Name : Kshitij V Darwhekar

Roll No: TETB19

Sub: Soft Computitng

Batch: B2

```
import tensorflow
```

```
from tensorflow import keras
```

```
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
```

```
(X_train, y_train) , (X_test, y_test) = keras.datasets.mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mni
11493376/11490434 [==============================] - 0s 0us/step
11501568/11490434 [==============================] - 0s 0us/step
```

```
len(X_train)
```

```
60000
```

```
len(X_test)
```

To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu

```
X_train[0].shape
```

```
(28, 28)
```

```
X_train[0]
```

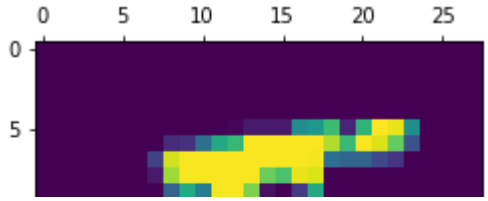
```
array([[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
          0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
          0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
          0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   3,
         18,  18,  18, 126, 136, 175,  26, 166, 255, 247, 127,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,  30,  36,  94, 154, 170,
        253, 253, 253, 253, 253, 225, 172, 253, 242, 195,  64,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,  49, 238, 253, 253, 253, 253,
        253, 253, 253, 253, 251,  93,  82,  82,  56,  39,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,  18, 219, 253, 253, 253, 253,
        253, 198, 182, 247, 241,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,  80, 156, 107, 253, 253,
        205,  11,   0,  43, 154,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,   0,  14,   1, 154, 253,
         90,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0, 139, 253,
        190,   2,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
       [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,  11, 190,
        253,  70,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
```

```
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,  35,
 241, 225, 160, 108,   1,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
  81, 240, 253, 253, 119,  25,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,  45, 186, 253, 253, 150,  27,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0,  16,  93, 252, 253, 187,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0,   0,   0, 249, 253, 249,  64,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,  46, 130, 183, 253, 253, 207,   2,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,  39,
 148, 229, 253, 253, 253, 250, 182,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,  24, 114, 221,
 253, 253, 253, 253, 201,  78,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,  23,  66, 213, 253, 253,
 253, 253, 198,  81,   2,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,  18, 171, 219, 253, 253, 253, 253,
 195,  80,   9,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,  55, 172, 226, 253, 253, 253, 253, 244, 133,
  11,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0, 136, 253, 253, 253, 212, 135, 132,  16,   0,
   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
                                                   0,   0,   0,
[                                                  0,   0,   0,
   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0],
[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,   0,
   0,   0]], dtype=uint8)
```

To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu

```
plt.matshow(X_train[0])
```

```
<matplotlib.image.AxesImage at 0x7efc976e77d0>
```




```
y_train[0]
```

```
5
```

```
# Scaling Technique

X_train = X_train / 255
X_test = X_test / 255
```

```
X_train[0]
```

```
array([[0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.18039216,
        0.50980392, 0.71764706, 0.99215686, 0.99215686, 0.81176471,
        0.00784314, 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.15294118, 0.58039216, 0.89803922,
        0.99215686, 0.99215686, 0.99215686, 0.98039216, 0.71372549,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.09411765, 0.44705882, 0.86666667, 0.99215686, 0.99215686,
        0.99215686, 0.99215686, 0.78823529, 0.30588235, 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.07058824, 0.67058824, 0.85882353, 0.99215686,
        0.99215686, 0.99215686, 0.99215686, 0.76470588, 0.31372549,
        0.03529412, 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.21568627,
        0.6745098 , 0.88627451, 0.99215686, 0.99215686, 0.99215686,
        0.99215686, 0.95686275, 0.52156863, 0.04313725, 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        ],
       [0.        , 0.        , 0.        , 0.        , 0.53333333,
        0.99215686, 0.99215686, 0.99215686, 0.83137255, 0.52941176,
        0.51764706, 0.0627451 , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
        0.        , 0.        , 0.        , 0.        , 0.        ,
```

To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu

```
         0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        ],
        [0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.        ]])
```

```
X_train_flattened = X_train.reshape(len(X_train), 28*28)
X_test_flattened = X_test.reshape(len(X_test), 28*28)
```

```
X_train_flattened.shape
```

```
(60000, 784)
```

```
X_test_flattened.shape
```

```
(10000, 784)
```

```
X_train_flattened[0]
```

```
       0.97647059, 0.25098039, 0.        , 0.        , 0.        ,
```

To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu

```
       0.        , 0.        , 0.        , 0.18039216, 0.50980392,
       0.71764706, 0.99215686, 0.99215686, 0.81176471, 0.00784314,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.15294118,
       0.58039216, 0.89803922, 0.99215686, 0.99215686, 0.99215686,
       0.98039216, 0.71372549, 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.09411765, 0.44705882, 0.86666667, 0.99215686, 0.99215686,
       0.99215686, 0.99215686, 0.78823529, 0.30588235, 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.        , 0.        , 0.        , 0.        ,
       0.        , 0.09019608, 0.25882353, 0.83529412, 0.99215686,
       0.99215686, 0.99215686, 0.99215686, 0.77647059, 0.31764706,
       0.00784314, 0.        , 0.        , 0.        , 0.        ,
```

```
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.07058824, 0.67058824, 0.85882353,
0.99215686, 0.99215686, 0.99215686, 0.99215686, 0.76470588,
0.31372549, 0.03529412, 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.21568627, 0.6745098 ,
0.88627451, 0.99215686, 0.99215686, 0.99215686, 0.99215686,
0.95686275, 0.52156863, 0.04313725, 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.53333333, 0.99215686, 0.99215686, 0.99215686,
0.83137255, 0.52941176, 0.51764706, 0.0627451 , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        , 0.        ,
0.        , 0.        , 0.        , 0.        ])
```

```
# 10 Output neuron and 784 in input neuron
model = keras.Sequential([
```

To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu

```
oid')
```

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

model.fit(X_train_flattened, y_train, epochs=5)
```

```
Epoch 1/5
1875/1875 [==============================] - 3s 1ms/step - loss: 0.4724 - accuracy: 0
Epoch 2/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.3034 - accuracy: 0
Epoch 3/5
1875/1875 [==============================] - 4s 2ms/step - loss: 0.2833 - accuracy: 0
Epoch 4/5
1875/1875 [==============================] - 5s 3ms/step - loss: 0.2731 - accuracy: 0
Epoch 5/5
1875/1875 [==============================] - 5s 3ms/step - loss: 0.2664 - accuracy: 0
<keras.callbacks.History at 0x7efc93077f50>
```

```
model.evaluate(X_test_flattened, y_test)
```

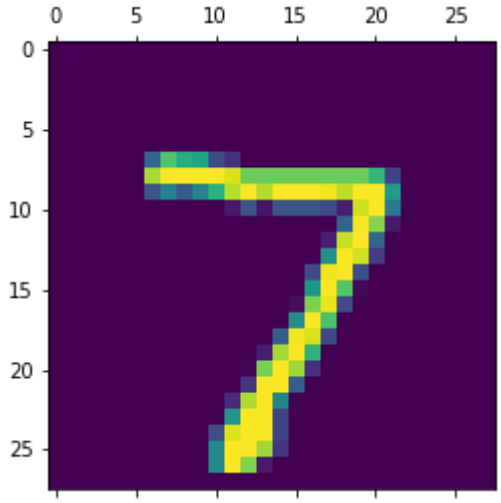
```
313/313 [==============================] - 1s 2ms/step - loss: 0.2695 - accuracy: 0.9
[0.2695145010948181, 0.9254000186920166]
```

```
y_predicted = model.predict(X_test_flattened)
y_predicted[0]
```

```
array([1.5027165e-02, 3.9325224e-07, 6.3410342e-02, 9.5975685e-01,
       3.2548308e-03, 1.0176152e-01, 1.0720740e-06, 9.9978119e-01,
       7.0441395e-02, 6.0749489e-01], dtype=float32)
```

```
plt.matshow(X_test[0])
```

```
<matplotlib.image.AxesImage at 0x7efc920031d0>
```




```
# np.argmax finds a maximum element from an array and returns the index of it
```



```
7
```

```
y_predicted_labels = [np.argmax(i) for i in y_predicted]
```

```
y_predicted_labels[:5]
```

```
[7, 2, 1, 0, 4]
```

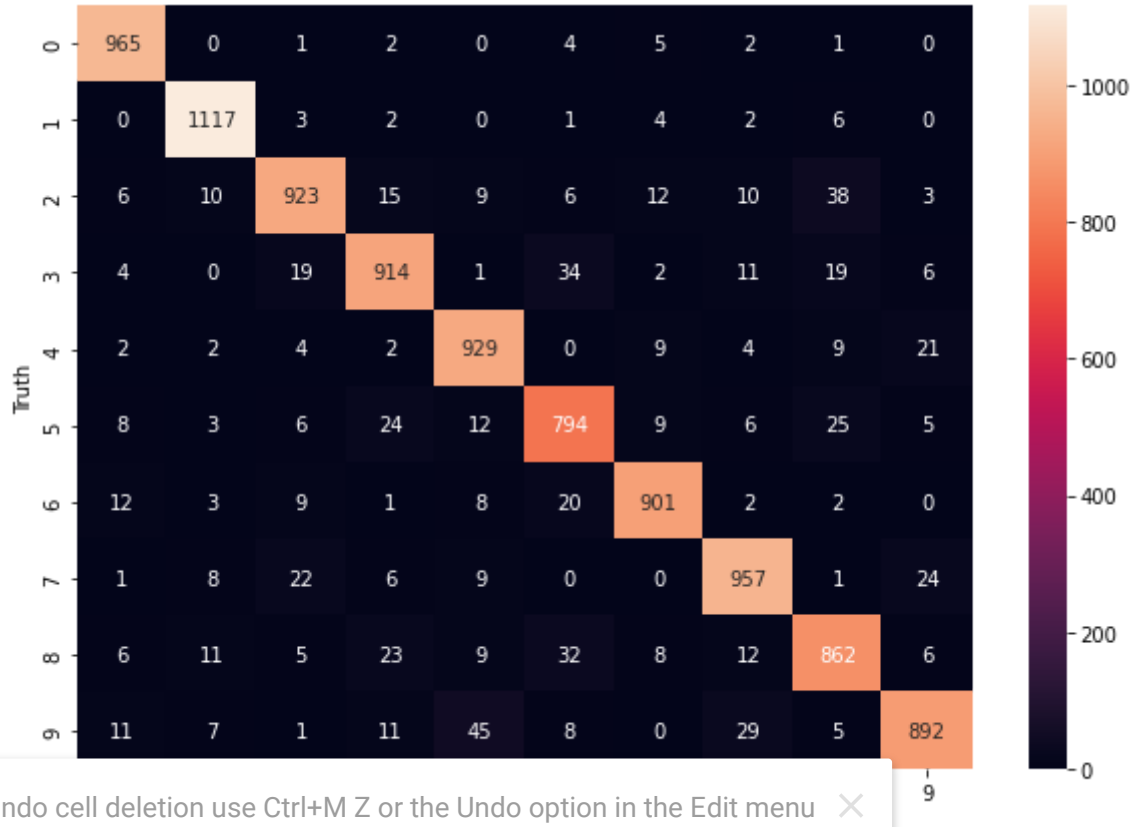
```
cm = tf.math.confusion_matrix(labels=y_test,predictions=y_predicted_labels)
cm
```

```
<tf.Tensor: shape=(10, 10), dtype=int32, numpy=
array([[ 965,    0,    1,    2,    0,    4,    5,    2,    1,    0],
       [   0, 1117,    3,    2,    0,    1,    4,    2,    6,    0],
       [   6,   10,  923,   15,    9,    6,   12,   10,   38,    3],
       [   4,    0,   19,  914,    1,   34,    2,   11,   19,    6],
```

```
       [   2,    2,    4,    2,  929,    0,    9,    4,    9,   21],
       [   8,    3,    6,   24,   12,  794,    9,    6,   25,    5],
       [  12,    3,    9,    1,    8,   20,  901,    2,    2,    0],
       [   1,    8,   22,    6,    9,    0,    0,  957,    1,   24],
       [   6,   11,    5,   23,    9,   32,    8,   12,  862,    6],
       [  11,    7,    1,   11,   45,    8,    0,   29,    5,  892]],
      dtype=int32)>
```

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

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



To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu

Using hidden layer

```
model = keras.Sequential([
    keras.layers.Dense(100, input_shape=(784,), activation='relu'),
    keras.layers.Dense(10, activation='sigmoid')
])

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

model.fit(X_train_flattened, y_train, epochs=5)
```

```
Epoch 1/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2726 - accuracy: 0
```

