

Exercise 7

Train SVM classifier using sklearn digits dataset (i.e. from sklearn.datasets import load_digits) and then,

1. Measure accuracy of your model using different kernels such as rbf and linear.
2. Tune your model further using regularization and gamma parameters and try to come up with highest accuracy score
3. Use 80% of samples as training data size

```
In [1]: import pandas as pd
import matplotlib.pyplot as plt
from sklearn.svm import SVC
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_digits
```

```
In [2]: data = load_digits()
dir(data)
```

```
Out[2]: ['DESCR', 'data', 'feature_names', 'frame', 'images', 'target', 'target_names']
```

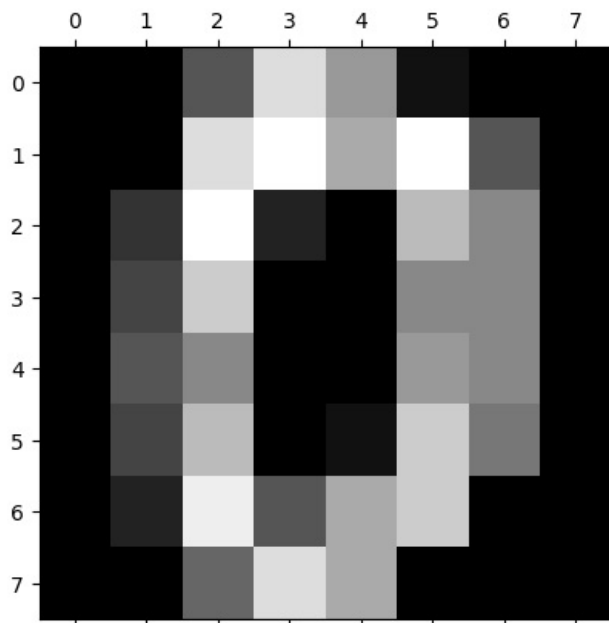
```
In [3]: data.data[0]
```

```
Out[3]: array([ 0.,  0.,  5., 13.,  9.,  1.,  0.,  0.,  0.,  0., 13., 15., 10.,
        15.,  5.,  0.,  0.,  3., 15.,  2.,  0., 11.,  8.,  0.,  0.,  4.,
        12.,  0.,  0.,  8.,  8.,  0.,  0.,  5.,  8.,  0.,  0.,  9.,  8.,
         0.,  0.,  4., 11.,  0.,  1., 12.,  7.,  0.,  0.,  2., 14.,  5.,
        10., 12.,  0.,  0.,  0.,  0.,  6., 13., 10.,  0.,  0.,  0.])
```

```
In [4]: plt.gray()
plt.matshow(data.images[0])
```

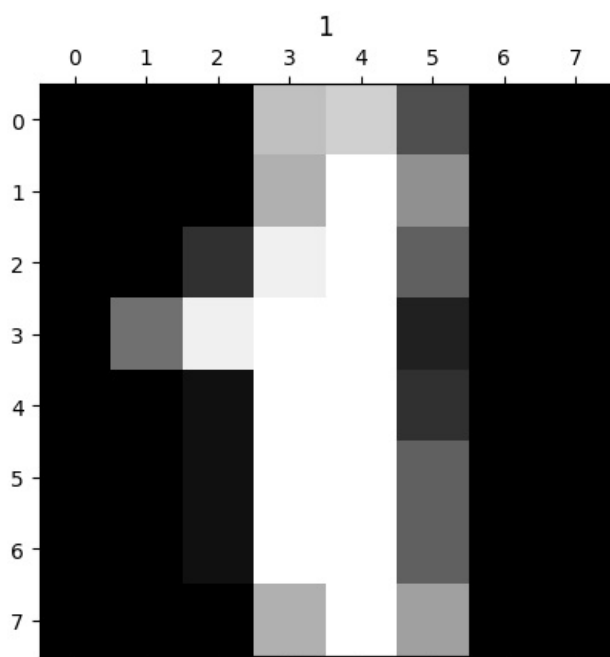
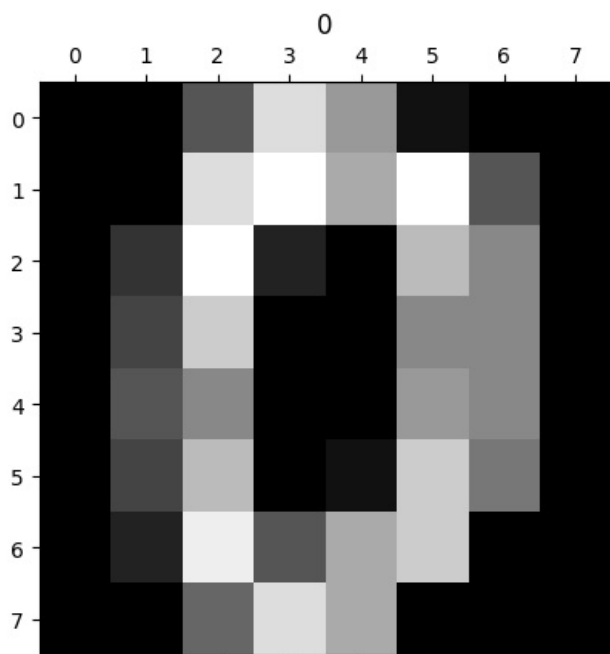
```
Out[4]: <matplotlib.image.AxesImage at 0x25e78f03550>
```

