IN [1] import cv2

import numpy as np

from keras.datasets import mnist

from keras.layers import Dense, Flatten, MaxPooling2D, Dropout

from keras.layers.convolutional import Conv2D

from keras.models import Sequential

from tensorflow.keras.utils import to\_categorical

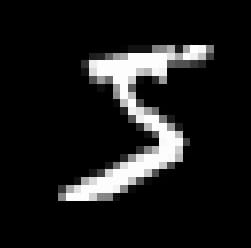
import matplotlib.pyplot as

IN [2] (X\_train, y\_train), (X\_test,

IN [3] plt.imshow(X\_train[0], cmap="gray")

plt.show()

print (y\_train[0])



IN[5] print ("Shape of X\_train: {}".format(X\_train.shape))

print ("Shape of y\_train: {}".format(y\_train.shape))

print ("Shape of X\_test: {}".format(X\_test.shape))

print ("Shape of y\_test: {}".format(y\_test.shape))

Shape of X\_train: (60000, 28, 28)

Shape of y\_train: (60000,)

Shape of X\_test: (10000, 28, 28)

Shape of y\_test: (10000,)

IN[6] # Reshaping so as to convert images for our model

X\_train = X\_train.reshape(60000, 28, 28, 1)

X\_test = X\_test.reshape(10000, 28, 28, 1)

IN[7] print ("Shape of X\_train: {}".format(X\_train.shape))

print ("Shape of y\_train: {}".format(y\_train.shape))

print ("Shape of X\_test: {}".format(X\_test.shape))

print ("Shape of y\_test: {}".format(y\_test.shape))

Shape of X\_train: (60000, 28, 28, 1)

Shape of y\_train: (60000,)

Shape of X\_test: (10000, 28, 28, 1)

Shape of y\_test: (10000,)

*IN[8] #one hot encoding*

y\_train **=** to\_categorical(y\_train)

y\_test **=** to\_categorical(y\_test)

IN[9] model **=** Sequential()

*## Declare the layers*

layer\_1 **=** Conv2D(64, kernel\_size**=**3, activation**=**'relu', input\_shape**=**(28, 28, 1))

layer\_2 **=** MaxPooling2D(pool\_size**=**2)

layer\_3 **=** Conv2D(32, kernel\_size**=**3, activation**=**'relu')

layer\_4 **=** MaxPooling2D(pool\_size**=**2)

layer\_5 **=** Dropout(0.5)

layer\_6 **=** Flatten()

layer\_7 **=** Dense(128, activation**=**"relu")

layer\_8 **=** Dropout(0.5)

layer\_9 **=** Dense(10, activation**=**'softmax')

*## Add the layers to the model*

model**.**add(layer\_1)

model**.**add(layer\_2)

model**.**add(layer\_3)

model**.**add(layer\_4)

model**.**add(layer\_5)

model**.**add(layer\_6)

model**.**add(layer\_7)

model**.**add(layer\_8)

model**.**add(layer\_9)