**DS-04-MLB4**

**MAJOR PROJECT**

**LOAN PREDICTION DATASET**

*Project by* ***NIMBARK JAY DHIRENDRAKUMAR***

*Data Science (April Batch)*

**Task 1:**

You are provided with a loan sanction dataset where you will have to identify whether the loan of a particular person is approved or not depending on the information the individual has provided.

|  |  |
| --- | --- |
| The dataset consists of the following attributes: **Atrribute** | **Significance** |
| Loan\_id | Unique loan Id |
| Gender | Male/female |
| Married | Applicant married (Y/N) |
| Dependents | Number of dependents |
| Education | Applicant education (Graduate/ Under graduate) |
| Self-employed | Self employed (Y/N) |
| ApplicantIncome | Applicant income |
| CoapplicantIncome | Coapplicant income |
| LoanAmount | Loan amount in thousands |
| Loan\_Amount\_Term | Term of loan in months |
| Credit\_history | Credit history meet guidelines |
| Property\_area | Urban / Semi urban / Rural |
| Loan\_status | Loan approved (Y/N) |

**Steps for you to follow:**

* Import all the necessary libraries
* Import the dataset provided
* Understand the data
* Deal with the missing values if any
* Do some visualization if necessary
* Divide the dataset into training and test datasets
* Build the machine learning model which ever is suitable for the dataset
* Fit the model on the training dataset
* Test the model and find the accuracy of the model on the test and the training datasets
* Create a confusion matrix

At last, draw conclusions based on the dataset provided and document the same on the jupyter notebook

Most importantly, please comment the usefulness of each cell.

**Task- 2**

On the same dataset draw conclusions from the dataset and create a tableau dashboard for the same.

**Task-1**

#Import dataframe from csv file.

import pandas as pd

df = pd.read\_csv("/content/loan.csv")

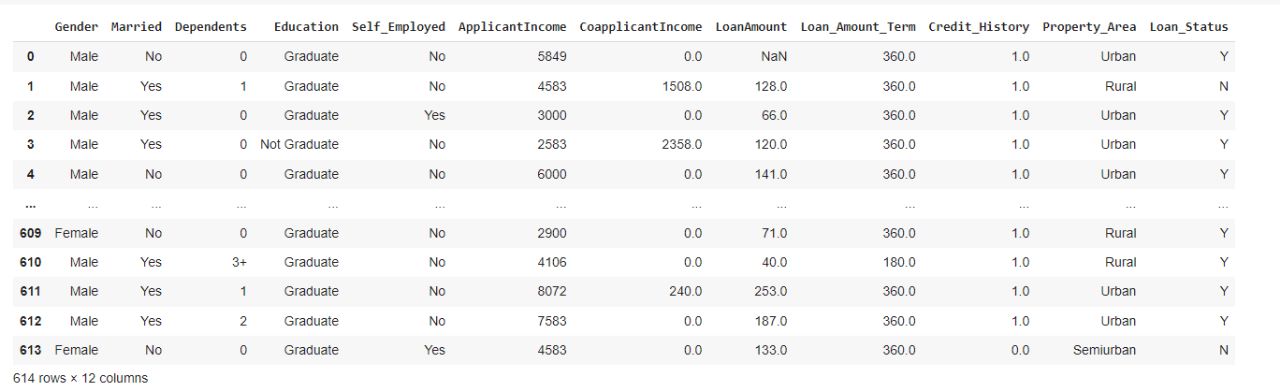
df

#DATA PRE-PROCESSING

#Remove unnecessary columns

df = df.drop(['Loan\_ID'], axis='columns')

