**TITLE OF PROJECT: HILL AND VALLEY PRDICTION USING LOGISTICS REGRESSION**

**OBJECTIVE:HEAR THE DATE IS TRAIN,TEST AND DATA PREPROSSING HAVE BEEN DONE**

**DATA SOURCE:GET HUB**

**IMPORT LIBRARY**

```
import pandas as pd

import numpy as np
```

**IMPORT DATASET**

```
df=pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/
Hill%20Valley%20Dataset.csv')

df

            V1        V2        V3        V4        V5        V6
V7  \
0        39.02     36.49     38.20     38.85     39.38     39.74
37.02
1         1.83      1.71      1.77      1.77      1.68      1.78
1.80
2     68177.69  66138.42  72981.88  74304.33  67549.66  69367.34
69169.41
3     44889.06  39191.86  40728.46  38576.36  45876.06  47034.00
46611.43
4         5.70      5.40      5.28      5.38      5.27      5.61
6.00
...        ...       ...       ...       ...       ...       ...
...
1207     13.00     12.87     13.27     13.04     13.19     12.53
14.31
1208     48.66     50.11     48.55     50.43     50.09     49.67
48.95
1209  10160.65   9048.63   8994.94   9514.39   9814.74  10195.24
10031.47
1210     34.81     35.07     34.98     32.37     34.16     34.03
33.31
1211   8489.43   7672.98   9132.14   7985.73   8226.85   8554.28
8838.87

            V8        V9       V10  ...       V92       V93       V94
\
0        39.53     38.81     38.79  ...     36.62     36.92     38.80

1         1.70      1.75      1.78  ...      1.80      1.79      1.77
```

```
2     73268.61  74465.84  72503.37  ...  73438.88  71053.35  71112.62

3     37668.32  40980.89  38466.15  ...  42625.67  40684.20  46960.73

4         5.38      5.34      5.87  ...      5.17      5.67      5.60

...        ...       ...       ...  ...       ...       ...       ...

1207     13.33     13.63     14.55  ...     12.48     12.15     13.15

1208     48.65     48.63     48.61  ...     46.93     49.61     47.16

1209  10202.28   9152.99   9591.75  ...   9068.11   9191.80   9275.04

1210     32.48     35.63     32.48  ...     32.76     35.03     32.89

1211   8967.24   8635.14   8544.37  ...   8609.73   9209.48   8496.33


           V95       V96       V97       V98       V99      V100
Class
0        38.52     38.07     36.73     39.46     37.50     39.10
0
1         1.74      1.74      1.80      1.78      1.75      1.69
1
2     74916.48  72571.58  66348.97  71063.72  67404.27  74920.24
1
3     44546.80  45410.53  47139.44  43095.68  40888.34  39615.19
0
4         5.94      5.73      5.22      5.30      5.73      5.91
0
...        ...       ...       ...       ...       ...       ...    ..
.
1207     12.35     13.58     13.86     12.88     13.87     13.51
1
1208     48.17     47.94     49.81     49.89     47.43     47.77
0
1209   9848.18   9074.17   9601.74  10366.24   8997.60   9305.77
1
1210     31.91     33.85     35.28     32.49     32.83     34.82
1
1211   8724.01   8219.99   8550.86   8679.43   8389.31   8712.80
0

[1212 rows x 101 columns]

df.head()

         V1        V2        V3        V4        V5        V6
V7  \
0     39.02     36.49     38.20     38.85     39.38     39.74
```

```
37.02
1      1.83      1.71      1.77      1.77      1.68      1.78  
1.80
2  68177.69  66138.42  72981.88  74304.33  67549.66  69367.34  
69169.41
3  44889.06  39191.86  40728.46  38576.36  45876.06  47034.00  
46611.43
4      5.70      5.40      5.28      5.38      5.27      5.61  
6.00

         V8        V9       V10  ...       V92       V93       V94  
V95  \
0     39.53     38.81     38.79  ...     36.62     36.92     38.80  
38.52
1      1.70      1.75      1.78  ...      1.80      1.79      1.77  
1.74
2  73268.61  74465.84  72503.37  ...  73438.88  71053.35  71112.62  
74916.48
3  37668.32  40980.89  38466.15  ...  42625.67  40684.20  46960.73  
44546.80
4      5.38      5.34      5.87  ...      5.17      5.67      5.60  
5.94

        V96       V97       V98       V99      V100  Class
0     38.07     36.73     39.46     37.50     39.10      0
1      1.74      1.80      1.78      1.75      1.69      1
2  72571.58  66348.97  71063.72  67404.27  74920.24      1
3  45410.53  47139.44  43095.68  40888.34  39615.19      0
4      5.73      5.22      5.30      5.73      5.91      0

[5 rows x 101 columns]
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1212 entries, 0 to 1211
Columns: 101 entries, V1 to Class
dtypes: float64(100), int64(1)
memory usage: 956.5 KB
```

```
df.describe()
```

```
                  V1            V2            V3            V4  \
count    1212.000000   1212.000000   1212.000000   1212.000000   
mean     8169.091881   8144.306262   8192.653738   8176.868738   
std     17974.950461  17881.049734  18087.938901  17991.903982   
min         0.920000      0.900000      0.850000      0.890000   
25%        19.602500     19.595000     18.925000     19.277500   
50%       301.425000    295.205000    297.260000    299.720000   
75%      5358.795000   5417.847500   5393.367500   5388.482500   
```

```
max    117807.870000  108896.480000  119031.350000  110212.590000

                  V5             V6             V7             V8  \
count    1212.000000    1212.000000    1212.000000    1212.000000
mean     8128.297211    8173.030008    8188.582748    8183.641543
std     17846.757963   17927.114105   18029.562695   18048.582159
min         0.880000       0.860000       0.870000       0.650000
25%        19.210000      19.582500      18.690000      19.062500
50%       295.115000     294.380000     295.935000     290.850000
75%      5321.987500    5328.040000    5443.977500    5283.655000
max    113000.470000  116848.390000  115609.240000  118522.320000

                  V9            V10  ...            V92            V93
\
count    1212.000000    1212.000000  ...    1212.000000    1212.000000

mean     8154.670066    8120.767574  ...    8120.056815    8125.917409

std     17982.390713   17900.798206  ...   17773.190621   17758.182403

min         0.650000       0.620000  ...       0.870000       0.900000

25%        19.532500      19.285000  ...      19.197500      18.895000

50%       294.565000     295.160000  ...     297.845000     295.420000

75%      5378.180000    5319.097500  ...    5355.355000    5386.037500

max    112895.900000  117798.300000  ...  113858.680000  112948.830000


                 V94            V95            V96            V97  \
count    1212.000000    1212.000000    1212.000000    1212.000000
mean     8158.793812    8140.885421    8213.480611    8185.594002
std     17919.510371   17817.945646   18016.445265   17956.084223
min         0.870000       0.880000       0.890000       0.890000
25%        19.237500      19.385000      19.027500      19.135000
50%       299.155000     293.355000     301.370000     296.960000
75%      5286.385000    5345.797500    5300.890000    5361.047500
max    112409.570000  112933.730000  112037.220000  115110.420000

                 V98            V99           V100        Class
count    1212.000000    1212.000000    1212.000000  1212.000000
mean     8140.195355    8192.960891    8156.197376     0.500000
std     17768.356106   18064.781479   17829.310973     0.500206
min         0.860000       0.910000       0.890000     0.000000
25%        19.205000      18.812500      19.145000     0.000000
50%       300.925000     299.200000     302.275000     0.500000
75%      5390.850000    5288.712500    5357.847500     1.000000
max    116431.960000  113291.960000  114533.760000     1.000000
```

```
[8 rows x 101 columns]
```

```
df.columns.tolist()
```

```
['V1',
 'V2',
 'V3',
 'V4',
 'V5',
 'V6',
 'V7',
 'V8',
 'V9',
 'V10',
 'V11',
 'V12',
 'V13',
 'V14',
 'V15',
 'V16',
 'V17',
 'V18',
 'V19',
 'V20',
 'V21',
 'V22',
 'V23',
 'V24',
 'V25',
 'V26',
 'V27',
 'V28',
 'V29',
 'V30',
 'V31',
 'V32',
 'V33',
 'V34',
 'V35',
 'V36',
 'V37',
 'V38',
 'V39',
 'V40',
 'V41',
 'V42',
 'V43',
 'V44',
 'V45',
 'V46',
 'V47',
 'V48',
 'V49',
 'V50',
 'V51',
 'V52',
 'V53',
 'V54',
 'V55',
 'V56',
 'V57',
 'V58',
 'V59',
 'V60',
 'V61',
 'V62',
 'V63',
 'V64',
 'V65',
 'V66',
 'V67',
 'V68',
 'V69',
 'V70',
 'V71',
 'V72',
 'V73',
 'V74',
 'V75',
 'V76',
 'V77',
 'V78',
 'V79',
 'V80',
 'V81',
 'V82',
 'V83',
 'V84',
 'V85',
 'V86',
 'V87',
 'V88',
 'V89',
 'V90',
 'V91',
 'V92',
 'V93',
 'V94',
 'V95',
 'V96',
 'V97',
 'V98',
 'V99',
 'V100',
 'Class']
```

```
df.shape
```

```
(1212, 101)
```

### GET UNIQUE VALUES(CLASS OR LABEL) IN Y VARIABLE

```
df['Class'].value_counts()
```

```
0    606
1    606
Name: Class, dtype: int64
```

```
df.groupby('Class').mean()
```

```
                V1           V2           V3           V4           V5
\
Class

0      7913.333251  7825.339967  7902.497294  7857.032079  7775.610198

1      8424.850512  8463.272558  8482.810182  8496.705396  8480.984224


                V6           V7           V8           V9          V10
...  \
Class
...
0      7875.436337  7804.166584  7722.324802  7793.328416  7686.782046
...
1      8470.623680  8572.998911  8644.958284  8516.011716  8554.753102
...

               V91          V92          V93          V94          V95
\
Class

0      7753.427244  7737.843366  7799.332079  7825.211700  7791.354010

1      8478.513399  8502.270264  8452.502739  8492.375924  8490.416832


               V96          V97          V98          V99         V100
```

```
Class

0       7927.237112  7874.502343  7844.227459  7875.338713  7855.181172

1       8499.724109  8496.685660  8436.163251  8510.583069  8457.213581


[2 rows x 100 columns]
```

**DEFINE Y(TARGET VARIABLE) AND X(INDEPENDENT VARIABLE)**

```
y=df['Class']

y.shape

(1212,)

y

0       0
1       1
2       1
3       0
4       0
       ..
1207    1
1208    0
1209    1
1210    1
1211    0
Name: Class, Length: 1212, dtype: int64

x=df[['V1',
 'V2',
 'V3',
 'V4',
 'V5',
 'V6',
 'V7',
 'V8',
 'V9',
 'V10',
 'V11',
 'V12',
 'V13',
 'V14',
 'V15',
 'V16',
 'V17',
 'V18',
 'V19',
 'V20',
 'V21',
 'V22',
 'V23',
 'V24',
 'V25',
 'V26',
 'V27',
 'V28',
 'V29',
 'V30',
 'V31',
 'V32',
 'V33',
 'V34',
 'V35',
 'V36',
 'V37',
 'V38',
 'V39',
 'V40',
 'V41',
 'V42',
 'V43',
 'V44',
 'V45',
 'V46',
 'V47',
 'V48',
 'V49',
 'V50',
 'V51',
 'V52',
 'V53',
 'V54',
 'V55',
 'V56',
 'V57',
 'V58',
 'V59',
 'V60',
 'V61',
 'V62',
 'V63',
 'V64',
 'V65',
 'V66',
 'V67',
 'V68',
 'V69',
 'V70',
 'V71',
 'V72',
 'V73',
 'V74',
 'V75',
 'V76',
 'V77',
 'V78',
 'V79',
 'V80',
 'V81',
 'V82',
 'V83',
 'V84',
 'V85',
 'V86',
 'V87',
 'V88',
 'V89',
 'V90',
 'V91',
 'V92',
 'V93',
 'V94',
 'V95',
 'V96',
 'V97',
 'V98',
 'V99',
 'V100']]
```

```
x.shape
```

```
(1212, 100)
```

```
x
```

```
            V1        V2        V3        V4        V5        V6
V7  \
0        39.02     36.49     38.20     38.85     39.38     39.74
37.02
1         1.83      1.71      1.77      1.77      1.68      1.78
1.80
2     68177.69  66138.42  72981.88  74304.33  67549.66  69367.34
69169.41
3     44889.06  39191.86  40728.46  38576.36  45876.06  47034.00
46611.43
```

```
4         5.70      5.40      5.28      5.38      5.27      5.61
6.00
...        ...       ...       ...       ...       ...       ...
...
1207     13.00     12.87     13.27     13.04     13.19     12.53
14.31
1208     48.66     50.11     48.55     50.43     50.09     49.67
48.95
1209  10160.65   9048.63   8994.94   9514.39   9814.74  10195.24
10031.47
1210     34.81     35.07     34.98     32.37     34.16     34.03
33.31
1211   8489.43   7672.98   9132.14   7985.73   8226.85   8554.28
8838.87

              V8        V9       V10  ...       V91       V92       V93
\
0          39.53     38.81     38.79  ...     37.57     36.62     36.92

1           1.70      1.75      1.78  ...      1.71      1.80      1.79

2       73268.61  74465.84  72503.37  ...  69384.71  73438.88  71053.35

3       37668.32  40980.89  38466.15  ...  47653.60  42625.67  40684.20

4           5.38      5.34      5.87  ...      5.52      5.17      5.67

...          ...       ...       ...  ...       ...       ...       ...

1207       13.33     13.63     14.55  ...     12.89     12.48     12.15

1208       48.65     48.63     48.61  ...     47.45     46.93     49.61

1209    10202.28   9152.99   9591.75  ...  10413.41   9068.11   9191.80

1210       32.48     35.63     32.48  ...     33.18     32.76     35.03

1211     8967.24   8635.14   8544.37  ...   7747.70   8609.73   9209.48


             V94       V95       V96       V97       V98       V99
V100
0          38.80     38.52     38.07     36.73     39.46     37.50
39.10
1           1.77      1.74      1.74      1.80      1.78      1.75
1.69
2       71112.62  74916.48  72571.58  66348.97  71063.72  67404.27
74920.24
3       46960.73  44546.80  45410.53  47139.44  43095.68  40888.34
39615.19
```

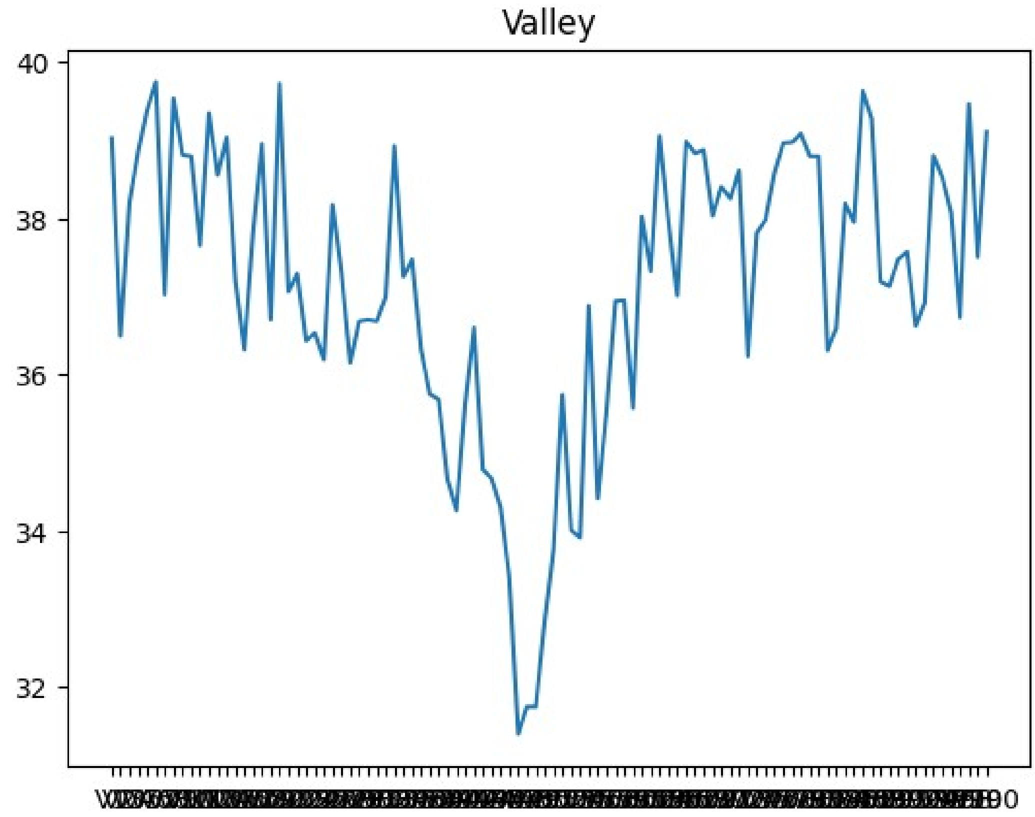
```
4          5.60       5.94       5.73       5.22       5.30       5.73
5.91
...         ...        ...        ...        ...        ...        ...
...
1207      13.15      12.35      13.58      13.86      12.88      13.87
13.51
1208      47.16      48.17      47.94      49.81      49.89      47.43
47.77
1209    9275.04    9848.18    9074.17    9601.74   10366.24    8997.60
9305.77
1210      32.89      31.91      33.85      35.28      32.49      32.83
34.82
1211    8496.33    8724.01    8219.99    8550.86    8679.43    8389.31
8712.80

[1212 rows x 100 columns]
```

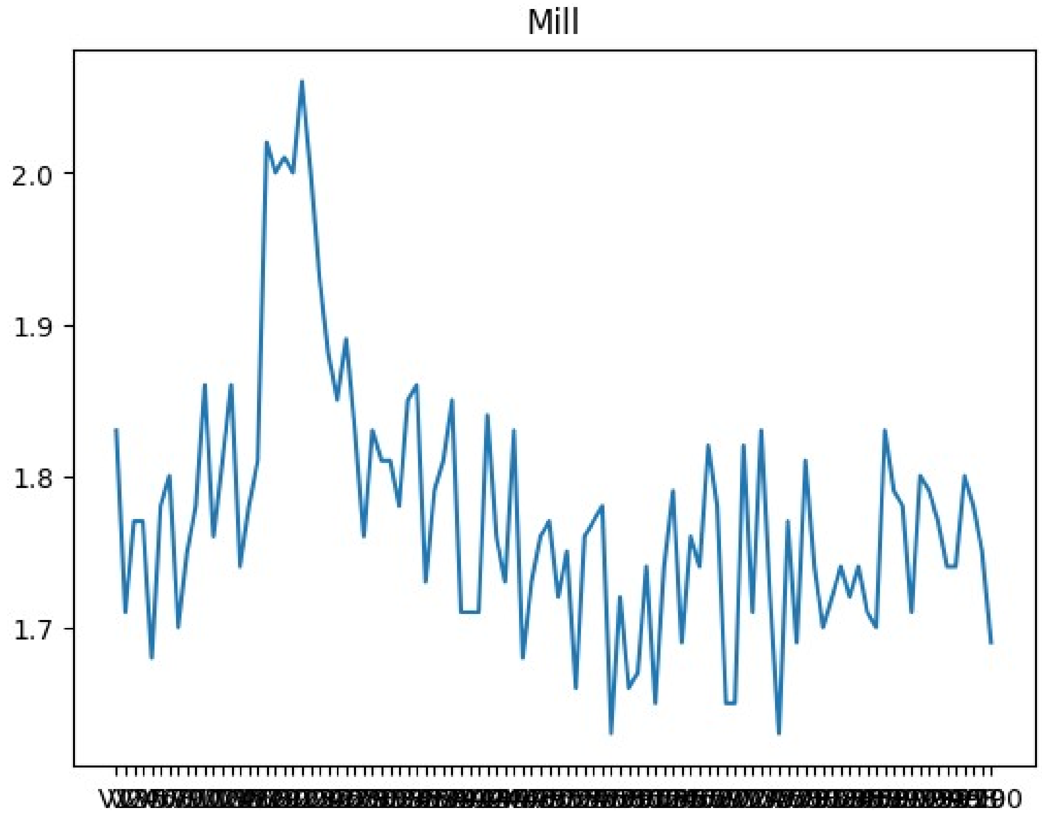
**GET PLOT OF FIRST TWO ROWS**

```
import matplotlib.pyplot as plt

plt.plot(x.iloc[0,:])
plt.title('Valley');
```



```
plt.plot(x.iloc[1,:])
plt.title('Mill');
```



**GET X VARIABLES STANDARDIZED**

```
from sklearn.preprocessing import StandardScaler

ss=StandardScaler()

X=ss.fit_transform(x)

X

array([[-0.45248681, -0.45361784, -0.45100881, ..., -0.45609618,
        -0.45164274, -0.45545496],
       [-0.45455665, -0.45556372, -0.45302369, ..., -0.45821768,
        -0.45362255, -0.45755405],
       [ 3.33983504,  3.24466709,  3.58338069, ...,  3.5427869 ,
         3.27907378,  3.74616847],
       ...,
       [ 0.11084204,  0.0505953 ,  0.04437307, ...,  0.12533312,
         0.04456025,  0.06450317],
       [-0.45272112, -0.45369729, -0.45118691, ..., -0.45648861,
        -0.45190136, -0.45569511],
       [ 0.01782872, -0.02636986,  0.05196137, ...,  0.03036056,
         0.01087365,  0.03123129]])
```

```
X.shape
```

```
(1212, 100)
```

