```python
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import
Dense,Conv2D,MaxPooling2D,Flatten,Dropout
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.datasets import mnist
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.models import load_model
```

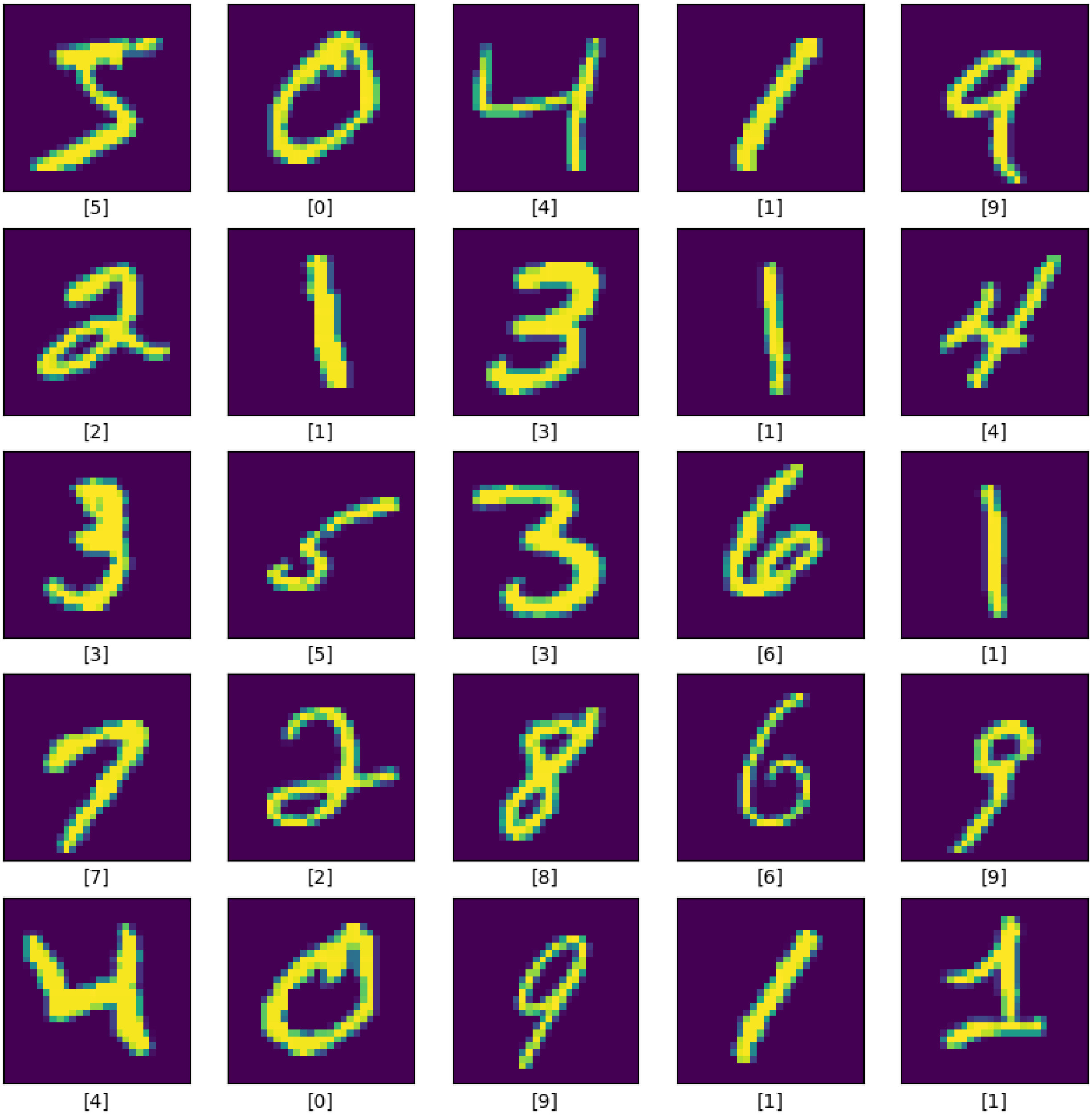
```python
(xtrain,ytrain),(xtest,ytest)=mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-
keras-datasets/mnist.npz
11490434/11490434 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
```

```python
xtrain.shape,ytrain.shape,xtest.shape,ytest.shape
```

