

A. 程式執行說明

1.

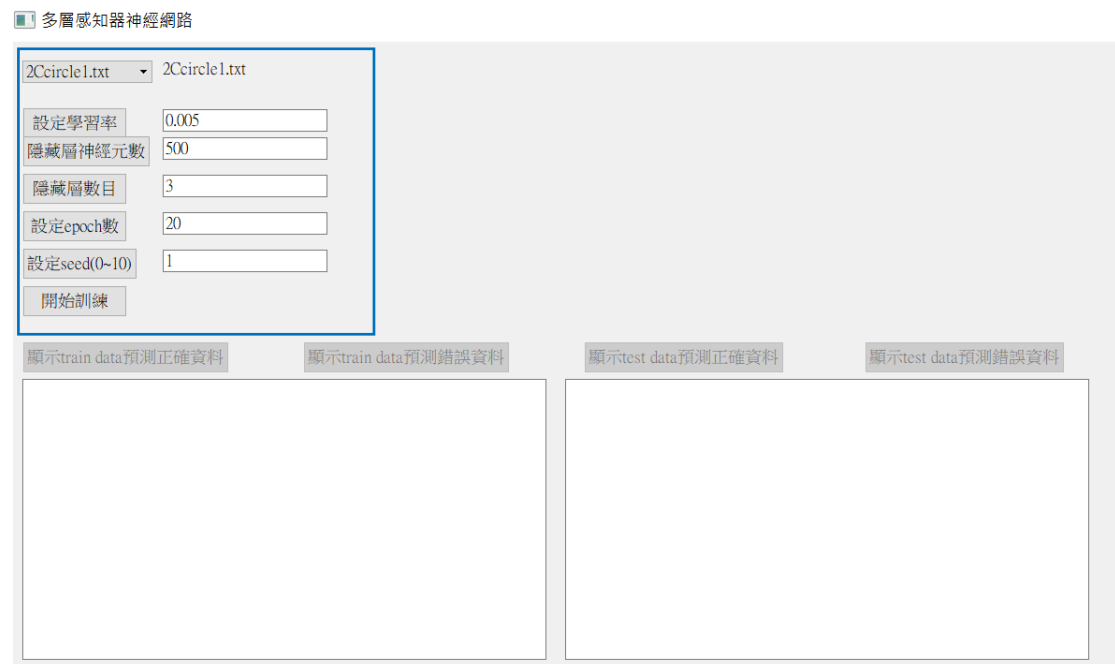
共有一個執行檔，名稱為 Hopfield_QT.exe、兩份 python 程式、兩個資料夾，名稱分別為 train 和 test，train 資料夾裡有所有助教給的訓練檔案，test 資料夾裡有所有助教給的測試檔案，執行執行檔時務必把所有檔案及資料夾放在同一個目錄下執行。

2.

執行方式為經由 cmd 到執行檔所在目錄，並輸入 `.\Hopfield_QT.exe`，執行執行檔時會花一些時間。

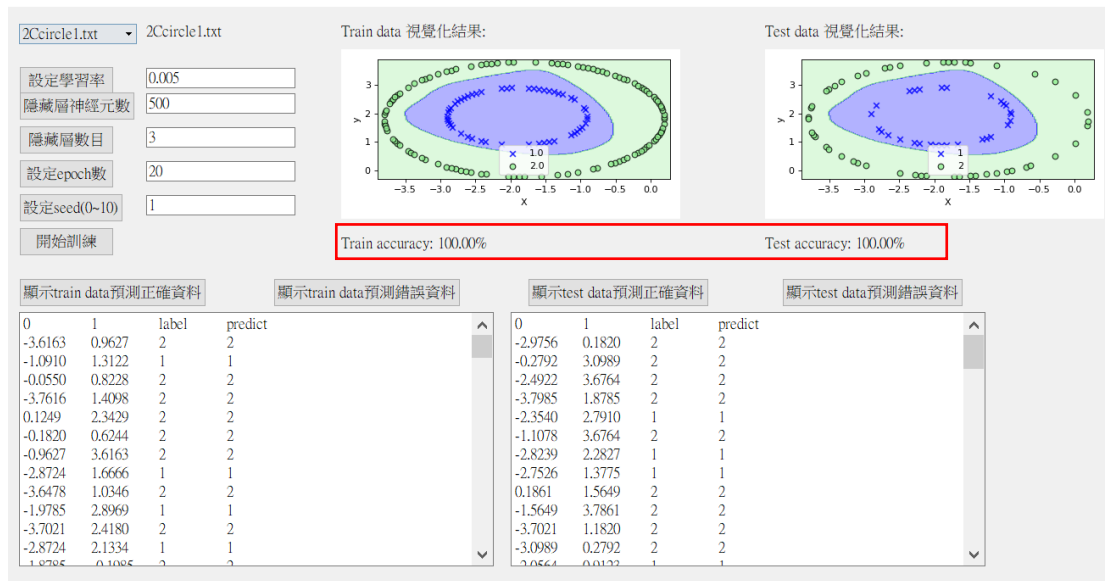
3.

