

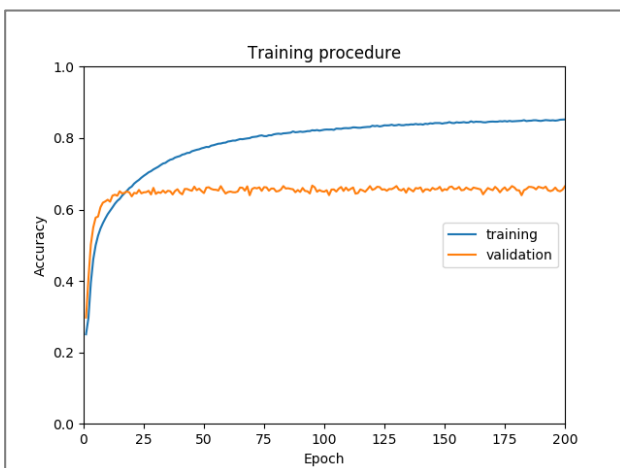
學號：B04901060 系級：電機三 姓名：黃文璵

1. (1%) 請說明你實作的 CNN model，其模型架構、訓練過程和準確率為何？

(Collaborators: 無)

答：

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 48, 48, 64)	640
conv2d_2 (Conv2D)	(None, 48, 48, 64)	36928
max_pooling2d_1 (MaxPooling2)	(None, 24, 24, 64)	0
conv2d_3 (Conv2D)	(None, 24, 24, 64)	36928
conv2d_4 (Conv2D)	(None, 24, 24, 64)	36928
max_pooling2d_2 (MaxPooling2)	(None, 12, 12, 64)	0
conv2d_5 (Conv2D)	(None, 12, 12, 128)	73856
conv2d_6 (Conv2D)	(None, 12, 12, 128)	147584
conv2d_7 (Conv2D)	(None, 12, 12, 128)	147584
max_pooling2d_3 (MaxPooling2)	(None, 6, 6, 128)	0
conv2d_8 (Conv2D)	(None, 6, 6, 256)	295168
conv2d_9 (Conv2D)	(None, 6, 6, 256)	590080
conv2d_10 (Conv2D)	(None, 6, 6, 256)	590080
flatten_1 (Flatten)	(None, 9216)	0
dense_1 (Dense)	(None, 1024)	9438208
dropout_1 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 256)	262400
dropout_2 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 7)	1799
Total params: 11,658,183		
Trainable params: 11,658,183		



CNN 架構基本為：2 至 3 層 conv2d 後接一個 max\_pooling2d（conv2d 含 zero padding）。重複數次直到圖片縮小到一定程度（本模型為 6x6），再接上兩層 fully-connected layer，dropout 皆為 0.5，最後是 7 個類別的 softmax。注意到越深層的 conv2d 有越多 filter。

另外由於 training data（扣除 validation）大約 25000 筆，相對較少，故在訓練時還使用了 Keras 內建的 ImageDataGenerator 來進行 data augmentation，使用了旋轉、平移、縮放和水平翻轉。

接著可以觀察 Training procedure，整體來說，在 validation accuracy 達到收斂後，training accuracy 也逐漸收斂，相較於接下來的 DNN 來說較不容易 overfit。

最後是準確率的部分，根據 Kaggle：

- Public score: 0.67623

- Private score: 0.68375

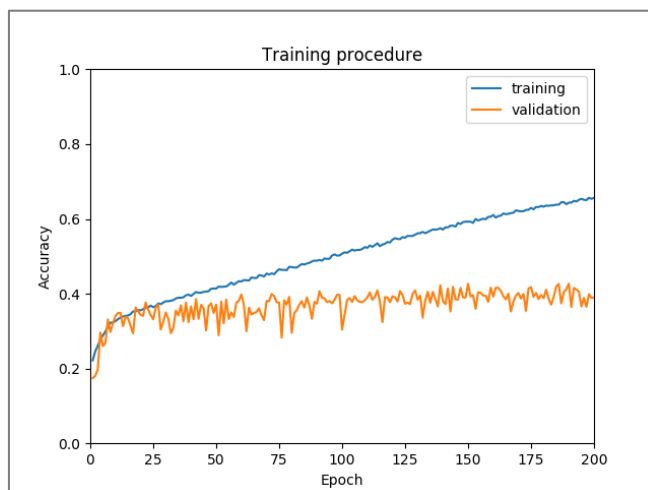
皆有通過 strong baseline。

2. (1%) 承上題，請用與上述 CNN 接近的參數量，實做簡單的 DNN model。其模型架構、訓練過程和準確率為何？試與上題結果做比較，並說明你觀察到了什麼？