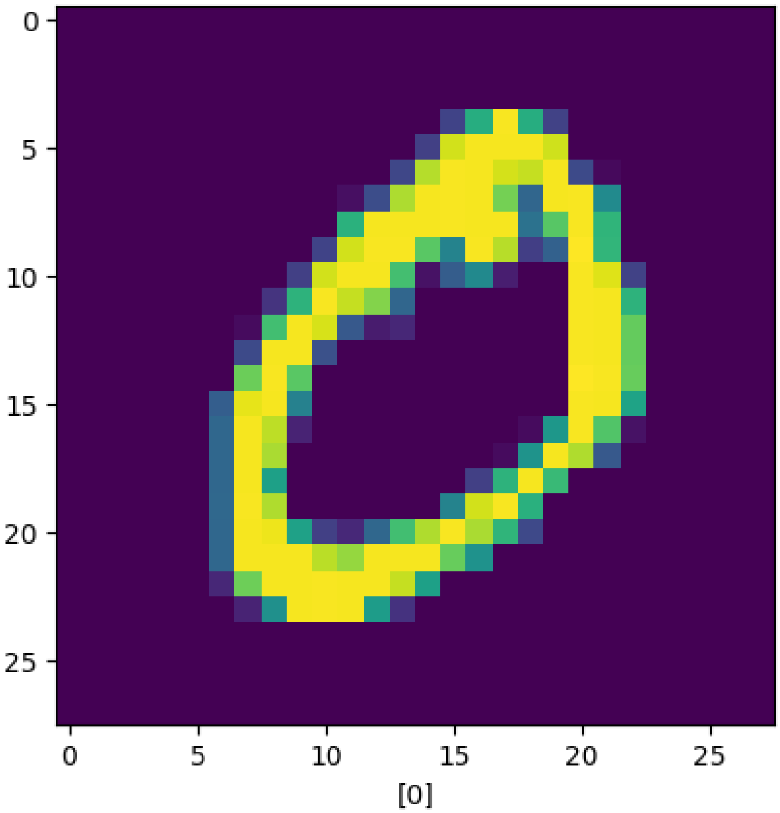
```
((60000, 28, 28), (60000,), (10000, 28, 28), (10000,))
```

```python
plt.figure(figsize=(10,10))
for i in range(25):
    plt.subplot(5,5,i+1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(xtrain[i])
    # The CIFAR labels happen to be arrays,
    # which is why you need the extra index
    plt.xlabel([ytrain[i]])
plt.show()
```



```python
def plot_pics(x, y, index):
    plt.imshow(x[index])
    plt.xlabel([y[index]])

plot_pics(xtrain,ytrain,1)
```



```
xtrain = xtrain.reshape(-1, 28, 28, 1)
xtest=xtest.reshape(-1, 28, 28, 1,)

xtrain=xtrain/255.0
xtest=xtest/255.0

ytrain=to_categorical(ytrain,10)
ytest=to_categorical(ytest,10)

model=Sequential()

model.add(Conv2D(32,(3,3),activation='relu',input_shape=(28,28,1)))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(64,(3,3),activation='relu',input_shape=(28,28,1)))
model.add(MaxPooling2D((2,2)))

model.add(Conv2D(128,(3,3),activation='relu',input_shape=(28,28,1)))
model.add(MaxPooling2D((2,2)))
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/
convolutional/base_conv.py:107: UserWarning: Do not pass an
`input_shape`/`input_dim` argument to a layer. When using Sequential
models, prefer using an `Input(shape)` object as the first layer in
the model instead.
```

```
  super().__init__(activity_regularizer=activity_regularizer,
**kwargs)

model.add(Flatten())
model.add(Dense(256,activation='relu'))
model.add(Dense(128,activation='relu'))
model.add(Dense(10,activation='softmax'))

model.summary()

Model: "sequential"

┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳
━━━━━━━━━━━━━━━━━┓
┃ Layer (type)                          ┃ Output Shape                 ┃
Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇
━━━━━━━━━━━━━━━━━┩
│ conv2d (Conv2D)                       │ (None, 26, 26, 32)           │
320 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ max_pooling2d (MaxPooling2D)          │ (None, 13, 13, 32)           │
0 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ conv2d_1 (Conv2D)                     │ (None, 11, 11, 64)           │
18,496 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ max_pooling2d_1 (MaxPooling2D)        │ (None, 5, 5, 64)             │
0 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ conv2d_2 (Conv2D)                     │ (None, 3, 3, 128)            │
73,856 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ max_pooling2d_2 (MaxPooling2D)        │ (None, 1, 1, 128)            │
0 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ flatten (Flatten)                     │ (None, 128)                  │
0 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
│ dense (Dense)                         │ (None, 256)                  │
33,024 │
├───────────────────────────────────────┼──────────────────────────────┼
─────────────────┤
```

```
│ dense_1 (Dense)                      │ (None, 128)                 │
32,896 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_2 (Dense)                      │ (None, 10)                  │
1,290 │
└──────────────────────────────────────┴─────────────────────────────┴
─────────────────┘

 Total params: 159,882 (624.54 KB)

 Trainable params: 159,882 (624.54 KB)

 Non-trainable params: 0 (0.00 B)
```

```
model.compile(optimizer='adam',loss='categorical_crossentropy',metrics
=['accuracy'])
```

```
m1=model.fit(xtrain,ytrain,epochs=10,validation_data=(xtest,ytest),val
idation_split=0.2)
```

```
Epoch 1/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 13s 4ms/step - accuracy: 0.8633 - loss:
0.4105 - val_accuracy: 0.9802 - val_loss: 0.0658
Epoch 2/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 14s 3ms/step - accuracy: 0.9797 - loss:
0.0667 - val_accuracy: 0.9774 - val_loss: 0.0724
Epoch 3/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 6s 3ms/step - accuracy: 0.9852 - loss:
0.0460 - val_accuracy: 0.9879 - val_loss: 0.0452
Epoch 4/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 10s 3ms/step - accuracy: 0.9893 - loss:
0.0353 - val_accuracy: 0.9871 - val_loss: 0.0440
Epoch 5/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 6s 3ms/step - accuracy: 0.9913 - loss:
0.0281 - val_accuracy: 0.9822 - val_loss: 0.0621
Epoch 6/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 5s 3ms/step - accuracy: 0.9920 - loss:
0.0243 - val_accuracy: 0.9856 - val_loss: 0.0557
Epoch 7/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 6s 3ms/step - accuracy: 0.9942 - loss:
0.0195 - val_accuracy: 0.9894 - val_loss: 0.0424
Epoch 8/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - accuracy: 0.9947 - loss:
0.0162 - val_accuracy: 0.9889 - val_loss: 0.0557
Epoch 9/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 5s 3ms/step - accuracy: 0.9943 - loss:
0.0183 - val_accuracy: 0.9866 - val_loss: 0.0665
Epoch 10/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 11s 3ms/step - accuracy: 0.9959 - loss:
0.0145 - val_accuracy: 0.9874 - val_loss: 0.0558
```

```
ypred=model.predict(xtest)
```

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step
```

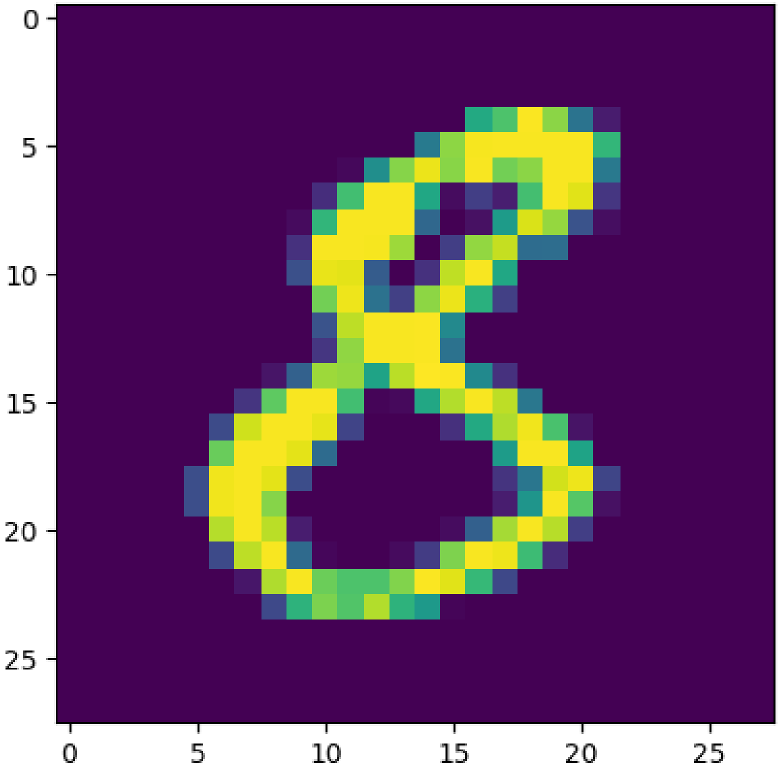
```
ypred
```

```
array([[5.12411554e-12, 8.94991388e-08, 1.42497720e-06, ...,
        9.99997616e-01, 2.43486842e-10, 7.36786987e-08],
       [2.83063017e-10, 5.53727528e-14, 1.00000000e+00, ...,
        2.04090123e-09, 1.30095790e-09, 1.86898730e-15],
       [2.90309607e-08, 9.99997377e-01, 1.38422305e-08, ...,
        1.98499976e-07, 7.93900028e-07, 3.90970145e-09],
       ...,
       [5.05278521e-19, 7.22510634e-15, 5.64773247e-16, ...,
        1.21358790e-14, 2.22275465e-09, 2.19341489e-11],
       [1.02611257e-12, 1.97863627e-14, 7.48100990e-14, ...,
        1.27629149e-13, 1.53589044e-10, 7.22185922e-10],
       [3.35784353e-06, 1.89585281e-09, 3.35665246e-10, ...,
        4.63142599e-13, 2.16324558e-09, 1.53126251e-07]],
dtype=float32)
```

```
ypred_lab = [np.argmax(element) for element in ypred]
ypred_lab
```

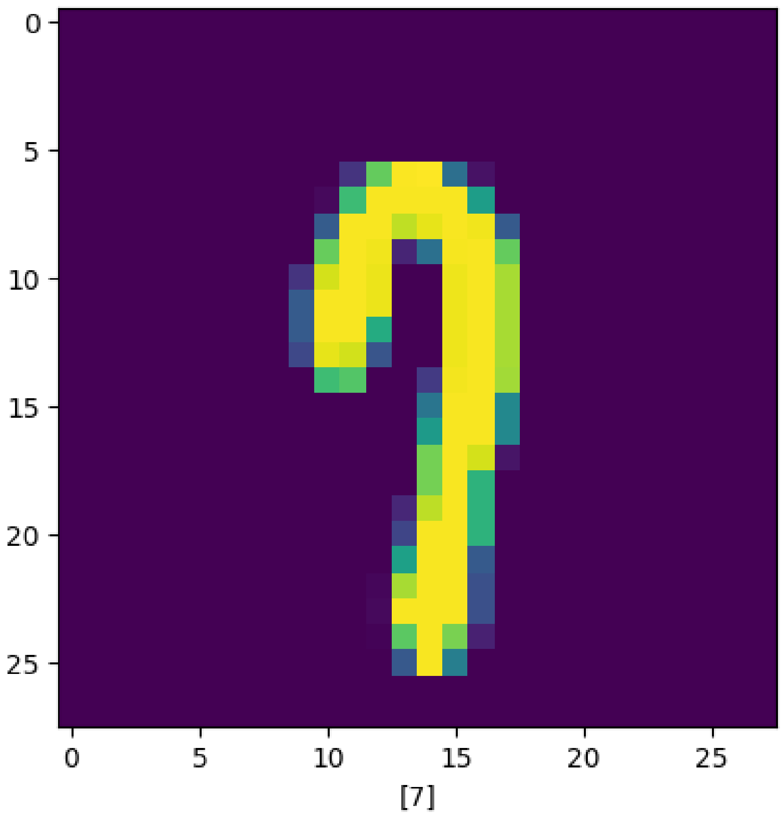
```
[7,
 2,
 1,
 0,
 4,
 1,
 4,
 9,
 5,
 9,
 0,
 6,
 9,
 0,
 1,
 5,
 9,
 7,
 3,
 4,
 9,
 6,
 6,
 5,
 4,
 0,
 7,
 4,
 0,
 1,
 3,
 1,
 3,
 4,
 7,
 2,
 7,
 1,
 2,
 1,
 1,
 7,
 4,
 2,
 3,
 5,
 1,
 2,
 4,
 4,
 6,
 3,
 5,
 5,
 6,
 0,
 4,
 1,
 9,
 5,
 7,
 8,
 9,
 3,
 7,
 4,
 6,
 4,
 3,
 0,
 7,
 0,
 2,
 9,
 1,
7,
3,
2,
9,
7,
7,
6,
2,
7,
8,
4,
7,
3,
6,
1,
3,
6,
9,
3,
1,
4,
1,
7,
6,
9,
6,
0,
5,
4,
9,
9,
2,
1,
9,
4,
8,
7,
3,
9,
7,
4,
4,
4,
9,
2,
5,
4,
7,
6,
 7,
 9,
 0,
 5,
 8,
 5,
 6,
 6,
 5,
 7,
 8,
 1,
 0,
 1,
 6,
 4,
 6,
 7,
 3,
 1,
 7,
 1,
 8,
 2,
 0,
 2,
 9,
 9,
 5,
 5,
 1,
 5,
 6,
 0,
 3,
 4,
 4,
 6,
 5,
 4,
 6,
 5,
 4,
 5,
 1,
 4,
 4,
 7,
 2,
 3,
 2,
 7,
 1,
 8,
 1,
 8,
 1,
 8,
 5,
 0,
 8,
 9,
 2,
 5,
 0,
 1,
 1,
 1,
 0,
 9,
 0,
 3,
 1,
 6,
 4,
 2,
 3,
 6,
 1,
 1,
 1,
 3,
 9,
 5,
 2,
 9,
 4,
 5,
 9,
 3,
 9,
 0,
 3,
 6,
 5,
 5,
 7,
 2,
 2,
 7,
 1,
 2,
 8,
 4,
 1,
 7,
 3,
 3,
 8,
 8,
 7,
 9,
 2,
 2,
 4,
 1,
 5,
 9,
 8,
 7,
 2,
 3,
 0,
 2,
 4,
 2,
 4,
 1,
 9,
 5,
 7,
 7,
 2,
 8,
 2,
 6,
 8,
 5,
 7,
 7,
 9,
 1,
 8,
 1,
 8,
 0,
 3,
 0,
 1,
 9,
 9,
 4,
 1,
 8,
 2,
 1,
 2,
 9,
 7,
 5,
 9,
 2,
 6,
 4,
 1,
 5,
 8,
 2,
 9,
 2,
 0,
 4,
 0,
 0,
 2,
 8,
 4,
 7,
 1,
 2,
 4,
 0,
 2,
 7,
 4,
 3,
 3,
 0,
 0,
 3,
 1,
 9,
 6,
 5,
 2,
 5,
 9,
 7,
 9,
 3,
 0,
 4,
 2,
 0,
 7,
 1,
 1,
 2,
 1,
 5,
 3,
 3,
 9,
 7,
 8,
 6,
 3,
 6,
 1,
 3,
 8,
 1,
 0,
 5,
 1,
 3,
 1,
 5,
 5,
 6,
 1,
 8,
 5,
 1,
 7,
 9,
 4,
 6,
 2,
 2,
 5,
 0,
 6,
 5,
 6,
 3,
 7,
 2,
 0,
 8,
 8,
 5,
 4,
 1,
 1,
 4,
 0,
 3,
 3,
 7,
 6,
 1,
 6,
 2,
 1,
 9,
 2,
 8,
 6,
 1,
 9,
 5,
 2,
 5,
 4,
 4,
 2,
 8,
 3,
 8,
 2,
 4,
 5,
 0,
 3,
 1,
 7,
 7,
 5,
 7,
 9,
 7,
 1,
 9,
2,
1,
4,
2,
9,
2,
0,
4,
9,
1,
4,
8,
1,
8,
4,
5,
9,
9,
8,
3,
7,
6,
0,
0,
3,
0,
2,
0,
6,
4,
9,
5,
5,
3,
2,
3,
9,
1,
2,
6,
8,
0,
5,
6,
6,
6,
3,
8,
8,
2,
7,
5,
8,
9,
6,
1,
8,
4,
1,
2,
5,
9,
1,
9,
7,
5,
4,
0,
8,
9,
9,
1,
0,
5,
2,
3,
7,
8,
9,
4,
0,
6,
3,
9,
5,
2,
1,
3,
1,
3,
6,
5,
7,
4,
2,
2,
6,
3,
2,
6,
5,
4,
8,
9,
3,
1,
3,
0,
3,
8,
3,
1,
9,
3,
4,
4,
6,
4,
2,
1,
8,
2,
5,
4,
8,
8,
4,
0,
0,
2,
3,
2,
7,
7,
0,
8,
7,
4,
4,
7,
9,
6,
9,
0,
9,
8,
0,
4,
6,
0,
6,
3,
5,
4,
8,
3,
3,
9,
3,
3,
3,
7,
8,
0,
2,
2,
1,
7,
0,
6,
5,
4,
3,
8,
0,
9,
6,
3,
8,
0,
9,
9,
6,
8,
6,
8,
5,
7,
8,
6,
0,
2,
4,
0,
2,
2,
3,
 1,
 9,
 7,
 5,
 1,
 0,
 8,
 4,
 6,
 2,
 6,
 7,
 9,
 3,
 2,
 9,
 8,
 2,
 2,
 9,
 2,
 7,
 3,
 5,
 9,
 1,
 8,
 0,
 2,
 0,
 5,
 6,
 1,
 3,
 7,
 6,
 7,
 1,
 2,
 5,
 8,
 0,
 3,
 7,
 7,
 4,
 0,
 9,
 1,
 8,
 6,
 7,
 7,
 4,
 3,
 4,
 9,
 1,
 9,
 5,
 1,
 7,
 3,
 9,
 7,
 6,
 9,
 1,
 3,
 2,
 8,
 3,
 3,
 6,
 7,
 2,
 8,
 5,
 8,
 5,
 1,
 1,
 4,
 4,
 3,
 1,
 0,
 7,
 7,
 0,
 7,
 9,
 8,
 4,
 8,
 5,
 5,
 4,
 0,
 8,
 2,
 9,
 0,
 8,
 4,
 5,
 0,
 4,
 0,
 6,
 1,
 5,
 3,
 2,
 6,
 7,
 2,
 6,
 9,
 3,
 1,
 4,
 6,
 2,
 5,
 4,
 2,
 0,
 6,
 2,
 1,
 7,
 3,
 4,
 1,
 0,
 5,
 4,
 3,
 1,
 1,
 7,
 4,
 9,
 9,
 4,
 8,
 4,
 0,
 2,
 4,
 5,
 1,
 1,
 6,
 4,
 7,
 1,
 9,
 4,
 2,
 4,
 1,
 5,
 5,
 3,
 8,
 3,
 1,
 4,
 5,
 6,
 8,
 9,
 4,
 1,
 5,
 3,
 8,
 0,
 3,
 2,
 5,
 1,
 2,
 8,
 3,
 4,
 4,
 0,
 8,
 8,
 3,
 3,
 1,
 7,
 3,
 5,
 9,
 6,
 3,
 2,
 6,
 1,
 3,
 6,
 0,
 7,
 2,
 1,
 7,
 1,
 4,
 2,
 4,
 2,
 1,
 7,
 9,
 6,
 1,
 1,
 2,
 4,
 8,
 1,
 7,
 7,
 4,
 8,
 0,
 7,
 3,
 1,
 3,
 1,
 0,
 7,
 7,
 0,
 3,
 5,
 5,
 2,
 7,
 6,
 6,
 9,
 2,
 8,
 3,
 5,
 2,
 2,
 5,
 6,
 0,
 8,
 2,
 9,
 2,
 8,
 8,
 8,
 8,
 7,
 4,
 9,
 3,
 0,
 6,
 6,
 3,
 2,
 1,
 3,
 2,
 2,
 9,
 3,
 0,
 0,
 5,
 7,
 8,
 1,
 4,
 4,
 6,
 0,
 2,
 9,
 1,
 4,
 7,
 4,
 7,
 3,
 9,
 8,
 8,
 4,
 7,
 1,
 2,
 1,
 2,
 2,
 3,
 2,
 3,
 2,
 3,
 9,
 1,
 7,
 4,
 0,
 3,
 5,
 5,
 8,
 6,
 5,
 2,
 6,
 7,
 6,
 6,
 3,
 2,
 7,
 9,
 1,
 1,
 7,
 5,
 6,
 4,
 9,
 5,
 1,
 3,
 3,
 4,
 7,
 8,
 9,
 1,
 1,
 0,
 9,
 1,
 4,
 4,
 5,
 4,
 0,
 6,
 2,
 2,
 3,
 1,
 5,
 1,
 2,
 0,
 3,
 8,
 1,
 2,
 6,
 7,
 1,
 6,
 2,
 3,
 9,
 0,
 1,
 2,
 2,
 0,
 8,
 9,
 ...]
```

```
plot_pics(xtest,ypred_lab,1234)
```



[8]

```
plot_pics(xtest,ypred_lab,5600)
```



```
mod=Sequential()

mod.add(Conv2D(64,(3,3),activation='relu',input_shape=(28,28,1)))
mod.add(MaxPooling2D((2,2)))

mod.add(Conv2D(128,(3,3),activation='relu'))
mod.add(MaxPooling2D((2,2)))

mod.add(Conv2D(256,(3,3),activation='relu'))

mod.add(Conv2D(256,(3,3),activation='relu'))
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/layers/
convolutional/base_conv.py:107: UserWarning: Do not pass an
`input_shape`/`input_dim` argument to a layer. When using Sequential
models, prefer using an `Input(shape)` object as the first layer in
the model instead.
  super().__init__(activity_regularizer=activity_regularizer,
**kwargs)
```

```
mod.add(Flatten())
mod.add(Dense(256,activation='relu'))
mod.add(Dense(128,activation='relu'))
mod.add(Dense(10,activation='softmax'))
mod.summary()

Model: "sequential_2"

┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳
━━━━━━━━━━━━━━━━━┓
┃ Layer (type)                         ┃ Output Shape                ┃
Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇
━━━━━━━━━━━━━━━━━┩
│ conv2d_7 (Conv2D)                    │ (None, 26, 26, 64)          │
640 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ max_pooling2d_5 (MaxPooling2D)       │ (None, 13, 13, 64)          │
0 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ conv2d_8 (Conv2D)                    │ (None, 11, 11, 128)         │
73,856 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ max_pooling2d_6 (MaxPooling2D)       │ (None, 5, 5, 128)           │
0 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ conv2d_9 (Conv2D)                    │ (None, 3, 3, 256)           │
295,168 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ conv2d_10 (Conv2D)                   │ (None, 1, 1, 256)           │
590,080 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ flatten_2 (Flatten)                  │ (None, 256)                 │
0 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_6 (Dense)                      │ (None, 256)                 │
65,792 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_7 (Dense)                      │ (None, 128)                 │
32,896 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_8 (Dense)                      │ (None, 10)                  │
1,290 │
└──────────────────────────────────────┴─────────────────────────────┴
─────────────────┘
```

```
 Total params: 1,059,722 (4.04 MB)

 Trainable params: 1,059,722 (4.04 MB)

 Non-trainable params: 0 (0.00 B)
```

```python
mod.compile(optimizer='adam',loss='categorical_crossentropy',metrics=[
'accuracy'])

datagen = ImageDataGenerator(
    rotation_range=9,       # Random rotation between 0 and 10 degrees
    width_shift_range=0.05,   # Randomly shift images horizontally
    height_shift_range=0.05,  # Randomly shift images vertically
)

m2=mod.fit(datagen.flow(xtrain,ytrain),epochs=10,validation_data=(xtes
t,ytest),validation_split=0.2)
```

```
Epoch 1/10

/usr/local/lib/python3.10/dist-packages/keras/src/trainers/
data_adapters/py_dataset_adapter.py:121: UserWarning: Your `PyDataset`
class should call `super().__init__(**kwargs)` in its constructor.
`**kwargs` can include `workers`, `use_multiprocessing`,
`max_queue_size`. Do not pass these arguments to `fit()`, as they will
be ignored.
  self._warn_if_super_not_called()

1875/1875 ━━━━━━━━━━━━━━━━━━━━ 32s 15ms/step - accuracy: 0.8783 -
loss: 0.3694 - val_accuracy: 0.9857 - val_loss: 0.0500
Epoch 2/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 33s 18ms/step - accuracy: 0.9792 -
loss: 0.0711 - val_accuracy: 0.9890 - val_loss: 0.0425
Epoch 3/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 36s 15ms/step - accuracy: 0.9854 -
loss: 0.0512 - val_accuracy: 0.9910 - val_loss: 0.0305
Epoch 4/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 42s 15ms/step - accuracy: 0.9882 -
loss: 0.0429 - val_accuracy: 0.9921 - val_loss: 0.0277
Epoch 5/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 40s 15ms/step - accuracy: 0.9883 -
loss: 0.0402 - val_accuracy: 0.9913 - val_loss: 0.0362
Epoch 6/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 28s 15ms/step - accuracy: 0.9901 -
loss: 0.0371 - val_accuracy: 0.9932 - val_loss: 0.0242
Epoch 7/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 28s 15ms/step - accuracy: 0.9915 -
loss: 0.0301 - val_accuracy: 0.9932 - val_loss: 0.0271
Epoch 8/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 40s 14ms/step - accuracy: 0.9926 -
loss: 0.0263 - val_accuracy: 0.9882 - val_loss: 0.0447
```

```
Epoch 9/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 41s 15ms/step - accuracy: 0.9925 -
loss: 0.0275 - val_accuracy: 0.9927 - val_loss: 0.0286
Epoch 10/10
1875/1875 ━━━━━━━━━━━━━━━━━━━━ 29s 15ms/step - accuracy: 0.9931 -
loss: 0.0259 - val_accuracy: 0.9938 - val_loss: 0.0264
```

```
ypred1=mod.predict(xtest)
```

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step
```

```
mod.evaluate(xtest,ytest)
```

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.9932 - loss:
0.0297

[0.026419930160045624, 0.9937999844551086]
```

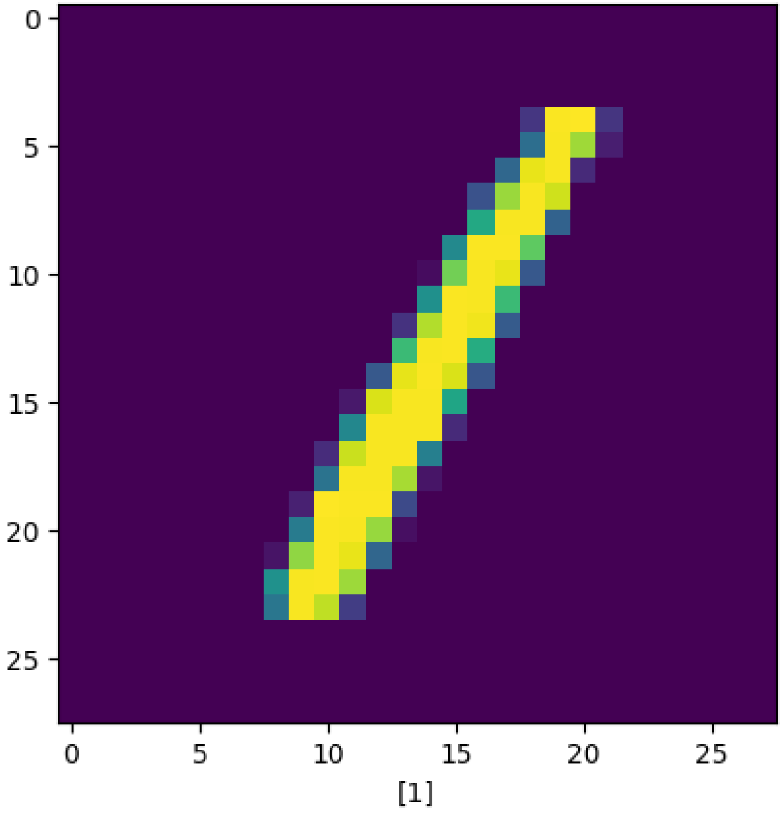
```
ypred1
```

```
array([[1.6442235e-12, 8.5846226e-12, 8.3807121e-11, ...,
1.0000000e+00,
        3.6864001e-13, 8.4666452e-09],
       [4.7247849e-15, 3.4900481e-16, 1.0000000e+00, ..., 1.8150330e-
15,
        9.9717879e-13, 8.7447963e-19],
       [4.0983774e-15, 1.0000000e+00, 8.3844589e-17, ..., 2.1924902e-
15,
        1.2217154e-12, 6.7783890e-20],
       ...,
       [7.7312321e-24, 1.2284736e-15, 3.4590332e-16, ..., 7.6229231e-
12,
        4.1242613e-12, 1.1923924e-09],
       [1.7437672e-17, 5.1925025e-16, 7.8109138e-16, ..., 7.0249901e-
15,
        5.7172855e-10, 4.3413161e-07],
       [1.2580512e-09, 2.0151354e-17, 1.0405259e-15, ..., 2.0101291e-
25,
        2.7480135e-10, 1.5081271e-17]], dtype=float32)
```

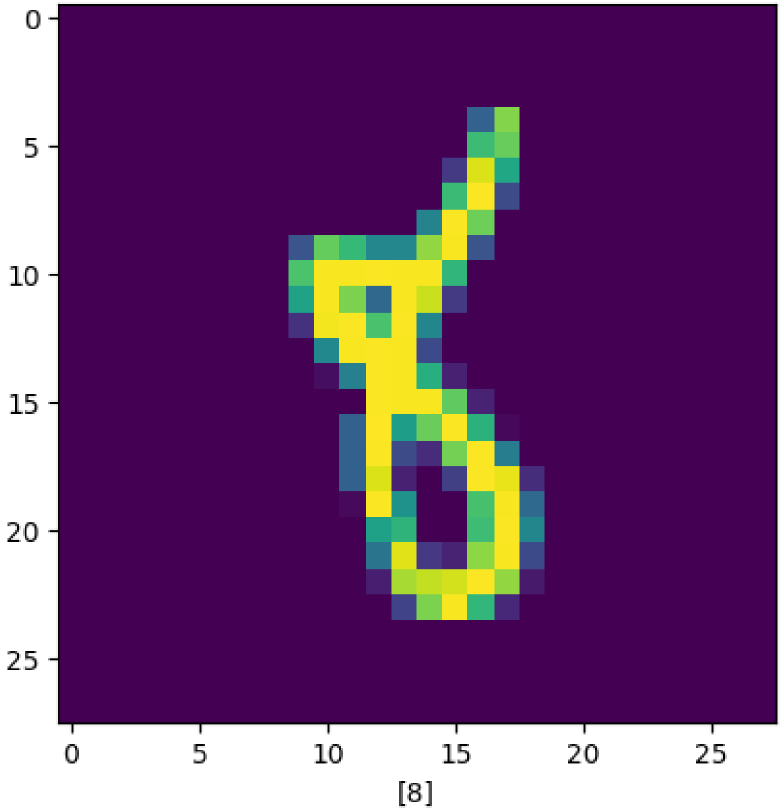
```
ypred_lab1 = [np.argmax(element) for element in ypred1]
ypred_lab1
```

```
[7,
 2,
 1,
 0,
 4,
 1,
 4,
 9,
 5,
 9,
 0,
 6,
 9,
 0,
 1,
 5,
 9,
 7,
 3,
 4,
 9,
 6,
 6,
 5,
 4,
 0,
 7,
 4,
 0,
 1,
 3,
 1,
 3,
 4,
 7,
 2,
 7,
 1,
 2,
 1,
 1,
 7,
 4,
 2,
 3,
 5,
 1,
 2,
 4,
 4,
 6,
 3,
 5,
 5,
 6,
 0,
 4,
1,
9,
5,
7,
8,
9,
3,
7,
4,
6,
4,
3,
0,
7,
0,
2,
9,
1,
7,
3,
2,
9,
7,
7,
6,
2,
7,
8,
4,
7,
3,
6,
1,
3,
6,
9,
3,
1,
4,
1,
7,
6,
9,
6,
0,
5,
4,
9,
9,
 2,
 1,
 9,
 4,
 8,
 7,
 3,
 9,
 7,
 4,
 4,
 4,
 9,
 2,
 5,
 4,
 7,
 6,
 7,
 9,
 0,
 5,
 8,
 5,
 6,
 6,
 5,
 7,
 8,
 1,
 0,
 1,
 6,
 4,
 6,
 7,
 3,
 1,
 7,
 1,
 8,
 2,
 0,
 2,
 9,
 9,
 5,
 5,
 1,
 5,
 6,
 0,
 3,
 4,
 4,
 6,
 5,
 4,
 6,
 5,
 4,
 5,
 1,
 4,
 4,
 7,
 2,
 3,
 2,
 7,
 1,
 8,
 1,
 8,
 1,
 8,
 5,
 0,
 8,
 9,
 2,
 5,
 0,
 1,
 1,
 1,
 0,
 9,
 0,
 3,
 1,
 6,
 4,
 2,
 3,
 6,
 1,
 1,
 1,
 3,
 9,
 5,
 2,
 9,
 4,
 5,
 9,
 3,
 9,
 0,
 3,
 6,
 5,
 5,
 7,
 2,
 2,
 7,
 1,
 2,
 8,
 4,
 1,
 7,
 3,
 3,
 8,
 8,
 7,
 9,
 2,
 2,
 4,
 1,
 5,
 9,
 8,
 7,
 2,
 3,
 0,
 4,
 4,
 2,
 4,
 1,
 9,
5,
7,
7,
2,
8,
2,
6,
8,
5,
7,
7,
9,
1,
8,
1,
8,
0,
3,
0,
1,
9,
9,
4,
1,
8,
2,
1,
2,
9,
7,
5,
9,
2,
6,
4,
1,
5,
8,
2,
9,
2,
0,
4,
0,
0,
2,
8,
4,
7,
 1,
 2,
 4,
 0,
 2,
 7,
 4,
 3,
 3,
 0,
 0,
 3,
 1,
 9,
 6,
 5,
 2,
 5,
 9,
 2,
 9,
 3,
 0,
 4,
 2,
 0,
 7,
 1,
 1,
 2,
 1,
 5,
 3,
 3,
 9,
 7,
 8,
 6,
 5,
 6,
 1,
 3,
 8,
 1,
 0,
 5,
 1,
 3,
 1,
 5,
 5,
 6,
 1,
 8,
 5,
 1,
 7,
 9,
 4,
 6,
 2,
 2,
 5,
 0,
 6,
 5,
 6,
 3,
 7,
 2,
 0,
 8,
 8,
 5,
 4,
 1,
 1,
 4,
 0,
 3,
 3,
 7,
 6,
 1,
 6,
 2,
 1,
 9,
 2,
 8,
 6,
 1,
 9,
 5,
 2,
 5,
 4,
 4,
 2,
 8,
 3,
 8,
 2,
 4,
 5,
 0,
 3,
 1,
 7,
 7,
 5,
 7,
 9,
 7,
 1,
 9,
 2,
 1,
 4,
 2,
 9,
 2,
 0,
 4,
 9,
 1,
 4,
 8,
 1,
 8,
 4,
 5,
 9,
 8,
 8,
 3,
 7,
 6,
 0,
 0,
 3,
 0,
 2,
 6,
 6,
 4,
 9,
 3,
 5,
 3,
 2,
 3,
 9,
 1,
 2,
 6,
 8,
 0,
 5,
 6,
 6,
 6,
 3,
 8,
 8,
 2,
 7,
 5,
 8,
 9,
 6,
 1,
 8,
 4,
 1,
 2,
 5,
 9,
 1,
 9,
 7,
 5,
 4,
 0,
 8,
 9,
 9,
 1,
 0,
 5,
 2,
 3,
 7,
 8,
 9,
 4,
 0,
 6,
 3,
 9,
 5,
 2,
 1,
 3,
 1,
 3,
 6,
 5,
 7,
 4,
 2,
 2,
 6,
 3,
 2,
 6,
 5,
 4,
 8,
 9,
 7,
 1,
 3,
 0,
 3,
 8,
 3,
 1,
 9,
 3,
 4,
 4,
 6,
 4,
 2,
 1,
 8,
 2,
 5,
 4,
 8,
 8,
 4,
 0,
 0,
 2,
 3,
 2,
 7,
 7,
 0,
 8,
 7,
 4,
 4,
 7,
 9,
 6,
 9,
 0,
 9,
 8,
 0,
 4,
 6,
 0,
 6,
 3,
 5,
 4,
 8,
 3,
 3,
 9,
 3,
 3,
 3,
 7,
 8,
 0,
 8,
 2,
 1,
 7,
 0,
 6,
 5,
 4,
 3,
 8,
 0,
 9,
 6,
 3,
 8,
 0,
 9,
 9,
 6,
 8,
 6,
 8,
 5,
 7,
 8,
 6,
 0,
 2,
 4,
 0,
 2,
 2,
 3,
 1,
 9,
 7,
 5,
 1,
 0,
 8,
 4,
 6,
 2,
 6,
 7,
 9,
 3,
 2,
 9,
 8,
 2,
 2,
 9,
 2,
 7,
 3,
 5,
 9,
 1,
 8,
 0,
 2,
 0,
 5,
 2,
 1,
 3,
 7,
 6,
 7,
 1,
 2,
 5,
 8,
 0,
 3,
 7,
 2,
 4,
 0,
 9,
 1,
 8,
 6,
 7,
 7,
 4,
 3,
 4,
 9,
 1,
 9,
 5,
 1,
 7,
 3,
 9,
 7,
 6,
 9,
 1,
 3,
 7,
 8,
 3,
 3,
 6,
 7,
 2,
 8,
 5,
 8,
 5,
 1,
 1,
 4,
 4,
 3,
 1,
 0,
 7,
 7,
 0,
 7,
 9,
 4,
 4,
 8,
 5,
 5,
 4,
 0,
 8,
 2,
 1,
 0,
 8,
 4,
 5,
 0,
 4,
 0,
 6,
 1,
 4,
 3,
 2,
 6,
 7,
 2,
 6,
 9,
 3,
 1,
 4,
 6,
 2,
 5,
 4,
 2,
 0,
 6,
 2,
 1,
 7,
 3,
 4,
 1,
 0,
 5,
 4,
 3,
 1,
 1,
 7,
 4,
 9,
 9,
 4,
 8,
 4,
 0,
 2,
 4,
 5,
 1,
 1,
 6,
 4,
 7,
 1,
 9,
 4,
 2,
 4,
 1,
 5,
 5,
 3,
 8,
 3,
 1,
 4,
 5,
 6,
 8,
 9,
 4,
 1,
 5,
 3,
 8,
 0,
 3,
 2,
 5,
 1,
 2,
 8,
 3,
 4,
 4,
 0,
 8,
 8,
 3,
 3,
 1,
 7,
 3,
 5,
 9,
 6,
 3,
 2,
 6,
 1,
 3,
 6,
 0,
 7,
 2,
 1,
 7,
 1,
 4,
 2,
 8,
 2,
 1,
 7,
 9,
 6,
 1,
 1,
 2,
 4,
 8,
 1,
 7,
 7,
 4,
 8,
 0,
 7,
 3,
 1,
 3,
 1,
 0,
 7,
 7,
 0,
 3,
 5,
 5,
 2,
 7,
 6,
 6,
 9,
 2,
 8,
 3,
 5,
 2,
 2,
 5,
 6,
 0,
 8,
 2,
 9,
 2,
 8,
 8,
 8,
 8,
 7,
 4,
 9,
 3,
 0,
 6,
 6,
 3,
 2,
 1,
 3,
 2,
 2,
 9,
 3,
 0,
 0,
 5,
 7,
 8,
 1,
 4,
 4,
 6,
 0,
 2,
 9,
 1,
 4,
 7,
 4,
 7,
 3,
 9,
 8,
 8,
 4,
 7,
 1,
 2,
 1,
 2,
 2,
 3,
 2,
 3,
 2,
 3,
 9,
 1,
 7,
 4,
 0,
 3,
 5,
 5,
 8,
 6,
 3,
 2,
 6,
 7,
 6,
 6,
 3,
 2,
 7,
 8,
 1,
 1,
 7,
 5,
 6,
 4,
 9,
 5,
 1,
 3,
 3,
 4,
 7,
 8,
 9,
 1,
 1,
 6,
 9,
 1,
 4,
 4,
 5,
 4,
 0,
 6,
 2,
 2,
 3,
 1,
 5,
 1,
 2,
 0,
 3,
 8,
 1,
 2,
 6,
 7,
 1,
 6,
 2,
 3,
 9,
 0,
 1,
 2,
 2,
 0,
 8,
 9,
 ...]
```

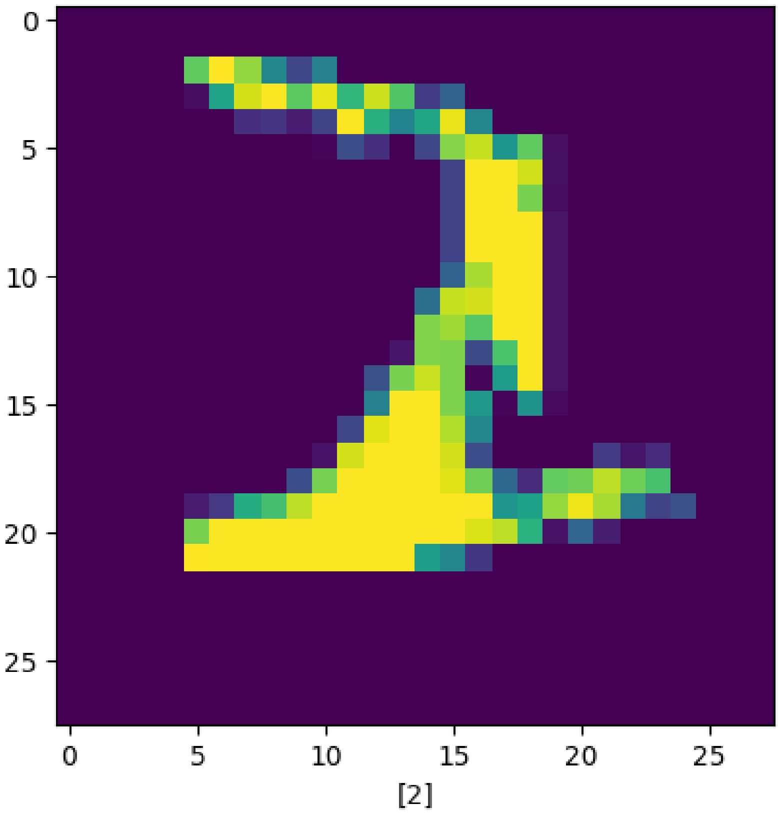
```
plot_pics(xtest,ypred_lab1,6789)
```



```
plot_pics(xtest,ypred_lab1,5678)
```



```
plot_pics(xtest,ypred_lab1,3239)
```



```
mod.save('number_cnn.h5')

WARNING:absl:You are saving your model as an HDF5 file via
`model.save()` or `keras.saving.save_model(model)`. This file format
is considered legacy. We recommend using instead the native Keras
format, e.g. `model.save('my_model.keras')` or
`keras.saving.save_model(model, 'my_model.keras')`.

# Load the saved model
mod_h5 = load_model('number_cnn.h5')

# Check model summary
mod_h5.summary()

WARNING:absl:Compiled the loaded model, but the compiled metrics have
yet to be built. `model.compile_metrics` will be empty until you train
or evaluate the model.

Model: "sequential_2"

┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳
━━━━━━━━━━━━━━━━┓
┃ Layer (type)                         ┃ Output Shape                ┃
Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇
```

```
━━━━━━━━━━━━━━━━━┩
│ conv2d_7 (Conv2D)                    │ (None, 26, 26, 64)          │
640 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ max_pooling2d_5 (MaxPooling2D)       │ (None, 13, 13, 64)          │
0 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ conv2d_8 (Conv2D)                    │ (None, 11, 11, 128)         │
73,856 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ max_pooling2d_6 (MaxPooling2D)       │ (None, 5, 5, 128)           │
0 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ conv2d_9 (Conv2D)                    │ (None, 3, 3, 256)           │
295,168 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ conv2d_10 (Conv2D)                   │ (None, 1, 1, 256)           │
590,080 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ flatten_2 (Flatten)                  │ (None, 256)                 │
0 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_6 (Dense)                      │ (None, 256)                 │
65,792 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_7 (Dense)                      │ (None, 128)                 │
32,896 │
├──────────────────────────────────────┼─────────────────────────────┼
─────────────────┤
│ dense_8 (Dense)                      │ (None, 10)                  │
1,290 │
└──────────────────────────────────────┴─────────────────────────────┴
─────────────────┘

 Total params: 1,059,724 (4.04 MB)

 Trainable params: 1,059,722 (4.04 MB)

 Non-trainable params: 0 (0.00 B)

 Optimizer params: 2 (12.00 B)
```

```python
!pip install gradio # install the missing package
import gradio as gr # import after successful installation
```

```
Requirement already satisfied: gradio in
/usr/local/lib/python3.10/dist-packages (4.44.0)
Requirement already satisfied: aiofiles<24.0,>=22.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (23.2.1)
Requirement already satisfied: anyio<5.0,>=3.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (3.7.1)
Requirement already satisfied: fastapi<1.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.115.0)
Requirement already satisfied: ffmpy in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.4.0)
Requirement already satisfied: gradio-client==1.3.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (1.3.0)
Requirement already satisfied: httpx>=0.24.1 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.27.2)
Requirement already satisfied: huggingface-hub>=0.19.3 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.24.7)
Requirement already satisfied: importlib-resources<7.0,>=1.3 in
/usr/local/lib/python3.10/dist-packages (from gradio) (6.4.5)
Requirement already satisfied: jinja2<4.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (3.1.4)
Requirement already satisfied: markupsafe~=2.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.1.5)
Requirement already satisfied: matplotlib~=3.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (3.7.1)
Requirement already satisfied: numpy<3.0,>=1.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (1.26.4)
Requirement already satisfied: orjson~=3.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (3.10.7)
Requirement already satisfied: packaging in
/usr/local/lib/python3.10/dist-packages (from gradio) (24.1)
Requirement already satisfied: pandas<3.0,>=1.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.1.4)
Requirement already satisfied: pillow<11.0,>=8.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (10.4.0)
Requirement already satisfied: pydantic>=2.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.9.2)
Requirement already satisfied: pydub in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.25.1)
Requirement already satisfied: python-multipart>=0.0.9 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.0.10)
Requirement already satisfied: pyyaml<7.0,>=5.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (6.0.2)
Requirement already satisfied: ruff>=0.2.2 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.6.8)
Requirement already satisfied: semantic-version~=2.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.10.0)
Requirement already satisfied: tomlkit==0.12.0 in
```

```
/usr/local/lib/python3.10/dist-packages (from gradio) (0.12.0)
Requirement already satisfied: typer<1.0,>=0.12 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.12.5)
Requirement already satisfied: typing-extensions~=4.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (4.12.2)
Requirement already satisfied: urllib3~=2.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (2.2.3)
Requirement already satisfied: uvicorn>=0.14.0 in
/usr/local/lib/python3.10/dist-packages (from gradio) (0.30.6)
Requirement already satisfied: fsspec in
/usr/local/lib/python3.10/dist-packages (from gradio-client==1.3.0-
>gradio) (2024.6.1)
Requirement already satisfied: websockets<13.0,>=10.0 in
/usr/local/lib/python3.10/dist-packages (from gradio-client==1.3.0-
>gradio) (12.0)
Requirement already satisfied: idna>=2.8 in
/usr/local/lib/python3.10/dist-packages (from anyio<5.0,>=3.0->gradio)
(3.10)
Requirement already satisfied: sniffio>=1.1 in
/usr/local/lib/python3.10/dist-packages (from anyio<5.0,>=3.0->gradio)
(1.3.1)
Requirement already satisfied: exceptiongroup in
/usr/local/lib/python3.10/dist-packages (from anyio<5.0,>=3.0->gradio)
(1.2.2)
Requirement already satisfied: starlette<0.39.0,>=0.37.2 in
/usr/local/lib/python3.10/dist-packages (from fastapi<1.0->gradio)
(0.38.6)
Requirement already satisfied: certifi in
/usr/local/lib/python3.10/dist-packages (from httpx>=0.24.1->gradio)
(2024.8.30)
Requirement already satisfied: httpcore==1.* in
/usr/local/lib/python3.10/dist-packages (from httpx>=0.24.1->gradio)
(1.0.5)
Requirement already satisfied: h11<0.15,>=0.13 in
/usr/local/lib/python3.10/dist-packages (from httpcore==1.*-
>httpx>=0.24.1->gradio) (0.14.0)
Requirement already satisfied: filelock in
/usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.19.3-
>gradio) (3.16.1)
Requirement already satisfied: requests in
/usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.19.3-
>gradio) (2.32.3)
Requirement already satisfied: tqdm>=4.42.1 in
/usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.19.3-
>gradio) (4.66.5)
Requirement already satisfied: contourpy>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio)
(1.3.0)
Requirement already satisfied: cycler>=0.10 in
```

```
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio)
(0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio)
(4.53.1)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio)
(1.4.7)
Requirement already satisfied: pyparsing>=2.3.1 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio)
(3.1.4)
Requirement already satisfied: python-dateutil>=2.7 in
/usr/local/lib/python3.10/dist-packages (from matplotlib~=3.0->gradio)
(2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.10/dist-packages (from pandas<3.0,>=1.0-
>gradio) (2024.2)
Requirement already satisfied: tzdata>=2022.1 in
/usr/local/lib/python3.10/dist-packages (from pandas<3.0,>=1.0-
>gradio) (2024.1)
Requirement already satisfied: annotated-types>=0.6.0 in
/usr/local/lib/python3.10/dist-packages (from pydantic>=2.0->gradio)
(0.7.0)
Requirement already satisfied: pydantic-core==2.23.4 in
/usr/local/lib/python3.10/dist-packages (from pydantic>=2.0->gradio)
(2.23.4)
Requirement already satisfied: click>=8.0.0 in
/usr/local/lib/python3.10/dist-packages (from typer<1.0,>=0.12-
>gradio) (8.1.7)
Requirement already satisfied: shellingham>=1.3.0 in
/usr/local/lib/python3.10/dist-packages (from typer<1.0,>=0.12-
>gradio) (1.5.4)
Requirement already satisfied: rich>=10.11.0 in
/usr/local/lib/python3.10/dist-packages (from typer<1.0,>=0.12-
>gradio) (13.8.1)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7-
>matplotlib~=3.0->gradio) (1.16.0)
Requirement already satisfied: markdown-it-py>=2.2.0 in
/usr/local/lib/python3.10/dist-packages (from rich>=10.11.0-
>typer<1.0,>=0.12->gradio) (3.0.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/usr/local/lib/python3.10/dist-packages (from rich>=10.11.0-
>typer<1.0,>=0.12->gradio) (2.18.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests->huggingface-
hub>=0.19.3->gradio) (3.3.2)
Requirement already satisfied: mdurl~=0.1 in
```

```
/usr/local/lib/python3.10/dist-packages (from markdown-it-py>=2.2.0-
>rich>=10.11.0->typer<1.0,>=0.12->gradio) (0.1.2)
```

```
def predict_image(image):
    # Resize and preprocess the input image to match the MNIST model's
expected input shape (28x28, grayscale)
    image = np.array(image).astype('float32')
    image = image.reshape(1, 28, 28, 1)  # Reshape to (1, 28, 28, 1)
for batch and channel dim
    image /= 255.0  # Normalize the pixel values

    # Perform prediction
    prediction = mod_h5.predict(image)
    predicted_class = np.argmax(prediction, axis=1)[0]  # Get the
predicted digit (0-9)

    return f"Predicted Digit: {predicted_class}"

# Create the Gradio interface
with gr.Blocks() as demo:
    gr.Markdown("# Handwritten Digit Recognition")

    # Image input: Allows the user to draw a digit or upload an image
    # Removed the invert_colors argument as it is not supported by
gr.Image
    image_input = gr.Image(image_mode="L", label="Draw a digit (0-9)")

    # Output box: Displays the prediction result
    output_text = gr.Textbox(label="Prediction")

    # Button to trigger prediction
    predict_button = gr.Button("Predict")

    # Define interaction: When the button is clicked, the input is
passed to the predict_image function
    predict_button.click(predict_image, inputs=image_input,
outputs=output_text)

# Launch the Gradio app
demo.launch()
```

```
Setting queue=True in a Colab notebook requires sharing enabled.
Setting `share=True` (you can turn this off by setting `share=False`
in `launch()` explicitly).

Colab notebook detected. To show errors in colab notebook, set
debug=True in launch()
Running on public URL: https://f01cbde726bb53b300.gradio.live

This share link expires in 72 hours. For free permanent hosting and
```

```
GPU upgrades, run `gradio deploy` from Terminal to deploy to Spaces
(https://huggingface.co/spaces)

<IPython.core.display.HTML object>
```