

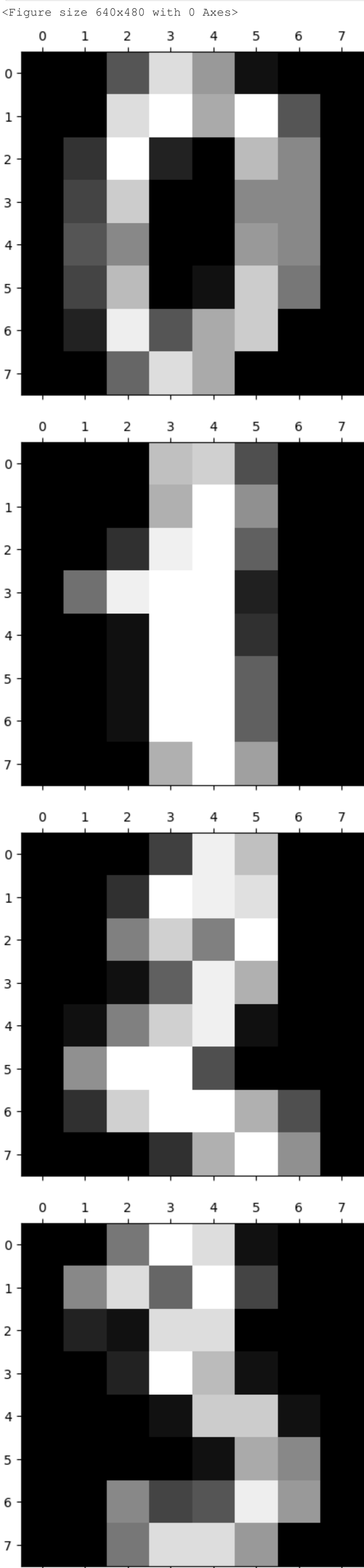
In [2]: `import pandas as pd
from sklearn.datasets import load_digits
digits=load_digits()`

In [3]: `dir(digits)`

Out[3]: `['DESCR', 'data', 'feature_names', 'frame', 'images', 'target', 'target_names']`

In [4]: `matplotlib inline
import matplotlib.pyplot as plt`

In [7]: `plt.gray()
for i in range(4):
 plt.matshow(digits.images[i])
plt.show()`



In [68]: `df=pd.DataFrame(digits.data)
df.head()`

Out[68]:

	0	1	2	3	4	5	6	7	8	9	...	54	55	56	57	58	59	60	61	62	63
0	0.0	0.0	5.0	13.0	9.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	6.0	13.0	10.0	0.0	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	0.0	0.0	0.0	0.0	11.0	16.0	10.0	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	...	5.0	0.0	0.0	0.0	0.0	3.0	11.0	16.0	9.0	0.0
3	0.0	0.0	7.0	15.0	13.0	1.0	0.0	0.0	0.0	8.0	...	9.0	0.0	0.0	0.0	7.0	13.0	13.0	9.0	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	0.0	0.0	0.0	0.0	2.0	16.0	4.0	0.0	0.0

5 rows × 64 columns

In [69]: `df.shape`

Out[69]: `(1797, 64)`

In [70]: `df['target']=digits.target`

In [71]: `df[0:12]`

Out[71]:

	0	1	2	3	4	5	6	7	8	9	...	55	56	57	58	59	60	61	62	63	target
0	0.0	0.0	5.0	13.0	9.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	6.0	13.0	10.0	0.0	0.0	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	0.0	0.0	0.0	11.0	16.0	10.0	0.0	0.0	1
2	0.0	0.0	0.0	4.0	15.0	12.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	3.0	11.0	16.0	9.0	0.0	2
3	0.0	0.0	7.0	15.0	13.0	1.0	0.0	0.0	0.0	8.0	...	0.0	0.0	0.0	7.0	13.0	13.0	9.0	0.0	0.0	3
4	0.0	0.0	0.0	1.0	11.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	2.0	16.0	4.0	0.0	0.0	4
5	0.0	0.0	12.0	10.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	9.0	16.0	16.0	10.0	0.0	0.0	5
6	0.0	0.0	0.0	12.0	13.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	1.0	9.0	15.0	11.0	3.0	0.0	6
7	0.0	0.0	7.0	8.0	13.0	16.0	15.0	1.0	0.0	0.0	...	0.0	0.0	0.0	13.0	5.0	0.0	0.0	0.0	0.0	7
8	0.0	0.0	9.0	14.0	8.0	1.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	11.0	16.0	15.0	11.0	1.0	0.0	8
9	0.0	0.0	11.0	12.0	0.0	0.0	0.0	0.0	2.0	...	...	0.0	0.0	0.0	9.0	12.0	13.0	3.0	0.0	0.0	9
10	0.0	0.0	1.0	9.0	15.0	11.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	1.0	10.0	13.0	3.0	0.0	0.0	0
11	0.0	0.0	0.0	0.0	14.0	13.0	1.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	1.0	13.0	16.0	1.0	0.0	1

