Loan Approval Prediction

Submitted in partial fulfillment of the requirements for the award of degree of

BACHELOR OF
TECHNOLOGY IN
COMPUTER SCIENCE & ENGINEERING

Submitted to: Prof.
Dr. Vishan Gupta

Submitted By:

NAME	(Enrollment No.)
Ukant Jadia	20CS002286
Aatmagyaa Upadhyay	20CS002286

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Sir Padampat Singhania University, Udaipur

April 2023

I. Index

1. Introduction

The banking sector faces two crucial questions: the level of risk posed by a borrower and whether it is worth lending to them given that risk. The answer to the first question determines the borrower's interest rate, which reflects their level of risk. The higher the interest rate, the riskier the borrower is perceived to be. The interest rate, among other factors such as the time value of money, helps to determine whether an applicant is suitable for a loan. Lenders provide loans to borrowers in exchange for interest-bearing repayment guarantees, which means that they only receive a return if the borrower repays the loan. However, if the borrower defaults on the loan, the lender loses money. Banks provide loans to customers in exchange for repayment guarantees, but some customers default on their debts, leaving the bank with a loss. Banks mitigate this risk by retaining insurance to minimize the possibility of failure in the event of a default. This insurance can cover the entire loan amount or a portion of it.

Manual processes have traditionally been used by banks to determine whether a borrower is suitable for a loan based on their results. While these manual processes have been effective, they are insufficient when there are a large number of loan applications, and making a decision can take a long time. To address this issue, loan prediction machine learning models can be used to evaluate a customer's loan status and develop strategies. These models extract and introduce the essential features of a borrower that influence their loan status, and then produce the expected performance, or loan status. These reports simplify and expedite the job of bank managers.

In conclusion, the banking sector faces two critical questions when assessing borrowers: their level of risk and whether lending to them is worth the risk. Interest rates reflect a borrower's risk level and help determine whether an applicant is suitable for a loan. Lenders and banks provide loans in exchange for repayment guarantees, but there is always the risk of default. To simplify and speed up the process of assessing loan status, loan prediction machine learning models extract essential borrower features to produce a loan status report. These reports help bank managers make more informed and quicker decisions.

1.1. Statement of the Problem

The objective of this project is to develop a machine learning model that predicts whether a loan applicant will be granted a loan or not. The model will be trained on a dataset of historical loan applications that includes applicant demographics, financial information, and other relevant features. The model will take into account factors such as credit score, income, debt-to-income ratio, loan amount, and loan purpose, among others. The accuracy and reliability of the model will be evaluated using various performance metrics. The ultimate goal is to provide a tool that can assist loan officers in making more informed and consistent loan approval decisions, while reducing the risk of default and improving the overall efficiency of the lending process.

2. Literature Survey

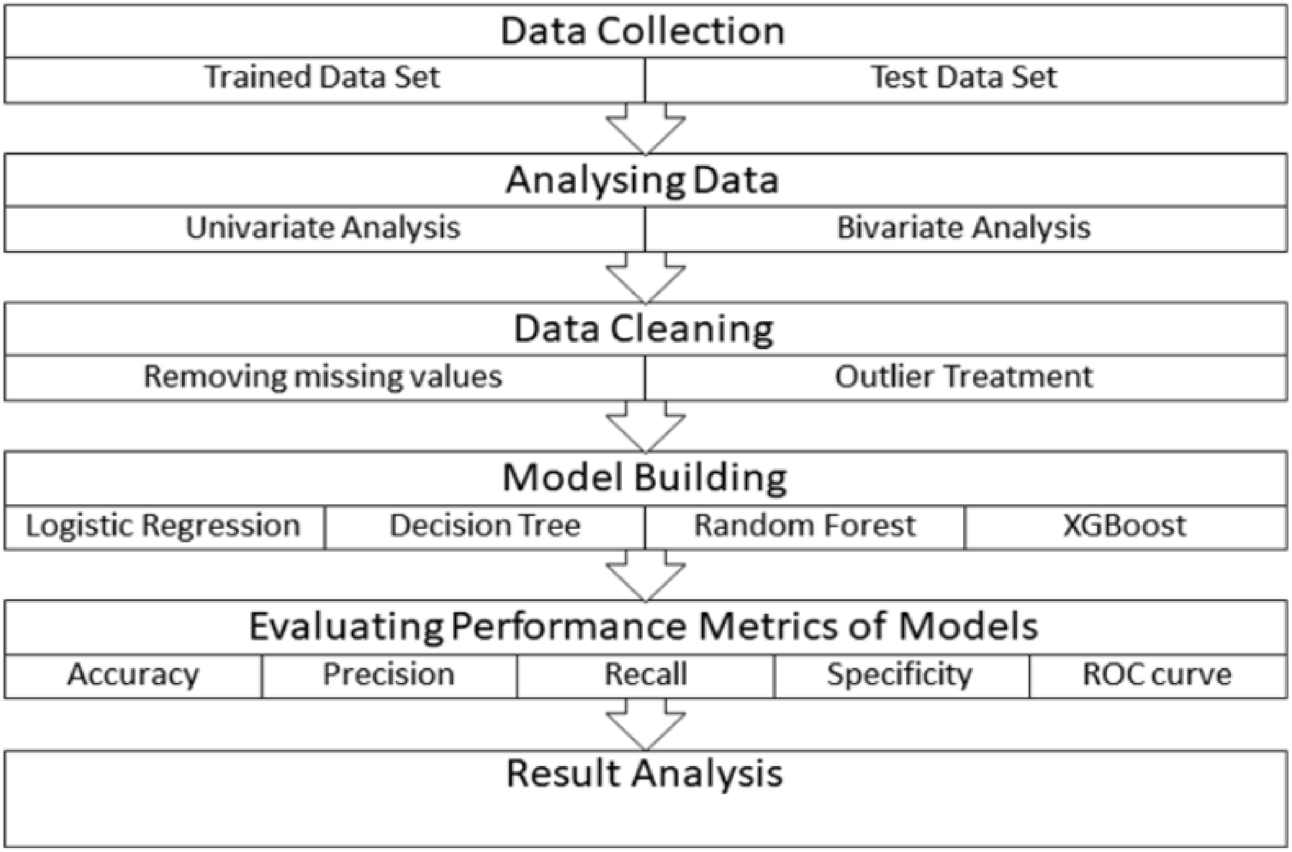
We could not find any literature review for loan prediction for specific Machine learning algorithms to use which would be a possible starting point for our paper. Instead, since loan prediction is a classification problem, we went with popular classification algorithms used for a similar problem.

