姓名:黃宗德　　系級:資工四　　學號:406410091

Execution description: Please describe the detailed steps how to execute your codes.

P1.py是第一題的程式，來源為範例程式，順利執行即可。
輸入 python P1.py 執行即可。
順利執行後會回傳一張圖片，而圖片中的點不會在線上。

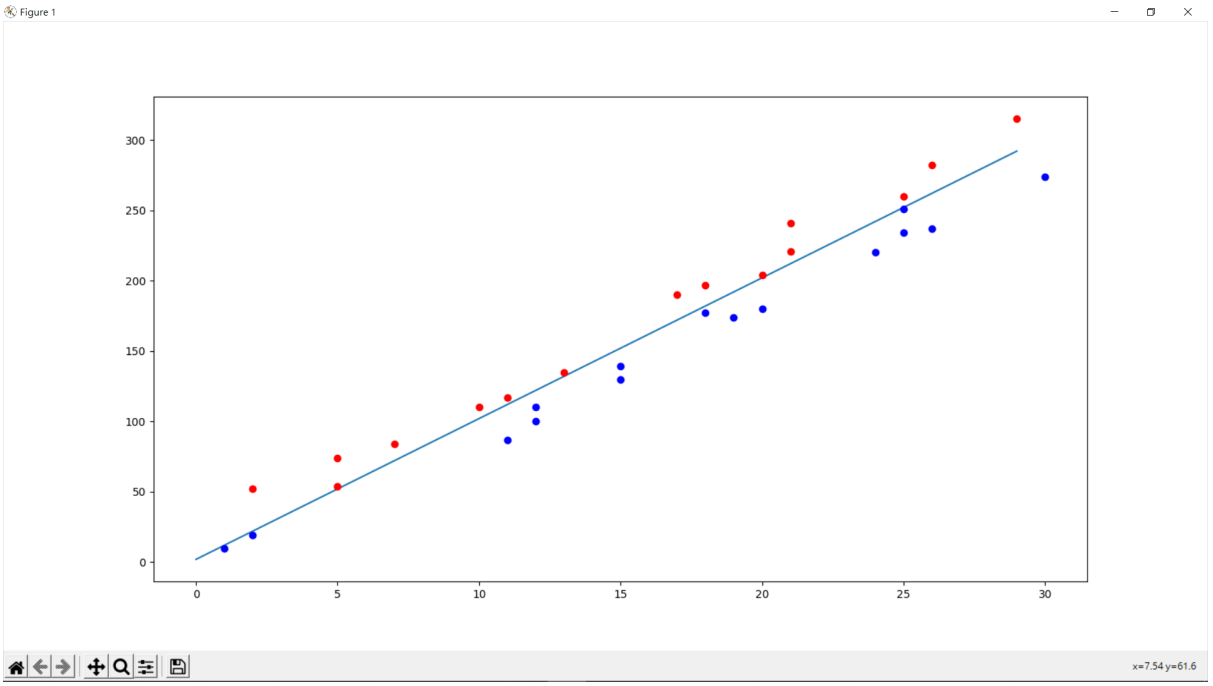
pla.py是第二題的程式。
輸入 python pla.py執行即可。
而其中有三筆sample，分別是dataset,dataset2,dataset3，每筆有30個資料。
順利執行後會回傳三張圖片，為將兩類點分類的線段。
執行時，會輸出error=x/y，其代表錯誤點的比例，x為錯誤點數，y為所有30個資料數。
最後成功切出線會輸出error=0/30。
最後會回傳count，代表iterations的次數，以及回傳三筆sample所iterations的平均次數。