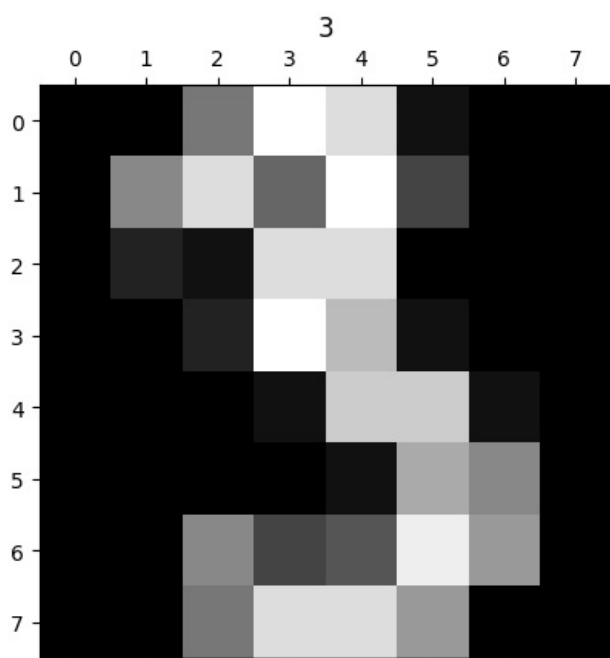
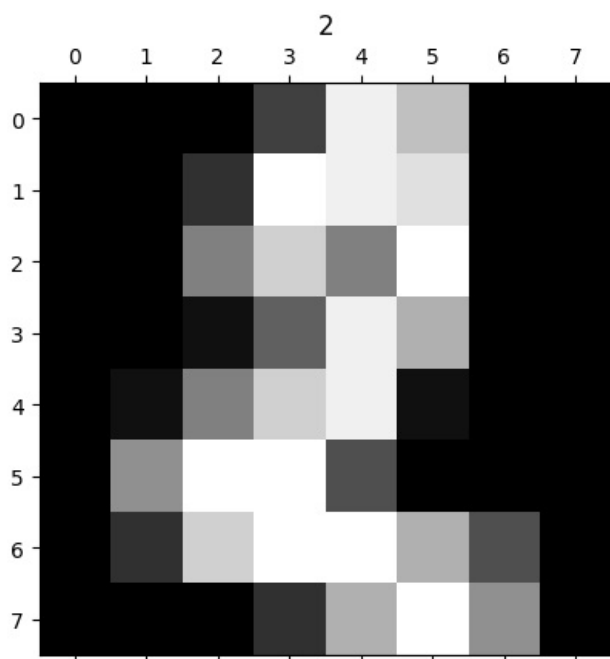
<Figure size 640x480 with 0 Axes>

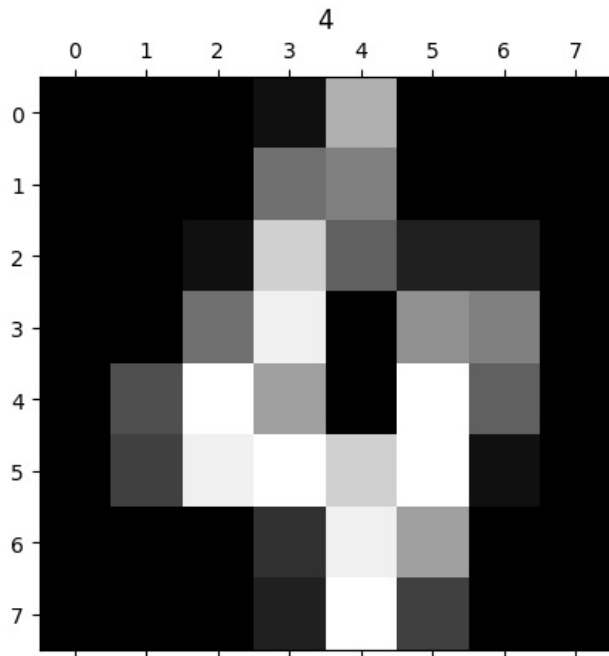


```
In [5]: plt.gray()
for i in range(5):
    plt.matshow(data.images[i])
    plt.title(data.target_names[i])
```

