Assignment #2 Are Deeper Networks Better?

F64126147 胡瑀真

注意事項 1: 書面報告可參考網路資料，但須理解與整理，自己理解與整理後撰寫報告，使用整段網路內容所獲的評分不高

注意事項 2: 勿分享報告給課程同學，相似內容的報告所獲評分不高

1. 說明是否參考或使用網路上程式，或程式自行撰寫?

(允許參考網路程式資源，在書面報告中須說明參考來源與有參考程式的範圍。如程式與模型全部自做，亦在報告中說明，有額外分數)

程式參考網路資料：iT 邦幫忙、生成式 AI ChatGPT，以及講義第五章內容。

2. a)類神經網路(Artificial Neural Network, ANN)待求解未知數為何?

b)激活函數在 ANN 扮演角色為何?

c)損失函數(Loss Function)在 ANN 扮演角色為何?

d) ANN 內的 Softmax 激活函數有何作用?

a) 類神經網路的待求解未知數（unknown parameters）為權重（weights）和偏差（biases）。在類神經網路中，要丟進激活函數（Activation Function）的參數 $z = w_1a_1 + \cdots + w_ia_i + \cdots + w_ka_k + b$，前式中的 w 和 b 分別就是權重和偏差。其中權重表示輸入特徵對預測結果的重要程度，偏差表示不輸入特徵仍會產生的模型偏移量，在訓練模型時，透過調整權重和偏差可使模型的預測更加準確。

b) 激活函數（Activation Function），若以生物學的角度說明，是用以控制單一神經元對訊號活躍程度，擁有將訊號衰減或是保留的能力；若用數值上的角度說明，則可以使類神經網路不再只有線性（如：上題中 $z = w_1a_1 + \cdots + w_ia_i + \cdots + w_ka_k + b$）的特性。透過激活函數可使類神經網路擁有非線性特性，讓類神經網路處理更加複雜的問題，否則，若無激活函數，則無論類神經網路疊多少層神經元，皆等同於一層神經元、一層線性轉換。舉例說明：Sigmoid Function $\sigma(z) = \frac{1}{1+e^{-z}}$便是一種激活函數，可以使輸出被限制在 0~1 之間，有讓數值再類神經網路中傳遞時不發散的優點。

c) 損失函數（Loss Function）是用以衡量「模型預測結果」和「真值」之間誤差的函數，若假設預測結果為$\hat{y}$，真值為 y，則損失函數計算$\hat{y}$和 y 的差距（$L(x) =$

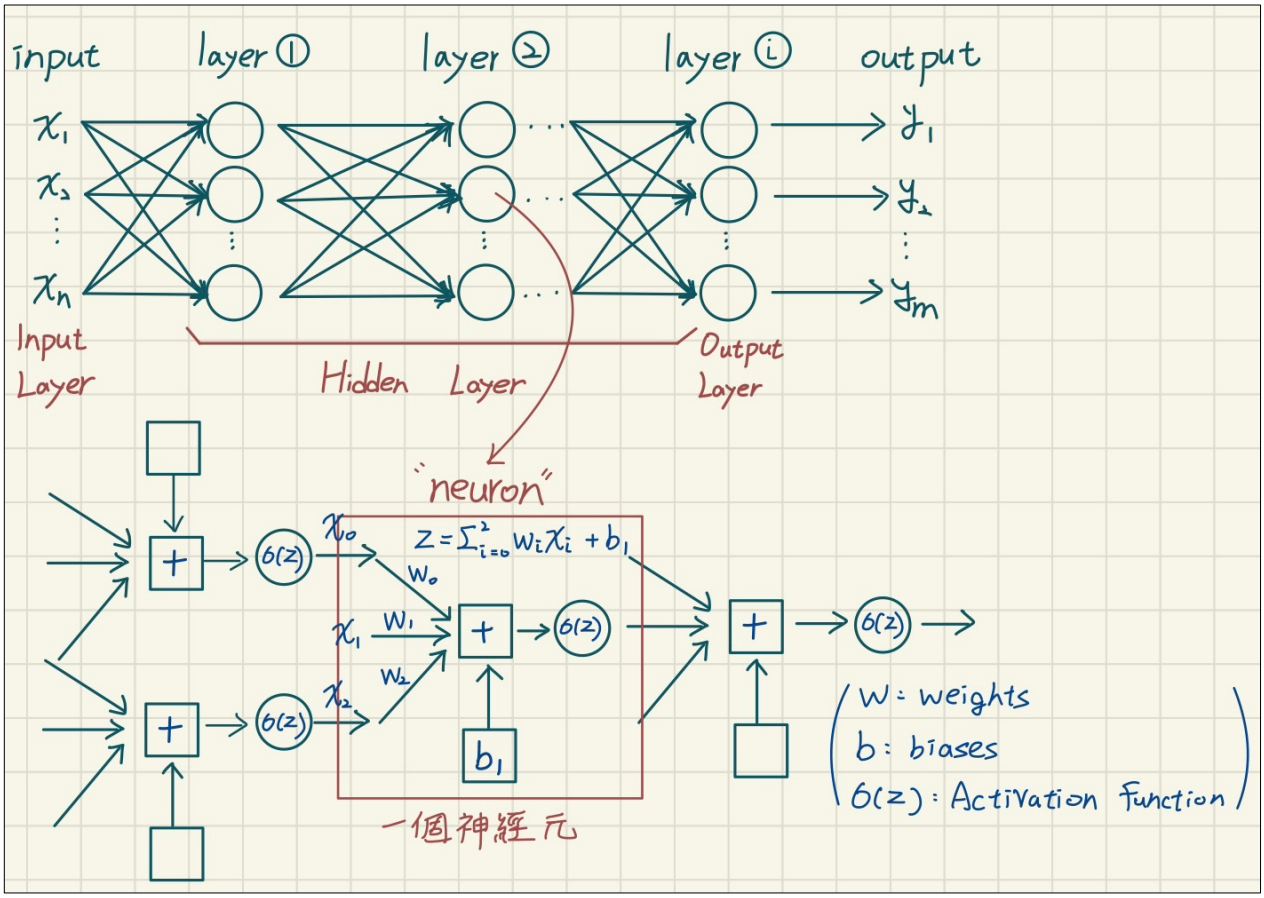
distance(y,ŷ))。在訓練模型時，藉由不斷最小化損失函數，使模型預測愈加準確，而其中一種最小化方法是梯度下降法（ Gradient Decent)。另外，在選擇 Loss Function 時，回歸問題使用 Mean Square Error，分類問題使用 Cross Entropy。

d) Softmax Function 具有兩個功能，分別是將神經網路的各個輸出轉換為機率，以及抑制較不可能的類別，突顯較可能的類別，達到 Single-peak Curve(例如：用 exponential，使大的更大，小的更小，讓單一類別最突出)。

3. 比較 10 個不同深度神經網路模型的分類準確度

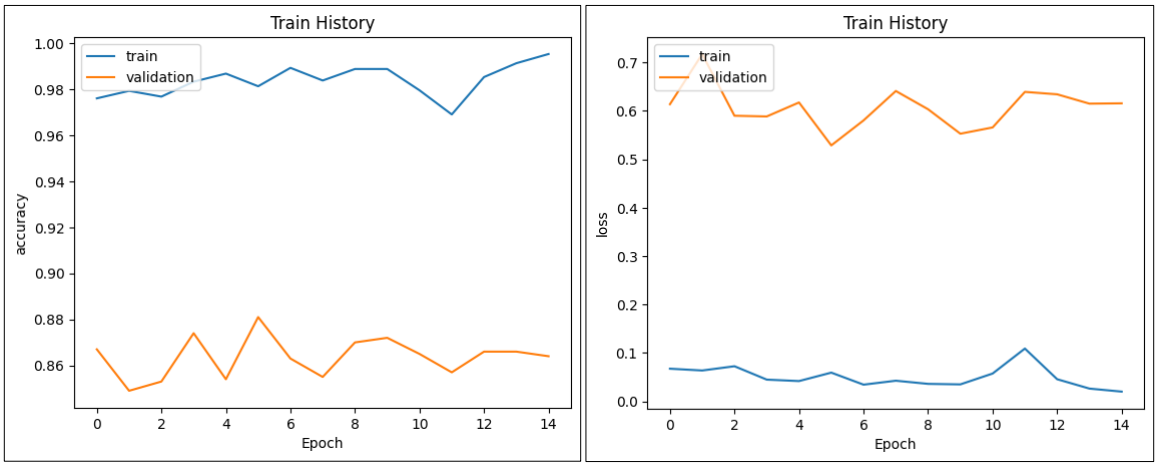
(列出神經網路模型簡易架構、比較這些神經網路模型分類準確度比較結果)

神經網路模型簡易架構



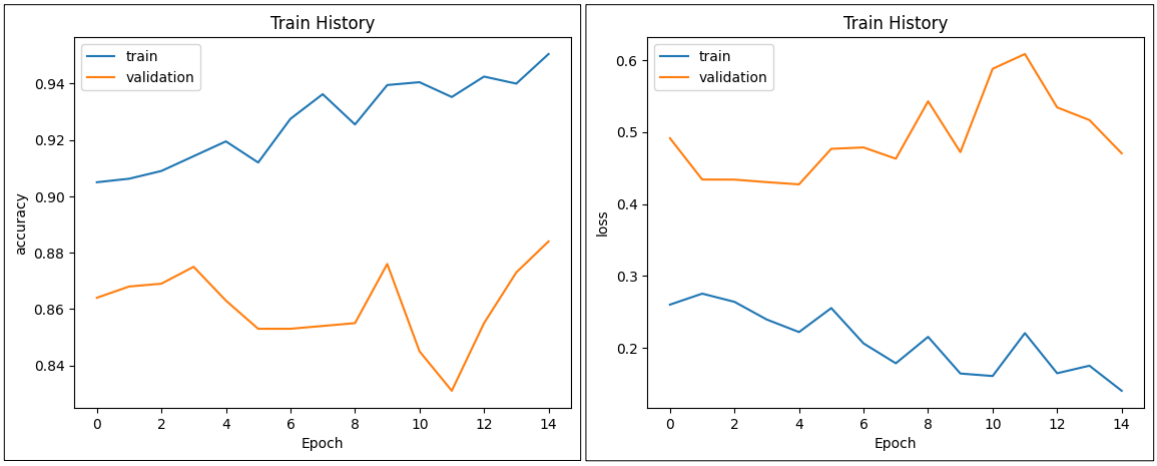
神經網路模型分類準確度比較結果

1 hidden layer



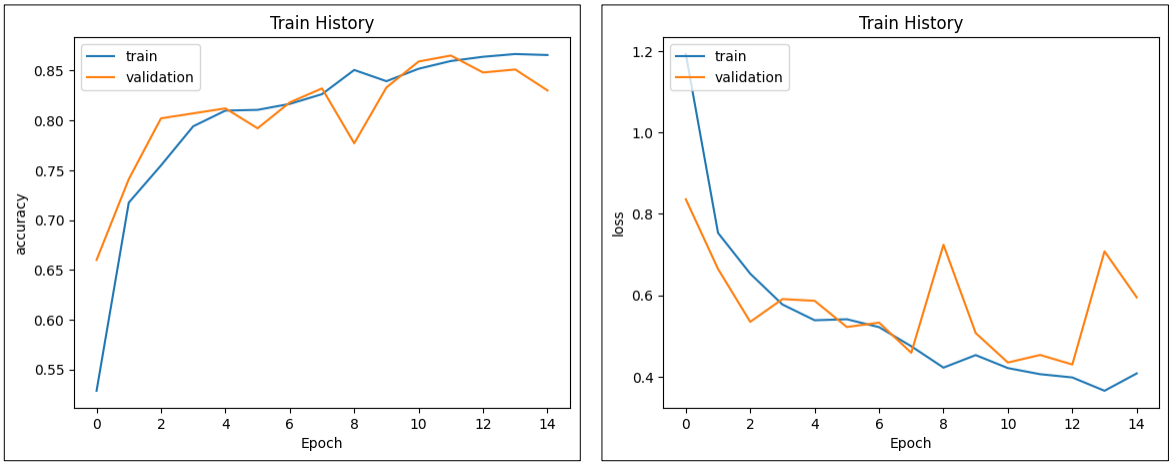
Accuracy of testing data = 83.7%

5 hidden layers



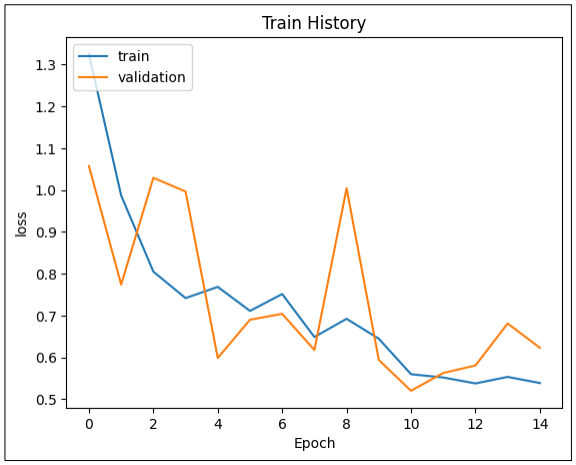
Accuracy of testing data = 83.0%

9 hidden layers



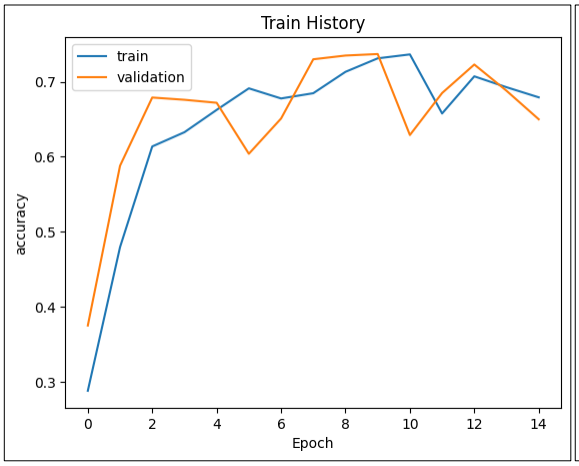
Accuracy of testing data = 80.0%

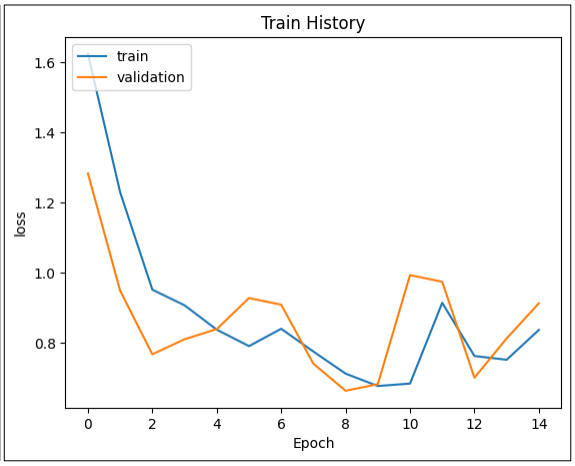
13 hidden layers



Accuracy of testing data = 75.3%

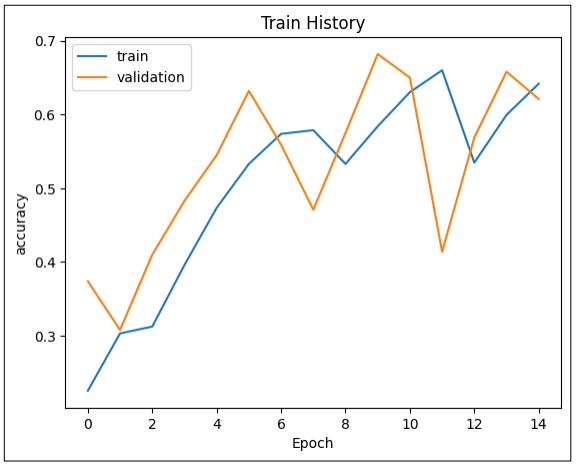
17 hidden layers

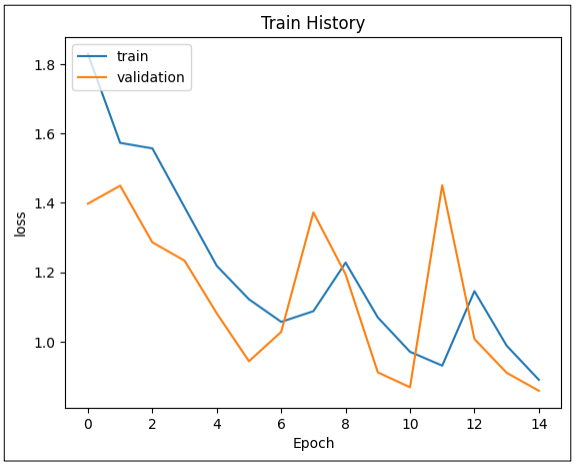




Accuracy of testing data = 61.9%

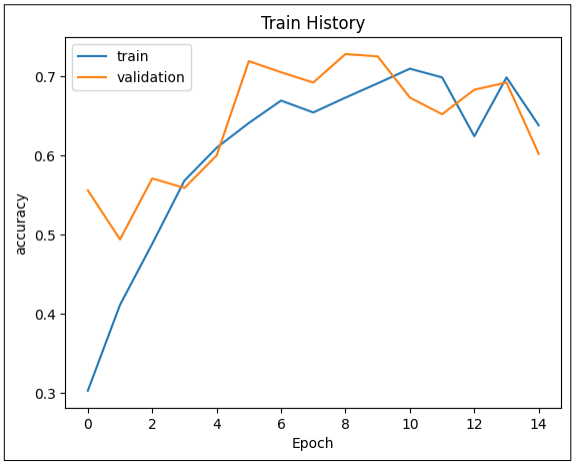
21 hidden layers

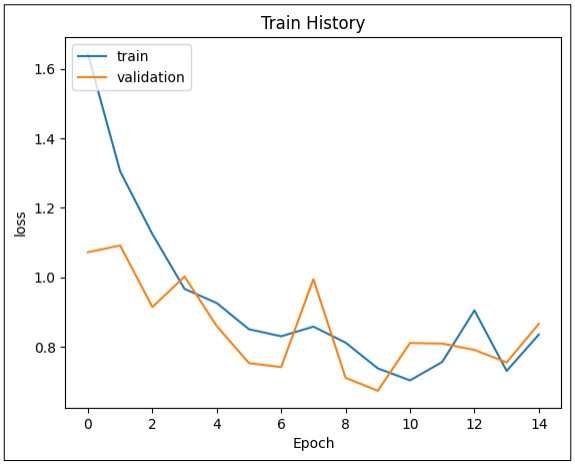




Accuracy of testing data = 59.9%

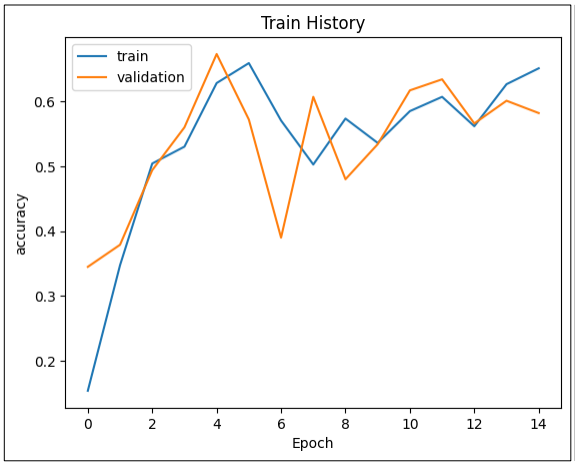
25 hidden layers

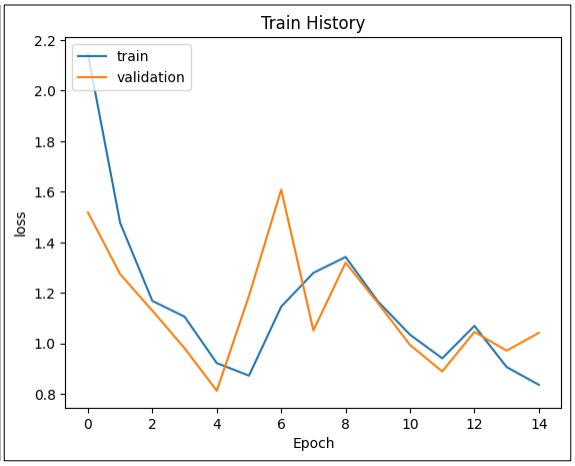




Accuracy of testing data = 59.0%

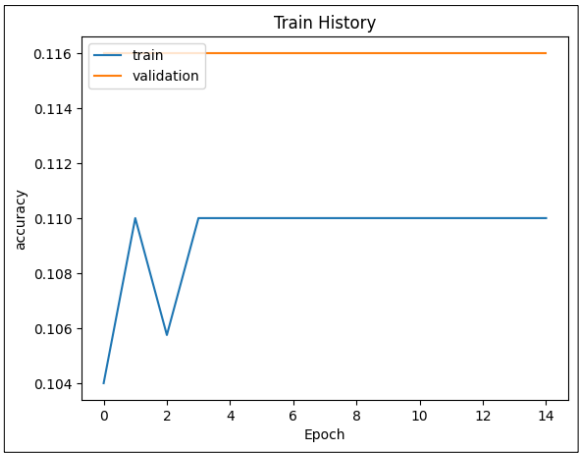
29 hidden layers

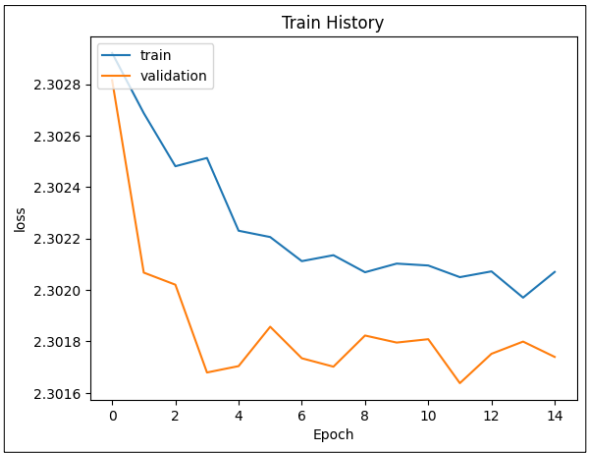




Accuracy of testing data = 56.5%

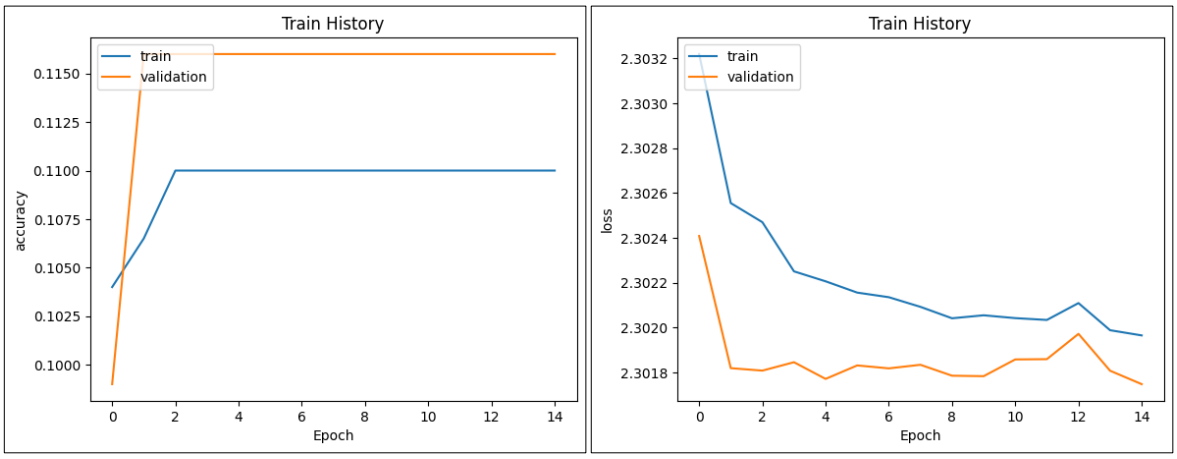
33 hidden layers



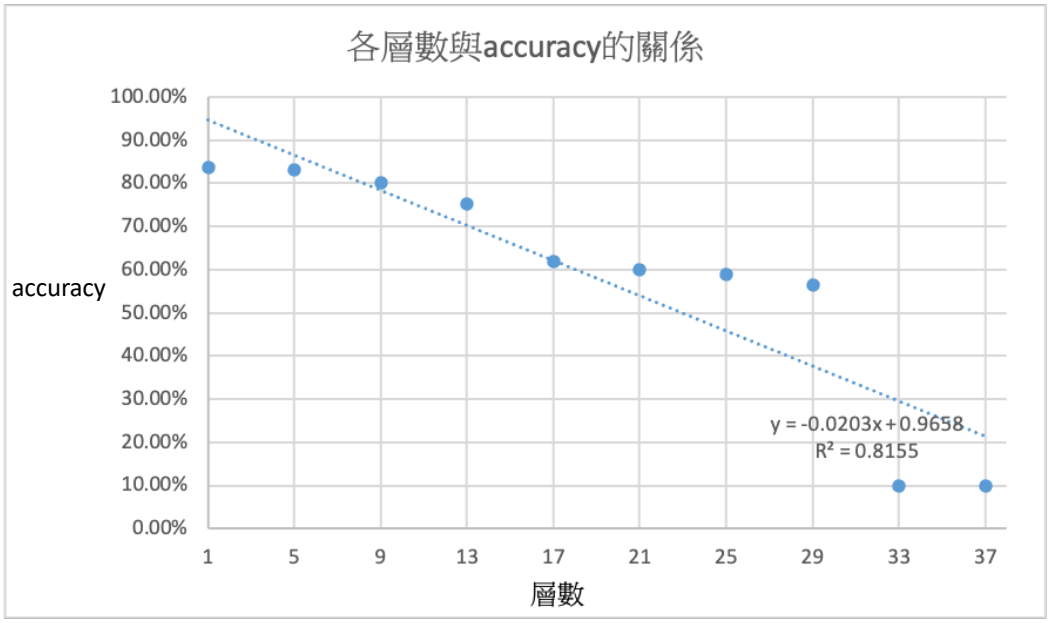


Accuracy of testing data = 10%

37 hidden layers



Accuracy of testing data = 10%



根據圖表，比較各深度與 accuracy 的關係，可發現深度和 accuracy 呈現負相關性：深度愈深，accuracy 愈低；深度愈淺，accuracy 愈高。

Accuracy 的最大值是 83.7%，對應的模型深度是 1 hidden layer；次高值是 83.0%，對應 5 hidden layers。accuracy 的最低值是 10%，對應的深度是 33 hidden layers 和 37 hidden layers；次低值是 56.5%，對應 29 hidden layers。

觀察圖表可發現，accuracy 在深度從 29 到 33 之間有明顯驟降，進一步觀察 Train History 的 accuracy 的圖表，可知：train 和 validation 的 accuracy 並沒有劇烈差異，且 training accuracy 較 validation accuracy 低，故推測不是過擬合（overfitting）所造成的 accuracy 下降，因為若是過擬合，training accuracy 應該會較 validation accuracy 高許多。

此外，從 Train History 的 loss 的圖表可看出：隨著深度加深，train 和 validation 的 loss 在訓練初期（epoch 較少，為 1 至 5 時）下降快速，在訓練中期（epoch 為 6 至 10 時）下降趨緩，雖偶有局部擺盪或小峰值，但整體趨勢仍維持穩定、沒有出現 NaN。

又在訓練末期（epoch 為 11 至 15 時）loss 趨穩，結合兩資料集的 accuracy 皆穩定偏低的現象，可推測不是梯度爆炸（exploding gradient）所造成的 accuracy 下降。

綜上所述，推測 accuracy 的明顯下降，是因爲網路深度過深導致梯度消失（vanishing gradient），由於梯度在 output layer 傳遞至 input layer 的過程中變得很小，最後趨近於零，使得後面的 layer 已收斂完畢，但前面的 layer 仍是初始亂數，令模型訓練的結果等同於亂猜。

4. 試著評論與回答”Are deeper networks betterg?”

不一定，從本次作業的結果可知，深度愈深不一定可使 accuracy 愈高。

當神經網路愈深，運算的計算量會愈大、耗時愈長，並且，因為常見的最小化損失函數方法——梯度下降法（gradient decent）的梯度問題源於神經網路過深，若神經網路太深，可能會使梯度消失（vanishing gradient）或梯度爆炸（exploding gradient）的情況產生。

若發生梯度消失（vanishing gradient）的情況，表示梯度從 output layer 傳遞至 input layer 的過程中逐層變小，最終趨近於零，導致權重更新的幅度極小，進而使模型無法有效學習；相對地，若發生梯度爆炸（exploding gradient），則梯度從 output layer 傳遞至 input layer 的過程中急遽放大，使未知參數（unknown parameters）的更新大，變得難以收斂，使訓練過程失敗、找不到最佳解，類似於誤差傳播，傳播愈久會使誤差愈大的情況。

另外，當神經網路愈深，模型會愈加複雜，因此當問題單純時，過複雜的模型易導致過擬合（overfitting），使模型在 training data 表現良好（高 R^2）但在 validation data 表現差勁（低 R^2），令模型失去應用在 training data 外的能力。

參考資料 (書面報告的參考資料)

1. 上課講義…..
2. http://www.deeplearning.com