

Handwriting

目的:使用 Mnist 資料庫製作手寫辨識的模組。

步驟:

1. 建立數據:

首先參照其他人使用 mnist 的方法，製作 `X_train`、`y_train`、`X_test`、`y_test`。作為訓練的資料，`X` 是數據，而 `y` 是標籤。試著用 `numpy.shape` 的功能，印出數據的資料型態以便之後載入模組。

```
Using TensorFlow backend.  
(60000, 1, 28, 28) (60000, 10)
```

2. 建立模組:

確認完資料無誤後，建立訓練模組，先做出卷基層，使用 `Conv2D` 的卷基函數，在進行 `MAX_Pooling` 最大池化，再來用 `Flatten` 使數據變成一維的，最後用 `dense(10,activation='softmax')` 將資料濃縮成十個值，`Softmax` 是一個梯度歸一化的數據處理方法，很適合在分類時使用，他可以使數據 $0 < x < 1$ ，且總和為一，而最大的數值即為答案。

3. 訓練:

進行 `fit()` 同時觀察 `accuracy` 和 `loss` 的值。

4. 改進:

我先把 `epoch` 調成五，也觀察進行到第幾次 `epoch` 時無法向上。這時就代表這個模組會許有需要加強的空間例如，我第一次只用一次的精準度從 0.91 進步的幅度就變很小。

```

# data pre proc
model = Sequential()

# Conv layer 1 output shape (32, 28, 28)
model.add(Convolution2D(
    batch_input_shape=(None, 1, 28, 28),
    filters=32,
    kernel_size=5,
    strides=1,
    padding='same',      # Padding method
    data_format='channels_first',
))
model.add(MaxPooling2D(2, 2, 'same', data_format='channels_first'))
model.add(Flatten())
model.add(Dense(10, activation='softmax'))
adam = Adam(lr=1e-4)

# show model

model.summary()

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

```

```

Instructions for updating:
keep_dims is deprecated, use keepdims instead
Training -----
Epoch 1/5
60000/60000 [=====] - 6s 104us/step - loss: 0.6818 - a
cc: 0.8418
Epoch 2/5
60000/60000 [=====] - 5s 79us/step - loss: 0.2983 - ac
c: 0.9145
Epoch 3/5
60000/60000 [=====] - 5s 78us/step - loss: 0.2549 - ac
c: 0.9274
Epoch 4/5
60000/60000 [=====] - 5s 79us/step - loss: 0.2245 - ac
c: 0.9365
Epoch 5/5
60000/60000 [=====] - 5s 78us/step - loss: 0.1994 - ac
c: 0.9444
teamp8@ocean:/data/teamp8/keras/keras/Mnist_HandWriting$ █

```

subscribing to the professional edition here: <http://mobaxterm.mobatek.net>

而我在池化後再加了一個卷積層，結果準確度就提升到 0.95 才慢下來。

最後我又再次將池化，準確度又從 0.95 提升到 0.97，可見卷積和池化的過程中不斷的提取特徵值是有利於手寫分類的。

心得:

這個作業其實網路上有很多範本，可是透過自己實作，一層一層的去增加卷積層和池化層，以及 dense() 了解模組在訓練的過程中是如何進分類的。雖然還是有很多的參數具體公式不是很了解，但是幸好網路上有很多資料，一個一個查還是可以足見摸索到該用什麼類型的方法進行數據處理。