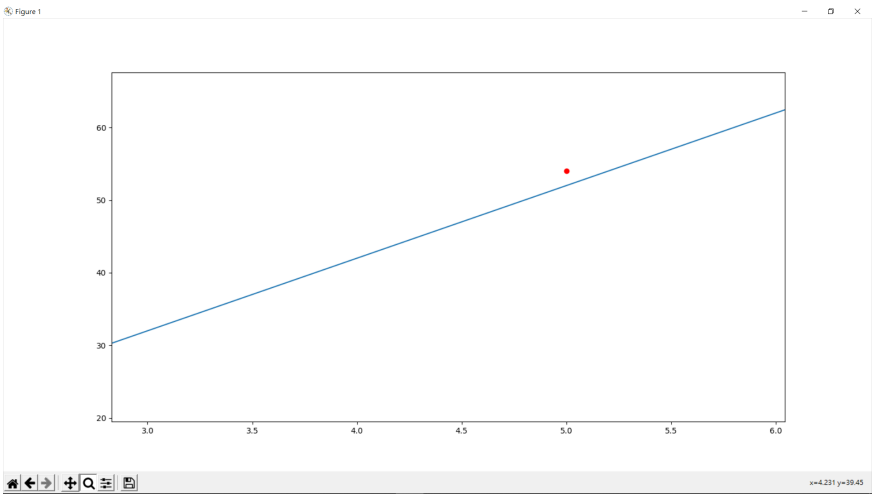
P3.py是第三題程式。
輸入 python P3.py執行即可。
給定pocket algorthim一個執行上限次數，根據測試的觀察，大約跑20次以上結果便固定，而繳交的程式以300次為上限為例。
接著會跑出pocket algorithm所產生的圖，而其產生的圖
最後會劃出一條線，而會有不少點為錯誤分類點，找出最小錯誤分類點數目為其目標。

接著會程式會import使用第二題所寫的PLA來執行sample，但由於sample筆數不少，所以需要跑一段時間，大約數分鐘(3mins~10mins)，成功執行後會回傳一條線將兩類點完全分類。

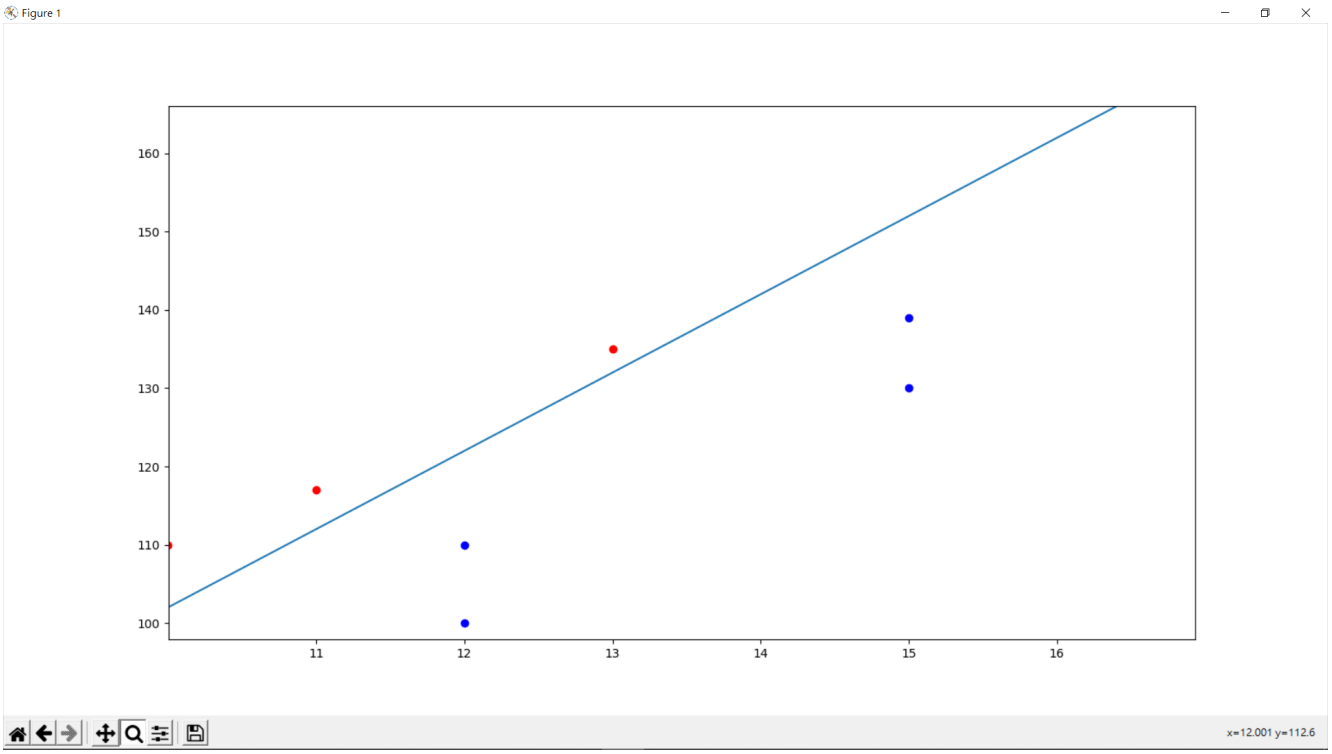
Experimental results: As specified in the assignment.

1.
透過範例程式，所產生的samples，而有些點看似在線上，但將其放大後，便可發現其不並不在線上。





放大上上圖中，座標5點多的紅色點為例，可發現其不在線上。



放大上上上圖中，座標點接近13的紅色點為例，可發現其不在線上，舉此兩點為例，其他類似的點不依依放大，結果都相同。。

2.

產生三次sample，每次總共30筆sample為例。

(第一次) X座標6的位置有一個紅色X貌似在線上，但放大後發現其不在線上，如下下圖。

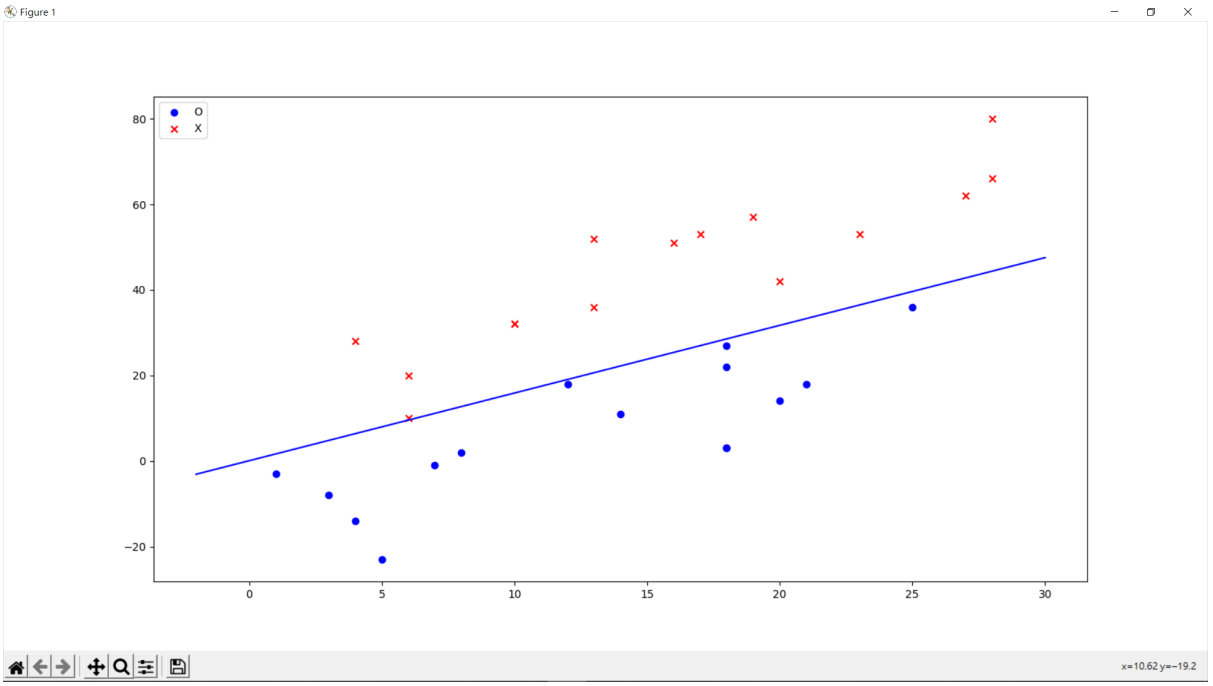


Figure 1
O
X
80
60
40
20
0
−20
0
5
10
15
20
25
30
x=10.62 y=−19.2

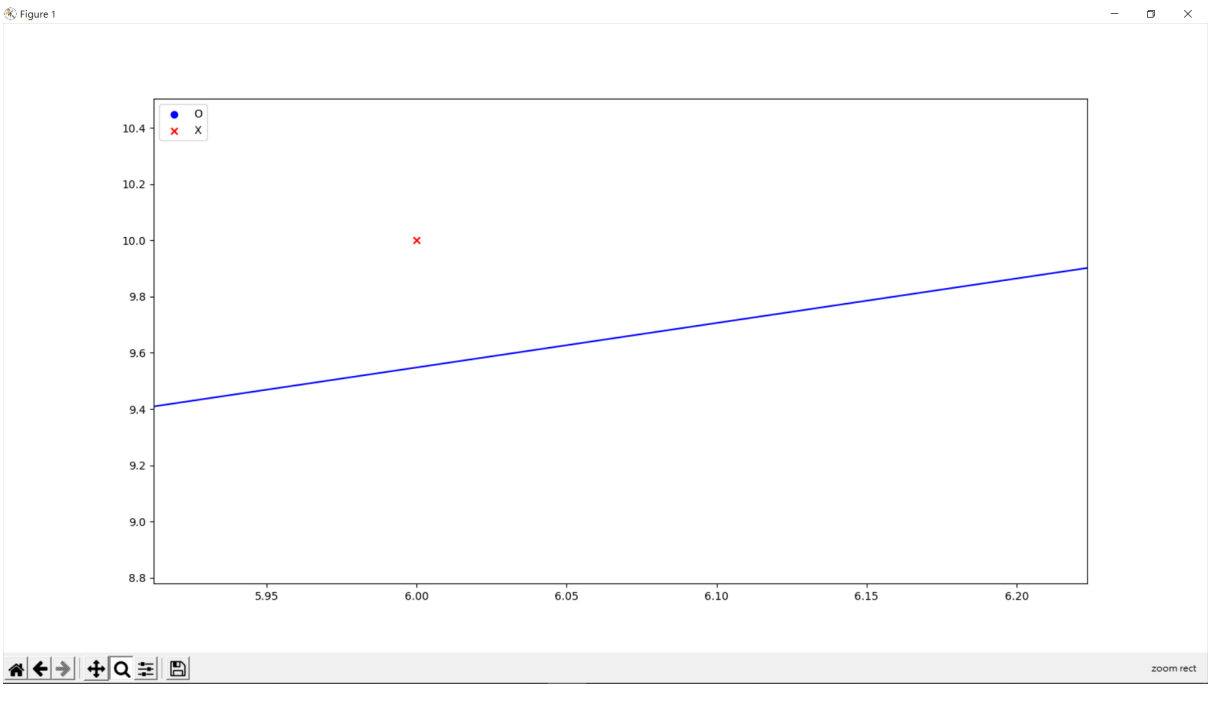
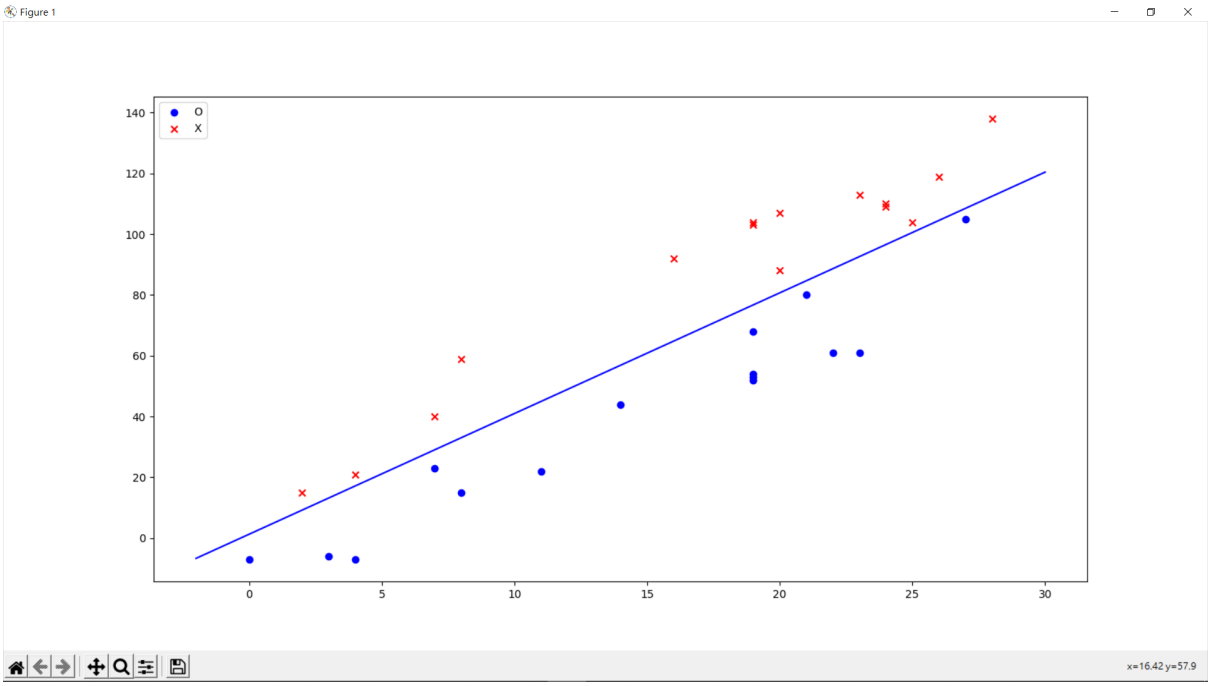
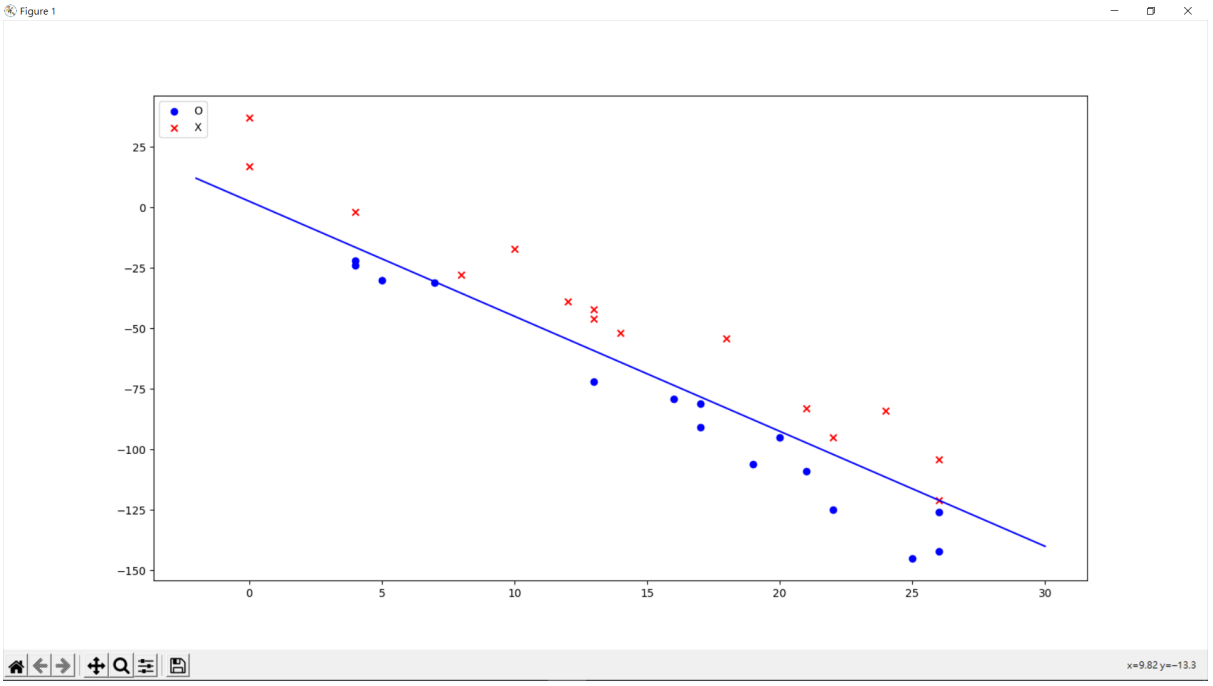


