MLDS hw1-1、hw1-2、hw1-3 Report

0. Team work：

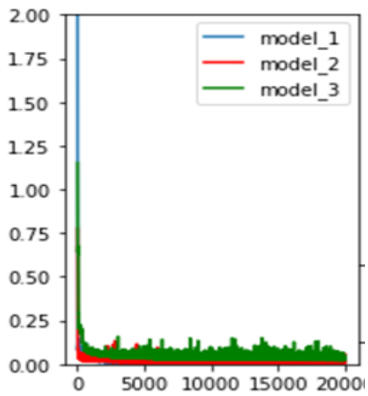
曾柏偉	程式撰寫、報告撰寫、比較結果
張嘉麟	報告撰寫、比較結果
劉宏國	程式撰寫，報告撰寫

1. Simulate a Function:

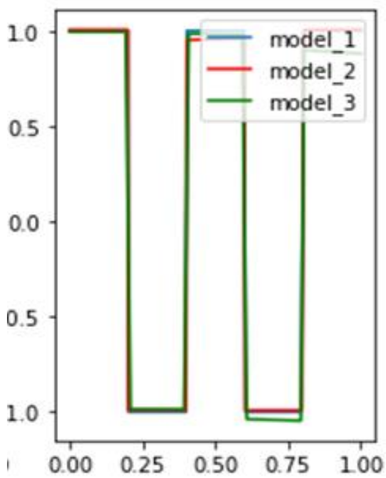
A. Describe the models you use, including the number of parameters (at least two models) and the function you use. (0.5%)

我使用助教投影片上三個 model 的範例，參數個數分別是：(571 , 572 , 571)，然後使用步階函數，訓練 20000 個 epochs。

B. In one chart, plot the training loss of all models. (0.5%)



C. In one graph, plot the predicted function curve of all models and the ground-truth function curve. (0.5%)



D. Comment on your results. (1%)

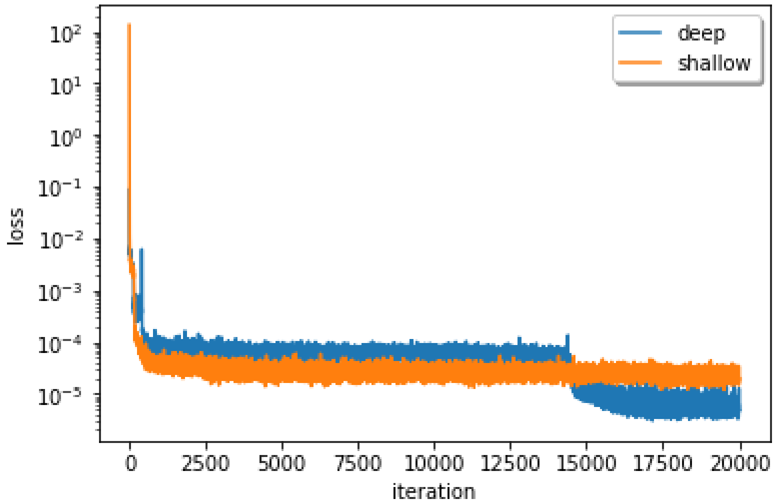
其實訓練上的 loss 三個模型都是滿小的，也幾乎在兩萬次 epochs 後都滿能 fit 上步階函數(在值域零至一內)，比較不好的應該是在模型訓練上可能比較不穩定。

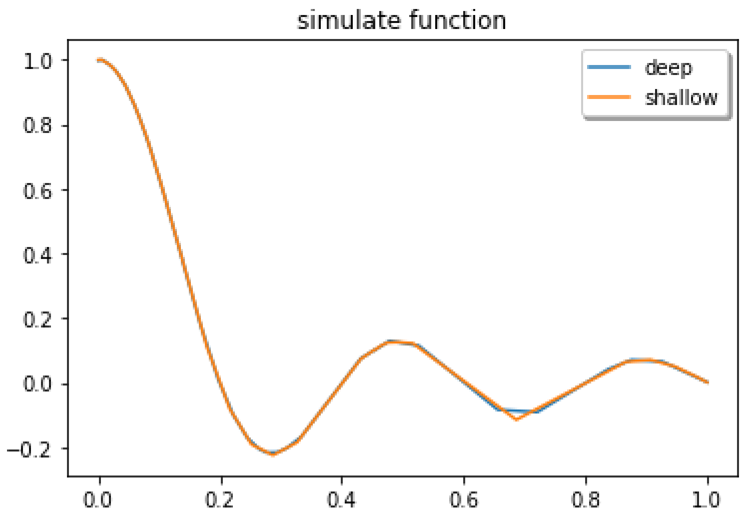
E. Use more than two models in all previous questions. (bonus 0.25%)

