HW4 P76074046 方鈞麒

環境:

作業系統: Windows 10

程式語言: Python 2.7

套件: numpy、matplotlib

**第一題**

執行方式: python hw4\_1.py

參數:

Iterative: 500

Batch size: 64

Learning rate: 0.001

程式說明:

1. 每次隨機選64個資料做訓練
2. 先做Forward propagation，記住error值和各neural layer輸出結果
3. 接著做Back propagation，計算出權重和偏差的微分值，最後更新權重和偏差
4. 重複步驟1~3 500次
5. 匯入測資並計算accuracy 和 loss value

結果圖:

兩種Neural network測試結果，準確率皆為0.1135



Fig. 1: Accuracy result

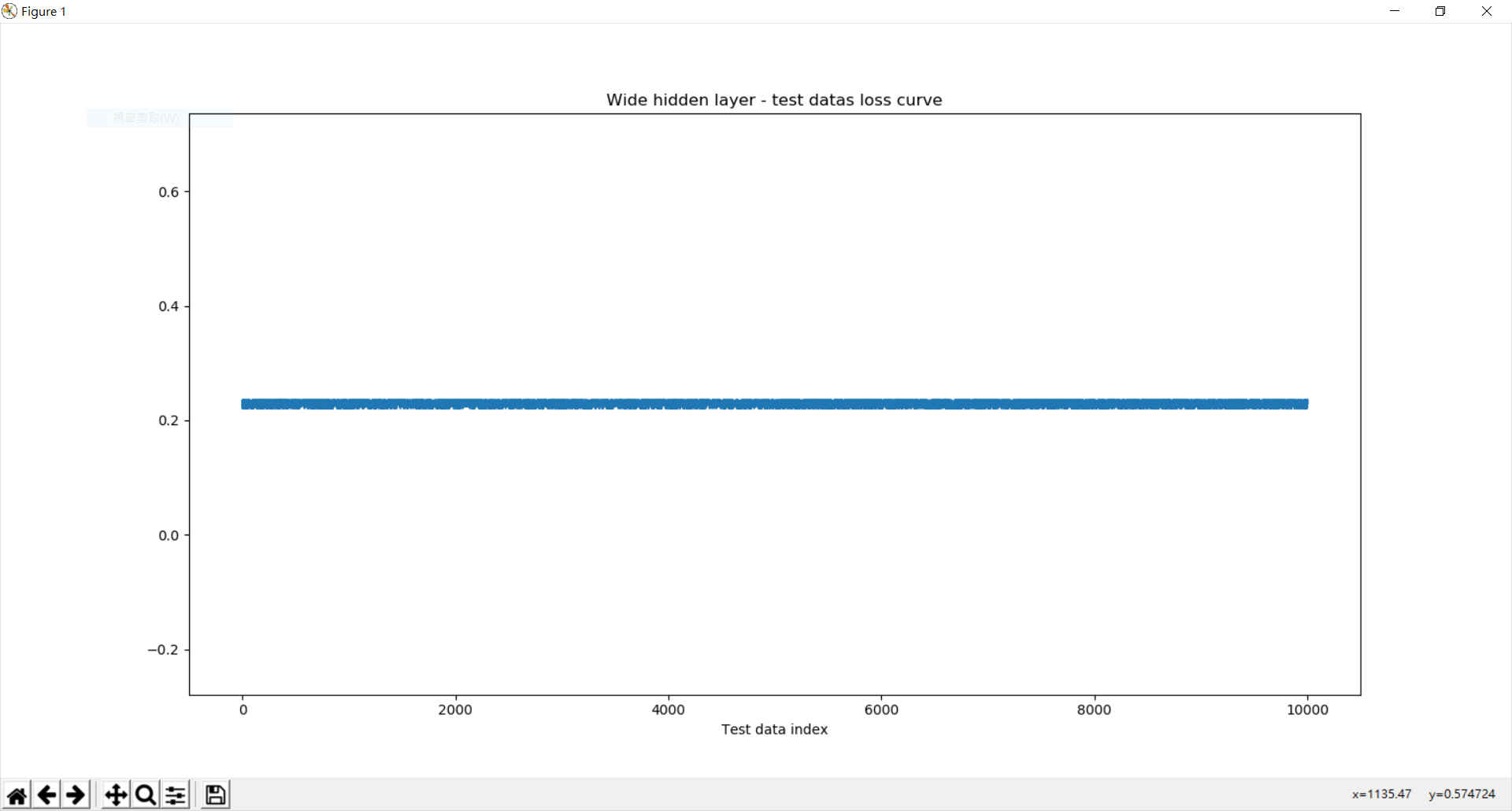


Fig. 2: Loss curve of wide hidden layer

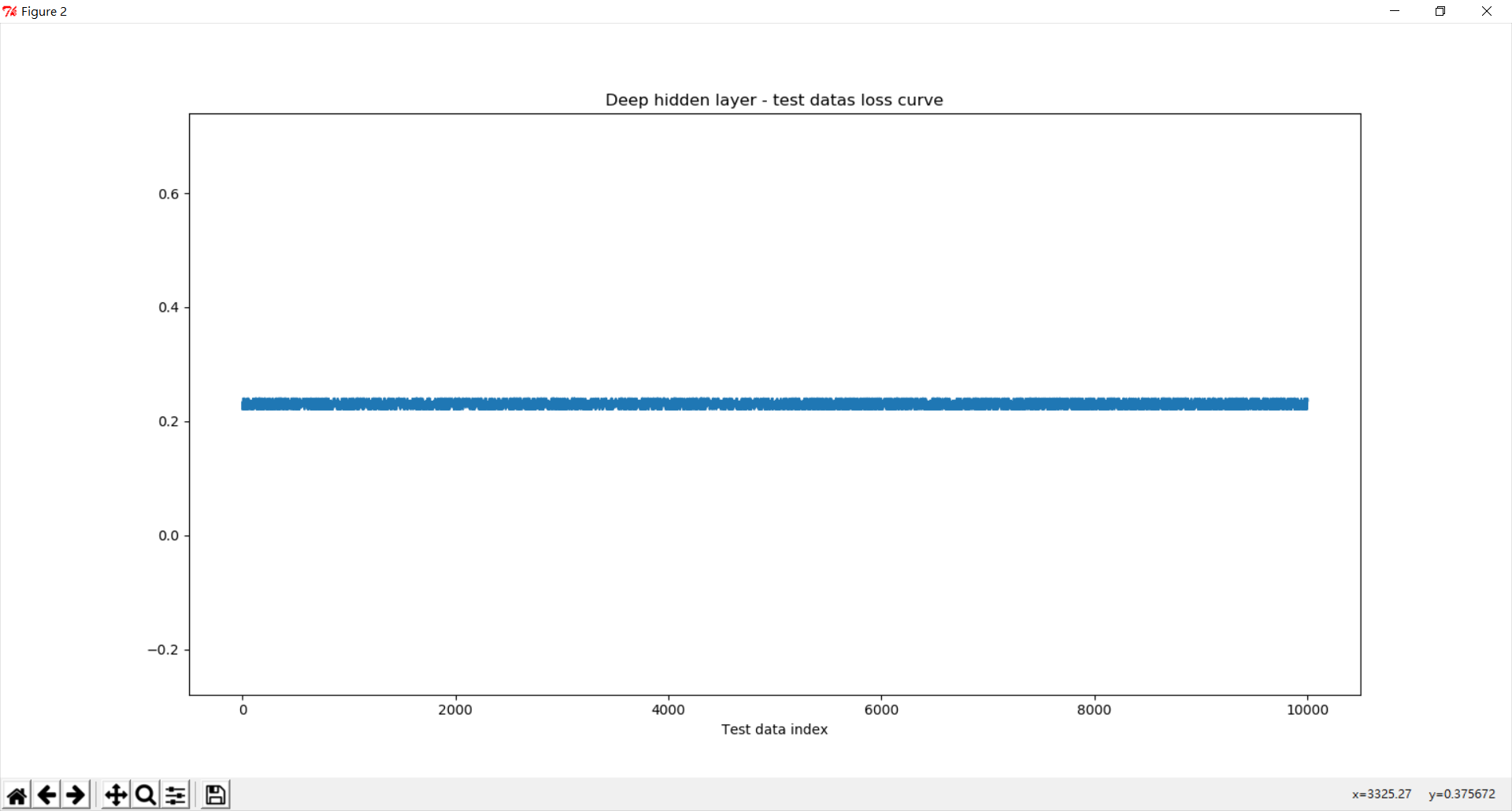


Fig. 3: Loss curve of deep hidden layer

**第二題**

執行方式: python hw4\_2.py

參數:

Iterative: 2000

Batch size: 64

Learning rate: 0.0001

程式說明:

1. 每次隨機選64個資料做訓練
2. 先做Forward propagation，記住error值和各neural layer輸出結果
3. 接著做Back propagation，計算出權重和偏差的微分值，最後更新權重和偏差
4. 重複步驟1~3 2000次
5. 匯入測資，將輸出結果降維至2維並繪圖
6. 將原圖和重建出的圖畫出來
7. 畫出filter (第一層hidden layer中的8個神經元的權重)

結果圖:

下圖是將原圖降維至128維後取前2維值畫出的，可以看到越偏右上的數字2筆畫也越粗。

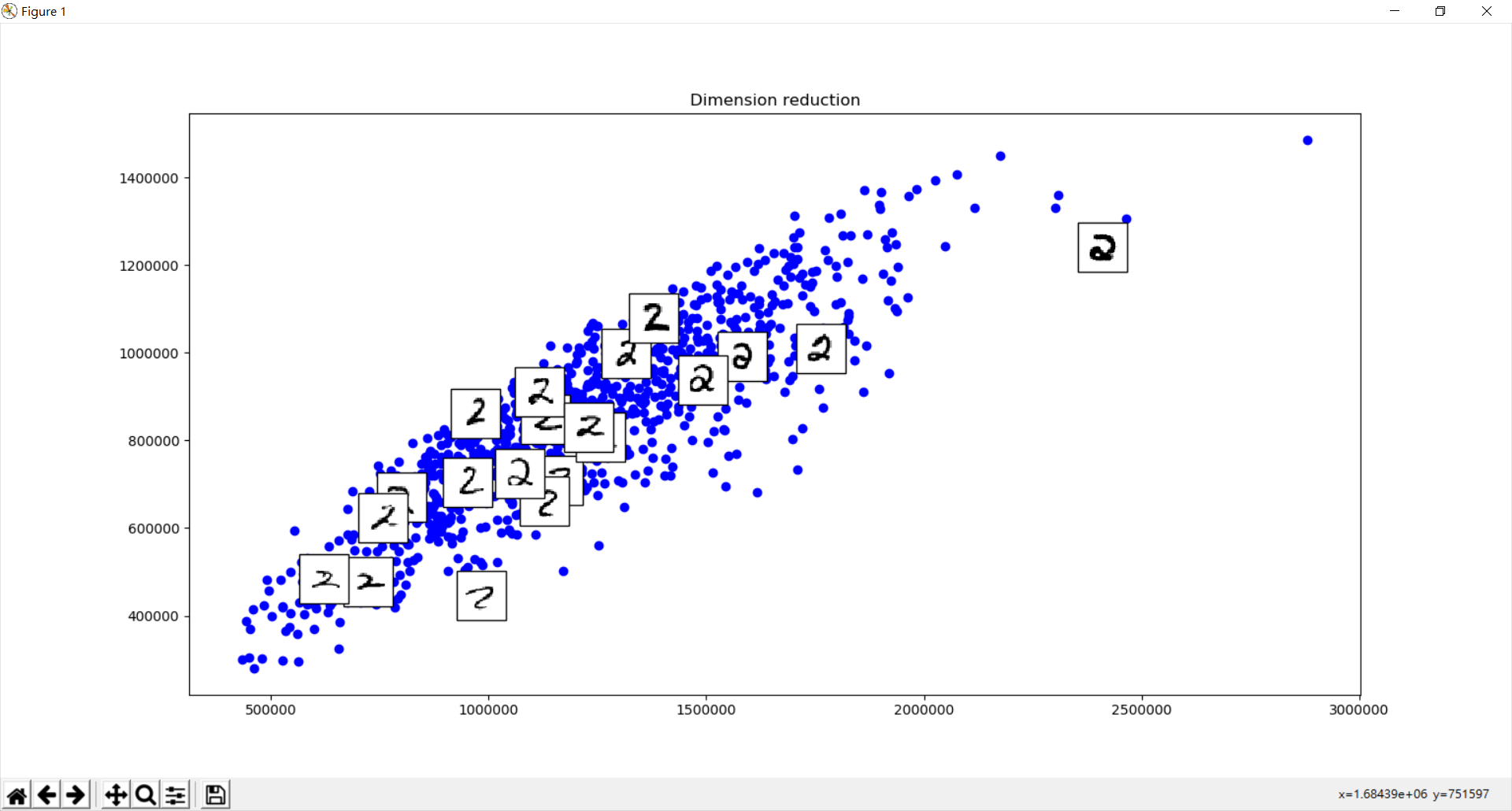


Fig. 4: 2-Dimension reduction

下圖同一列中兩兩圖片一組，左邊的是原圖影像，右邊的是重建出的圖像。重建圖依稀可以看出原圖的輪廓，但存在有躁點和圖案不連續的情形，這部分或許能透過套用某些演算法來優化。

