

HW3-實作 Hopfield

資工三 A 108502003 林緯翔

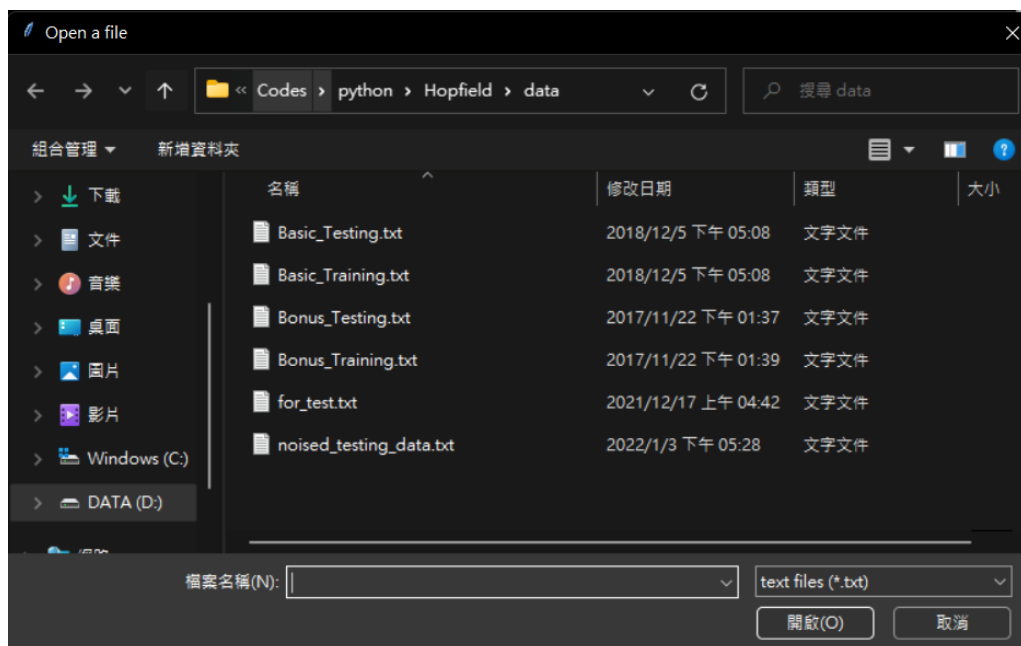
- 加分項目
 - Bonus 的 testing data 能部分回想(10/15)
 - 可以自動生成有噪音的檔案
 - 程式執行說明。(如何操作、使用):
- 開啟 gui.exe，開啟後 GUI 長這個樣子