執行程式時會出現如圖一的視窗圖，左上角藍色框部分是選擇要訓練的資料集、及設定神經網路學習率、隱藏層的神經元數目、隱藏層數目、epoch 數及隨機取訓練資料和測試資料的 seed。



(圖一)

設定完參數之後，按下開始訓練按鈕，會開始訓練並把結果視覺化(圖二)，紅色框會顯示訓練資料的準確率和測試資料準確率。下方紅色框的四個按鈕，是分別顯示訓練資料預測正確筆數的詳細資料、訓練資料預測錯誤筆數的詳細資料、測試資料預測正確筆數的詳細資料、測試資料預測錯誤筆數的詳細資料(圖三)。



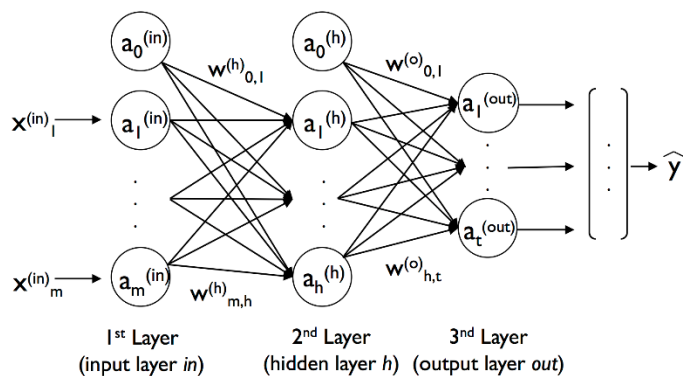
(圖二)

顯示train data預測正確資料	顯示train data預測錯誤資料	顯示test data預測正確資料	顯示test data預測錯誤資料
<div>01labelpredict</div> <div>-3.61630.962722</div> <div>-1.09101.312211</div> <div>-0.05500.822822</div> <div>-3.76161.409822</div> <div>0.12492.342922</div> <div>-0.18200.624422</div> <div>-0.96273.616322</div> <div>-2.87241.666611</div> <div>-3.64781.034622</div> <div>-1.97852.896911</div> <div>-3.70212.418022</div> <div>-2.87242.133411</div> <div>1.87850.108522</div>		<div>01labelpredict</div> <div>-2.97560.182022</div> <div>-0.27923.098922</div> <div>-2.49223.676422</div> <div>-3.79851.878522</div> <div>-2.35402.791011</div> <div>-1.10783.676422</div> <div>-2.82392.282711</div> <div>-2.75261.377511</div> <div>0.18611.564922</div> <div>-1.56493.786122</div> <div>-3.70211.182022</div> <div>-3.09890.279222</div> <div>2.05640.013211</div>	

(圖三)

B. 程式簡介

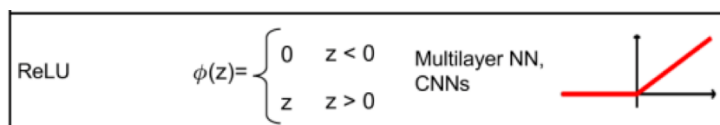
由 MLP.py 和 MLP_QT.py 兩份 python 程式組成，MLP_QT.py 是編寫 GUI 圖形介面程式，MLP.py 則是多層感知機神經網路，除了一個輸入層和一個輸出層外，使用者可以藉由圖形介面設定隱藏層數目以及隱藏層的神經元數目。設計多層感知機神經網路架構如圖四，其中包含幾項重要功能



