類神經網路 作業一\_單層感知機 大氣4A 黃展皇 106601015

工作環境：x64 windows10，conda 4.8.3，python3.6.10

1. 程式執行說明

建立conda虛擬環境並且選擇性安裝依賴模組，並且將matplotlib卸載並安裝3.1.1版本，使用pyinstaller -F main.py打包成.exe檔，並且記得將資料夾NN\_HW1\_DataSet複製到.exe檔所在路徑，執行.exe檔後呈現一圖形化介面，可輸入學習率、欲執行資料檔案名稱、最大執行次數，經過計算後分別呈現2維分群結果圖片、訓練與測試準確度以及最終的鍵結值。

1. 程式碼簡介

if \_\_name\_\_ == '\_\_main\_\_':

執行tkinter gui介面

def main():

執行資料讀取、資料解析分割、基礎值設定、誤差迴圈計算、權重調整

def renew\_w(w, d, x, learning\_rate, labels):

根據最小均方誤差法計算誤差並調整權重

def get\_expected\_d(w, x, labels):

硬限制器，權重跟x的內積>0則傳回labels大值，反之則傳回小值

def plot(dataset, w, X\_train, label\_train, X\_test, label\_test):

針對資料點與權重分割線繪圖

def linear\_equations(w, x):

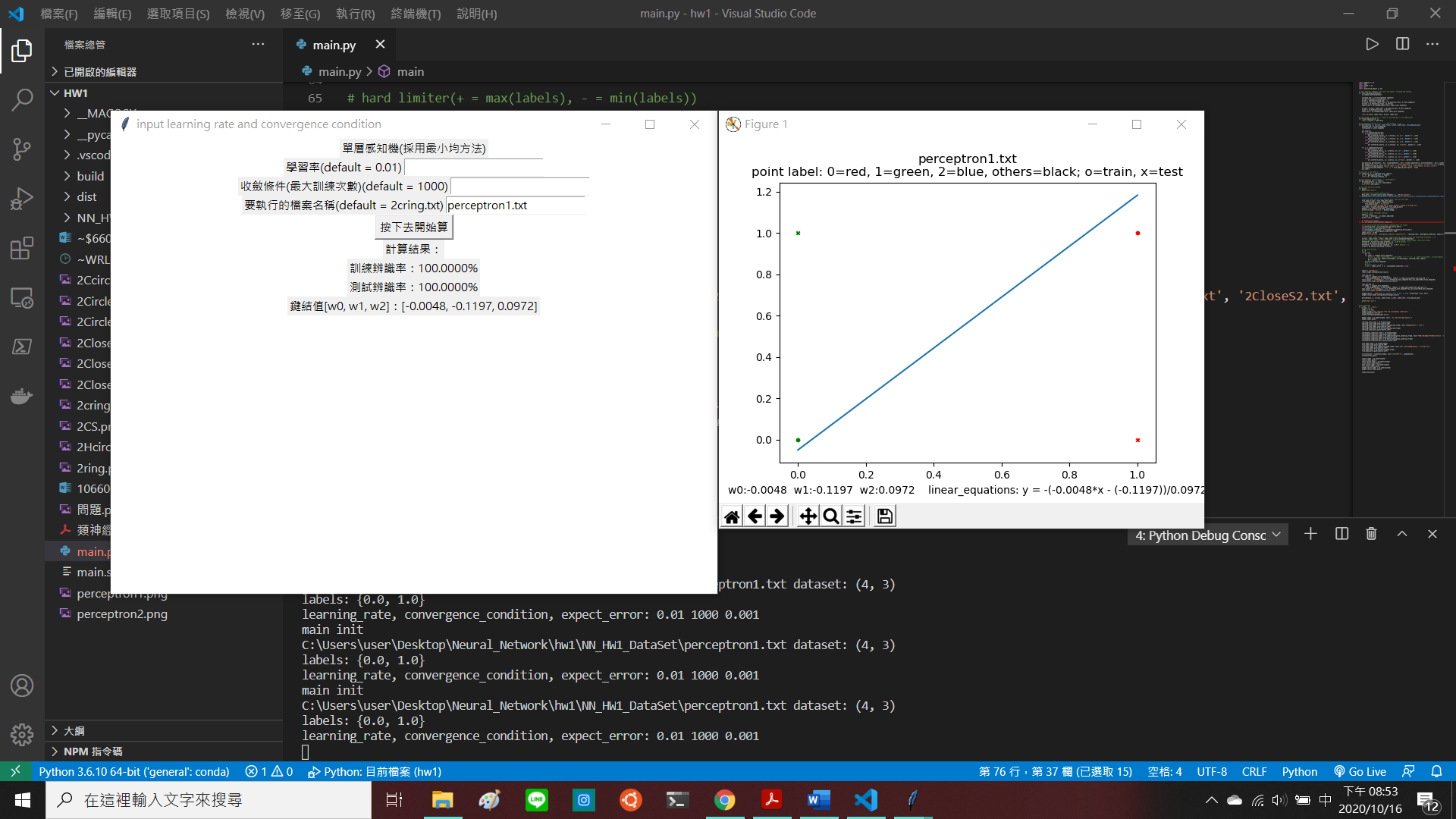
輸入權重回傳線性方程式

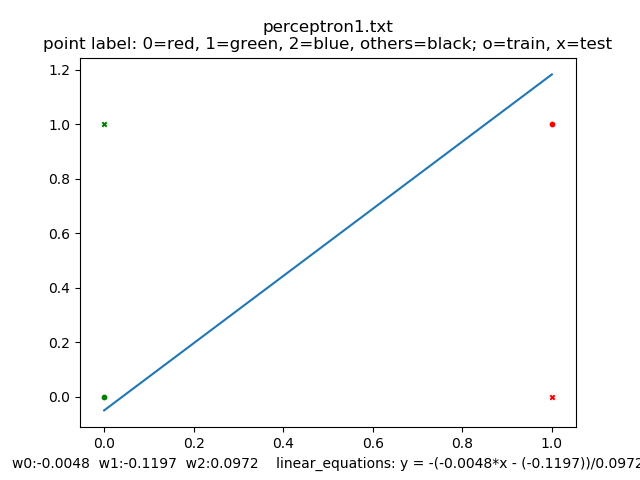
def split\_2D\_dataset(dataset):

輸入原始資料集，分成2/3的訓練資料+labels與1/3的測試資料+labels

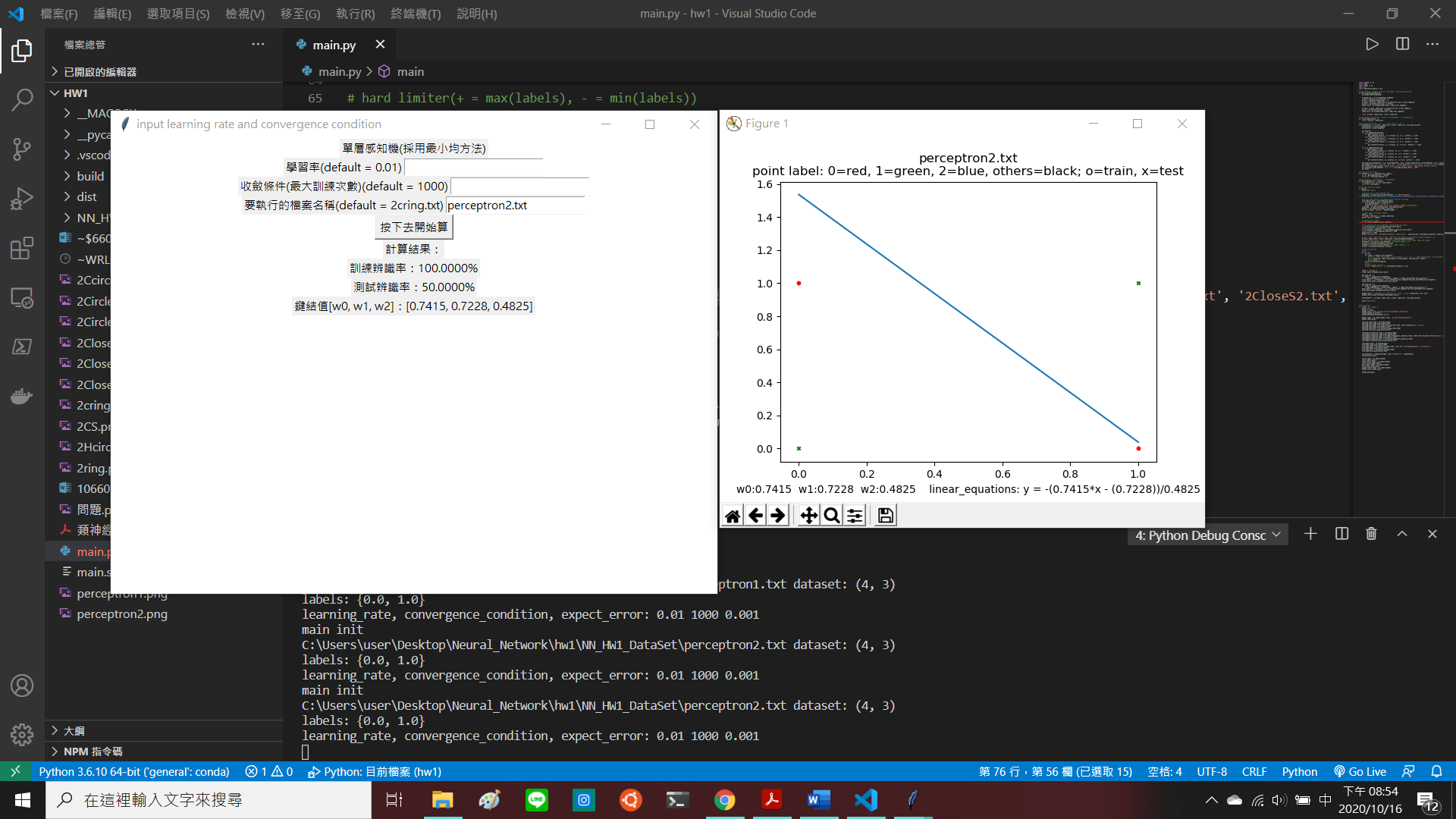
1. 實驗結果 (所有資料集都須有實驗結果和截圖及說明)

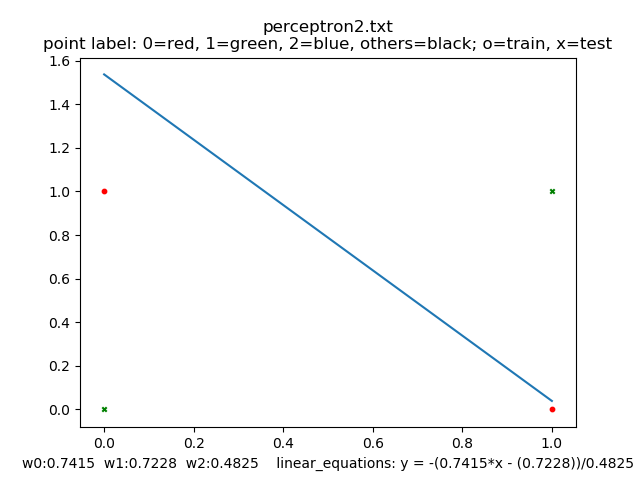
perceptron1.txt



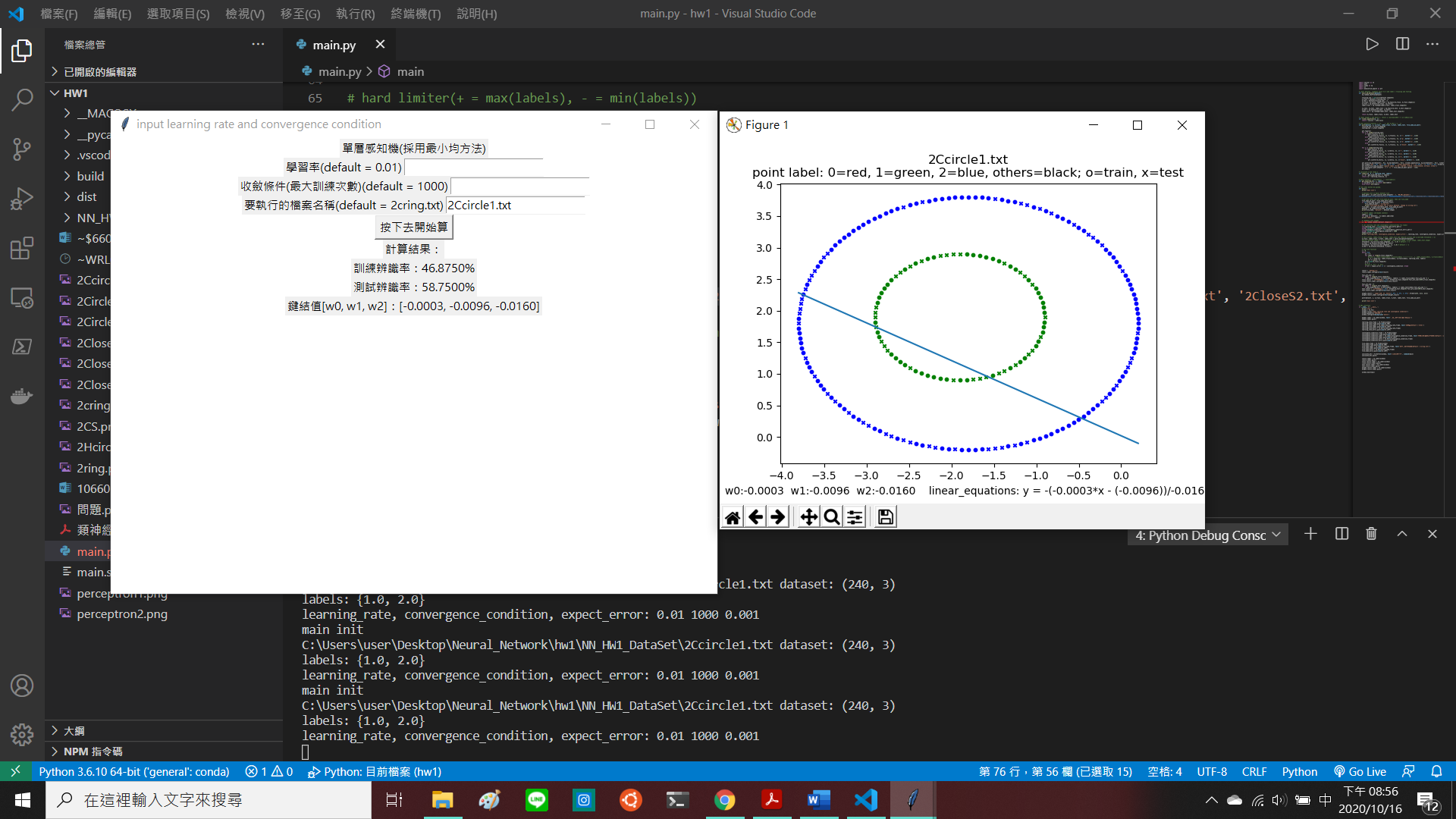


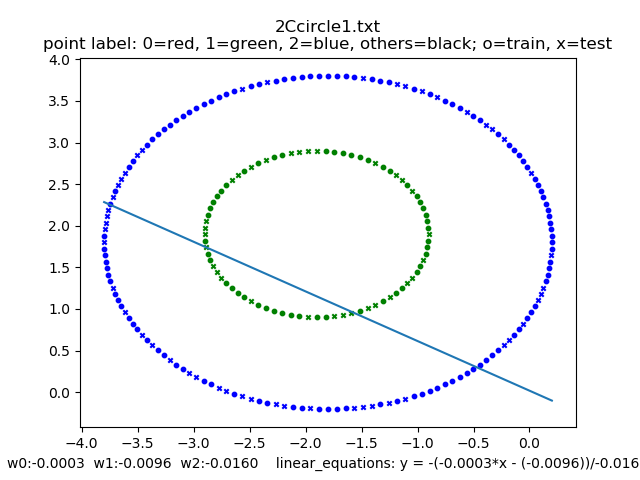
perceptron2.txt



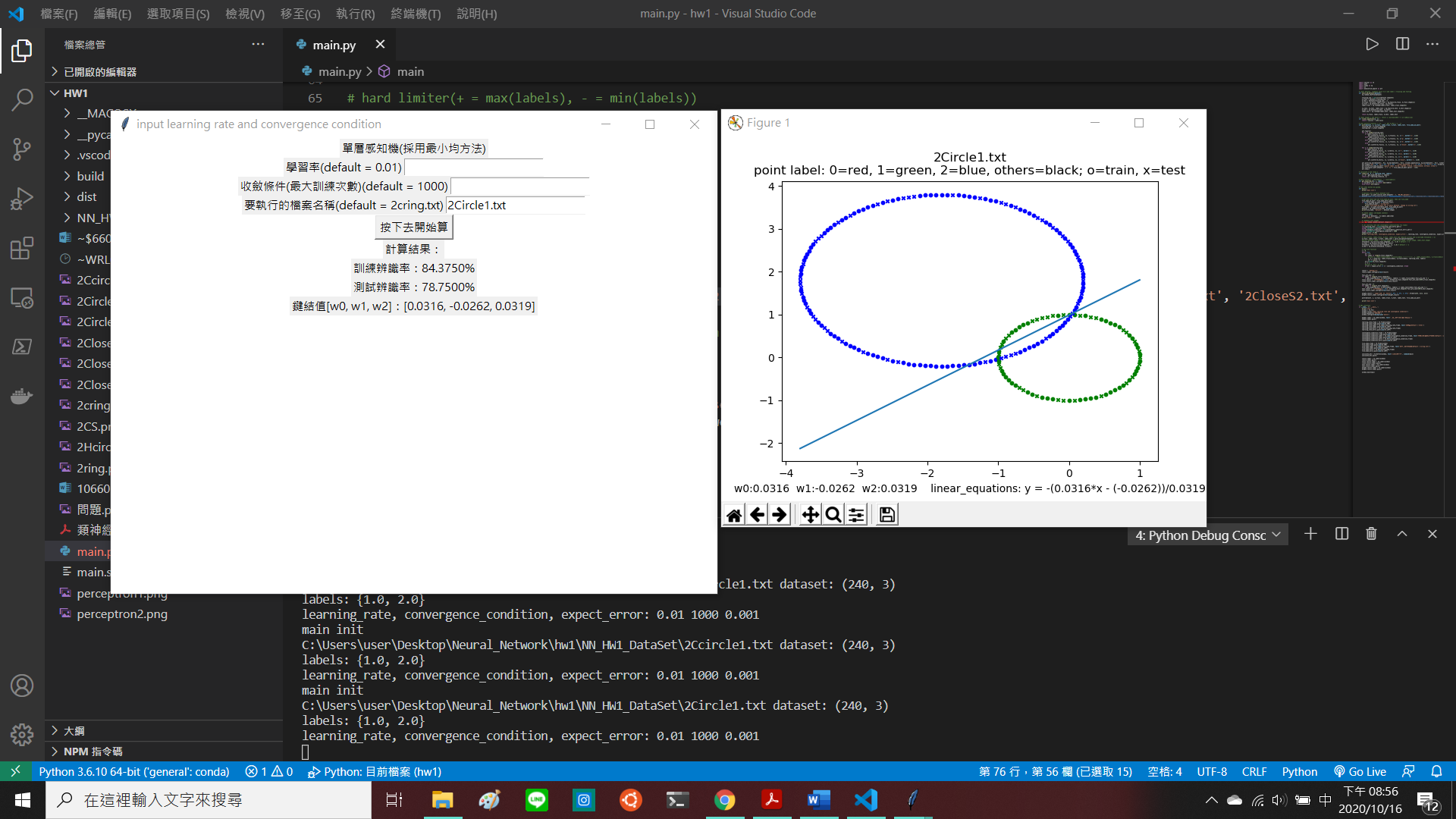


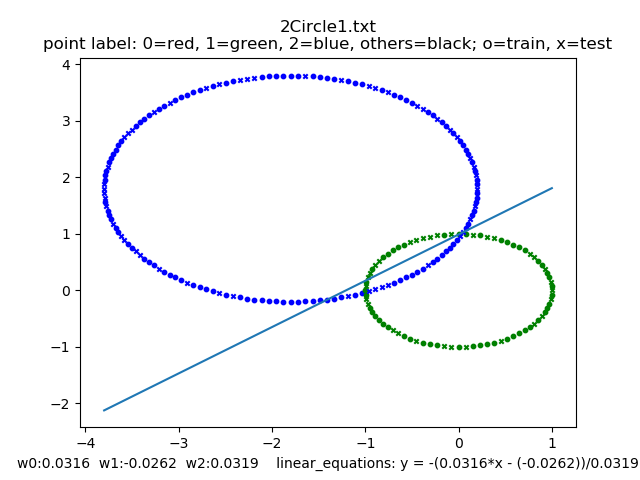
2Ccircle1.txt



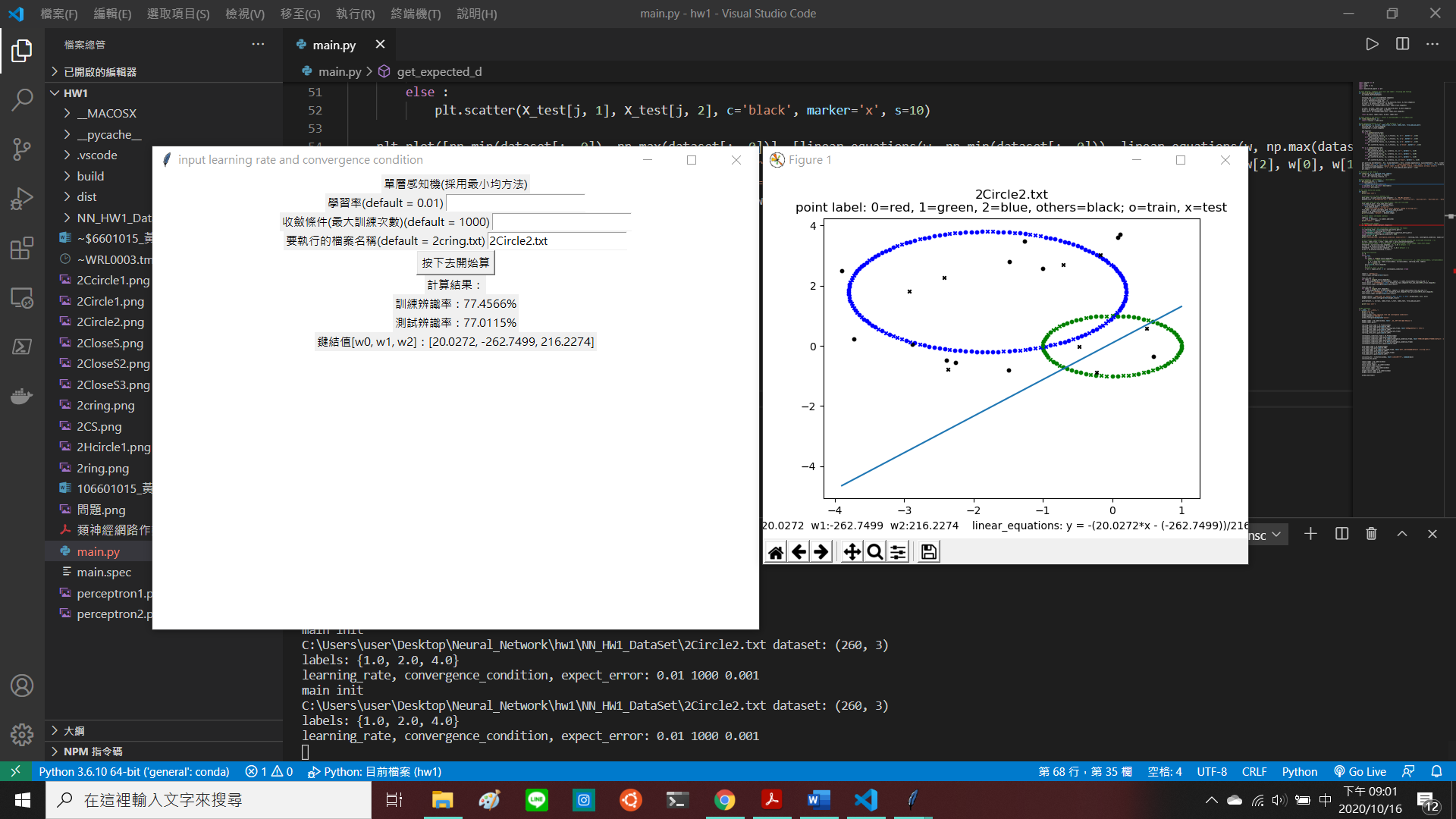


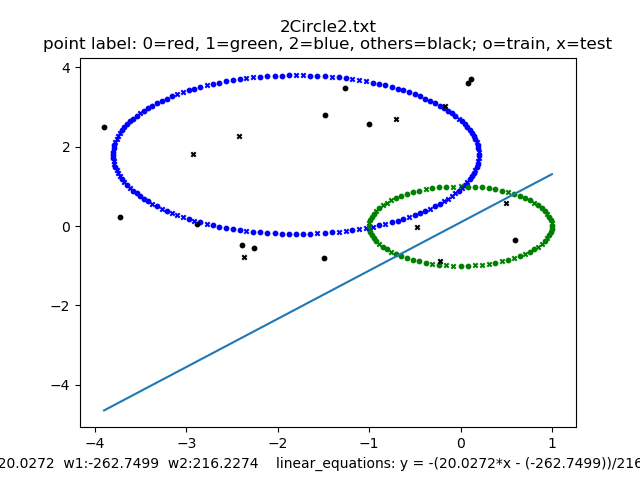
2Circle1.txt



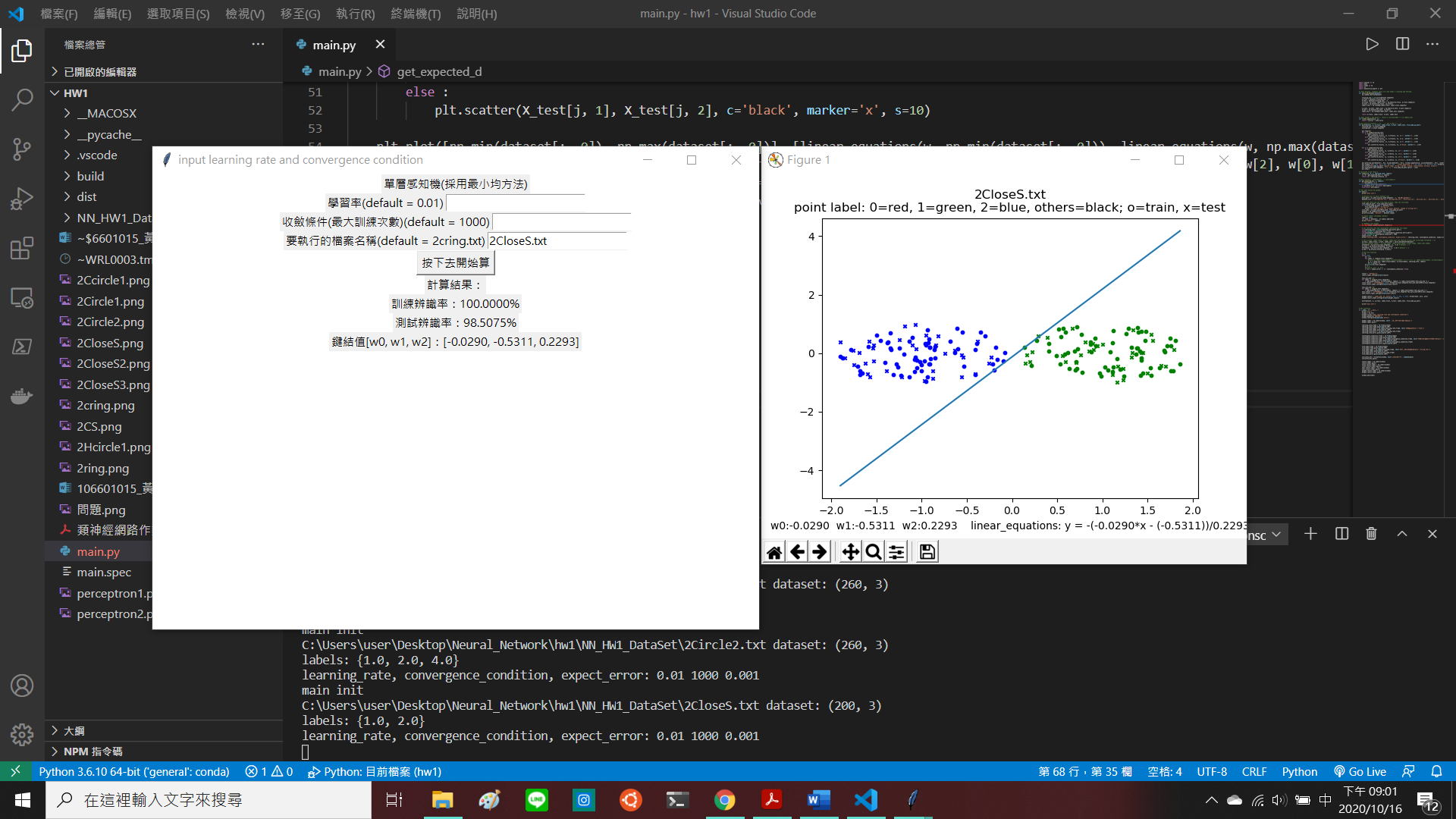


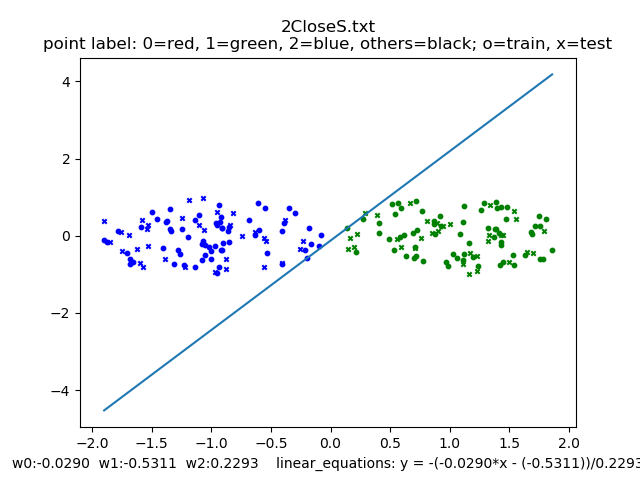
2Circle2.txt



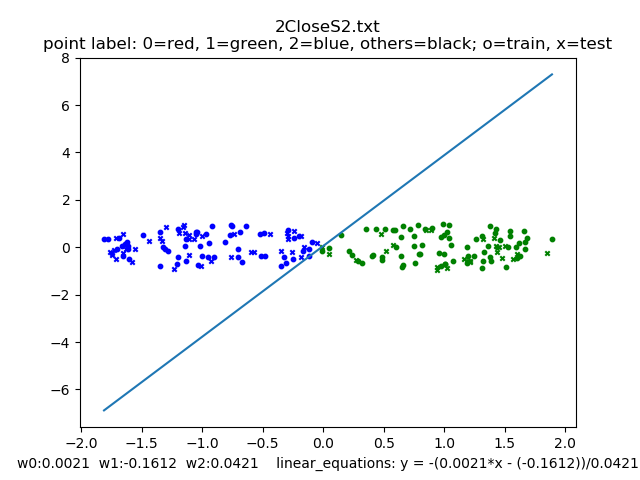
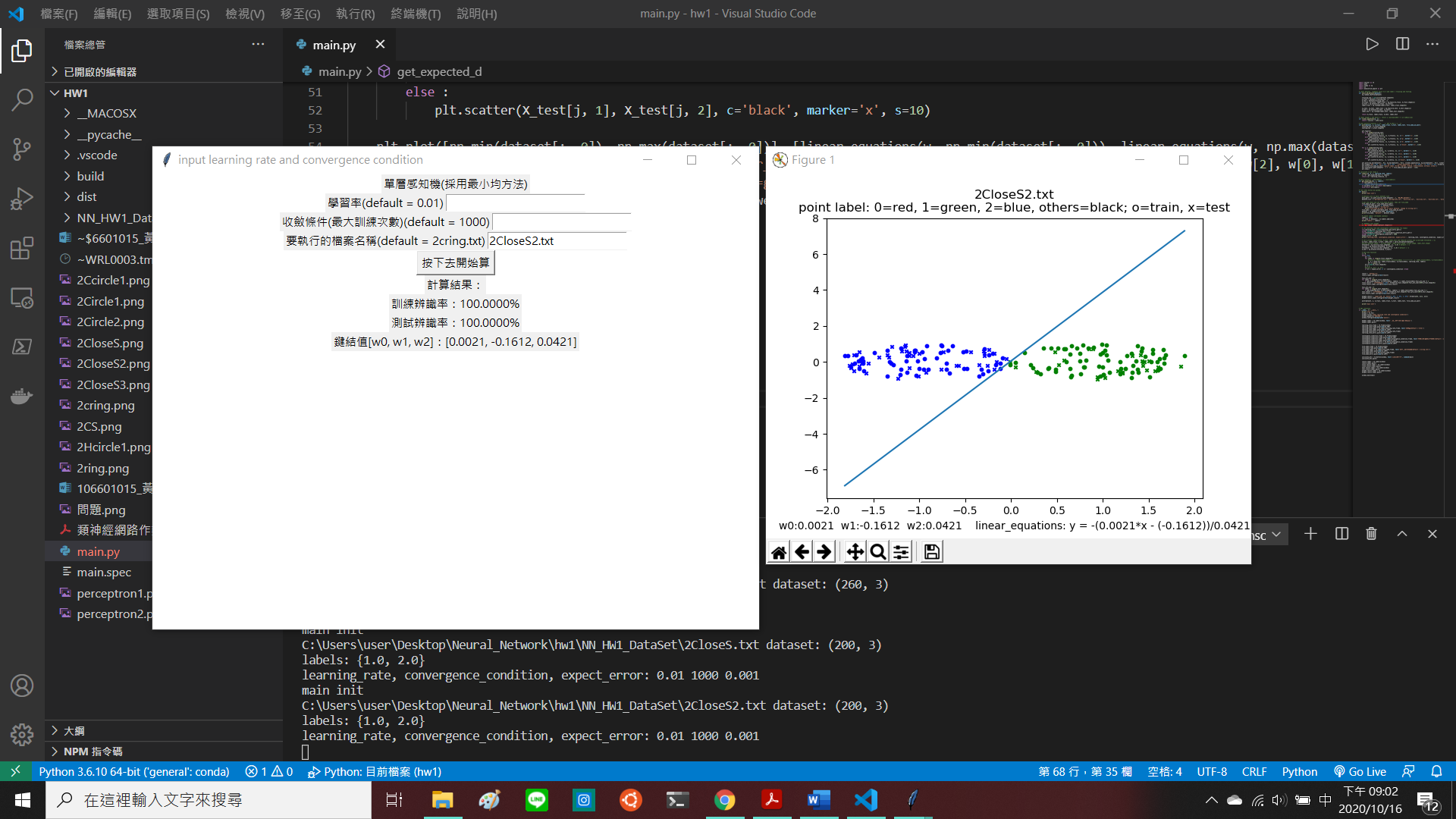


2CloseS.txt

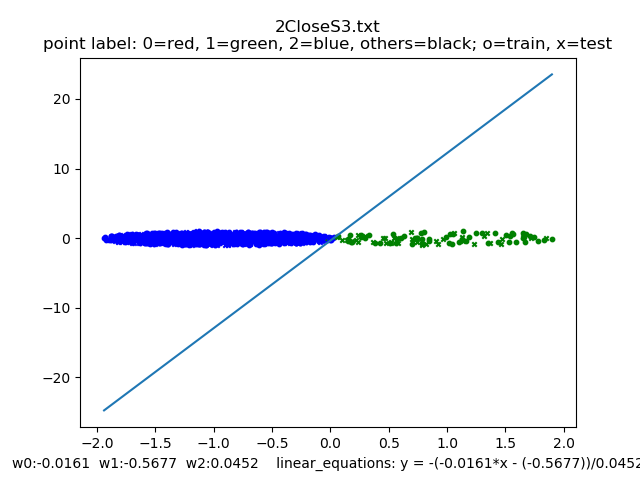
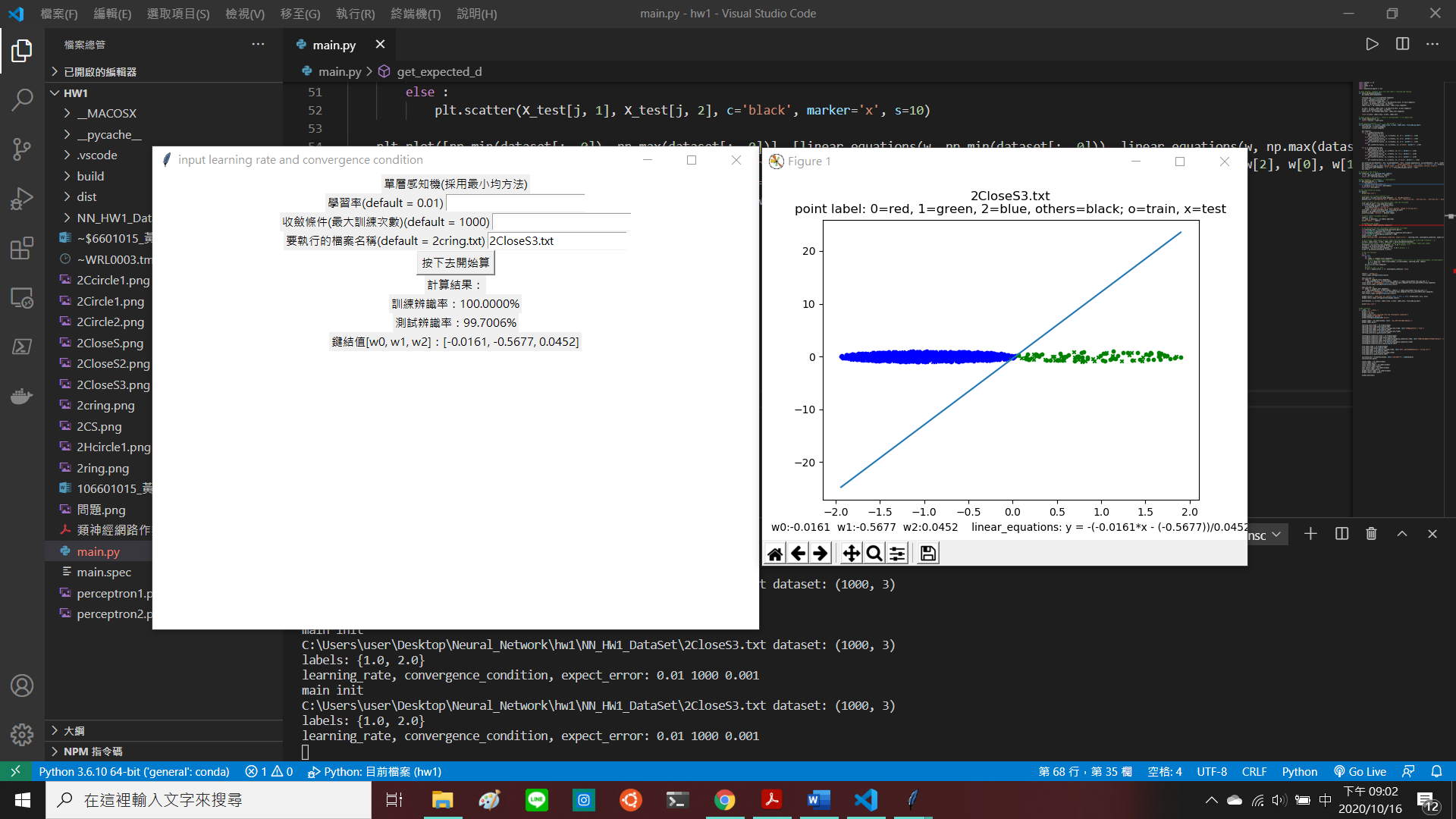




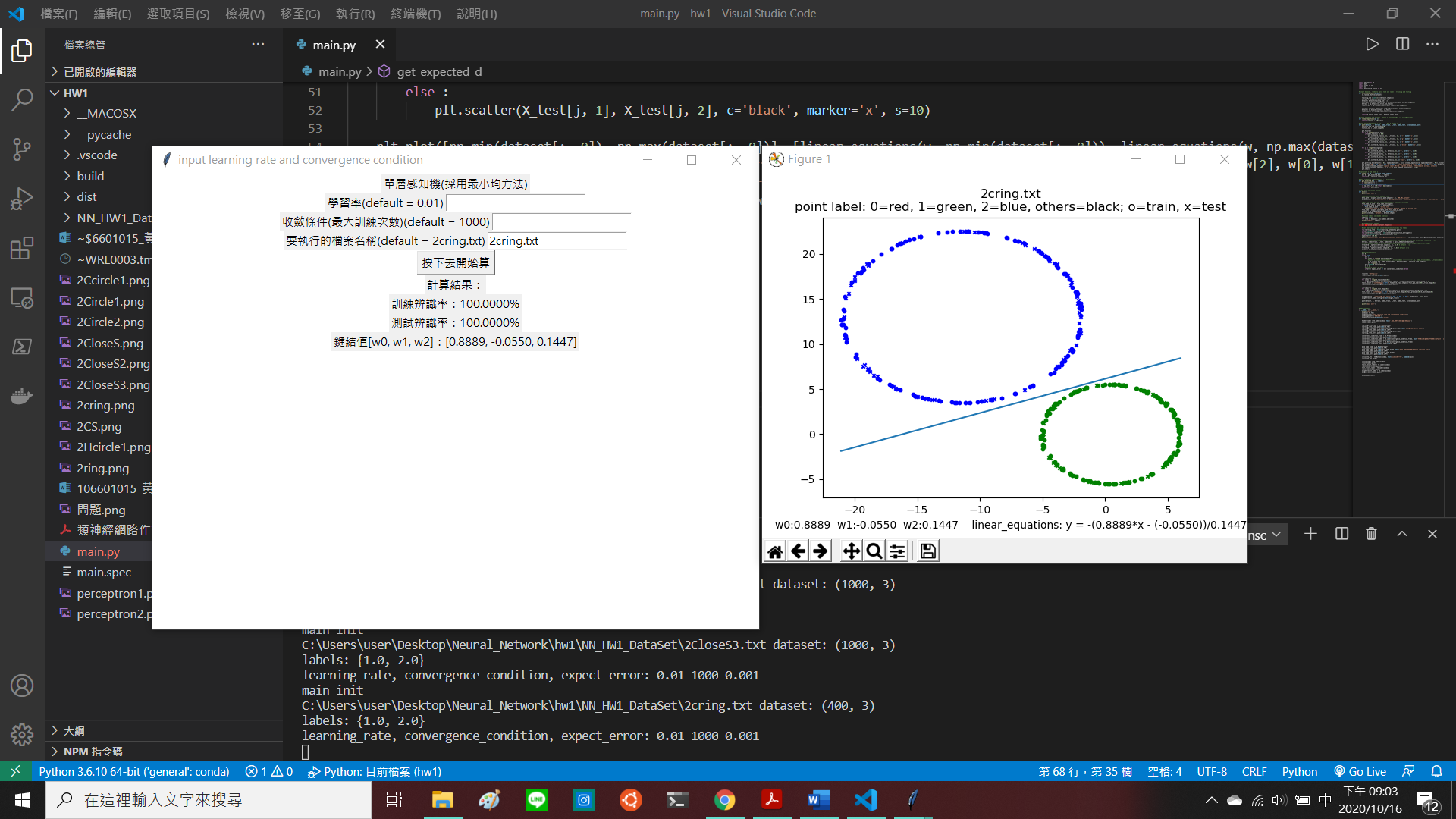
2CloseS2.txt

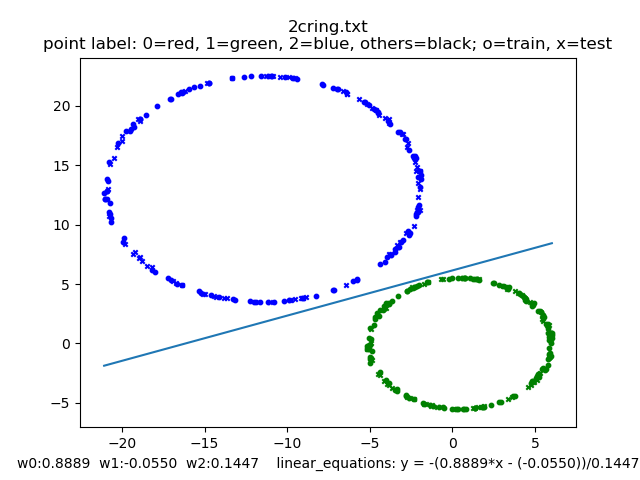


2CloseS3.txt

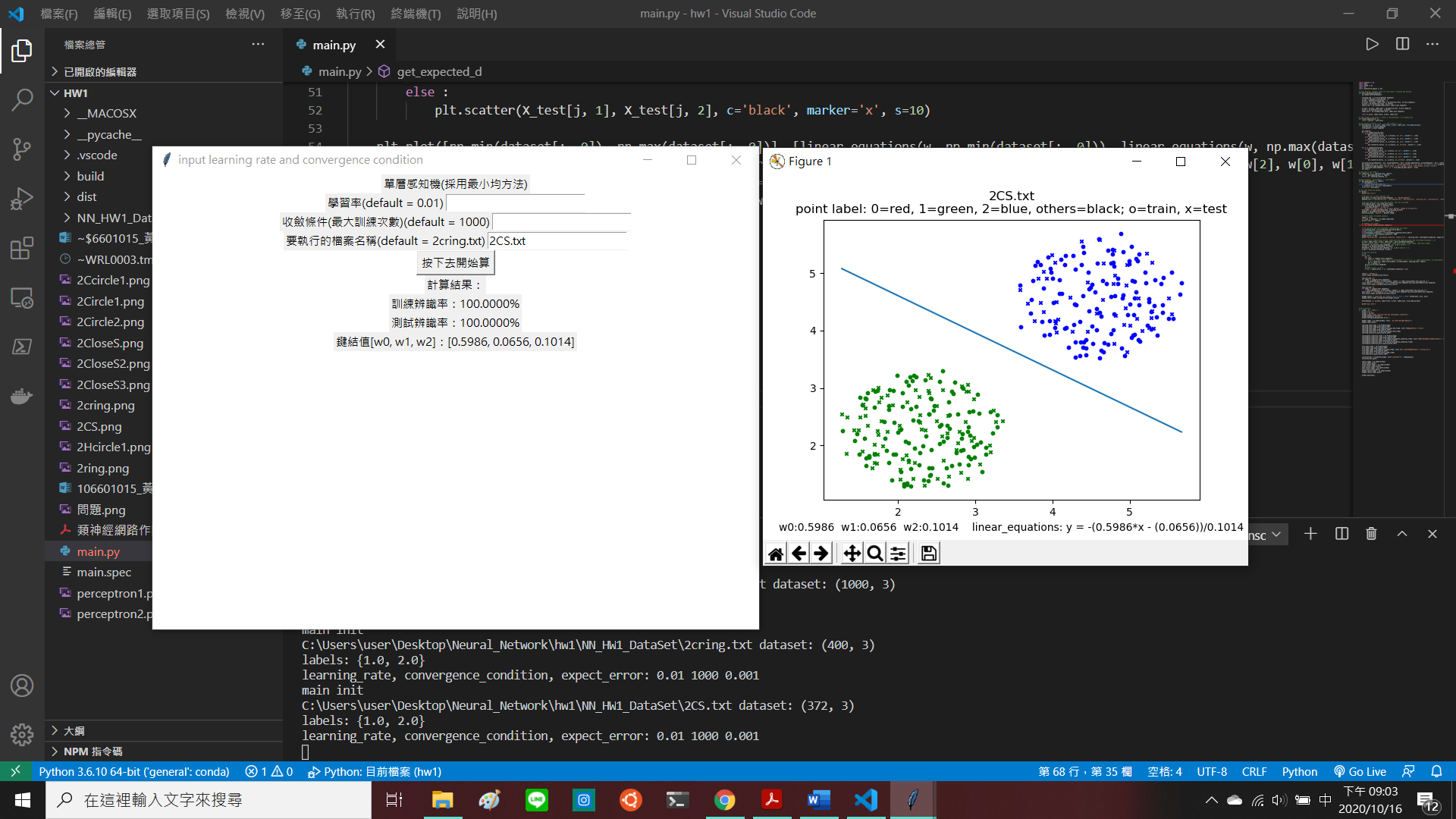


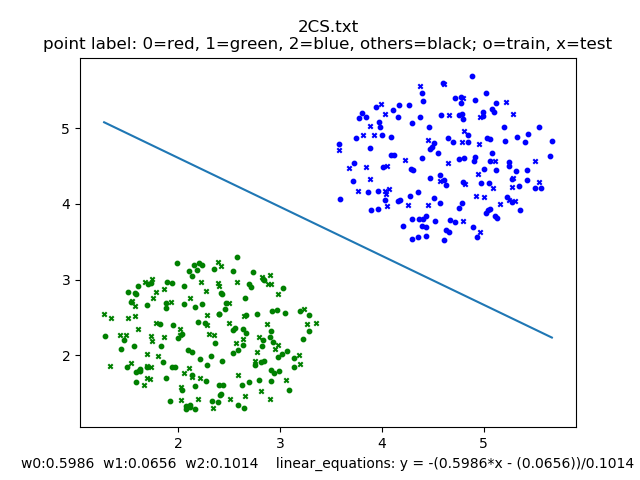
2cring.txt



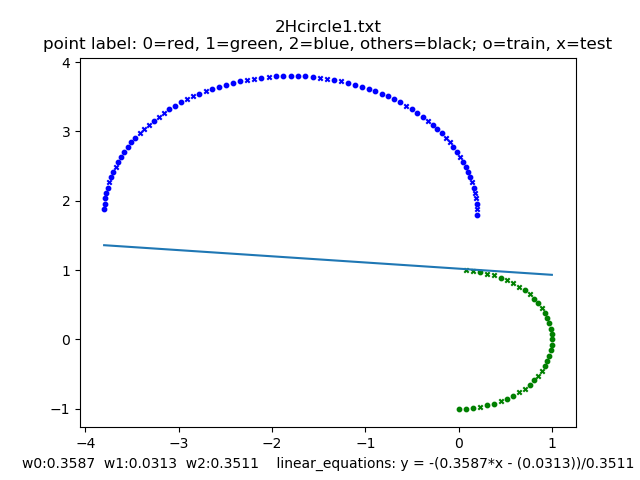
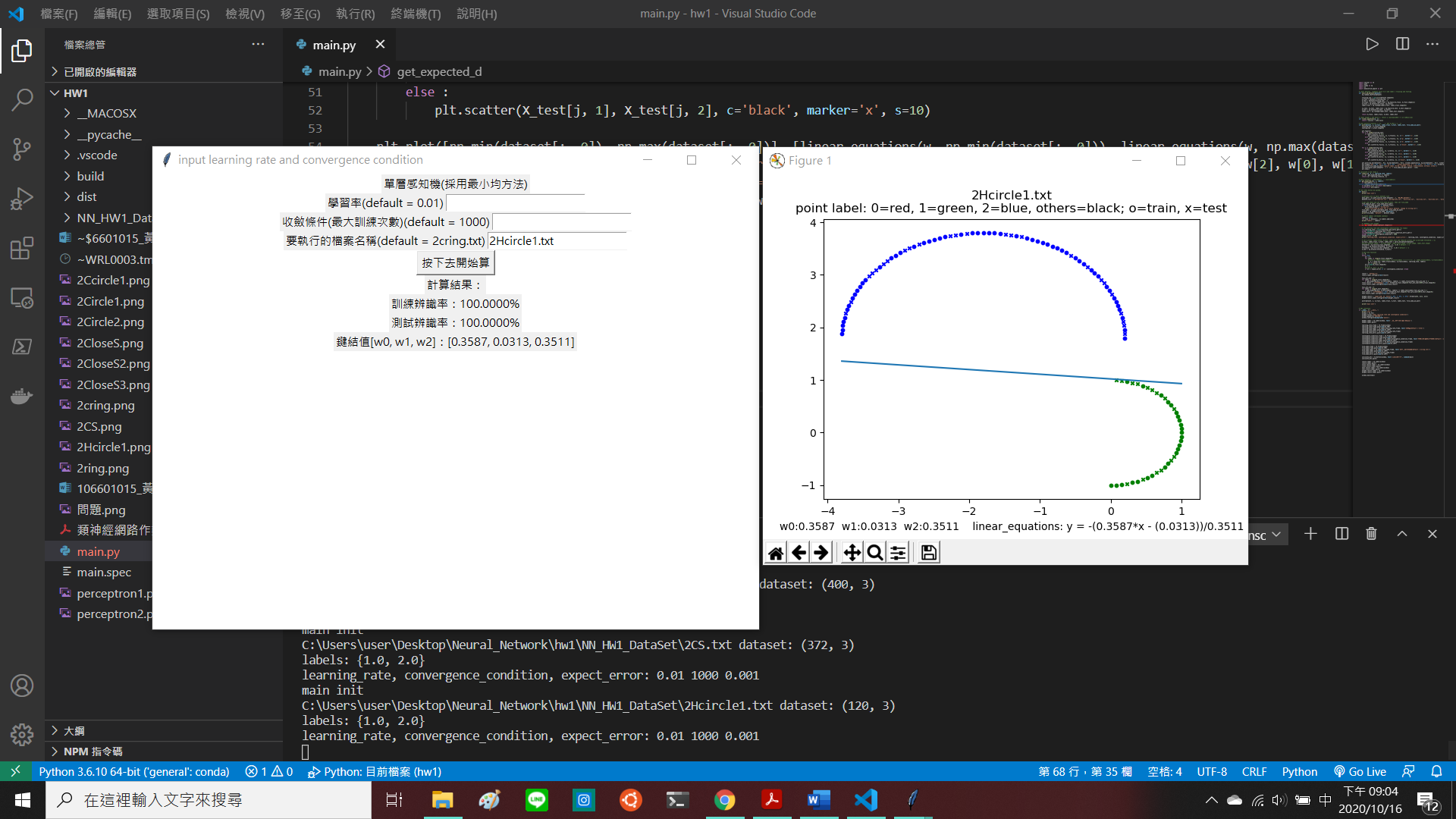


2CS.txt

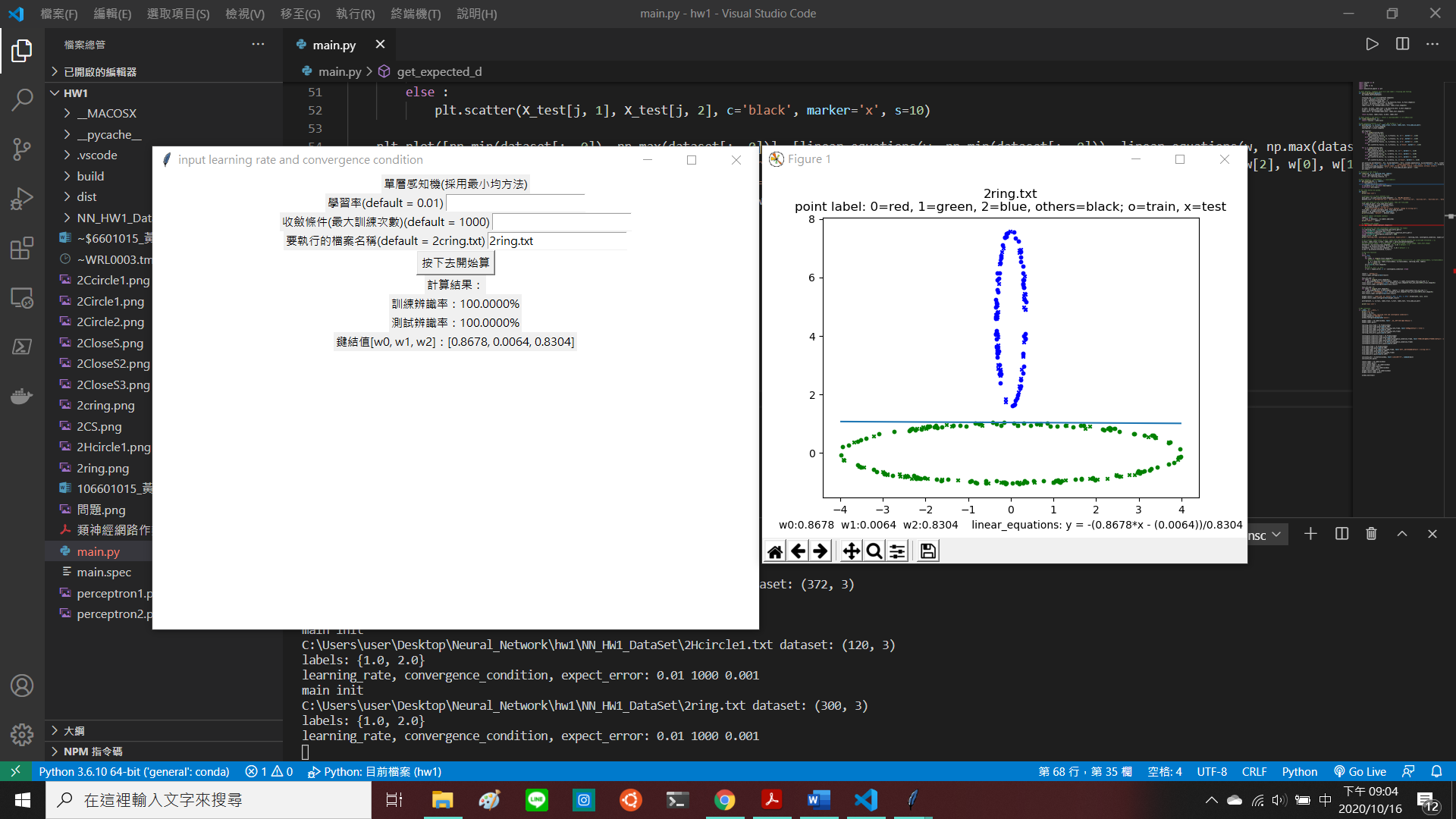


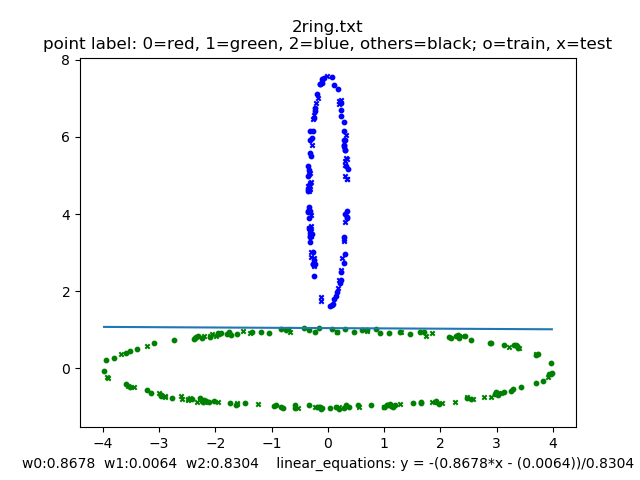


2Hcircle1.txt



2ring.txt





1. 實驗結果分析及討論。

非線性可分割資料集：perceptron2.txt、2Ccircle1.txt、2Circle1.txt、2Circle2.txt，其他均為可線性分割資料集。

如以上截圖的說明與資料呈現，鍵結值w直接的關連到線性方程的係數，因此也可以在多X維度時創建出超平面的切割方程，理論上可以解決所有可線性切割的資料集(train/test預測值趨近於100%)，並且針對不可線性切割的資料集找的到最低均方誤差mse的線性方程以分離不同的labels。

訓練次數的部分預設值為1000次，基本上都已經能達到很好的訓練效果。其實針對可線性切割的資料集只需要少量的訓練次數即可完美分割，訓練次數的提升主要用以提升不可線性切割的資料集的準確度。

學習率預設值為0.01，在1000次的訓練下為適合的尺度，並且在過程中不會有所改變，泛化資料後可考慮變動學習率，即初始學習率稍大，而後每經過固定次數的訓練即降低學習率，不過目前固定學習率成果相當良好故不考慮。

最後訓練正確率與測試正確率，線性可分割資料集均可達到兩者極高準確率，除非資料集樣本過少如perceptron2.txt才會有因為抽樣隨機2/3訓練樣本數過少而導致低準確度的情況(只有抓到一種label之類的情況)；線性不可分割資料集則在訓練正確率與測試正確率中均低於線性可分割資料集，且分割線的出現位置(即權重)不如線性可分割資料集穩定，受隨機權重initial的影響較大，但仍可透過多次訓練而產生最佳解。