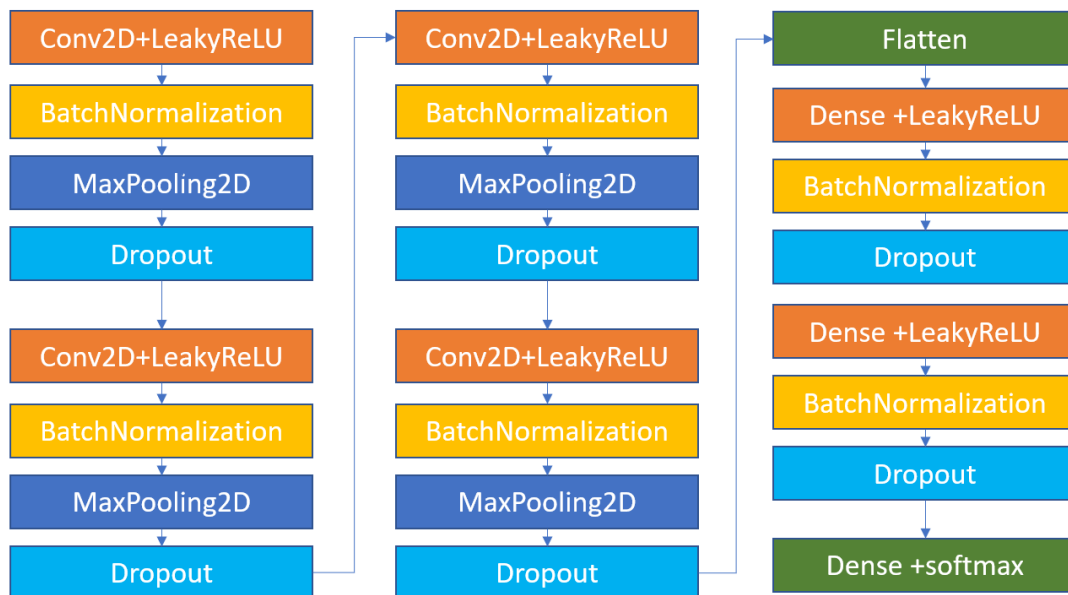


1. (2%) 請說明你實作的 CNN model，其模型架構、訓練參數和準確率為何？並請用與上述 CNN 接近的參數量，實做簡單的 DNN model，同時也說明其模型架構、訓練參數和準確率為何？並說明你觀察到了什麼？
kaggle 上最高分 public:0.73335 private :0.72722 為數個 model 集合投票而成

以下介紹其中單個最高分數的 model

	Public	Private
CNN	0.71719	0.71217
DNN	0.49010	0.49345

CNN 架構: Total params: 15,033,863



四層 Conv2D filter 分別為 512,512,768,768 個，kernal_size 皆為(3,3)