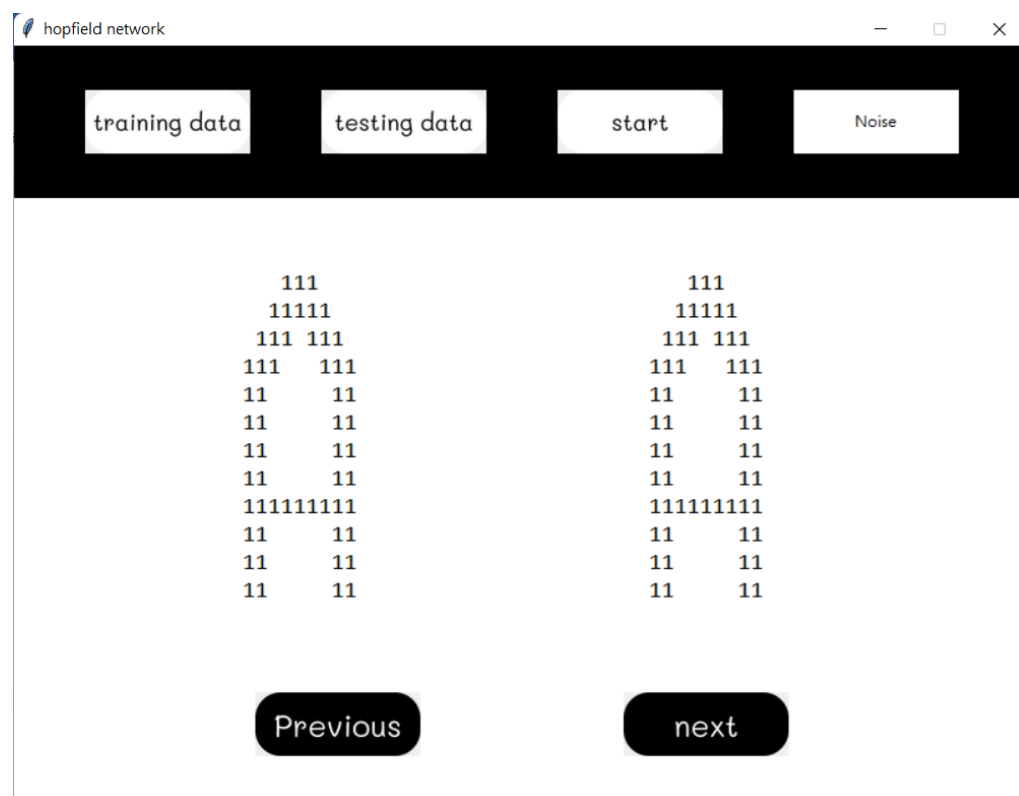
Previous

next

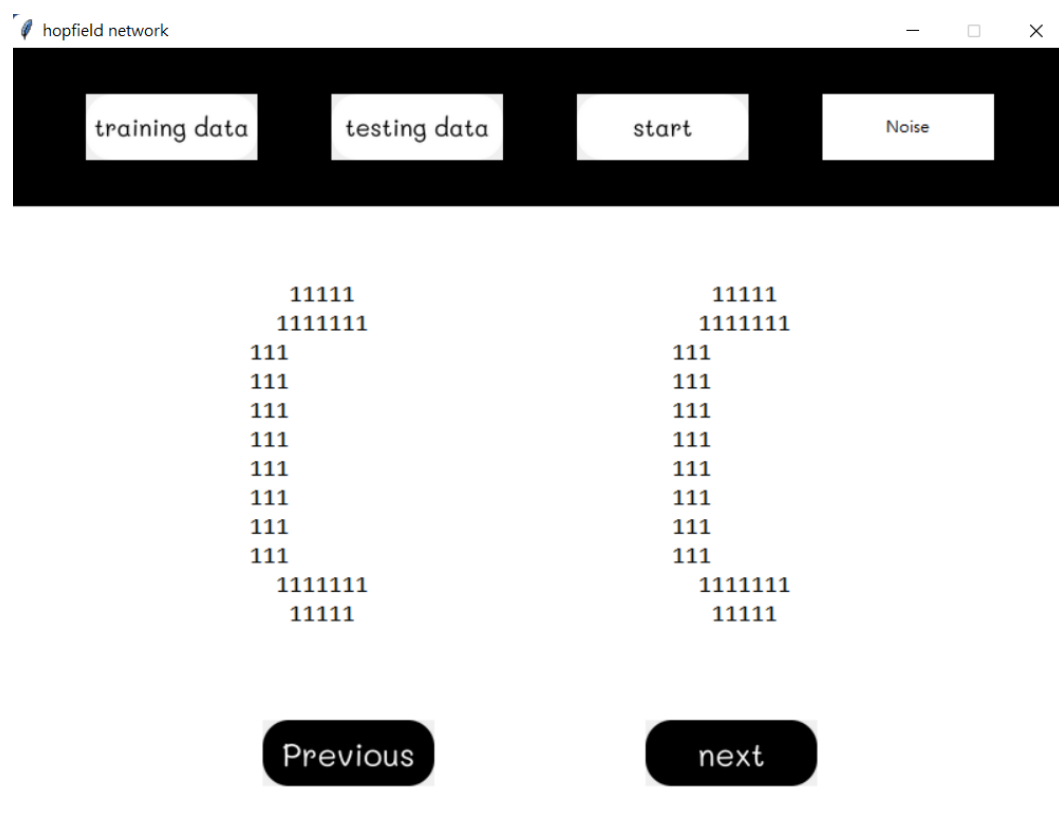
按下 training data 按鈕可以選擇要用的 training data，預設會開啟 ./data 資料夾



按下 **testing data** 按鈕可以選擇要用的 **testing data**
按下 **start** 後，可在下方看到回想結果



左方顯示的是應該要回想出的圖形，右方則是網路回想的結果
按下 **previous**、**next** 可以看前一個或下一個回想結果



- 自行將訓練資料集加入雜訊

如果想要將訓練資料集加入雜訊，可以先按下 **testing data** 按鈕選取一個 **testing data**，選完 **testing data** 後按下 **noise** 按鈕會在 `./data` 資料夾中生成 `noised_testing_data.txt`，之後就可以用這個檔案當作 **testing data** 囉!

- 程式簡介、須包含實作架構(Hopfield)

有 `gui.py` 和 `model.py` 兩個 `.py` 檔，以下為相關介紹

Gui.py: GUI 相關程式碼

✚ `add_noise`: 生成有噪音的測試資料

Model.py:

✚ `load_data`: 讀取 `.txt` 後回傳一個 `ndarray`，提供網路做預測

✚ `get_model`: 根據 **training data** 生成模型(包含權重和 `theat`，但後來 `theta` 並沒有使用)並回傳

✚ `predict`: 給一筆 **training data**，輸入模型後回傳預測結果(採用非同步的更新)