```
Epoch 2/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.1230 - accuracy: 0
Epoch 3/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0852 - accuracy: 0
Epoch 4/5
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0660 - accuracy: 0
Epoch 5/5
1875/1875 [==============================] - 3s 1ms/step - loss: 0.0521 - accuracy: 0
<keras.callbacks.History at 0x7efc8ed501d0>
```

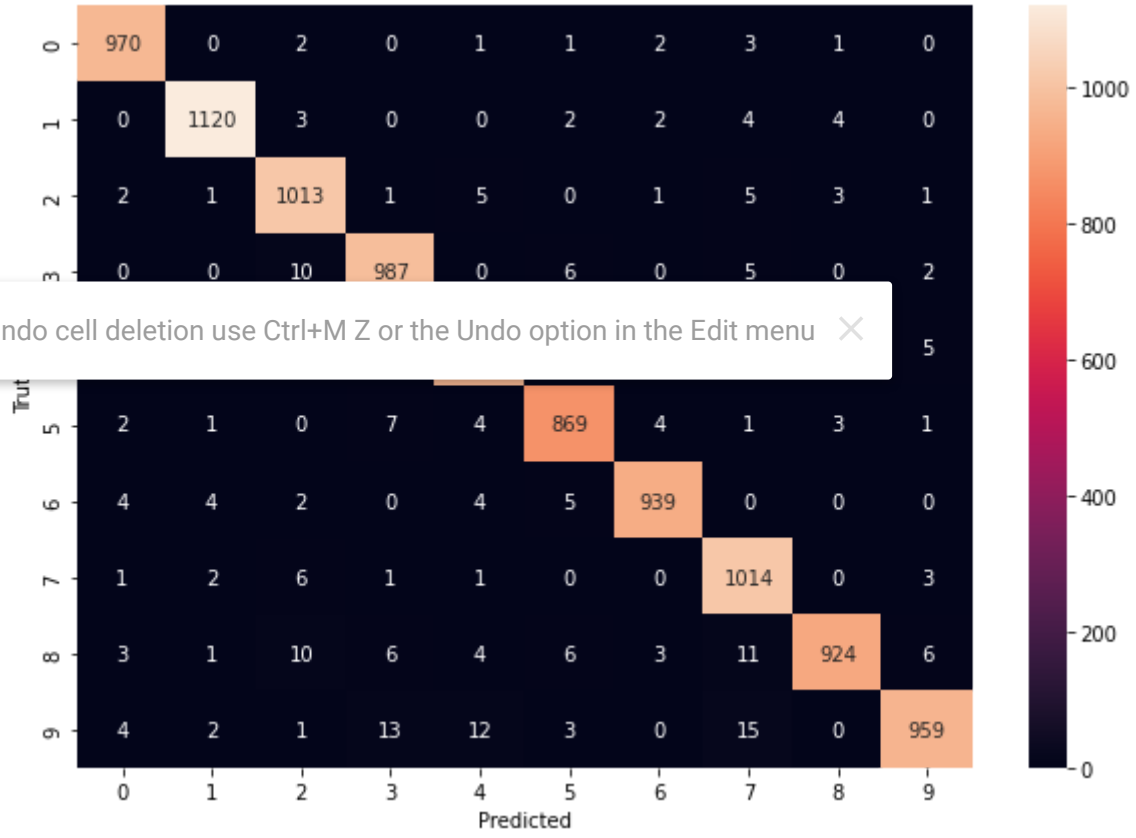
```
model.evaluate(X_test_flattened,y_test)
```

```
313/313 [==============================] - 0s 1ms/step - loss: 0.0786 - accuracy: 0.9
[0.07863084971904755, 0.9763000011444092]
```

```
y_predicted = model.predict(X_test_flattened)
y_predicted_labels = [np.argmax(i) for i in y_predicted]
cm = tf.math.confusion_matrix(labels=y_test,predictions=y_predicted_labels)

plt.figure(figsize = (10,7))
sn.heatmap(cm, annot=True, fmt='d')
plt.xlabel('Predicted')
plt.ylabel('Truth')
```

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



Using Flatten layer so that we don't have to call .reshape on input dataset

```
model = keras.Sequential([
    keras.layers.Flatten(input_shape=(28, 28)),
    keras.layers.Dense(100, activation='relu'),
    keras.layers.Dense(10, activation='sigmoid')
])

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

model.fit(X_train, y_train, epochs=10)
```

```
Epoch 1/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.2693 - accuracy: 0
Epoch 2/10
1875/1875 [==============================] - 3s 1ms/step - loss: 0.1207 - accuracy: 0
Epoch 3/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0839 - accuracy: 0
Epoch 4/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0657 - accuracy: 0
Epoch 5/10
1875/1875 [==============================] - 6s 3ms/step - loss: 0.0490 - accuracy: 0
Epoch 6/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0408 - accuracy: 0
Epoch 7/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0332 - accuracy: 0
Epoch 8/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0281 - accuracy: 0
Epoch 9/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0225 - accuracy: 0
Epoch 10/10
1875/1875 [==============================] - 3s 2ms/step - loss: 0.0196 - accuracy: 0
<keras.callbacks.History at 0x7efc92f47c90>
```



```
313/313 [==============================] - 0s 1ms/step - loss: 0.0913 - accuracy: 0.9
[0.0912703275680542, 0.9740999937057495]
```

0s completed at 4:17 PM

To undo cell deletion use Ctrl+M Z or the Undo option in the Edit menu