**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‧x + s + r計算出來，最後主程式用x = np.arang (30)畫出斜線，將紅藍進行切割。

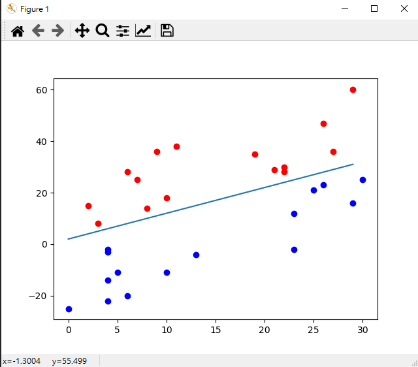
第二題的程式，我們第一題用random所產生的出來的數據作為Dataset，之後以此為基礎再加上PLA的程式，在此我先定義出幾個函式，有PLA跟check error，其函式的內容顧名思義，check error負責判斷紅色跟藍色是否有分類錯誤，假如有錯誤便將錯誤率印出，並且將iteration的次數+1，最後即可將iteration的次數印出。而PLA的函式便是PLA演算法的主體，先用np.zero(3)建立出填滿0的陣列，且當error率不為0時便不斷的進行pla的運算w = w + s‧x1，藉由不斷更新權重進行校正直到能夠將紅藍兩色的點分開。

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datatime的套件，藉由此套件我們可以輕鬆的計算時間。

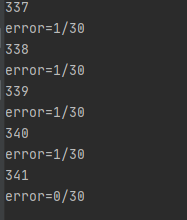
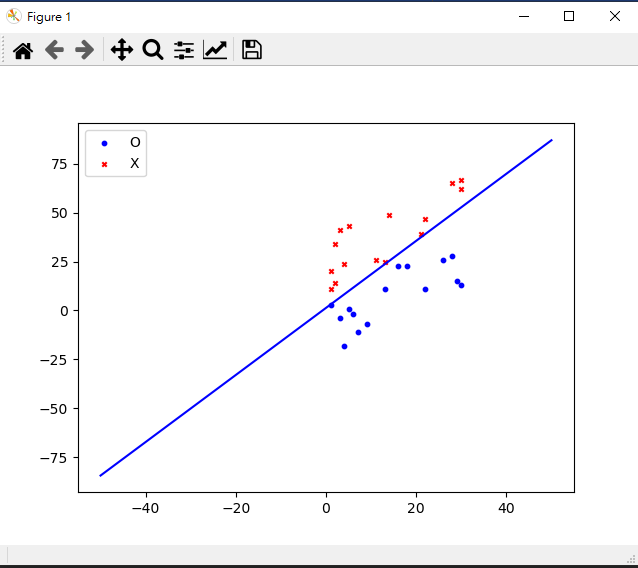
1. 實驗結果(Experimental results)：

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:

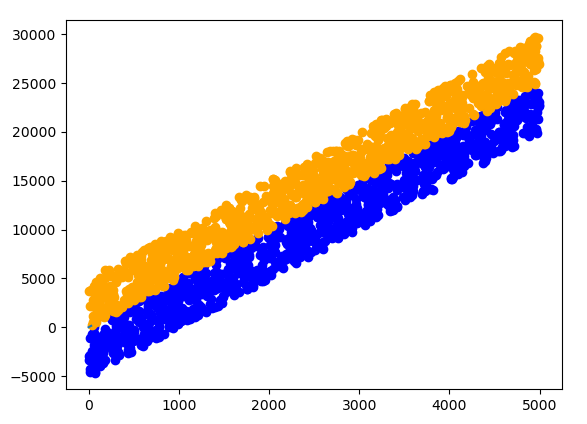


第二題:



結果圖與iteration次數

第三題:



PLA跟Pocket Algorithm執行時間&結果圖



1. 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。

1. 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。