In [1]:
```
import pandas as pd
```

In [2]:
```
data=pd.read_csv('train_u6lujuX_CVtuZ9i[1].csv')
```

In [3]:
```
data.tail()
```

Out[3]:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	L
609	LP002978	Female	No	0	Graduate	No	2900	0.0	71.0	360.0	1.0	Rural	
610	LP002979	Male	Yes	3+	Graduate	No	4106	0.0	40.0	180.0	1.0	Rural	
611	LP002983	Male	Yes	1	Graduate	No	8072	240.0	253.0	360.0	1.0	Urban	
612	LP002984	Male	Yes	2	Graduate	No	7583	0.0	187.0	360.0	1.0	Urban	
613	LP002990	Female	No	0	Graduate	Yes	4583	0.0	133.0	360.0	0.0	Semiurban	

In [4]:
```
data.head()
```

Out[4]:

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loa
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	

In [5]:
```
data.shape
```

Out[5]:
```
(614, 13)
```

In [6]:
```
data.info()
```

Out[6]:
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Loan_ID            614 non-null    object 
 1   Gender             601 non-null    object 
 2   Married            611 non-null    object 
 3   Dependents         599 non-null    object 
 4   Education          614 non-null    object 
 5   Self_Employed      582 non-null    object 
 6   ApplicantIncome    614 non-null    int64  
 7   CoapplicantIncome  614 non-null    float64
 8   LoanAmount         592 non-null    float64
 9   Loan_Amount_Term   600 non-null    float64
 10  Credit_History     564 non-null    float64
 11  Property_Area      614 non-null    object 
 12  Loan_Status        614 non-null    object 
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
```

In [7]:
```
data.isnull().sum()
```

Out[7]:
```
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 [8]:
```
data.isnull().sum()*100/len(data)
```

Out[8]:
```
Loan_ID               0.000000
Gender                2.117264
Married               0.488599
Dependents            2.442997
Education             0.000000
Self_Employed         5.211726
ApplicantIncome       0.000000
CoapplicantIncome     0.000000
LoanAmount            3.583062
Loan_Amount_Term      2.280130
Credit_History        8.143322
Property_Area         0.000000
Loan_Status           0.000000
dtype: float64
```

In [9]:
```
data = data.drop('Loan_ID', axis=1)
```

In [10]:
```
data.head(1)
```

Out[10]:

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y

In [11]:
```
columns=['Gender','Dependents','LoanAmount','Loan_Amount_Term']
```

In [12]:
```
data=data.dropna(subset=columns)
```

In [13]:
```
data.isnull().sum()*100/len(data)
```

Out[13]:
```
Gender               0.000000
Married              0.000000
Dependents           0.000000
Education            0.000000
Self_Employed        5.424955
ApplicantIncome      0.000000
CoapplicantIncome    0.000000
LoanAmount           0.000000
Loan_Amount_Term     0.000000
Credit_History       8.679928
Property_Area        0.000000
Loan_Status          0.000000
dtype: float64
```

In [14]:
```
data['Self_Employed'].mode()[0]
```

Out[14]:
```
'No'
```

In [15]:
```
data['Self_Employed'] = data['Self_Employed'].fillna(data['Self_Employed'].mode()[0])
```

In [16]:
```
data.isnull().sum()*100/len(data)
```

Out[16]:
```
Gender               0.000000
Married              0.000000
Dependents           0.000000
Education            0.000000
Self_Employed        0.000000
ApplicantIncome      0.000000
CoapplicantIncome    0.000000
LoanAmount           0.000000
Loan_Amount_Term     0.000000
Credit_History       8.679928
Property_Area        0.000000
Loan_Status          0.000000
dtype: float64
```

In [17]:
```
data['Credit_History'].mode()[0]
```

Out[17]:
```
1.0
```

In [18]:
```
data['Credit_History'] = data['Credit_History'].fillna(data['Credit_History'].mode()[0])
```

In [19]:
```
data.isnull().sum()*100/len(data)
```

Out[19]:
```
Gender               0.0
Married              0.0
Dependents           0.0
Education            0.0
Self_Employed        0.0
ApplicantIncome      0.0
CoapplicantIncome    0.0
LoanAmount           0.0
Loan_Amount_Term     0.0
Credit_History       0.0
Property_Area        0.0
Loan_Status          0.0
dtype: float64
```

In [20]:
```
data.sample(18)
```

Out[20]:

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
29	Female	No	2	Graduate	No	3750	2083.0	120.0	360.0	1.0	Semiurban	Y
567	Male	Yes	3+	Graduate	No	3400	2500.0	123.0	360.0	0.0	Rural	N
205	Female	No	0	Not Graduate	No	4408	0.0	120.0	360.0	1.0	Semiurban	Y
369	Male	Yes	0	Graduate	No	19730	5266.0	570.0	360.0	1.0	Rural	N
548	Female	No	0	Graduate	No	5000	0.0	103.0	360.0	0.0	Semiurban	N
3	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
578	Male	Yes	1	Graduate	No	1782	2232.0	107.0	360.0	1.0	Rural	Y
424	Male	Yes	0	Graduate	No	14880	0.0	96.0	360.0	1.0	Semiurban	Y
222	Male	No	0	Graduate	No	2971	2791.0	144.0	360.0	1.0	Semiurban	Y
96	Female	Yes	0	Graduate	No	2484	2302.0	137.0	360.0	1.0	Semiurban	Y
260	Male	Yes	1	Graduate	No	6083	4250.0	330.0	360.0	1.0	Urban	Y
257	Male	Yes	3+	Graduate	No	5250	0.0	94.0	360.0	1.0	Urban	N
274	Male	Yes	2	Graduate	No	3900	0.0	90.0	360.0	1.0	Semiurban	Y
116	Female	Yes	0	Graduate	No	3167	2283.0	154.0	360.0	1.0	Semiurban	Y
347	Male	Yes	2	Not Graduate	No	3083	2168.0	126.0	360.0	1.0	Urban	Y
416	Female	No	1	Graduate	No	2600	0.0	160.0	360.0	1.0	Urban	N
155	Male	Yes	3+	Graduate	No	39999	0.0	600.0	180.0	0.0	Semiurban	Y
599	Male	Yes	2	Graduate	No	5780	0.0	192.0	360.0	1.0	Urban	Y

In [21]:
```
data['Dependents']=data['Dependents'].replace(to_replace='3+',value='4')
```

In [22]:
```
data['Dependents'].unique()
```

Out[22]:
```
array(['1', '0', '2', '4'], dtype=object)
```

In [23]:
```
data['Gender'].unique()
data['Married'].unique()
data['Education'].unique()
data['Self_Employed'].unique()
data['Property_Area'].unique()

data['Loan_Status'].unique()
```

Out[23]:
```
array(['N', 'Y'], dtype=object)
```

In [24]:
```
data['Property_Area'].unique()
```

Out[24]:
```
array(['Rural', 'Urban', 'Semiurban'], dtype=object)
```

In [25]:
```
data['Gender']=data['Gender'].map({'Male':1 ,'Female':0}).astype('int')
data['Married']=data['Married'].map({'Yes':1 ,'No':0}).astype('int')
data['Education']=data['Education'].map({'Graduate':1 ,'Not Graduate':0}).astype('int')
data['Self_Employed']=data['Self_Employed'].map({'No':1 ,'Yes':0}).astype('int')
data['Loan_Status']=data['Loan_Status'].map({'N':1 ,'Y':0,}).astype('int')
data['Property_Area']=data['Property_Area'].map({'Rural':0, 'Urban':1, 'Semiurban':2}).astype('int')
```

In [26]:
```
data['Gender'].unique()
```

Out[26]:
```
array([1, 0])
```

In [27]:
```
X = data.drop('Loan_Status', axis=1)
```

In [28]:
```
y = data['Loan_Status']
y
```

Out[28]:
```
1      1
2      0
3      0
4      0
5      0
      ..
609    0
610    0
611    0
612    0
613    1
Name: Loan_Status, Length: 553, dtype: int32
```

In [29]:
```
data.head()
```

Out[29]:

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
1	1	1	1	1	1	4583	1508.0	128.0	360.0	1.0	0	1
2	1	1	0	1	0	3000	0.0	66.0	360.0	1.0	1	0
3	1	1	0	0	1	2583	2358.0	120.0	360.0	1.0	1	0
4	1	0	0	1	1	6000	0.0	141.0	360.0	1.0	1	0
5	1	1	2	1	0	5417	4196.0	267.0	360.0	1.0	1	0

In [30]:
```
colm=['ApplicantIncome','CoapplicantIncome','LoanAmount','Loan_Amount_Term']
```

In [31]:
```
from sklearn.preprocessing import StandardScaler
st= StandardScaler()
(X[colm])=st.fit_transform(X[colm])
```

In [32]:
```
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import accuracy_score
import numpy as np
```

In [33]:
```
model_df = {}

def model_val(model, X, y):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=42)

    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    print(f"{model} accuracy is {accuracy}")

    scores = cross_val_score(model, X, y, cv=5)
    print(f"{model} Avg cross val score is {np.mean(scores)}")

    model_df[model] = round(np.mean(scores) * 100, 2)
```

In [34]:
```
data.head()
```

Out[34]:

	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
1	1	1	1	1	1	4583	1508.0	128.0	360.0	1.0	0	1
2	1	1	0	1	0	3000	0.0	66.0	360.0	1.0	1	0
3	1	1	0	0	1	2583	2358.0	120.0	360.0	1.0	1	0
4	1	0	0	1	1	6000	0.0	141.0	360.0	1.0	1	0
5	1	1	2	1	0	5417	4196.0	267.0	360.0	1.0	1	0

In [36]:
```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
model_val(model, X, y)
```

```
LogisticRegression() accuracy is 0.8018018018018018
LogisticRegression() Avg cross val score is 0.8047829647829647
```

In [ ]: