***IMPORT LIBRARY***

1] Import pandas as pd

[2] import numpy as np

[3] import matplotlib.pyplot as pit

***IMPORT DATA***

✓[4] from sklearn.datasets import load\_digits

[5] df load\_digits()

[6] axes plt.subplots(nrows-1, ncols-4, figsize-(10, 3))

***DATA PRE-PROCESSING***

for ax, image, label in zip(axes, df.images, df.target): ax.set\_axis\_off() ax.imshow(image, cmap-plt.cm.gray\_r, interpolation-"nearest") ax.set\_title("Training: %1" % label

df.images.shape

(1797, 8, 8)

✓[8] df.images[0]

array([[ 0., 0., 5., 13., 9., 1., 0., 0.],

[ 0., 0., 13., 15., 10., 15., 5., 0.], [0., 3., 15., 2., 0., 11., 8., θ.],

[0., 4., 12., 0., 0., 8., 8., θ.],

[0., 5., 8., 0., 0., 9., 8., 0.], [θ., 4., 11., 0., 1., 12., 7., 8.1,

[0., 2., 14., 5., 10., 12., 0., 0.], [ 0., 0., 6., 13., 10., 0., 0., 0.]])

df.images[0].shape

(x)

(8,8)

[10] len(df.images)

1797

✓[11

] n\_samples len(df.images) data df.images.reshape((n\_samples, -1))

✓[12] data[0]

array([ 0., 0., 5., 13., 9., 1., 0., 0., 0., 15., 5., 0., 0., 12., 0., 0., 8., Θ., 0., 4., 11., 18., 12., 0., 0., 3., 15., 2., 0., 11., 8., 8., Θ., 5., 8., 0., 1., 12., 7., 8., 0., 0., 6., 13., 10., 0., 13., 15., 10., 8., 8., θ., 4., 8., 8., 9., 8., 8., 2., 14., 5., 0., 0., 0.1)

✓[13] data[0].shape(64,)

✓[14] data.shape

(1797, 64)

***Scaling image data***

✓[15] data.min()

0.0

(x)

[16] data.max()

☐ ✓

16.0

✓[17] data data/16

[18] data.min()

0.0

✓[19] data.max()

1.0

Y

Subscribe

data[0]

array ([0. , θ. , 0.3125, 0.8125, 0.5625, 0.0625, 0. , 0.

0. 0. 0.8125, 0.9375, 0.625 0.9375, 0.3125, 0.

0.

, 0.1875, 0.9375, 0.125, 0. , 0.6875, 0.5 , 0.

0. 0. , 0.25, 0.75 , 0.3125, 0.5 , θ. 0. , 0. 0. , 0.5 , 0.5 0.

, 0.5625, 0.5 , 0.0625, 0.75 0. , 0.4375, 0. 0. 0.25 , 0.6875, 0.

0. , 0.125, 0.875, 0.3125, 0.625, 0.75 , θ. , θ.

0. 0. , 0.375, 0.8125, 0.625, 0. , 0. , 0.

Train Test Split Data

j)

***Train Test Split Data***

✓[21] from sklearn.model\_selection import train\_test\_split

✓[22] X\_train, X\_test, y\_train, y\_test train\_test\_split(data, df.target, test\_size=(0.3)

✓[23] X\_train.shape, X\_test.shape, y\_train.shape, y\_test.shape ((1257, 64), (540, 64), (1257,), (540,))

***Random Forest Model***

***Predict Test Data***

[27] y\_pred rf.predict(X\_test)

(x)

Y

Subscribe

✓[28] y\_pred

] from sklearn.metrics import confusion\_matrix, classification\_report

(x)

✓[30] confusion\_matrix(y\_test, y\_pred)

array([[60, 0, 0, 0, 1, 0, 0, 0, 0, 0],

[0, 60, 0, 1, 0, 0, 0, 0, 0, 0], [

0, 0, 61, 0, 0, 0, 0, 0, 0, 1],

[0, 0, 0, 54, 0, 0, 0, 1, 0, 0],

[0, 0, 0, 0, 40, 0, 0, 1, 0, 0],

[0, 0, 0, 0, 0, 58, 1, 0, 1, 0],

[0, 1, 0, 0, 1, 0, 50, 0, 0, 0],

[0, 0, 0, 0, 0, 0, 0, 52, 0, 0], [8, 2, 0, 0, 1, 1, 0, 0, 46, 0],

[0, 0, 0, 0, 0, 0, 0, 1, 0, 45]])

array([6, 5, 2, 6, 8, 8, 2, 7, 5, 1, 1, 4, 3, 7, 2, 0, 0, 7, 0, 2, 5, 3, 2, 5, 2, 7, 5, 9, 1, 8, 3, 3, 1, 1, 8, 4, 0, 4, 2, 1, 3, 1, 6, 9, 8, 8, 6, 7, 5, 4, 3, 8, 5, 8, 6, 8, 7, 1, 1, 1, 6, 1, 9, 3, 8, 8, 8, 1, 5, 8, 6, 2, 5, 9, 3, 5, 6, 8, 3, 5, 4, 0, 4, 1, 9, 1, 1, 3, 0, 3, 7, 1, 6, 3, 9, 7, 2, 9, 6, 2, 7, 8, 6, 9, 9, 3, 5, 4, 4, 6, 3, 6, 2, 3, 9, 6, 3, 7, 2, 7, 3, 1, 3, 6, 1, 9, 8, 5, 2, 9, 6, 1, 3, 4, 1, 6, 7, 2, 8, 8, 2, 7, 7, 8, 7, 1, 2, 5, 0, 0, 3, 1, 5, 8, 0, 3, 0, 1, 5, 5, 4, 8, 8, 7, 9, 5, 5, 5, 6, 4, 3, 6, 9, 3, 6, 0, 2, 5, 9, 5, 7, 6, 4, 2, 2, 5, 7, 7, 2, 8, 8, 1, 0, 7, 6, 7, 2, 7, 6, 6, 1, 4, 4, 7, 5, 0, 2, 9, 8, 4, 0, 0, 8, 1, 9, 4, 2, 4, 6, 1, 0, 4, 5, 1, 6, 3, 1, 4, 8, 3, 7, 3, 3, 9, 1, 2, 5, 5, 0, 7, 1, 6, 8, 3, 3, 6, 1, 1,

2, 7, 3, 0, 7, 9, 4, 7, 3, 5, 8, 1, 3, 3, 7, 8, 6, 9, 9, 6, 7, 8, 3, 9, 8, 8, 7, 7, 1, 5, 3, 2, 2,

1, 4, 2, 5, 2, 0, 9, 5, 1, 2, 1, 8, 8, 4, 2, 9, 8, 3, 2,

6, 7, 1, 1, 4, 8, 9, 5, 1, 5, 6, 4, 0, 7, 3, 7, 4, 7, 7, 9, 5, 0,

5, 0, 9, 2, 0, 1, 1, 9, 3, 7, 4, 7, 0, 0, 3, 9, 6, 2, 8, 8, 9, 9,

6, 7, 1, 2, 6, 3, 9, 8, 1, 6, 9, 3, 5, 1, 6, 6, 7, 9, 1, 0, 1, 2,

5, 2, 1, 4, 2, 8, 9, 5, 1, 7, 3, 2, 8, 9, 8, 3, 6, 4, 2, 2, 9, 1,

2, 8, 4, 8, 4, 2, 8, 3, 8, 8, 6, 2, 0, 1, 5, 7, 6, 5, 5, 8, 6, 5,

5, 6, 2, 5, 8, 8, 4, 7, 4, 8, 7, 0, 0, 2, 0, 8, 4, 8, 1, 6, 4, 2, 70, 4, 7, 1, 1, 1, 2, 6, 2, 8, 6, 1, 9, 1, 2, 0, 3, 6, 8, 5, 5,

97, 0, 0, 6, 3, 4, 5, 3, 9, 9, 8, 1, 2, 4, 4, 5, 4, 2, 5, 8, 6, 9, 3, 1, 5, 6, 5, 7, 2, 2, 6, 5, 2, 4, 9, 7, 3, 8, 5, 1,

2,9 2, 3, 4, 8, 5, 7, 7, 4, 0, 0, 1, 9, 2, 1, 6, 7, 3, 2, 0, 5, 8, 7, 5,

2, 0, 5, 0, 0, 4, 8, 5, 5, 7, 9, 3])