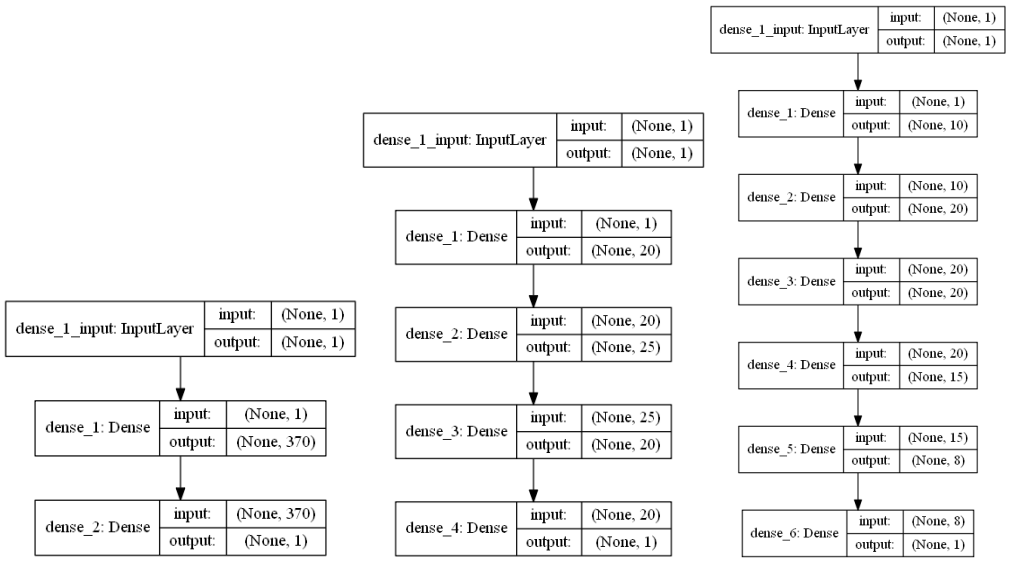
MLDS HW1
洪子庭 r06631014 生機碩一 :1-2
陳偉　 r06944043 網媒碩一 :1-3
曾啟軒 r06631004 生機碩一 :1-1,1-2

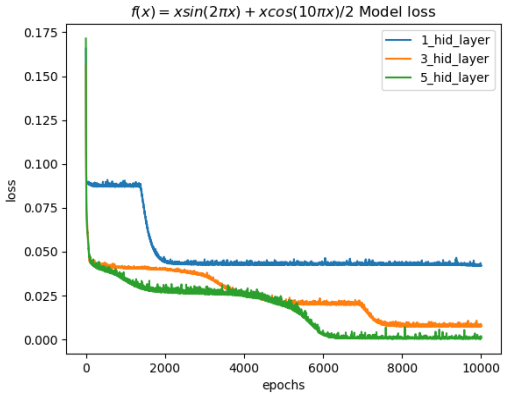
Hw 1-1(曾啟軒)

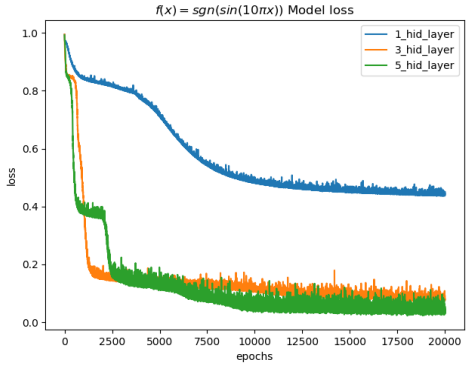
- Simulate a Function:
  - Describe the models you use, including the number of parameters (at least two models) and the function you use. (0.5%)



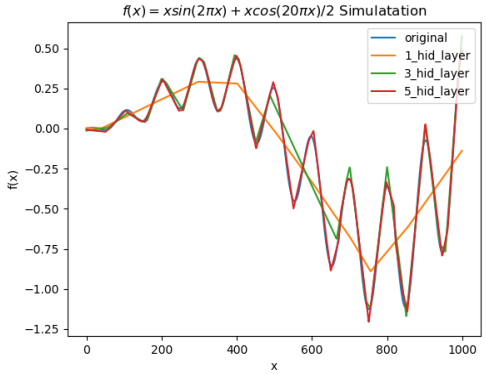
dnn (1hidden) model1 parameters:1111
dnn (3hidden) model2 parameters:1106
dnn (5hidden) model3 parameters:1103
function1: $F(x)=x\sin(2\pi x)+(x\cos(20\pi x))/2$
function2: $F(x)=\mathrm{sgn}(\sin(10\pi x))$

