Basic:

1. 遇到的問題:

   (1)在寫 linear_activation_backward 這個 function 時，因為對於 linear_cache, activation_cache 的用法不是很懂，將錯誤的 cache 傳到其他 backward 會用到的 function 時，導致答案怎麼試都不對，雖然是很小的錯誤，但也花了我蠻多時間找出來的。

   (2)一開始寫的時候，有些矩陣運算(如 sigmoid)地方助教提示說用一行 code 就可以完成，因為不是很熟悉 python，所以一開始還是用 for 迴圈遍歷每個元素，之後上網查資料才知道如何用更便捷的方式完成。

   整體來說，助教在 template 給的提示都很清楚了，只要一步一步做基本上不會遇到很嚴重的 bug。

2. 實作 build the binary classifier

   主要的步驟在最後 implementation 中的 L_layer_model 都表示得很清楚了:

   (1) initialize_parameters，初始化 W 跟 b。

   (2) 根據當前的 parameters 執行 L_model_forward，也就是 forward propagation

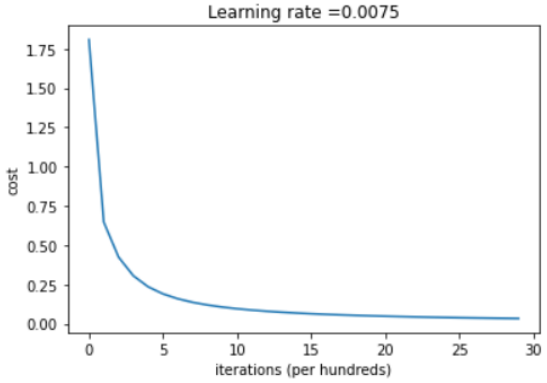
   (3) 將得到的 AL 輸入 compute_BCE_cost 得到 cost

   (4) 執行 L_model_backward 得到 gradiant，也就是 backward propagation

   (5) 根據得到的 gradiant 去更新 parameters，做完之後回到第(2)步驟，繼續下一次的 iteration。

3. 結果

   最後執行 implementation 時，layers dimension 設成[4,1]，因為是 binary classifier，所以結果只會是 1 或 0，最後的 node 也只會有一個。至於執行 function 的部分，learning_rate = 0.0075, , num_iterations = 3000。執行的結果 cost 可以降到 0.0329，train_data 跟 validation_data 的 accuracy 都是 1。



Bonus:

1. 遇到的問題:

(1) 寫 bonus 時，我其實沒有遇到甚麼困難，只有在 Linear_activation backward 中的 softmax_CCE_backward 寫比較久。因為我花了一些時間看懂 derivative of the softmax function，看懂之後發現其實 code 很簡單。

2. 實作 build the multi-class classifier:

跟 basic 不同的是，最後的節點不只一個，也就是說最後的分類選項不只一個。Bonus 用的 activation function 是 softmax，所以時做的部分只有跟 basic 比只多了 forward propagation 的 softmax function, backward propagation 的 derivative of softmax 還有 Categorical cross-entropy loss 這三個部分。

在 L_layer_model 中，會看 classes 的數量來決定要用哪一種 cost 的計算方式。forward 跟 backward propagation 中也會依照 classess 的數量決定 activation function。

3. 結果

最後執行 implementation 時，經過不斷的測試調整參數，layers dimension 設成[64, 64, 32, 16, 4]，train_data 跟 validation_data 的比例是 7:2。至於執行 function 的部分，learning_rate = 0.0075, ，num_iterations = 30000。執行的結果 cost 可以降到 0.0947，train_data 的 accuracy 約為 0.999375，validation_data 的 accuracy 約為 0.8874。

一開始看到 train_data 的 accuracy 可以到接近 1，我當時覺得可能會有 overfit 的問題，但 validation_data 的 accuracy 也很高。我推測是因為測資的分類結果只有 4 個，而一開始的 input X 有 64 個，所以 validation 的 accuracy 才可以到接近 9 成。我也有嘗試其他的 layers dimension 和不同的 iteration，出來的 cost 會有些不同，但最後的 train 跟 validation 的 accuracy 都沒辦法再有明顯升高了。