Rajiv Kumar and Vinod Jain [1] proposed a model using machine learning algorithms to predict the loan approval of customers. They applied three machine learning algorithms, Logistic Regression (LR), Decision Tree (DT), and Random Forest (RF) using Python on a test data set. From the results, they concluded that the Decision Tree machine learning algorithm performs better than Logistic Regression and Random Forest machine learning approaches. It also opens other areas on which the Decision Tree algorithm is applicable.

Ashlesha Vaidya [2] used logistic regression as a probabilistic and predictive approach to loan approval prediction. The author pointed out how Artificial neural networks and Logistic regression are most used for loan prediction as they are easier to comparatively develop and provide the most accurate predictive analysis. One of the reasons behind this is that other Algorithms are generally bad at predicting from non-normalized data. But the nonlinear effect and power terms are easily handled by Logistic regression as there is no need for the independent variables on which the prediction takes place to be normally distributed.

Anchal Goyal and Ranpreet Kaur [9] discuss various ensemble algorithms. Ensemble algorithm is a supervised machine learning algorithm that is a combination of two or more algorithms to get better predictive performance. They carried out a systematic literature review to compare ensemble models with various stand-alone models such as neural network, SVM, regression, etc.

3. Methodology



3.1. Machine Learning Concept

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

- Logistic Regression

This is a classification algorithm which uses a logistic function to predict binary outcome (True/False, 0/1, Yes/No) given an independent variable. The aim of this model is to find a relationship between features and probability of particular outcome. The logistic function used is a logit function which is a log of odds in the favor of the event. Logit function develops a s-shaped curve with the probability estimate similar to a step function.

- Decision Trees

This is a supervised machine learning algorithm mostly used for classification problems. All features should be discretized in this model, so that the population can be split into two or more homogeneous sets or subsets. This model uses a different algorithm to split a node into two or more sub-nodes. With the creation of more sub-nodes, homogeneity and purity of the nodes increases with respect to the dependent variable.

- Support Vector Machine

Support Vector Machines (SVM) is a popular algorithm used in supervised machine learning for classification and regression tasks' SVM seeks to find a hyperplane that separates data into different classes while maximizing the margin between the classes. SVM has proven to be a powerful and flexible algorithm with good performance on a variety of tasks, making it a popular choice for machine learning practitioners

- K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a simple yet effective algorithm used in supervised machine learning for classification and regression tasks. KNN works by finding the k closest data points in the training set to a new data point and using them to make a prediction. KNN is easy to understand, implement, and interpret, making it a popular choice for machine learning practitioners

4. Libraries for Data Analysis

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.

5. Understanding the Data

Columns	Description
Loan_ID	A uniques loan ID
Gender	Male/Female
Married	Married(Yes)/ Not married(No)
Dependents	Number of persons depending on the client
Education	Applicant Education (Graduate /Undergraduate)
Self_Employed	Self emplyored (Yes/No)
ApplicantIncome	Applicant income
Coapplicant income	Coapplicant Income
LoanAmount	Loan amount in thousands
Loan_Amount_Term	Term of lean in months
Credit_Hostory	Credit history meets guidelines
Property_Area	Urban/Semi and Rural
Loan_Status	Loan approved (Y/N)

5.1 Data Description

Loan amount: This is the total amount of money that the loan applicant is requesting to borrow.

Loan amount term: This refers to the length of time over which the loan will be repaid

Applicant income: This refers to the amount of money that the person applying for the loan earns on a regular basis.

Gender: This is the loan applicant's gender, which is recorded for statistical purposes only.

Married: This refers to the loan applicant's marital status. Lenders may use this information to evaluate the applicant's financial stability and ability to repay the loan.

Dependents: This refers to the number of people that the loan applicant supports financially (such as children or elderly parents).

Education: This refers to the loan applicant's level of education.

Employed: This refers to the loan applicant's employment status.

Credit history: This refers to the loan applicant's past credit behavior, including any loans or credit cards they may have had in the past.

Co-applicant income: If the loan applicant has a co-applicant (such as a spouse or partner), their income will also be taken into consideration by the lender.

5.2 Data Types

```
The Datatypes of features are as :

Loan_ID               object
Gender                object
Married               object
Dependents            object
Education             object
Self_Employed         object
ApplicantIncome        int64
CoapplicantIncome    float64
LoanAmount           float64
Loan_Amount_Term     float64
Credit_History       float64
Property_Area         object
Loan_Status           object
dtype: object


We have 614 rows and 13 columns. 12 independent variable and target variable:
(614, 13)
```

6. Innovations in Project

7. Source Code and Images

7.1. Source Code `model.py`

```python
import pandas as pd
# import pickle
import joblib
## Preparing model
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
model = LogisticRegression(random_state=42)
## Preparing encoder
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
## Collecting the dataset
URL = "https://raw.githubusercontent.com/ukantjadia/Loan-Prediction/Main/Data/train_u6lujuX_CVtuZ9i.csv"
df = pd.read_csv(URL)
# df = pd.read_csv("../Data/train_u6lujuX_CVtuZ9i.csv")
## Droping the Loan id
df.drop('Loan_ID',axis=1,inplace=True)

## Converting the data type of credit_history int to object
df['Credit_History'] = df['Credit_History'].astype('O')

# %%
## Seprating the data types
Tcat_data = []
Tnum_data = []
for name, dtype in enumerate(df.dtypes):
    if dtype == object:
        Tcat_data.append(df.iloc[:,name])
    else:
        Tnum_data.append(df.iloc[:,name])
Tcat_data = pd.DataFrame(Tcat_data).T
Tnum_data = pd.DataFrame(Tnum_data).T

## Handling the missing value in categorical and numerical data
### categorical data
Tcat_data = Tcat_data.apply(lambda x: x.fillna(x.value_counts().index[0]))
### Numerical data
```



```python
        Tnum_data.append(df.iloc[:,name])
Tcat_data = pd.DataFrame(Tcat_data).T
Tnum_data = pd.DataFrame(Tnum_data).T

## Handling the missing value in categorical and numerical data
### categorical data
Tcat_data = Tcat_data.apply(lambda x: x.fillna(x.value_counts().index[0]))
### Numerical data
```



```python
Tnum_data.fillna(method='bfill',inplace=True)

## Transforming the categorical data
target_values = {'Y':0,'N':1}
target = Tcat_data['Loan_Status']
Tcat_data.drop('Loan_Status',axis=1,inplace=True)
target = target.map(target_values)

Transform_cat_data = pd.DataFrame()
for data in Tcat_data:
    Transform_cat_data[data] = le.fit_transform(Tcat_data[data])
X = pd.concat([Tnum_data,Transform_cat_data],axis=1)
y = target
## Spliting the dataset for training and testing
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2)
## Training and testing the dataset
model.fit(X_train,y_train)
y_pred = model.predict(X_test)

## Calculating accuracy score
from sklearn.metrics import accuracy_score
acc = accuracy_score(y_test,y_pred)
print(acc)

## Saving moodel
joblib.dump(model,'model.sav')
```

7.2. Source Code `app.py`

```python
import streamlit as st
import pandas as pd
import numpy as np
import warnings
import joblib

from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()

def load_model(modelfile):
    loaded_model = joblib.load(modelfile)
    return loaded_model


def main():
    # title
    html_temp = """
    <div>
    <h1 style="color:MEDIUMSEAGREEN;text-align:left;"> Loan Prediction 💵🏦 </h1>
    </div>
    """
    st.markdown(html_temp, unsafe_allow_html=True)

    st.sidebar.write(""" ## How does it work ❓
The loan approval prediction model uses your credit score, income, education, loan amount and other various factor  to determine whether your loan will be approved or not.  It outputs a "yes" or "no" decision.
                """)

```

```python
                """)

    with st.sidebar.expander(" ℹ Information", expanded=True):
        st.write("""
**Applicant income**: This refers to the amount of money that the person applying for the loan earns on a regular basis.

**Co-applicant income**: If the loan applicant has a co-applicant (such as a spouse or partner), their income will also be taken into consideration by the lender.

**Loan amount**: This is the total amount of money that the loan applicant is requesting to borrow.

**Loan amount term**: This refers to the length of time over which the loan will be repaid.

**Gender**: This is the loan applicant's gender, which is recorded for statistical purposes only.

**Married**: This refers to the loan applicant's marital status. Lenders may use this information to evaluate the applicant's financial stability and ability to repay the loan.

**Dependents**: This refers to the number of people that the loan applicant supports financially (such as children or elderly parents).

**Education**: This refers to the loan applicant's level of education.

**Employed**: This refers to the loan applicant's employment status.

**Credit history**: This refers to the loan applicant's past credit behavior, including any loans or credit cards they may have had in the past.
        """)

```

```python

    with st.expander(" ℹ About", expanded=True):
        st.write("""
                Loan Prediction is very helpful for employee of banks
                as well as for the applicant also.
                Dream housing Finance Company
                deals in all loans. They have presence across all urban,
                semi urban and rural areas. Customer first apply for loan
                after that company or bank validates the customer
                eligibility for loan. Company or bank wants to automate
                the loan eligibility process (real time) based on customer
                details provided while filling application form. These
                details are Gender, Marital Status, Education, Number of
                Dependents, Income, Loan Amount, Credit History and
                other
        """)



    with st.form("my_form"):
        col4,col6 = st.columns([2,2])
        with col4:
            education = st.radio("Education", ['Graduate','Not Graduate'])
            married = st.radio("Married", ['Yes','No'])
            loanAmountTerm = st.number_input("Loan Amount Term", 10.0,600.0)
            applicantIncome = st.number_input("Applicant Income", 150.0,100000.0)
            propertyArea = st.radio("Property Area", ['Ruler','Semiurban','Urban'])
        with col6:
            employed = st.radio("Self Employed", ['Yes','No'])
            gender = st.radio("Gender", ['Male','Female'])
            coApplicantIncome = st.number_input("Co Applicant Income", 0.0,50000.0)
            loanAmount = st.number_input("Loan Amount", 10.0,1000.0)
            creditHistory = str(st.slider("Credit History",1,3000))
            dependents = st.slider("Dependents", 0,4)

```

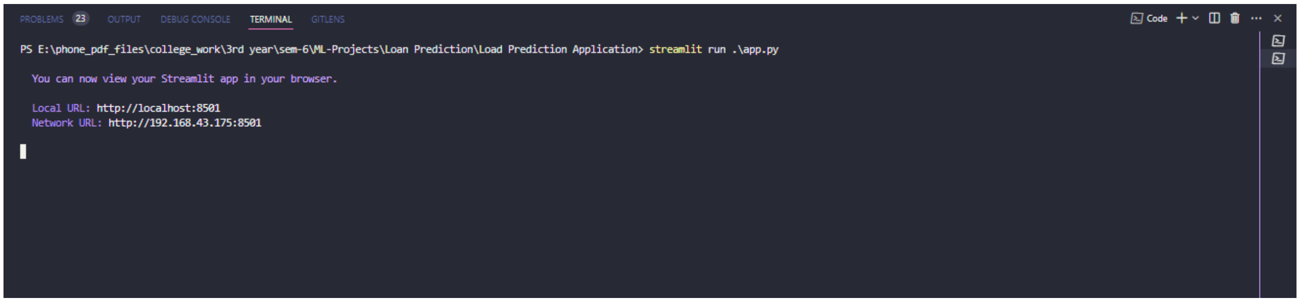
```python
        # Every form must have a submit button.
        submitted = st.form_submit_button("Submit")
        if submitted:
            num_data = [applicantIncome, coApplicantIncome, loanAmount, loanAmountTerm]
            cat_data = [gender, married, dependents, education, employed,creditHistory, propertyArea]
            test=[]
            column=['applicagoinIncome','coApplicantIncome','loanAmount','loanAmountTrem','gender','married','dependent','education','employed','creditHistory','propertyArea']
            test = le.fit_transform(cat_data).tolist()
            feature_list = num_data + test
            # df = pd.DataFrame([feature_list],columns=column)
            single_pred = np.array(feature_list).reshape(1,-1)
            # single_pred = np.array(feature_list)
            st.write(feature_list)
            loaded_model = load_model('model.sav')
            prediction = loaded_model.predict(single_pred)
            st.write('''
            ## Results 🔍
            ''')
            if prediction[0] == "0":
                st.write("Yes")
            else:
                st.write("No")


hide_menu_style = """
        <style>
        #MainMenu {visibility: hidden;}
        </style>
        """
st.markdown(hide_menu_style, unsafe_allow_html=True)

if __name__ == '__main__':
    main()
```

