A PROJECT REPORT

ON

LOAN PREDICTION USING MACHINE LEARNING ALGORITHMS

By

Ms. Sai Kumudini

In partial fulfilment of

COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)

Under

University of JNTUA

Department of computer science and engineering

Academic year 2021-2025

MADANAPALLE INSTITUTE OF TECHNOLOGY AND SCIENCES.

INDEX

ACKNOWLEDGEMENT

I undertook this project and completed the report under the guidance of Sandeepkaur (mentor) and academor consultancy . I am great full to the academor staff for their patience and assistance during my training at their company . It was a good learning experience for me.

The success of any project is never limited to an individual undertaking the project. It is collective effort of people around the individual that spell success. There are some key personalities involved, whose role has been very vital to pave way for success of the project . I take the opportunity to express our sincere thanks and gratitude to them.

Ms. Sai Kumudini

Computer science and engineering (Data science)

ABSTRACT

Process of providing loan can be tedious and time consuming. We must have a standard set of defined rules that can be applied to an entire population and help us to determine if one person is eligible to get a loan or not. In this project we have described a very effective way for customer loan prediction. Our main interest is todecide whether a customer will get the loan approved or not based on several factors. We are trying to auto mate the loan eligibility process (real time) based on customer details provided while filling an online application form. We have applied Logistics Regression and Random Forest to analyze and predict. Logistics regression and Random Forest gives us the probability whether a customer should get loan or not. Depending on the accuracy of these models we will select which one will best fit our data.

Introduction

For the past decade, for the extraction and manipulation of the data, data mining has become very efficient inorder to devise some patterns and to take accurate decisions. As we already know, to decrease randomness, we must increase information. Data mining has proven to be a very effective method of accumulating data and analyze it.In 1997, Berry proposed that the there are six different data mining phases for any human problem that can bestated as:

1.Classification
2.Estimation
3.Prediction
4.Affinity
5.Grouping
6.Description Of Problems

The whole process is called as “Knowledge Discovery”, that goes hand in hand with the statement of decreasing
randomness by increasing data. In 1998, Weiss classified Data mining into two parts: knowledge discovery and prediction. First part includes classification, regression whereas second part defines association rules and summarization. Knowledge Discovery Database (KDD) has three stages.
•Data Pre-Processing
•Data Mining
•Data Post-Processing For the initial stage, data processing is done which results in data collection, data smoothing, data transformation, data cleansing and data reduction.In the second stage which is called data mining which involves data classification commonly termed as prediction. The final and the third stage which we called data post-processing, which shows the conclusion part drawn from the analysis in the previous stage, on the basis of which we devise our further course of action.

SCOPE OF PROJECT

Online loan prediction is aims serving for validates the customer eligibility for loan. This paper is exclusively for the managing authority of finance company, whole process of prediction is done privately no stakeholders would be able to alter the processing. Result against particular Loan id can be send to various department of company so that they can take appropriate action on application. This helps all others department to carried out other formalities.

IMPLEMENTATION

Machine learning model used :-

1. Random forest classifier.
2. Logistic regression model.

Language used:-

Python

Tools used:-

Tableau

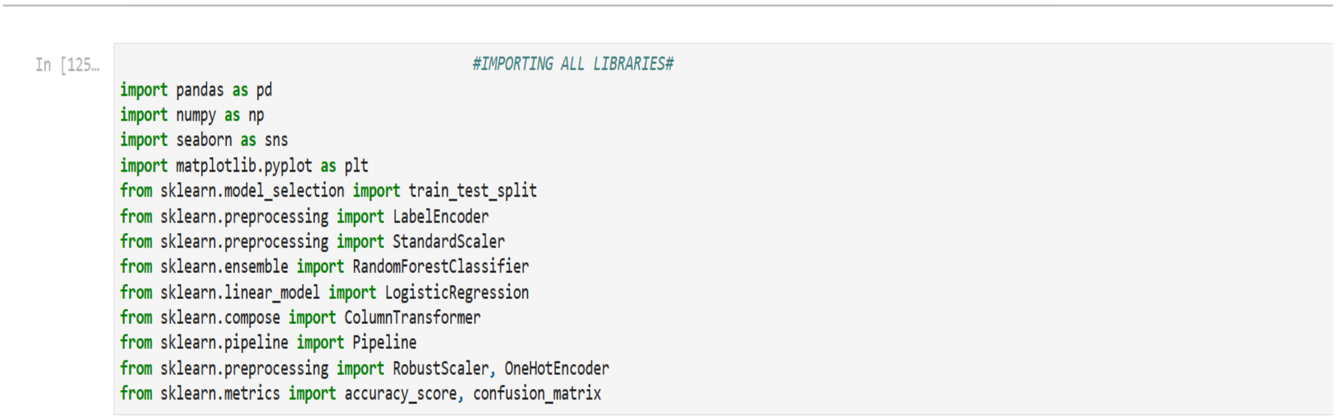
DATA

1. Loan_ID
2. Gender
3. Married
4. Dependents
5. Education
6. Self Employed
7. Applicant Income
8. Coapplicant Income
9. Loan amount
10. Loan amount term
11. Credit history
12. property Area
13. Loan status

These are the columns given in a loan prediction dataset.

We need to do apply libraries, machine learning algorithms and visualizations to the given above dataset.

