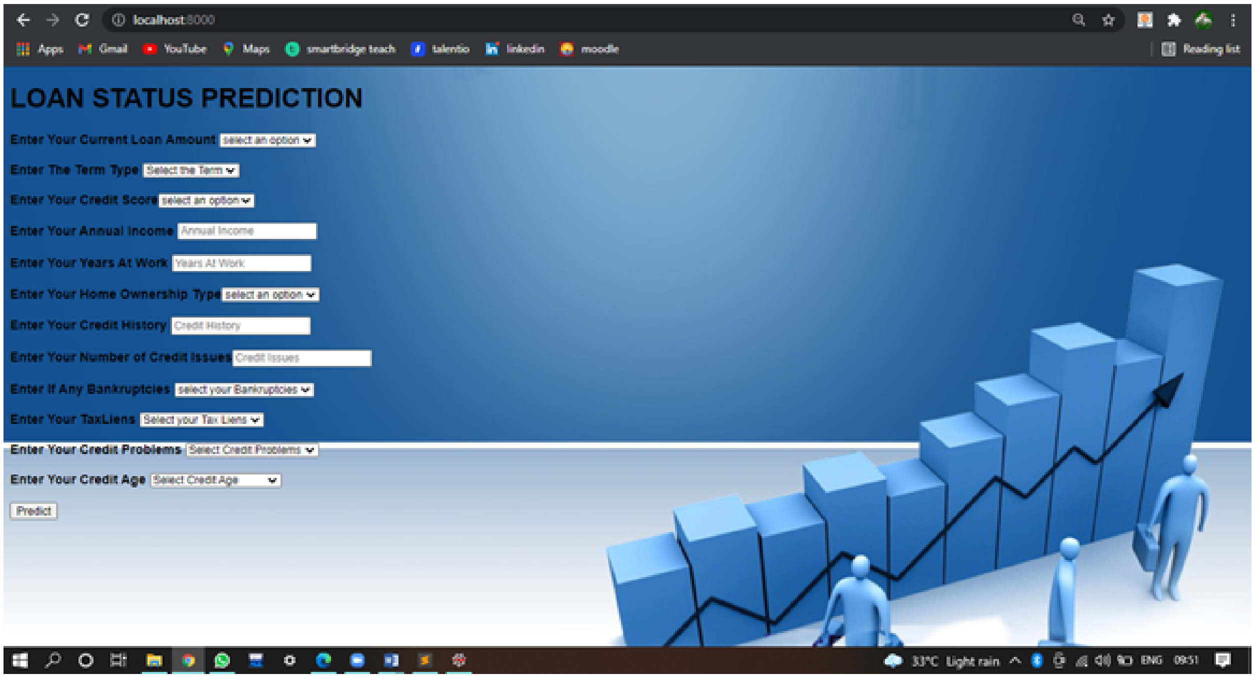
Build Flask Application

Flask Frame Work with Machine Learning Model In this section, we will be building a web application that is integrated into the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.



```
import numpy as np
import pandas as pd
from flask import Flask, request, render_template
import pickle
import os

app = Flask(__name__)
model = pickle.load(open('Loan.pkl', 'rb'))


@app.route('/')
def home():
    return render_template('LoanStatus.html')


@app.route('/predict', methods=['POST'])
def predict():
    input_features = [float(x) for x in request.form.values()]
    features_value = [np.array(input_features)]

    features_name = ['Current Loan Amount', 'Term', 'Credit Score', 'Annual Income',
                     'Years in current job', 'Home Ownership', 'Years of Credit History',
                     'Number of Credit Problems', 'Bankruptcies', 'Tax Liens',
                     'Credit Problems', 'Credit Age']

    df = pd.DataFrame(features_value, columns=features_name)
    output = model.predict(df)
    if output == 1:
        return render_template('FullyPaid.html')
    else:
        return render_template('ChargedOff.html')


if __name__ == '__main__':
    #app.run(debug=True)
    app.run('0.0.0.0', 8000)
```

