

人工智慧單晶片電腦實務應用

1. 政大蔡炎龍教授 MOOCS 課程(成為 python 數據分析達人的第一門課)，

第 1 次作業(進度: 1-1 ~ 1-10):

BMI 程序

$$\text{BMI} = \text{體重(公斤)}/\text{身高(公尺)}^2$$

第 2 次作業(進度: 2-1 ~ 2-6):

BMI 程序寫成 “函式”形式

第 3 次作業(進度: 2-7 ~ 2-13):

猜一猜，使用亂數產生 0 ~ 99 之間的整數，你先猜此亂數是大於(含等於)50 或是小於 50，你輸入 0 或 1 分別表達小於 50 或大於(含等於)50。若是輸入與亂數結果一致，則你猜對了，反之則猜錯了。

第 4 次作業(進度: 3-1 ~ 3-6):

利用 `numpy.array` 之特性(不使用 `for` 迴圈)，計算三位同學 BMI，請參考以下範例: 計算三個三角形面積

```
In [2]: import numpy as np
```

```
In [14]: B = np.array([4,5,6])  
H = np.array([1,2,3])  
A = B*H/2  
print("Area = ", A)
```

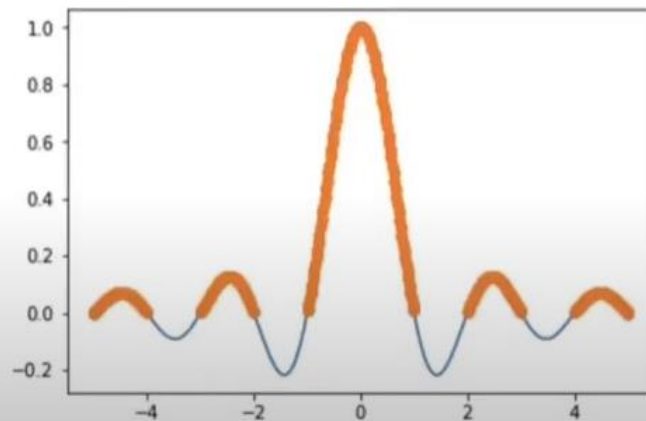
```
Area = [2. 5. 9.]
```

第 5 次作業(進度: 3-7 ~ 3-13):

請參考 3-12 講義與下圖，使用最少行數的程式，請畫出 $\sin(x)$ 波形(振幅在 $-2 \sim 2$ 之間)，並標記該圖形振幅在 $1 \sim 2$ 之間的圖形，如下圖。

```
In [8]: plt.plot(x,y)  
plt.plot(x[y>0], y[y>0], 'o')
```

```
Out[8]: [<matplotlib.lines.Line2D at 0x1090dedd8>]
```



就會變成 哇

```
In [ ]: |
```

第 6 次作業(進度: 4-1 ~ 4-8):

請參考 4-7, 4-8 所提及互動指令 “interact_manual”，如下所示:

```
In [8]: def draw(n):
plt.scatter(X, Y, s=100, c='r')
y = 0.5*np.sin(n*x*2*np.pi) + 0.5
plt.plot(x, y)
plt.show()

In [9]: interact_manual(draw, n=(1., 200.))
```



我們可以在這邊拉數值滑桿

```
In [3]: def BMICAL(身高, 體重):
身高 = float(身高)/100
體重 = float(體重)
bmi = 體重 / (身高**2)
print("你的 BMI = {:.2f}".format(bmi))

In [4]: interact_manual(BMICAL, 身高="請輸入你的身高", 體重="請輸入你的體重")
```



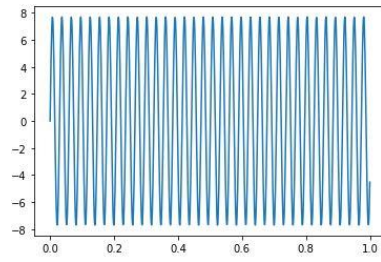
就請輸入你的身高 比方說呢

問題: 請設計一款互動式介面，可以調整正旋波的振幅與頻率，圖形如下:

```
In [20]: interact_manual(draw, n=(1., 200.), a=(1., 10.))
```

n
a

Run Interact



第 7 次作業(進度: 5-1 ~ 5-8):

問題: 請重做講義 5-7 的內容請參考如下說明:

+ 程式碼 + 文字



```
1 %pip install pandas
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/l
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/p
Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/
Requirement already satisfied: six>=1.5 in /usr/local/lib/pytho
```

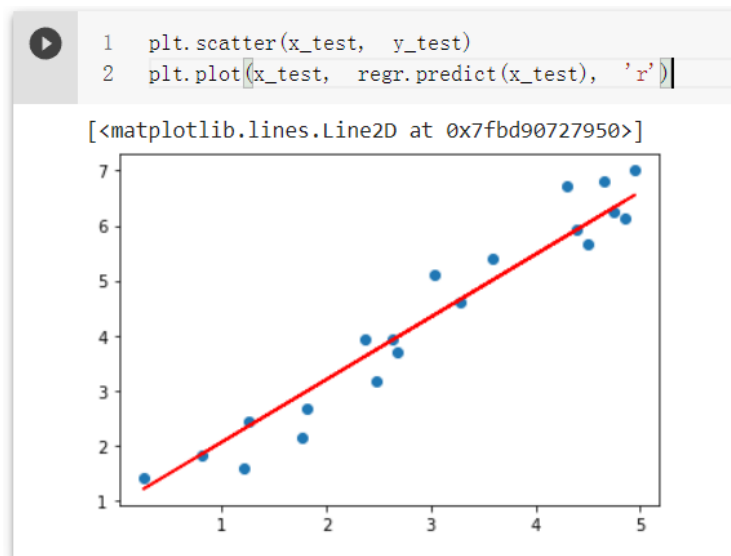
```
[ ] 1 import pandas_datareader.data as web
```

```
[ ] 1 import yfinance as yf
2 df = yf.download('AAPL', start='2012-09-01', end='2023-08-31', progress=False,)
3 df.head()
```

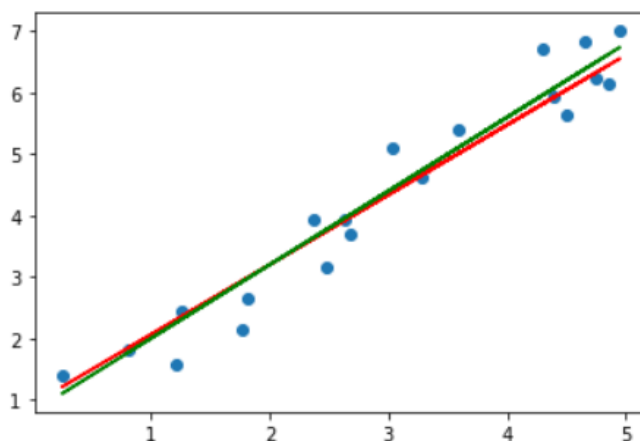
	Open	High	Low	Close	Adj Close	Volume
Date						
2012-09-04	23.777143	24.112143	23.732143	24.106071	20.548996	367892000
2012-09-05	24.127501	24.155357	23.914286	23.936787	20.404697	336375200
2012-09-06	24.041786	24.224644	23.957144	24.152500	20.588581	391196400
2012-09-07	24.216070	24.374287	24.134644	24.301430	20.715525	329666400
2012-09-10	24.301786	24.403214	23.646429	23.669287	20.176668	487998000

第 8 次作業(進度: 政大蔡教授講義 6-1 ~ 6-8):

問題: 請參考講義 6-5 「用線性回歸作預測」, 請將例題操作一遍, 其測試與線性回歸預測如下圖



最後需將原來線性關係 $y = 1.2 * x + 0.8$ 加上, 如下圖綠線所示:



第 9 次作業(進度: 政大蔡教授講義 7-1 ~ 7-8):

參考講義 7-5 「做鳶尾花的分類」, 訓練集 X2 擷取了資料集 X(鳶尾花的花萼、花瓣之長與寬)的第 2, 3 行的資料, 即花瓣的長與寬

印出鳶尾花資料集之資訊

```
[3] 1 iris = load_iris()
    1 print(iris.DESCR)
[] 1 X = iris.data
[] 1 Y = iris.target
```

讀取鳶尾花資料集

讀取鳶尾花資料之分類, 共計三類

```
[65] 1 X2 = X[:, 2:]
```

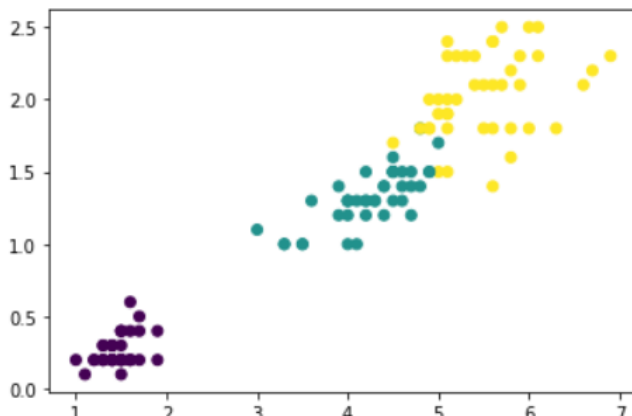
花瓣之長與寬

```
1 from sklearn.model_selection import train_test_split
```

```
[55] 1 x_train, x_test, y_train, y_test = train_test_split(X2, Y, test_size=0.2, random_state=87)
```

```
1 plt.scatter(x_train[:,0], x_train[:,1], c=y_train)
```

<matplotlib.collections.PathCollection at 0x7f85b627a710>



問題:

- (1) 若是讀取鳶尾花的**花萼之長與寬**，重複原程式的過程，請問該如何修改。
- (2) 使用鳶尾花的**花萼之長與寬**，參考 **Kmeans** 自動分類方式(講義 7-7 「Kmeans 自動分類」)，若分為 3 類，請比較 Kmeans 自動分類與原來已分好類別的差異性。

2. 蔡炎龍教授－Python 實現人工智慧

第 10 次作業:

參考: 政大蔡教授「Python 實現人工智慧」講義 2-1 ~ 2-8

問題: 該講義課程所提類神經網路, 正確率僅有 20%, 請問如何改善它, 使正確率提升到 90% 以上。

提示: 請修改以下之參數

修改, 如用 10, 20, 30, ...

```
model.add(Dense(4, input_dim=784))  
model.add(Activation('sigmoid'))
```

第二層 hidden layer 因為前面輸出是 4, 現在輸入是 2, 就不用再說了! 這裡的

```
model.add(Dense(2))  
model.add(Activation('sigmoid'))
```

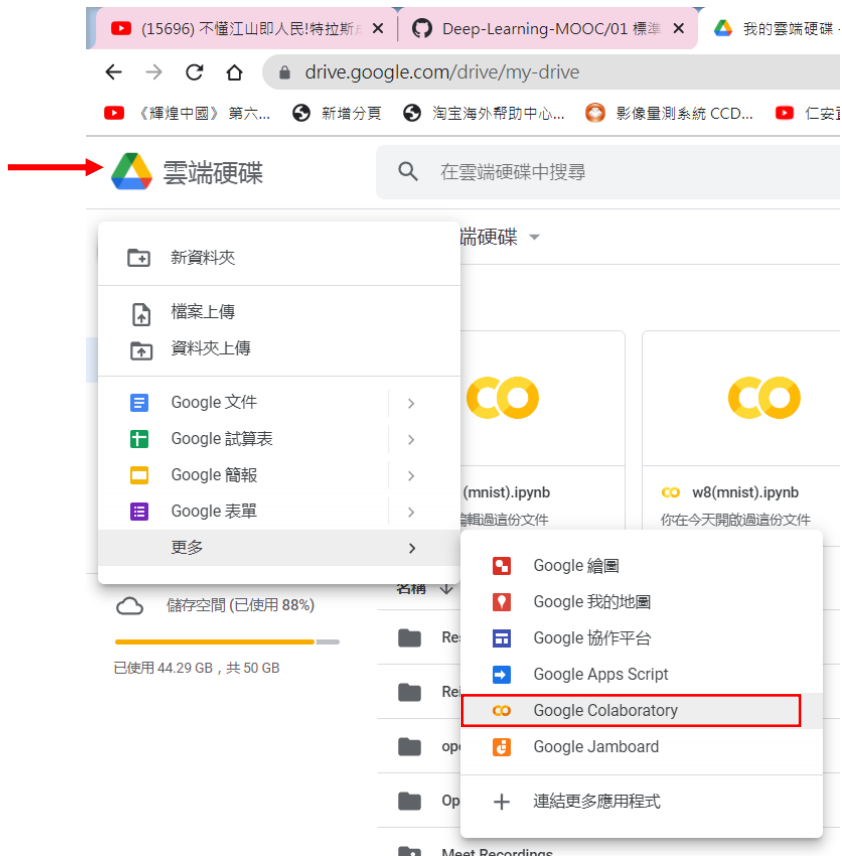
輸出有 10 個數字, 所以輸出層的神經元是 10 個! 而如果我們的網路輸出是

