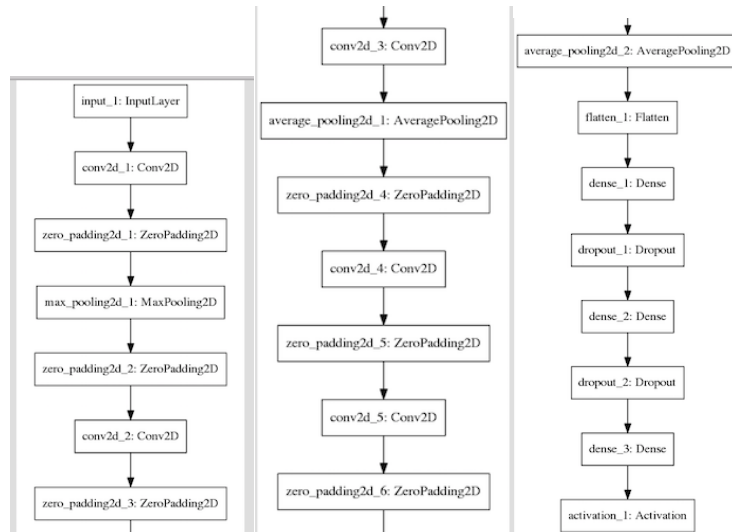


學號：R06922113 系級：資工所碩一 姓名：陳宣伯~

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？

✧ 模型架構：

為求排版方便以下圖順序為由左而右，由上而下。



✧ 訓練過程與辨識率：

■ 資料前處理

◆ 標準化

訓練資料中的圖都經過標準化(normalize)，使每張圖的 pixel 的值都介於-1 到 1 之間

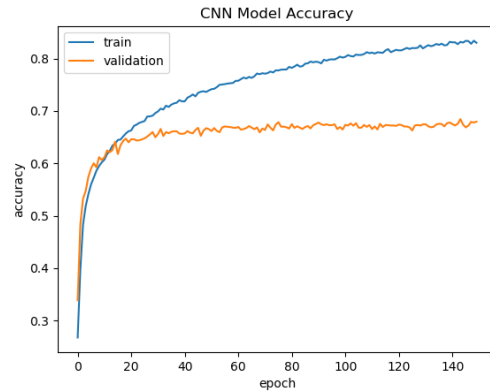
◆ ImageDataGenerator

再將前述經標準化的訓練資料，丟進 keras 提供的 ImageDataGenerator 去產生旋轉(rotation)、翻轉(flip)等等的新的資料，詳細參數如以下：

```
def gen_imgGenerator(X_train, Y_train, batch_size):
    train_datagen = ImageDataGenerator(
        rotation_range=10,
        width_shift_range=0.10,
        height_shift_range=0.10,
        shear_range=0.10,
        zoom_range=0.10,
        horizontal_flip=True,
        fill_mode='nearest')
```

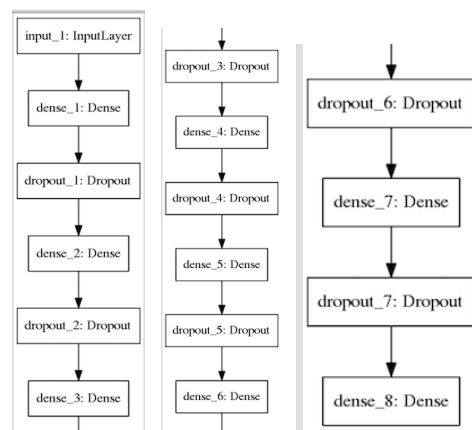
■ 訓練過程

我總共跑 150 次 epoch，每次 batch size 為 256。下圖為 accuracy 對 epoch 做圖的結果，由圖可知大概跑 20 個 epoch 的時候 validation set 的辨識率已經接近飽和(約 0.675)



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？

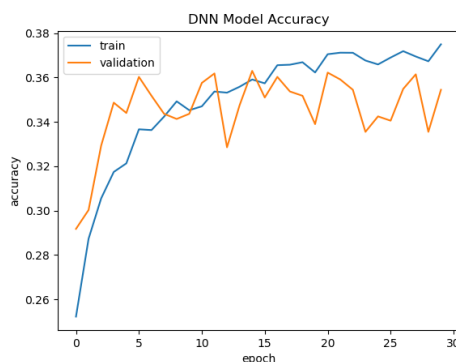
✧ 模型架構：



為求排版方便，上圖順序為由左而右，由上而下。而我在第 1 小題的 CNN 參數個數為 4,631,559 個，本題 DNN 參數個數為 4,467,975

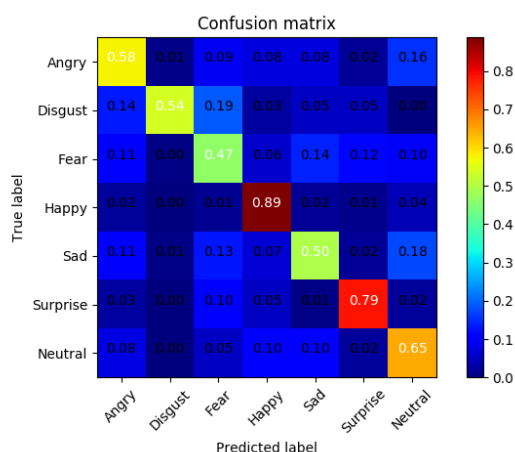
✧ 訓練過程與辨識率：

可以很容易看出大概在跑到 10 次 epoch 時，validation set 的辨識率便已呈現飽和，約 0.35 左右，所以 CNN 的辨識率幾乎是 DNN 的一倍

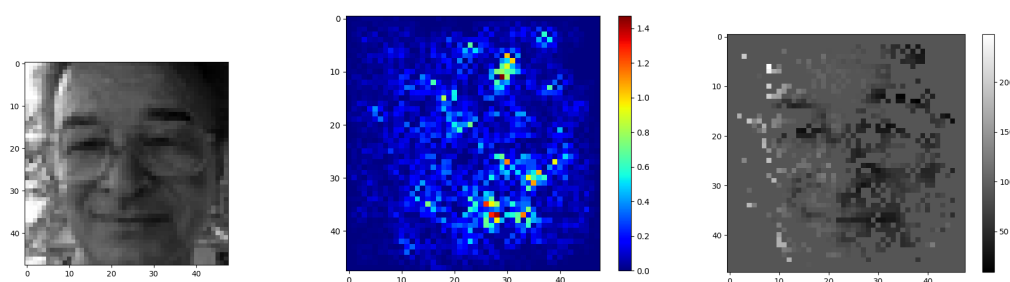


3. (1%) 觀察答錯的圖片中，哪些 class 彼此間容易用混？[繪出 confusion matrix 分析]

Confusion matrix 如下圖，觀察到難過(Sad)與恐懼(Fear)之間很容易搞混。

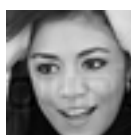


4. (1%) 從(1)(2)可以發現，使用 CNN 的確有些好處，試繪出其 saliency maps，觀察模型在做 classification 時，是 focus 在圖片的哪些部份？
我選用 id=24 之圖片(GT=3)，如下由左至右為原圖、saliency map 與 mask



可以看到在做分類時主要是 focus 在人臉的五官上，例如眼睛、嘴巴等。

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。



輸入 id=39 的圖片(GT=5)，



Layer0:

可以看到第 3 行、第 3 列的第 35 個 filter 最容易被驚訝的圖片 activate