學號：B03901030 系級： 電機三 姓名：蕭晨豪

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：

   我的 CNN 使用的架構如下:

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 46, 46, 64)	640
conv2d_2 (Conv2D)	(None, 44, 44, 64)	36928
max_pooling2d_1 (MaxPooling2	(None, 22, 22, 64)	0
dropout_1 (Dropout)	(None, 22, 22, 64)	0
conv2d_3 (Conv2D)	(None, 20, 20, 128)	73856
max_pooling2d_2 (MaxPooling2	(None, 10, 10, 128)	0
dropout_2 (Dropout)	(None, 10, 10, 128)	0
conv2d_4 (Conv2D)	(None, 8, 8, 256)	295168
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 256)	0
dropout_3 (Dropout)	(None, 4, 4, 256)	0
flatten_1 (Flatten)	(None, 4096)	0
dense_1 (Dense)	(None, 1700)	6964900
dropout_4 (Dropout)	(None, 1700)	0
dense_2 (Dense)	(None, 7)	11907

Total params: 7,383,399.0
Trainable params: 7,383,399.0
Non-trainable params: 0.0

   從原資料中隨機選 3500 筆做為 validation data，之後加入 image data generator 來產生有 noise 的資料，從中抽取 sample 來 train。使用的 optimizer 是 adamax，並且存 validation accuracy 最高的 model 來作 prediction，在 Kaggle 上的正確率為 69.629%

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：

   我使用數層 fully connected 來產生，架構如下:

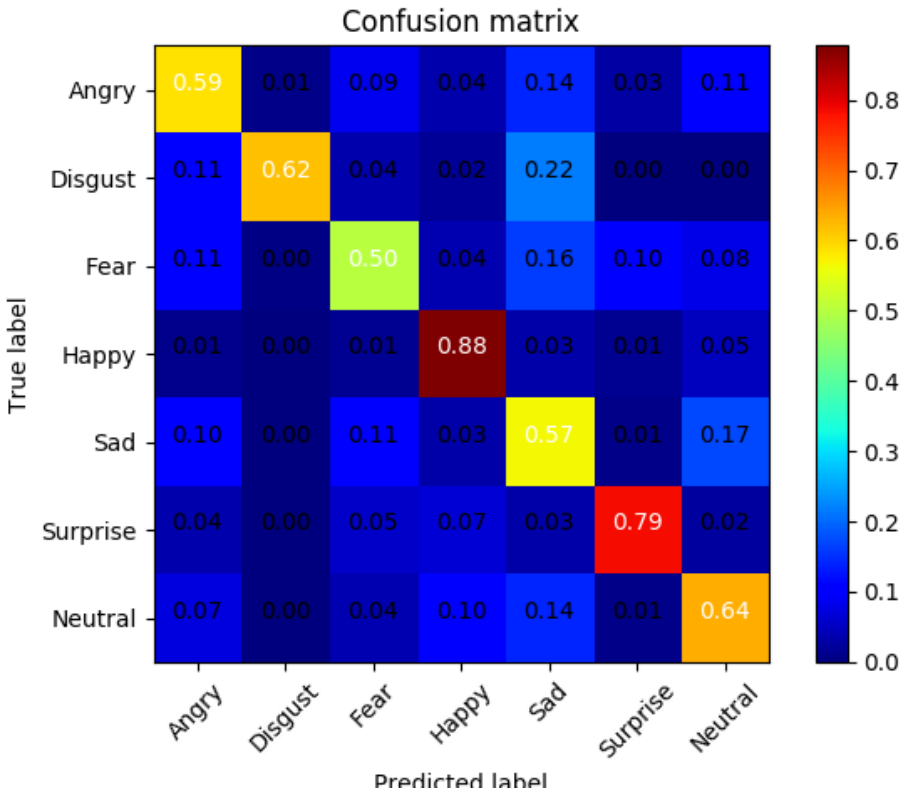
Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 1024)	2360320
dense_2 (Dense)	(None, 1024)	1049600
dropout_1 (Dropout)	(None, 1024)	0
dense_3 (Dense)	(None, 1024)	1049600
dropout_2 (Dropout)	(None, 1024)	0
dense_4 (Dense)	(None, 1024)	1049600
dropout_3 (Dropout)	(None, 1024)	0
dense_5 (Dense)	(None, 1024)	1049600
dropout_4 (Dropout)	(None, 1024)	0
dense_6 (Dense)	(None, 1024)	1049600
dropout_5 (Dropout)	(None, 1024)	0
dense_7 (Dense)	(None, 7)	7175

Total params: 7,615,495.0
Trainable params: 7,615,495.0
Non-trainable params: 0.0

在參數量接近的情況下，DNN 的正確率在 kaggle 上只有 41.376%，與 CNN 相差非常多，並且在訓練過程中可以看到 validation accuracy 一直上下浮動，除了前幾個 epoch 以外沒有顯著的進步趨向。

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：



我從 training data 中隨機取出 3500 筆作為 validation data，繪出 confusion matrix

從 confusion matrix 中可以看出，大部分的(除了 happy 和 surprise)表情都有蠻高的機率被誤判成 sad，在我看來這是合理的，因為我們可以大致將表情分成正面(happy, surprise)和負面(sad, angry, disgust, fear)，可能 sad 擁有較多負面表情的共通點因此容易被 activate。

對於正面表情來說，surprise 容易被判斷為 happy，而 happy 較不容易被判斷為 surprise，猜測這代表 happy 擁有的特點比較 general，而 surprise 可能有一些特別的部分只有 surprise 才有因此不容易將 happy 分為 surprise

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：

我從 training data 中選出 32 張圖，將原圖以及其濾出 gradient 較高部分的圖並排(原圖在左而 saliency map 在右)

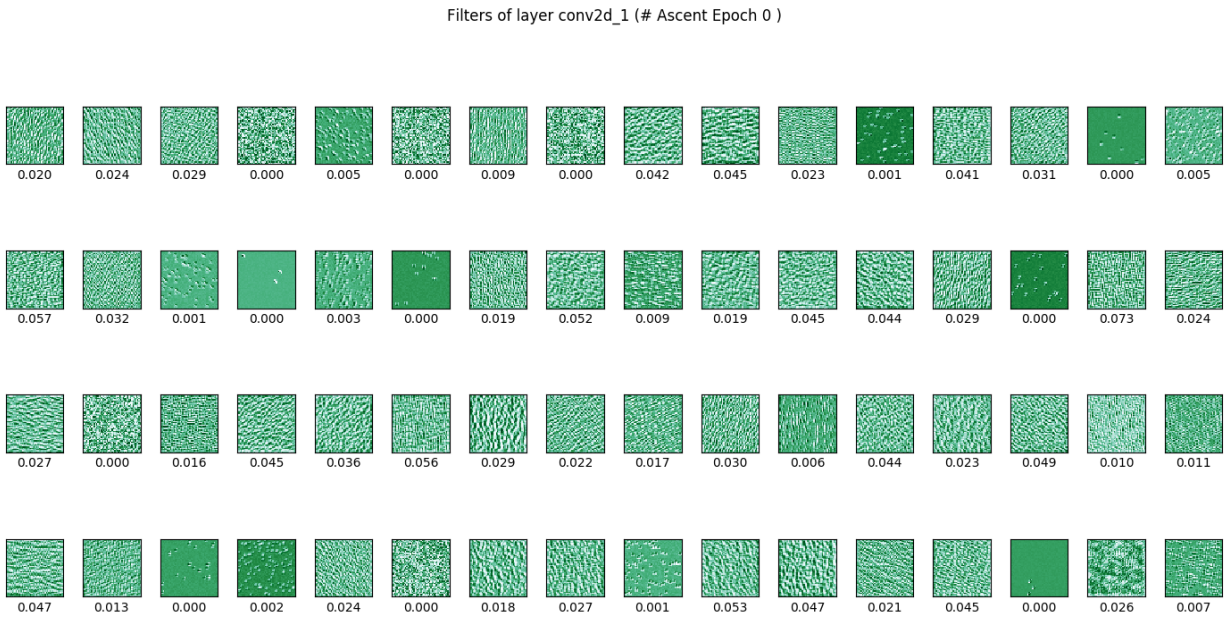
可從上圖的比較中看出，對 output 影響最大的部分主要集中在「眼睛」「眉毛」和「嘴巴」，這些也恰好是在心理學上我們判斷情緒時主要會關注的區域。簡單來說，要是我們想要用較少的臉部特徵傳達某種情緒，我們會選擇以「眼睛」、「眉毛」以及「嘴巴」來表示(例如表情符號的組成)

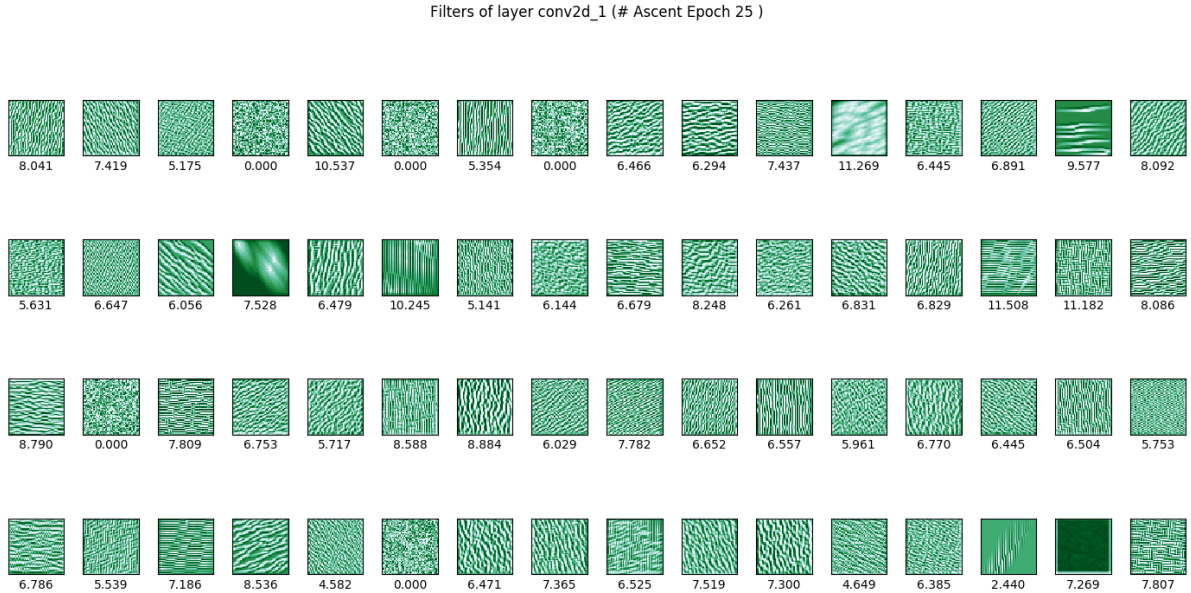
因此我認為這次我的 CNN model 是成功的，CNN model 很好的偵測了重要的臉部區域並且以這些區域作為判斷依據。

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：

   我的 gradient ascent 使用的 learning rate 為 1，以下輸出第一層(conv2d_1)在最一開始以及第 25 epoch 的 gradient ascent 結果，選擇全部的 64 個 filter



