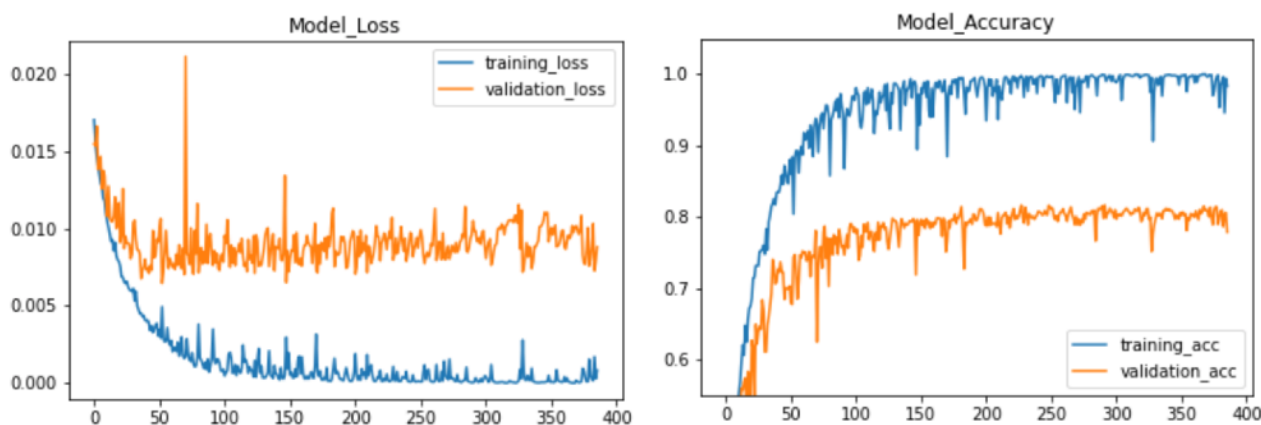


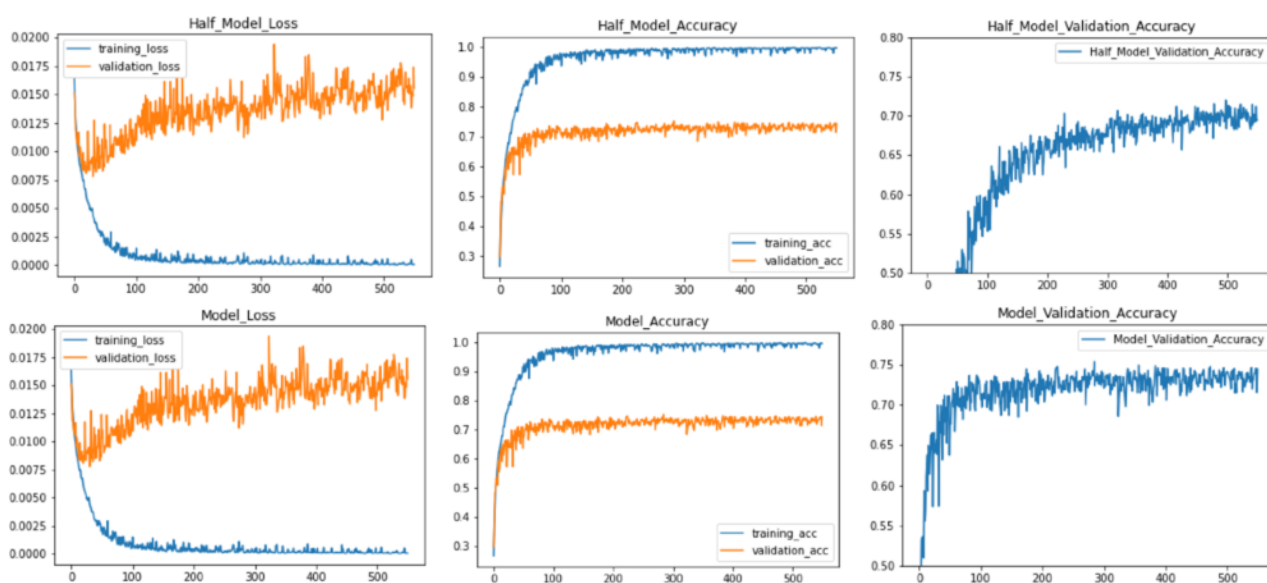
學號：R08945034 系級：生醫電資甲碩一 姓名：劉政倫

- 一. 請說明你實作的 CNN 模型(best model)，其模型架構、訓練參數量和準確率為何？(1%)
- 二. 請實作與第一題接近的參數量，但 CNN 深度（CNN 層數）減半的模型，並說明其模型架構、訓練參數量和準確率為何？(1%)
- 三. 請實作與第一題接近的參數量，簡單的 DNN 模型，同時也說明其模型架構、訓練參數和準確率為何？(1%)
- 四. 請說明由 1 ~ 3 題的實驗中你觀察到了什麼？(1%)
- 五. 請嘗試 data normalization 及 data augmentation，說明實作方法並且說明實行前後對準確率有什麼樣的影響？(1%)
- 六. 觀察答錯的圖片中，哪些 class 彼此間容易用混？[繪出 confusion matrix 分析](1%)