**GET TRAIN TEST SPLIT**

```
from sklearn.model_selection import train_test_split
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,strat
ify=y,random_state=2529)
```

```
x_train,x_test,y_train,y_test
```

```
(             V1         V2         V3         V4         V5         V6
V7  \
 39        1.15       1.13       1.11       1.12       1.19       1.23
1.15
 825   18403.91   18791.43   17248.50   17358.56   18218.28   19797.07
18389.42
 803    1411.90    1271.87    1304.82    1232.86    1342.59    1279.33
1374.18
 1002     87.42      89.35      92.51      97.49      97.58      84.31
85.40
 1199    829.75     972.46     858.79    1005.28    1085.81     942.76
976.23
 ...        ...        ...        ...        ...        ...        ...
...
 1119  29252.19   27124.19   26850.32   26428.38   25680.15   27900.23
29756.82
 1205     19.68      19.83      18.75      19.86      18.91      19.59
18.69
 1077   9713.77    9866.55    9856.44    9519.88    9565.03   10141.24
10346.35
 656    4919.03    4800.84    4922.32    4664.80    4540.30    4555.06
4748.18
 883   19063.94   19141.80   19269.15   18769.04   19448.64   18188.17
18430.66

             V8         V9        V10  ...        V91        V92        V93
\
 39        1.26       1.10       1.16  ...       1.16       1.10       1.18

 825   20107.85   19629.55   18720.03  ...   16990.12   19298.35   17917.37

 803    1257.90    1332.47    1285.78  ...    1344.54    1336.85    1362.11

 1002     78.47      75.03      77.86  ...     106.41      97.79      95.11
 1199   1038.28   1122.89   1173.10  ...    865.04    784.87    909.33

 ...         ...       ...       ...  ...       ...       ...       ...

 1119  28856.70  27632.01  28905.38  ...  28041.54  27293.57  29894.58

 1205     19.15     18.54     18.70  ...     19.74     19.25     18.97

 1077   9566.27  10029.66  10413.35  ...  10273.72   9802.09   9903.01

 656    4527.50   4687.04   4902.13  ...   5047.18   4960.88   4652.16

 883   19191.29  18167.67  19147.81  ...  18233.22  19443.77  19204.76


           V94       V95       V96       V97       V98       V99
V100
 39        1.20      1.15      1.16      1.10      1.19      1.10
1.21
 825   16679.75  17972.59  19478.52  17445.78  18270.03  18958.11
20015.01
 803    1277.79   1296.11   1283.58   1404.15   1229.94   1228.66
1325.00
 1002    106.75    105.01    100.76     93.47    103.87     98.75
103.72
 1199    913.77    780.31    842.00    819.85    921.77    898.24
925.33
 ...         ...       ...       ...       ...       ...       ...
...
 1119  29236.92  29765.27  28139.29  30269.94  27848.40  29613.36
30187.86
 1205     19.57     19.01     19.08     19.86     18.83     18.54
19.72
 1077  10484.86   9892.37   9776.84  10736.77   9757.62   9901.04
10129.28
 656    4621.01   4992.29   4800.86   4699.57   4732.88   4883.27
4590.57
 883   18371.04  18893.55  18893.17  19225.03  18889.97  18599.81
19107.41

 [848 rows x 100 columns],
            V1        V2        V3        V4        V5        V6
V7  \
 1062   2551.13   2760.26   2754.30   2467.05   2651.10   2338.02
2602.59
 246     322.03    328.49    315.19    315.51    327.35    301.59
324.16
 1021   6335.65   5595.55   5682.19   6320.50   6303.45   5834.27
5866.48
```

```
 1063     114.01     105.95     106.98     122.44     105.54     111.83
114.28
 474       57.98      56.15      61.54      58.39      58.63      56.04
61.80
 ...         ...        ...        ...        ...        ...        ...
...
 316       12.54      12.86      12.62      12.62      12.64      12.70
12.91
 219        5.14       5.88       5.19       5.76       5.15       5.86
5.73
 861     7925.86    8578.76    8053.74    8687.85    8538.32    8180.69
8567.72
 600       35.45      33.58      34.47      33.31      34.30      35.89
34.36
 892    86876.03   84189.64   86735.68   80612.64   79373.57   82359.80
79575.23

               V8         V9        V10  ...        V91        V92        V93
\
 1062    2750.32    2615.82    2845.37  ...    2796.74    2472.38    2398.64

 246      302.80     368.04     305.61  ...     360.45     346.85     342.07

 1021    5712.36    5814.06    5857.72  ...    6621.09    6631.35    6308.15

 1063     113.47     118.92     120.63  ...     110.87     107.71     108.95

 474       56.81      61.23      56.17  ...      63.19      59.79      56.06

 ...         ...        ...        ...  ...        ...        ...        ...

 316       12.79      12.77      12.52  ...      12.76      12.85      12.44

 219        5.21       5.32       5.52  ...       5.21       5.66       5.26

 861     8344.50    8156.91    8222.51  ...    8449.96    8405.27    8130.02

 600       33.55      34.02      35.62  ...      34.26      32.85      33.52

 892    82675.50   82224.86   84340.33  ...   81549.24   83823.49   80038.85


              V94        V95        V96        V97        V98        V99
V100
 1062    2724.29    2676.36    2334.51    2791.89    2581.60    2817.14
2315.20
 246      341.72     338.36     350.27     292.88     315.08     335.89
302.74
 1021    6067.72    5847.85    6012.55    5802.20    6264.30    6174.12
6414.87
```

```
 1063    110.01    110.45    119.71    110.29    106.88    121.07
125.78
 474      62.29     58.76     62.59     56.58     59.79     56.69
59.88
 ...        ...       ...       ...       ...       ...       ...
...
 316      12.94     12.76     12.54     13.01     12.77     12.98
12.50
 219       5.93      5.90      5.24      5.81      5.39      5.52
5.86
 861    8155.11   8374.81   8284.99   8655.68   8423.88   8671.66
8369.78
 600      35.03     34.77     32.65     35.44     35.13     32.94
32.18
 892   83987.24  81932.38  79913.73  85908.51  86786.82  83506.53
84415.09

 [364 rows x 100 columns],
 39      1
 825     0
 803     0
 1002    0
 1199    1
        ..
 1119    1
 1205    0
 1077    1
 656     0
 883     1
 Name: Class, Length: 848, dtype: int64,
 1062    0
 246     1
 1021    0
 1063    1
 474     1
        ..
 316     0
 219     1
 861     1
 600     1
 892     1
 Name: Class, Length: 364, dtype: int64)
```

```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

```
((848, 100), (364, 100), (848,), (364,))
```

### GET MODEL TRAIN

```
from sklearn.linear_model import LogisticRegression

lr=LogisticRegression()

lr.fit(x_train,y_train)

/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/
_logistic.py:458: ConvergenceWarning: lbfgs failed to converge
(status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as
shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(

LogisticRegression()
```

### GET MODEL PREDICTION

```
y_pred=lr.predict(x_test)

y_pred.shape

(364,)

y_pred

array([0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1,
1,
       0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1,
1,
       0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0,
1,
       0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0,
0,
       1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1,
1,
       1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 1,
0,
       0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0,
0,
       0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,
1,
       0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0,
1,
       1, 1, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1,
1,
       0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1,
1,
       0, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1,
0,
       1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0,
0,
       1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1,
0,
       1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0,
0,
       0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0,
0,
       0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1])
```

## GET PROBABILITY OF EACH PREDICTED CLASS