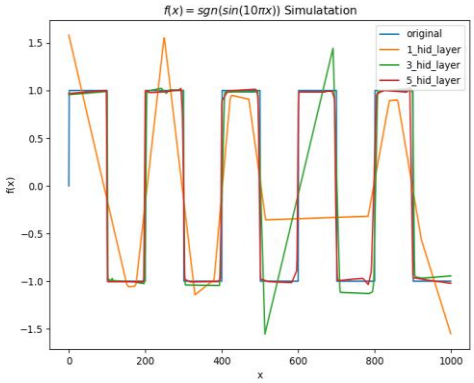
- In one chart, plot the training loss of all models. (0.5%)





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

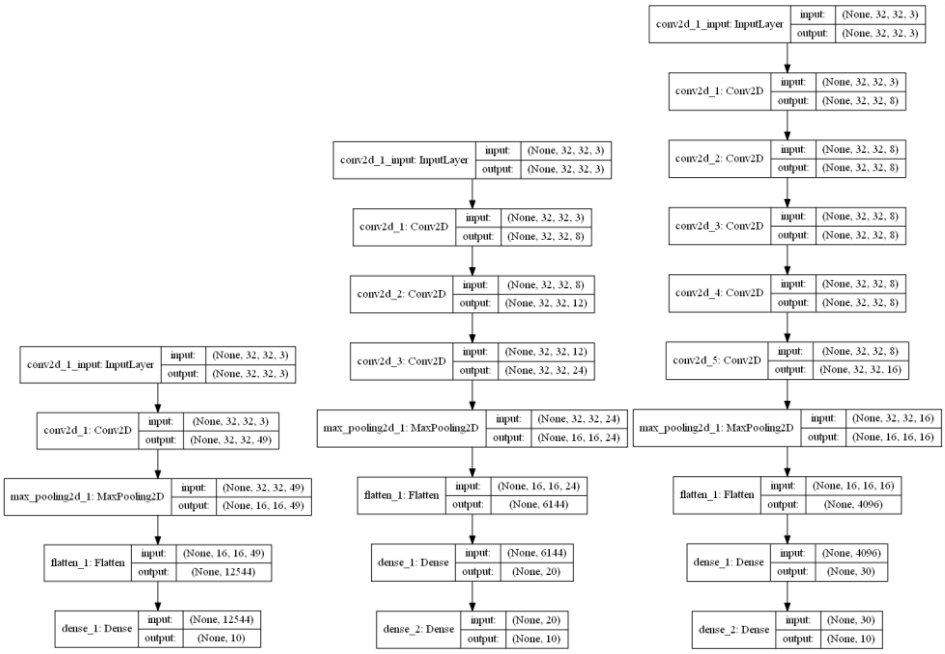




- Comment on your results. (1%)
  由上述結果看出在差不多數量的參數中，5 層的 model 所描繪出的圖形最接近原始圖形且 loss 較低，3 層的 model 也有不錯的效果，單一層的效果較差，觀察兩個 function 中，在 sin 波圖形上可以描繪較方波精準。

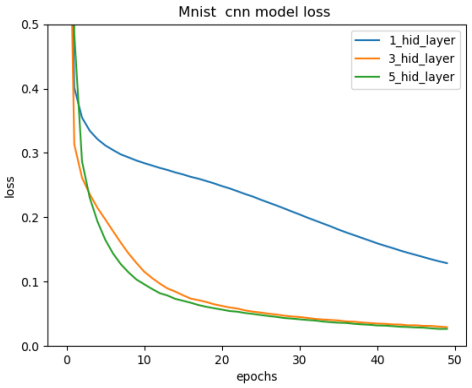
- Use more than two models in all previous questions. (bonus 0.25%)
  均有使用 3 個 model 和兩個 function
- Use more than one function. (bonus 0.25%)
  均有使用 3 個 model 和兩個 function