Step 1:- Import all the necessary libraries

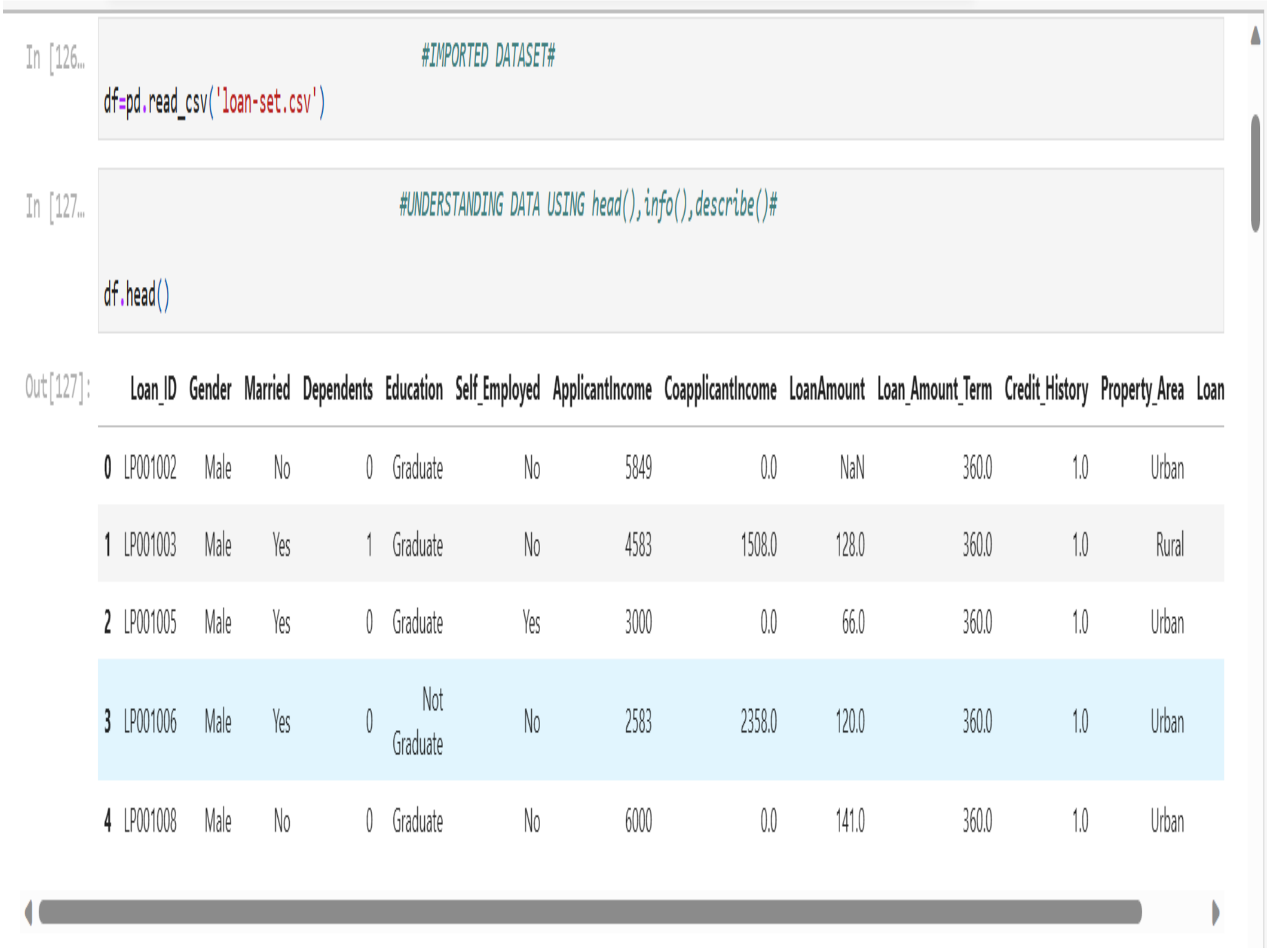


In [125…

```
                                        #IMPORTING ALL LIBRARIES#
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import RobustScaler, OneHotEncoder
from sklearn.metrics import accuracy_score, confusion_matrix
```

Step 2:- Import the provided dataset

Step 3:- understanding the dataset



In [126…

```
                          #IMPORTED DATASET#
df=pd.read_csv('loan-set.csv')
```

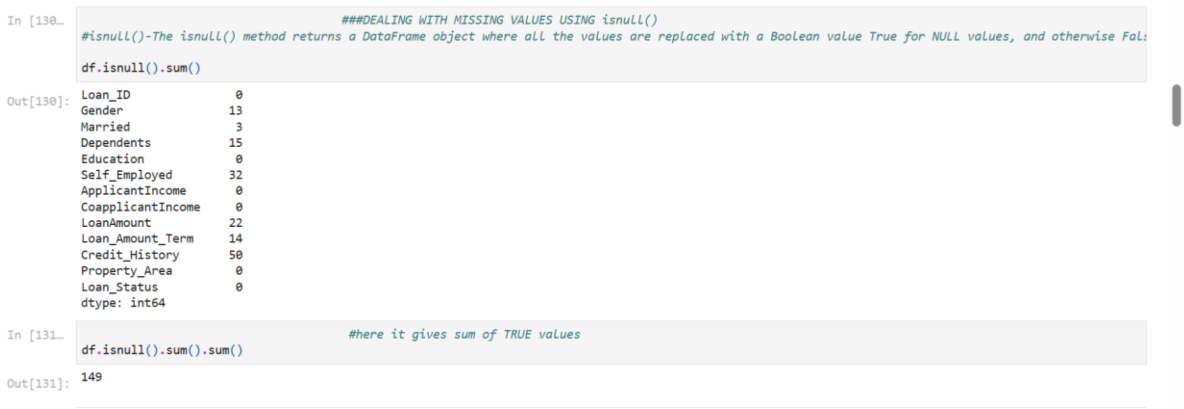
In [127…

```
                          #UNDERSTANDING DATA USING head(),info(),describe()#


df.head()
```

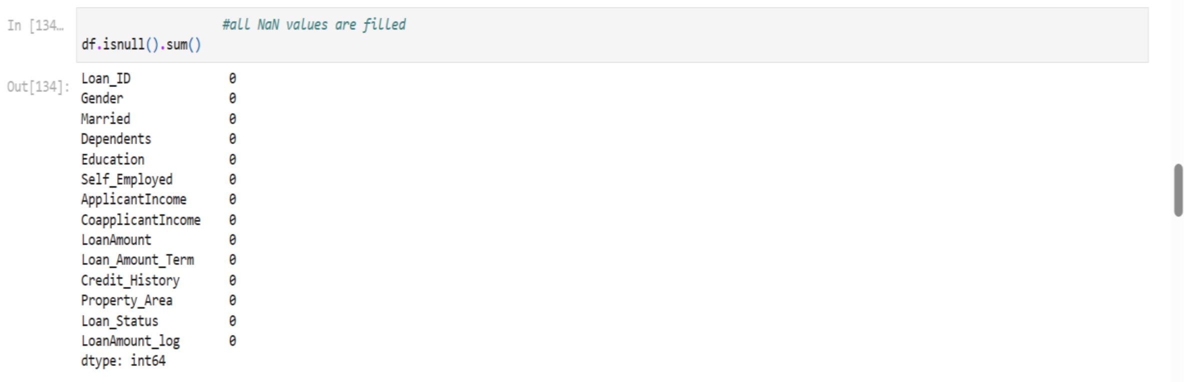
Out[127]:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	

Step 4:- dealing with missing values

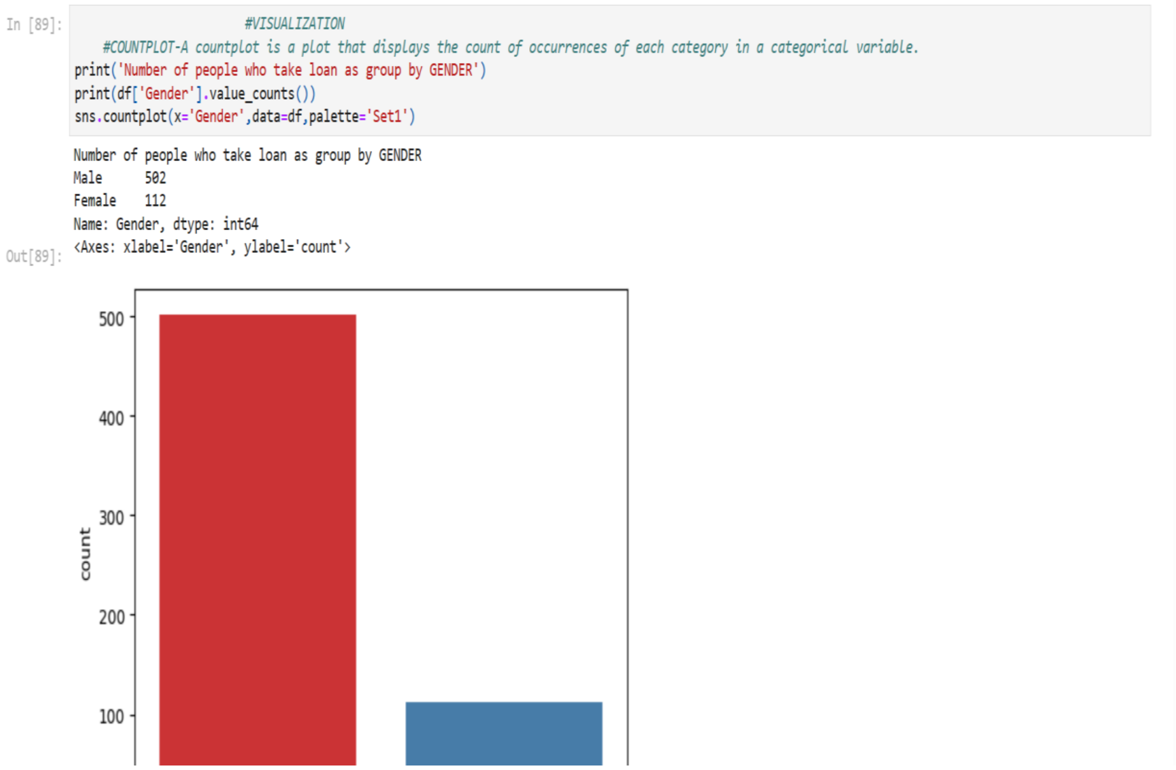


```
In [130…                        ###DEALING WITH MISSING VALUES USING isnull()
         #isnull()-The isnull() method returns a DataFrame object where all the values are replaced with a Boolean value True for NULL values, and otherwise Fal

         df.isnull().sum()

Out[130]: Loan_ID              0
          Gender              13
          Married              3
          Dependents          15
          Education            0
          Self_Employed       32
          ApplicantIncome      0
          CoapplicantIncome    0
          LoanAmount          22
          Loan_Amount_Term    14
          Credit_History      50
          Property_Area        0
          Loan_Status          0
          dtype: int64

In [131…                        #here it gives sum of TRUE values
         df.isnull().sum().sum()

Out[131]: 149
```

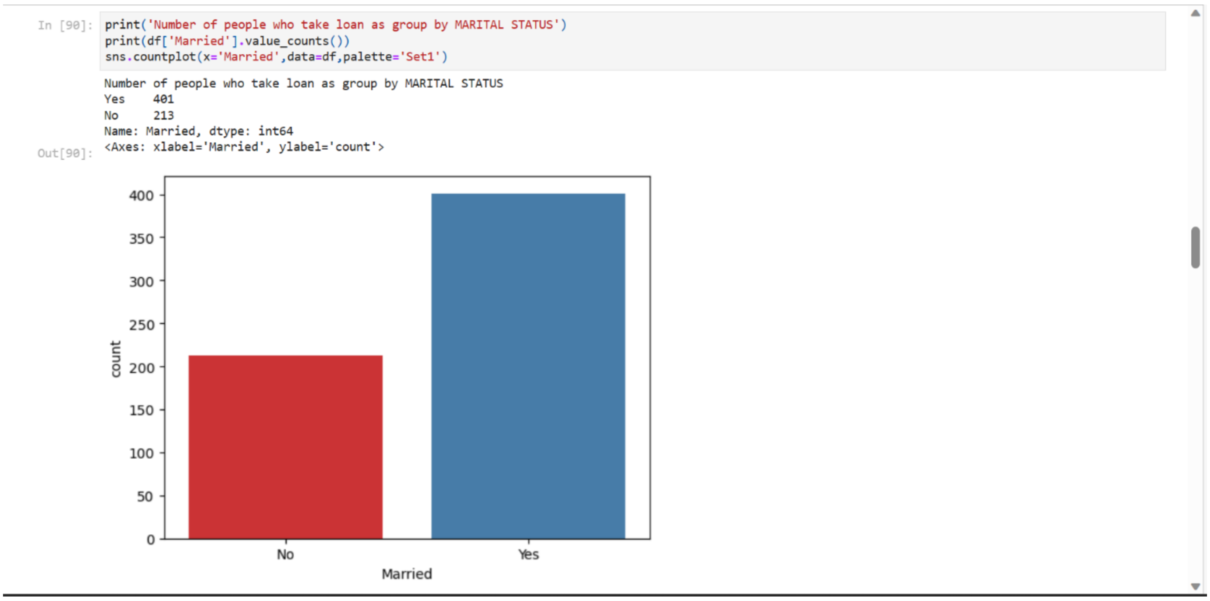


```
In [134…            #all NaN values are filled
         df.isnull().sum()

Out[134]: Loan_ID              0
          Gender               0
          Married              0
          Dependents           0
          Education            0
          Self_Employed        0
          ApplicantIncome      0
          CoapplicantIncome    0
          LoanAmount           0
          Loan_Amount_Term     0
          Credit_History       0
          Property_Area        0
          Loan_Status          0
          LoanAmount_log       0
          dtype: int64
```