```
lr.predict_proba(x_test)
```

```
array([[1.00000000e+000, 9.79892438e-084],
       [1.44980731e-007, 9.99999855e-001],
       [1.00000000e+000, 2.10296309e-117],
       [1.17920074e-003, 9.98820799e-001],
       [3.82913563e-002, 9.61708644e-001],
       [6.34551814e-001, 3.65448186e-001],
       [2.38041494e-001, 7.61958506e-001],
       [1.00000000e+000, 3.92122501e-262],
       [9.85433421e-001, 1.45665790e-002],
       [9.95262593e-001, 4.73740668e-003],
       [3.31102876e-002, 9.66889712e-001],
       [9.99999960e-001, 3.97297756e-008],
       [1.00000000e+000, 0.00000000e+000],
       [9.94685028e-001, 5.31497227e-003],
       [1.00000000e+000, 9.75775486e-030],
       [1.00000000e+000, 0.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [3.80894701e-002, 9.61910530e-001],
       [4.29245109e-001, 5.70754891e-001],
       [5.25543081e-001, 4.74456919e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [5.28184881e-001, 4.71815119e-001],
       [0.00000000e+000, 1.00000000e+000],
       [6.33165320e-001, 3.66834680e-001],
       [4.30569599e-003, 9.95694304e-001],
       [7.23053445e-001, 2.76946555e-001],
       [1.80871375e-008, 9.99999982e-001],
       [4.81787277e-001, 5.18212723e-001],
       [2.13833626e-002, 9.78616637e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 1.44264516e-012],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 0.00000000e+000],
       [5.22413478e-001, 4.77586522e-001],
       [5.02749688e-001, 4.97250312e-001],
       [5.19258270e-001, 4.80741730e-001],
       [4.00169445e-001, 5.99830555e-001],
       [9.99999988e-001, 1.22625206e-008],
       [1.00000000e+000, 5.00751650e-084],
       [4.75058982e-001, 5.24941018e-001],
       [1.00000000e+000, 6.51229539e-022],
       [0.00000000e+000, 1.00000000e+000],
       [7.75814297e-006, 9.99992242e-001],
       [1.00000000e+000, 1.61665320e-030],
       [0.00000000e+000, 1.00000000e+000],
       [2.36476476e-001, 7.63523524e-001],
       [3.26827010e-010, 1.00000000e+000],
       [9.58715329e-011, 1.00000000e+000],
       [6.27568416e-001, 3.72431584e-001],
       [5.10889158e-001, 4.89110842e-001],
       [1.00000000e+000, 7.10243398e-052],
       [4.10852494e-001, 5.89147506e-001],
       [7.84805272e-003, 9.92151947e-001],
       [1.00000000e+000, 2.91454899e-013],
       [4.42946941e-001, 5.57053059e-001],
       [4.94325797e-001, 5.05674203e-001],
       [4.06990731e-001, 5.93009269e-001],
       [1.00000000e+000, 3.92771256e-133],
       [0.00000000e+000, 1.00000000e+000],
       [2.60758044e-006, 9.99997392e-001],
       [4.73794282e-001, 5.26205718e-001],
       [4.85916253e-001, 5.14083747e-001],
       [0.00000000e+000, 1.00000000e+000],
       [9.73310025e-001, 2.66899752e-002],
       [4.35935728e-001, 5.64064272e-001],
       [1.00000000e+000, 3.16309649e-083],
       [0.00000000e+000, 1.00000000e+000],
       [5.47146569e-001, 4.52853431e-001],
       [0.00000000e+000, 1.00000000e+000],
       [7.81748402e-001, 2.18251598e-001],
       [4.96715479e-001, 5.03284521e-001],
       [9.99796291e-001, 2.03709367e-004],
       [2.87095651e-001, 7.12904349e-001],
       [9.33634962e-001, 6.63650377e-002],
       [1.00000000e+000, 4.62565257e-131],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 0.00000000e+000],
       [1.00000000e+000, 3.03889065e-062],
       [4.59824170e-001, 5.40175830e-001],
       [5.09985517e-001, 4.90014483e-001],
       [1.22876861e-001, 8.77123139e-001],
       [0.00000000e+000, 1.00000000e+000],
       [4.59241598e-001, 5.40758402e-001],
       [1.50939924e-006, 9.99998491e-001],
       [5.15704004e-001, 4.84295996e-001],
       [8.66581013e-001, 1.33418987e-001],
       [4.63125347e-001, 5.36874653e-001],
       [1.34706626e-003, 9.98652934e-001],
       [1.00000000e+000, 0.00000000e+000],
       [5.13973370e-001, 4.86026630e-001],
       [9.97413706e-001, 2.58629387e-003],
       [1.00000000e+000, 3.49310807e-031],
       [4.90419681e-001, 5.09580319e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 1.09493221e-195],
       [5.45773338e-001, 4.54226662e-001],
       [1.00000000e+000, 6.88240287e-089],
       [1.00000000e+000, 6.44629434e-019],
       [4.44089210e-016, 1.00000000e+000],
       [1.00000000e+000, 7.33421093e-087],
       [8.10785227e-001, 1.89214773e-001],
       [5.26713582e-006, 9.99994733e-001],
       [1.00000000e+000, 0.00000000e+000],
       [3.58448291e-001, 6.41551709e-001],
       [4.81286566e-010, 1.00000000e+000],
       [4.89323001e-001, 5.10676999e-001],
       [9.56397036e-002, 9.04360296e-001],
       [0.00000000e+000, 1.00000000e+000],
       [3.25729914e-001, 6.74270086e-001],
       [7.53744722e-001, 2.46255278e-001],
       [5.50456221e-001, 4.49543779e-001],
       [1.00000000e+000, 2.62177266e-176],
       [0.00000000e+000, 1.00000000e+000],
       [2.12646204e-001, 7.87353796e-001],
       [1.06314514e-008, 9.99999989e-001],
       [1.00000000e+000, 0.00000000e+000],
       [1.00000000e+000, 4.07249673e-015],