df

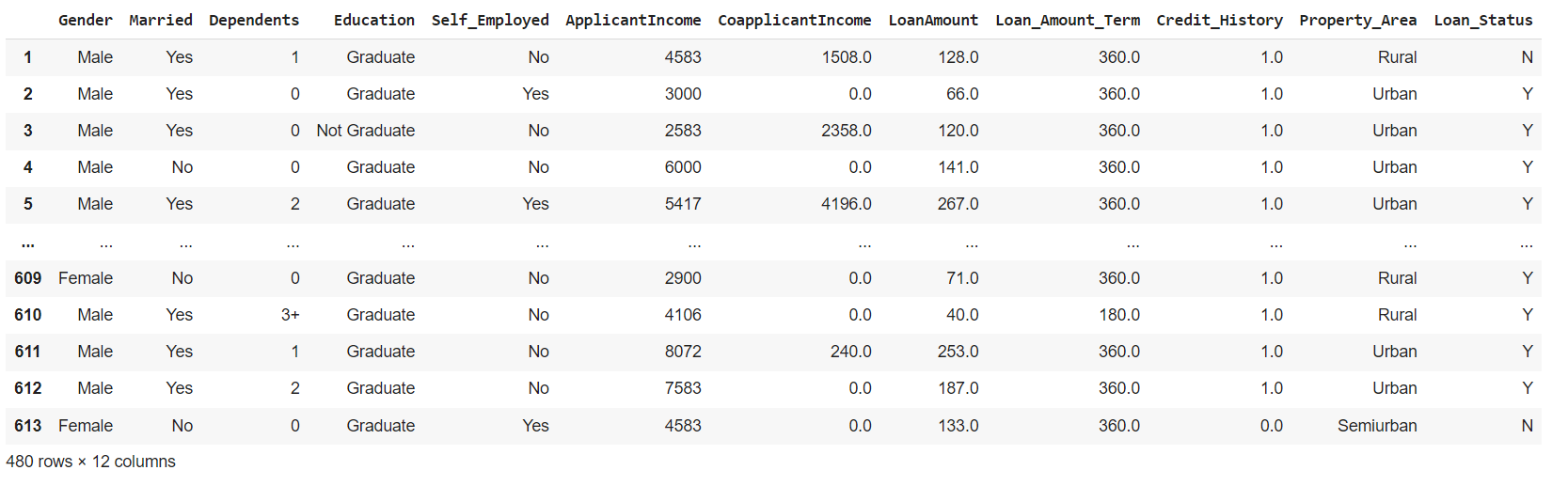


#DATA PRE-PROCESSING

#Remove rows with null values

df.dropna(inplace=True)

df



# Checking if null values still remain

df['Loan\_Status'].isnull().sum()

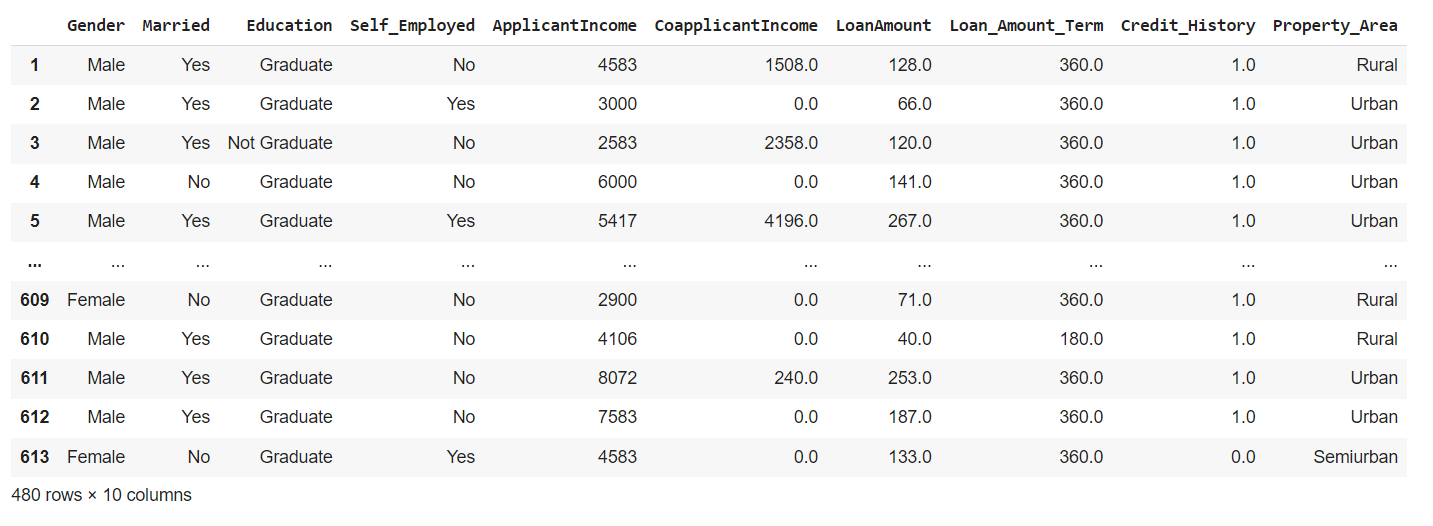
0

df = df.drop(['Dependents'],axis='columns')

