學號：B05705006 系級： 資管四 姓名：李和維

1. (1%) 請說明這次使用的model架構，包含各層維度及連接方式。

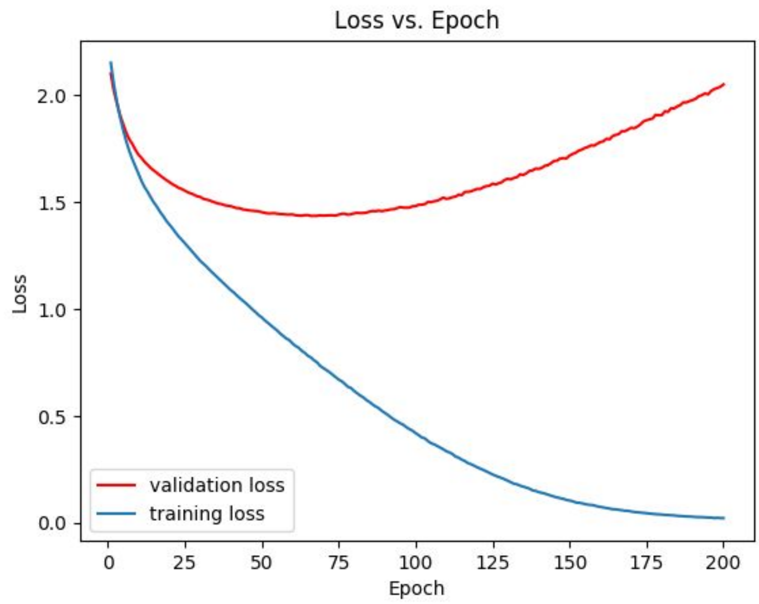
   此次作業使用了 torchvision.models 中的 Resnet18 並修改了 output 那層的 node 數為 7（代表七種情緒），其架構、維度與連接方式如下：

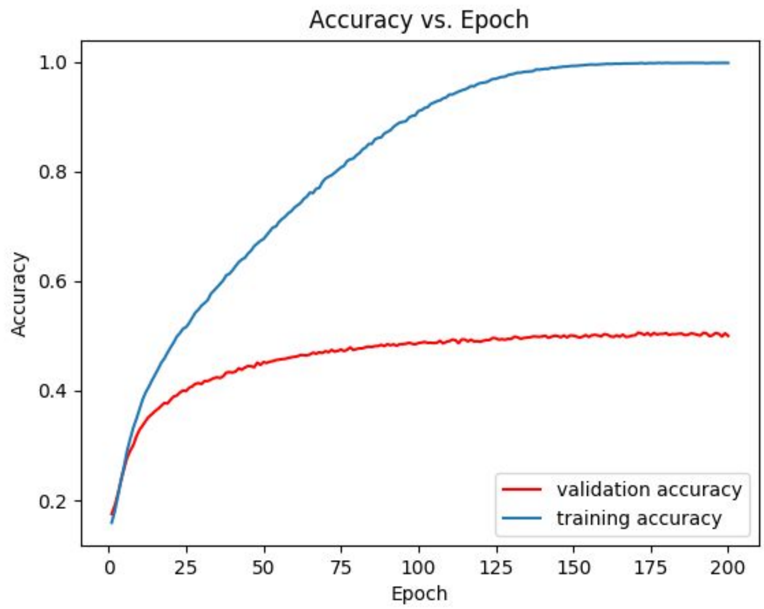
   （假設 input size = N * N，Conv2d 參數：input channel, output channel, kernel size，且每層 Conv2d 後都接了 BatchNormalize 和 ReLU）

layer name	output size	architecture
input stem	N/2 * N/2	Conv2d (3, 64, 7*7, stride = 2) 3 * 3 max pool, stride = 2
layer 1	N/4 * N/4	Conv2d (64, 64, 3*3) Conv2d (64, 64, 3*3)
layer 2	N/8 * N/8	Conv2d (64, 128, 3*3) Conv2d (128, 128, 3*3)
layer 3	N/16 * N/16	Conv2d (128, 256, 3*3) Conv2d (256, 256, 3*3)
layer 4	N/64 * N/64	Conv2d (256, 512, 3*3) Conv2d (512, 512, 3*3)
avgpool	1 * 1	AdaptiveAvgPool2d (output size = (1, 1))
output	7	Linear (in features = 512, out features = 7)

