

Machine Learning - Homework 3 Report

學號：B06902049 系級：資工二 姓名：林首志

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？

(Collaborators: 無)

答：

由於Kaggle上best model是多個模型ensemble的結果，這裡我用其中一個CNN model來作答。

以下是我的CNN架構，

(BN: Batch Normalization, Conv(F , K): Conv2D with F filters and kernel size K)

Layers	Shape
Input	(48, 48, 1)
Conv(30, 3x3) + ReLU + BN + Conv(30, 3x3) + ReLU + BN + Conv(30, 3x3) + ReLU	(48, 48, 30)
MaxPooling2D(2x2)	(24, 24, 30)
BN + Conv(60, 3x3) + ReLU + BN + Conv(60, 3x3) + ReLU + BN + Conv(60, 3x3) + ReLU	(24, 24, 60)
MaxPooling2D(2x2)	(12, 12, 60)
BN + Conv(90, 3x3) + ReLU + BN + Conv(90, 3x3) + ReLU + BN + Conv(90, 3x3) + ReLU	(12, 12, 90)
MaxPooling2D(2x2)	(6, 6, 90)
BN + Conv(120, 3x3) + ReLU + BN + Conv(120, 1x1) + ReLU + BN + Conv(7, 1x1)	(6, 6, 7)
GlobalAveragePooling2D()	7
Softmax	7

其中Convolution Layers都有做padding，讓輸入輸出的圖片大小一致(keras中的padding='same')。

每個Convolution Layers都有對非bias的參數做 $\lambda = 0.0005$ 的L2 regularization，並使用He uniform對非bias的參數做初始化。

Keras計算出的總參數量是408,217，其中可以訓練的參數量是406,657。

輸入預處理是將資料的 $\frac{9}{10}$ 當作訓練資料，先將每個像素除以255，再套用keras內建的数据 augmentation (ImageDataGenerator)，其中包含水平翻轉、旋轉至多20度、水平或垂直平移至多15%，放大或縮小至多25%。

我用categorical crossentropy當作loss function，並使用SGD+nesterov做訓練，其中learning rate是0.1，momentum設為0.9，當Loss連續三個epoch沒有減少0.01以上時，會自動將learning rate乘上0.5。總Epoch數設為100，Batch size是256。

經過100個epoch後，此CNN Model在training set的準確率是78.27%，在validation set的準確率是68.37%，在public test set的準確率是68.877%。

接下來我用相似的參數量，做了一個DNN model。以下是我的架構，

Layers	Shape
Input	(48, 48, 1)
Flatten	2304
Dense(170 neurons) + ReLU	170
Dense(100 neurons) + ReLU	100
Dense(7 neurons)	7
Softmax	7

初始化和Regularization的參數和CNN一樣。

總參數量是409,657，全部都是可訓練的參數。

訓練參數大部分都和CNN相同，唯二的差別是DNN沒有做data augmentation，learning rate設為0.01。

取100個epoch中validation accuracy最高的一次，此DNN Model在training set的準確率是57.41%，在validation set的準確率是44.31%，在public test set的準確率是42.630%。

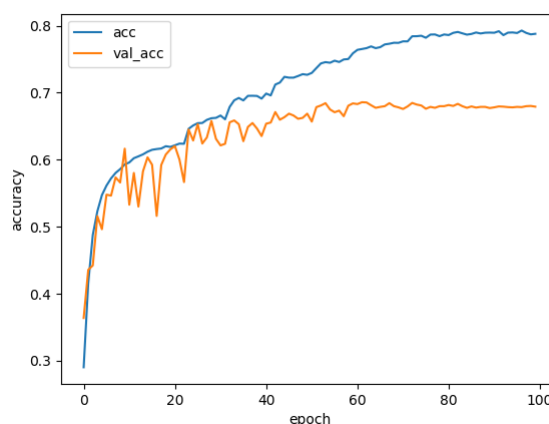
比較CNN Model和DNN Model，可以發現使用相近的參數量，CNN的表現比DNN好很多。