(Collaborators: 無)

答：

Layer (type)	Output Shape	Param #
flatten_1 (Flatten)	(None, 2304)	0
dense_1 (Dense)	(None, 2048)	4720640
dropout_1 (Dropout)	(None, 2048)	0
dense_2 (Dense)	(None, 2048)	4196352
dropout_2 (Dropout)	(None, 2048)	0
dense_3 (Dense)	(None, 2048)	4196352
dropout_3 (Dropout)	(None, 2048)	0
dense_4 (Dense)	(None, 7)	14343
Total params: 13,127,687		
Trainable params: 13,127,687		



上題中的 CNN 約有 1200 萬個參數，本題使用接近參數的 DNN。

DNN 架構為三層 2048 個 unit 的 fully-connected layer，分別都有 0.5 的 dropout。最後同樣是 7 個類別的 softmax。

觀察 Training procedure，可以發現 training accuracy 在 200 個 epoch 後仍持續上升，但 validation 早已收斂，此外也可以觀察出 validation accuracy 較為浮動，可以猜測 DNN 相對於 CNN 更容易有 overfitting 的情況。另外注意到 DNN 的 validation 大約收斂在 40%，和 CNN 的 68% 有很大差別。

最後是準確率的部分，根據 Kaggle：

- Public score: 0.38701

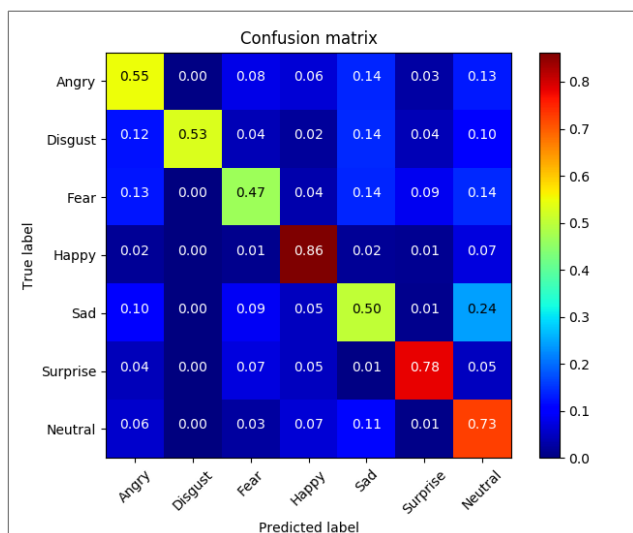
- Private score: 0.40930

比 CNN 差了約 25%。

3. (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: 無)

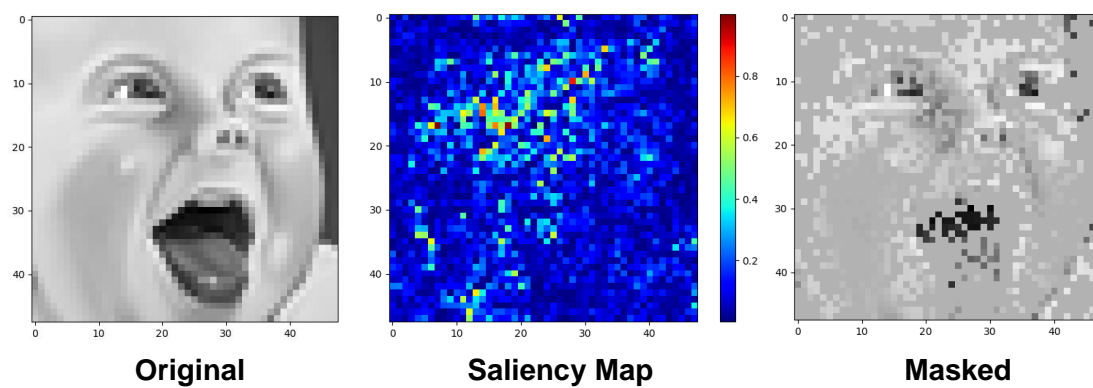
答：



4. (1%) 從(1)(2)可以發現，使用 CNN 的確有些好處，試繪出其 saliency maps，觀察模型在做 classification 時，是 focus 在圖片的哪些部份？