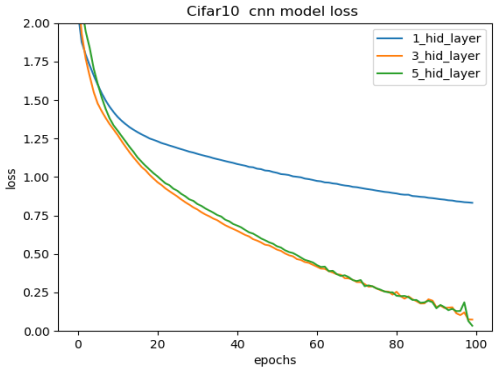
- Train on Actual Tasks:
  - Describe the models you use and the task you chose. (0.5%)



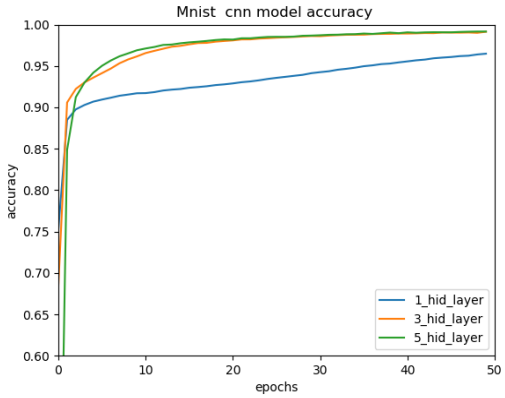
cnn (1hidden) model1 parameters:126822
cnn (3hidden) model2 parameters:126826
cnn (5hidden) model3 parameters:126364
task: mnist, cifar10

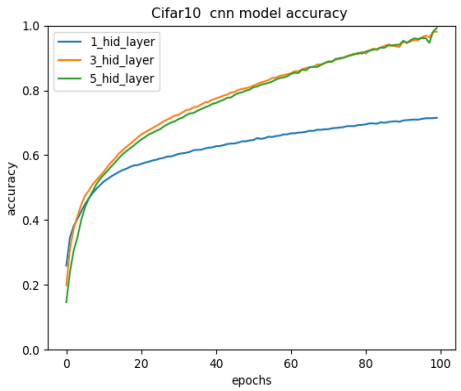
  - In one chart, plot the training loss of all models. (0.5%)





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





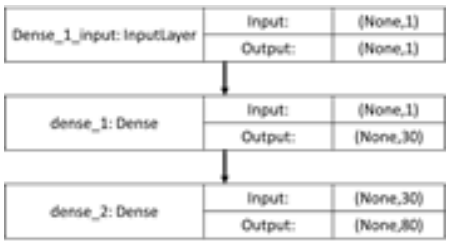
- Comment on your results. (1%)
  由上述結果看出在差不多數量的參數中，5 層的 model 有最高的 accuracy 和最低的 loss，不過 3 層的 model 在 mnist 和 cifar10 中也很接近的成效，單一層的 model 效果就差比較多。

- Use more than two models in all previous questions. (bonus 0.25%)
  均有使用 3 個 model 和 mnist 及 cifar10 兩個 task
- Train on more than one task. (bonus 0.25%)
  均有使用 3 個 model 和 mnist 及 cifar10 兩個 task

HW1-2(曾啟軒、洪子庭)

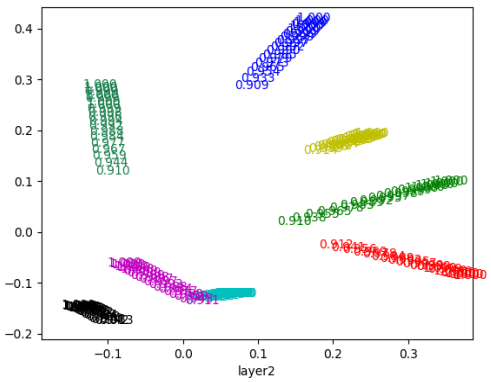
Visualize the optimization process(曾啟軒、洪子庭)

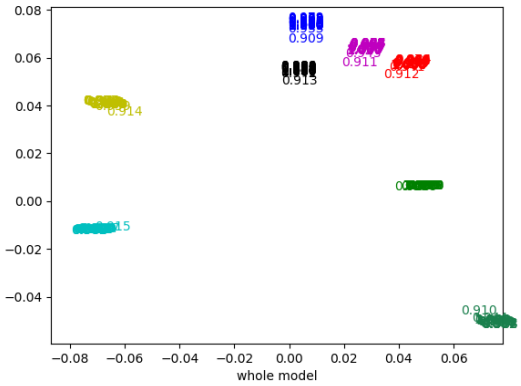
- Describe your experiment settings.



設定 Epoch=60，每 3 個 Epoch 紀錄一次，共 20 筆資料
Optimizer: Adam
利用 SVD(Singular value decomposition)做降維，將 weights 降至 2 維
參數數量:26840

- Train the model for 8 times, selecting the parameters of any one layer and whole model and plot them on the figures separately(曾啟軒、洪子庭)

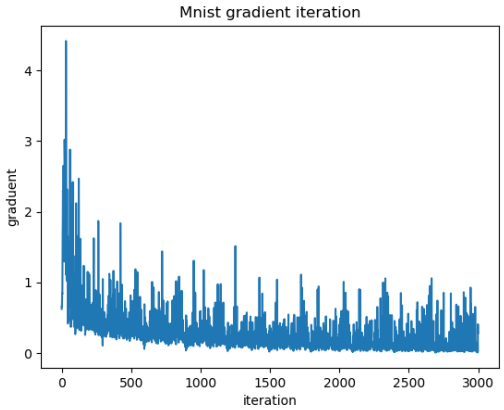


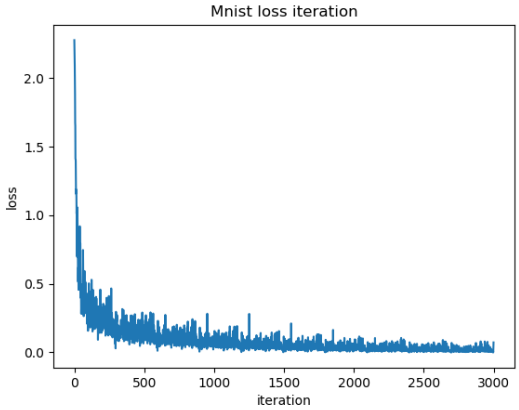


- Comment on your result.

上圖為第二層的 weights 跟整個 model 的 weights 變化趨勢，在訓練過程中，weights 在二維平面上可以看出，會往固定方向做改變，準確率也逐漸提升，從每次迭代的距離會越來越短，可以看出 weights 的變化越來越小，由於單一一層取的是第二層 dense 的參數，所以降為後圖形的趨勢會和 whole model 有所不同。

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





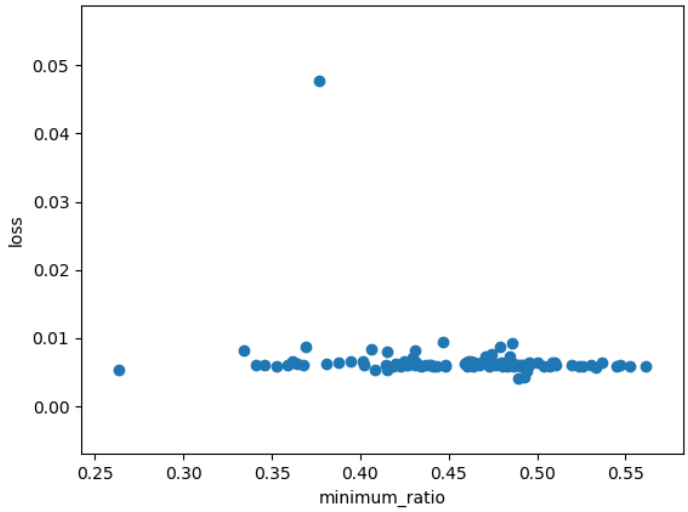
  - Comment your result. (1%)

此次所則用 mnist 作為 training data，總共 iteration 為 3000 次，可以看出 loss 和 gradient 都有逐漸減少的趨勢，但 loss 的震盪小，gradient 的震盪則非常大。