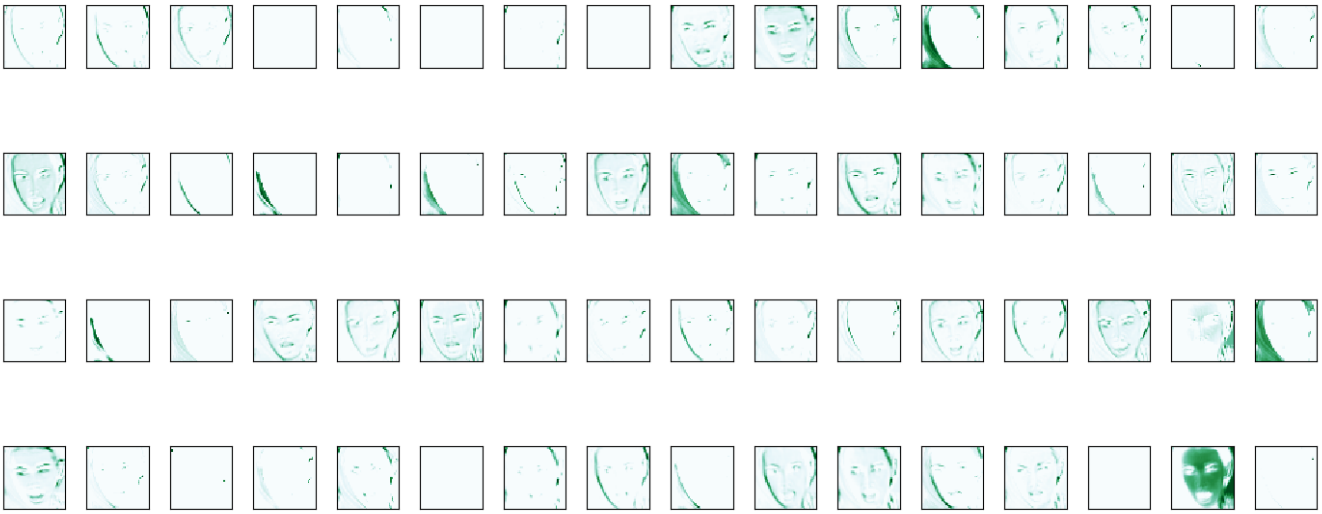


圖片下標為我們想要遞增的 value(loss 的相反)，可視為該圖片與 filter 的相似程度，由此可以看到經過 25 個 epoch 之後大部分的圖片都被 ascent 成能激活該 filter 的形狀。

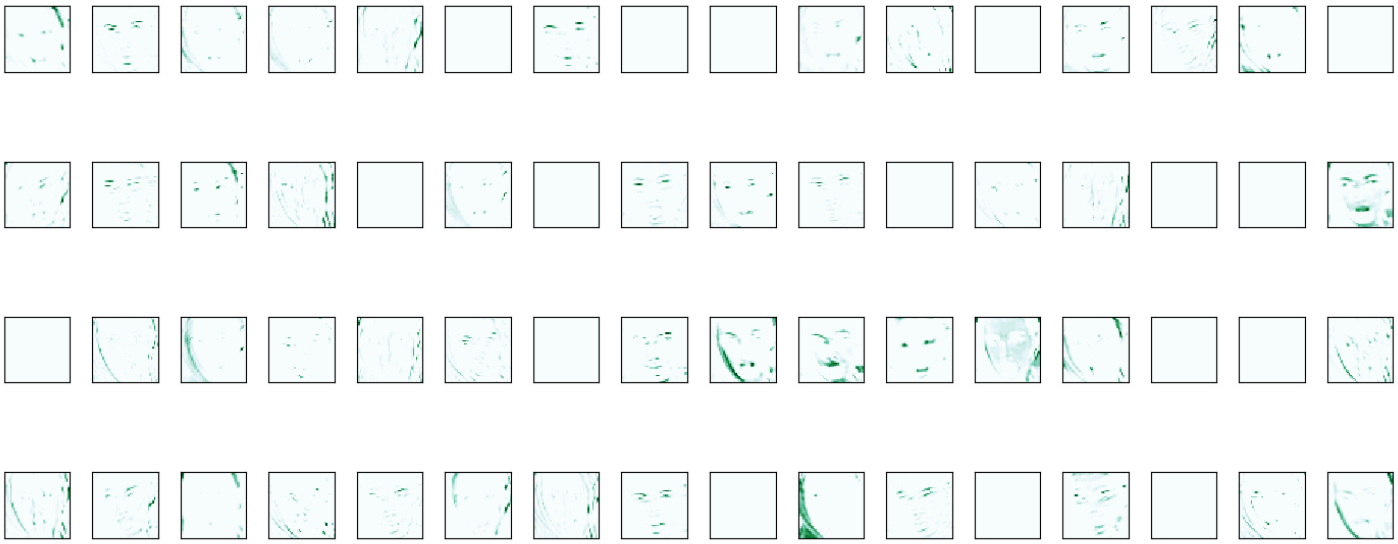
對於特定的圖片，以下分別輸出 conv2d_1,conv2d_2,conv2d_3 的 output(有經過 relu)