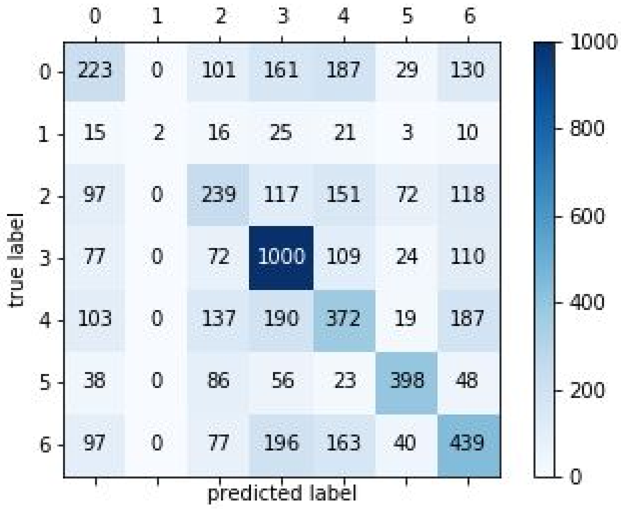
2. (1%) 請附上model的training/validation history (loss and accuracy)。

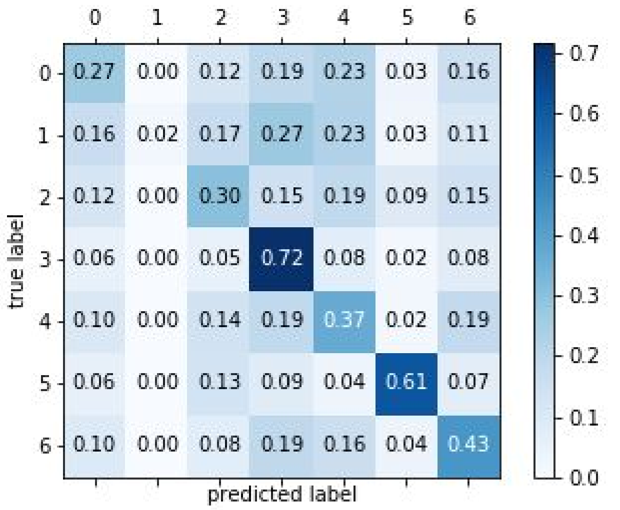
   如下圖。其中 Optimizer 使用 Adam (lr = 1e-6)，training epoch = 200，loss function = cross entropy。





3. (1%) 畫出confusion matrix分析哪些類別的圖片容易使model搞混，並簡單說明。





（此 model 的 training epoch = 60，類別：0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral）

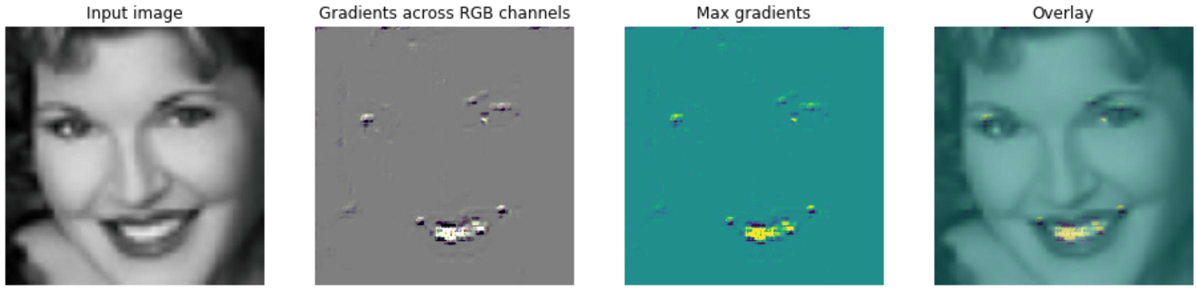
由 confusion matrix 可知，model 對於類別 1 幾乎都會搞混，且特別容易看成 3 或 4。

另外，對於 0, 2 這兩個類別也很容易誤判，都很容易看成 4。

[關於第四及第五題]