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

選用 F(x)=sin(5πx)/5πx的函數，從 0 到 1 中選取 10000 的點作為 training data，先用 mse 作為 loss function 訓練 12000 iteration(120 epochs)，在用 gradient norm 做為新的 loss function 訓練讓 gradient norm 繼續降低，當 gradient norm 小於 0.005 時停止，將此時的 weight 取出作 hessian，將 hessian matrix 計算 eigenvalue，將所有大於 0 的 eigenvalue 比例做為 minimal ratio

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



- Comment your result. (1%)
  在 100 次的 model training 中記錄其 gradient norm 小於 0.005 時的 minimal ration 值，大部分的 loss 都小於 0.01 且 minimal ration 分布在 0.35-0.55 之間，只有一的點的 loss 偏高及另一個點 minimal ration 值偏小。

- Bonus (1%)
  - Use any method to visualize the error surface.
  - Concretely describe your method and comment your result.
    (暫時不會做)

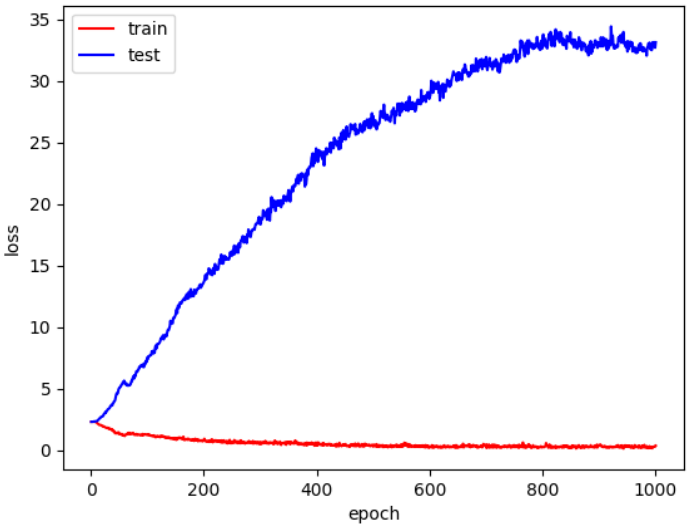
Hw 1.3(陳偉)

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

Answer：experiment on MNIST dataset，learning rate = 0.001，optimizer 為 Adam，input 為完全 random， 四層的 DNN ，28*28 ----> 512 ----> 256 ----> 256 ----> 10。
DNN 可以 fit dataset，越簡單的 dataset，DNN 的 fit 能力越強，即使是 random 的 label 也可以 fit。 我嘗試 DNN fit 在 CIFAR10 上的效果沒有

MNIST 這麼好，而且我發現 CNN 不能 fit random label， 即使是部分 random，越是 深度大參數多的 CNN， 越早 fail 到 average possibility。

- 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%)

Answer：experiment on CIFAR10 dataset， 10 CNN structure。
從 model1 到 model10 參數數目分別是：
[2172, 51666, 138330, 303018, 550570, 668602, 1208138,1671114, 2262922, 5626378]

structure 分別是：

structure 1	structure 2	structure 3
nn.Conv2d(3, 4, 3, 1, 1)	nn.Conv2d(3, 8, 3, 1, 1)	nn.Conv2d(3, 16, 3, 1, 1)
nn.Conv2d(4, 4, 3, 1, 1)	nn.Conv2d(8, 8, 3, 1, 1)	nn.Conv2d(16, 16, 3, 1, 1)
nn.MaxPool2d(2)	nn.MaxPool2d(2)	nn.MaxPool2d(2)
nn.Conv2d(4, 8, 3, 1, 1)	nn.Conv2d(8, 16, 3, 1, 1)	nn.Conv2d(16, 32, 3, 1, 1)
nn.Conv2d(8, 8, 3, 1, 1)	nn.Conv2d(16, 16, 3, 1, 1)	nn.Conv2d(32, 32, 3, 1, 1)
nn.MaxPool2d(2)	nn.MaxPool2d(2)	nn.MaxPool2d(2)
nn.Conv2d(8, 16, 3, 1, 1)	nn.Conv2d(16, 32, 3, 1, 1)	nn.Conv2d(32, 64, 3, 1, 1)
nn.Conv2d(16, 16, 3, 1, 1)	nn.Conv2d(32, 32, 3, 1, 1)	nn.Conv2d(64, 64, 3, 1, 1)
nn.MaxPool2d(2)	nn.MaxPool2d(2)	nn.MaxPool2d(2)
nn.Linear(16*4*4, 64)	nn.Linear(32*4*4, 64)	nn.Linear(64*4*4, 64)
nn.Linear(64, 10)	nn.Linear(64, 10)	nn.Linear(64, 10)

