

1. Power of Neural Networks

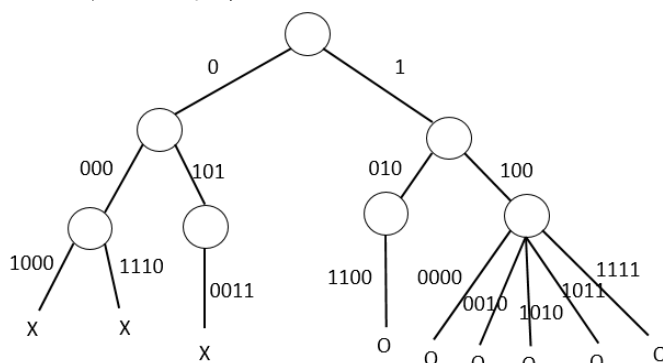
1. 不可能，因為在 3-2-1 就已經無法 implement 了，再多加一層也無濟於事，需要做的是增加每層的 neurons 數目，若增加層數可以使得可以 implement 的 function 數變多的話，那每層都用 1 個 neuron，然後一直增加層數即可。
2. 當 input 有 n 維時，分類的可能性有 2^n 種可能，而 hidden layer 每個 neuron 最多可以將 input 切成兩半，當 hidden layer 有 N 個 neurons 時，最佳情況下可以將 input 切成 2^N 種可能，所以 $N-N-1$ 能夠實現所有的 functions.

2. Hidden Tree in Multilayer Perceptron

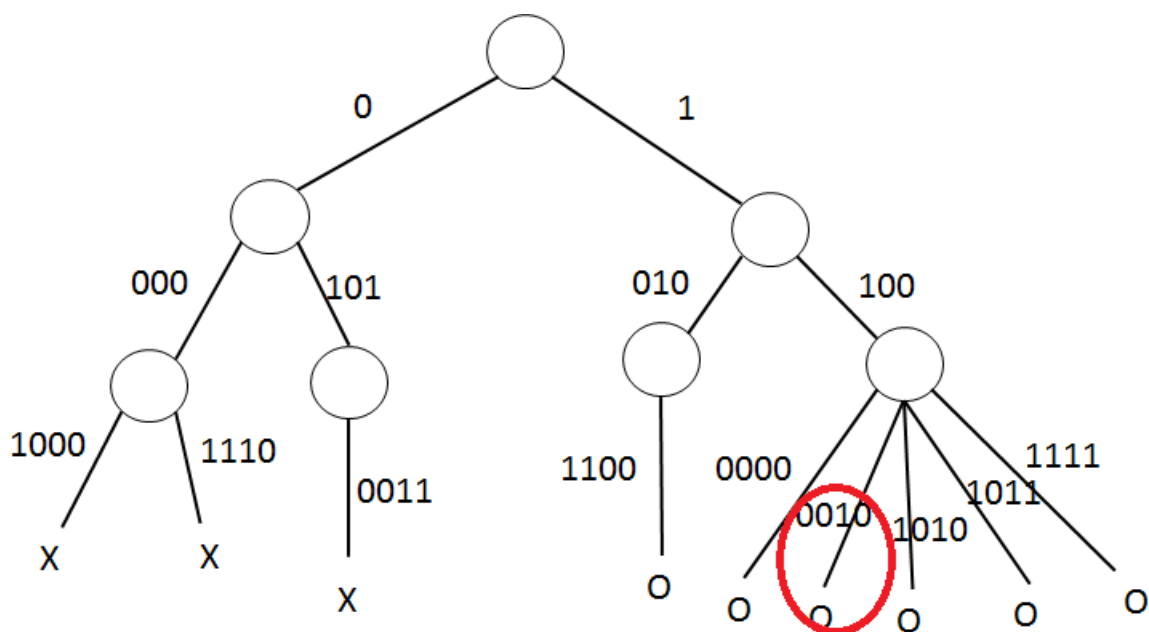
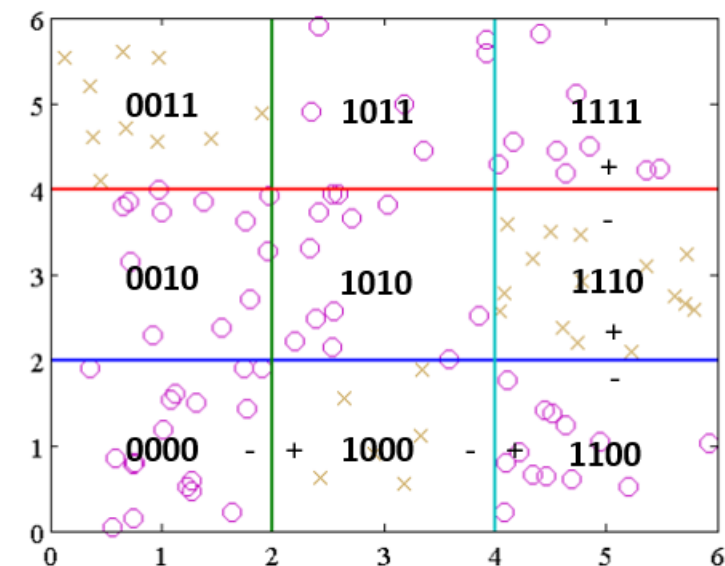
1. 自行產生如圖(a)的 data set，利用第三題寫出來的程式，建構出一 2-4-3-1 的 neural network，並固定第一層 hidden layer 的 weights，使得可以畫出如圖(a)中的四條線，其他 hidden layer 則正常的去 training，iterations 次數為 200。Training 完畢後，丟入 9 個點的資料，分別為(1,1), (3,1), (5,1), (1,3), (3,3), (5,3), (1,5), (3,5), (5,5)，看看每一層 hidden layer 的 neurons output value 為何，大於 0.5 則設為 1 反之則為 0。例：輸入(1, 1)，第一層 hidden layer 輸出(0,0,0,0)，第二層 hidden layer 輸出(1,0,0)，output layer 輸出(1)。

```
1
1
=====
0.268941421369995
0.047425873177567
0.268941421369995
0.047425873177567
=====
0.880764037921632
0.004195313077413
0.001038564256211
=====
0.990333637461859
=====
```

如此一來便可建構出 Hidden tree。



2.



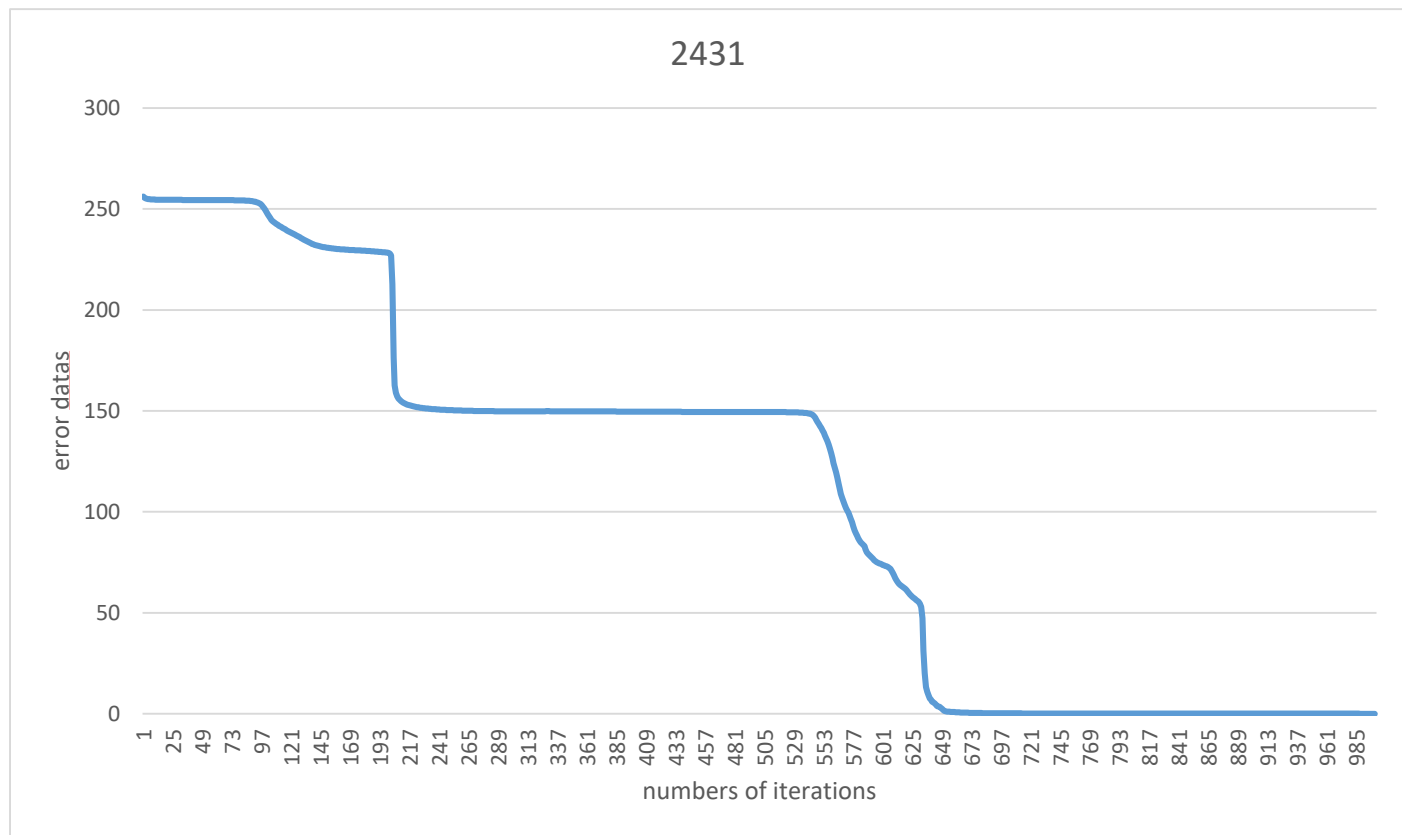
圈起來的部分會有 mixed set

3. Backpropagation Algorithm

1.

