機器學習及其深度化與結構化 HW1

姓名	學號	工作
吳政軒	R06922118	1-1-2、1-2-1、1-3-1、1-3-2、report
黃敬庭	R06944049	1-2-2、1-2-3、1-2 bonus、1-3-3、1-3 bonus、report
馬欣婕	R06946010	1-1-1、1-1-1 bonus、1-1-2 bonus、report

1-1

- Simulate a Function:
    - Describe the models you use, including the number of parameters (at least two models) and the function you use. (0.5%)

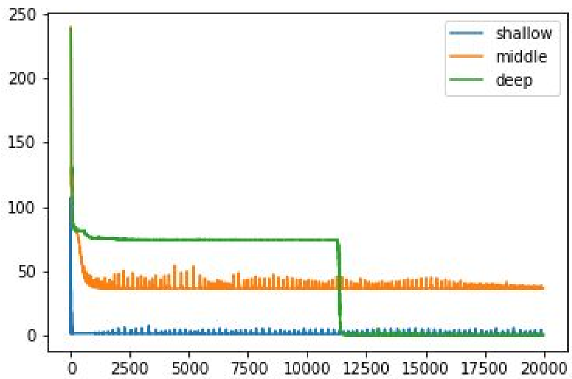
    (1) 使用模型

	shallow	middle	deep
hidden layer 數	1	3	6
unit 數	200	12, 20, 15	12, 10, 10, 10, 10, 10
parameter數	601	615	605
訓練	均為 Adam(lr=0.01), MSE loss		

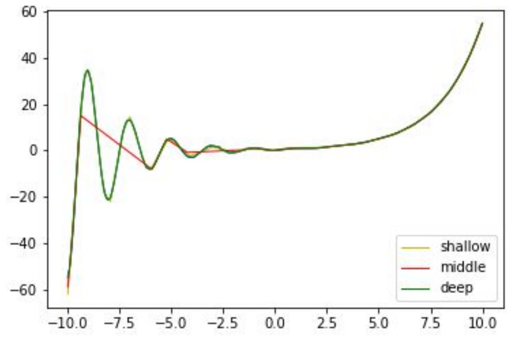
    (2) fit 的式子

    $$F_\nu = \frac{1}{\sqrt{5}}\left\{\left(\frac{1+\sqrt{5}}{2}\right)^{\nu} - \left(\frac{2}{1+\sqrt{5}}\right)^{\nu}\cos(\nu\pi)\right\},$$

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



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



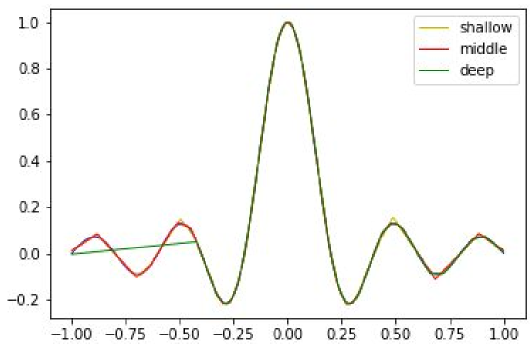
- Comment on your results. (1%)

  shallow 跟 deep模型均可達到完整fit 效果，middle 模型反而無法。

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

  使用三種模型如前述。

- Use more than one function. (bonus 0.25%)
  fit function 2: $y = sin(5\pi x)/5\pi x$



- Train on Actual Tasks:
  - Describe the models you use and the task you chose. (0.5%)

    我們做的是MNIST的手寫數字辨識，我們使用以下三個 DNN model。

    1. 總參數量：134823, Adam, learning rate：0.00001, MSE loss

	layer1	layer2	layer3
in	784	128	247
out	128	247	10

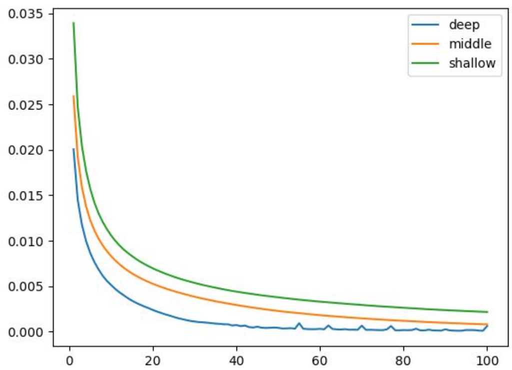
2. 總參數量：134794, Adam, learning rate：0.00001, MSE loss