B，C 小題分別都做了三個 model 的圖了。

F. Use more than one function. (bonus 0.25%)

使用了 sinc function：





2. Train on Actual Tasks:

A. Describe the models you use and the task you chose. (0.5%)

Model 1：CNN + pooling + flatten + hidden layer

Model 2：CNN + pooling + CNN + pooling + flatten + hidden layer

loss function = softmax_cross_entropy_with_logits

optimizer = Adam

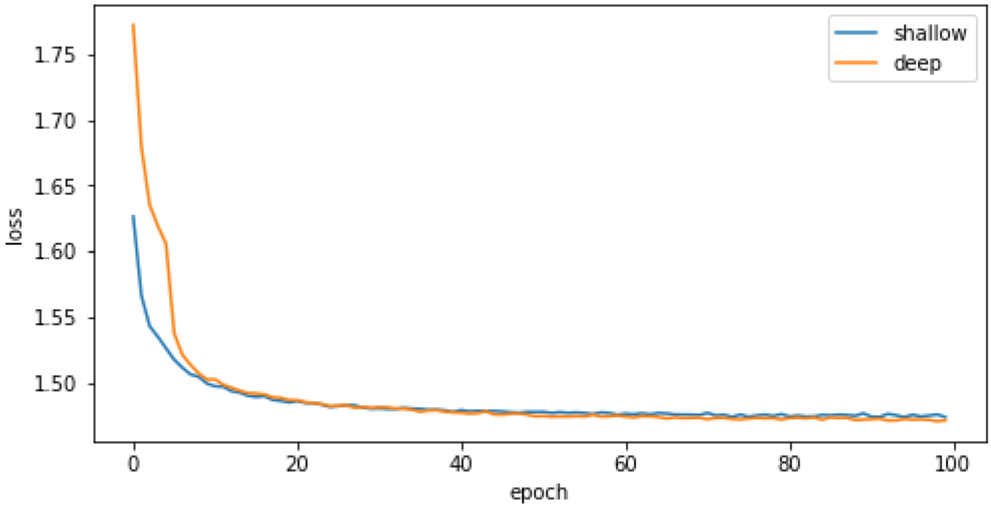
learning_rate = 0.0001

trainEpochs = 100

batchSize = 300

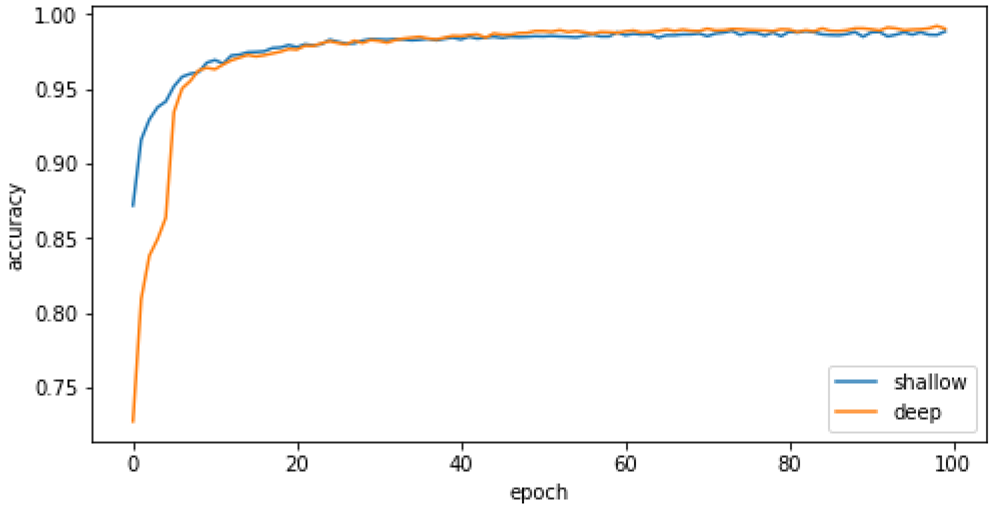
B. In one chart, plot the training loss of all models. (0.5%)

