學號：60972015H 系級： 圖傳碩三 姓名：鄒明倫

1. (1%) 實作early-stopping，繪製training, validation loss/acc 的 learning curve，比較實作前後的差異，並說明early-stopping的運作機制

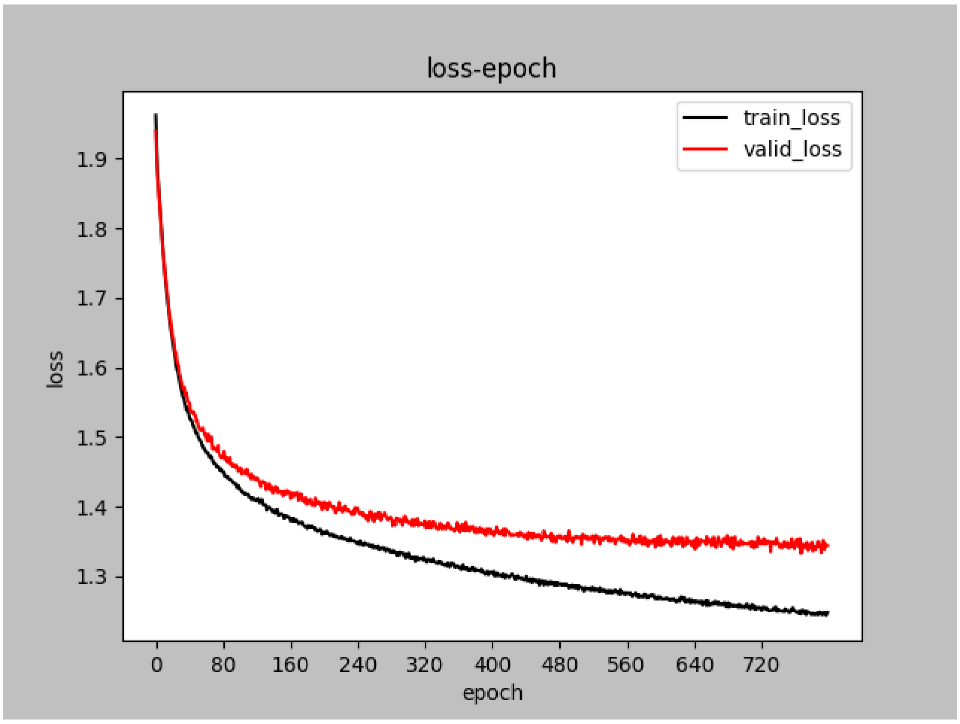
<code>

```
        if loss_valid < loss_min:
            loss_min = loss_valid
            early_stop = 0
        else:
            early_stop += 1
            if early_stop == args.early_stop:
                print("early stop! epoch = %5d, train_loss = %3.5f, valid_loss = %3.5f, acc = %1.5f" % (epoch, loss_ave, loss_valid, accuracy))
                break
```

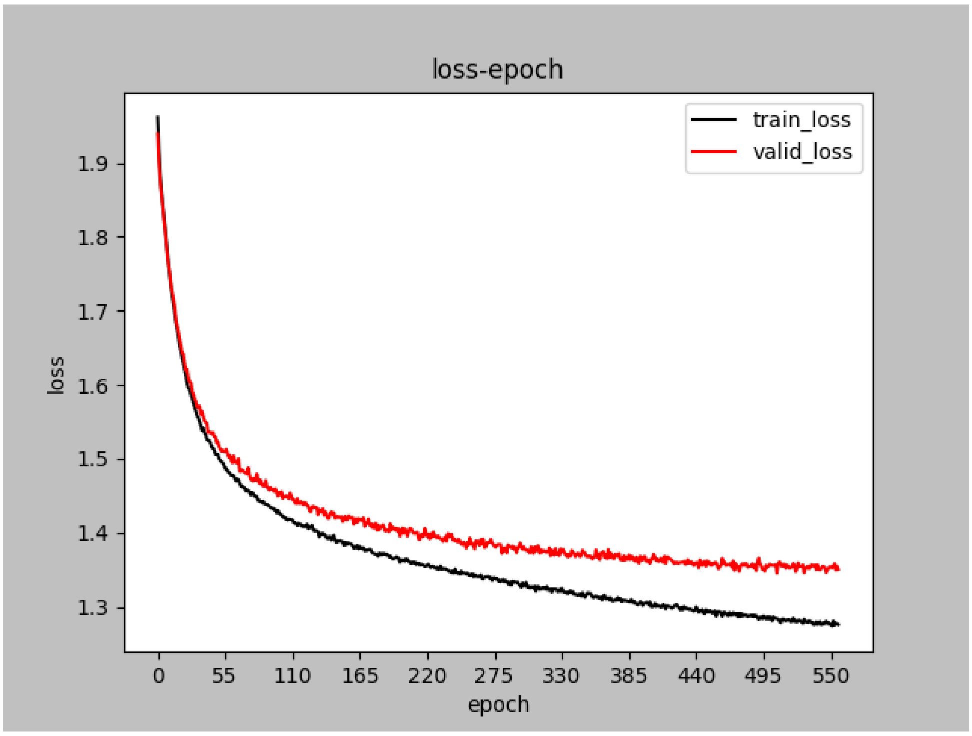
運作機制:

每一 epoch training 結束後，對 valid dataset 計算平均 loss 得到 loss_valid。訓練開始時有預設 loss_min (一開始給一個大值)，當 loss_valid 比 loss_min 大時，early_stop 計數器 +1，當 loss_valid 比 loss_min 小時，early_stop 計數器歸零，且 loss_min 更新等於 loss_valid。若 early_stop 計數器持續增加，代表 training 卡住或是 overfitting，直到 early_stop 數與預設的 args.early_stop 數一樣時，中止訓練。

<loss plot>



<no early stop>



<early stop>

差異:

沒有 early stop 時，到最後雖然training loss在下降，但 valid loss 會不斷上升，越來越 overfit training data 不夠 generalize。有 early stop 則可以在開始上升的時候結束訓練，不浪費時間。

2. (1%) 嘗試使用 augmentation，說明實作細節並比較有無該 trick 對結果表現的影響(validation 或是 testing 擇一即可)，且需說明為何使用這些 augmentation的原因。
(ref: https://pytorch.org/vision/stable/transforms.html)
traick: RandomHorizontalFlip(p=0.5)
使用原因:

因為資料集中人臉都是頭頂朝上，人像左右翻轉能夠增加圖片多樣性，而上下顛倒就可能與此資料集較不相關。

<valid loss>

for Cnn model in 800 epoch, early_stop = 50

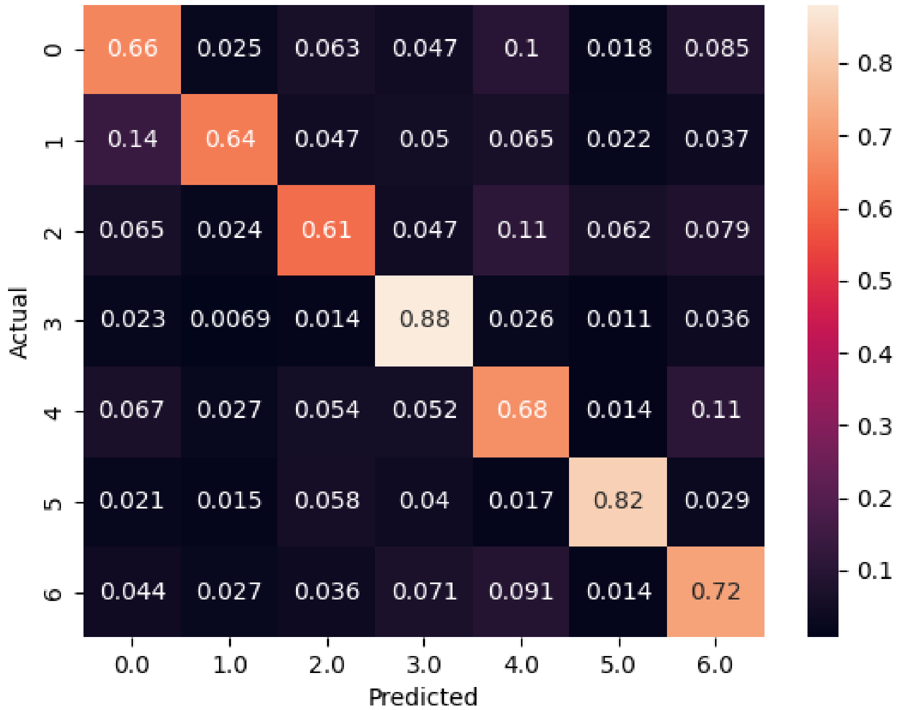
aug:

epoch = 481, train_loss = 1.28932, valid_loss = 1.35828, acc = 0.49151

no aug:

epoch = 473, train_loss = 1.24919, valid_loss = 1.38488, acc = 0.48090

3. (1%) 畫出 confusion matrix 分析**哪**些類別的圖片容易使 model **搞**混，找出模型出錯的例子，並分析可能的原因。
(ref: https://en.wikipedia.org/wiki/Confusion_matrix)