structure 4
```
nn.Conv2d(3, 32, 3, 1, 1)
nn.Conv2d(32, 32, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(32, 64, 3, 1, 1)
nn.Conv2d(64, 64, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(64, 96, 3, 1, 1)
nn.Conv2d(96, 96, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(96*4*4, 64)
nn.Linear(64, 10)
```

structure 5
```
nn.Conv2d(3, 32, 3, 1, 1)
nn.Conv2d(32, 32, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(32, 64, 3, 1, 1)
nn.Conv2d(64, 64, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(64, 128, 3, 1, 1)
nn.Conv2d(128, 128, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(128*4*4, 128)
nn.Linear(128, 10)
```

structure 6
```
nn.Conv2d(3, 48, 3, 1, 1)
nn.Conv2d(48, 48, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(48, 96, 3, 1, 1)
nn.Conv2d(96, 96, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(96, 128, 3, 1, 1)
nn.Conv2d(128, 128, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(128*4*4, 128)
nn.Linear(128, 10)
```

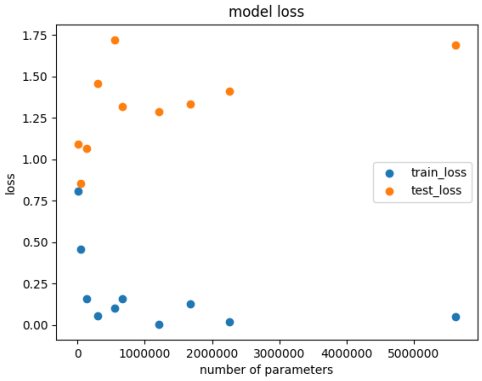
structure 7
```
nn.Conv2d(3, 64, 3, 1, 1)
nn.Conv2d(64, 64, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(64, 128, 3, 1, 1)
nn.Conv2d(128, 128, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(128, 192, 3, 1, 1)
nn.Conv2d(192, 192, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(192*4*4, 128)
nn.Linear(128, 10)
```

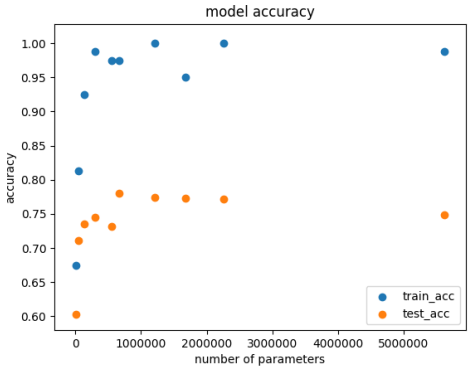
structure 8
```
nn.Conv2d(3, 64, 3, 1, 1)
nn.Conv2d(64, 64, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(64, 128, 3, 1, 1)
nn.Conv2d(128, 128, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(128, 256, 3, 1, 1)
nn.Conv2d(256, 256, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(256*4*4, 128)
nn.Linear(128, 10)
```

structure 9
```
nn.Conv2d(3, 128, 3, 1, 1)
nn.Conv2d(128, 128, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(128, 192, 3, 1, 1)
nn.Conv2d(192, 192, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(192, 256, 3, 1, 1)
nn.Conv2d(256, 256, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(256*4*4, 128)
nn.Linear(128, 10)
```

structure 10
```
nn.Conv2d(3, 128, 3, 1, 1)
nn.Conv2d(128, 128, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(128, 256, 3, 1, 1)
nn.Conv2d(256, 256, 3, 1, 1)
nn.MaxPool2d(2)
nn.Conv2d(256, 512, 3, 1, 1)
nn.Conv2d(512, 512, 3, 1, 1)
nn.MaxPool2d(2)
nn.Linear(512*4*4, 128)
nn.Linear(128, 10)
```