Mnist dataset



C. In one chart, plot the training accuracy. (0.5%)

Mnist dataset



D. Comment on your results. (1%)

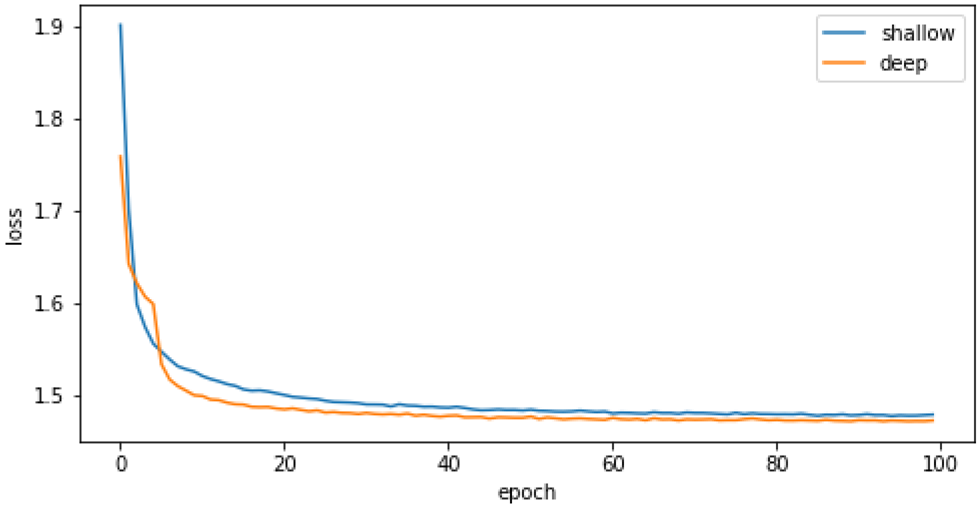
由上述兩張圖所示，在前幾個 epoch 中，shallow 的表現或許較 deep 的 model 要來的好，但是越多 epoch，deep model 的表現會比 shallow 好一點，但是沒有較明顯的差異，可能是因為 task 本身較簡單的因素。

E. Use more than two models in all previous questions. (bonus 0.25%)

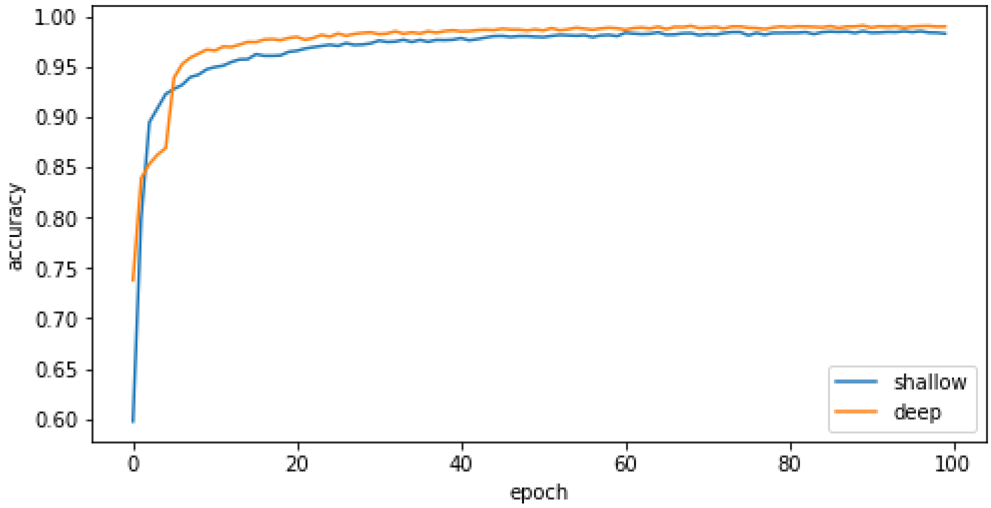
由上圖所示已有比較兩種 model。

F. Train on more than one task. (bonus 0.25%)

Cifar dataset Loss



Cifar dataset accuracy



=============================hw1-2============================

3. Visualize the optimization process.

A. Describe your experiment settings. (The cycle you record the model parameters, optimizer, dimension reduction method, etc) (1%)