       [9.97276047e-001, 2.72395324e-003],
       [0.00000000e+000, 1.00000000e+000],
       [5.33781304e-001, 4.66218696e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.62866438e-001, 4.37133562e-001],
       [5.61236343e-001, 4.38763657e-001],
       [3.98695153e-002, 9.60130485e-001],
       [9.99999114e-001, 8.86282112e-007],
       [9.64362956e-001, 3.56370445e-002],
       [1.00000000e+000, 5.06598485e-043],
       [1.87043776e-005, 9.99981296e-001],
       [1.00000000e+000, 9.30765770e-058],
       [8.20574118e-001, 1.79425882e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 2.29430314e-069],
       [4.20409967e-001, 5.79590033e-001],
       [9.99999993e-001, 6.94328018e-009],
       [4.50882397e-001, 5.49117603e-001],
       [0.00000000e+000, 1.00000000e+000],
       [3.99586281e-002, 9.60041372e-001],
       [4.72283847e-001, 5.27716153e-001],
       [1.00000000e+000, 3.04057517e-238],
       [6.00020540e-001, 3.99979460e-001],
       [9.92355338e-001, 7.64466237e-003],
       [6.91603994e-001, 3.08396006e-001],
       [1.00000000e+000, 5.78343357e-013],
       [4.98219918e-001, 5.01780082e-001],
       [1.00000000e+000, 0.00000000e+000],
       [6.13993450e-001, 3.86006550e-001],
       [5.25942153e-001, 4.74057847e-001],
       [4.47762272e-001, 5.52237728e-001],
       [5.24645958e-001, 4.75354042e-001],
       [1.00000000e+000, 0.00000000e+000],
       [5.24275850e-001, 4.75724150e-001],
       [5.17588835e-001, 4.82411165e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.18644778e-001, 4.81355222e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 3.11627120e-046],
       [9.99999999e-001, 7.82231601e-010],
       [5.40208711e-001, 4.59791289e-001],
       [1.00000000e+000, 0.00000000e+000],
       [9.99493139e-001, 5.06860936e-004],
       [9.34342960e-001, 6.56570398e-002],
       [3.86231859e-001, 6.13768141e-001],
       [1.00000000e+000, 4.93087857e-218],
       [1.00000000e+000, 1.93788307e-092],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 3.75738639e-018],
       [9.95846453e-001, 4.15354671e-003],
       [9.99997984e-001, 2.01576954e-006],
       [5.73323540e-001, 4.26676460e-001],
       [1.00000000e+000, 3.09913854e-242],
       [8.82419944e-001, 1.17580056e-001],
       [1.00000000e+000, 4.80333724e-070],
       [8.55115978e-012, 1.00000000e+000],
       [1.00000000e+000, 0.00000000e+000],
       [5.19285470e-001, 4.80714530e-001],
       [1.00000000e+000, 0.00000000e+000],
       [1.00000000e+000, 1.82788401e-065],
       [8.11300717e-001, 1.88699283e-001],
       [6.01802788e-001, 3.98197212e-001],
       [4.63623605e-001, 5.36376395e-001],
       [1.00000000e+000, 0.00000000e+000],
       [4.96494314e-001, 5.03505686e-001],
       [7.87576875e-001, 2.12423125e-001],
       [0.00000000e+000, 1.00000000e+000],
       [9.99999980e-001, 2.00926008e-008],
       [1.09338292e-003, 9.98906617e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.09770090e-001, 4.90229910e-001],
       [0.00000000e+000, 1.00000000e+000],
       [3.47100126e-012, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [4.58297030e-001, 5.41702970e-001],
       [8.27184300e-001, 1.72815700e-001],
       [1.00000000e+000, 1.39756991e-012],
       [3.30191510e-006, 9.99996698e-001],
       [1.43263759e-004, 9.99856736e-001],
       [4.95561832e-004, 9.99504438e-001],
       [5.22931414e-001, 4.77068586e-001],
       [1.00000000e+000, 1.18321730e-030],
       [1.00000000e+000, 4.51037216e-021],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [1.50549523e-001, 8.49450477e-001],
       [4.91293832e-001, 5.08706168e-001],
       [3.48512361e-001, 6.51487639e-001],
       [4.70615517e-001, 5.29384483e-001],
       [1.00000000e+000, 8.13524272e-076],
       [1.00000000e+000, 1.33577067e-143],
       [3.85562417e-001, 6.14437583e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 1.09385850e-186],
       [0.00000000e+000, 1.00000000e+000],
       [5.18506056e-001, 4.81493944e-001],
       [3.64168555e-001, 6.35831445e-001],
       [5.16228474e-001, 4.83771526e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [5.22178575e-001, 4.77821425e-001],
       [2.13505491e-007, 9.99999786e-001],
       [1.00000000e+000, 5.54172182e-223],
       [0.00000000e+000, 1.00000000e+000],
       [5.05802166e-001, 4.94197834e-001],
       [0.00000000e+000, 1.00000000e+000],
       [4.72021428e-001, 5.27978572e-001],
       [1.57739722e-001, 8.42260278e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.26869830e-001, 4.73130170e-001],
       [0.00000000e+000, 1.00000000e+000],
       [4.47176565e-001, 5.52823435e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [8.34001285e-001, 1.65998715e-001],
       [1.00000000e+000, 1.04751016e-022],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [2.73642469e-001, 7.26357531e-001],
       [7.96329942e-001, 2.03670058e-001],
       [0.00000000e+000, 1.00000000e+000],
       [2.33005017e-003, 9.97669950e-001],