- 一. 我實作的CNN模型(best model)有參考VGG16與VGG19，將整體架構加深，在最後的fully connected層layer的neuron數則縮小，因為傳統VGG架構訓練的類別有一千多個，而我們的食物類別只有11個，所以我將最後fully connected層改為第一層 $512 \times 7 \times 7$ 到1024，第二層1024到512，最後一層則是512到11，訓練參數量的話convolution那部分與VGG19一樣，fully connected的話則是 $512 \times 7 \times 7$ (flatten)到1024、1024到512、512到11，我還試過許多模型架構，試過的經驗寫在第四題。



二. 由於最終版的cnn模型過深要訓練的時間太久，所以我是使用助教colab的範例程式稍微修改之後，再用這個操為修改後的模型減半去做比較，而我修改的地方就是把colab範例程式加data normalization與dropout，learning rate調為0.0001，兩者都訓練1000個epoch後丟到kaggle，減半前accuracy=0.82，減半後卻只剩下0.74左右，而相關的accuracy與loss圖如下，下圖最右邊那直行是因為y軸刻度較明顯，所以能更清楚地看出減半模型的validation accuracy穩定的地方較低，注意顏色的部分第一二直行的圖藍色代表training，但第三直行的圖藍線代表training。



三. DNN模型就是移除前面的convolution layer與maxpooling layer等等，而只留下fully connected，也就是完全不管圖片相鄰pixel間的相關性，一開始就把圖片做flatten，與參數相當的cnn架構相比，準確度確實降低很多。而這也不意外，畢竟cnn做過convolution，可得到相鄰pixel間關聯，

四. 我發現做影像辨識時，要做convolution(擷取圖片相鄰pixel間擁有的特徵)和maxpooling(downsampling)，才能更有效辨識出正確物體，而且convolution的kernel還能分享參數，沒有影像這些特性才能用DNN這種一開始就直接flatten成一堆參數的，而且在accuracy部分只要model不要太爛通常training accuracy都能衝到很高，但training accuracy高不代表好，要validation accuracy高才能代表我們應用在未知的data上，能正確辨識我們需要的東西。然後optimizer我選用的是對於影像辨識效果較好的sgdm，實驗結果也是比Adam來得優秀。再來，如果把model減半，我們發現辨識能力會差一些，所以才會一直說"deep" learning，

但是如果太deep，可能會發生gradient vanishing的問題，而導致辨識正確率不增反減，雖然說activation function從sigmoid等等非線性函數改為線性的relu已經看似解決，但是由於relu在橫軸小於零的部分斜率還是零，只要斜率為零的部分一多，neuron就會有很多處於無法作用的狀態，也就是深層網路用relu卻還是會造成的Gradient Vanishing，所以才有leaky relu或是maxout等等方法出現！

五. data normalization 及 data augmentation都可以在import torchvision.transforms as transforms的transforms.Compose內加入，data normalization會使圖片擁有mean=0, variance=1的tensor，以老師上課的投影片為例，才不會讓標靶變成橢圓標靶，loss的更新才不會忽陡忽緩的。而 data augmentation 就是將原本的圖片資料作隨機翻轉與隨機旋轉，除了助教幫我們做好的水平翻轉，我們也可以另外再加上垂直的，而旋轉角度也可以由助教的15度延伸至45度，這樣就可以把360度都cover到了！這麼做可以避免使training結果overfitting，不做data augmentation的話，可能testing data轉個向辨識準確率就下降，這不是我們想要的，實驗結果也顯示，確實不做data normalization 及 data augmentation的話，validation的準確率會比沒做的穩在低一些的位置，雖然training accuracy還是可以穩在九十多趴，但training accuracy比較不是我們要看的重點，因為有overfitting的可能性存在！

六. 照我用人眼看的去辨識大概可看出的類別是：(0)看不出 (1)奶 (2)甜點 (3)蛋 (4)看不出 (5)肉 (6)麵 (7)飯 (8)海鮮 (9)湯 (10)蔬果。confusion matrix的意思就是，可以看出{真真,真假,假真,假假}集合的東西，每一橫列相加總數代表該類別總data數，橫列代表類別去做分類的結果會在該對應的直行+=1，最後可算出機率，也可看出哪些類別容易搞混，可以以此為依據自行實做decision tree之類方式去改善結果！而我得到的confusion matrix如下，可看出第6,9的真實辨識正確率較高，第1類別的data容易辨認成第二類別的，第五類別的data容易辨認成第三類別的，諸如此類的資訊都可從圖上得知！

