```
import pandas as pd
from sklearn.datasets import load_digits
digits = load_digits()
```

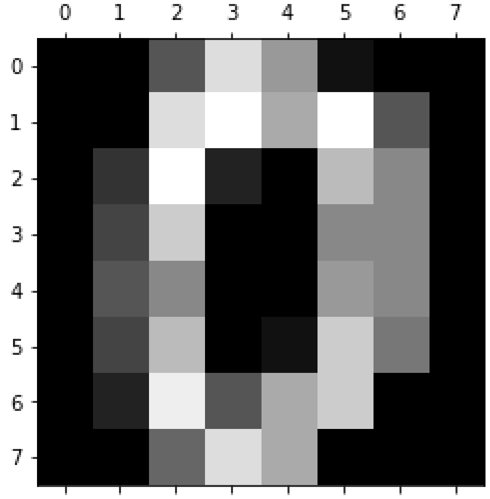
```
dir(digits)
```

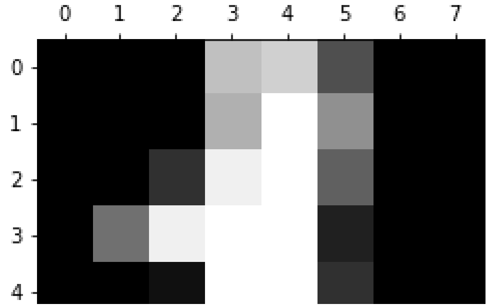
```
['DESCR', 'data', 'images', 'target', 'target_names']
```

```
%matplotlib inline
import matplotlib.pyplot as plt
```

```
plt.gray()
for i in range(4):
    plt.matshow(digits.images[i])
```

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





```
df = pd.DataFrame(digits.data)
df.head()
```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0.0 | 0.0 | 5.0 | 13.0 | 9.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 13.0 | 15.0 | 10.0 | 15.0 | 5.0 | 0.0 | 0.0 | 3 |
| **1** | 0.0 | 0.0 | 0.0 | 12.0 | 13.0 | 5.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 11.0 | 16.0 | 9.0 | 0.0 | 0.0 | 0.0 | 0 |
| **2** | 0.0 | 0.0 | 0.0 | 4.0 | 15.0 | 12.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.0 | 16.0 | 15.0 | 14.0 | 0.0 | 0.0 | 0.0 | 0 |
| **3** | 0.0 | 0.0 | 7.0 | 15.0 | 13.0 | 1.0 | 0.0 | 0.0 | 0.0 | 8.0 | 13.0 | 6.0 | 15.0 | 4.0 | 0.0 | 0.0 | 0.0 | 2 |
| **4** | 0.0 | 0.0 | 0.0 | 1.0 | 11.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 7.0 | 8.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0 |

```
df['target'] = digits.target
```

```
df[0:12]
```

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0.0 | 0.0 | 5.0 | 13.0 | 9.0 | 1.0 | 0.0 | 0.0 | 0.0 | 0.0 | 13.0 | 15.0 | 10.0 | 15.0 | 5.0 | 0.0 | 0. |
| **1** | 0.0 | 0.0 | 0.0 | 12.0 | 13.0 | 5.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 11.0 | 16.0 | 9.0 | 0.0 | 0.0 | 0. |
| **2** | 0.0 | 0.0 | 0.0 | 4.0 | 15.0 | 12.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.0 | 16.0 | 15.0 | 14.0 | 0.0 | 0.0 | 0. |
| **3** | 0.0 | 0.0 | 7.0 | 15.0 | 13.0 | 1.0 | 0.0 | 0.0 | 0.0 | 8.0 | 13.0 | 6.0 | 15.0 | 4.0 | 0.0 | 0.0 | 0. |
| **4** | 0.0 | 0.0 | 0.0 | 1.0 | 11.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 7.0 | 8.0 | 0.0 | 0.0 | 0.0 | 0. |
| **5** | 0.0 | 0.0 | 12.0 | 10.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 14.0 | 16.0 | 16.0 | 14.0 | 0.0 | 0.0 | 0. |
| **6** | 0.0 | 0.0 | 0.0 | 12.0 | 13.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 5.0 | 16.0 | 8.0 | 0.0 | 0.0 | 0.0 | 0. |
| **7** | 0.0 | 0.0 | 7.0 | 8.0 | 13.0 | 16.0 | 15.0 | 1.0 | 0.0 | 0.0 | 7.0 | 7.0 | 4.0 | 11.0 | 12.0 | 0.0 | 0. |

```
X = df.drop('target',axis='columns')
y = df.target
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.2)
```

```
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=20)
model.fit(X_train, y_train)
```

```
RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
                       criterion='gini', max_depth=None, max_features='auto',
                       max_leaf_nodes=None, max_samples=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, n_estimators=20,
                       n_jobs=None, oob_score=False, random_state=None,
                       verbose=0, warm_start=False)
```

```
model.score(X_test, y_test)
```

```
0.95
```