       [1.00000000e+000, 0.00000000e+000],
       [1.11022302e-015, 1.00000000e+000],
       [4.34208851e-001, 5.65791149e-001],
       [9.99702529e-001, 2.97470851e-004],
       [1.00000000e+000, 1.04185015e-294],
       [1.90052202e-003, 9.98099478e-001],
       [1.27471981e-007, 9.99999873e-001],
       [5.14540398e-001, 4.85459602e-001],
       [5.27537237e-001, 4.72462763e-001],
       [0.00000000e+000, 1.00000000e+000],
       [4.06852730e-001, 5.93147270e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.04841348e-001, 4.95158652e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 1.94541000e-058],
       [5.10261222e-001, 4.89738778e-001],
       [4.79115611e-001, 5.20884389e-001],
       [7.70211184e-001, 2.29788816e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 1.68905018e-055],
       [3.59712260e-014, 1.00000000e+000],
       [5.16740257e-001, 4.83259743e-001],
       [7.72137342e-004, 9.99227863e-001],
       [1.71187086e-002, 9.82881291e-001],
       [4.23439380e-001, 5.76560620e-001],
       [5.04769519e-001, 4.95230481e-001],
       [1.00000000e+000, 1.62057426e-183],
       [4.92858827e-001, 5.07141173e-001],
       [2.21253726e-001, 7.78746274e-001],
       [3.91545588e-001, 6.08454412e-001],
       [9.99998914e-001, 1.08555734e-006],
       [5.45061325e-001, 4.54938675e-001],
       [2.58833001e-003, 9.97411670e-001],
       [0.00000000e+000, 1.00000000e+000],
       [4.54169441e-001, 5.45830559e-001],
       [1.00000000e+000, 0.00000000e+000],
       [4.84217991e-001, 5.15782009e-001],
       [4.54806553e-001, 5.45193447e-001],
       [4.64100426e-001, 5.35899574e-001],
       [4.81058143e-001, 5.18941857e-001],
       [1.00000000e+000, 8.26925871e-012],
       [9.40321597e-001, 5.96784032e-002],
       [9.30280788e-001, 6.97192124e-002],
       [9.99985537e-001, 1.44634314e-005],
       [4.86293495e-001, 5.13706505e-001],
       [4.83620879e-001, 5.16379121e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.18281205e-001, 4.81718795e-001],
       [9.60398281e-001, 3.96017192e-002],
       [8.82909189e-001, 1.17090811e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.04468116e-001, 4.95531884e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 1.97398402e-030],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [4.34564272e-001, 5.65435728e-001],
       [0.00000000e+000, 1.00000000e+000],
       [3.01242927e-001, 6.98757073e-001],
       [5.00275748e-001, 4.99724252e-001],
       [4.92565011e-001, 5.07434989e-001],
       [6.52830789e-009, 9.99999993e-001],
       [2.32227425e-003, 9.97677726e-001],
       [0.00000000e+000, 1.00000000e+000],
       [5.14616047e-001, 4.85383953e-001],
       [0.00000000e+000, 1.00000000e+000],
       [8.17351722e-002, 9.18264828e-001],
       [5.05428197e-001, 4.94571803e-001],
       [6.19886664e-001, 3.80113336e-001],
       [3.37263011e-005, 9.99966274e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [4.95515895e-001, 5.04484105e-001],
       [9.99473050e-001, 5.26950107e-004],
       [1.00000000e+000, 0.00000000e+000],
       [6.07083833e-001, 3.92916167e-001],
       [5.87437776e-001, 4.12562224e-001],
       [6.74626098e-001, 3.25373902e-001],
       [0.00000000e+000, 1.00000000e+000],
       [4.80435513e-001, 5.19564487e-001],
       [0.00000000e+000, 1.00000000e+000],
       [7.56738046e-001, 2.43261954e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [1.99840144e-015, 1.00000000e+000],
       [1.00000000e+000, 0.00000000e+000],
       [7.08044289e-001, 2.91955711e-001],
       [1.00000000e+000, 1.85797288e-263],
       [7.69182585e-001, 2.30817415e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [9.99968245e-001, 3.17551713e-005],
       [0.00000000e+000, 1.00000000e+000],
       [3.32923826e-001, 6.67076174e-001],
       [1.00000000e+000, 1.62314341e-015],
       [9.48752574e-001, 5.12474263e-002],
       [1.00000000e+000, 0.00000000e+000],
       [5.99059985e-001, 4.00940015e-001],
       [1.00000000e+000, 1.42930712e-022],
       [1.00000000e+000, 4.76450603e-283],
       [3.69432386e-001, 6.30567614e-001],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 5.86623071e-289],
       [2.64316783e-006, 9.99997357e-001],
       [9.98722571e-001, 1.27742879e-003],
       [5.55240240e-007, 9.99999445e-001],
       [9.73949152e-001, 2.60508480e-002],
       [6.12533448e-001, 3.87466552e-001],
       [7.79811273e-001, 2.20188727e-001],
       [1.00000000e+000, 1.15872222e-051],
       [3.99680289e-015, 1.00000000e+000],
       [5.61000379e-001, 4.38999621e-001],
       [0.00000000e+000, 1.00000000e+000],
       [9.75329693e-004, 9.99024670e-001],
       [5.15104680e-001, 4.84895320e-001],
       [1.00000000e+000, 0.00000000e+000],
       [5.48029744e-001, 4.51970256e-001],
       [4.58225522e-001, 5.41774478e-001],
       [0.00000000e+000, 1.00000000e+000],
       [2.54686035e-001, 7.45313965e-001],
       [0.00000000e+000, 1.00000000e+000]])
```

### GET MODEL EXALUATION

