CHANDAN KUMAR

ID: GO_STP_13267

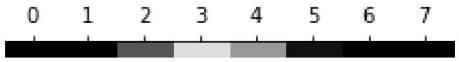
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import sklearn
from sklearn.svm import SVC
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix
```

```
digits=load_digits()
digits.keys()
```

```
dict_keys(['data', 'target', 'target_names', 'images', 'DESCR'])
```

```
plt.gray()
for i in range(2):
  plt.matshow(digits.images[i])
plt.show()
```

```
<Figure size 432x288 with 0 Axes>
```




```
df1=pd.DataFrame(digits.data)
df2=pd.DataFrame(digits.target)
```

```
X=df1; Y=df2
x_train, x_test, y_train, y_test=train_test_split(X,Y, test_size=0.2, random_state=32)
```

```
model1=SVC(kernel='rbf')
model1.fit(x_train, y_train)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:760: DataConversic
  y = column_or_1d(y, warn=True)
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='rbf',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

```
y_pred1=model1.predict(x_test)
```

```
print("Accuracy:",accuracy_score(y_test, y_pred1)*100)
```

```
Accuracy: 99.44444444444444
```

```
print("Confusion matrix:\n",confusion_matrix(y_test, y_pred1))
```

```
Confusion matrix:
 [[38  0  0  0  0  0  0  0  0  0]
 [ 0 36  0  0  0  0  0  0  0  0]
 [ 0  0 32  0  0  0  0  0  0  0]
 [ 0  0  0 56  0  0  0  0  0  0]
 [ 0  0  0  0 30  0  0  0  0  1]
 [ 0  0  0  0  0 35  0  0  0  1]
 [ 0  0  0  0  0  0 34  0  0  0]
 [ 0  0  0  0  0  0  0 34  0  0]
 [ 0  0  0  0  0  0  0  0 27  0]
 [ 0  0  0  0  0  0  0  0  0 36]]
```

```
model2=SVC(kernel='linear')
model2.fit(x_train, y_train)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/validation.py:760: DataConversic
  y = column_or_1d(y, warn=True)
SVC(C=1.0, break_ties=False, cache_size=200, class_weight=None, coef0=0.0,
    decision_function_shape='ovr', degree=3, gamma='scale', kernel='linear',
    max_iter=-1, probability=False, random_state=None, shrinking=True,
    tol=0.001, verbose=False)
```

```
y_pred2=model1.predict(x_test)
```

```
1 print("Accuracy:",accuracy_score(y_test, y_pred2)*100)
```

```
Accuracy: 99.44444444444444
```

```
1 print("Confusion matrix:\n",confusion_matrix(y_test, y_pred2))
```

```
Confusion matrix:
 [[38  0  0  0  0  0  0  0  0  0]
 [ 0 36  0  0  0  0  0  0  0  0]
 [ 0  0 32  0  0  0  0  0  0  0]
 [ 0  0  0 56  0  0  0  0  0  0]
 [ 0  0  0  0 30  0  0  0  0  1]
 [ 0  0  0  0  0 35  0  0  0  1]
 [ 0  0  0  0  0  0 34  0  0  0]
 [ 0  0  0  0  0  0  0 34  0  0]
 [ 0  0  0  0  0  0  0  0 27  0]
 [ 0  0  0  0  0  0  0  0  0 36]]
```

✓ 0s completed at 5:55 PM