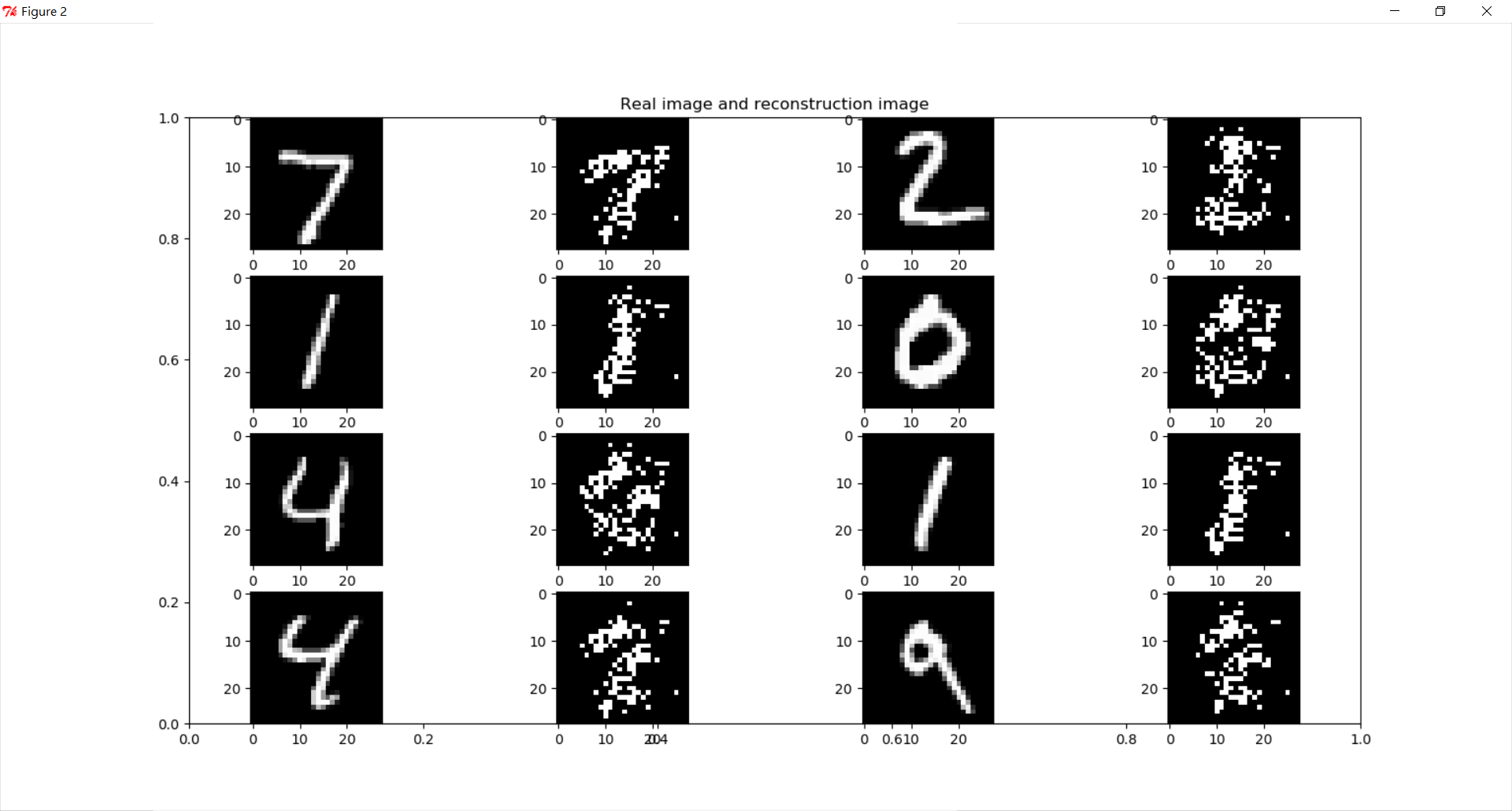


Fig. 5: Real image & reconstruct image

下圖為784 -> 128維度的權重值中取前8個neural，將其繪出，可以看到結果明顯偏向某些數字。

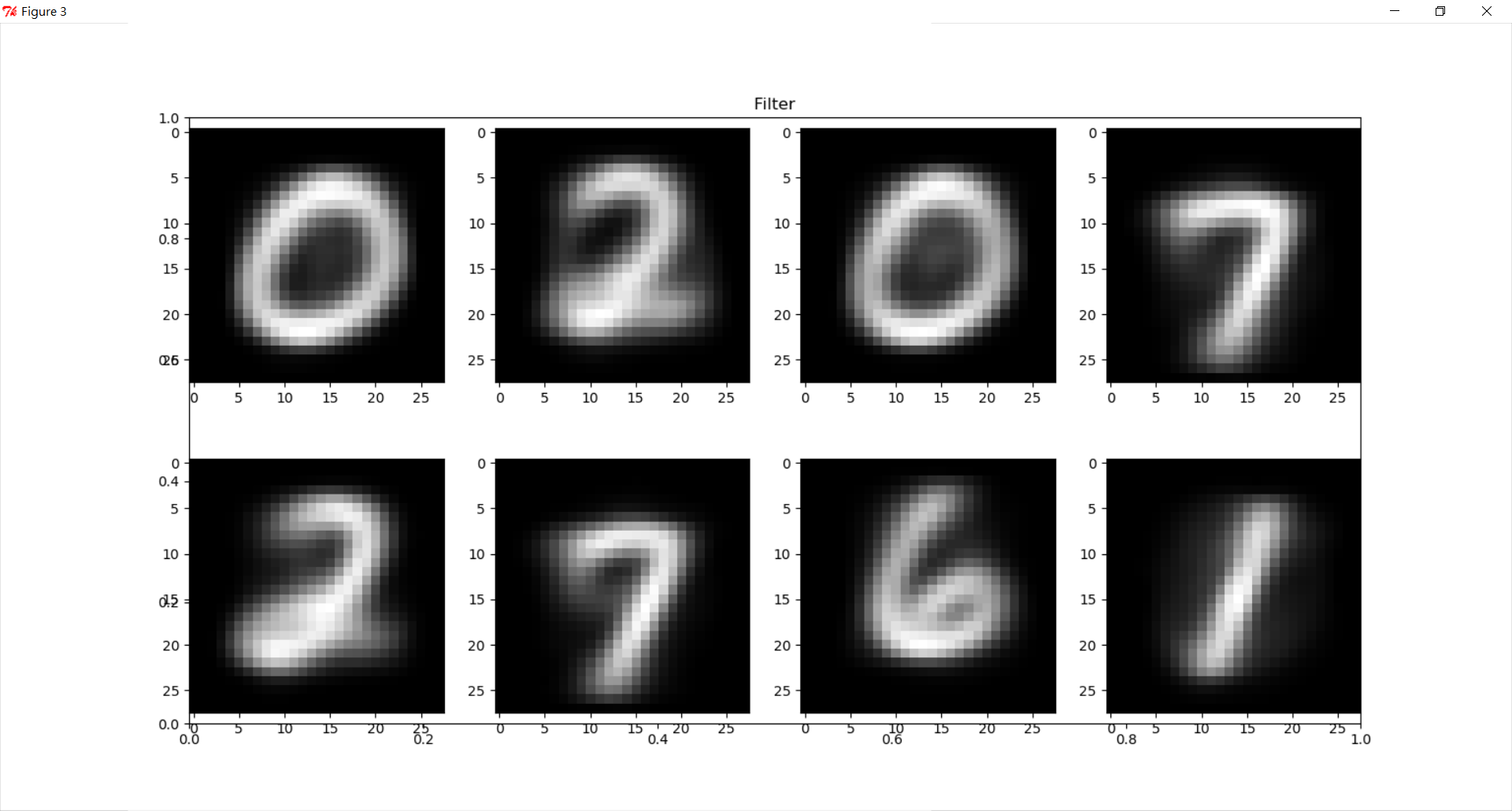


Fig. 6: Filter