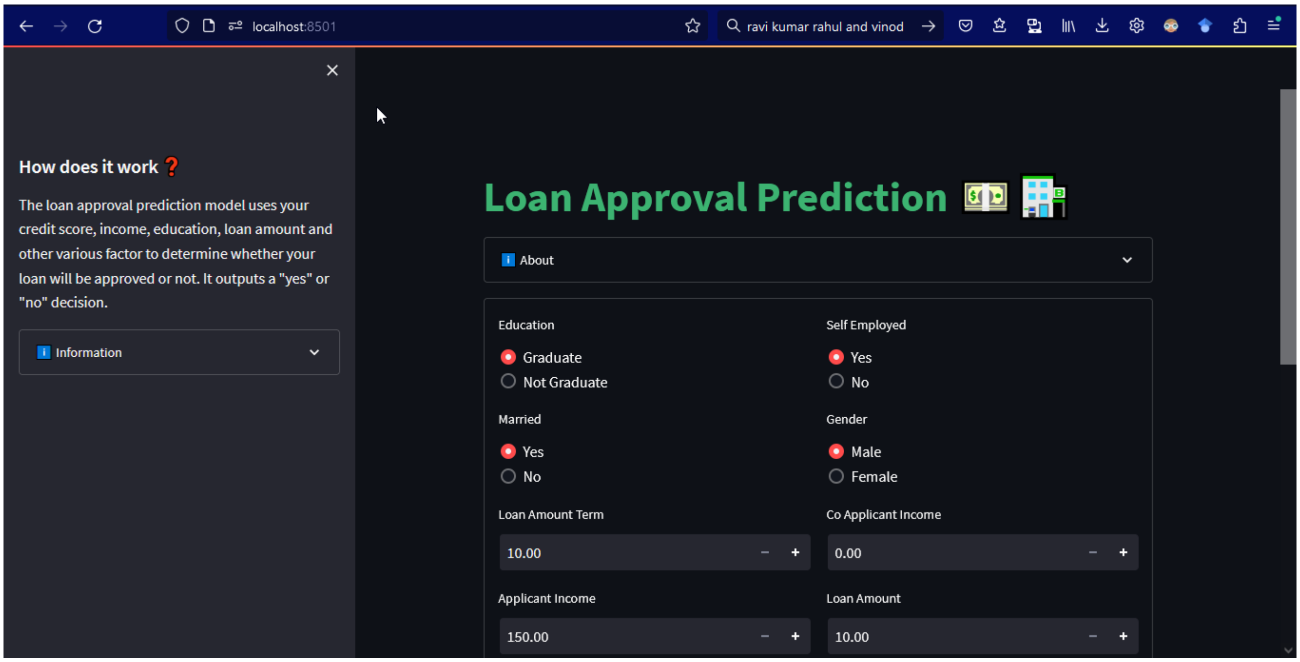
```python
footer="""<style>
a:link , a:visited{
color: white;
background-color: transparent;
text-decoration: underline;
}

a:hover,  a:active {
color: red;
background-color: transparent;
text-decoration: underline;
}

.footer {
position: relative;
left: 0;
bottom: 0;
width: 100%;
background-color: #0e1117;
color: white;
text-align: center;
}
</style>
<div class="footer">
<p>Developed with 💛 by <a style='display: block; text-align: center;' href="https://ukantjadia.me/linkedin" target="_blank">Ukant Jadia & Aatmgyaa Upadhyay</a></p>
</div>
"""
st.markdown(footer,unsafe_allow_html=True)
```