使用 Mnist 資料集，每 3 個 epochs 存一次 weight，一個循環存 8 次，總共 8 個循環。

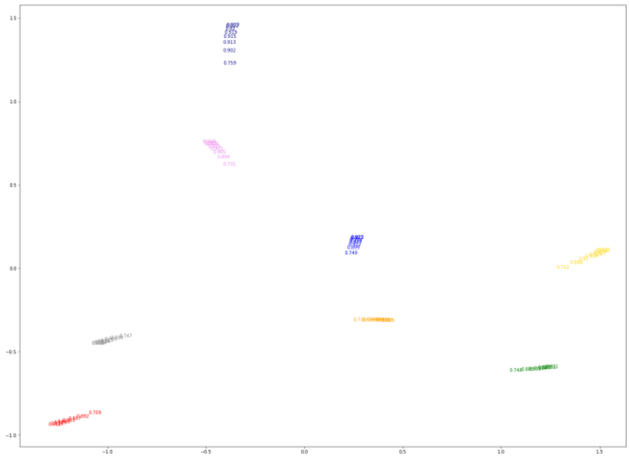
activation = softmax、

Loss = categorical_crossentropy、

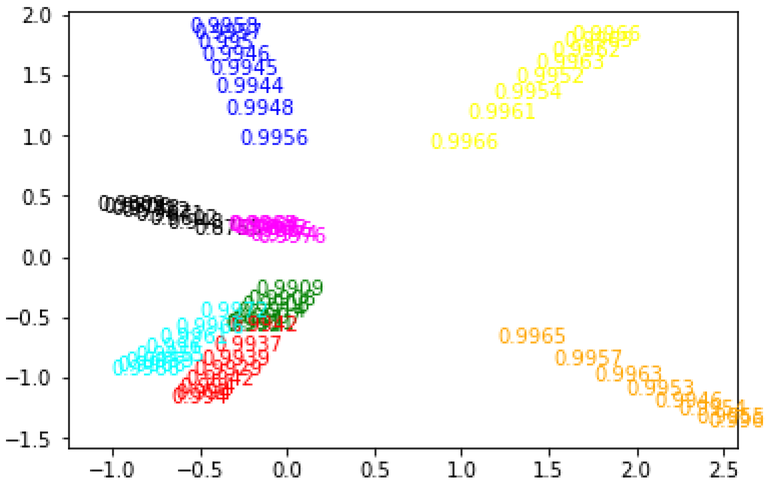
optimizer=adam。

B. Train the model for 8 times, selecting the parameters of any one layer and whole model and plot them on the figures separately. (1%)

以下 one-layer 做出來的圖:



以下是全部 model 的參數下去做出來的圖:



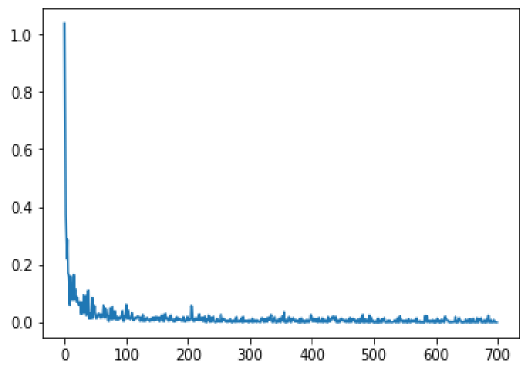
C. Comment on your result. (1%)

上圖的數值表示準確度，而由上圖所知，隨著訓練的 epoch 越多，數值所代表的點越集中到中央。

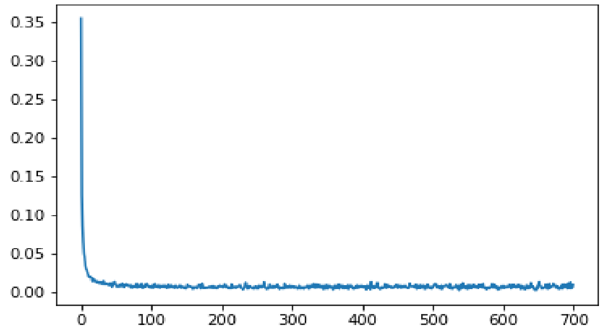
4. Observe gradient norm during training.