12 rows × 65 columns

In [72]: `x=df.drop('target',axis='columns')
y=df.target`

In [73]: `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)`

In [80]: `from sklearn.ensemble import RandomForestClassifier
model=RandomForestClassifier(n_estimators=70)
model.fit(x_train,y_train)`

Out[80]:

RandomForestClassifier

RandomForestClassifier(n\_estimators=70)

In [81]: `model.score(x_test, y_test)`

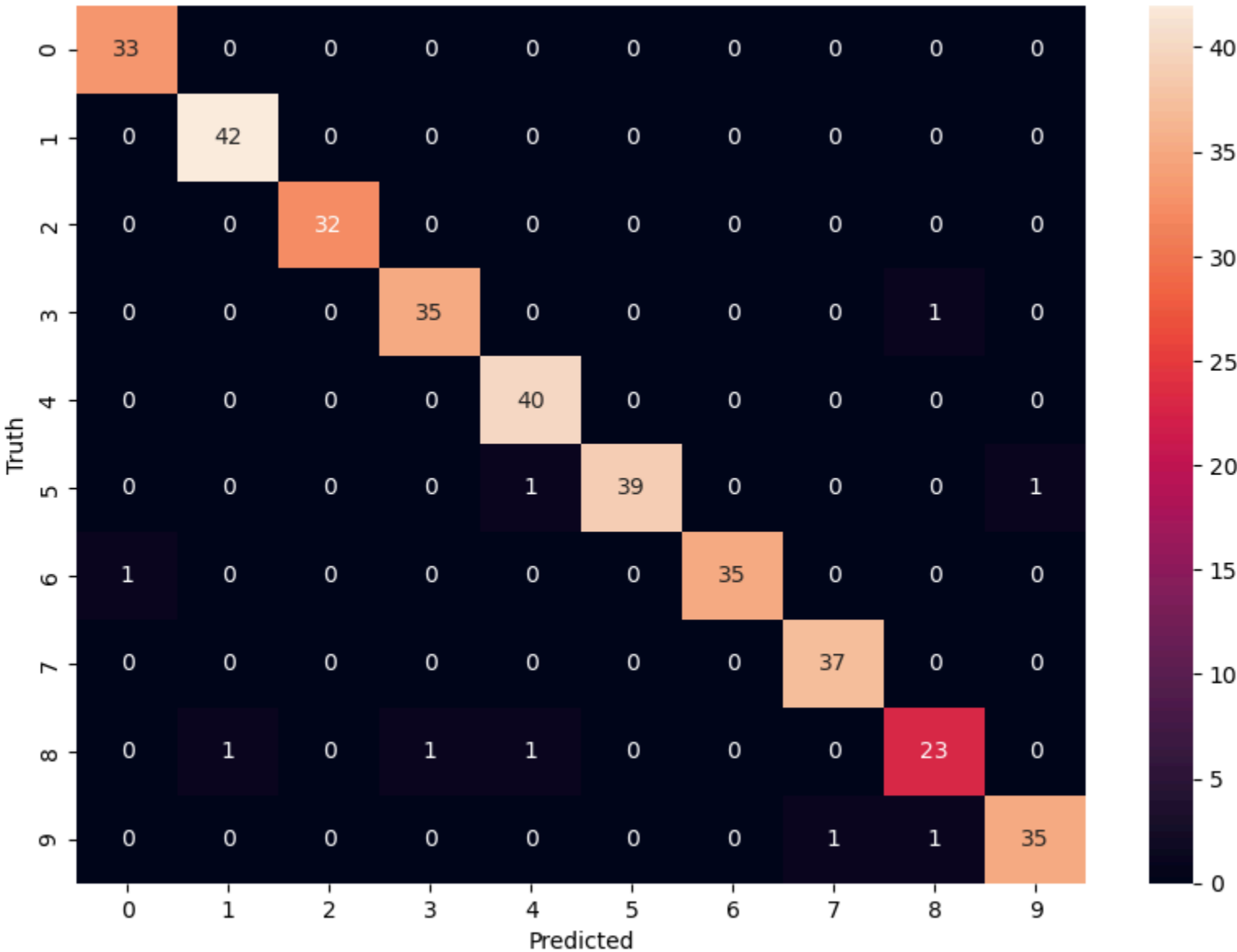
Out[81]: `0.975`

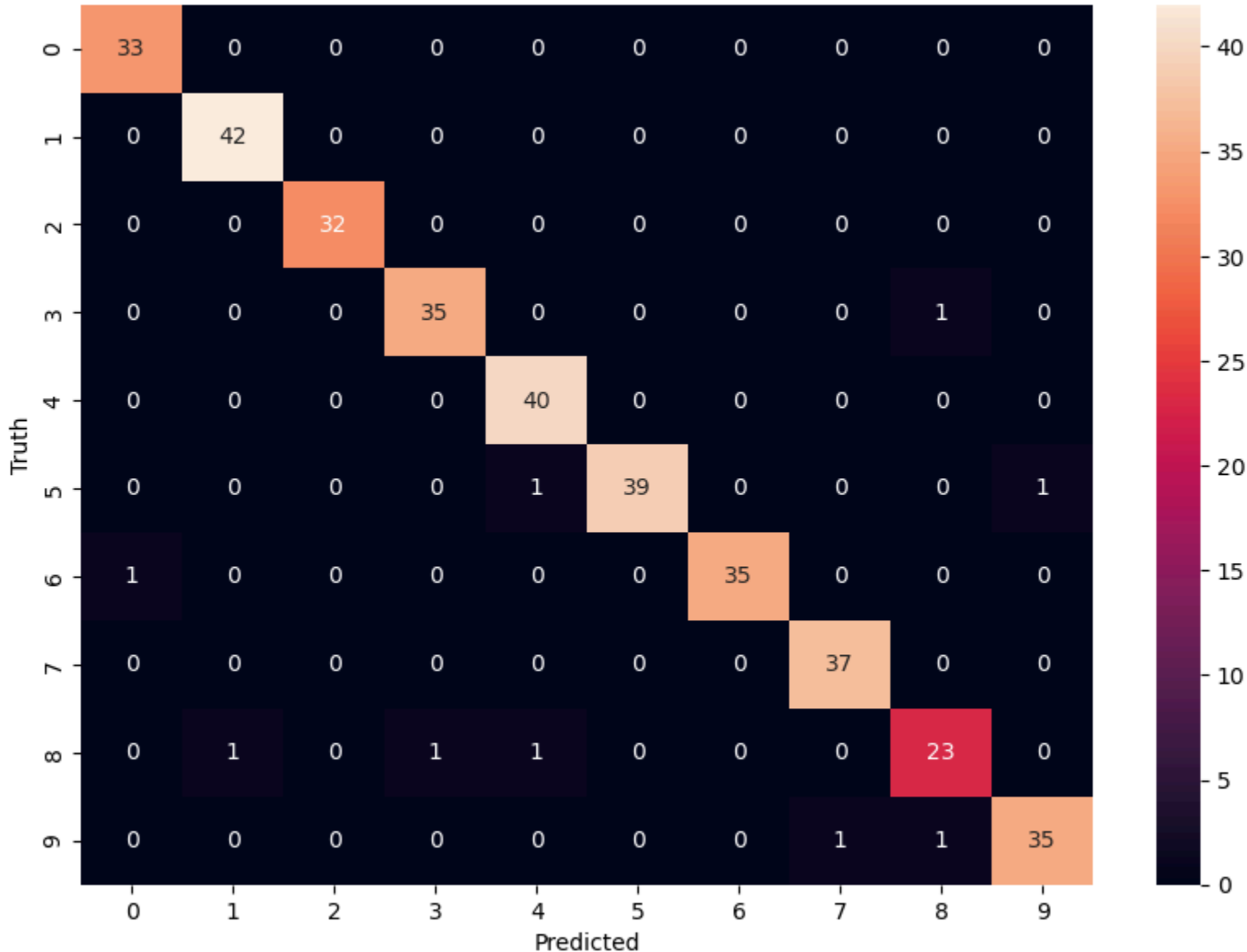
In [82]: `y_predicted=model.predict(x_test)`

In [83]: `from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_predicted)
cm`

Out[83]: `array([[33, 0, 0, 0, 0, 0, 0, 0, 0, 0],
 [ 0, 42, 0, 0, 0, 0, 0, 0, 0, 0],
 [ 0, 0, 32, 0, 0, 0, 0, 0, 0, 0],
 [ 0, 0, 0, 35, 0, 0, 0, 0, 1, 0],
 [ 0, 0, 0, 0, 40, 0, 0, 0, 0, 0],
 [ 0, 0, 0, 0, 1, 39, 0, 0, 0, 1],
 [ 1, 0, 0, 0, 0, 0, 35, 0, 0, 0],
 [ 0, 0, 0, 0, 0, 0, 0, 37, 0, 0],
 [ 0, 1, 0, 1, 1, 0, 0, 0, 23, 0],
 [ 0, 0, 0, 0, 0, 0, 1, 1, 35]]) dtype=int64)`

In [85]: `import seaborn as sn
plt.figure(figsize=(10,7))
sn.heatmap(cm, annot=True)
plt.xlabel('Predicted')
plt.ylabel('Truth')
plt.show()`





In [ ]: