

機器學習 Machine Learning LAB 1

系所：電機所

學號：609415074

姓名：蔡承宏

1. 執行說明(Execution description)：

第一題的程式，我們使用 Random 進行亂數的產生 30 個點，並且藉由 -1 與 1 區分為左跟右各 15 個，而點的 y 軸座標由 $y = m \cdot x + s + r$ 計算出來，最後主程式用 `x = np.arange(30)` 畫出斜線，將紅藍進行切割。

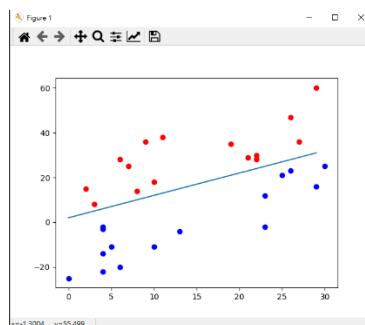
第二題的程式，我們第一題用 random 所產生的出來的數據作為 Dataset，之後以此為基礎再加上 PLA 的程式，在此我先定義出幾個函式，有 PLA 跟 check error，其函式的內容顧名思義，check error 負責判斷紅色跟藍色是否有分類錯誤，假如有錯誤便將錯誤率印出，並且將 iteration 的次數+1，最後即可將 iteration 的次數印出。而 PLA 的函式便是 PLA 演算法的主體，先用 `np.zeros(3)` 建立出填滿 0 的陣列，且當 error 率不為 0 時便不斷的進行 pla 的運算 $w = w + s \cdot x_1$ ，藉由不斷更新權重進行校正直到能夠將紅藍兩色的點分開。

第三題的程式，我們也是基於第一題的 data 進行運算，先將產生出來的 data 改為 2000，之後再以 PLA 的架構下去進行修改，由於 Pocket 演算法會先將權重修正前的結果跟權重修正後的結果進行比較，如果修正後的結果比修正前還差，則不會進行修正，因此需要多設一個變數將原本的資料進行存儲，然後再用 if 來比較，由於題目需要計算執行的時間，因此我們 impor 名為 datetime 的套件，藉由此套件我們可以輕鬆的計算時間。

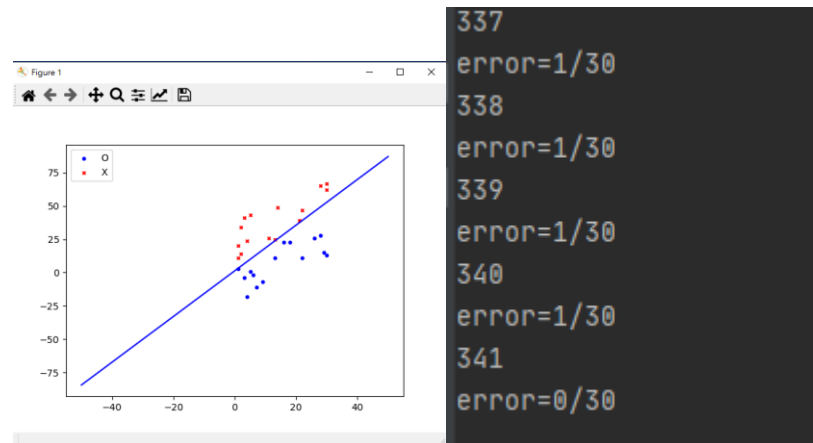
2. 實驗結果(Experimental results)：

第一題產生出 30 個點、第二題以 PLA 將紅藍點分開、第三題計算 PLA 跟 Pocket 演算法的時間並比較。

第一題的圖：

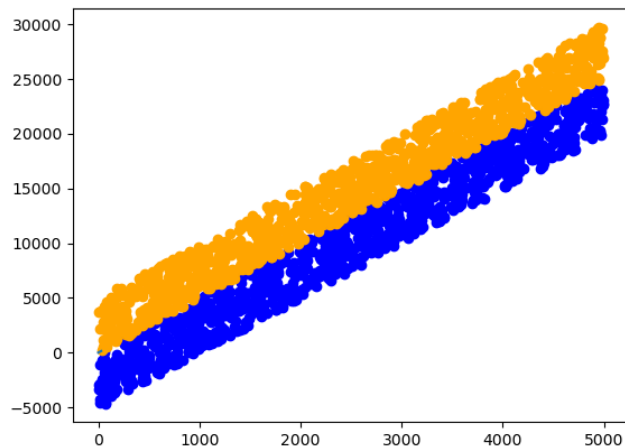


第二題:



結果圖與 iteration 次數

第三題:



PLA 跟 Pocket Algorithm 執行時間&結果圖

```
PLA execution time = 0:00:16.794424
Pocket Algorithm execution time = 0:00:27.985977
```

3. 結論(Conclusion)：

PLA 的演算法比較耗時，並且會有無法收斂導致沒有結果的情況發生，而口袋演算法(Pocket Algorithm)則不會有這樣的問題，因為口袋演算法可以設定 iteration 的上限，一旦到達設定的臨界值將會 break，從原本不斷 iteration 的迴圈中跳出，並且畫出最後的答案。

4. 討論(Discussion)：

在這次的 LAB 作業中，我個人覺得我遇到最大的困難是...我不會 python，所以只能買書、問學長、問同學、上網查資料，不斷地進行學習，而在我從完全不會到現在能夠看得懂這次的作業的過程中，我歷經了千辛萬苦，如我第一項執行說明所說的，第 2、3 題都是基於第 1 題的 Data 進行測試，因此我在於資料轉換的過程中花了非常多的時間，因為在 python 中[]和()似乎有著不同的意思，而且常常跳出 type error，因此我不斷地請教別

人，最後才終於知道要怎麼轉換型態，相信這學期修完這堂課，我的 Python 能力會大幅提升，因為我的起始點是 0。最後很感謝老師與助教願意提供範例 Code 在 Ecourse2 上面，如果連範例程式都沒有，我真的會做到哭，因此再次感謝助教與老師。