Step 5:- visualizations



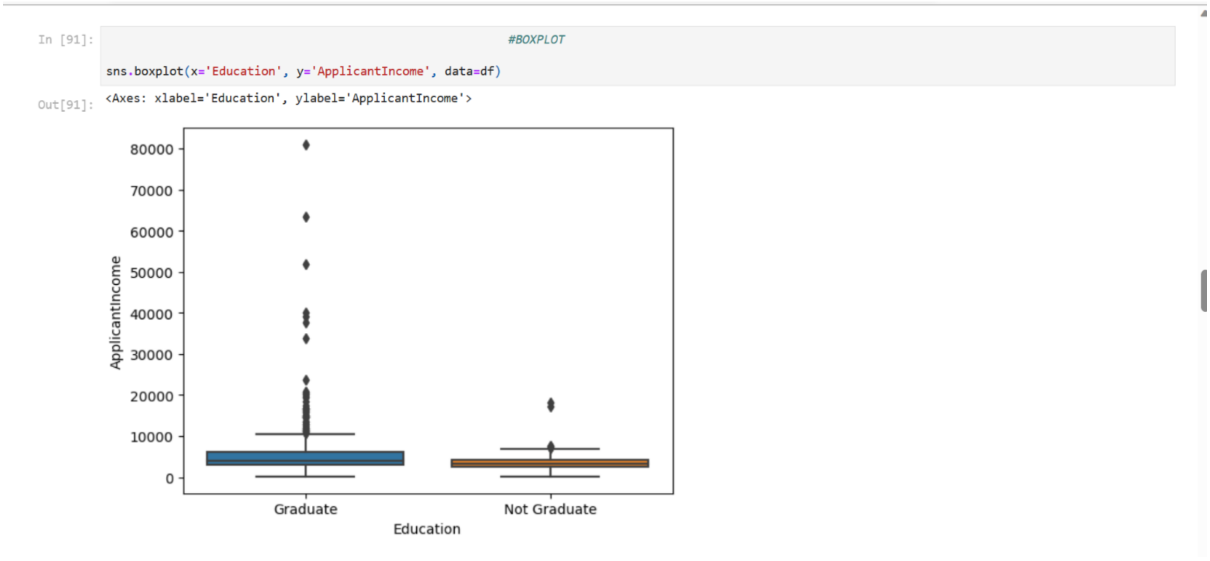
```
In [89]:                     #VISUALIZATION
            #COUNTPLOT-A countplot is a plot that displays the count of occurrences of each category in a categorical variable.
         print('Number of people who take loan as group by GENDER')
         print(df['Gender'].value_counts())
         sns.countplot(x='Gender',data=df,palette='Set1')

         Number of people who take loan as group by GENDER
         Male      502
         Female    112
         Name: Gender, dtype: int64
Out[89]: <Axes: xlabel='Gender', ylabel='count'>
```



```
In [90]: print('Number of people who take loan as group by MARITAL STATUS')
         print(df['Married'].value_counts())
         sns.countplot(x='Married',data=df,palette='Set1')
```

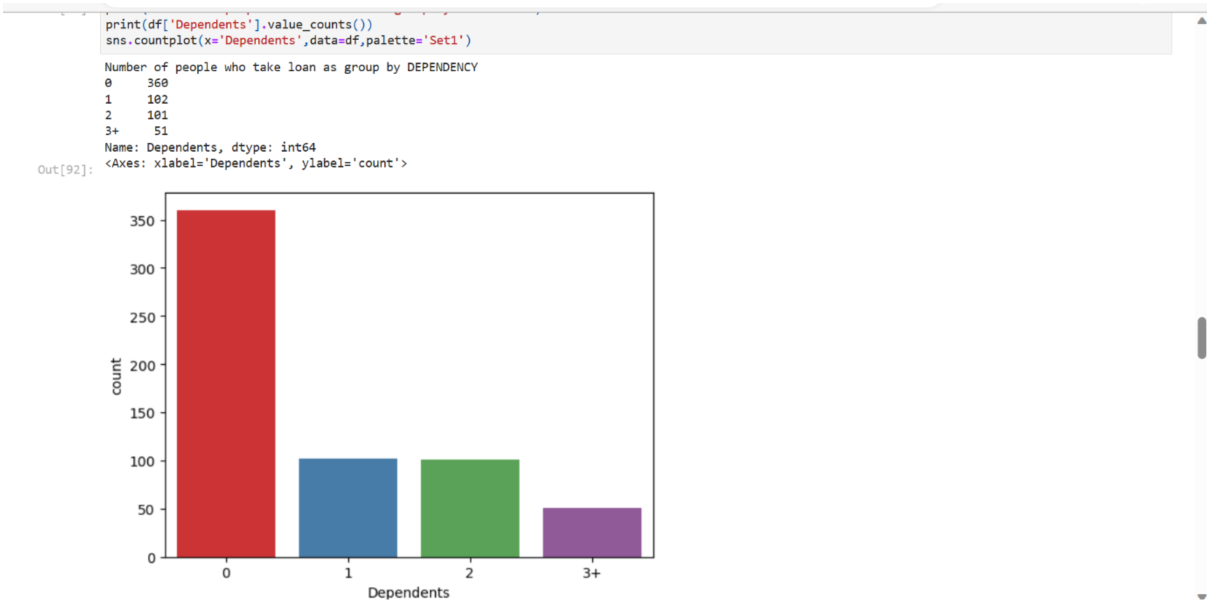
```
Number of people who take loan as group by MARITAL STATUS
Yes    401
No     213
Name: Married, dtype: int64
```

```
Out[90]: <Axes: xlabel='Married', ylabel='count'>
```



```
In [91]:                                   #BOXPLOT

         sns.boxplot(x='Education', y='ApplicantIncome', data=df)
```