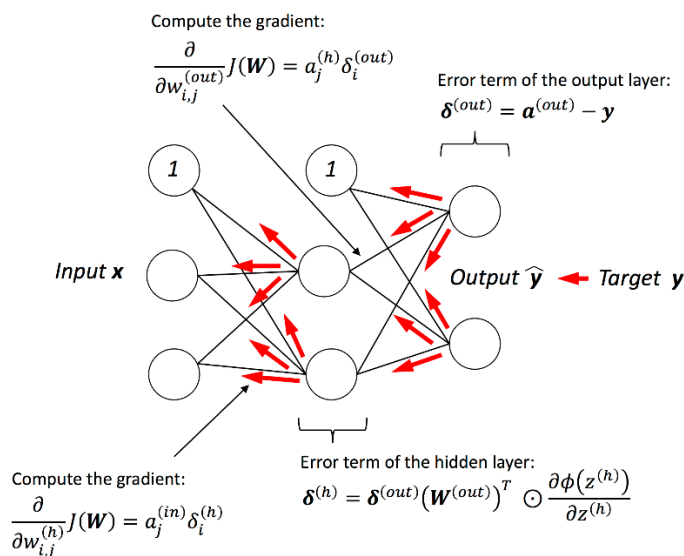
(圖四)

1. 對資料的 label 做 one-hot encoding

2. 實作前饋式神經網路
3. 使用邏輯斯成本函數(logistic loss function)計算神經網路的 loss function
4. 使用 relu 函數當作每層的啟動函數



5. 實作反傳遞法更新神經網路的 weight 值(圖五)



(圖五)

C. 實驗結果

以下是對感知器神經網路超參數設定：

設定好參數之後，把感知器神經網路對每份資料開始訓練結果如下：

1. 2CloseS.txt 檔案：

網路參數：

學習率: 0.005

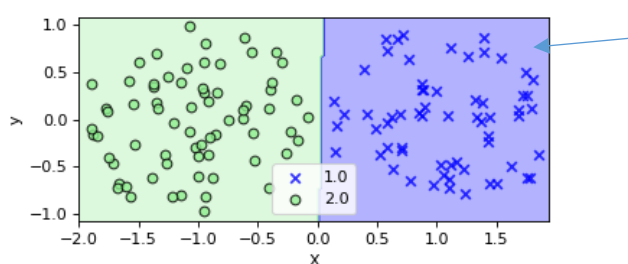
隱藏層神經元數: 50

隱藏層數: 1

Epoch 數: 20

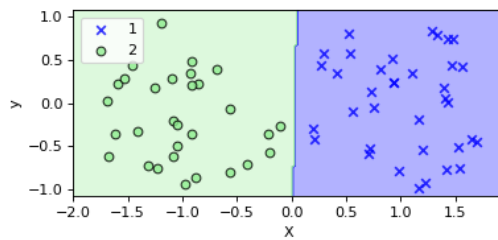
Random seed: 1

Train data 結果：



準確率: 100%

Test data 結果:



準確率: 100%

結論:視覺化的結果中，平面的顏色為神經網路對所有資料集的形成二維資料平面，對每個座標點進行預測，並對預測 label 畫上對應顏色，而平面上的標記點和顏色則是實際測試資料座標點的 label，由圖可以看出，神經網路預測是否正確(以下資料以此類推)。

而 2CloseS.txt 檔案中，神經網路可以對所有的 label 做正確分類

2. 2Ccircle1.txt

網路參數:

學習率: 0.005

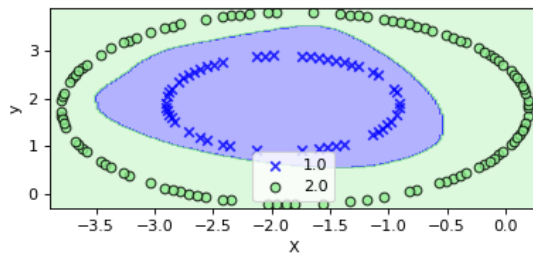
隱藏層神經元數:500

隱藏層數:3

Epoch 數:20

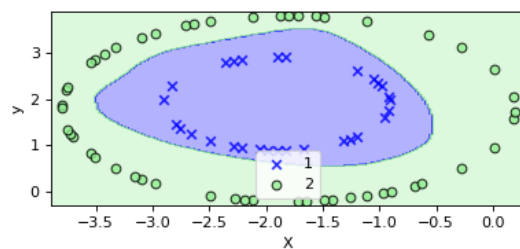
Random seed:1

Train data 結果:



準確率: 100.00%

Test data 結果:



準確率:100.00%

結論: 神經網路可以完美辨識非線性可分的資料集

3.

2Circle1.txt

網路參數:

學習率: 0.005

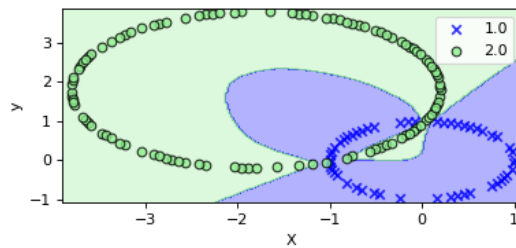
隱藏層神經元數:800

隱藏層數:3

Epoch 數:20

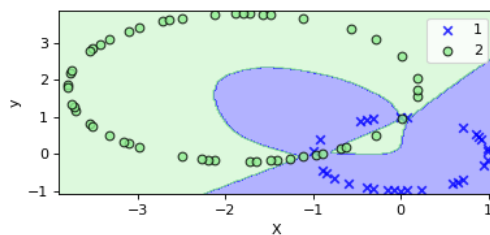
Random seed:1

Train data 結果:



準確率:98.75%

Test data 結果:



準確率: 93.75%

結論: 可以正確分類 90%以上資料

4. 2Circle2.txt

網路參數:

學習率: 0.005

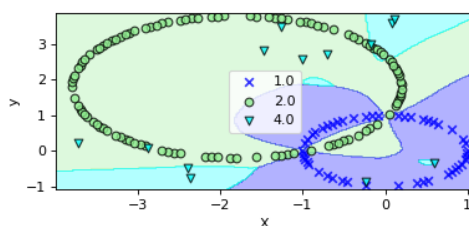
隱藏層神經元數:800

隱藏層數:4

Epoch 數:20

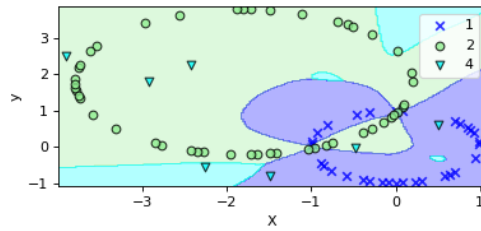
Random seed:1

Train data 結果:



準確率: 92.44%

Test data 結果:



準確率: 88.37%

結論: 神經網路可以辨識非線性可分的資料集中 85%以上的資料。

5. 2CloseS2.txt

網路參數:

學習率: 0.005

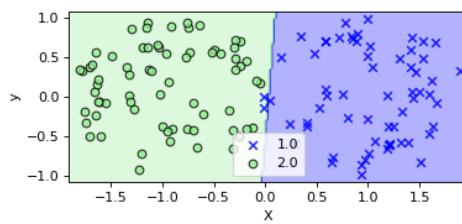
隱藏層神經元數:10

隱藏層數:1

Epoch 數:20

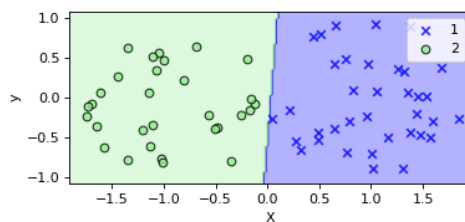
Random seed:1

Train data:



準確率: 98.48%

Test data:



準確率: 100%

結論:可以接近 100%分類資料

6. 2CloseS3.txt

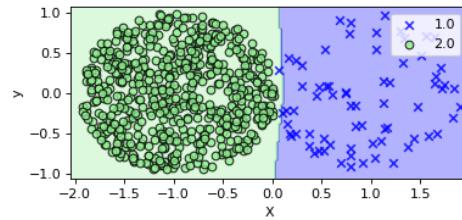
網路參數:

學習率: 0.005

隱藏層神經元數:300

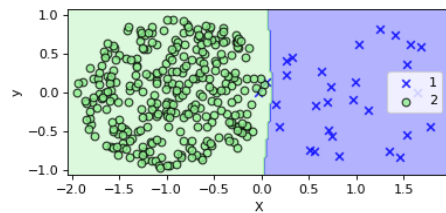
隱藏層數:2
Epoch 數:20
Random seed:1

Train data:



準確率: 99.85%

Test data:



準確率: 99.40%

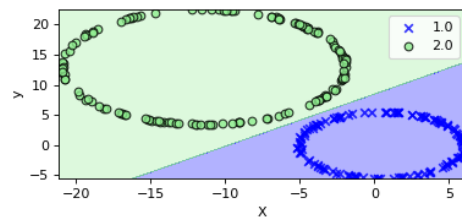
結論:可以接近 100%分類資料

7. 2cring.txt

網路參數:

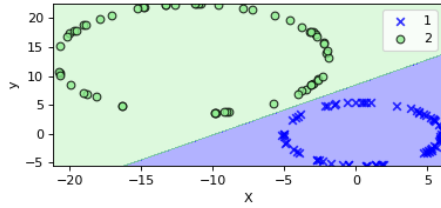
學習率: 0.005
隱藏層神經元數:10
隱藏層數:1
Epoch 數:20
Random seed:1

Train data:



準確率: 100.00%

Test data:



準確率: 100.00%

結論: 可以接近 100% 分類資料

8. 2CS.txt

網路參數:

學習率: 0.005

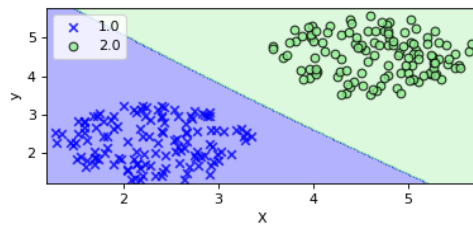
隱藏層神經元數: 10

隱藏層數: 1

Epoch 數: 20

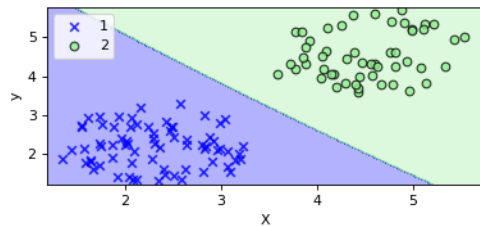
Random seed: 1

Train data:



準確率: 100.00%

Test data:



準確率: 100.00%

結論: 可以接近 100% 分類資料

9. 2Hcircle1.txt

網路參數:

學習率: 0.005

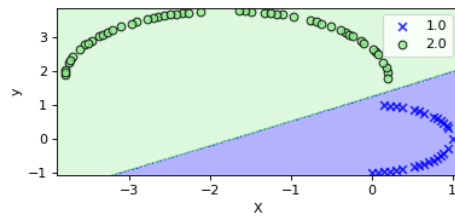
隱藏層神經元數: 10

隱藏層數: 1

Epoch 數: 20

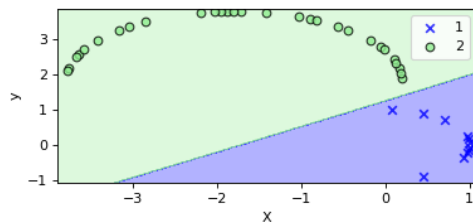
Random seed: 1

Train data:



準確率: 100.00%

Test data:



準確率: 100.00%

結論:可以接近 100%分類資料

10. 2ring.txt

網路參數:

學習率: 0.005

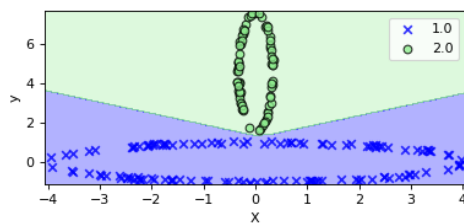
隱藏層神經元數:10

隱藏層數:1

Epoch 數:20

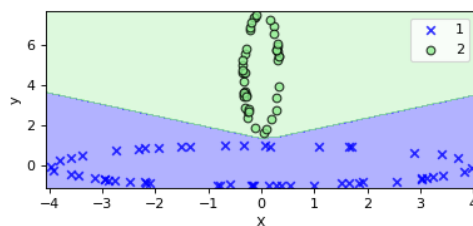
Random seed:1

Train data:



準確率: 100.00%

Test data:



準確率: 100.00%

結論: 可以接近 100% 分類資料

11. 4-satellite-6.txt

網路參數:

學習率: 0.005

隱藏層神經元數: 10

隱藏層數: 1

Epoch 數: 20

Random seed: 1

Train data: 因為維度資料超過二維, 很抱歉無法視覺化呈現 train data!

準確率: 27.80%

Test data: 因為維度資料超過二維, 很抱歉無法視覺化呈現 test data!

準確率: 27.37%

結論: 分類結果不好

12. 5CloseS1.txt

網路參數:

學習率: 0.005

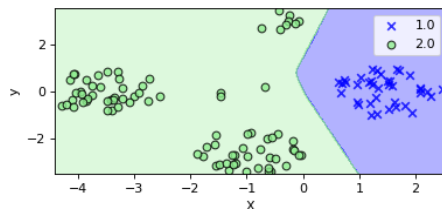
隱藏層神經元數: 20

隱藏層數: 1

Epoch 數: 20

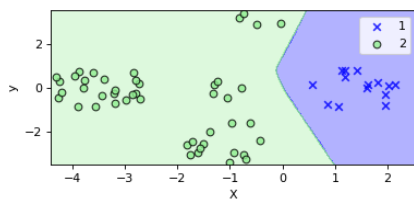
Random seed: 1

Train data:



準確率: 100.00%

Test data:



準確率: 100.00%

結論: 可以接近 100% 分類資料

13. 8XOR.txt

網路參數:

學習率: 0.005
隱藏層神經元數:100
隱藏層數:3
Epoch 數:20
Random seed:1

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 100.00%

Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 60.00%

結論:可以分類 60%筆測試資料

14. C10D.txt

網路參數:

學習率: 0.005
隱藏層神經元數:20
隱藏層數:1
Epoch 數:20
Random seed:1

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 100.00%

Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 100.00%

結論:可以分類 100%筆測試資料

15. C3D.txt

網路參數:

學習率: 0.005
隱藏層神經元數:20
隱藏層數:1
Epoch 數:20
Random seed:1

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 100.00%

Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 100.00%

結論:可以分類 100%筆測試資料

16. IRIS.TXT

網路參數:

學習率: 0.005
隱藏層神經元數:80

隱藏層數:4

Epoch 數:20

Random seed:1

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 69.00%

Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 68.00%

結論:可以分類 60%筆測試資料

17. Number.txt

網路參數:

學習率: 0.005

隱藏層神經元數:20

隱藏層數:1

Epoch 數:20

Random seed:1

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 33.33%

Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 0.00%

結論: 資料筆數太少, 導致訓練 overfitting

18. Perceptron1.txt

網路參數:

學習率: 0.005

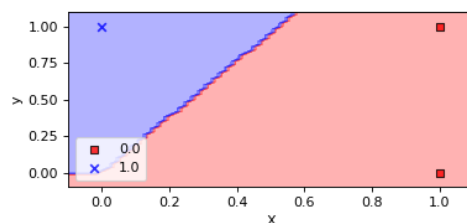
隱藏層神經元數:10

隱藏層數:3

Epoch 數:10

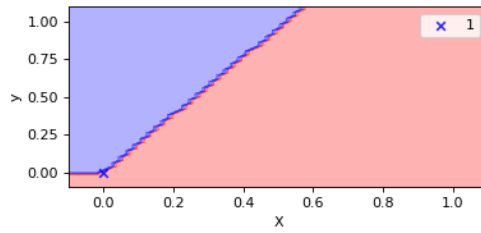
Random seed:6

Train data:



準確率: 100.00%

Test data:



準確率: 0.00%

結論: 資料筆數太少，導致神經網路訓練 overfitting

19. Perceptron2.txt

網路參數:

學習率: 0.005

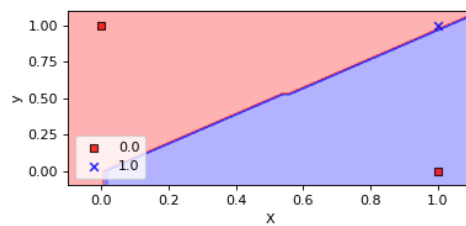
隱藏層神經元數:10

隱藏層數:3

Epoch 數:10

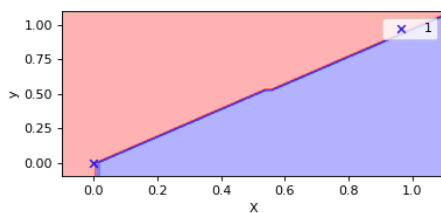
Random seed:6

Train data:



準確率: 33.33%

Test data:



準確率: 0.00%

結論: 資料筆數太少，導致神經網路訓練 overfitting

20. Perceptron3.txt

網路參數:

學習率: 0.005

隱藏層神經元數:10

隱藏層數:3

Epoch 數:10

Random seed:6

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 66.7%

Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 0.00%

結論: 資料筆數太少, 導致訓練 overfitting

21. Perceptron4.txt

網路參數:

學習率: 0.005

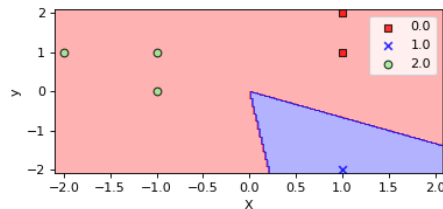
隱藏層神經元數:500

隱藏層數:3

Epoch 數:10

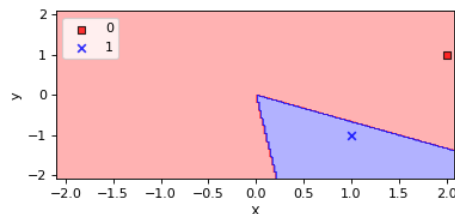
Random seed:6

Train data:



準確率: 33.33%

Test data:



準確率: 50.00%

結論: 資料筆數太少, 導致神經網路訓練 overfitting

22. wine.txt

網路參數:

學習率: 0.005

隱藏層神經元數:15

隱藏層數:10

Epoch 數:10

Random seed:6

Train data: 因為維度資料超過二維,很抱歉無法視覺化呈現 train data!

準確率: 35.59%

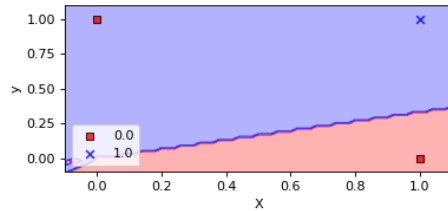
Test data: 因為維度資料超過二維,很抱歉無法視覺化呈現 test data!

準確率: 27.12%

結論: 訓練結果不好

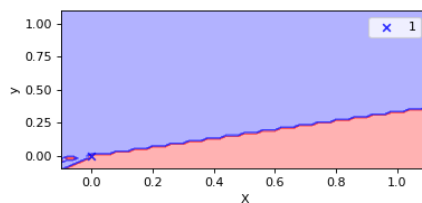
23. xor.txt

Train data:



準確率: 66.67%

Test data:



準確率: 0.00%

結論: 資料筆數太少，導致神經網路訓練 overfitting

D. 實驗結果分析與討論

由上述實驗可以發現，當訓練的資料是不是線性可分割平面時，自己設計的神經網路可以藉由增加隱藏層數目來增加準確率。但相同問題是如果資料筆數太少，訓練時容易 overfitting，除此之外大部分線性可分割資料及多 label 資料都可以準確分辨 80% 資料。

E. 預計可以完成的加分項目

- A. 能夠處理多維資料（三維以上）
- B. 能夠處理多群資料（三群以上）
- C. 隱藏層層數可設定
- D. 隱藏層的神經元個數可設定