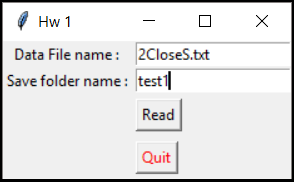
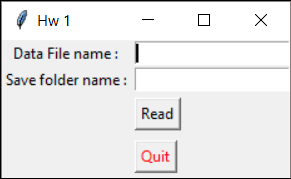
類神經網路

學號：104501527  
姓名：高禾

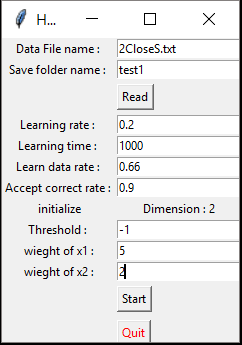
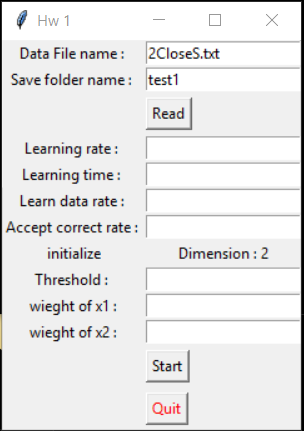
程式執行之簡介與說明：

語言：python3.5

引入的函數庫：matplotlib(繪圖)  
 tkinter(圖形介面)  
 os(寫檔案)  
 random(打亂測資)  
 Axes3D(3D圖表)  
 numpy(為了產生連續數列，在3D圖表上使用)

  
Data File name 為開啟與想分析的data檔案名稱  
Save folder name為想要儲存學習過程的資料的檔案位置

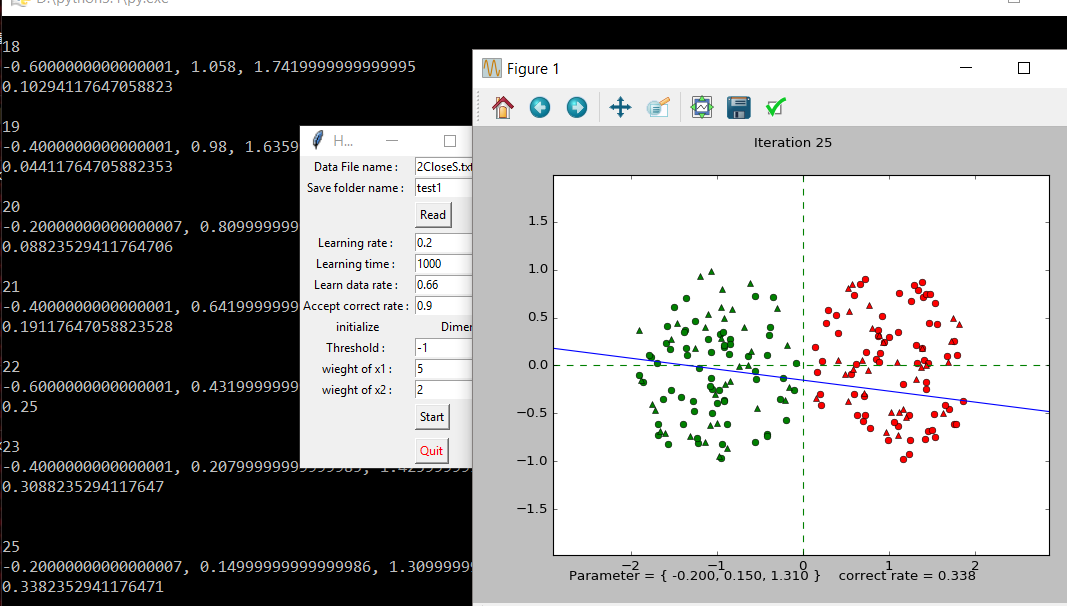
按下read 之後便會進入到主畫面



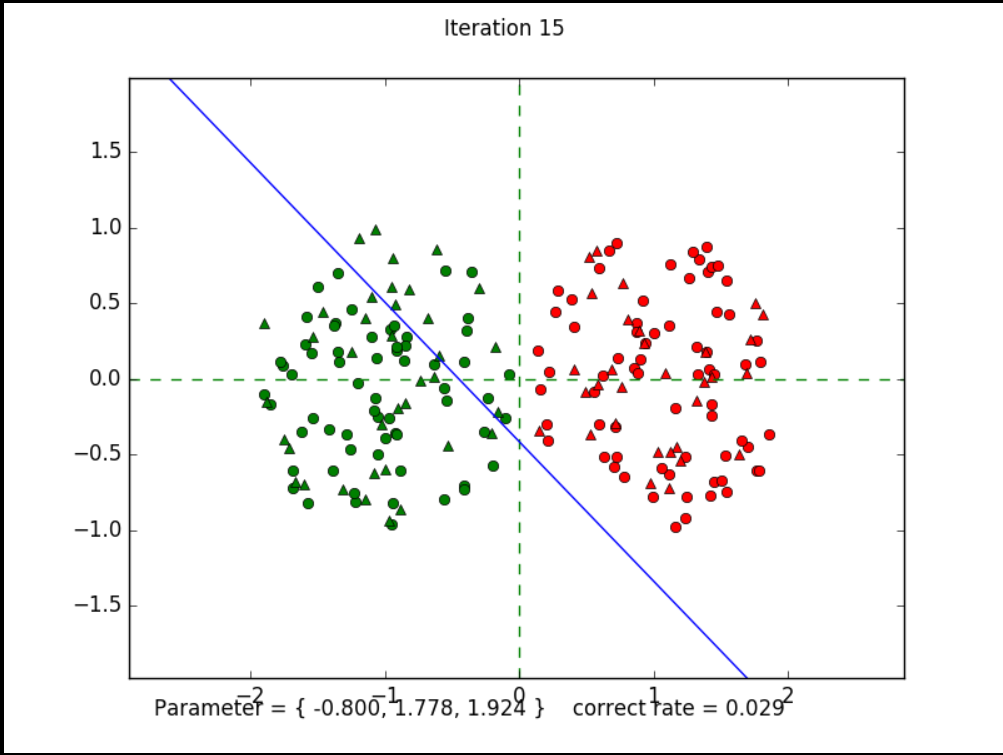
而可以填寫的資料分別是

學習率  
學習次數  
學習的資料百分比  
接受的測驗數據正確百分比  
閥值與其餘間鍵結值得初始化設定

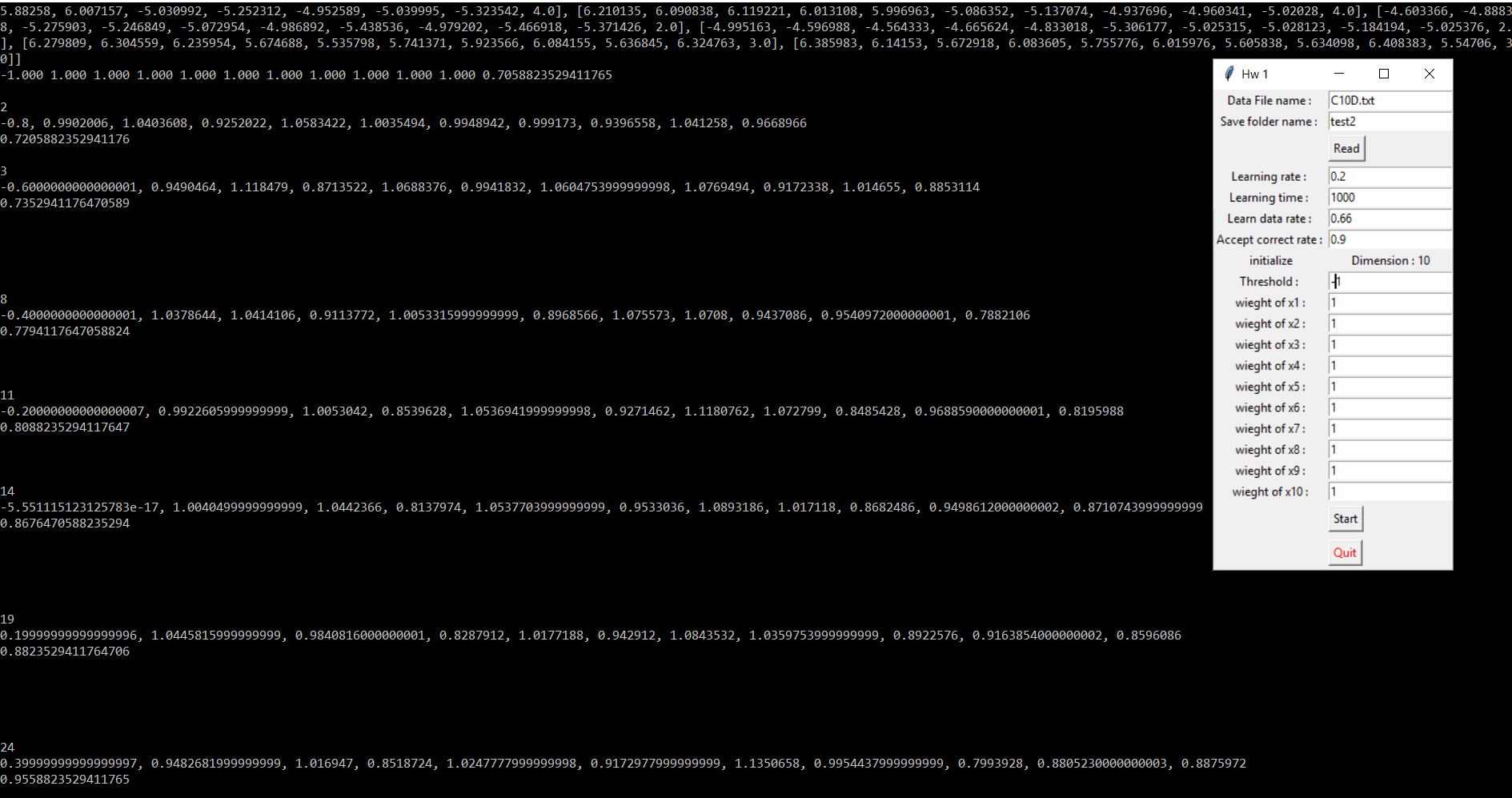
而出來的結果就是第幾次疊代、每一步的鍵結值的更新與測驗值的準確率



但是真正的資料並不是在此，而是在之前設定的儲存位置。



也可以執行隨意幾維的資料



討論：

本來難以了解何時電腦把學習數據學習完畢，但試想一下便能理解為：把目標訂得比100%還要大，換句話說電腦要不斷的學習直到訓練資料都無法再改進了，這時(最後一張圖)，便是完成學習後的結果與測驗分數。

基本上只要是線性可分割的資料都可以在有限的執行次數內完成，但是當學習率過低時，所需的迭代便會需要非常大。而如果是線性不可分割的話，分割線便會不斷的亂跑，畢竟無法找到能夠逼近的答案的方式。

而至於測驗正確率，其實都非常非常的高，我有做打亂資料的處哩，所以有時候電腦只拿到片面資料時，的確是分數比較不理想，但是總體而言基本上都是訓練過後的電腦能夠百分之百答對測驗數據。

在高次元空間的數據，這版本的感知機感覺效率非常的差，正確率並不會穩定上升。而高次元加上多區分割就更顯得無力，所以應該需要用到到傳遞演算法會比較容易解決較複雜的問題。

額外完成：

* 多維資料處裡(無上限)
* 分割不只兩區(無上限)
* 3D的圖表，雖然感覺上還是非常難以產生直覺。
* 做統整，正確率的變動