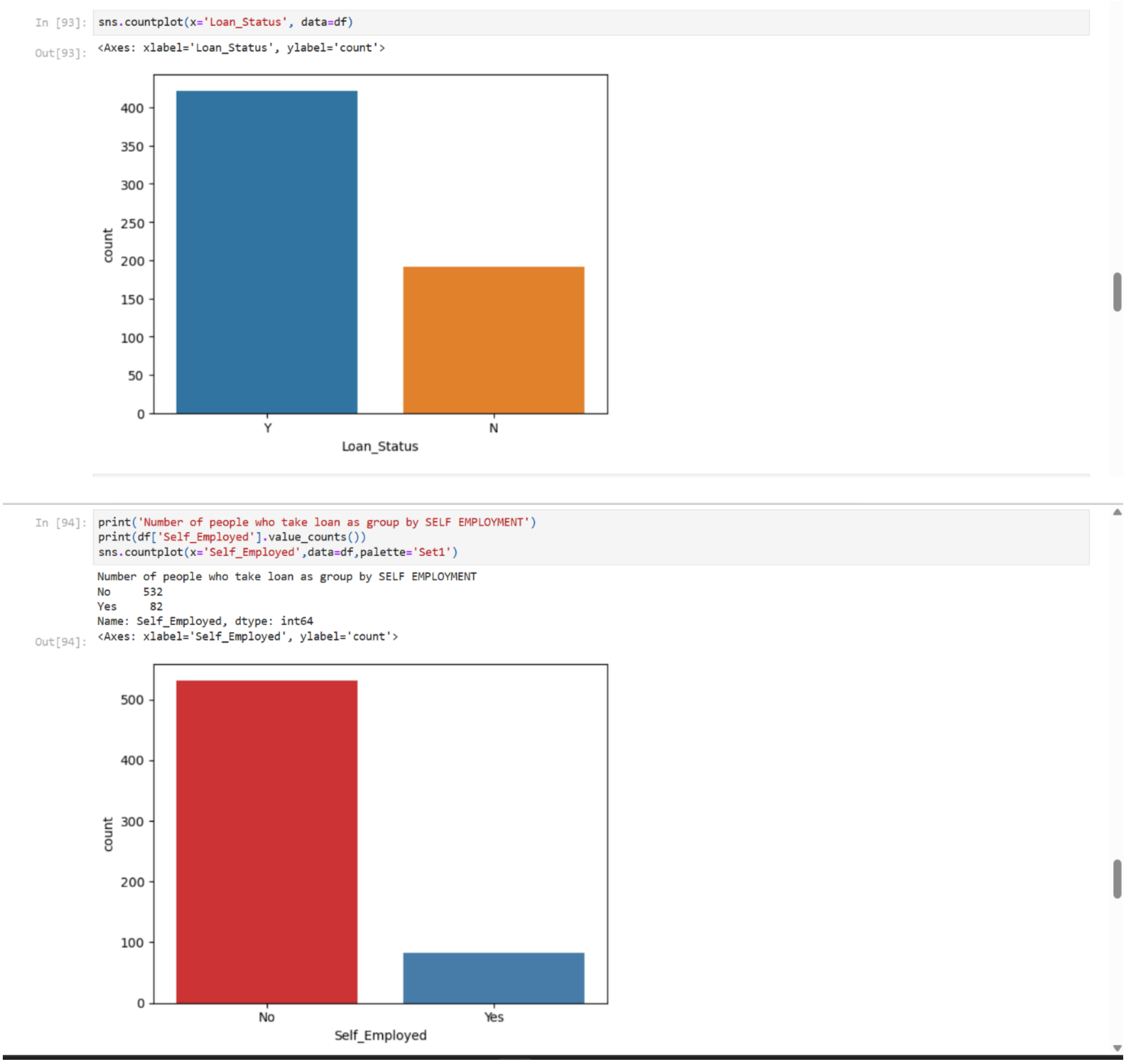
```
Out[91]: <Axes: xlabel='Education', ylabel='ApplicantIncome'>
```



```
         print(df['Dependents'].value_counts())
         sns.countplot(x='Dependents',data=df,palette='Set1')
```