# Taking every column except target column Loan\_Status as input for model.

inputs = df.drop(['Loan\_Status'], axis='columns')

inputs

****

#Label encode each value from categorical to numerical

from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

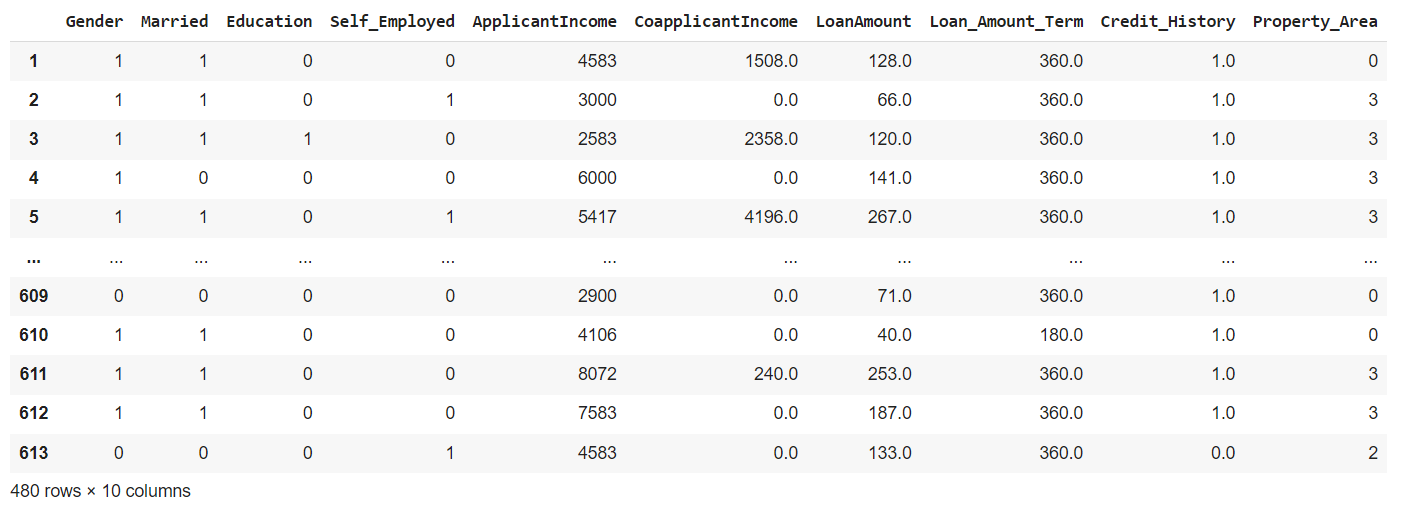
inputs['Gender'] = le.fit\_transform(inputs['Gender'])

inputs['Married'] = le.fit\_transform(inputs['Married'])

inputs['Education'] = le.fit\_transform(inputs['Education'])

inputs['Self\_Employed'] = le.fit\_transform(inputs['Self\_Employed'])

inputs['Property\_Area'] = le.fit\_transform(inputs['Property\_Area'])

****

# Loan\_Status column in target variable. It is then label encoded.

target = df.drop(inputs, axis='columns')

target['Loan\_Status'] = le.fit\_transform(target['Loan\_Status'])

target

| **Loan\_Status** |
| --- |
| **1** | 0 |
| **2** | 1 |
| **3** | 1 |
| **4** | 1 |
| **5** | 1 |
| **...** | ... |
| **609** | 1 |
| **610** | 1 |
| **611** | 1 |
| **612** | 1 |
| **613** | 0 |

480 rows × 1 columns

# Importing accuracy score libraries and setting test\_size to 20%.

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import accuracy\_score

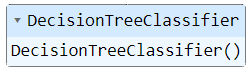
inputs\_train, inputs\_test, target\_train, target\_test = train\_test\_split(inputs,target,test\_size=0.2)

from sklearn import tree

model = tree.DecisionTreeClassifier()

# Fitting our training data to DecisionTreeClassifier.

model.fit(inputs\_train, target\_train)

****

# Testing model accuracy score by putting against previously unseen testing data.

model.score(inputs\_test,target\_test)

0.71875

# Creating confusion matrix.

from sklearn.metrics import confusion\_matrix

target\_pred= model.predict(inputs\_test)

cm = confusion\_matrix(target\_test, target\_pred)

cm

array([[20, 8], [19, 49]])

# **Task-2** Sheet 1

Loan Status

40K

20K

100K

50K

Sheet 2

0

Dependents

No Yes

1500K

1000K

500K

0K

400K

200K

0K

Null

1

2

3+

Sheet-3

Property Area Rural Semiurban Urban

Education / Gender / Property Area

Graduate

Not Graduate

Null Female Male N.. Female Male

60

Applicant Income

Credit History

40

20

Coapplicant Income

0

Rural Semiurban

Urban Rural Semiurban

Urban Rural Semiurban

Urban Rural Rural Semiurban

Urban Rural Semiurban

Urban