(Collaborators: 無)

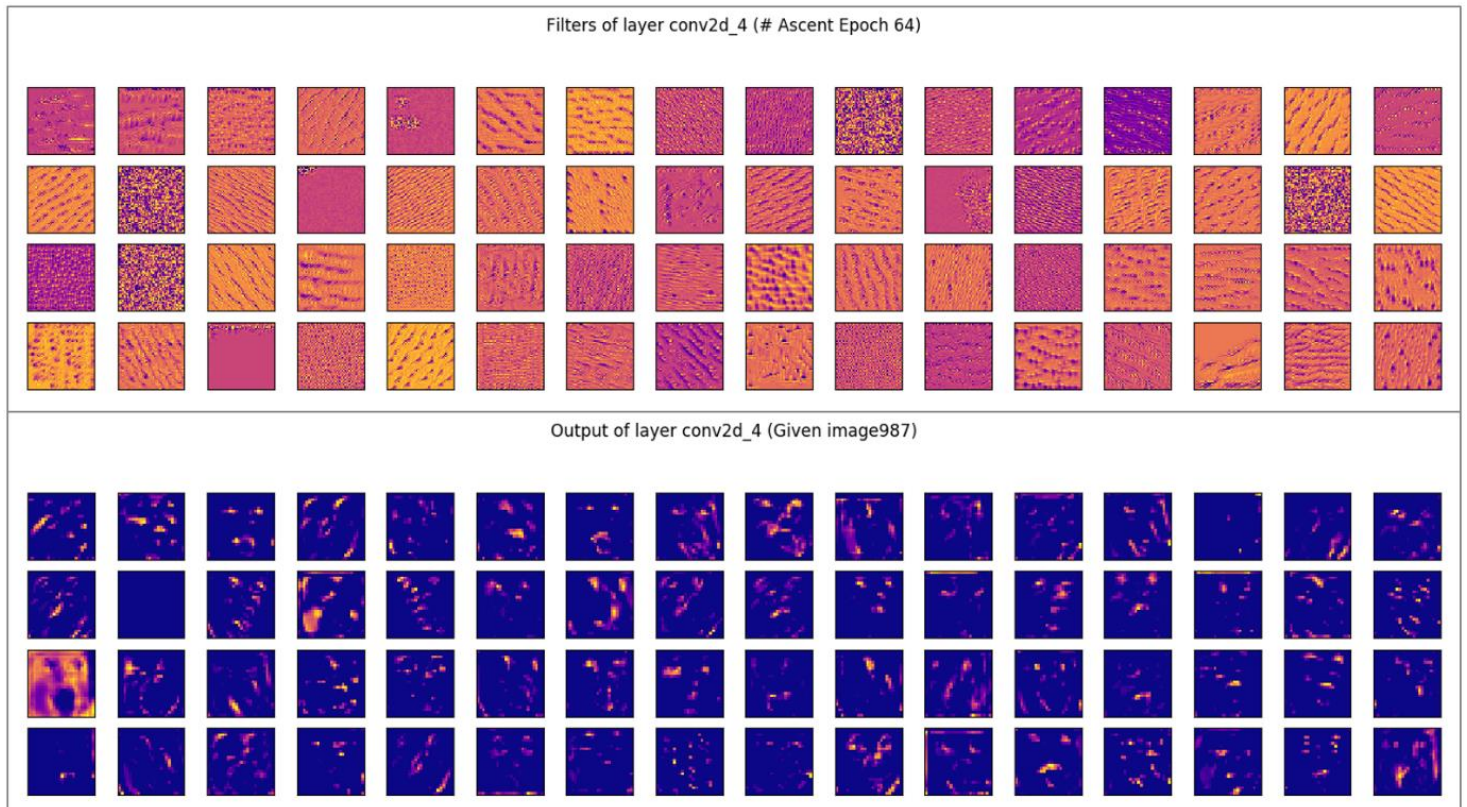
答：



5. (1%) 承(1)(2)，利用上課所提到的 **gradient ascent** 方法，觀察特定層的 **filter** 最容易被哪種圖片 **activate**。

(Collaborators: 無)

答：



取出第四層 conv2d 的其中 64 個 filter 和輸出來觀察，測試圖片和第 4 題使用的相同。