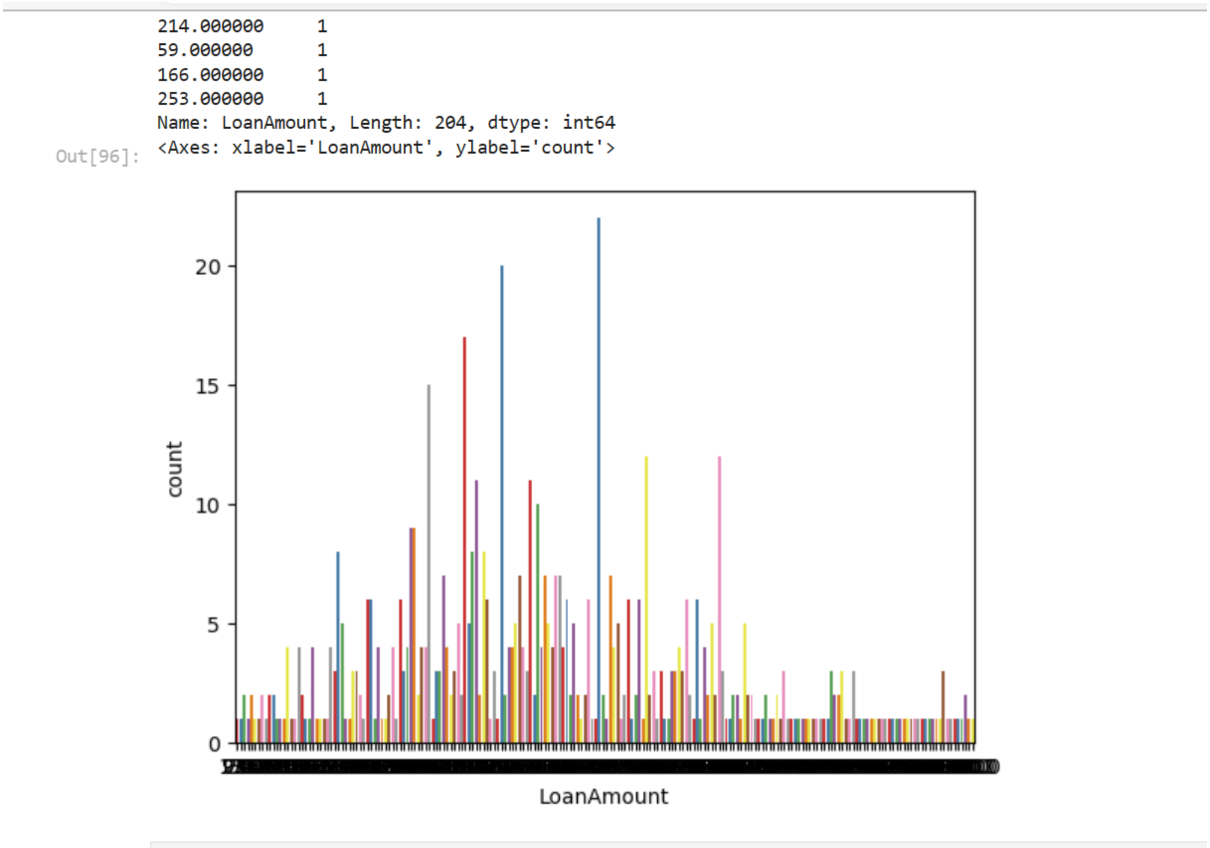
```
Number of people who take loan as group by DEPENDENCY
0     360
1     102
2     101
3+     51
Name: Dependents, dtype: int64
```

```
Out[92]: <Axes: xlabel='Dependents', ylabel='count'>
```



```
In [93]: sns.countplot(x='Loan_Status', data=df)

Out[93]: <Axes: xlabel='Loan_Status', ylabel='count'>
```

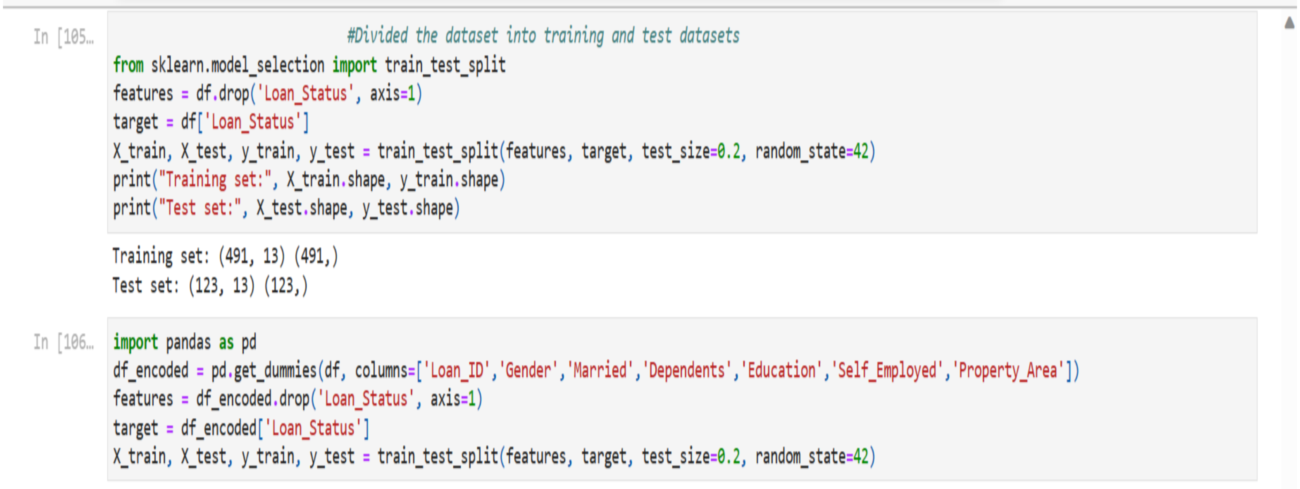
```
In [94]: print('Number of people who take loan as group by SELF EMPLOYMENT')
         print(df['Self_Employed'].value_counts())
         sns.countplot(x='Self_Employed',data=df,palette='Set1')

         Number of people who take loan as group by SELF EMPLOYMENT
         No     532
         Yes     82
         Name: Self_Employed, dtype: int64
Out[94]: <Axes: xlabel='Self_Employed', ylabel='count'>
```



```
214.000000     1
59.000000      1
166.000000     1
253.000000     1
Name: LoanAmount, Length: 204, dtype: int64
```

```
Out[96]: <Axes: xlabel='LoanAmount', ylabel='count'>
```

