

```
In [66]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [67]: from sklearn.linear_model import LogisticRegression
```

```
In [115]: df=pd.read_csv(r"C:\Users\Admin\Downloads\C8_loan-test - C8_loan-test.csv")
df
```

Out[115]:

| Education | Self_Employed | ApplicantIncome | CoapplicantIncome | LoanAmount | Loan_Amount_Term |
|--------------|---------------|-----------------|-------------------|------------|------------------|
| Graduate | No | 5720 | 0 | 110.0 | 360.0 |
| Graduate | No | 3076 | 1500 | 126.0 | 360.0 |
| Graduate | No | 5000 | 1800 | 208.0 | 360.0 |
| Graduate | No | 2340 | 2546 | 100.0 | 360.0 |
| Not Graduate | No | 3276 | 0 | 78.0 | 360.0 |
| ... | ... | ... | ... | ... | ... |
| Not Graduate | Yes | 4009 | 1777 | 113.0 | 360.0 |
| Graduate | No | 4158 | 709 | 115.0 | 360.0 |
| Graduate | No | 3250 | 1993 | 126.0 | 360.0 |
| Graduate | No | 5000 | 2393 | 158.0 | 360.0 |
| Graduate | Yes | 9200 | 0 | 98.0 | 180.0 |

```
In [116]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 367 entries, 0 to 366
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype  
---  -
 0   Loan_ID               367 non-null   object 
 1   Gender                356 non-null   object 
 2   Married              367 non-null   object 
 3   Dependents           357 non-null   object 
 4   Education             367 non-null   object 
 5   Self_Employed        344 non-null   object 
 6   ApplicantIncome      367 non-null   int64  
 7   CoapplicantIncome    367 non-null   int64  
 8   LoanAmount           362 non-null   float64 
 9   Loan_Amount_Term     361 non-null   float64 
10  Credit_History        338 non-null   float64 
11  Property_Area         367 non-null   object 
dtypes: float64(3), int64(2), object(7)
memory usage: 34.5+ KB
```

```
In [117]: df1=df.dropna()
```

```
In [128]: d1=df1.iloc[:,7:11]
          d2=df1.iloc[:, -2]
```

```
In [129]: d1.shape
```

```
Out[129]: (289, 4)
```

```
In [130]: d2.shape
```

```
Out[130]: (289,)
```

```
In [131]: from sklearn.preprocessing import StandardScaler
```

```
In [132]: a=StandardScaler().fit_transform(d1)
```

```
In [133]: lr=LogisticRegression()
          lr.fit(a,d2)
```

```
Out[133]: LogisticRegression()
```

```
In [138]: obs=[[10,12,22,33]]
```

```
In [139]: pdt=lr.predict(obs)
          print(pdt)
```

```
[1.]
```

```
In [140]: lr.classes_
```

```
Out[140]: array([0., 1.])
```

```
In [141]: lr.predict_proba(obs)[0][0]
```

```
Out[141]: 0.0
```

```
In [142]: lr.predict_proba(obs)[0][1]
```

```
Out[142]: 1.0
```

```
In [ ]:
```

```
In [ ]:
```