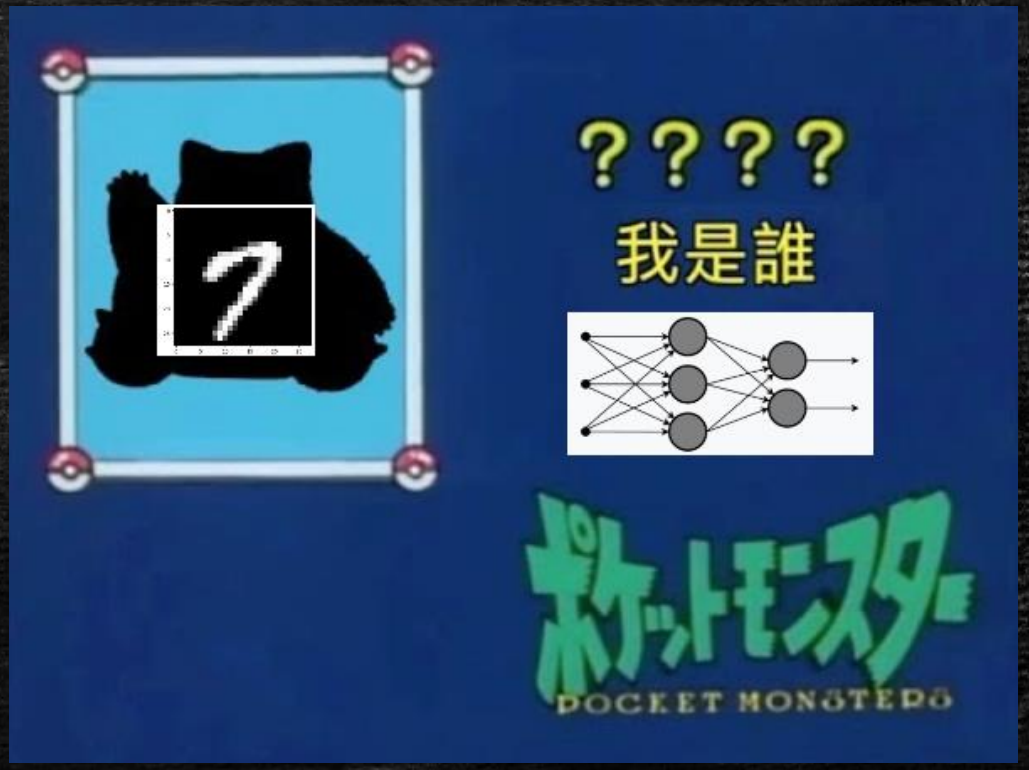
2024 鐵人賽 – 我數學就爛要怎麼來學 DNN 模型安全
Day 07 – Hello World DNN 模型