Step 6:- Dividing the dataset into training and testing

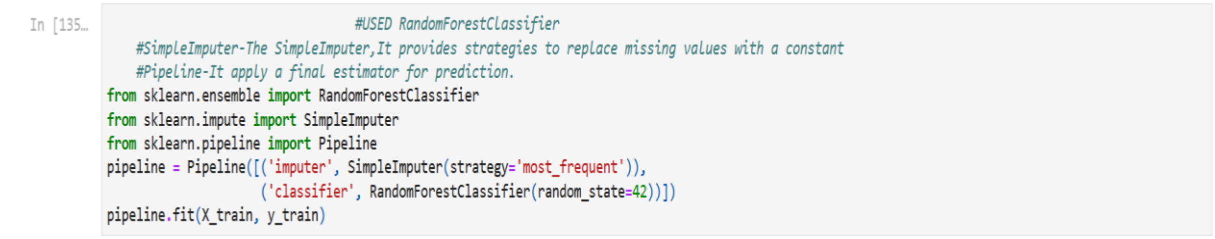


```
In [105…                          #Divided the dataset into training and test datasets
from sklearn.model_selection import train_test_split
features = df.drop('Loan_Status', axis=1)
target = df['Loan_Status']
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2, random_state=42)
print("Training set:", X_train.shape, y_train.shape)
print("Test set:", X_test.shape, y_test.shape)
```

```
Training set: (491, 13) (491,)
Test set: (123, 13) (123,)
```

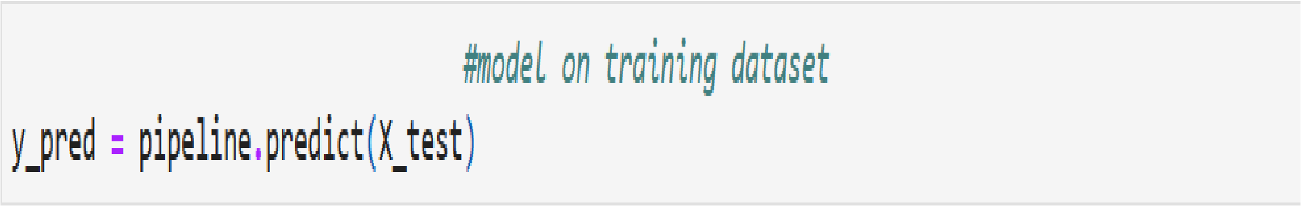
```
In [106…
import pandas as pd
df_encoded = pd.get_dummies(df, columns=['Loan_ID','Gender','Married','Dependents','Education','Self_Employed','Property_Area'])
features = df_encoded.drop('Loan_Status', axis=1)
target = df_encoded['Loan_Status']
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2, random_state=42)
```

Step 7:- Building the Machine Learning Model



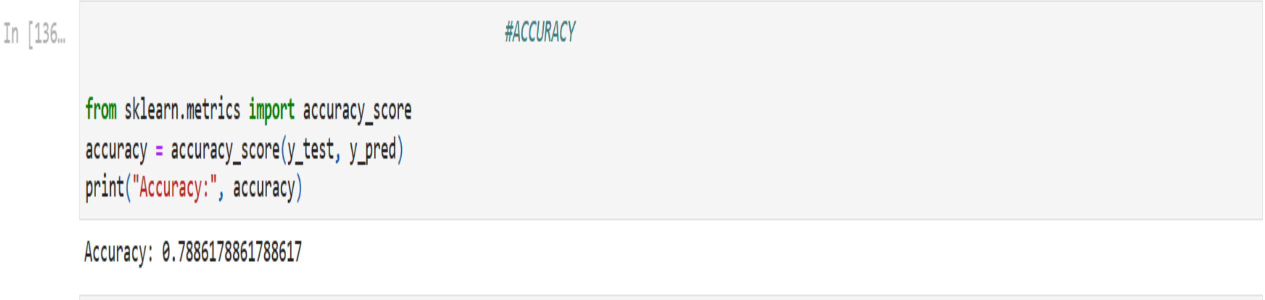
```
In [135…                                  #USED RandomForestClassifier
    #SimpleImputer-The SimpleImputer,It provides strategies to replace missing values with a constant
    #Pipeline-It apply a final estimator for prediction.
from sklearn.ensemble import RandomForestClassifier
from sklearn.impute import SimpleImputer
from sklearn.pipeline import Pipeline
pipeline = Pipeline([('imputer', SimpleImputer(strategy='most_frequent')),
                     ('classifier', RandomForestClassifier(random_state=42))])
pipeline.fit(X_train, y_train)
```

Step 8:- Fit the model on training set



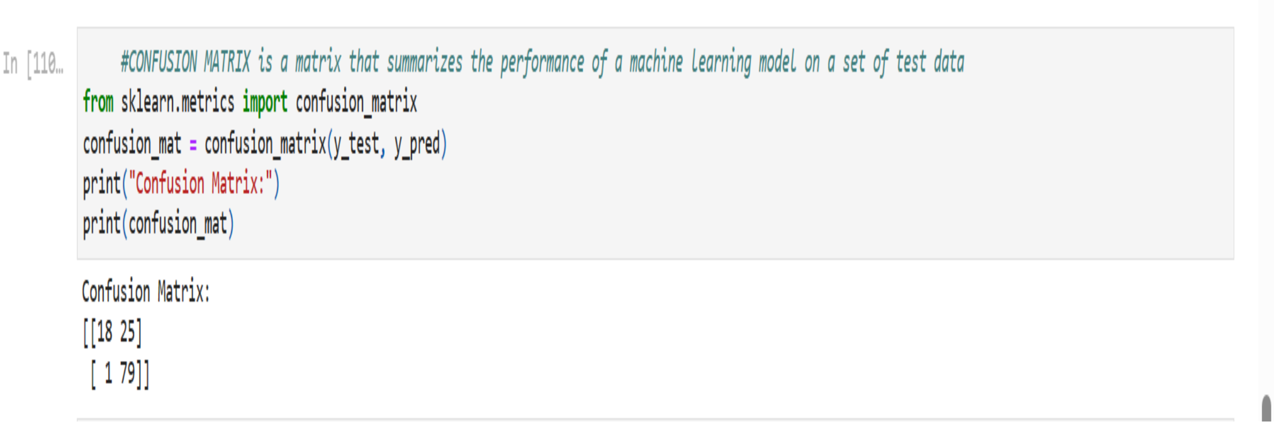
```
                    #model on training dataset
y_pred = pipeline.predict(X_test)
```

Step 9:- Accuracy of the model



```
In [136…                      #ACCURACY

      from sklearn.metrics import accuracy_score
      accuracy = accuracy_score(y_test, y_pred)
      print("Accuracy:", accuracy)

      Accuracy: 0.7886178861788617
```