Figure 1
O
X
10.4
10.2
10.0
9.8
9.6
9.4
9.2
9.0
8.8
5.95
6.00
6.05
6.10
6.15
6.20
zoom rect

(第二次)

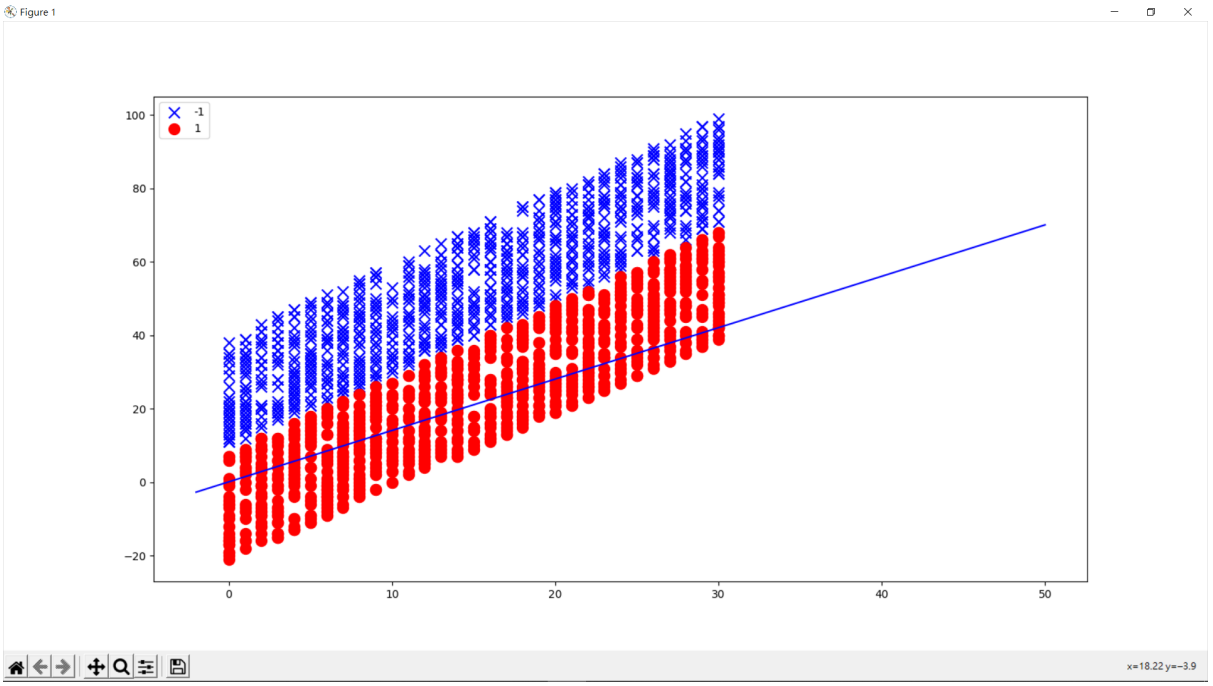


(第三次)

