

## MNIST Digit Classification code:

```
import tensorflow as tf

from tensorflow.keras import layers, models

from tensorflow.keras.datasets import mnist

import matplotlib.pyplot as plt

# Load the MNIST dataset

(x_train, y_train), (x_test, y_test) = mnist.load_data()

# Normalize the pixel values to be between 0 and 1

x_train, x_test = x_train / 255.0, x_test / 255.0

# Define the model

model = models.Sequential([

    layers.Flatten(input_shape=(28, 28)),

    layers.Dense(128, activation='relu'),

    layers.Dropout(0.2),

    layers.Dense(10, activation='softmax')

])

# Compile the model

model.compile(optimizer='adam',

              loss='sparse_categorical_crossentropy',

              metrics=['accuracy'])

# Train the model

history = model.fit(x_train, y_train, epochs=5, validation_data=(x_test, y_test))

# Evaluate the model

test_loss, test_acc = model.evaluate(x_test, y_test)

print('Test accuracy:', test_acc)

# Plot training history

plt.plot(history.history['accuracy'], label='accuracy')

plt.plot(history.history['val_accuracy'], label='val_accuracy')
```

```
plt.xlabel('Epoch')  
plt.ylabel('Accuracy')  
plt.ylim([0.5, 1])  
plt.legend(loc='lower right')  
plt.show()
```