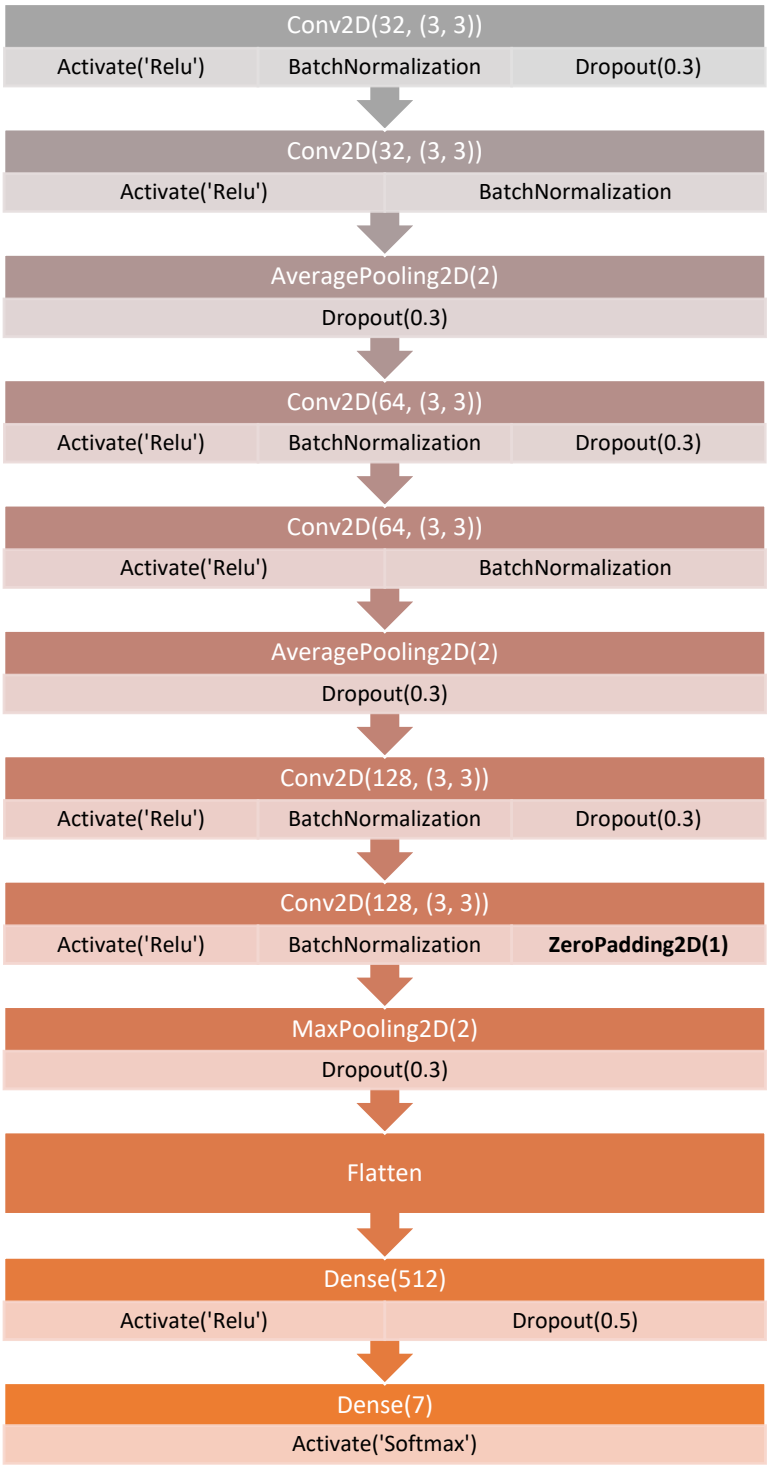
學號：B04902053 系級：資工二 姓名：鄭淵仁

**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？**

答：

- 模型架構：




最後的總參數量是 882,151。

- **訓練過程：**

我訓練了很多組 model，每一組 model 的架構都一樣，只有參數有些微變化。最後我把這些 model predict 出來的機率加起來，輸出機率最大的表情。

其中，對於每一組 model，我的架構都是：先對所有圖片**前處理**，接著使用上面描述過的模型架構**訓練參數**，之後**處理 testing data**，再 **predict** 出結果。