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





- Comment your result. (1%)

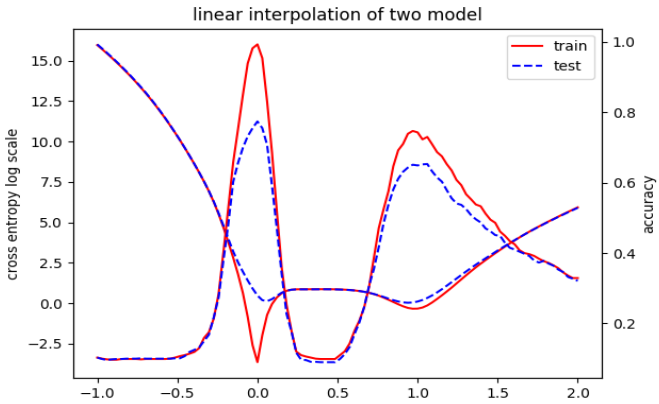
可以看出 model 的 structure 相同的情況下，隨著 parameters 數目增加，training accuracy 和 testing accuracy 都隨之增加，上升趨勢逐漸趨於平緩，得到結論，parameter 數目增加，可以提高 accuracy 的同時，也有更好的 generelization。
loss 上的表現為，training loss 隨 parameters 增加逐漸下降，和 accuracy 的觀察結果一致， 但 testing loss 卻有些反常。

- Flatness v.s. Generalization
  - Part 1:
    - Describe the settings of the experiments (e.g. which task,

what training approaches) (0.5%)

Answer：experiment on CIFAR10 dataset，通過改變 Learnig rate 分別為 0.005， 0.001，theta_a = a * theta_1 + (1-a) * theta_2。theta_1 由 LR = 0.001 的 training process 得到，theta_2 由 LR = 0.005 的 training process 得到。

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



- Comment your result. (1%)

（1）不論是 LR = 0.001 或 0.005，trainning 都達到了 local minimum。

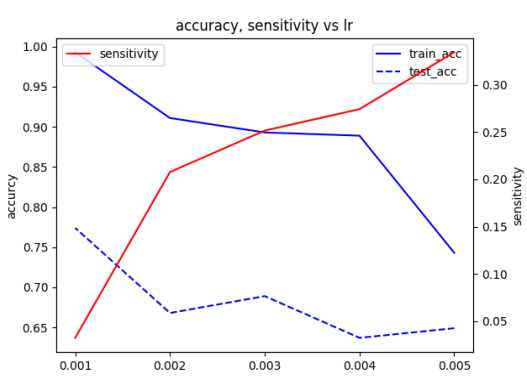
（2）LR = 0.001 比 LR = 0.005 trainning 得到 error surface 更低的點

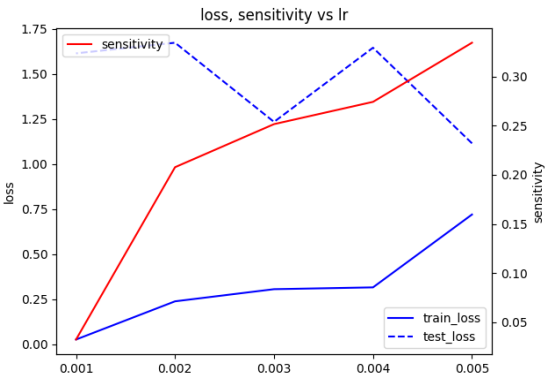
（3）model 之間的 parameter 的 interpolation 的 error surface 都比 原由 gradient 得到的 model perfarmence 差。

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

Answer：experiment on CIFAR10 dataset，通過改變 Learnig rate 分別為 [0.005, 0.004, 0.003, 0.002, 0.001]

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





- Comment your result. (1%)

隨 Learning rate 增加， model performance 變差，train accuracy 和 test accuracy 變小，training loss 變大，testing loss 變化不明顯(理應變大)，sensitivity 變大。 一定程度上可以通過觀察 sensitivity 看出 model 的 generalization 怎麼樣。

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

(暫時還不會做。。。。)