<Figure size 640x480 with 0 Axes>







Splitting to Training and Testing datasets

```
In [6]: X_train, X_test, y_train, y_test = train_test_split(data.data, data.target, test_size=0.2)
```

Model training and tuning

```
In [7]: model = SVC(C=10)
```

```
In [8]: model.fit(X_train, y_train)
```

```
Out[8]: SVC
SVC(C=10)
```

Measuring accuracy score

```
In [9]: model.score(X_test, y_test)
```

```
Out[9]: 0.9916666666666667
```

Highest accuracy score

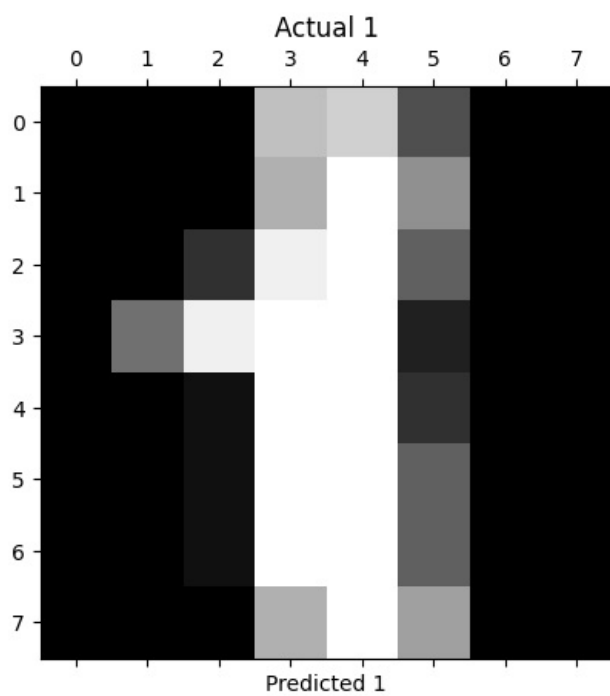
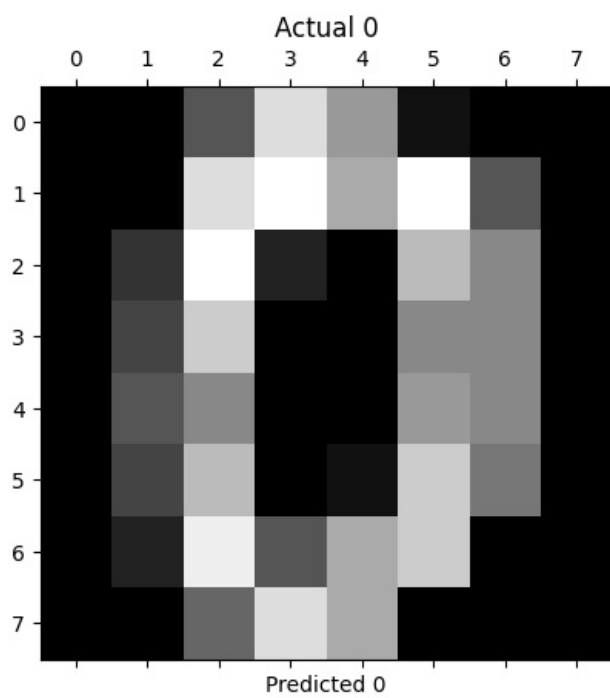
```
0.9861111111111112
```