圖片前處理的作法如下：

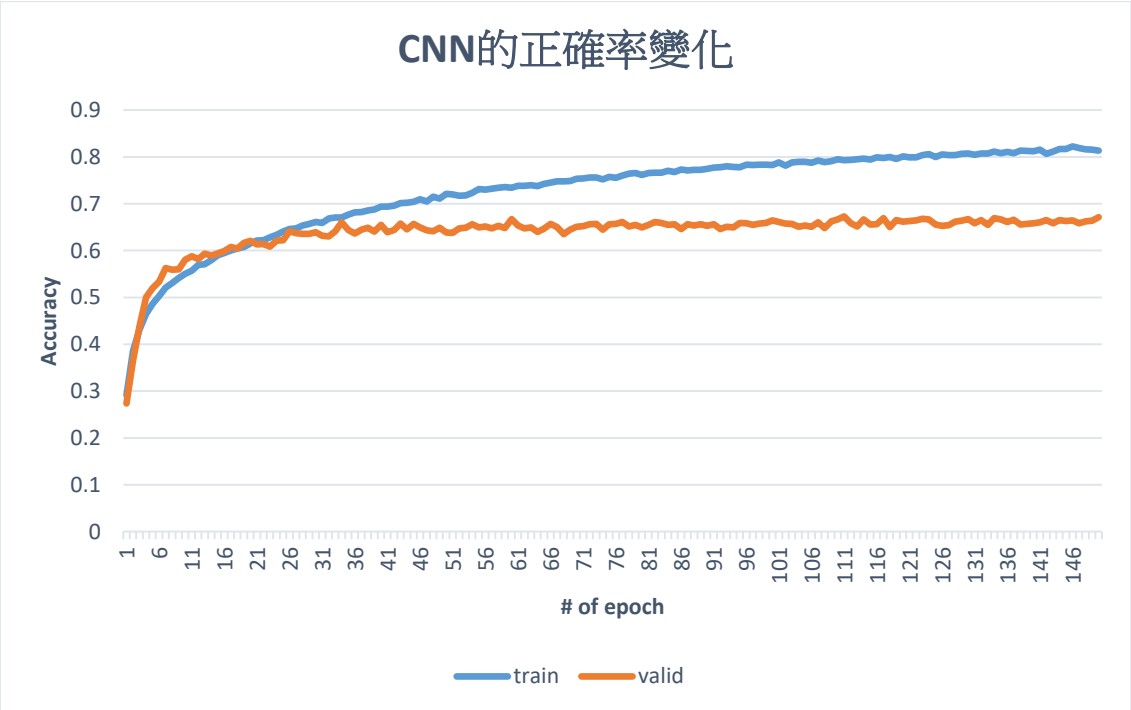
1) 對所有圖片（training data、testing data）以單一圖片為單位做 **normalization**，以減少 bias。
2) 使用 keras 的 Image Data Generator，在每個 epoch 前隨機對圖片做**鏡像**、**旋轉**、**裁切**等操作，以增加資料量。

而在 predict testing data 之前，我先把 testing data 也生成出**上下左右移動**以及**旋轉**等多組 data，再用這些 data 一一 predict 出個別表情的機率。接著把機率加起來，使用最大機率的表情做為結果輸出。

- **準確率：**

單一個 model 訓練出來的結果在 training set 上的 accuracy 是 0.8135，在自己切的 validation set 上的 accuracy 是 0.6714。最後把多個 model 加總起來之後，在 validation set 上的 accuracy 是 0.6838。

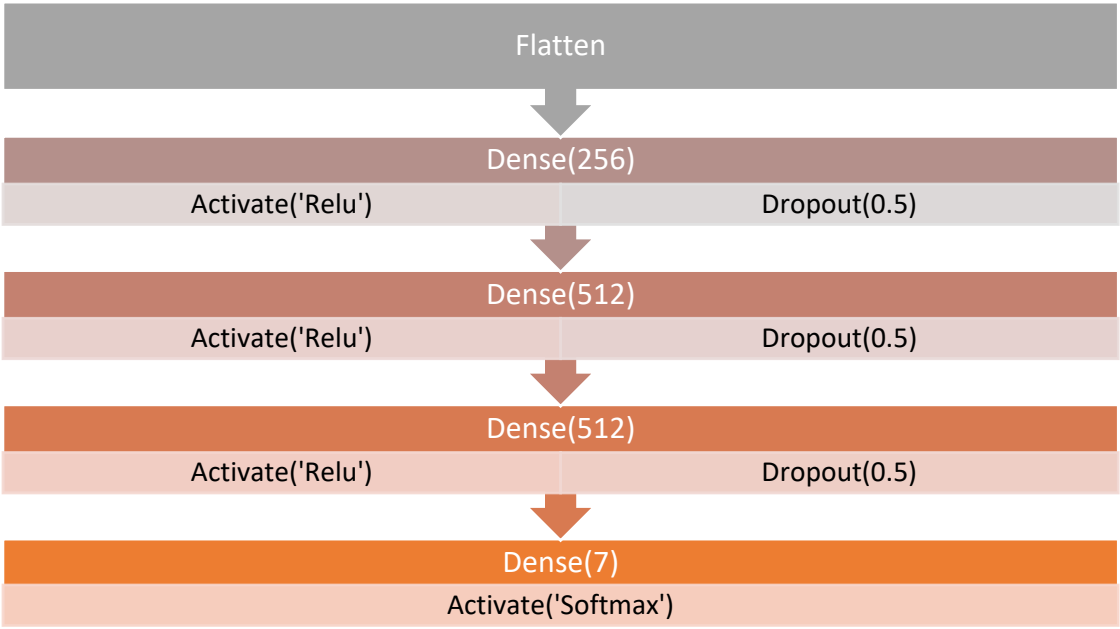
在只有一個 model 的訓練過程中，各個 epoch 的 accuracy 變化如下圖所示：



**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？**

答：

- **模型架構：**




總參數量則是 987,911，大約等於第 1 題 CNN 的訓練參數量（882,151）。

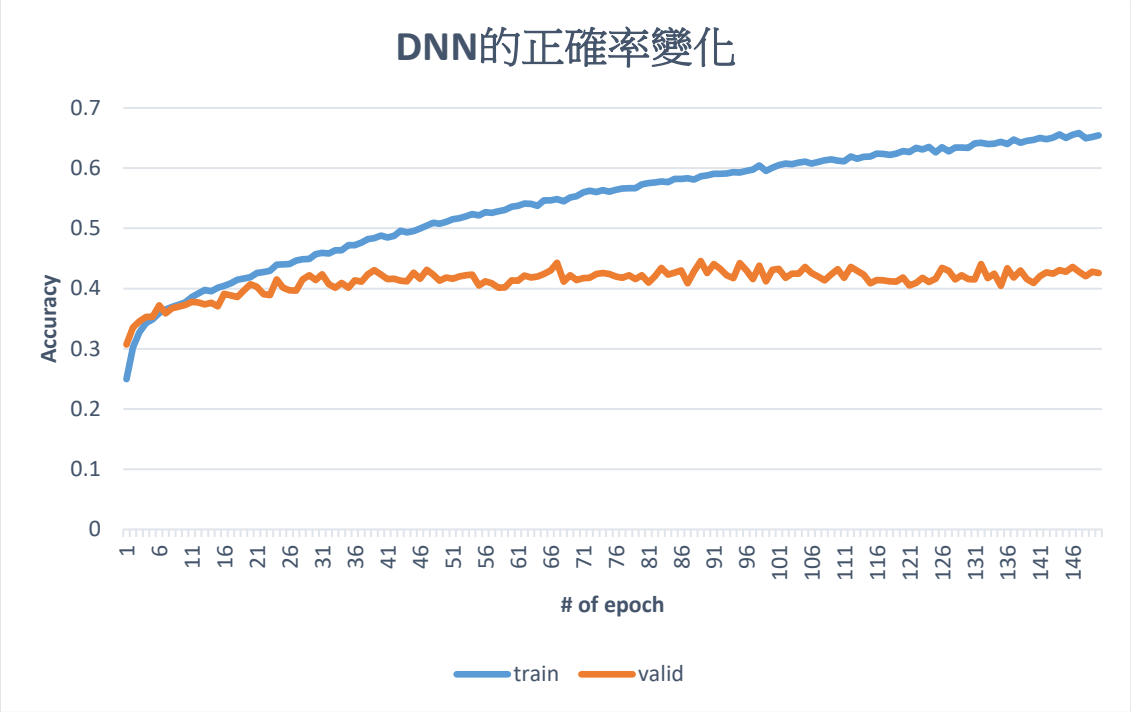
- **訓練過程：**

為了能和上一題 CNN 的數據比較，我的訓練過程和 DNN 的處理方式一模一樣，也就是：一樣生出多個 model 再把機率加總起來取最大值、一樣有圖片前處理、一樣在 predict 前處理 test data。