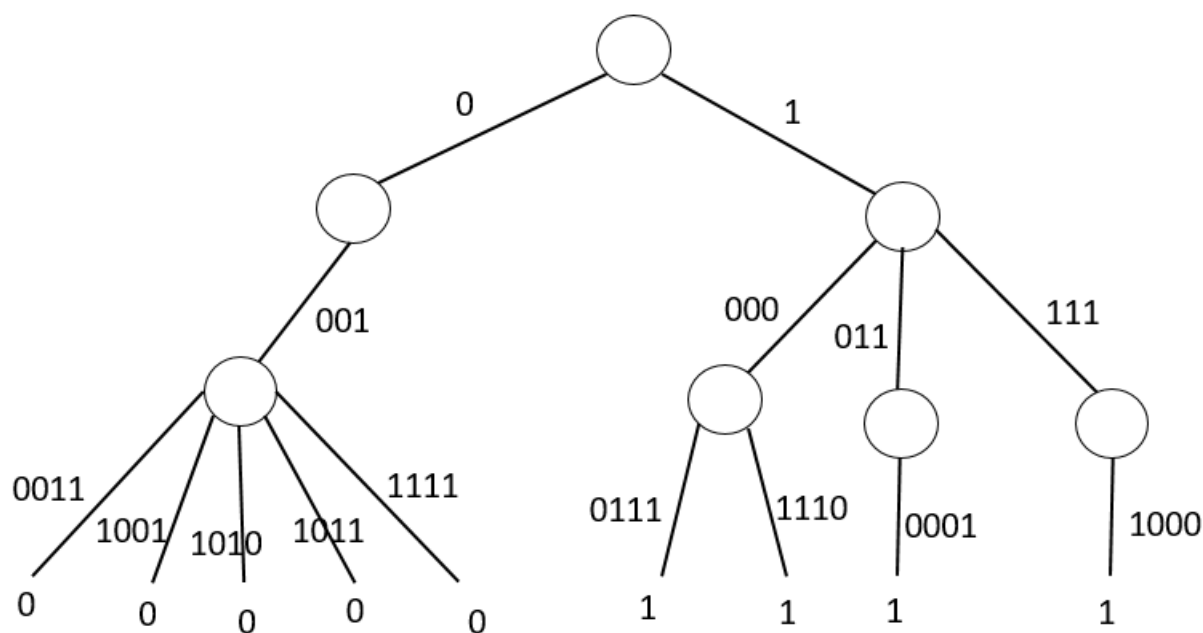
Initial weight range = 0~1, learning rate = 0.5, number of iterations = 1000.

Error curve:

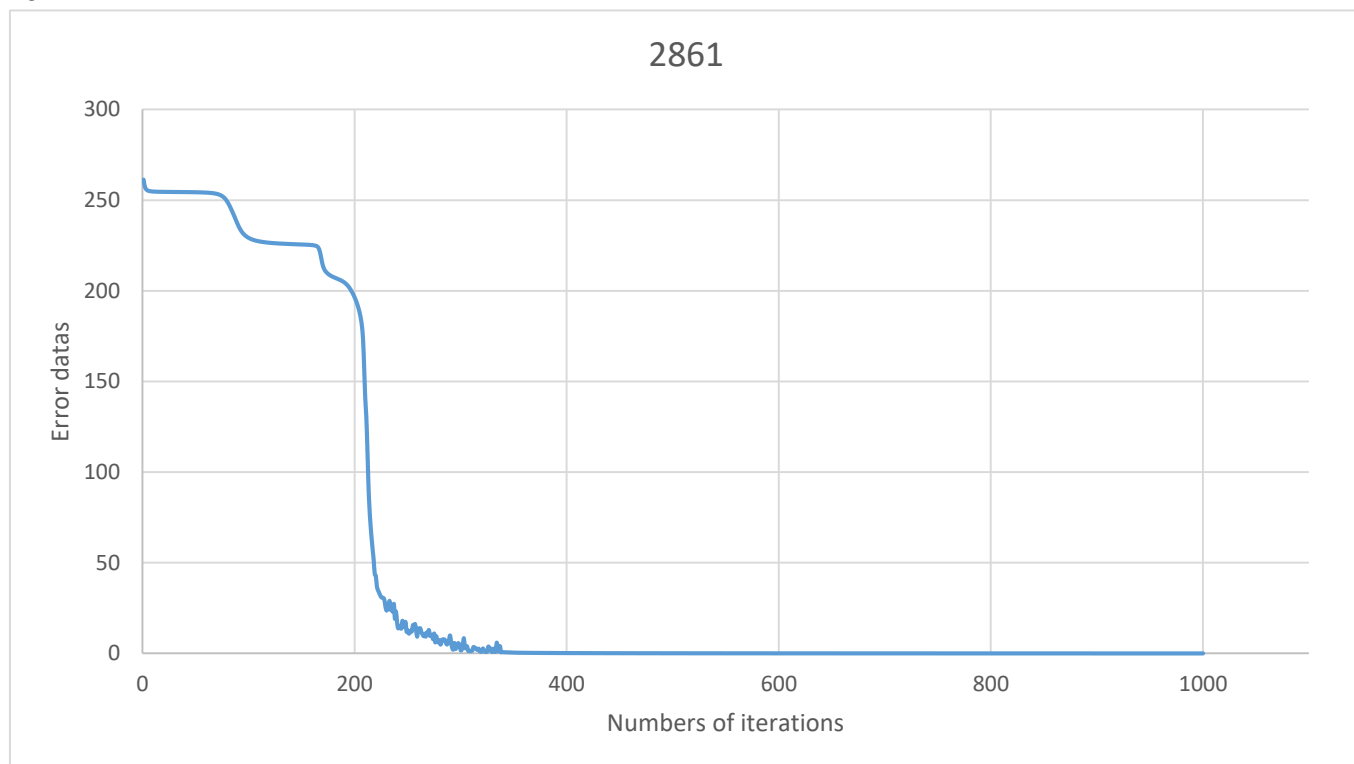


2.

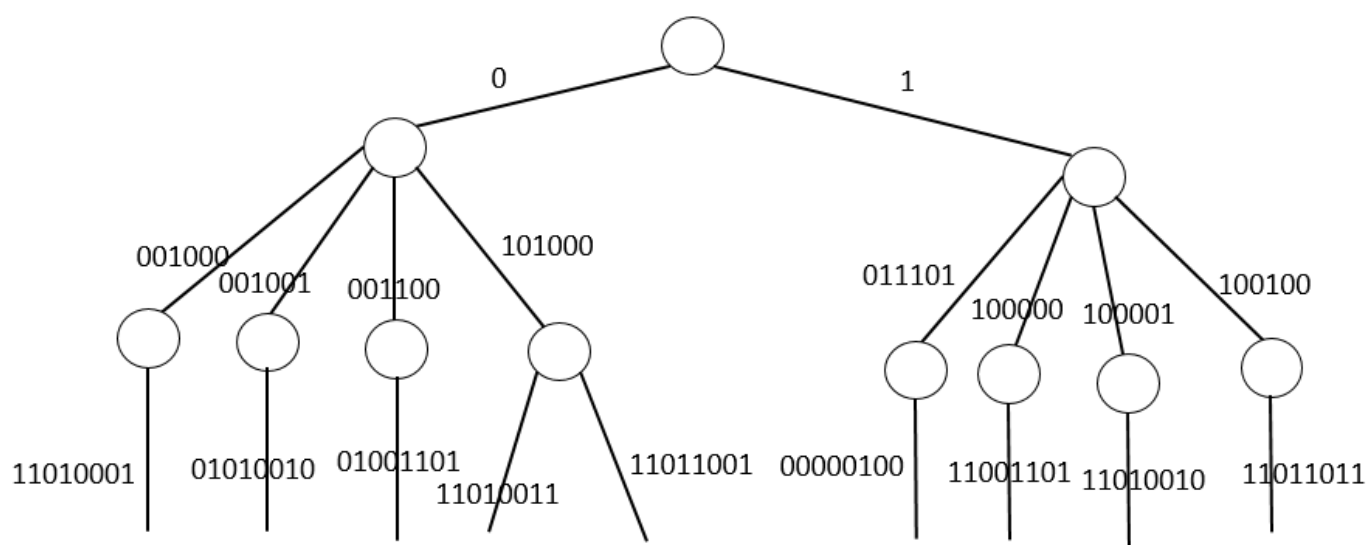
Hidden tree 建法同第二題



3. Error curve:



Hidden tree:



每層 hidden layer 的 neurons 數目較多，可以收斂的較快，且較沒有停滯的問題，實驗結果來看，iterations 次數只需要 2-4-3-1 的一半便可收斂到不錯的結果。

4. 執行程式：

程式語言：matlab

編譯環境：MATLAB R2016a

主要程式碼為 ./src/hw1.m，使用 MATLAB 開啟並執行即可。