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)

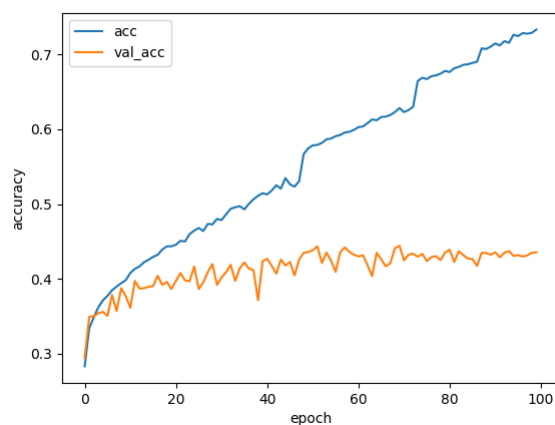
(Collaborators: 無)

答：

下圖是CNN model的訓練過程，



下圖是DNN model的訓練過程，



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: 無)

答：

	training set accuracy	validation set accuracy	public test set accuracy
都不使用	99.72%	62.38%	63.276%
data normalization	99.74%	64.82%	65.032%
data normalization + data augmentation (第一小題的 CNN model)	78.27%	68.37%	68.877%

這裡Data normalization的實作方法是對每個像素除以255。由於原本的像素數值範圍是[0, 255]，除以255後範圍變成[0.0, 1.0]，適合用來訓練一般的神經網路。

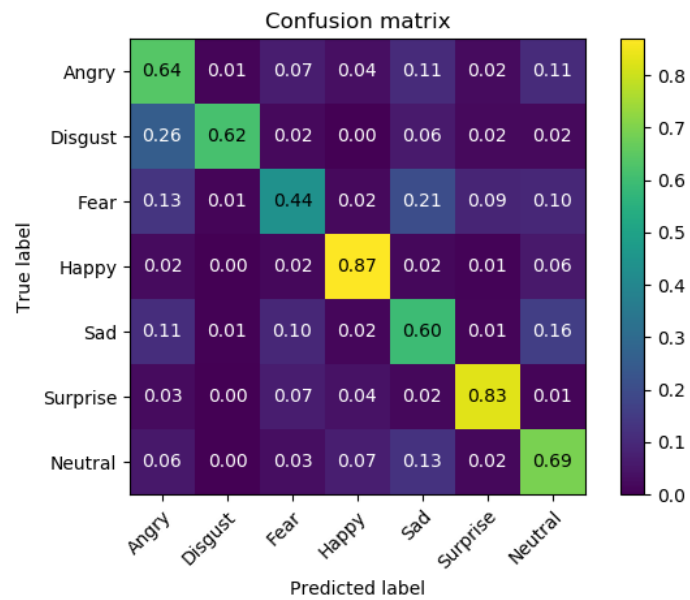
Data augmentation的作法同第一小題，我使用keras內建的数据 augmentation (ImageDataGenerator)，其中包含隨機水平翻轉、隨機旋轉至多20度、隨機水平或垂直平移至多15%，隨機放大或縮小至多25%。

觀察表格可以發現，實作data normalization對準確度有不錯的幫助（可能是因為batch normalization的關係，做與不做的差異沒有想像中大）。實作data augmentation有類似於regularization的效果，雖然training set accuracy降低了，但是validation set accuracy和test set accuracy都有顯著的提昇。

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: 無)

答：



下表是搞混的前五名，

原類別	預測類別	機率(原類別有多少機率被預測成預測類別)
Disgust	Angry	0.26
Fear	Sad	0.21
Sad	Neutral	0.16
Fear	Angry	0.13
Neutral	Sad	0.13

可以看出

1. Disgust常常被誤判成Angry。
2. Fear的預測正確率很低 (只有0.44)，容易和其他類別搞混。
3. Sad和Neutral之間常常互相搞混。