

學號：B04901069 系級：電機四 姓名：林志皓

1. (2%) 請說明你實作的 **CNN model**，其模型架構、訓練參數和準確率為何？並請用與上述 **CNN** 接近的參數量，實做簡單的 **DNN model**，同時也說明其模型架構、訓練參數和準確率為何？並說明你觀察到了什麼？
(Collaborators: none)

答：

CNN model：

我使用的是近似於VGG的架構，自己有進行一些微調。總共使用了16層的convolution layer, 4 層max-pooling layer, 3層full-connected layer，都有使用batch normalization, drop-out. 訓練參數總共是21116615個，在test set上的準確率大約是0.68。

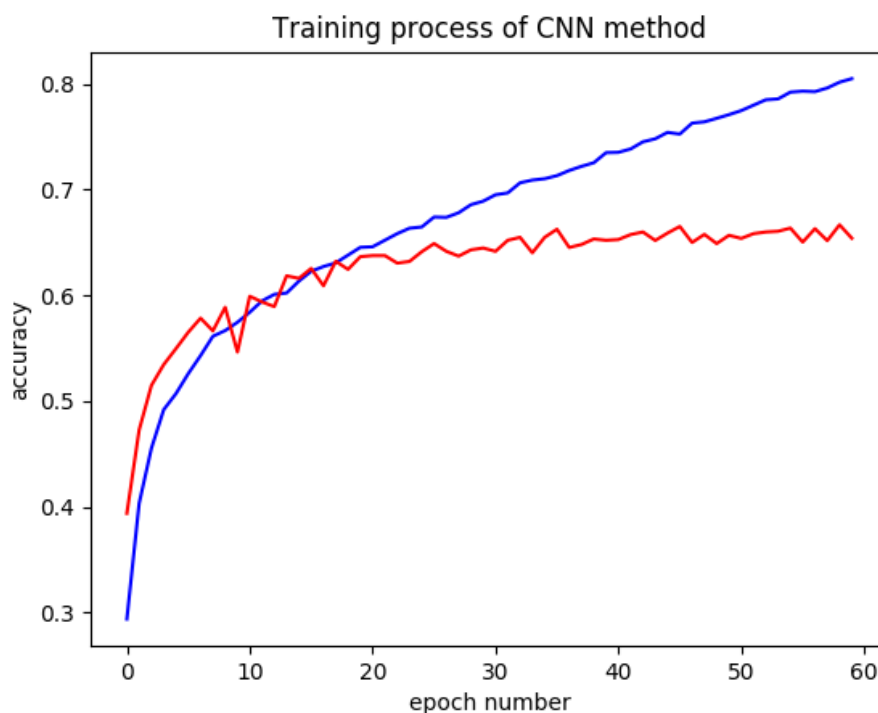
DNN model: 我使用21層的full-connected layer

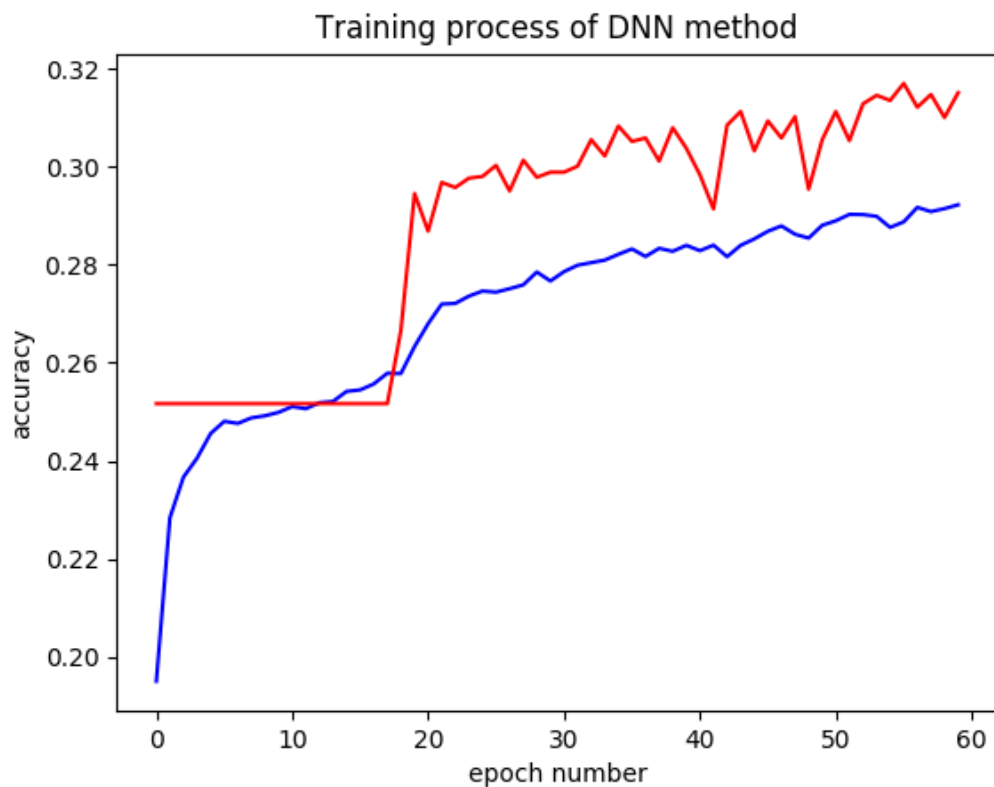
來搭建模型，總共的參數量是21282183個，.準確率大概在0.3左右。

觀察：兩個參數數量相近的模型，準確率相差了一倍，可以發現在圖像識別的任務上，convolution operation能更加有效率的獲取特徵資訊，使的表現優異。

2. (1%) 承上題，請分別畫出這兩個 **model** 的訓練過程 (i.e., loss/accuracy v.s. epoch)
(Collaborators: none)

答：圖中藍線表示 **training set** 上的準確率;紅線表示 **validation set** 上的準確率





3. (1%) 請嘗試 **data normalization**, **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？

(Collaborators: none)

答：

normalization 部份，我將所有值scale到0~1之間。

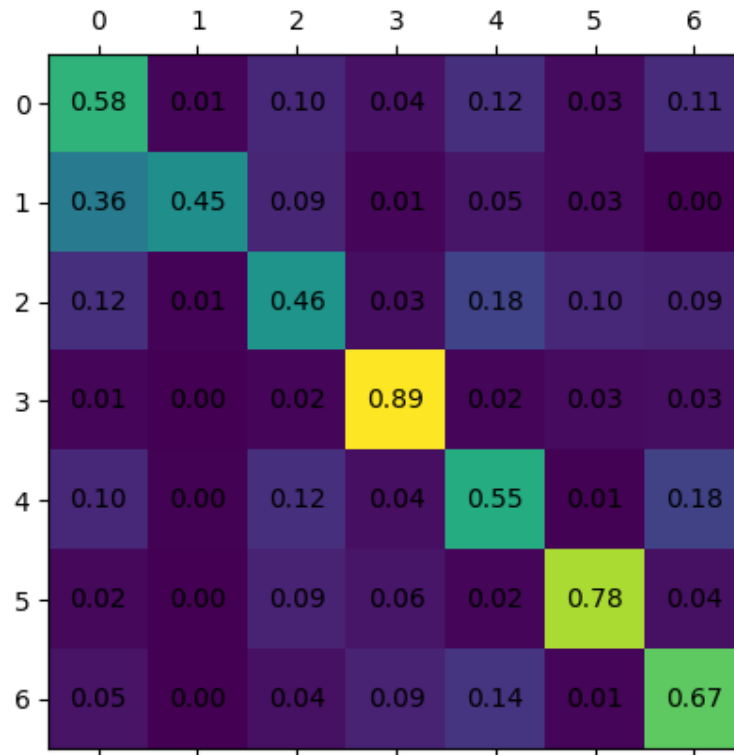
Augmentation部份，我在訓練時隨機將圖片擷取出44*44作為輸入，也隨機水平翻轉，隨機旋轉正負10度，隨機放大縮小0.9~1.1倍，如此一來，就可以有效的增加**training data** 的變化性，讓模型在訓練的時候不會輕易over-fitting。

在實施以上技巧之前，模型在**training set**上的準確率可以輕易達到99%的準確率，就算模型相當的單純，然而在**validation set**上的準確率大概就只能接近50%。

使用了**normalization**, **data augmentation**之後，有效降低over-fitting的狀況，在**training set**上的準確率下降到73%，但**validation set**上的準確率卻上升到65%以上，效果十分不錯。

4. (1%) 觀察答錯的圖片中，哪些 **class** 彼此間容易用混？[繪出 **confusion matrix** 分析]
(Collaborators: none)

答：



圖中所示，每一列代表一個類別，被模型分類到各個類別的比率。(在validation set 上) 在圖中可以清楚看到，類別1(厭惡)當中有0.45被分類正確，也同時有0.36被分類到類別0(生氣)，但”生氣”的類別卻不太會被分類到”厭惡”，個人推測原因，除了因為這兩個類別本身就相近之外，也可能是因為資料的數量不平均，在train.csv中，有3995張”生氣”，卻只有436張”厭惡”，十分懸殊的比例，也可能造成這樣的結果。