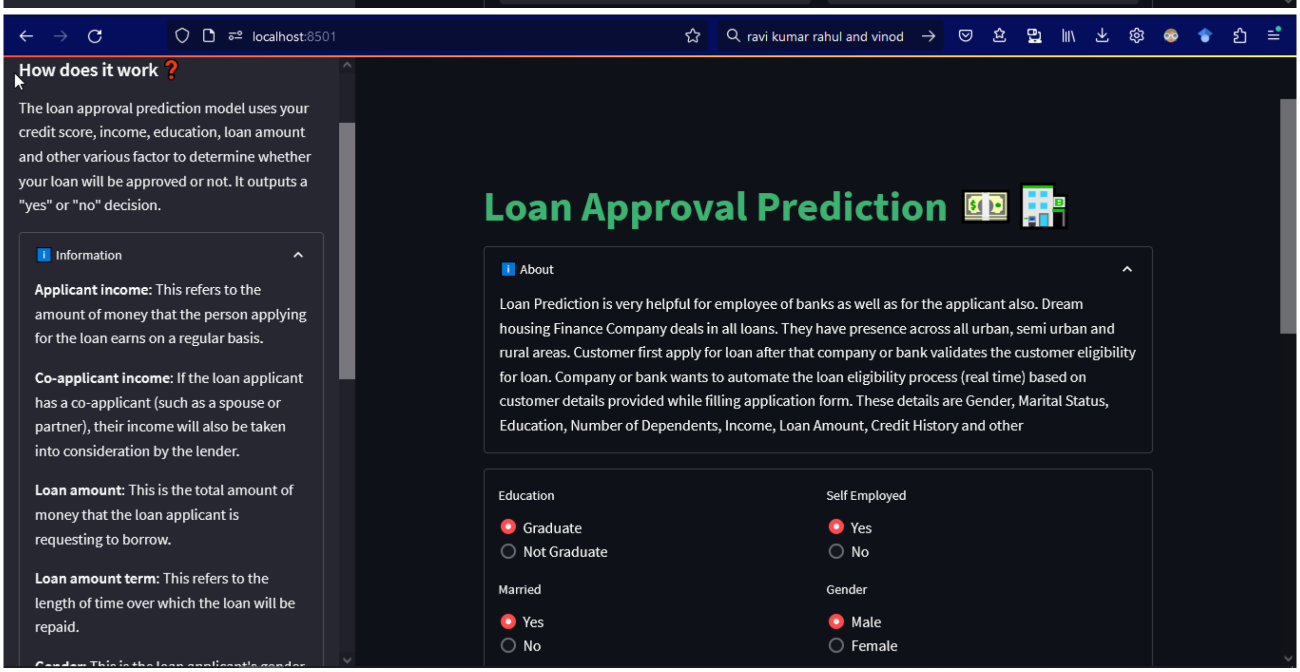
7.3. Output of `model.py`

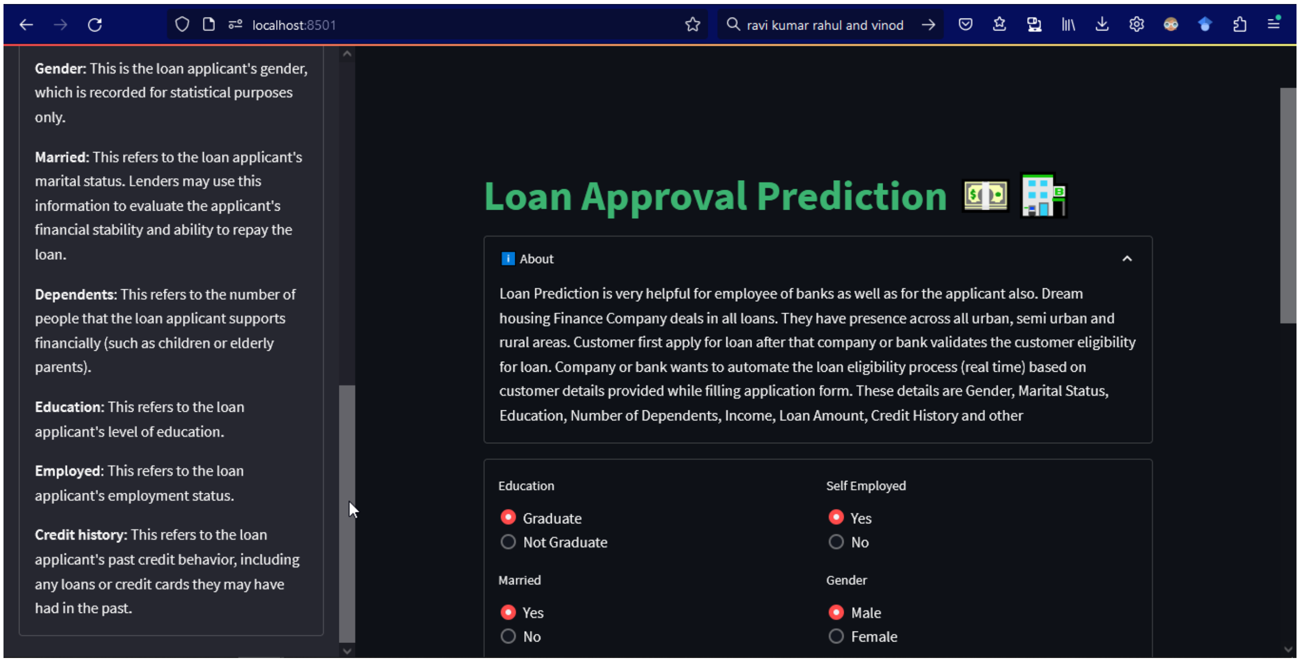
```
PS E:\phone_pdf_files\college_work\3rd year\sem-6\ML-Projects\Loan Prediction> python -u "e:\phone_pdf_files\college_work\3rd year\sem-6\ML-Projects\Loan Prediction\Load Prediction Application\model.py"
Accuracy of the model is  0.8373983739837398
```