- **準確率：**

訓練出來的結果在 training set 上的 accuracy 是 0.6546，在自己切的 validation set 上的 accuracy 是 0.4307。

訓練過程中，各個 epoch 的 accuracy 變化如下圖：



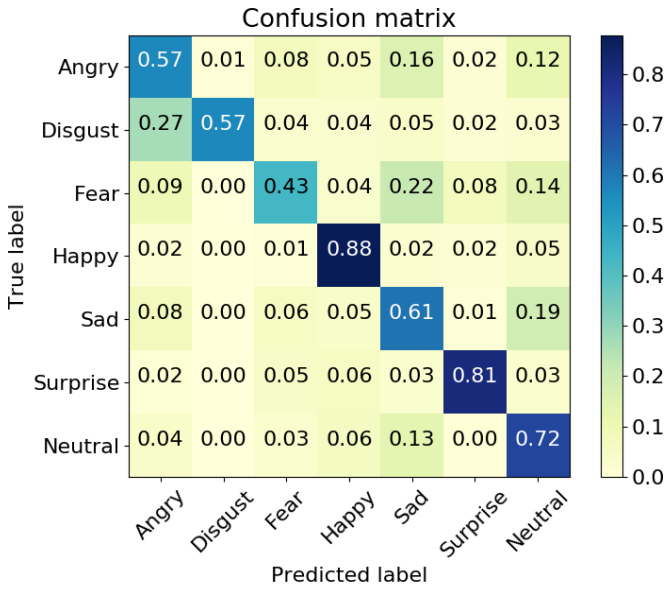

- **觀察到了什麼：**

我觀察到：DNN 在 training set 和 validation set 上面的正確率明顯都比 CNN 低，上升速度也慢很多。

**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]**

**答：**

下圖是我在 validation set 上 predict 之後，輸出的結果與正確答案的 confusion matrix：



從圖中可以發現：許多原本 label 是 disgust 的圖片，被辨識成 angry，其次則是 fear 被辨識成 sad。

但是原本被 label 成 disgust 的圖片只占所有圖片中的 1.5%左右，所以 disgust 的辨識會比較不準，可能不具有代表性。

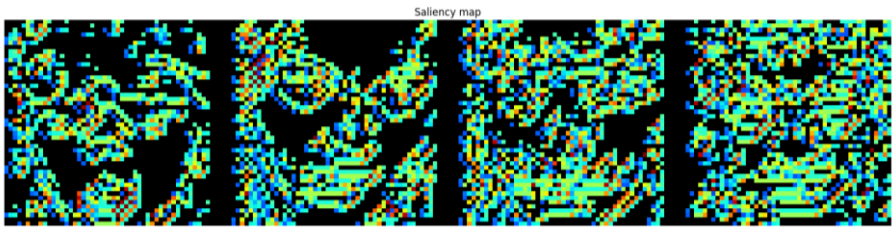
相較之下，fear 和 sad 都有一定數量，如果有更多的 data，這兩者容易用混的程度可能比 disgust 和 angry 要高。

**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？**

**答：**

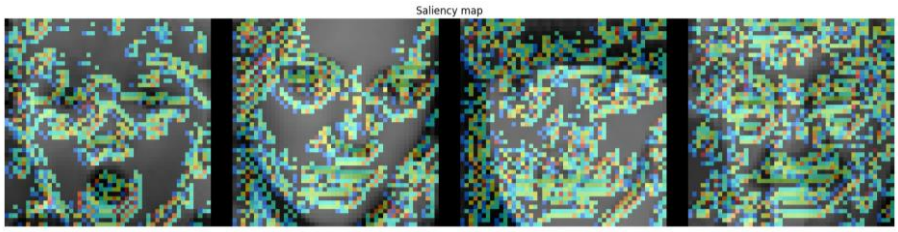
我使用 keras vis 的套件，針對第一層 convolution layer 畫 saliency map，畫出的結果如下：