<分析>

模型在 class1 判別時，有 14% 判成 class 0，厭惡會判成生氣。

<class 1>

<class0>

可能原因是厭惡表情在眼部特徵與生氣相像，且又無其他特別於生氣的特徵，另一原因是厭惡的資料數量較少，約**為**生氣的1/9，模型易偏向增加生氣的機率**值**。

4. (1%) 請統計訓練資料中不同類別的數量比例，並說明：
對 testing 或是 validation 來說，不針對特定類別，直接選擇機率最大的類別會是最好的結果**嗎**？
(ref: https://arxiv.org/pdf/1608.06048.pdf, or hints: imbalanced classification)
<Ans> Training data 中 {0: 3579, 1: 402, 2: 3736, 3: 6525, 4: 4321, 5: 2872, 6: 4452}

直接選擇機率大的可能會造成模型偏頗，classl size 約為其他 class size 的 1/10，需要在 crossentropyloss 中加上 weight 來平衡。

5. (4%)Refer to math problem https://hackmd.io/@lH2AB7kCSAS3NPw2FffsGg/r1otQp7Gi?fbclid=IwAR0cs5CajVy_zhDmHEDgze2V1_Jlxp95N45BF6hg1l6CgG-6IViYGAIGReE

Hw2-math

Problem 1 (1%)

Consider a generative classification model of K classes $C_1, ..., C_K$ described by prior probabilities $p(C_k) = \pi_k$ and conditional probabilities $p(x|C_k)$, where x is the input feature vector. Suppose we are given a training data set $((x_i, t_i))_{i=1}^N$ where $i = 1, ..., N$, and $t_i = (t_i^1, ..., t_i^K) \in \{0, 1\}^K$ is the 1-of-K encoding given by

$$t_i^k = \begin{cases} 1 & \text{, if the } i\text{'th pattern is from class } C_k \\ 0 & \text{, otherwise} \end{cases}$$

Assume the data points are drawn independently from this model. Show that the maximum-likelihood solution for the prior probabilities is given by

$$\pi_k = \frac{N_k}{N}$$

where N_k is the number of data points belonging to class C_k.

<Ans>

$p(\varphi_n, C_j) = p(\varphi_n|C_j)p(C_j)$ (chain rule)

$p(t_i, \varphi_i|\pi) = \Pi_{j=1}^K (\pi_j p(\varphi_i|C_j))^{t_{nj}}$

get loss likelihood

$$\ln p(D|\pi) = \ln p(t, \Phi|\pi) = \sum_{n=1}^N \sum_{j=1}^K t_n j(ln(\pi_j) + ln(p(\varphi_n|C_j)))$$

use $\sum_{j=1}^K \pi_j = 1$ and lagrange function

1. $\sum_{n=1}^N \frac{t_n^k}{\pi_k} + \lambda = \frac{N_k}{\pi_k} + \lambda = 0$

$\rightarrow \pi_k = -N_k/\lambda$

2. $\sum_{j=1}^{K} \pi_j = /cfrac{-1}\lambda \sum_{j=1}^{K} N_j = 1$

$\rightarrow \lambda = -N$

$$\therefore \pi_k = \frac{-N_k}{-N}$$

Problem 2 (1%)

a.<proof>

$$f(w) = w^T A w$$

$$\begin{aligned} f(w+h) - f(w) &= (w+h)^T A(w+h) - w^T A w \\ &= w^T A w + w^T A h + h^T A w + h^T A h - w^T A w \\ &= h^T (A^T w + A w) + c \end{aligned}$$

$$\therefore v = A^T w + A w$$

$$\text{and for } A \text{ is symmetric matrix } A^T = A,\ v = 2Aw$$

b.<proof>

$$\frac{\partial tr(AB)}{\partial a_{ij}} = \frac{\partial (\sum_{i=1}^{m} \sum_{j=1}^{m} a_{ij} b_{ji})}{\partial a_{ij}} = b_{ji}$$

c.

Reference

1. Pattern Recognition and Machine Learning 2006