

1. 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%)

模型架構：

Layer (type)	Output Shape	Param #
=====		
conv2d_1 (Conv2D)	(None, 128, 128, 64)	1792
batch_normalization_1 (Batch Normalization)	(None, 128, 128, 64)	256
max_pooling2d_1 (MaxPooling2D)	(None, 64, 64, 64)	0
conv2d_2 (Conv2D)	(None, 64, 64, 128)	73856
batch_normalization_2 (Batch Normalization)	(None, 64, 64, 128)	512
conv2d_3 (Conv2D)	(None, 64, 64, 128)	147584
batch_normalization_3 (Batch Normalization)	(None, 64, 64, 128)	512
max_pooling2d_2 (MaxPooling2D)	(None, 32, 32, 128)	0
conv2d_4 (Conv2D)	(None, 32, 32, 256)	295168
batch_normalization_4 (Batch Normalization)	(None, 32, 32, 256)	1024
conv2d_5 (Conv2D)	(None, 32, 32, 256)	590080
batch_normalization_5 (Batch Normalization)	(None, 32, 32, 256)	1024
max_pooling2d_3 (MaxPooling2D)	(None, 16, 16, 256)	0
conv2d_6 (Conv2D)	(None, 16, 16, 512)	1180160
batch_normalization_6 (Batch Normalization)	(None, 16, 16, 512)	2048
conv2d_7 (Conv2D)	(None, 16, 16, 512)	2359808
batch_normalization_7 (Batch Normalization)	(None, 16, 16, 512)	2048

max_pooling2d_4 (MaxPooling2 (None, 8, 8, 512) 0

dropout_1 (Dropout) (None, 8, 8, 512) 0

flatten_1 (Flatten) (None, 32768) 0

dense_1 (Dense) (None, 1024) 33555456

dense_2 (Dense) (None, 512) 524800

dropout_2 (Dropout) (None, 512) 0

dense_3 (Dense) (None, 11) 5643

=====

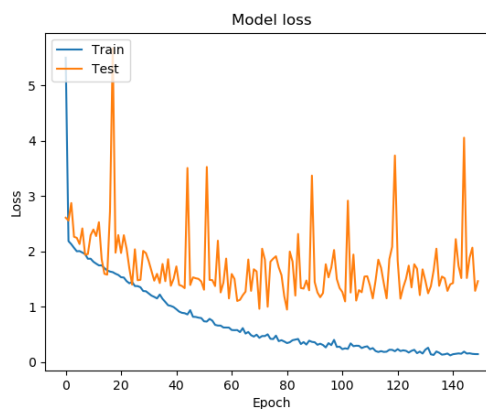
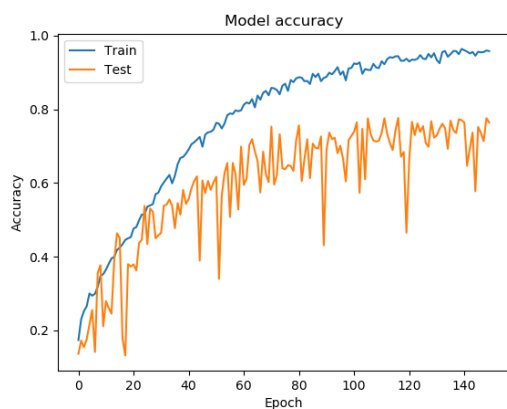
訓練參數量：

Total params: 38,741,771

Trainable params: 38,738,059

Non-trainable params: 3,712

準確率：0.80693 on Kaggle public leaderboard，0.78 on validation set



2. 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%)

模型架構：只留四層 convolution，每層都 max pooling 與 batch normalize，一層全聯接層，然後 softmax 輸出結果

訓練參數量：35490827

準確率：0.85 on testing set 0.56 on validation set

3. 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%)

模型架構：三層全聯接層，每層都 batch normalize，然後 softmax 輸出結果

訓練參數量：36080755

準確率：0.32 on testing set 0.35 on validation set

4. 請說明由 1 ~ 3 題的實驗中你觀察到了什麼？(1%)

從一、二題的比較，得知 CNN 層數如果過少，可能無法很好的辨識各個特徵，也因此得到的準確率很低。但由於 CNN 本身還是很強，所以在 training set 上的準確率還是蠻高的。

從一、三題的比較，得知 dnn 就算參數跟原先的 model 一樣，在 validation set 上的辨識還是很糟。因為 dnn 是取全部的特徵，包括那些沒必要的，因此在跟 cnn 相同的參數數目上，得到的結果一定比較糟。

5. 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%)

我最後上傳的就是 normalization 及 argumentation 都有做的。

Normalization 是把每個 pixel 都除以 255.0

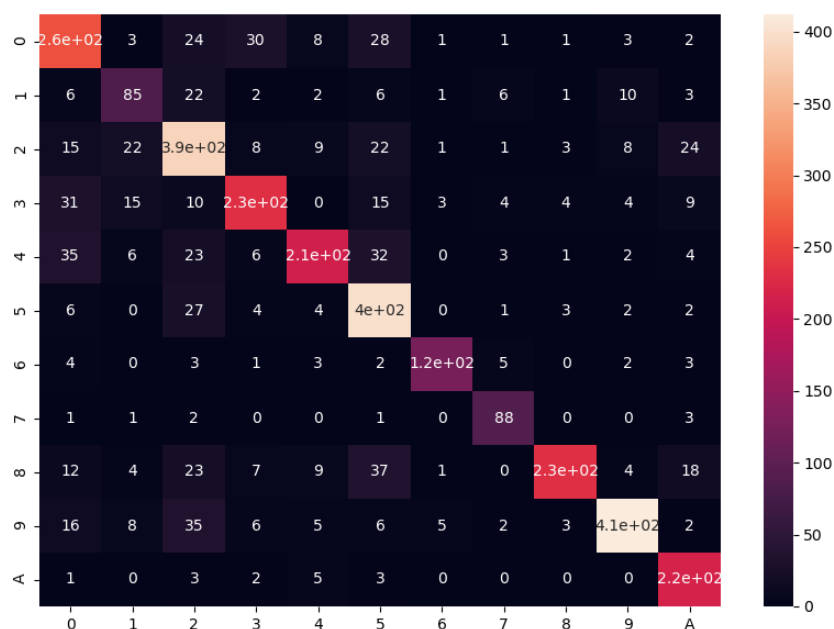
Argumentation 是用 keras 的 image generator 來處理圖片，分別是：旋轉、放大、斜變、平移、左右翻轉。

在沒做 normalization 之前，training accuracy 跟 validation accuracy 都很低；沒做 argumentation 則是 training 很高、validation 很低。

都做之後不只 training accuracy 可以超過 95%，也能有效減低 overfitting 的問題。

6. 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%)

0-麵包 1-乳製品 2-甜點 3-蛋 4-炸物 5-牛肉 6-麵食 7-飯食 8-海鮮 9-湯 A-蔬果



其中「乳製品」跟「甜點」是最容易用混的。