`load_data_1`、`load_data_2`、`predict_2`、`predict_3` 其實都沒有用到

- 實驗結果(所有資料集都須有實驗結果集說明)。

基本題: 三個都回想正確，自行加入雜訊後還是回想正確

111	111	11111	11111
11111	11111	1111111	1111111
111 111	111 111	111	111
111 111	111 111	111	111
11 11	11 11	111	111
11 11	11 11	111	111
11 11	11 11	111	111
11 11	11 11	111	111
111111111	111111111	111	111
11 11	11 11	111	111
11 11	11 11	111	111
11 11	11 11	1111111	1111111
		11111	11111
111	111		
111	111		
111	111		
111	111		
111	111		
111	111		
111	111		
111	111		
111	111		
111111111	111111111		
111111111	111111111		

加分題: 15 個中有 10 個回想正確。自行加入雜訊後回想結果一樣

1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

[illegible]

```

1 1 1 1
1 1 1
1 1 1
1 1 1 1
1 1 1
1 1 1
1 1 1 1
1 1 1
1 1 1
1 1 1 1

```

```

1 1 1 1
1 1 1
  1 1 1
1 1 1 1
  1 1 1
    1 1 1
1 1 1 1
  1 1 1
    1 1 1
1 1 1 1

```

```

1111111111
1          1
1 111111 1
1 1      1 1
1 1 11 1 1
1 1 11 1 1
1 1      1 1
1 111111 1
1          1
1111111111

```

```

1111111111
1          1
1 111111 1
1 1      1 1
1 1 11 1 1
1 1 11 1 1
1 1      1 1
1 111111 1
1          1
1111111111

```

1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1

[illegible][illegible][illegible]

```

111 1      1
 1  111 11
   1 1 11 1
    111    1
11  1  111
 1 111    1
1 11  1  1
   1 111
11 111  1
 1  1  111

```

```

11  11  11
11  11  11
    11  11
    11  11
11  11  11
11  11  11
    11  11
    11  11
11  11  11
11  11  11

```

11 11 11
 11 11 11
 11 11
 11 11
 11 11 11
 11 11 11
 11 11
 11 11
 11 11 11
 11 11 11

```

11  11  11
11  11  11
   11  11
   11  11
11  11  11
11  11  11
   11  11
   11  11
11  11  11
11  11  11

```

```

11111
1   1 111
1   1 111
1   1 111
11111
          11111
    111 1   1
    111 1   1
    111 1   1
          11111

```

```

11111
11111  1
111 1   1
11 11
11111
          11111
          11 11
1       1 111
  1     11111
        11111

```

```
11111
11111
11111
11111
11111
      11111
      11111
      11111
      11111
      11111
```

```
11111
11111
11111
11111
11111
      11111
      11111
      11111
      11111
      11111
```

```

1  1111  1
11 1111
111 1111
1111 1111
  1111 111
    1111 11
1  1111  1
11 1111
111 1111
1111 1111

```

```

1 1 1 1
1 1 1
1 1 1
1 1 1 1
1 1 1
1 1 1
1 1 1 1
1 1 1
1 1 1
1 1 1 1

```

```

1 1 1 1
 1 1 1
   1 1 1
1 1 1 1
 1 1 1
   1 1 1
1 1 1 1
 1 1 1
   1 1 1
1 1 1 1

```

```

1 1 1 1 111111111
 1 1 1 1 1      1
   1 1 1 1 1      1
1 1 1 1 1 1 1 1 1
 1 1 1 1 1 11 1
   1 1 1 1 1 11 1
1 1 1 1 1 1 1 1 1
 1 1 1 1 1      1
   1 1 1 1 1      1
1 1 1 1 1 111111111

```

```

1111111111
1      1
 1 11111 1
1 1      1 1
1 1 11 1 1
1 1 11 1 1
1 1 11 1 1
1 1      1 1
1 11111 1
1      1
1111111111

```

```

1111111111
1      1
1 11111 1
1 1      1 1
1 1 11 1 1
1 1 11 1 1
1 1      1 1
1 11111 1
1      1
1111111111

```

```

1111111111
1      1
1 11111 1
1 1      1 1
1 1 11 1 1
1 1 11 1 1
1 1      1 1
1 11111 1
1      1
1111111111

```

• 實驗結果分析及討論

一開始我有使用 **theta**，結果在 **basic dataset** 中，所有回想結果都正確，但在 **bonus** 中的回想結果非常的爛。所以後來我把 **theta** 移除，結果 **bonus** 就可以回想正確 10 個。

為了讓剩餘的五個能夠回想正確，我嘗試將網路的神經元變多來增加網路的記憶量，包含直接讓輸入的向量變兩倍長(原本:1100 變 11001100)或讓圖形變寬(原本:1010 變 11001100)，但回想效果都不佳，所以我認為 **bonus** 的回想錯誤並不是要回想的圖形種類太多的問題，而是有些圖形就很難被回想起來(可能訓練資料中有其他圖形跟他太像了)