

```
1 from keras.utils import np_utils
2 from keras.datasets import mnist
3 from keras.models import Sequential
4 from keras.layers import Dense
5 # MNIST data
6 (train_images, train_labels), (test_images, test_labels) = mnist.load_data()
7 print(train_images.shape, train_labels.shape, test_images.shape, test_labels.shape)
8 train_images = train_images.reshape(train_images.shape[0], 784).astype('float32')/255.0
9 test_images = test_images.reshape(test_images.shape[0], 784).astype('float32')/255.0
10 train_labels = np_utils.to_categorical(train_labels) # One-Hot Encoding
11 test_labels = np_utils.to_categorical(test_labels) # One-Hot Encoding
12 # Model
13 model = Sequential()
14 model.add(Dense(1, activation='relu')) # units=256, activation='relu'
15 model.add(Dense(256, activation='relu')) # units=256, activation='relu'
16 model.add(Dense(512, activation='relu')) # units=256, activation='relu'
17 model.add(Dense(10, activation='softmax')) # units=10, activation='softmax'
18 model.compile(loss='categorical_crossentropy', optimizer='sgd', metrics=['accuracy'])
19 # Training
20 model.fit(train_images, train_labels, epochs=10, batch_size=128, verbose=1)
21 # Testing
22 _, accuracy = model.evaluate(test_images, test_labels)
23 print('Accuracy: ', accuracy)
24 model.summary()
25
26
```