Hill and Valley

Objective

To develop an Machine Learning Model to predict Hill and Valley using Logistic Regression method

Data Source

This dataset was taken from the github library which is maintained at YBI Foundation. Each record represents 100 points on a two dimensional graph. When plotted in order (from 1 through 100) as the Y coordinate, the point will create either a Hill (a "bump" in the terrain) or a Valley (a "Dip" in the terrain).

[1]:
```
#import libraries
import pandas as pd
```

[2]:
```
import numpy as np
```

[7]:
```
#import data
hill= pd.read_csv('Hill Valley Dataset.csv')
```

[9]:
```
hill.head()
```

[9]:
```
         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]
```

[10]:

```
#Describe data
hill.describe()
```

[10]:

```
                  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]
```

[11]:
```
#data preprocessing
hill.columns
```

```
[11]: Index(['V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9',
'V10',
            …
   'V92', 'V93', 'V94', 'V95', 'V96', 'V97', 'V98', 'V99', 'V100',
         'Class'],
        dtype='object',
        length=101)
```

[12]:
```
hill['Class'].value_counts()
```

```
[12]: 0   606
      1       606
      Name: Class, dtype: int64
```

[13]:
```
#define target(y) and feature(X)
y=hill['Class']
```

[14]:
```
y.shape
```

```
[14]: (1212,)
```

[15]:
```
y
```

```
[15]: 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
```

[16]:
```
X=hill.drop('Class',axis=1)
```

[17]:
```
X.shape
```

[17]:
```
(1212, 100)
```

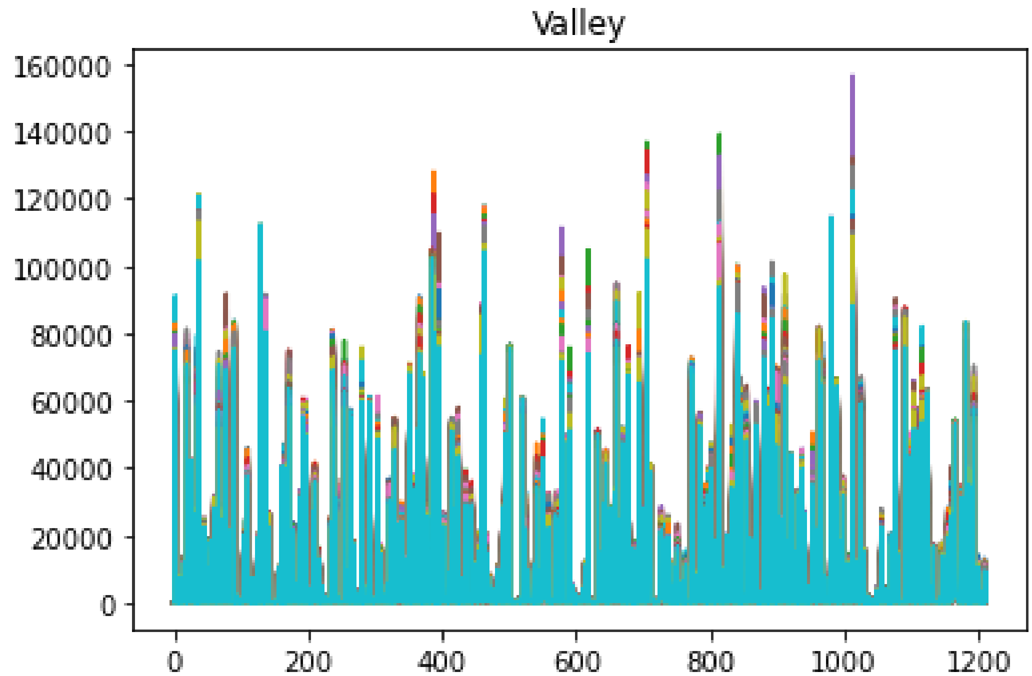
[18]:
```
X
```

[18]:
```
           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
```

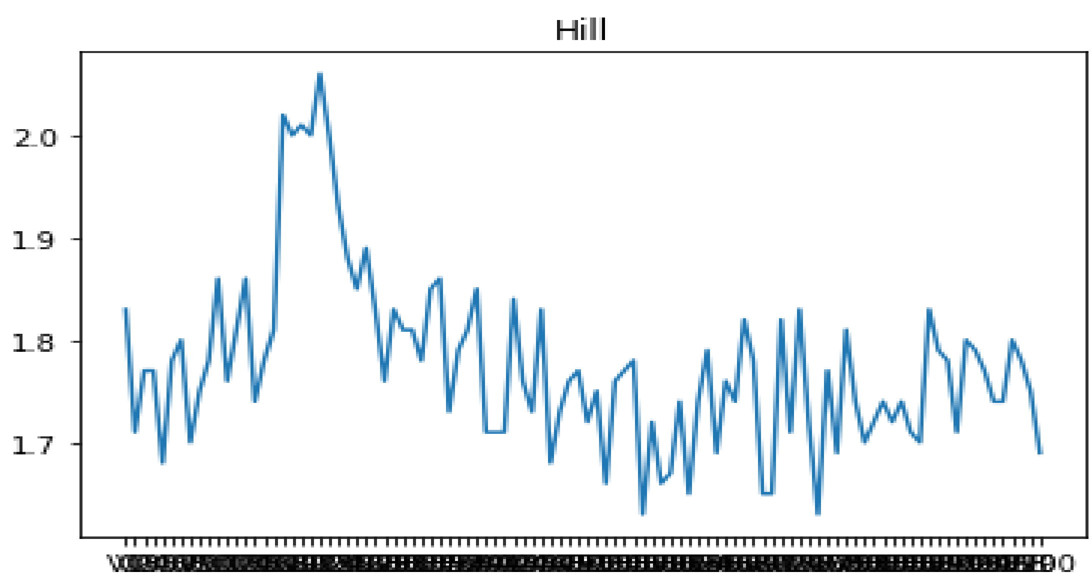
```
1211    8489.43  7672.98    9132.14    7985.73    8226.85
           8554.28    8838.87

          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]
```

[19]:
```
#data visualization
import matplotlib.pyplot as plt
```

[20]:
```
plt.plot(X.iloc[0:])
plt.title('Valley');
```



[21]:
```
plt.plot(X.iloc[1,:])
plt.title('Hill');
```



[22]:
```
#train test split
from sklearn.model_selection import train_test_split
```

[23]:
```
X_train, X_test, y_train, y_test= train_test_split(X,y, test_size=0.3,␣
↪stratify=y,random_state=2529)
```

[24]:
```
X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

[24]:
```
((848, 100), (364, 100), (848,), (364,))
```

[25]:
```
#modelling
from sklearn.linear_model import LogisticRegression
```

[49]:
```
LR=LogisticRegression(max_iter=5500)
```

[50]:
```
LR.fit(X_train,y_train)
```

[50]:
```
LogisticRegression(max_iter=5500)
```

[51]:
```
#model prediction
y_pred=LR.predict(X_test)
```

[52]:
```
y_pred
```

[52]:
```
array([0, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1,
0, 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, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 1,
       0, 0,
       1, 1, 0, 1, 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, 0, 1, 1, 1, 0,
       0, 0,
       1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0,
       1, 1,
       1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 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,
        0, 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, 1, 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, 0, 0, 1, 0,
        0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1], dtype=int64)
```

[53]:
```
y_pred.shape
```

[53]:
```
(364,)
```

[54]:
```
LR.predict_proba(X_test)
```

[54]:
```
array([[1.00000000e+000, 0.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 0.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [8.93886078e-001, 1.06113922e-001],
       [2.52789431e-003, 9.97472106e-001],
       [1.00000000e+000, 0.00000000e+000],
       [1.00000000e+000, 1.12727895e-017],
       [1.00000000e+000, 1.79064935e-017],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 2.09806770e-119],
       [1.00000000e+000, 0.00000000e+000],
       [1.00000000e+000, 3.11868986e-018],
       [1.00000000e+000, 0.00000000e+000],
       [1.00000000e+000, 0.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [9.92168586e-001, 7.83141402e-003],
       [8.92861272e-001, 1.07138728e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [8.88261887e-001, 1.11738113e-001],
       [0.00000000e+000, 1.00000000e+000],
       [9.99467095e-001, 5.32905335e-004],
       [0.00000000e+000, 1.00000000e+000],
       [9.99312833e-001, 6.87166750e-004],
       [0.00000000e+000, 1.00000000e+000],
       [2.67878210e-001, 7.32121790e-001],
       [0.00000000e+000, 1.00000000e+000],
       [0.00000000e+000, 1.00000000e+000],
       [1.00000000e+000, 6.12652184e-229],
```

```
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[8.53090233e-001, 1.46909767e-001],
[6.31344198e-001, 3.68655802e-001],
[9.43698992e-001, 5.63010084e-002],
[3.35609571e-003, 9.96643904e-001],
[1.00000000e+000, 9.43164353e-153],
[1.00000000e+000, 0.00000000e+000],
[2.60921550e-001, 7.39078450e-001],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[8.43008774e-010, 9.99999999e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[9.99340959e-001, 6.59040774e-004],
[8.03304466e-001, 1.96695534e-001],
[1.00000000e+000, 5.76060640e-029],
[2.47030397e-003, 9.97529696e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 5.83153562e-250],
[2.11440200e-002, 9.78855980e-001],
[2.67969120e-001, 7.32030880e-001],
[1.68493714e-003, 9.98315063e-001],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.46074226e-001, 8.53925774e-001],
[2.34025962e-001, 7.65974038e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 3.41165019e-019],
[3.51765050e-002, 9.64823495e-001],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[9.07547171e-001, 9.24528288e-002],
[0.00000000e+000, 1.00000000e+000], [1.00000000e+000,
1.04341639e-012],
[5.69135694e-001, 4.30864306e-001],
[1.00000000e+000, 3.05251838e-040],
[1.54220290e-008, 9.99999985e-001],
[1.00000000e+000, 9.22195737e-015],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
```

```
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[9.49147695e-002, 9.05085231e-001],
[6.25801738e-001, 3.74198262e-001],
[3.77364806e-012, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[6.39822260e-002, 9.36017774e-001],
[0.00000000e+000, 1.00000000e+000],
[6.95162811e-001, 3.04837189e-001],
[1.00000000e+000, 2.89815603e-012],
[6.49617970e-002, 9.35038203e-001],
[2.40652812e-001, 7.59347188e-001],
[1.00000000e+000, 0.00000000e+000],
[3.63512557e-001, 6.36487443e-001],
[1.00000000e+000, 1.47213165e-040],
[1.00000000e+000, 0.00000000e+000],
[3.89868130e-001, 6.10131870e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[8.97779479e-001, 1.02220521e-001],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 3.43406524e-106],
[9.99999999e-001, 1.22407808e-009],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[2.31994934e-009, 9.99999998e-001],
[0.00000000e+000, 1.00000000e+000],
[3.12140174e-001, 6.87859826e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.54700339e-004, 9.99845300e-001],
[1.00000000e+000, 4.68415684e-027],
[9.09348878e-001, 9.06511220e-002],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[4.02842643e-008, 9.99999960e-001], [0.00000000e+000,
1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 2.97801810e-172],
[1.00000000e+000, 1.62599556e-080],
[0.00000000e+000, 1.00000000e+000],
```

```
[9.96372873e-001, 3.62712653e-003],
[0.00000000e+000, 1.00000000e+000],
[9.77645678e-001, 2.23543223e-002],
[9.36617585e-001, 6.33824148e-002],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 1.89818496e-132],
[1.00000000e+000, 2.26958319e-023],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 1.85721511e-015],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[3.93759912e-004, 9.99606240e-001],
[1.00000000e+000, 4.14785662e-118],
[5.24564587e-002, 9.47543541e-001],
[0.00000000e+000, 1.00000000e+000],
[8.72635297e-014, 1.00000000e+000],
[2.25226983e-001, 7.74773017e-001],
[1.00000000e+000, 0.00000000e+000],
[9.97345740e-001, 2.65426002e-003],
[1.00000000e+000, 1.19796929e-019],
[9.99968964e-001, 3.10363173e-005],
[1.00000000e+000, 0.00000000e+000],
[3.56999410e-001, 6.43000590e-001],
[1.00000000e+000, 0.00000000e+000],
[9.99839956e-001, 1.60043984e-004],
[8.35205993e-001, 1.64794007e-001],
[1.57266973e-001, 8.42733027e-001],
[7.41350702e-001, 2.58649298e-001],
[1.00000000e+000, 0.00000000e+000],
[8.24638660e-001, 1.75361340e-001],
[7.22189654e-001, 2.77810346e-001],
[0.00000000e+000, 1.00000000e+000],
[7.43704207e-001, 2.56295793e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 8.98068503e-091],
[7.69566064e-001, 2.30433936e-001],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 1.10647962e-047],
[1.00000000e+000, 6.04626515e-021], [3.58484221e-005,
9.99964152e-001],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
```

```
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 2.12065813e-044],
[1.00000000e+000, 3.09649443e-089],
[9.75320961e-001, 2.46790390e-002],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 3.62729618e-022],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[7.31080463e-001, 2.68919537e-001],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[9.99999771e-001, 2.28589158e-007],
[9.99995102e-001, 4.89778591e-006],
[1.16081620e-001, 8.83918380e-001],
[1.00000000e+000, 0.00000000e+000],
[2.52193376e-001, 7.47806624e-001],
[9.99229833e-001, 7.70167001e-004],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 2.05976970e-154],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[6.45160417e-001, 3.54839583e-001],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.04708919e-001, 8.95291081e-001],
[1.00000000e+000, 2.67274464e-010],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 5.16290737e-017],
[8.62304834e-001, 1.37695166e-001],
[1.00000000e+000, 1.74383215e-208],
[1.00000000e+000, 1.51944366e-289],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[4.17532675e-012, 1.00000000e+000],
[3.03567390e-001, 6.96432610e-001],
[1.25318932e-002, 9.87468107e-001],
[1.15122640e-001, 8.84877360e-001],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000], [9.29719761e-003,
9.90702802e-001],
```

```
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[6.86215795e-001, 3.13784205e-001],
[2.12505171e-003, 9.97874948e-001],
[6.48811392e-001, 3.51188608e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[7.77544282e-002, 9.22245572e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[6.15401109e-001, 3.84598891e-001],
[0.00000000e+000, 1.00000000e+000],
[1.98479971e-001, 8.01520029e-001],
[3.55271368e-015, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[8.88012362e-002, 9.11198764e-001],
[0.00000000e+000, 1.00000000e+000],
[5.01920646e-002, 9.49807935e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 2.54358287e-010],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[8.15999881e-004, 9.99184000e-001],
[9.96917709e-001, 3.08229147e-003],
[0.00000000e+000, 1.00000000e+000],
[1.39888101e-014, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.36283130e-002, 9.86371687e-001],
[1.00000000e+000, 3.90902599e-045],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[6.83784857e-001, 3.16215143e-001],
[9.75019101e-001, 2.49808986e-002],
[0.00000000e+000, 1.00000000e+000],
[3.67752008e-005, 9.99963225e-001],
[0.00000000e+000, 1.00000000e+000],
[7.13638805e-001, 2.86361195e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
```

[6.72532692e-001, 3.27467308e-001], [2.80940075e-001, 7.19059925e-001],
[9.99999958e-001, 4.17845855e-008],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[7.28228817e-001, 2.71771183e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[2.75370969e-003, 9.97246290e-001],
[5.70892231e-001, 4.29107769e-001],
[1.00000000e+000, 0.00000000e+000],
[3.11061298e-001, 6.88938702e-001],
[3.60828611e-009, 9.99999996e-001],
[4.00852594e-004, 9.99599147e-001],
[1.00000000e+000, 1.19612975e-110],
[9.36536948e-001, 6.34630516e-002],
[8.39328607e-014, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.20373245e-001, 8.79626755e-001],
[1.00000000e+000, 0.00000000e+000],
[3.47497421e-001, 6.52502579e-001],
[5.92440557e-004, 9.99407559e-001],
[1.04608405e-001, 8.95391595e-001],
[2.28030469e-001, 7.71969531e-001],
[1.00000000e+000, 4.78388497e-160],
[1.00000000e+000, 7.11281014e-017],
[1.00000000e+000, 2.61653483e-021],
[1.00000000e+000, 8.05121833e-051],
[7.68282174e-001, 2.31717826e-001],
[3.05396732e-001, 6.94603268e-001],
[0.00000000e+000, 1.00000000e+000],
[6.38138742e-001, 3.61861258e-001],
[1.00000000e+000, 1.00008554e-083],
[1.00000000e+000, 1.94802314e-011],
[0.00000000e+000, 1.00000000e+000],
[5.53363396e-001, 4.46636604e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 1.43489515e-218],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[7.51285633e-003, 9.92487144e-001],
[0.00000000e+000, 1.00000000e+000],
[5.25431594e-006, 9.99994746e-001],

```
[6.13817323e-001, 3.86182677e-001],
[3.12279901e-001, 6.87720099e-001],
[0.00000000e+000, 1.00000000e+000], [0.00000000e+000,
1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[7.35165935e-001, 2.64834065e-001],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[6.97177338e-001, 3.02822662e-001],
[9.99999557e-001, 4.42997583e-007],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[4.15595919e-001, 5.84404081e-001],
[1.00000000e+000, 1.10782666e-052],
[1.00000000e+000, 0.00000000e+000],
[9.99962084e-001, 3.79160350e-005],
[1.42663659e-011, 1.00000000e+000],
[8.58969566e-001, 1.41030434e-001],
[0.00000000e+000, 1.00000000e+000],
[2.79453642e-001, 7.20546358e-001],
[0.00000000e+000, 1.00000000e+000],
[9.74466484e-001, 2.55335158e-002],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[9.96798135e-001, 3.20186500e-003],
[1.00000000e+000, 0.00000000e+000],
[9.99997478e-001, 2.52241855e-006],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 7.31861323e-040],
[0.00000000e+000, 1.00000000e+000],
[2.70982570e-010, 1.00000000e+000],
[1.00000000e+000, 1.31721590e-163],
[9.99999998e-001, 1.85057072e-009],
[1.00000000e+000, 0.00000000e+000],
[9.76478453e-001, 2.35215470e-002],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[6.24885302e-008, 9.99999938e-001],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 0.00000000e+000],
[1.00000000e+000, 1.15872040e-295],
```

```
[1.00000000e+000, 9.87545537e-041],
[0.00000000e+000, 1.00000000e+000],
[1.00000000e+000, 5.88794054e-043],
[8.84634748e-001, 1.15365252e-001],
[9.99999983e-001, 1.70063477e-008], [1.00000000e+000,
0.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[9.83737548e-001, 1.62624524e-002],
[0.00000000e+000, 1.00000000e+000],
[0.00000000e+000, 1.00000000e+000],
[7.10565584e-001, 2.89434416e-001],
[1.00000000e+000, 0.00000000e+000],
[8.98433814e-001, 1.01566186e-001],
[1.64197639e-001, 8.35802361e-001],
[0.00000000e+000, 1.00000000e+000],
[1.13044168e-007, 9.99999887e-001],
[0.00000000e+000, 1.00000000e+000]])
```

[55]:
```
#model evaluation
from sklearn.metrics import confusion_matrix, classification_report
```

[56]:
```
print(confusion_matrix(y_test,y_pred))
```

```
[[176   6]
 [  6 176]]
```

[57]:
```
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.97      0.97      0.97       182
           1       0.97      0.97      0.97       182

    accuracy                           0.97       364
   macro avg       0.97      0.97      0.97       364
weighted           0.97      0.97      0.97       364
avg
```

Explanation

Accuracy in machine leaning model is used for Classification. Accuracy score in Machine Learning model means number of correct predictions. It is the ratio of number of correct predictions to the total number of predictions. In machine learning model accuracy score above 0.7 is treated as good-to-go-model; here, our accuracy score is 0.97 therefore our Machine learning model is 97% accurate in correct predictions.