MaxPooling 的 pool_size=(2,2)，Dense 的神經數皆為 512，Dropout 比例皆為 0.25

使用的 Activation function 皆為 LeakyReLU，最後使用的 optimizer 為 Adamax

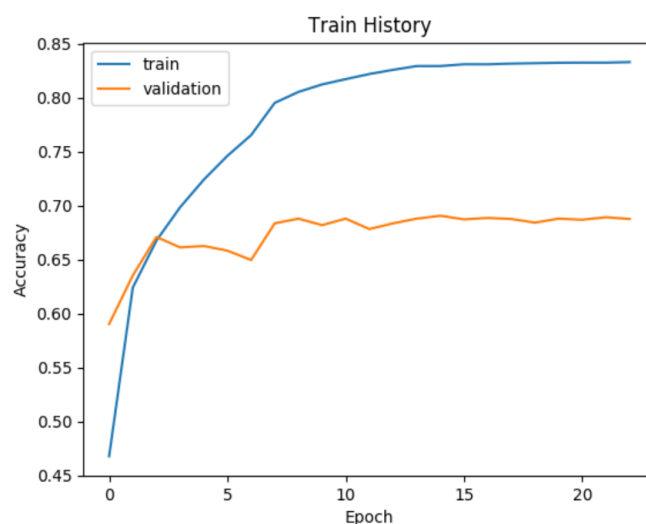
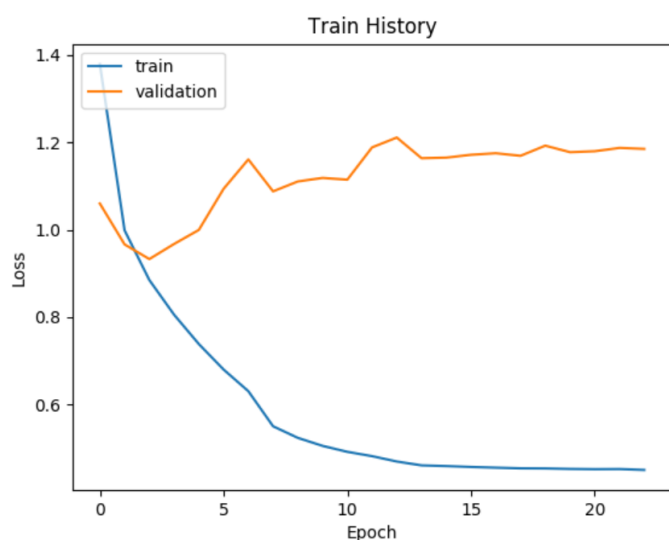
DNN 架構: Total params: 14,962,695

先經過 flatten，再由 13 個單位串接而成，每個單位為 1024 個神經元以 LeakyReLU 作為 Activation function，最後再接上輸出層的 7 個神經元

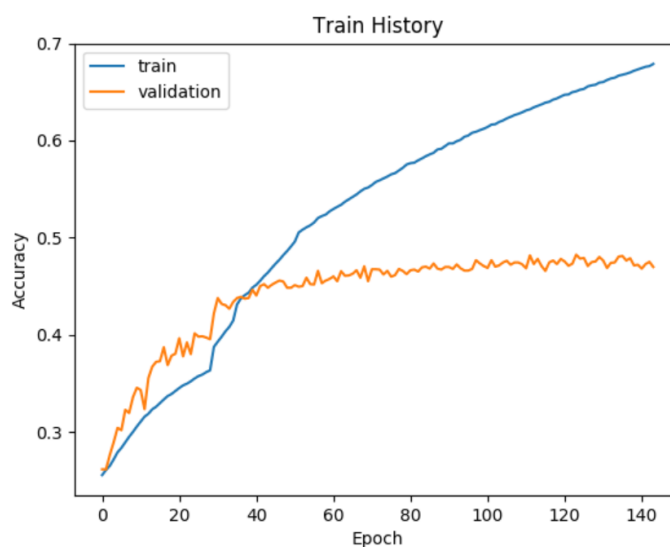
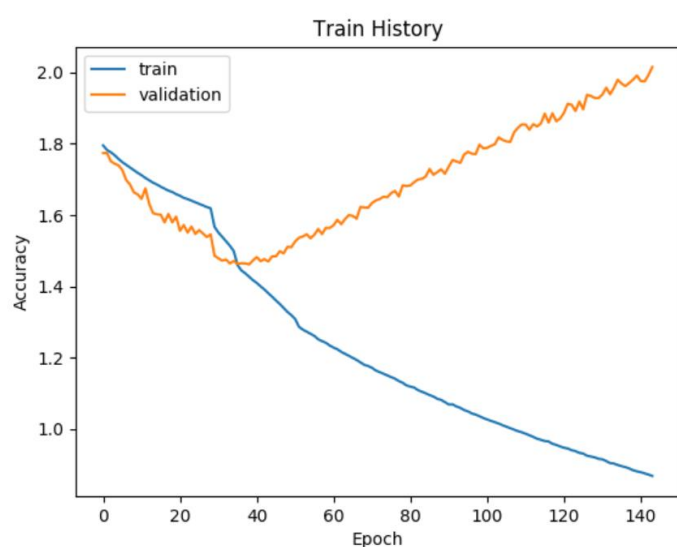
因 DNN 並沒有考慮到相鄰像素較有關係的問題，因此預測能力明顯較 CNN 為低

