

```

1  # from https://www.tensorflow.org/tutorials
2
3  import tensorflow as tf
4  import numpy as np
5
6  # specify path to training data and testing data
7
8  train_x_location = "mnist_x_train.csv"
9  train_y_location = "mnist_y_train.csv"
10 test_x_location = "mnist_x_test.csv"
11 test_y_location = "mnist_y_test.csv"
12
13 print("Reading training data")
14 x_train_2d = np.loadtxt(train_x_location, dtype="uint8", delimiter=",")
15 x_train_3d = x_train_2d.reshape(-1,28,28,1)
16 x_train = x_train_3d
17 y_train = np.loadtxt(train_y_location, dtype="uint8", delimiter=",")
18
19 print("Preprocessing x of training data")
20 x_train = x_train / 255.0
21
22 # define the training model
23 model = tf.keras.models.Sequential([
24     tf.keras.layers.Flatten(input_shape=(28, 28,1)),
25     # regularization can be added to most layers
26     tf.keras.layers.Dense(512, activation=tf.nn.relu,
27                             kernel_regularizer=tf.keras.regularizers.l2(0.001)),
28     tf.keras.layers.Dense(10, activation=tf.nn.softmax)
29 ])
30 model.compile(optimizer='adam',
31               loss='sparse_categorical_crossentropy',
32               metrics=['accuracy'])
33
34 print("train")
35 model.fit(x_train, y_train, epochs=5, batch_size=32)
36 # default batch size is 32
37
38 print("Reading testing data")
39 x_test_2d = np.loadtxt(test_x_location, dtype="uint8", delimiter=",")
40 x_test_3d = x_test_2d.reshape(-1,28,28,1)
41 x_test = x_test_3d
42 y_test = np.loadtxt(test_y_location, dtype="uint8", delimiter=",")
43
44 print("Preprocessing testing data")
45 x_test = x_test / 255.0
46
47 print("evaluate")
48 model.evaluate(x_test, y_test)

```