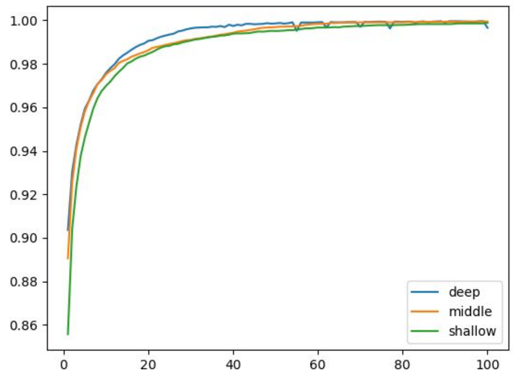
	layer1	layer2	layer3	layer4
in	784	128	128	128
out	128	128	128	10

3. 總參數量：134560, Adam, learning rate：0.00001, MSE loss

	L1	L2	L3	L4	L5	L6	L7	L8
in	784	100	100	100	100	100	100	50
out	100	100	100	100	100	100	50	10

- In one chart, plot the training loss of all models. (0.5%) 見下左圖
- In one chart, plot the training accuracy. (0.5%) 見下右圖





- Comment on your results. (1%)

  我們得到的結果和助教的差不多，較深層的model即使參數量差不多，甚至還略少，但是loss 是比較低的，至於accuracy的部份由淺至深分別是99.88%, 99.95%, 99.93%。

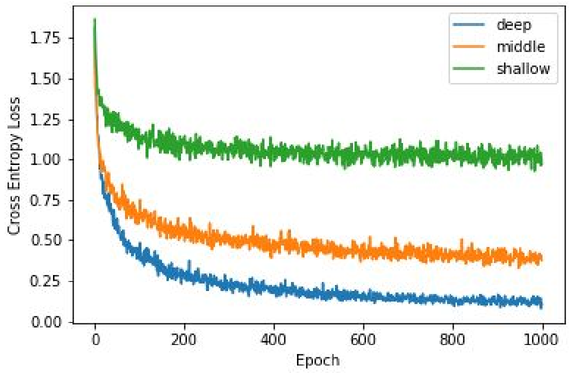
- Use more than two models in all previous questions. (bonus 0.25%)

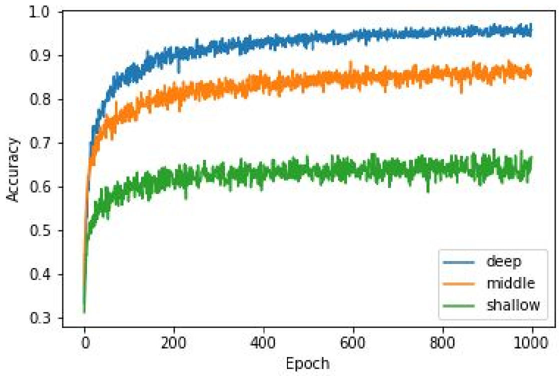
使用三種models如前述。

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

在CIFAR10上面使用相似參數量不同層數CNN做訓練，結果如下圖，可發現同樣參數的條件下，deep 的模型 loss 較低，accuracy 也較高。

	shallow	middle	deep
Conv 層數	1	2	4
FC 層數	2	2	2
parameters 數	173169	172317	173209
訓練	Adam(lr=0.001)，cross entropy loss		



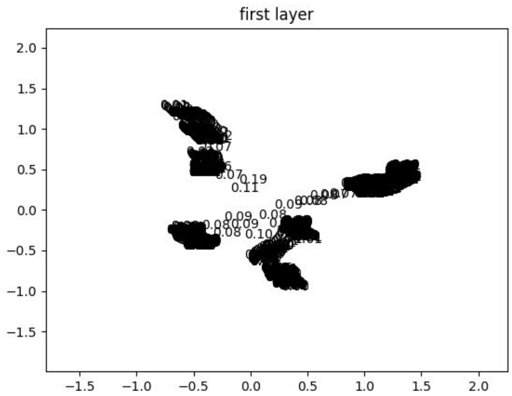


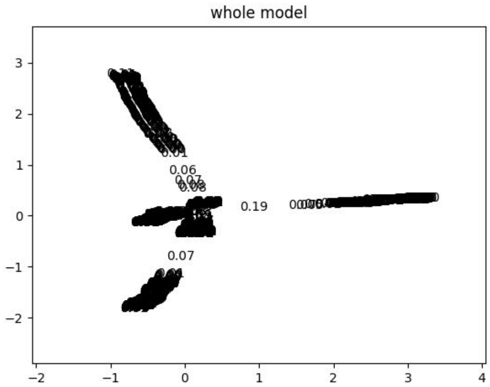
1-2

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

    我們每3個epoch紀錄一次，用 Adam optimizer，lr=0.0001，然後用PCA 降成2維。

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

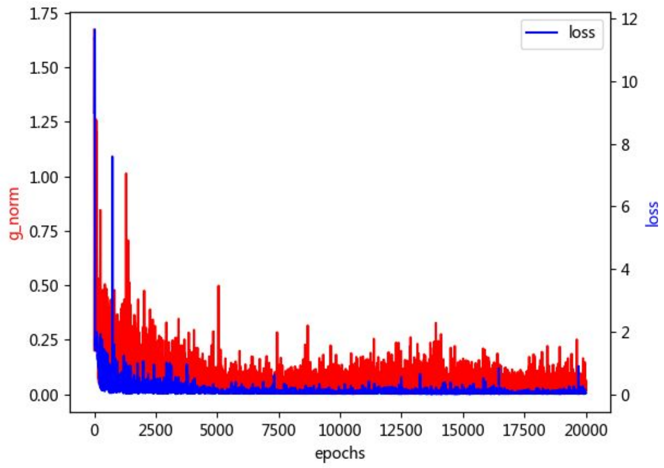




- Comment on your result. (1%)

  由上圖可以發現中間初始的地方loss比較高，隨著training的進行，參數開始往四周移動，loss也跟著降低。每次training參數移動的方向都不同，表示loss的曲線應該是很複雜的，會有不只一個 local minumum。

- Observe gradient norm during training.
  - Plot one figure which contain gradient norm to iterations and the loss to iterations. (1%)



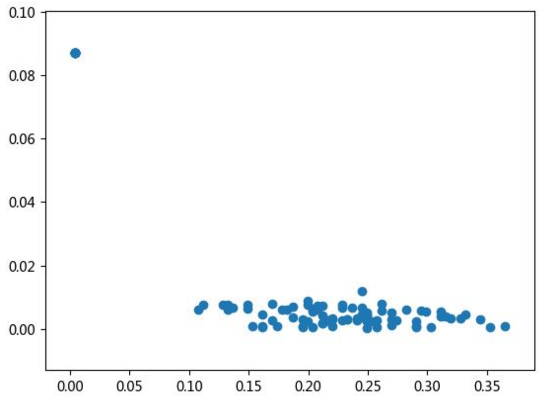
  - Comment your result. (1%)

    由上圖可以發現隨著training的進行，不只loss會下降，gradient norm也有下降的趨勢，不過gradient norm在training前期下降的比較明顯，後期還是有明顯震盪。

- What happens when gradient is almost zero?
  - State how you get the weight which gradient norm is zero and how you define the minimal ratio. (2%)

    先對model進行一般的training之後，最後將training的loss改為gradient的norm繼續做training，train到loss幾乎不再下降為止。Minimal ratio 的算法為分析 hessian matrix 的eigenvalues大於0的比例佔多少而定，總共進行了100次training。

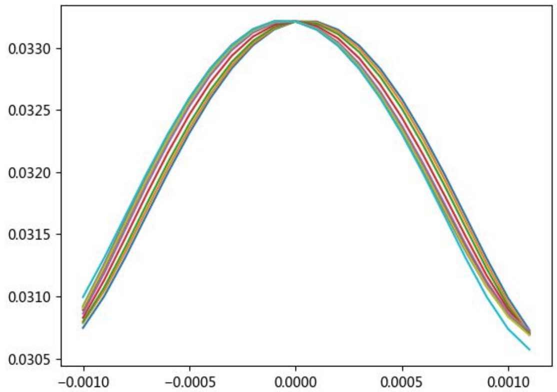
  - Train the model for 100 times. Plot the figure of minimal ratio to the loss. (2%)

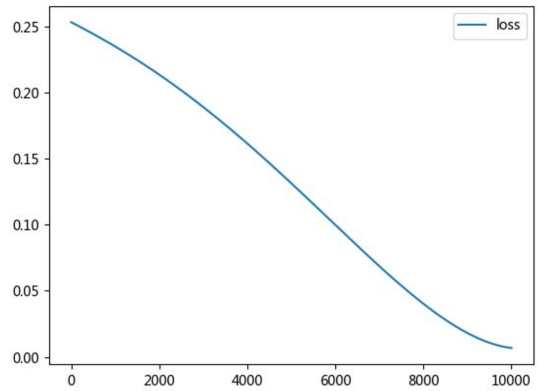


  - Comment your result. (1%)

    因為最後結果loss較高的只有一點，而他的minimal ratio也是最小的,觀察loss趨近於0的點幾乎minimal ratio也都大於0.1以上，感覺結果多少有印證到上課所提的觀點。

- Bonus (1%)
  - Use any method to visualize the error surface.