- 畫出的 saliency map：




- 原圖：

- 將 saliency map 和原圖結合：



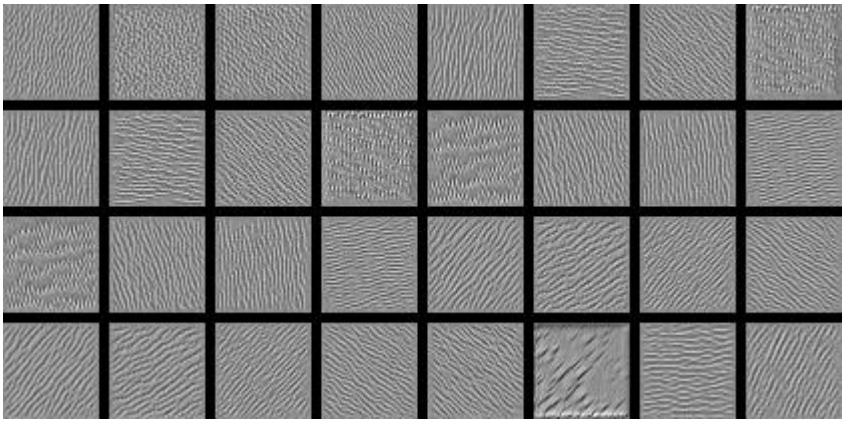

觀察上面的圖可以發現：我的模型在做 classification 時，主要會 focus 在圖片裡面人的五官以及臉部輪廓。但是如果圖片上被蓋了浮水印、或是臉部皺紋很多，就會影

響到模型 focus 的部分。

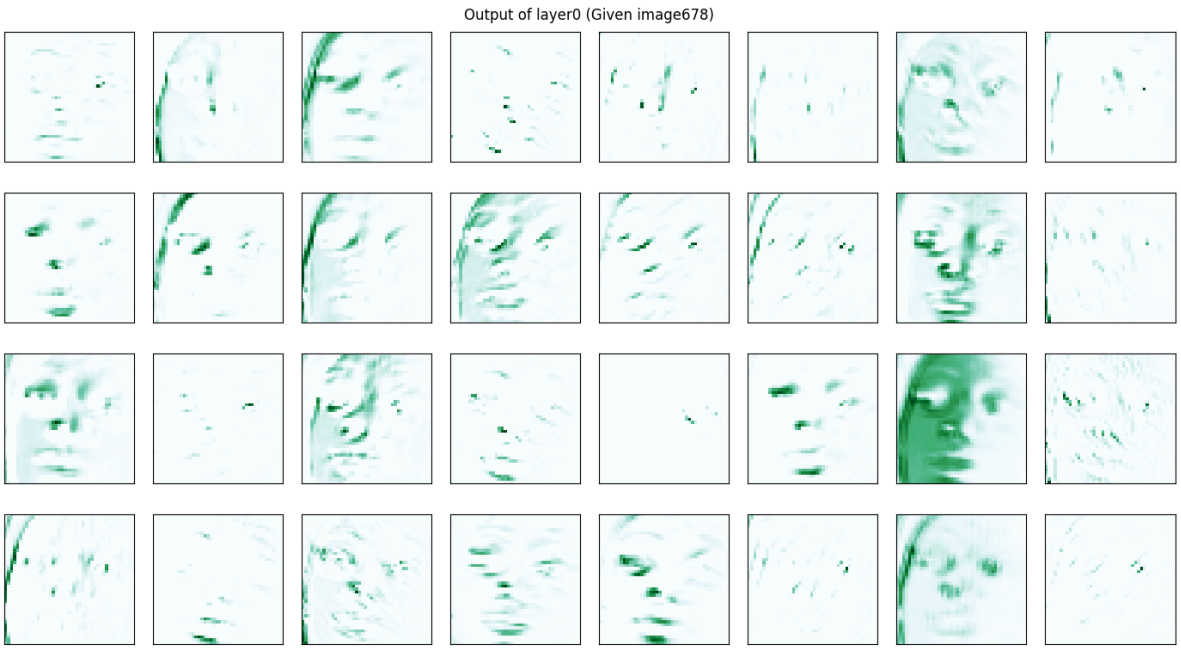
**5. (1%) 承(1)(2)，利用上課所提到的 gradient ascent 方法，觀察特定層的 filter 最容易被哪種圖片 activate。**