Enter the values, press the predict button, and the result/prediction will be displayed on the web page



LOAN STATUS PREDICTION

Enter Your Current Loan Amount select an option

Enter The Term Type Select the Term

Enter Your Credit Score select an option

Enter Your Annual Income Annual Income

Enter Your Years At Work Years At Work

Enter Your Home Ownership Type select an option

Enter Your Credit History Credit History

Enter Your Number of Credit Issues Credit Issues

Enter If Any Bankruptcies select your Bankruptcies

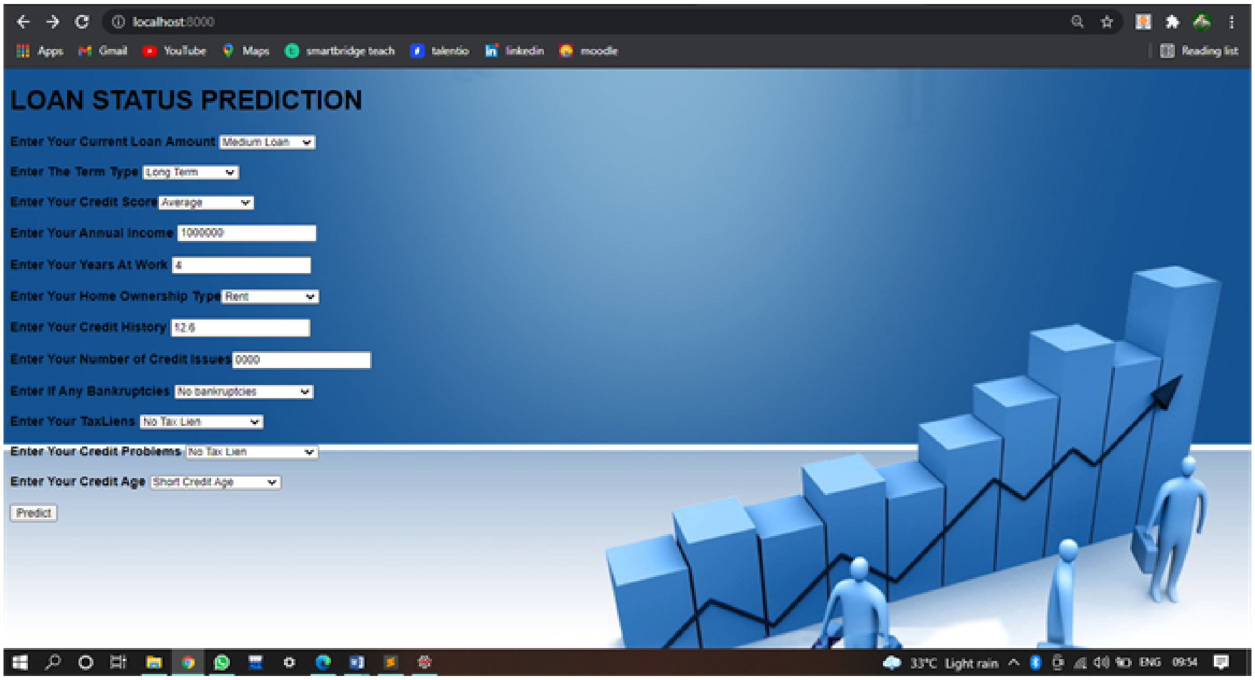
Enter Your TaxLiens Select your Tax Liens

Enter Your Credit Problems Select Credit Problems

Enter Your Credit Age Select Credit Age

Predict

Input:



LOAN STATUS PREDICTION

Enter Your Current Loan Amount Medium Loan

Enter The Term Type Long Term

Enter Your Credit Score Average

Enter Your Annual Income 1000000

Enter Your Years At Work 4

Enter Your Home Ownership Type Rent

Enter Your Credit History 12.6

Enter Your Number of Credit Issues 0000

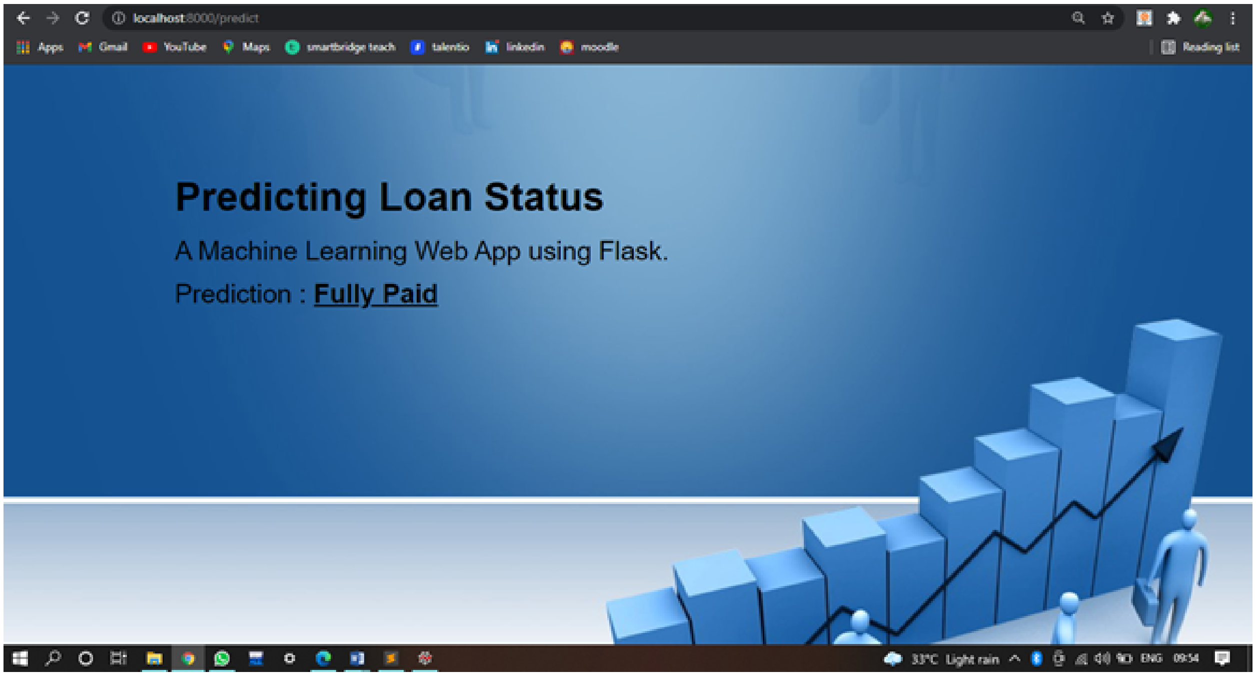
Enter If Any Bankruptcies No bankruptcies

Enter Your TaxLiens No Tax Lien

Enter Your Credit Problems No Tax Lien

Enter Your Credit Age Short Credit Age

output:



Predicting Loan Status

A Machine Learning Web App using Flask.

Prediction : Fully Paid

Conclusion:

We performed exploratory data analysis on the dataset's attributes to see how they are distributed.

We used charts to perform bivariate and multivariate analysis to understand how their features impacted one another.
We looked at each variable to see if the data was clean and evenly distributed. The data was cleansed and NA values were deleted.
We also devised hypotheses to demonstrate a link between the Independent and Target variables. And we inferred whether or not there is a link based on the findings.
We computed correlations between independent variables and discovered a substantial relationship between applicant income and loan amount.
To build the model, we added fake variables.
We built models that took a variety of variables into consideration and discovered that credit credit history has the greatest impact on loan approval. Finally, we generated the most accurate model using coapplicant income and credit history as independent variables.
We tested the data and found it to be 69.17 percent accurate.