- Concretely describe your method and comment your result.

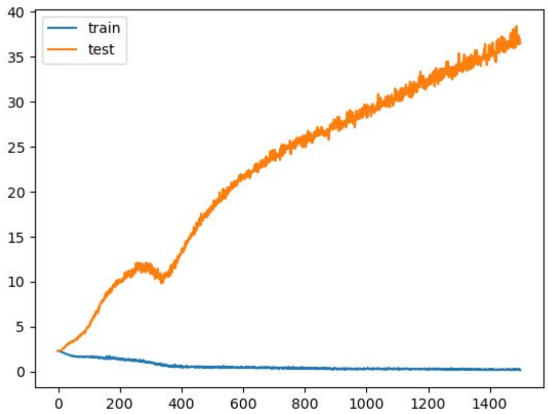
  用了助教ppt bonus example裡面的後面兩個，所用的model都是單純的nn，總參數量241個，並做在fit function上，第一個做的是weight的偏移，縱軸為loss橫軸為偏移量，不過結果是兩邊偏移都會讓loss下降，蠻奇怪的。第二題是利用初始model與最終 model 的weight，在中間線性sample了10000個點，最終結果一樣也有點奇怪，是loss一路平滑的往下降，似乎與example中loss劇烈的震盪有所不同。

1-3

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

    我們做的task是MNIST的手寫數字辨識，使用三層hidden layer各256個units，使用Adam optimizer，learning rate 0.001。

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

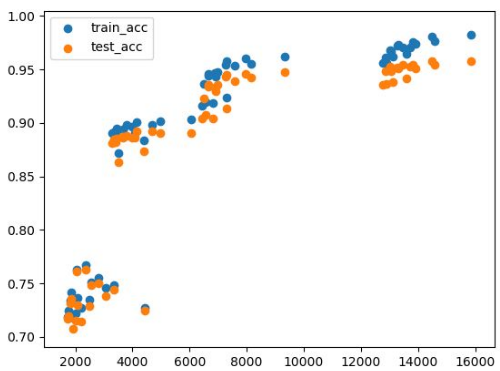


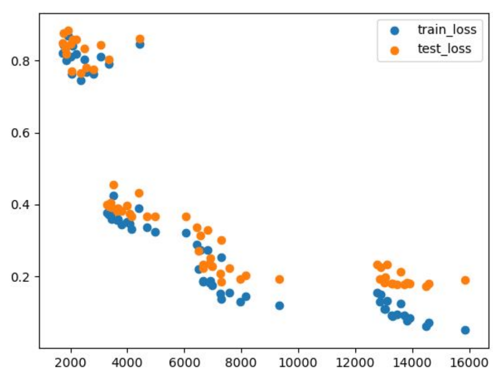
- Number of parameters v.s. Generalization
  - Describe your settings of the experiments. (e.g. which task, the 10 or more structures you choose) (1%)

    在mnist上做實驗，loss使用cross entropy，optimizer用adam，model是一個4層NN的model，每一個hidden layer設定4種units數，排列組合總共有64種model，下表是3個hidden layer的units數：

layer1	2	4	8	16
layer2	4	8	16	32
layer3	8	16	32	64

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





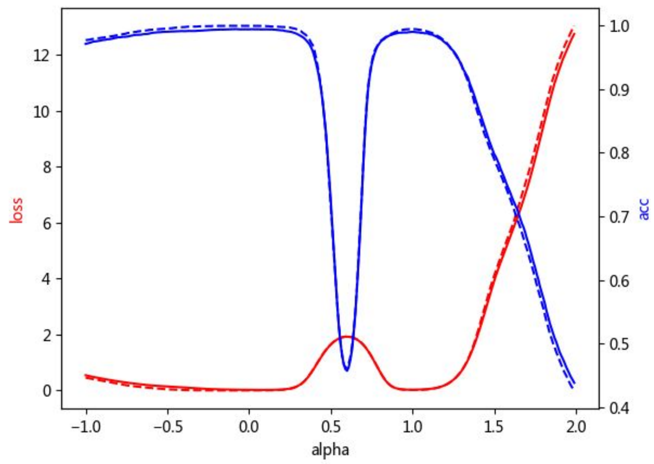
  - Comment your result. (1%)

由實驗結果可以發現，參數量和學習成果有明顯的趨勢，當參數量較多時，model的確學的比較好，不只是training時的表現進步，testing也有明顯的進步，所以參數較多的確generalize的能力比較強。

- Flatness v.s. Generalization
  - Part 1:
    - Describe the settings of the experiments (e.g. which task, what training approaches) (0.5%)