**答：**

我取第 2 層 convolution 的 filter 出來觀察，最容易 activate 這一層 filter 的圖片如下圖所示：



把圖放進 filter 後，產生的結果如下圖所示：




由上面兩張圖，我觀察到由於是前面層數的 filter，所以 filter 基本上還是在辨識直線和曲線。而在不同 filter 裡面，人臉中不一樣方向的曲線也被辨別出來了。很有可能在後層的 filter 的時候，就可以讓程式用這些曲線去觀察人臉上不同的特徵。

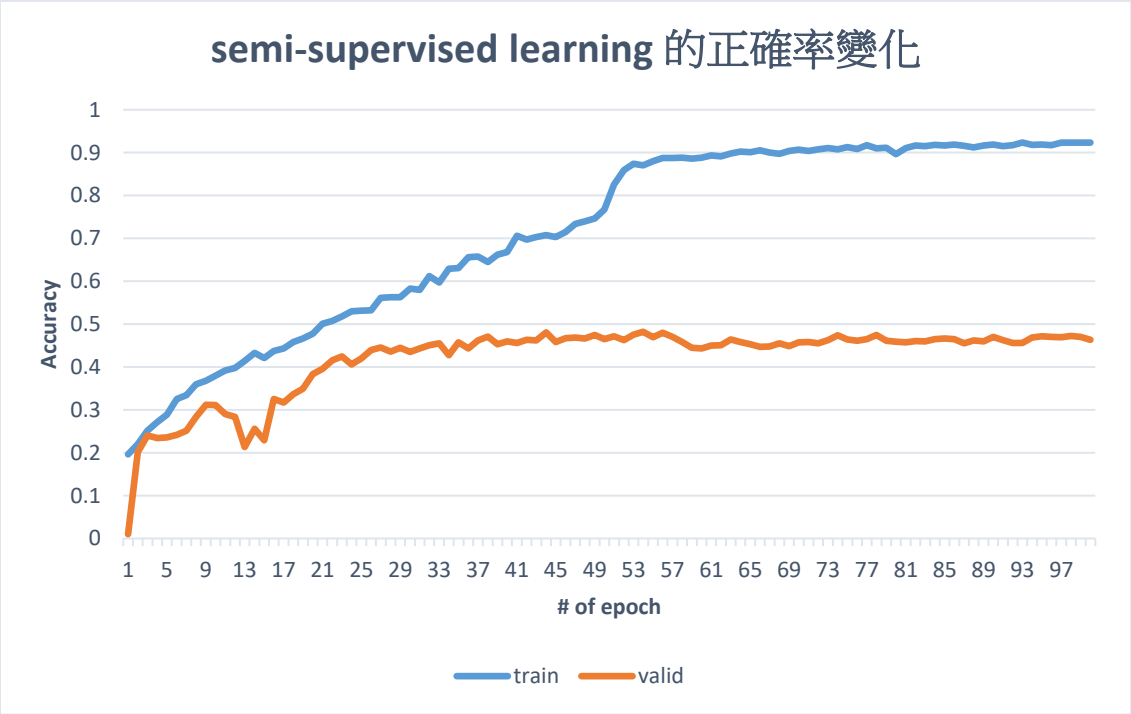
**[Bonus] (1%) 從 training data 中移除部份 label，實做 semi-supervised learning**

我先切 1400 個圖片作為 validation set，剩下的 90%的資料去掉 label。

只使用 training data，再使用 CNN 訓練 100 個 epoch，在 validation set 上面的正確率是 0.472142。

而如果把沒有 label 的資料也加進來用 semi-supervised learning 的話，正確率則上升到 0.480714。

semi-supervised learning 的訓練過程中，各個 epoch 的 accuracy 變化如下圖：




**[Bonus] (1%) 在 Problem 5 中，提供了 3 個 hint，可以嘗試實作及觀察 (但也可以不限於 hint 所提到的方向，也可以自己去研究更多關於 CNN 細節的資料)，並說明你做了些什麼？ [完成 1 個: +0.4%, 完成 2 個: +0.7%, 完成 3 個: +1%]**