```
from sklearn.metrics import confusion_matrix,classification_report

confusion_matrix(y_test,y_pred)

array([[175,   7],
       [  4, 178]])
```

```
classification_report(y_test,y_pred)

{"type":"string"}
```

### GET FUTURE PREDICTIONS

```
x_new=df.sample(1)

x_new

        V1     V2    V3     V4     V5     V6     V7     V8     V9
V10  ...  \
328  60.38  59.25  58.7  59.69  60.09  57.99  59.59  61.01  61.23
58.58  ...

       V92    V93    V94    V95    V96    V97    V98    V99   V100
Class
328  56.43  58.92  58.81  57.01  60.19  56.91  60.07  58.64  58.91
1

[1 rows x 101 columns]

x_new.shape

(1, 101)

x_new=x_new.drop('Class',axis=1)

x_new

        V1     V2    V3     V4     V5     V6     V7     V8     V9
V10  ...  \
328  60.38  59.25  58.7  59.69  60.09  57.99  59.59  61.01  61.23
58.58  ...

       V91    V92    V93    V94    V95    V96    V97    V98    V99
V100
328  57.56  56.43  58.92  58.81  57.01  60.19  56.91  60.07  58.64
58.91

[1 rows x 100 columns]

x_new.shape

(1, 100)

x_new=ss.fit_transform(x_new)

y_pred_new=lr.predict(x_new)

/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439:
UserWarning: X does not have valid feature names, but
```

```
LogisticRegression was fitted with feature names
  warnings.warn(

y_pred_new

array([0])

lr.predict_proba(x_new)

/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439:
UserWarning: X does not have valid feature names, but
LogisticRegression was fitted with feature names
  warnings.warn(

array([[0.50005196, 0.49994804]])
```

### EXPLAINATION

```
#step1:import library
#step2:import data
#step3: define y and x
#step4|: split
#step5:select a model
#step6:train model
#step7:predict
```