修改為 50

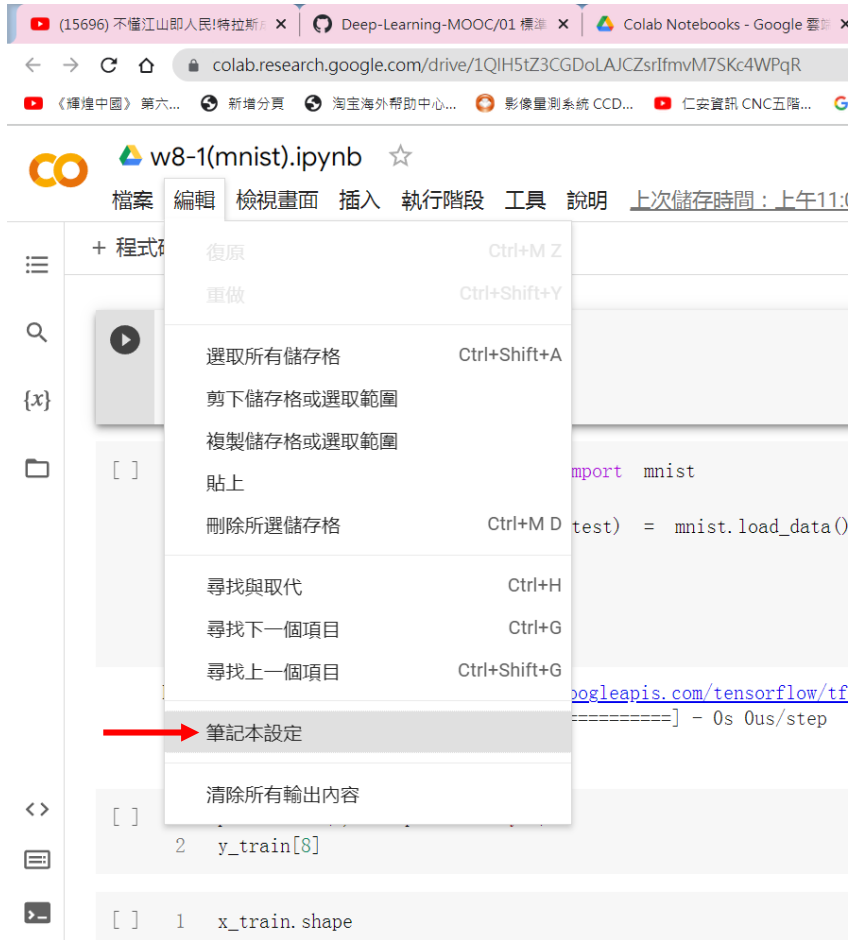
```
model.fit(x_train, y_train, batch_size=100, epochs=20)
```

使用 google colab :

- 登入 google email
- 開啓 雲端硬碟
- 使用 左上角 “增加”, 開啓 colab, 如下圖



- 設定筆記本，選用 "GPU" 增加執行運算速度



筆記本設定

硬體加速器

→ GPU ?

如要發揮 Colab 的最大效益，請避免在不必要的情況下使用 GPU。

[瞭解詳情](#)

儲存這個筆記本時，忽略程式碼儲存格輸出內容

取消

儲存

第 11 次作業:

參考政大蔡教授「成為 python 數據分析達人的第一門課」講義 7-5

「做鳶尾花的分類」，由資料庫 `sklearn.datasets` 讀入 鳶尾花 150 組資料(含資料組集合 X ，分類組集合 Y)，製作三層”類神經”網路(參考本題下方)，然而它不是完整程序，完整程序應包括: 資料讀取與準備、”類神經”網路設計製作、組合與測試。

- 資料讀取與準備:

(1) 讀取鳶尾花 150 組資料，並將該資料隨機分為兩組，訓練

組(x_{train}, y_{train})與測試組(x_{test}, y_{test})，分別具有 120 與

30 組資料， 使用指令如

```
train_test_split(X, Y, test_size= 0.2, random_state = 87)
```

其中 資料組集合 $X = x_{train} + x_{test}$ ，分類組集合 $Y =$

$y_{train} + y_{test}$

(2) y_{train} 與 y_{test} 需做 1-hot-encoding，使用指令如

```
to_categorical(y_train, 3)
```

因為 y_{train} 包含 3 類，所以設定 3

在該指令中。

- “類神經”網路設計製作、組合與測試，如下圖。

```
1 from tensorflow.keras.models import Sequential
2 from tensorflow.keras.layers import Dense, Activation
3 from tensorflow.keras.optimizers import SGD
4
5 x_train.shape
```