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 的 Loss 及 Accuracy 在 train 及 validation set 內的變化曲線



DNN 的 Loss 及 Accuracy 在 train 及 validation set 內的變化曲線



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:)

此處用較簡單的 CNN model Total params: 2,803,207

若有 normalize 則將各張圖片的各位置減去平均再除以標準差

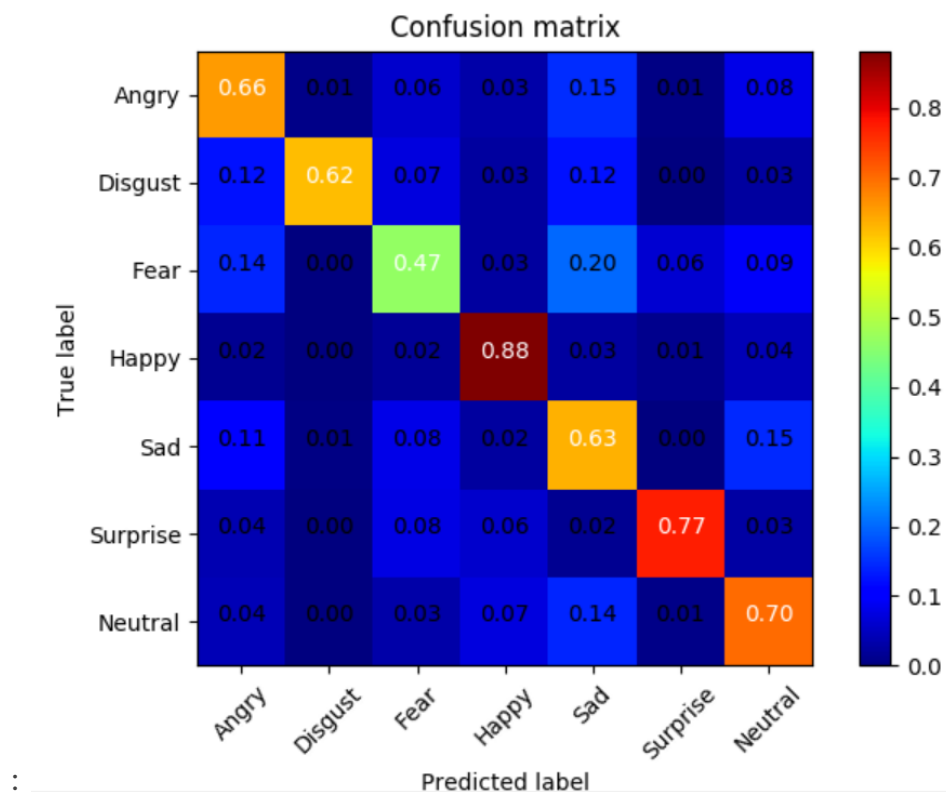
若有做 augmentation 則使用 model.fit 反之則用 model.fit_generator

```
generate = ImageDataGenerator(  
    rotation_range=30,  
    width_shift_range=0.2,  
    height_shift_range=0.2,  
    zoom_range=[0.8, 1.2],  
    shear_range=0.2,  
    horizontal_flip=True)  
generate.fit(X_train)
```

data normalization	data augmentation	Public	Private
No	No	0.62719	0.62440
No	Yes	0.69378	0.68487
Yes	No	0.62190	0.63555
Yes	Yes	0.69016	0.67511

在 HW3 內做 data normalization 效果不佳 甚至有反效果，應是因為對各 pixel 做標準化沒有整體意義
但 data augmentation 效果極佳，可有效透過縮放、旋轉等方式增加 data 數，進而改善 model

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:)



誤認率最大的三個為

Fear 被誤認為 Sad 機率 0.2

Sad 被誤認為 Neutral 機率 0.15，Angry 被誤認為 Sad 機率 0.15

Fear 被誤認為 Angry 機率 0.14，Neutral 被誤認為 Sad 機率 0.14