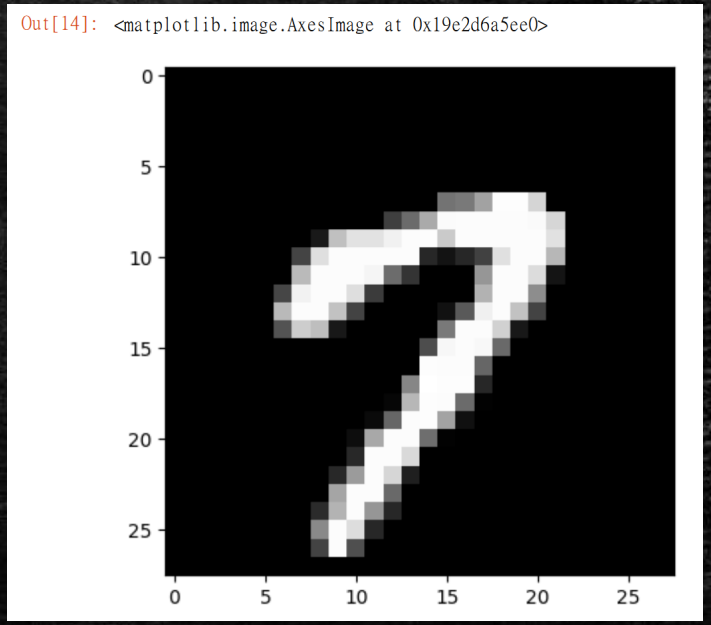
大綱

- DNN 模型界的 Hello World
  - MNIST 手寫辨識模型訓練及使用
  - One Hot Encoding
- 結論



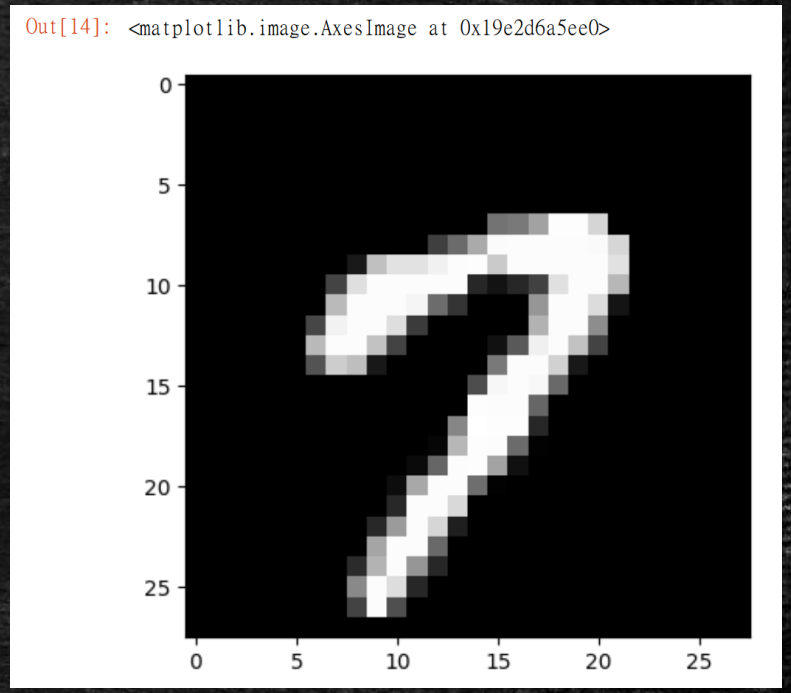
背景介紹

- MNIST 資料庫是一個大型手寫數位資料庫，其內容由 28 × 28 的灰階手寫圖示組成，一共有 60000筆訓練資料跟 10000的測試資料
- 每個像素值由0(黑)到255(白)的亮度值表示



先來討論個嚴肅的問題

- 這個辨識模型的輸出應該長甚麼樣子?
- 真的要直接輸出 0~9 嗎?
- 辨識出 7 跟 3 彼此之間有大小關係嗎?



One Hot Encoding

獨熱 [編輯]

條目 討論 汉 漢 臺灣正體

獨熱[1]（英語：One-hot）在數位電路和機器學習中被用來表示一種特殊的位元組或向量，該位元組或向量裏僅容許其中一位爲1，其他位都必須爲0[2]。其被稱爲獨熱因爲其中只能有一個1，若情況相反，只有一個0，其餘爲1，則稱爲**獨冷**（One-cold）[3]。在統計學中，虛擬變數代表了類似的概念。

In [17]:
```
#num_classes : Total number of classes. If None, this would be inferred as max(x) + 1. Defaults to None.
num_classes=4
training_labels = [0,1,2,3]
training_labels = tf.keras.utils.to_categorical(training_labels, num_classes)
print(training_labels)
```

```
[[1. 0. 0. 0.]
 [0. 1. 0. 0.]
 [0. 0. 1. 0.]
 [0. 0. 0. 1.]]
```

來寫個程式吧

In [ ]:
```
# 把後面二維的部分攤平成一維
training_images = training_images.reshape(60000, 784)
test_images = test_images.reshape(10000, 784)

#轉換格式為 float32
training_images = training_images.astype('float32')
test_images = test_images.astype('float32')

# 將數值做正規化
training_images  = training_images / 255.0
test_images = test_images / 255.0

# 如果使用 sparse_categorical_crossentropy 就不需要做 One Hot Encoding
# https://axk51013.medium.com/%E4%B8%8D%E8%A6%81%E5%86%8D%E5%81%9Aone-hot-encoding-b5126d3f8a63
num_classes=10
training_labels = tf.keras.utils.to_categorical(training_labels, num_classes)
test_labels = tf.keras.utils.to_categorical(test_labels, num_classes)
```

In [ ]:
```
# 建立屬於自己的 model
model = Sequential()
model.add(Dense(128, input_dim=784, activation=tf.nn.relu))
model.add(Dense(10, activation=tf.nn.softmax))
```

結論

- 手寫圖片的辨識模型雖然簡單且像素也不大，但練習過程中可以學到 DNN 大部分需要用到的技巧

- DNN 模型的先備知識差不多就到這邊了，明天就會開始介紹 DNN 模型安全，之後的實作也會透過這個 DNN 模型作為要攻擊的對象