可以使用簡單的 3-layer CNN model [64, 128, 512] 進行實作。

4. (1%) 畫出CNN model的saliency map，並簡單討論其現象。

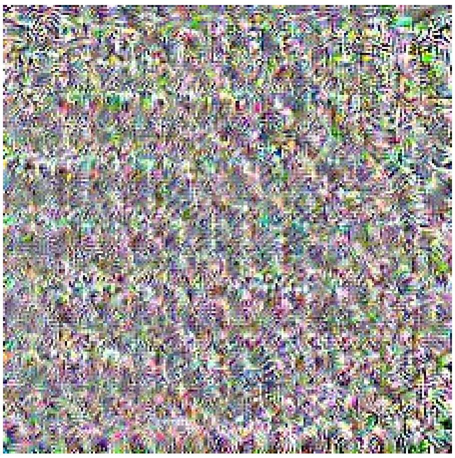


target class = 3 (Happy)，代表圖中顏色較深的部分對 predict happy 有最大的 effect。換句話說，微笑時的牙齒與彎彎的眼睛周圍最能 convince model 將它分類成 happy。

5. (1%) 畫出最後一層的filters最容易被哪些feature activate。

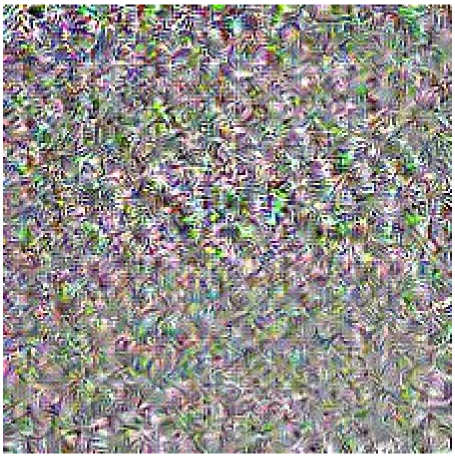
選了最後一層的前 3 個 filter，固定 model 參數並透過 gradient descent 來分別 optimize 能最大化 average activation 的 input feature，得到前三個 filter 最容易被哪些 feature activate。結果如下：

filter 1



filter 2

filter 3



6. (3%)Refer to math problem

https://hackmd.io/JIZ_0Q3dStSw0t0O0w6Ndw

No.

Date

① $$\left(B, \left\lfloor \frac{(W+2P_1)-K_1}{S_1} \right\rfloor + 1, \left\lfloor \frac{(H+2P_2)-K_2}{S_2} \right\rfloor + 1, \text{output\_channels}\right)$$

② $$\frac{\partial l}{\partial \hat{x}_i} = \frac{\partial l}{\partial y_i}\gamma, \quad \frac{\partial l}{\partial \sigma_\beta^2} = \sum_{i=1}^{m} \frac{\partial l}{\partial \hat{x}_i}(x_i - \mu_\beta)\frac{-1}{2}(\sigma_\beta^2 + \epsilon)^{-\frac{3}{2}},$$

$$\frac{\partial l}{\partial \mu_\beta} = \left(\sum_{i=1}^{m} \frac{\partial l}{\partial \hat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_\beta^2 + \epsilon}}\right) + \frac{\partial l}{\partial \sigma_\beta^2} \cdot \frac{\sum_{i=1}^{m} -2(x_i - \mu_\beta)}{m},$$

$$\frac{\partial l}{\partial x_i} = \frac{\partial l}{\partial \hat{x}_i} \cdot \frac{1}{\sqrt{\sigma_\beta^2 + \epsilon}} + \frac{\partial l}{\partial \sigma_\beta^2} \cdot \frac{2(x_i - \mu_\beta)}{m} + \frac{\partial l}{\partial \mu_\beta} \cdot \frac{1}{m},$$

$$\frac{\partial l}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial l}{\partial y_i} \cdot \hat{x}_i, \quad \frac{\partial l}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial l}{\partial y_i}$$

③ $\frac{\partial l_t}{\partial z_t} = -y_t \frac{\partial \log \hat{y}_t}{\partial z_t} = -y_t \frac{1}{\hat{y}_t} \frac{\partial \hat{y}_t}{\partial z_t}$

$= -\frac{y_t}{\hat{y}_t} \hat{y}_t (1-\hat{y}_t) = -y_t + y_t \hat{y}_t = \hat{y}_t - y_t$ //