(120, 4)

```
[ ] 1 model = Sequential()
2     model.add(Dense(4, input_shape=(4,), activation='relu', name='fc1'))
3     model.add(Dense(2, activation='relu', name='fc2'))
4     model.add(Dense(3, activation='softmax', name='output'))
```

```
[ ] 1 model.compile(loss='mse', optimizer=SGD(learning_rate=0.087), metrics=['accuracy'])
```

```
1 model.summary()
```

```
[ ] 1 model.fit(x_train, y_train, batch_size=5, epochs=20)
```

```
[ ] 1 score = model.evaluate(x_test, y_test)
2     print('loss:', score[0])
3     print('正確率', score[1])
```

```
1/1 [=====] - 0s 98ms/step - loss: 0.1001 - accuracy: 0.7000
loss: 0.10013426840305328
正確率 0.699999988079071
```

問題:

(1)請完善以上程序，使其能正常工作；

(2)該程序僅能達成約 70%正確率，請問該如何修改，使其達成

90%的正確率。

第 12 次作業:

參考: (1)政大蔡教授「Python 實現人工智慧」講義 3-1 ~ 3-11

參考: (2)[資料分析&機器學習] 第 5.1 講: 卷積神經網絡介紹 (Convolutional Neural Network)

<https://medium.com/jameslearningnote/%E8%B3%87%E6%96%99%E5%88%86%E6%9E%90-%E6%A9%9F%E5%99%A8%E5%AD%B8%E7%BF%92-%E7%AC%AC5-1%E8%AC%9B-%E5%8D%B7%E7%A9%8D%E7%A5%9E%E7%B6%93%E7%B6%B2%E7%B5%A1%E4%BB%8B%E7%B4%B9-convolutional-neural-network-4f8249d65d4f>

修改 政大蔡教授的講義例題 “02-1. 用 CNN 做圖形辨識.ipynb”，該例題包括 3 次捲積 2D convolution，每次接 max-pooling (如下圖)，若改為 1 次捲積 2D convolution 與 max-pooling，

問題:

- (1) 請問該程式如何修改，並能完整執行；
- (2) 請問正確率為何。

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 28, 28, 32)	320
activation (Activation)	(None, 28, 28, 32)	0
max_pooling2d (MaxPooling2D)	(None, 14, 14, 32)	0
conv2d_1 (Conv2D)	(None, 14, 14, 64)	18496
activation_1 (Activation)	(None, 14, 14, 64)	0
max_pooling2d_1 (MaxPooling2D)	(None, 7, 7, 64)	0
conv2d_2 (Conv2D)	(None, 7, 7, 128)	73856
activation_2 (Activation)	(None, 7, 7, 128)	0
max_pooling2d_2 (MaxPooling2D)	(None, 3, 3, 128)	0
flatten (Flatten)	(None, 1152)	0
dense (Dense)	(None, 200)	230600
activation_3 (Activation)	(None, 200)	0
dense_1 (Dense)	(None, 10)	2010
activation_4 (Activation)	(None, 10)	0

02-1. 用 CNN 做圖形辨識.ipynb 的類神經網路架構

第 13 次作業:

參考: **Classification Example with Keras CNN (Conv1D) model in Python** ,

<https://www.datatechnotes.com/2020/02/classification-example-with-keras-cnn.html>

參考上述例題(1-D CNN 用於鳶尾花的檢測), 若修改以下程式

- `test_size = 0.3`

```
1 xtrain, xtest, ytrain, ytest=train_test_split(x, y, test_size=0.3)
```

- `Conv1D(32, 2,.....)`

```
1 model = Sequential()
2 model.add(Conv1D(32, 2, activation="relu", input_shape=(4,1)))
3 model.add(Dense(16, activation="relu"))
4 model.add(MaxPooling1D())
5 model.add(Flatten())
6 model.add(Dense(3, activation = 'softmax'))
7 model.compile(loss = 'sparse_categorical_crossentropy', optimizer = "adam", metrics = ['accuracy'])
8 model.summary()
```

問題:

- (1) 請問修改程式後, 正確率為何,
- (2) `confusion_matrix` 之輸出為何,
- (3) 除 `test_size = 0.3` 不能變動外, 如何修改程式提高正確率, 其 `confusion_matrix` 之輸出為何, 例如取消 `model.add(Flatten())` 等, 並說明原因為何如此修改。

第 14 次作業:

參考政大蔡教授「Python 實現人工智慧」，請試用例題「04-1. 用 RNN 做情意分析」，其測試集正確率不高於 85%，

問題: ，

(1) 為何 LSTM 需要 128 個單元，不能少些嗎?若使用少些單元，如 127 等，會發生什麼問題。

```
1 model = Sequential()
```

```
1 model.add(Embedding(10000, 128))
```

```
1 model.add(LSTM(128))
```

```
1 model.add(Dense(1, activation='sigmoid'))
```