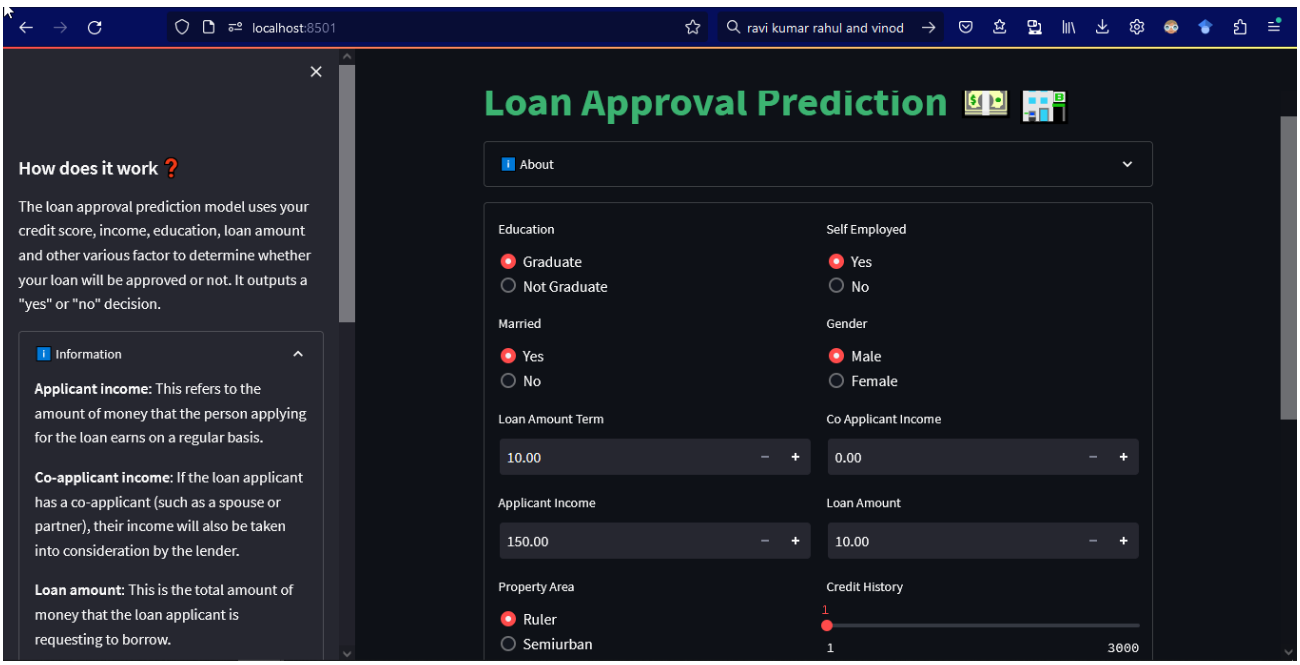
7.4.Output of `app.py`

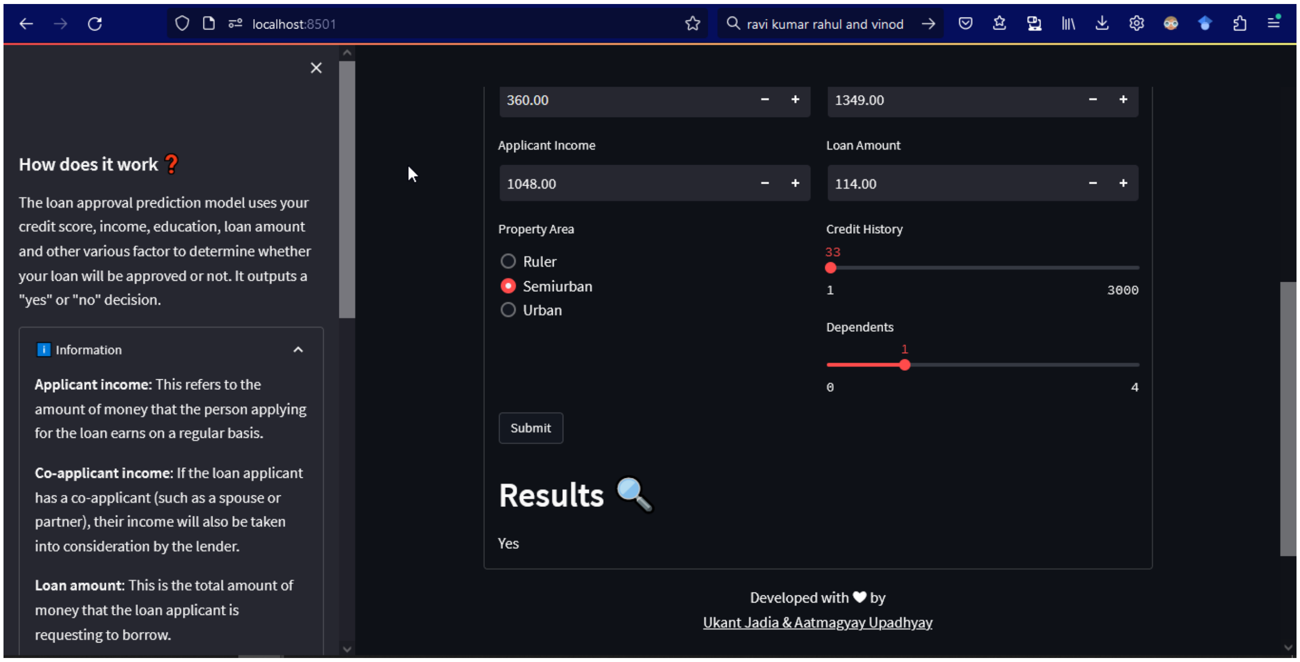







localhost:8501
ravi kumar rahul and vinod
Gender: This is the loan applicant's gender, which is recorded for statistical purposes only.
Married: This refers to the loan applicant's marital status. Lenders may use this information to evaluate the applicant's financial stability and ability to repay the loan.
Dependents: This refers to the number of people that the loan applicant supports financially (such as children or elderly parents).
Education: This refers to the loan applicant's level of education.
Employed: This refers to the loan applicant's employment status.
Credit history: This refers to the loan applicant's past credit behavior, including any loans or credit cards they may have had in the past.
Loan Approval Prediction
About
Loan Prediction is very helpful for employee of banks as well as for the applicant also. Dream housing Finance Company deals in all loans. They have presence across all urban, semi urban and rural areas. Customer first apply for loan after that company or bank validates the customer eligibility for loan. Company or bank wants to automate the loan eligibility process (real time) based on customer details provided while filling application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and other
Education
Graduate
Not Graduate
Self Employed
Yes
No
Married
Yes
No
Gender
Male
Female