A. Plot one figure which contain gradient norm to iterations and the loss to iterations. (1%)

以下為 gradient norm to iteration :



以下為 loss to iteration :



B. Comment your result. (1%)

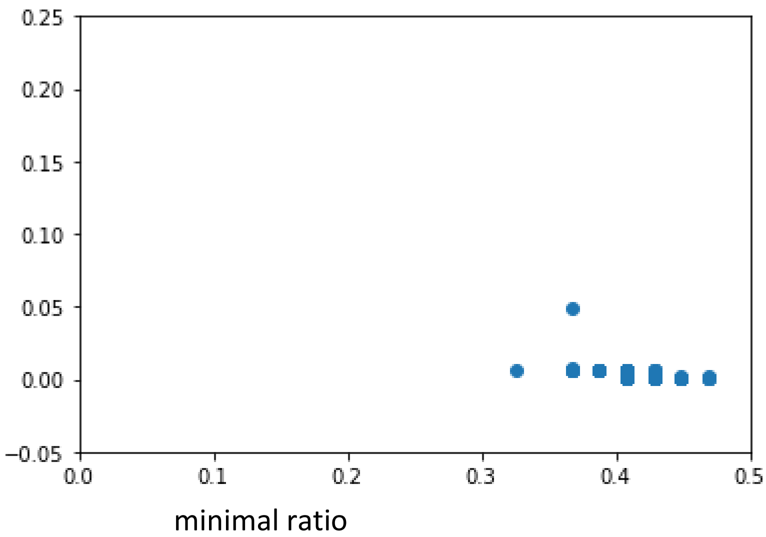
隨著 training epochs 增加，應該就越接近 local minima /global minima 了，所以 gradient norm 越來越小，故甚至已經快走不動，從圖表看起來已經卡在 minima 內走不出去了。

5. What happens when gradient is almost zero?

A. State how you get the weight which gradient norm is zero and how you define the minimal ratio. (2%)

我先訓練原本的 cost function ： MSE ，然後在換成 gradient norm 訓練，gradient 是幾乎是趨近於 0 了，然後 minimal ration 的定義是 hessian matrix 的特徵值大於零個數的比值。

B. Train the model for 100 times. Plot the figure of minimal ratio to the loss. (2%)



C. Comment your result. (1%)

沒有辦法像助教做的那麼漂亮，感覺是因為 loss 的差距其實都在 0.00001 那種很小的數字還有在當下的 minimal ration 也都是差 0.0001 的數字才導致點都黏在一起了，但是到最後 100epochs 的 minimal ration 就大於 0.5 了，完全收斂了。

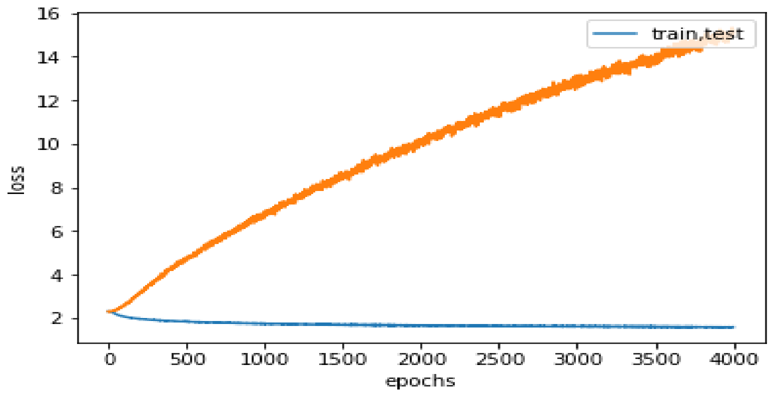
6. **Bonus (1%)**
   A. Use any method to visualize the error surface.
   B. Concretely describe your method and comment your result.

========================hw1-3==============================

7. **Can network fit random variables?**
   A. Describe your settings of the experiments. (e.g. which task, learning rate, optimizer) (1%)

   我是使用 MNIST 資料集，並且隨機 shuffle 過標籤，使用的是二層 Dense，一層 output (softmax)，分別是(1024 , 625 ,10) ，每層 dense 間有接 dropout (機率為 0.5) 。

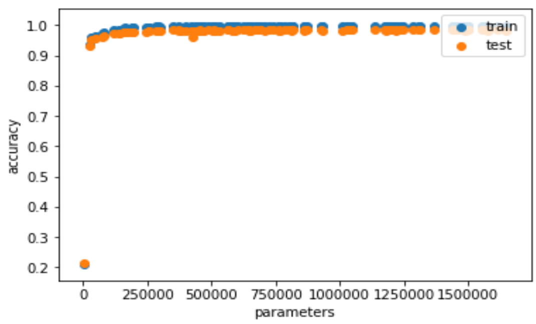
   B. Plot the figure of the relationship between training and testing, loss and epochs. (1%)

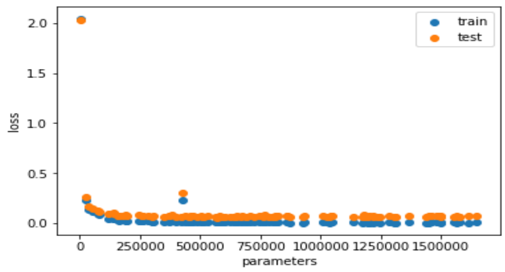


8. **Number of parameters v.s. Generalization**
   A. Describe your settings of the experiments. (e.g. which task, the 10 or more structures you choose) (1%)

   使用 MNIST 資料集，並且訓練 100 個不同的模型，模型參數是我在[1,1024]間隨機選取兩個參數丟進去模型內訓練。

   B. Plot the figures of both training and testing, loss and accuracy to the number of parameters. (1%)





C. Comment your result. (1%)

在模型參數越來越多的情況下，模型越能記住整個 training data 的形式，但是其實好像是 overfitting，所以對於這個 MNIST 可能用不到這麼多參數就可以達到在 training data 上面很好的效果，但是 testing data 上面就不顯著了。

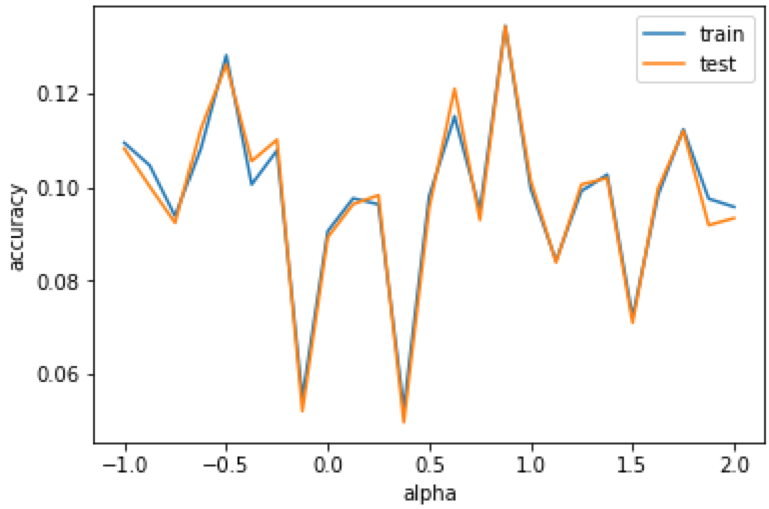
9. Flatness v.s. Generalization

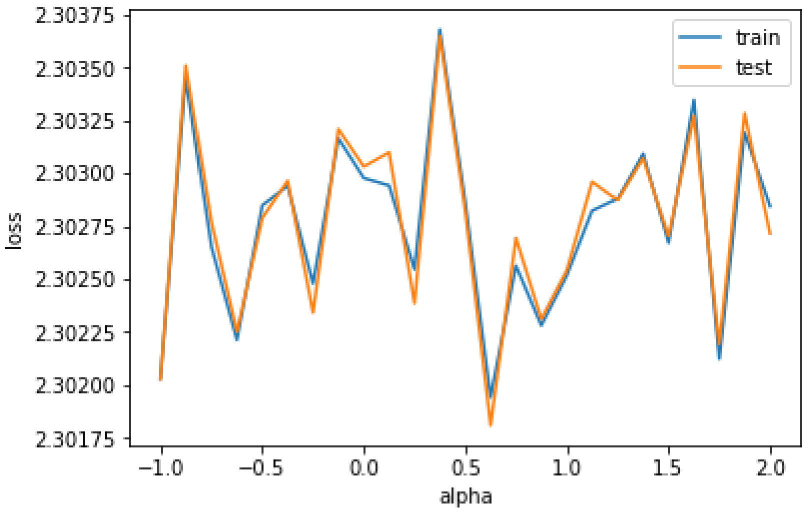
A. Part 1:

1. Describe the settings of the experiments (e.g. which task, what training approaches) (0.5%)

   我使用 batch size = 128 and 256 訓練兩個不同的模型，內差方面我是在(-1,2)區間內平均分割 25 個點，也就是有 25 個 alpha，模型的參數都和一二題一樣為三層 dense。

2. Plot the figures of both training and testing, loss and accuracy to the number of interpolation ratio. (1%)





3. Comment your result. (1%)

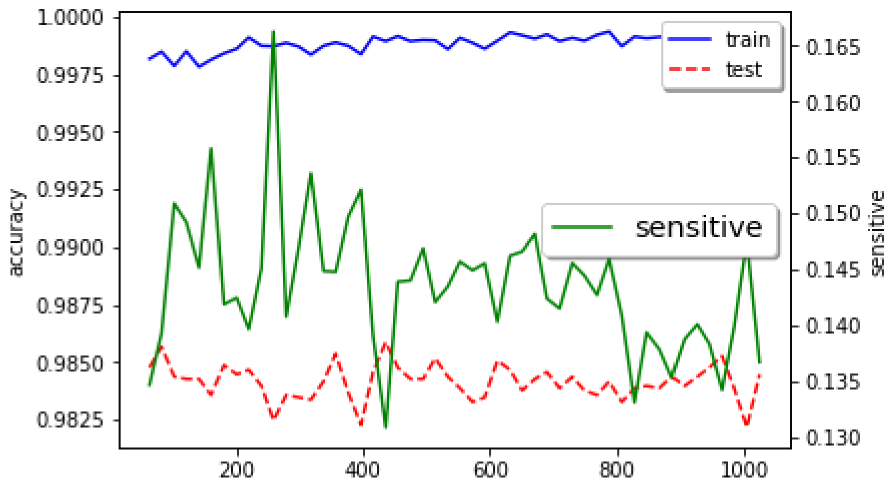
圖跟助教的圖形比起來尖了一些應該是取樣的 alpha 太少了，但是 train 和 test 內插過後的 loss 和準確率看起來是不會差太多的。

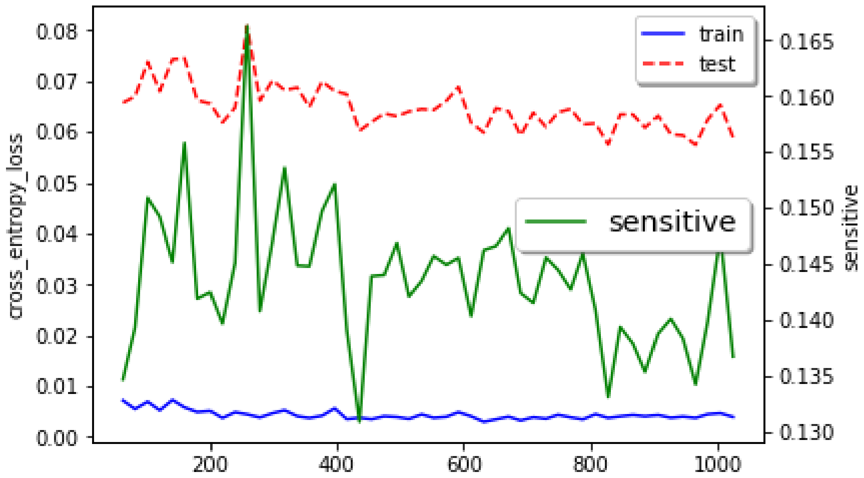
B. Part 2 :

1. Describe the settings of the experiments (e.g. which task, what training approaches) (0.5%)

我使用不同的訓練方法是不斷的改變 batch size，在(64,1024)區間內取 50 個值去訓練，模型參數和上面都是相同的，sensitive 就是依照助教給的定義去實作，最後橫軸是 batch_size 的大小。

2. Plot the figures of both training and testing, loss and accuracy, sensitivity to your chosen variable. (1%)





3. Comment your result. (1%)

其實改變 batch_size 的 sensitive 的變化起伏滿大的，但是在 train 和 test 的 loss 差異很大的時候 sensitive 的值就大了，然後差異較小的時候值也會掉下來，雖然圖形感覺挺奇怪的，但好像還是有一點意義。

10.Bonus :

A. Use other metrics or methods to evaluate a model's ability to generalize and concretely describe it and comment your results.