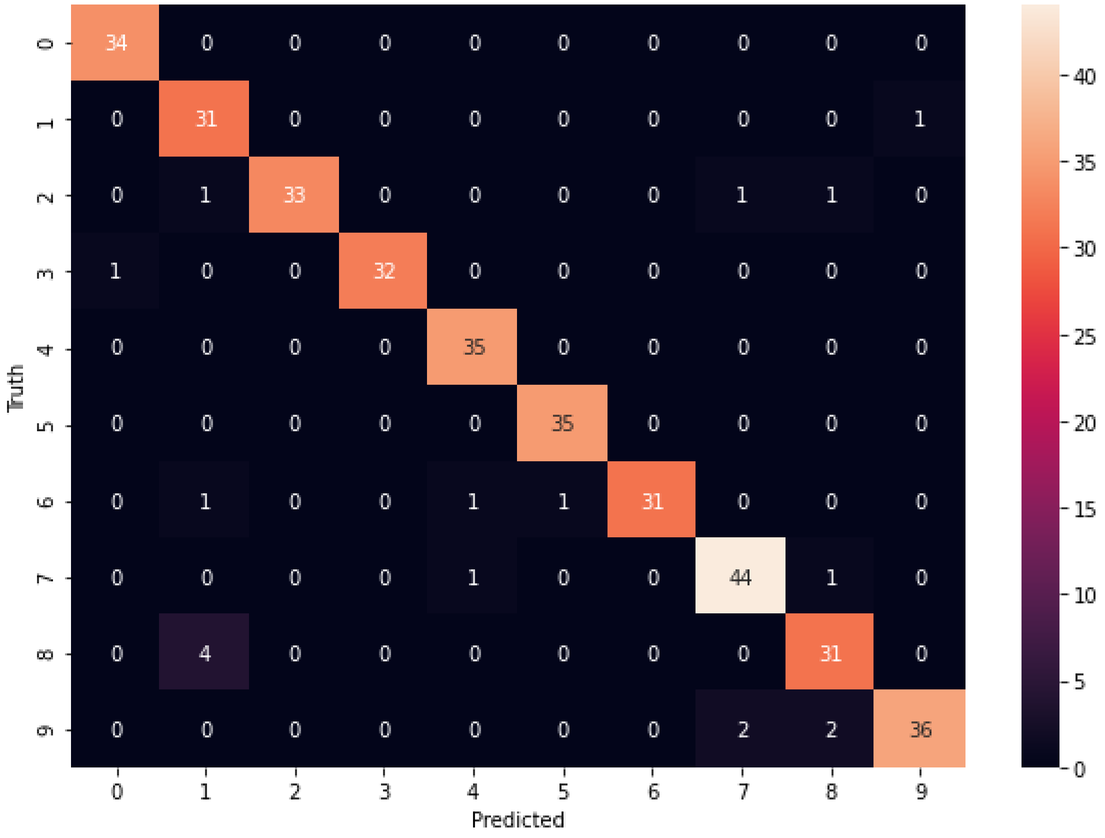
```
y_predicted = model.predict(X_test)
```

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_predicted)
cm
```

```
array([[34,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0, 31,  0,  0,  0,  0,  0,  0,  0,  1],
       [ 0,  1, 33,  0,  0,  0,  0,  1,  1,  0],
       [ 1,  0,  0, 32,  0,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0, 35,  0,  0,  0,  0,  0],
       [ 0,  0,  0,  0,  0, 35,  0,  0,  0,  0],
       [ 0,  1,  0,  0,  1,  1, 31,  0,  0,  0],
       [ 0,  0,  0,  0,  1,  0,  0, 44,  1,  0],
       [ 0,  4,  0,  0,  0,  0,  0,  0, 31,  0],
       [ 0,  0,  0,  0,  0,  0,  0,  2,  2, 36]])
```

```
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sn
plt.figure(figsize=(10,7))
sn.heatmap(cm, annot=True)
plt.xlabel('Predicted')
plt.ylabel('Truth')
```

```
    Text(69.0, 0.5, 'Truth')
```



✓ 1s completed at 10:54 ● ✕