localhost:8501
ravi kumar rahul and vinod
How does it work ❓
The loan approval prediction model uses your credit score, income, education, loan amount and other various factor to determine whether your loan will be approved or not. It outputs a "yes" or "no" decision.
Information
Applicant income: This refers to the amount of money that the person applying for the loan earns on a regular basis.
Co-applicant income: If the loan applicant has a co-applicant (such as a spouse or partner), their income will also be taken into consideration by the lender.
Loan amount: This is the total amount of money that the loan applicant is requesting to borrow.
Loan Approval Prediction
About
Education
Graduate
Not Graduate
Self Employed
Yes
No
Married
Yes
No
Gender
Male
Female
Loan Amount Term
10.00
Co Applicant Income
0.00
Applicant Income
150.00
Loan Amount
10.00
Property Area
Ruler
Semiurban
Credit History
1
1
3000


localhost:8501
ravi kumar rahul and vinod
How does it work ❓
The loan approval prediction model uses your credit score, income, education, loan amount and other various factor to determine whether your loan will be approved or not. It outputs a "yes" or "no" decision.
Information
Applicant income: This refers to the amount of money that the person applying for the loan earns on a regular basis.
Co-applicant income: If the loan applicant has a co-applicant (such as a spouse or partner), their income will also be taken into consideration by the lender.
Loan amount: This is the total amount of money that the loan applicant is requesting to borrow.
360.00
1349.00
Applicant Income
1048.00
Loan Amount
114.00
Property Area
Ruler
Semiurban
Urban
Credit History
33
1
3000
Dependents
1
0
4
Submit
Results 🔍
Yes
Developed with ❤ by
Ukant Jadia & Aatmagyay Upadhyay

8. Result

We have created a loan approval prediction model using logistic regression. The model has an accuracy of 83%, which means that it correctly predicts loan approvals 83% of the time. This is a good result and suggests that our model is effective in predicting loan approvals. We can use this model to predict loan approvals for new loan applications and improve our lending process.

9. Conclusion

So here, it can be concluded with confidence that the Logistic Regression model is efficient and gives a better result when compared to other models. It works correctly and fulfills all requirements of bankers. This system properly and accurately calculates the result. It predicts whether the loan is approved or rejected to the loan applicant or customer very. We have created a loan approval prediction model using logistic regression. The model has an accuracy of 83%, which means that it correctly predicts loan approvals 83% of the time. This is a good result and suggests that our model is effective in predicting loan approvals. We can use this model to predict loan approvals for new loan applications and improve our lending process accurately.

10. Future Scope

Improve the model's accuracy: While an accuracy of 83% is good, there is always room for improvement. You can explore different machine learning algorithms and techniques to increase the accuracy of your model. This can involve adjusting the model's parameters, fine-tuning its hyperparameters, or using more advanced algorithms like neural networks.

Gather more data: We can collect and incorporate additional data points into your model to make it more robust and accurate. This can include data related to the borrower's employment history, income, and expenses, among other factors. Gathering more data can also help to address issues like class imbalance or missing values, which can affect the model's performance.

Explore real-time data: We can consider the use of real-time data to improve the accuracy and speed of loan approvals. However, it is important to ensure that the data is reliable and accurate, and that appropriate data privacy and security measures are in place.

11. Acknowledgement

It gives us immense pleasure in presenting the preliminary project report on ‘Loan Approval Prediction’. We would like to take this opportunity to thank our internal guide Prof. Vishan Gupta for giving us all the help and guidance we needed. We are really grateful to him for all kinds of support. Her valuable suggestions were very helpful. In the end we are also grateful to Assist. Prof. Chandani Joshi, Department of Computer Science Engineering, for his indispensable suggestions.

12. Bibliography

1. S. S Athreyas, T. B K, V. Kashyap S, S. S Bairy, and H. K, "A Comparative Study of Machine Learning Algorithms for Predicting Loan Default and Eligibility", *pices*, vol. 5, no. 12, pp. 116-118, Apr. 2022.

2. A. Vaidya, "Predictive and probabilistic approach using logistic regression: Application to prediction of loan approval," *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Delhi, India, 2017, pp. 1-6, doi: 10.1109/ICCCNT.2017.8203946
3. Goyal, A. and Kaur, R., 2016. Accuracy prediction for loan risk using machine learning models. *Int. J. Comput. Sci. Trends Technol*, *4*(1), pp.52-57.