

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？

CNN model 架構如下：

- 4 層 Convolution Layer，每一層在 convolution 後都有 LeakyReLU、Batch Normalization 以及 Max Pooling。
- 3 層 Fully Connected Layer，前 2 層均有 ReLU 以及 Batch Normalization；最後一層則是以 softmax 輸出最後結果。

訓練時前 4 層使用 0.25-0.35 的 Dropout，第 5、6 層則是加重到 0.5。

Optimizer 的部分選用 Adam (learning rate=0.0005)，並搭配 data normalization 及 augmentation，每次訓練約 150-250 個 epoch。最後我訓練了 3 個架構相同的 model 並 ensemble，在 public 的 accuracy 為 0.69685，private 則是 0.68041。

DNN model 架構如下：

- 7 層 Fully Connected Layer，前 6 層均有 ReLU 及 Batch Normalization；最後一層則是以 softmax 輸出最後結果。
- Dimension：

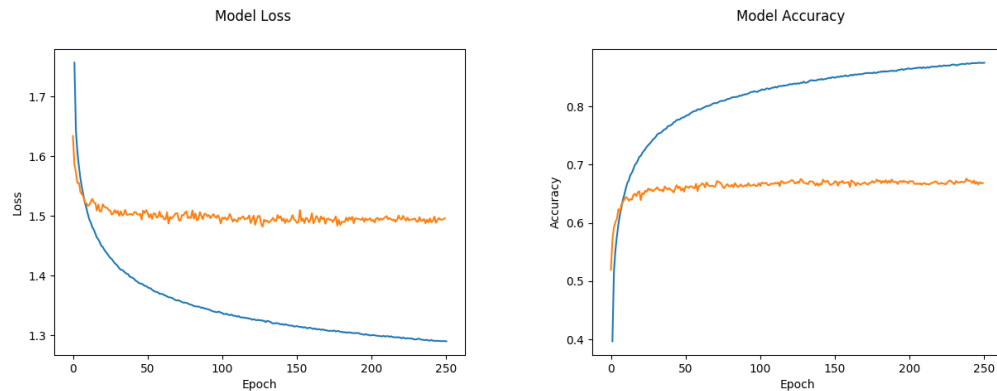
$2304 \rightarrow 1024 \rightarrow 2048 \rightarrow 512 \rightarrow 256 \rightarrow 64 \rightarrow 64 \rightarrow 7$

訓練時除了最後一層外都有使用 0.5 的 Dropout，其餘細節與 CNN 相同。最後在 validation set 上的 accuracy 為 0.49249，而在 public 及 private 則分別是 0.49400 和 0.48509。

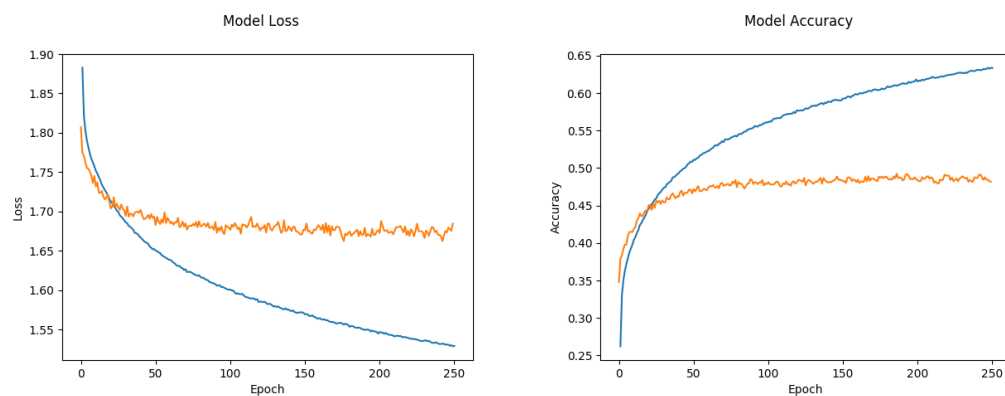
由這兩者比較可以發現，CNN 比起 DNN 更能夠抓到圖片的特徵，在影像方面的 task 有很大的優勢，但是相對地其訓練過程也會因為 convolution 而需要較多的時間。

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)

CNN model



DNN model



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？

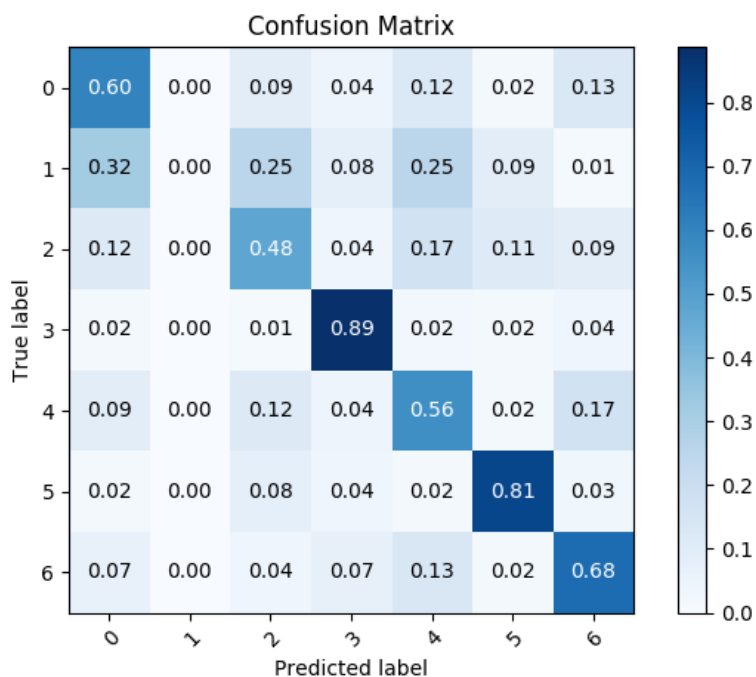
Data Normalization：針對每一張圖片，減去其平均並除以標準差。

Data Augmentation：使用 numpy 以及 scipy 將 training data 水平翻轉、上下左右平移、順/逆時針旋轉及縮放，最後可得到 9 倍的資料量。

Normalization	Augmentation	validation set	public	private
✗	✓	0.66748	0.66508	0.66898
✓	✗	0.62068	0.61883	0.61186
✗	✗	0.61718	0.61047	0.61326

由表格可以發現，Data Augmentation 可以相當程度上的提升 model 的表現；Data Normalization 在這次的作業中幫助並沒有很大，但若要將 model 應用在更 general 的其他 data 上的話也是必要的。

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]



由 confusion matrix 可知，class 3 的準確率最高，而 class 1 的資料完全被判斷為其他 class。關於 class 1 的奇特現象，推測是因為其照片量本來就比其他 class 少，加上訓練過程將資料 shuffle 後分割 train/valid set 時資料分布不均勻的緣故，導致 model 學會「忽略」class 1 的資料。

除了 class 1 之外，class 2 的資料較容易被判斷成其他 class，其中 class 2 及 class 4 最容易造成 model 混淆。