

類神經網路作業三

學號：111526009 姓名：薛竣祐

一. 程式執行說明

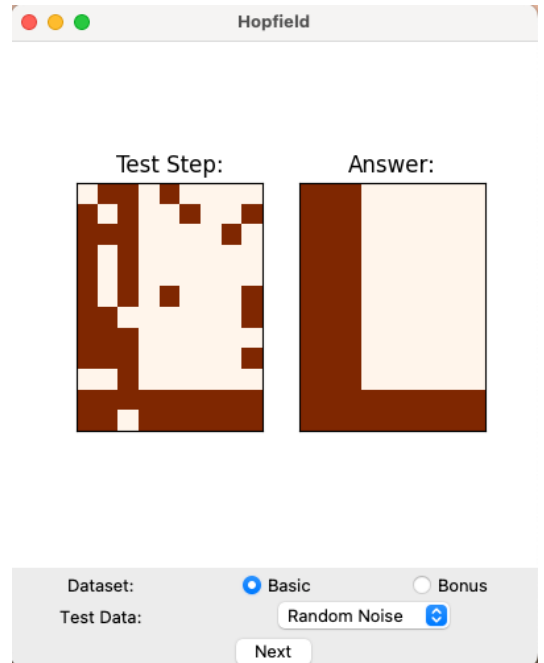
程式提供兩執行檔，一為exe執行檔，一為Unix執行檔，兩執行檔結果相同，擇一執行。請先確認Hopfield_dataset資料夾與執行檔在同一執行目錄下，且資料夾內包含4個Basic/Bonus的Training/Testing資料。執行後會呈現此畫面，可為Model設定Dataset的選擇並設定要測試的資料。

Dataset：可選擇Basic或Bonus。

Test Data：可選擇隨機的noise data或是來自預設測試集的資料。

完成設定後，請點擊Next按鈕，等待執行完畢後會在Test Step圖中更新一次經過Hopfield過的結果，可重複執行來呈現Hopfield逐漸聯想的過程，並可以重新選擇Dataset或Test Data來重置結果。

如果重複執行後，Test Step畫面不再更新，即代表已聯想完成，無法再進行下一步的聯想。



二. 程式碼簡介

Source Code為Python，分為兩個檔案。

main.py：程式進入點及介面的呈現。

Hopfield.py：Hopfield模型，包含訓練及預測的函式，以及示範執行用的程式。

主要使用到的套件有：numpy、matplotlib、tkinter。

Hopfield檔案中定義了Hopfield模型的各個功能，初始化時須事先設定資料的長寬。在輸入資料時都是將二維資料攤平為一維的方式來計算的，這樣在設計及計算權重時較簡易，以Basic的輸入來說攤平後就是1*108的資料。

另外有關鍵的三個func：

1. `make_weights()` 將逐格計算模型的weight，最終將算出一個2維陣列，大小將會是攤平資料大小的平方，以Basic的輸入來說就會是108*108的二維陣列大小。計算方式是`weight[i][j]`等於每筆訓練資料的第i格乘上每筆訓練資料的第j格的總和，這裡的計算方式稍微簡化自老師上課教的公式。
2. `sgn()` 可以對計算結果執行類似Signum function。即，大於0則為1，小於0則為-1，等於0時較特別，需設定為前一個Step的同index資料，因為初始Step已將資料整理為1或-1，所以選擇前一個step時不須再次執行`sgn()`。
3. `step()` 則會計算下一步的結果，計算方式只需將weight乘上輸入的倒矩陣即可。

三. 實驗結果

對於Noisy Data都是一律使用0.25的機率來加入雜訊。

以下為實驗結果，左右兩張圖為一組，左圖為初始狀態，右圖為結束狀態：



Hopfield


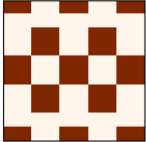
Hopfield

Hopfield

Hopfield


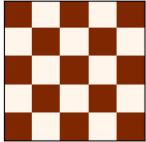
Test Step:

Answer:



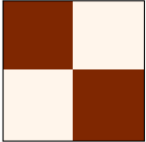

Test Step:

Answer:



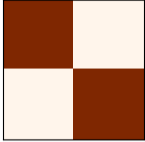
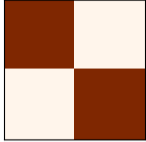
Test Step:

Answer:



Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

Class 2

Next

Dataset:

Test Data:

☐ Basic

☒ Bonus

Class 2

Next

Dataset:

Test Data:

☐ Basic

☒ Bonus

Class 3

Next

Dataset:

Test Data:

☐ Basic

☒ Bonus

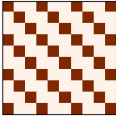
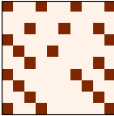
Class 3

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

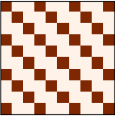
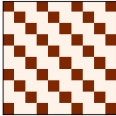
Class 4

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

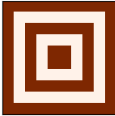
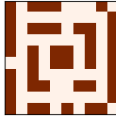
Class 4

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus


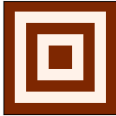
Class 5

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

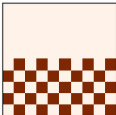
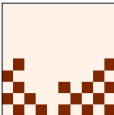
Class 5

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

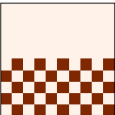
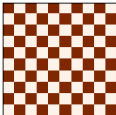
Class 6

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

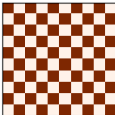
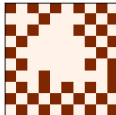
Class 6

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

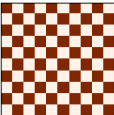
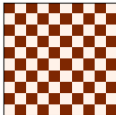
Class 7

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

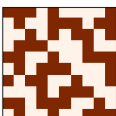
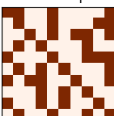
Class 7

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

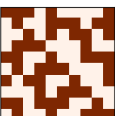
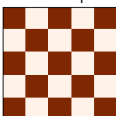
Class 8

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

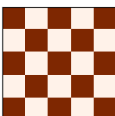
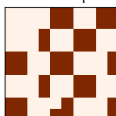
Class 8

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

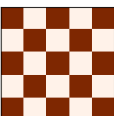
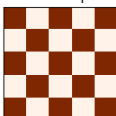
Class 9

Next

Hopfield

Test Step:

Answer:



Dataset:

Test Data:

☐ Basic

☒ Bonus

Class 9

Next



四. 實驗結果分析與討論

可以透過實驗結果的圖示得知，大部分測試資料都能夠得到正確的資料，尤其是Basic資料。在不包含Noise的資料下，Basic在測試資料可以得到100%的正確率，而Bonus可以得到50%的正確率。但如果包括Noise資料去計算的話Basic可以獲得90%~100%的正確率，而Bonus可以得到70%~80%的正確率。

我認為造成聯想錯誤的原因很大一部份是某個聯想資料完全(或絕大部分)被包含於另一個聯想資料，這樣就很容易使模型去聯想+1數值較多的那個聯想資料，或是也有兩張聯想資料被結合的狀況。我覺得這算是Hopfield的特點，它絕對會收斂於某個結果，但不一定會聯想正確。