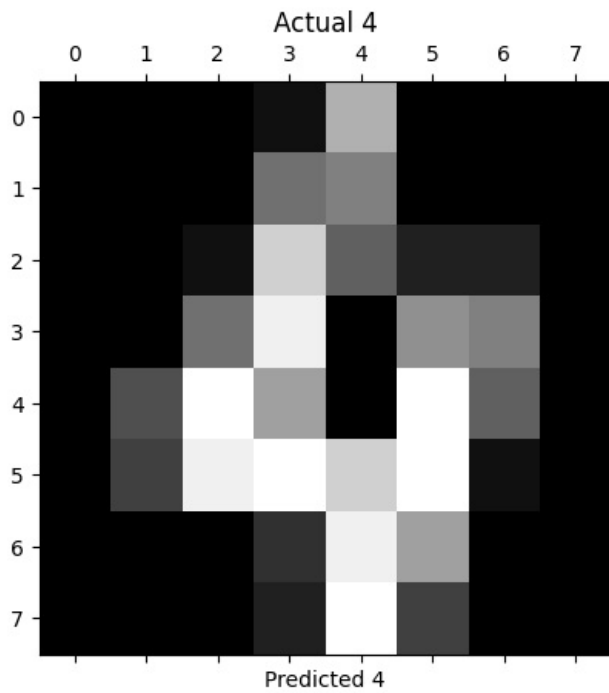
```
In [10]: model.predict([data.data[1]])
```

```
Out[10]: array([1])
```

```
In [11]: plt.gray()
for i in range(5):
    plt.matshow(data.images[i])
    plt.title(f"Actual {data.target[i]}")
    plt.xlabel(f"Predicted {int(model.predict([data.data[i]]))}")
```

<Figure size 640x480 with 0 Axes>





```
In [12]: predict = model.predict(X_test)
correct_count = 0
wrong_count = 0
for i in range(len(predict)):
    print(f"Actual {y_test[i]} Predicted {predict[i]}")
    if y_test[i] == predict[i]:
        correct_count += 1
    else:
        wrong_count += 1
print(f"Correct prediction {correct_count}")
print(f"Wrong prediction {wrong_count}")
```

```
Actual 3 Predicted 3
Actual 6 Predicted 6
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 6 Predicted 6
Actual 6 Predicted 6
Actual 8 Predicted 8
Actual 1 Predicted 1
Actual 1 Predicted 1
Actual 5 Predicted 5
Actual 9 Predicted 9
Actual 0 Predicted 0
Actual 7 Predicted 7
Actual 9 Predicted 9
Actual 2 Predicted 2
Actual 8 Predicted 8
Actual 6 Predicted 6
Actual 5 Predicted 5
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 1 Predicted 1
Actual 4 Predicted 4
Actual 1 Predicted 1
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 6 Predicted 6
Actual 7 Predicted 7
Actual 0 Predicted 0
Actual 8 Predicted 8
Actual 5 Predicted 5
Actual 4 Predicted 4
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 5 Predicted 5
Actual 3 Predicted 3
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 1 Predicted 1
```

Actual 8 Predicted 8
Actual 8 Predicted 8
Actual 0 Predicted 0
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 3 Predicted 3
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 0 Predicted 0
Actual 7 Predicted 7
Actual 6 Predicted 6
Actual 4 Predicted 4
Actual 6 Predicted 6
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 5 Predicted 5
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 7 Predicted 7
Actual 1 Predicted 1
Actual 4 Predicted 4
Actual 2 Predicted 2
Actual 3 Predicted 3
Actual 1 Predicted 1
Actual 6 Predicted 6
Actual 7 Predicted 7
Actual 9 Predicted 9
Actual 7 Predicted 7
Actual 1 Predicted 1
Actual 5 Predicted 5
Actual 9 Predicted 9
Actual 7 Predicted 7
Actual 6 Predicted 6
Actual 9 Predicted 9
Actual 8 Predicted 8
Actual 2 Predicted 2
Actual 0 Predicted 0
Actual 5 Predicted 5
Actual 2 Predicted 2
Actual 1 Predicted 1
Actual 9 Predicted 9
Actual 8 Predicted 8
Actual 6 Predicted 6
Actual 2 Predicted 2
Actual 7 Predicted 7
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 9 Predicted 9
Actual 9 Predicted 9
Actual 9 Predicted 9
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 5 Predicted 5
Actual 7 Predicted 7
Actual 1 Predicted 1
Actual 5 Predicted 5
Actual 2 Predicted 2
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 0 Predicted 0
Actual 0 Predicted 0
Actual 0 Predicted 0
Actual 4 Predicted 4
Actual 1 Predicted 1
Actual 0 Predicted 0
Actual 4 Predicted 4
Actual 1 Predicted 1
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 0 Predicted 0
Actual 5 Predicted 5
Actual 8 Predicted 8
Actual 3 Predicted 3
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 7 Predicted 7
Actual 2 Predicted 2
Actual 4 Predicted 4
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 8 Predicted 8

Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 6 Predicted 6
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 5 Predicted 5
Actual 9 Predicted 9
Actual 6 Predicted 6
Actual 6 Predicted 6
Actual 2 Predicted 2
Actual 5 Predicted 5
Actual 8 Predicted 8
Actual 0 Predicted 0
Actual 2 Predicted 2
Actual 4 Predicted 4
Actual 9 Predicted 9
Actual 7 Predicted 7
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 2 Predicted 2
Actual 0 Predicted 0
Actual 0 Predicted 0
Actual 1 Predicted 1
Actual 4 Predicted 4
Actual 6 Predicted 6
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 5 Predicted 6
Actual 6 Predicted 6
Actual 9 Predicted 9
Actual 5 Predicted 5
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 4 Predicted 4
Actual 5 Predicted 5
Actual 9 Predicted 9
Actual 2 Predicted 2
Actual 4 Predicted 4
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 9 Predicted 9
Actual 3 Predicted 3
Actual 7 Predicted 7
Actual 9 Predicted 9
Actual 5 Predicted 5
Actual 4 Predicted 4
Actual 1 Predicted 1
Actual 5 Predicted 5
Actual 6 Predicted 6
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 0 Predicted 0
Actual 6 Predicted 6
Actual 5 Predicted 5
Actual 1 Predicted 1
Actual 9 Predicted 9
Actual 1 Predicted 1
Actual 6 Predicted 6
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 0 Predicted 0
Actual 4 Predicted 4
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 5 Predicted 5
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 7 Predicted 7
Actual 2 Predicted 2
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 8 Predicted 8
Actual 5 Predicted 5
Actual 1 Predicted 1
Actual 8 Predicted 8
Actual 8 Predicted 8
Actual 1 Predicted 1
Actual 7 Predicted 7
Actual 9 Predicted 9
Actual 3 Predicted 2

Actual 9 Predicted 9
Actual 4 Predicted 4
Actual 1 Predicted 1
Actual 9 Predicted 9
Actual 1 Predicted 1
Actual 4 Predicted 4
Actual 9 Predicted 9
Actual 5 Predicted 5
Actual 7 Predicted 7
Actual 3 Predicted 3
Actual 7 Predicted 7
Actual 4 Predicted 4
Actual 5 Predicted 5
Actual 9 Predicted 9
Actual 0 Predicted 0
Actual 6 Predicted 6
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 6 Predicted 6
Actual 2 Predicted 2
Actual 2 Predicted 2
Actual 5 Predicted 5
Actual 1 Predicted 1
Actual 9 Predicted 9
Actual 0 Predicted 0
Actual 1 Predicted 1
Actual 8 Predicted 8
Actual 1 Predicted 1
Actual 7 Predicted 7
Actual 4 Predicted 4
Actual 8 Predicted 8
Actual 1 Predicted 1
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 7 Predicted 7
Actual 2 Predicted 2
Actual 8 Predicted 8
Actual 9 Predicted 9
Actual 2 Predicted 2
Actual 3 Predicted 3
Actual 0 Predicted 0
Actual 2 Predicted 2
Actual 7 Predicted 7
Actual 6 Predicted 6
Actual 1 Predicted 1
Actual 7 Predicted 7
Actual 5 Predicted 5
Actual 7 Predicted 7
Actual 9 Predicted 9
Actual 2 Predicted 2
Actual 9 Predicted 9
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 6 Predicted 6
Actual 4 Predicted 4
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 7 Predicted 7
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 0 Predicted 0
Actual 8 Predicted 8
Actual 1 Predicted 1
Actual 8 Predicted 8
Actual 6 Predicted 6
Actual 9 Predicted 9
Actual 9 Predicted 9
Actual 9 Predicted 9
Actual 6 Predicted 6
Actual 1 Predicted 1
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 3 Predicted 3
Actual 8 Predicted 8
Actual 6 Predicted 6
Actual 0 Predicted 0

Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 5 Predicted 5
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 0 Predicted 0
Actual 4 Predicted 4
Actual 2 Predicted 2
Actual 1 Predicted 1
Actual 0 Predicted 0
Actual 4 Predicted 4
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 0 Predicted 0
Actual 2 Predicted 2
Actual 4 Predicted 4
Actual 1 Predicted 1
Actual 3 Predicted 3
Actual 4 Predicted 4
Actual 6 Predicted 6
Actual 6 Predicted 6
Actual 7 Predicted 7
Actual 9 Predicted 9
Actual 3 Predicted 3
Actual 3 Predicted 3
Actual 3 Predicted 3
Actual 1 Predicted 1
Actual 1 Predicted 1
Actual 2 Predicted 2
Actual 5 Predicted 5
Actual 6 Predicted 6
Actual 0 Predicted 0
Actual 9 Predicted 9
Actual 2 Predicted 2
Actual 1 Predicted 1
Actual 8 Predicted 8
Actual 8 Predicted 8
Actual 1 Predicted 1
Actual 5 Predicted 5
Actual 8 Predicted 8
Actual 3 Predicted 3
Actual 6 Predicted 6
Actual 4 Predicted 4
Actual 7 Predicted 7
Actual 6 Predicted 6
Actual 8 Predicted 1
Actual 4 Predicted 4
Actual 8 Predicted 8
Actual 4 Predicted 4
Actual 3 Predicted 3
Actual 7 Predicted 7
Actual 3 Predicted 3
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 5 Predicted 5
Actual 4 Predicted 4
Actual 2 Predicted 2
Actual 5 Predicted 5
Actual 2 Predicted 2
Actual 2 Predicted 2
Actual 0 Predicted 0
Actual 5 Predicted 5
Actual 0 Predicted 0
Actual 1 Predicted 1
Actual 1 Predicted 1
Actual 8 Predicted 8
Actual 7 Predicted 7
Actual 8 Predicted 8
Actual 2 Predicted 2
Actual 2 Predicted 2
Correct prediction 357
Wrong prediction 3

```
In [14]: len(X_test)
```

```
Out[14]: 360
```

```
In [15]: y_test
```

```
Out[15]: array([3, 6, 4, 3, 6, 6, 8, 1, 1, 5, 9, 0, 7, 9, 2, 8, 6, 5, 6, 0, 9, 1,
4, 1, 0, 9, 6, 7, 0, 8, 5, 4, 5, 0, 0, 9, 5, 3, 4, 3, 1, 8, 8, 0,
5, 0, 3, 6, 0, 9, 0, 7, 6, 4, 6, 3, 8, 5, 6, 0, 7, 1, 4, 2, 3, 1,
6, 7, 9, 7, 1, 5, 9, 7, 6, 9, 8, 2, 0, 5, 2, 1, 9, 8, 6, 2, 7, 7,
8, 9, 9, 9, 3, 8, 5, 7, 1, 5, 2, 7, 8, 7, 0, 0, 0, 4, 1, 0, 4, 1,
4, 3, 0, 5, 8, 3, 6, 0, 7, 2, 4, 7, 8, 8, 0, 9, 6, 7, 8, 5, 9, 6,
6, 2, 5, 8, 0, 2, 4, 9, 7, 6, 0, 9, 2, 0, 0, 1, 4, 6, 8, 7, 5, 6,
9, 5, 8, 7, 4, 5, 9, 2, 4, 4, 3, 9, 3, 7, 9, 5, 4, 1, 5, 6, 8, 7,
3, 8, 0, 6, 5, 1, 9, 1, 6, 7, 8, 0, 4, 0, 9, 5, 6, 0, 7, 2, 3, 8,
8, 5, 1, 8, 8, 1, 7, 9, 3, 9, 4, 1, 9, 1, 4, 9, 5, 7, 3, 7, 4, 5,
9, 0, 6, 8, 7, 6, 2, 2, 5, 1, 9, 0, 1, 8, 1, 7, 4, 8, 1, 5, 0, 9,
4, 3, 8, 7, 7, 2, 8, 9, 2, 3, 0, 2, 7, 6, 1, 7, 5, 7, 9, 2, 9, 7,
8, 6, 4, 5, 0, 7, 3, 8, 4, 3, 0, 8, 1, 8, 6, 9, 9, 9, 6, 1, 4, 3,
3, 8, 6, 0, 4, 3, 5, 8, 7, 0, 4, 2, 1, 0, 4, 5, 0, 0, 2, 4, 1, 3,
4, 6, 6, 7, 9, 3, 3, 3, 1, 1, 2, 5, 6, 0, 9, 2, 1, 8, 8, 1, 5, 8,
3, 6, 4, 7, 6, 8, 4, 8, 4, 3, 7, 3, 5, 0, 5, 4, 2, 5, 2, 2, 0, 5,
0, 1, 1, 8, 7, 8, 2, 2])
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js