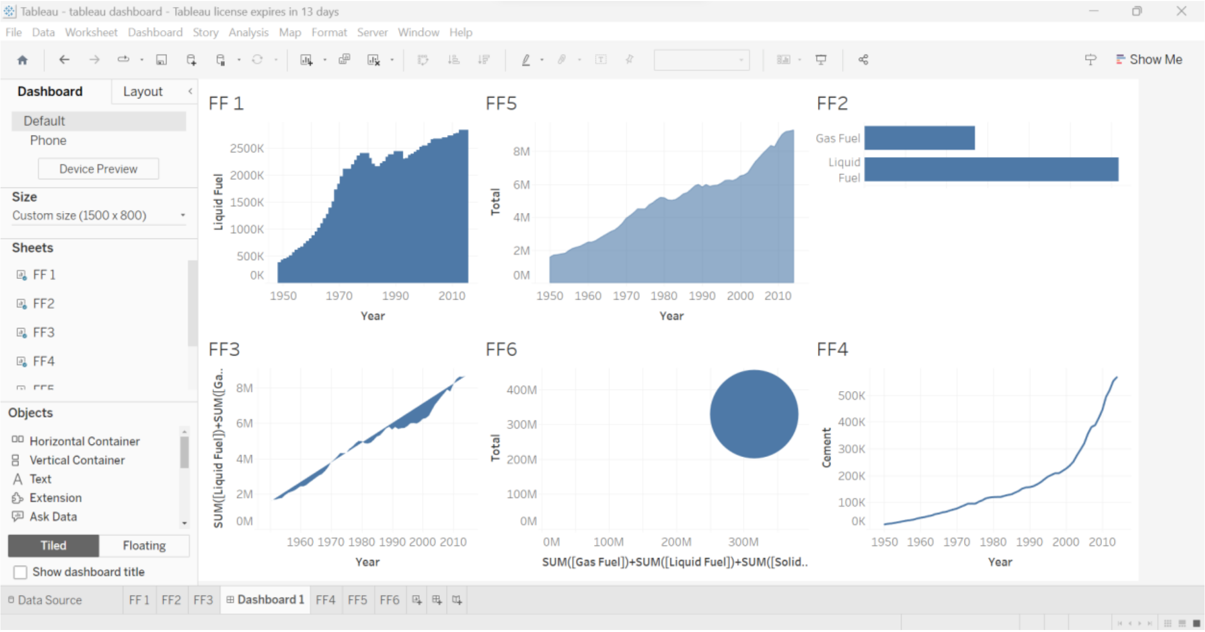
Step 10:- creating the confusion matrix

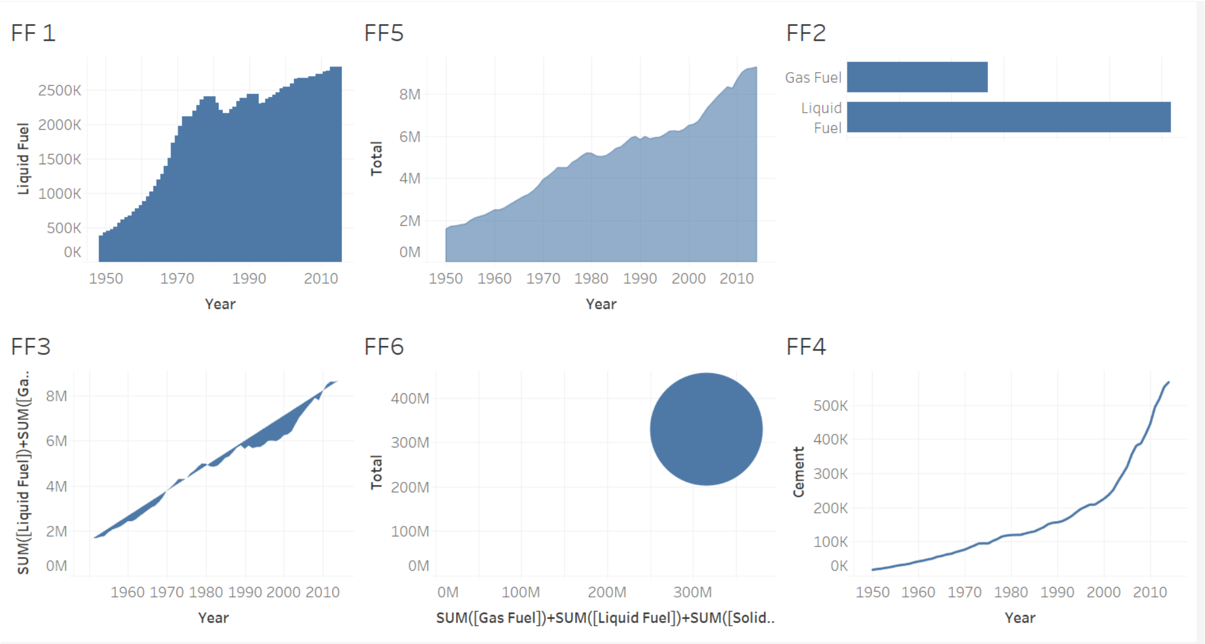


```
In [110…     #CONFUSION MATRIX is a matrix that summarizes the performance of a machine learning model on a set of test data
      from sklearn.metrics import confusion_matrix
      confusion_mat = confusion_matrix(y_test, y_pred)
      print("Confusion Matrix:")
      print(confusion_mat)

      Confusion Matrix:
      [[18 25]
       [ 1 79]]
```

TASK-2

CREATING THE TABLEAU DASHBOARD





- The above tableau dashboard is created using the fossil fuel data set.

CONCLUSION

I Just finished my first major project Loan prediction for a given dataset using machine learning models and creating tableau dashboard.

In this project, I learned how to:-

1.importing the libraries

2. importing the datasets

3.dealing with the missing values

4.doing data visualization

5.building the machine learning models

6.testing and training the dataset

7.finding accuracy

8.creating confusion matrix

9.creating the tableau dashboard

These tools will continue to help you throughout our many programming adventures.