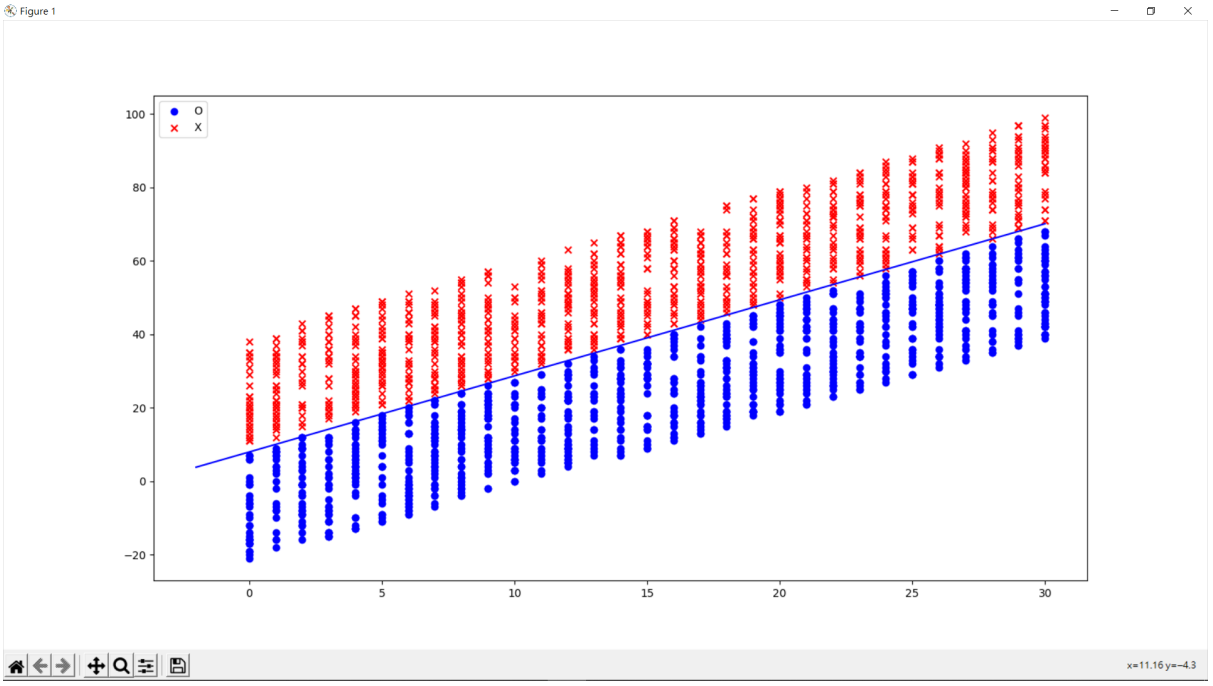


count(總共找了幾次)= 107920
average number of iterations 35973.333333333336

3.



使用pocket algorthm所產生的圖，最小錯誤分類點為574。



使用PLA在2000筆所產生出的圖，筆數不小，導致需要不少時間來成功跑完。

Conclusion: The observation from your results.
依照助教的example所產生的測資，是由一條線y=mx+b，對其值b做加減以1~30值的範圍來進行移動，來產生測試樣本，所以所產生出的資料，理論上都是線性可分的。
由於為線性可分，所以PLA演算法最終都會找到一條線將兩類點所分類。
而pocket algorithm會盡可能的縮小錯誤分類點的數目。
而第三題中pocket algorithm會給定上限次數，且在次數不大的情況下，便能跑出最後固定的結果，而PLA則需要跑數分鐘才得以停止回傳正確值，所以在時間上來看，pocket algorthin勝於PLA，但是助教所產生的測資線性可分，因此，若將把準確率算入，pocket所產生的錯誤率

遠大於PLA, 所以將準確率算入的話, PLA跟pocket相比, PLA在線性可分時, 整體上優於pocket algorithm。

Discussion: The questions or the difficulties you met during the implementation.
在筆數2000時, PLA需要不少時間才得以成功執行完成, 效率極差, 大約要5mins左右。
範例所產生的資料，要如何放入PLA及pocket algorthim中執行，PLA所找出的線，要如何使用w來推回, 及做內積運算時, 要如何一一對應乘出正確的結果, 以及判斷所畫出的圖是否符合預期的結果，。