      在mnist上做實驗，loss使用cross entropy,optimizer用adam ，model是一個雙層CNN+雙層DNN，training procedure為train到loss幾乎不在下降為止，兩個不同的batch size 為 1024與 64 ，紀錄alpha由-1到2間loss的改變。

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



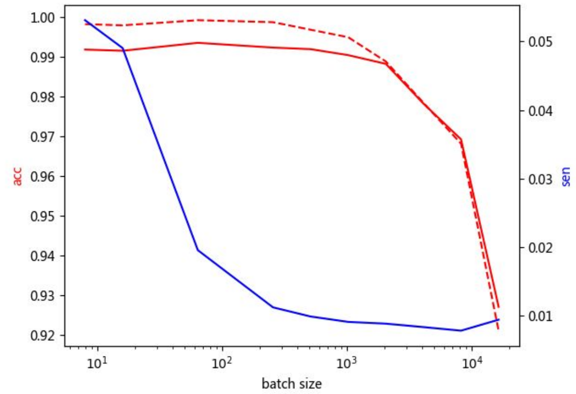
    - Comment your result. (1%)

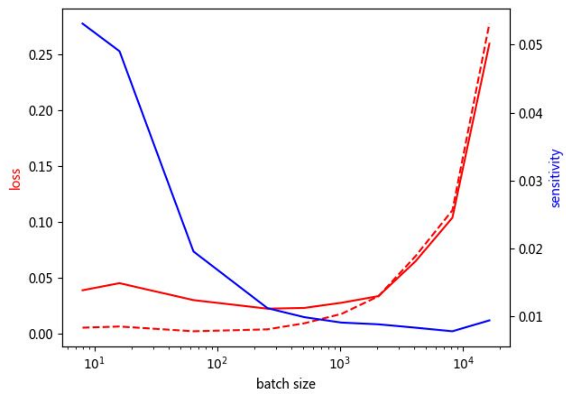
      當alpha為-1到1之間時model的loss基本上與最後的loss無太大差異，但一旦alpha變更大時，loss與acc就明顯的變差了。

  - Part 2 :
    - Describe the settings of the experiments (e.g. which task, what training approaches) (0.5%)

      在mnist上做實驗，loss使用cross entropy,optimizer用adam，model是一個雙層CNN+雙層DNN，training procedure為train 2000個epoch，對一連串不同的batch size進行實驗。

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

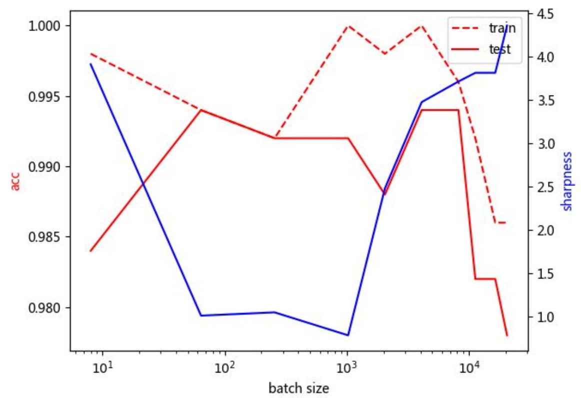


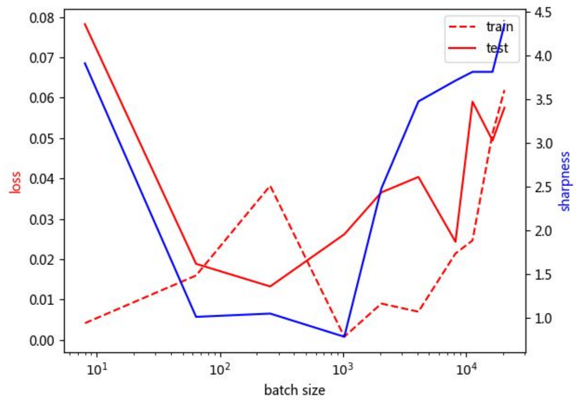


- Comment your result. (1%)

結果顯示出sensitivity對應個batch size是一路往下滑，有點奇怪，最後在超過10000的時候些微上升，如果再大可能就會觀察到batch size過大造成sensitivity變大的狀況了(但gpu也oom了)。

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





這題是以minst dataset最為實驗對象，用了雙層cnn + 雙層dnn，parameters數量大概24000個左右，sharpness的具體算法是記錄最終model對500筆資料的每一個layer的hessian matrix找出eigen value，找出最大eigen value所在的那個layer的hessian matrix做norm(2)，最後觀察到batch size在太大or太小的時候sharpness都會比較高，而相對應的testing與training的結果都沒有比極端的batch size的結果來得好。