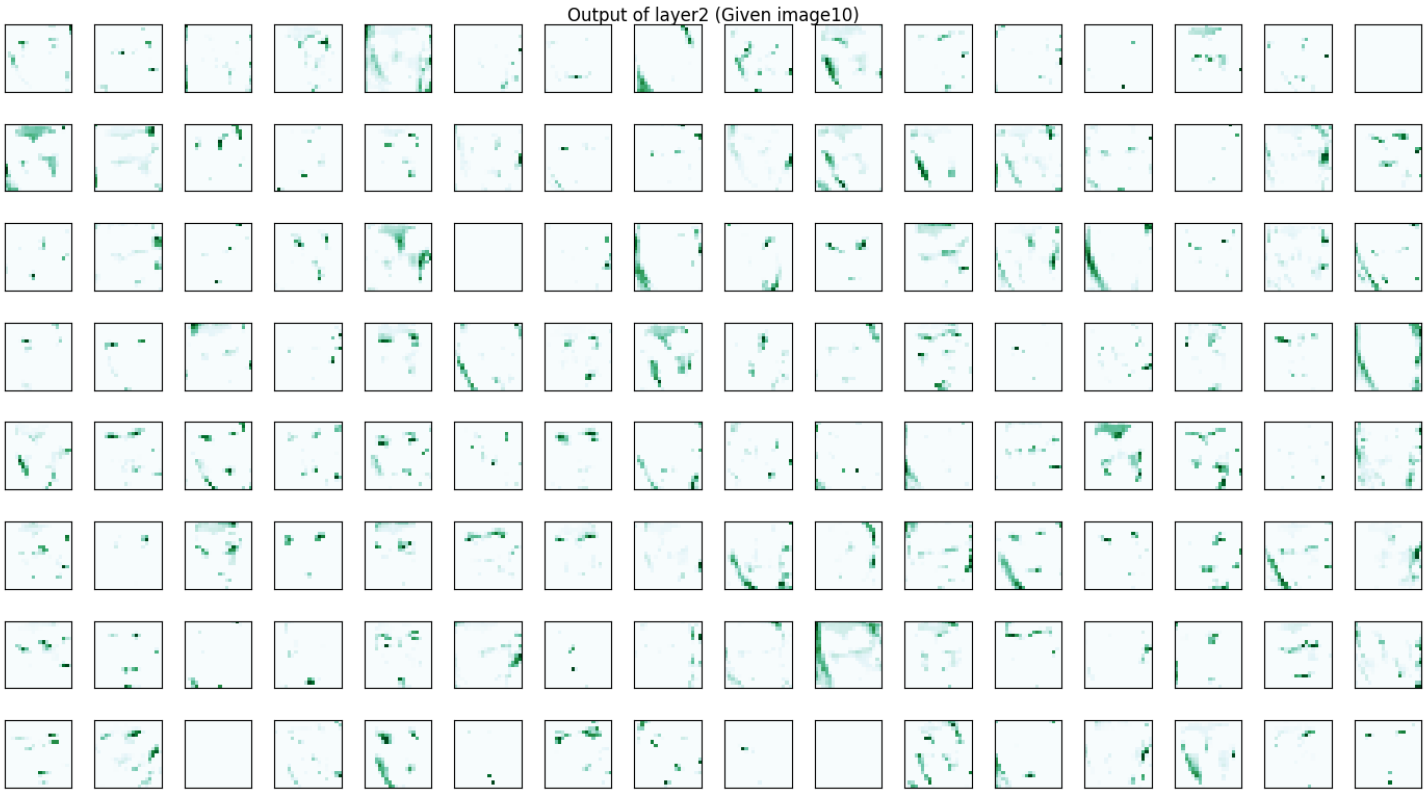
來看各層 filter 會激活圖片中的哪些部分

Output of layer0 (Given image10)



Output of layer1 (Given image10)

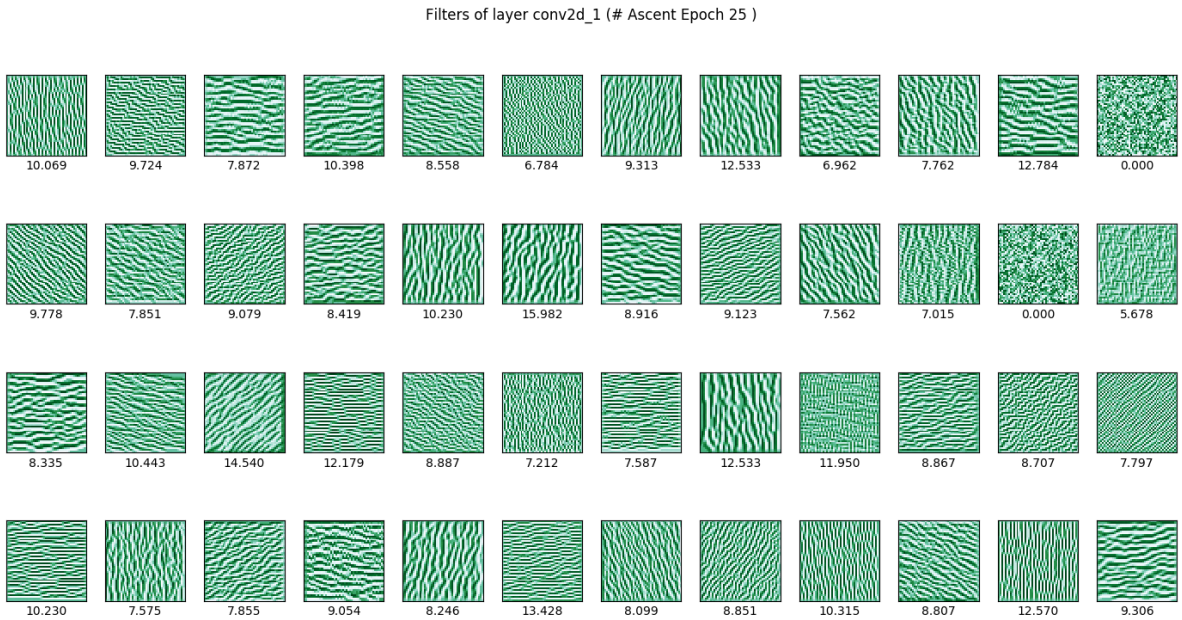




可以看出 filter 大部分濾出來的部分為眼睛和嘴巴，符合我們第四題 saliency map 的結果，另外由於 layer2(也就是 conv2d_3)是前面的 output 經過 maxpooling 的結果，因此圖片會有些許扭曲，但由於還不是太深(第三層)還能勉強看出臉部輪廓。

[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？

(1)首先我實作了一個比較淺層效果比較差的 CNN model (kaggle 上分數為 54.667%)，對第一層的 filter 做 gradient ascent 後效果如下



和前面較好的 model 比較起來，這個 model 的 filter 感覺只是簡單的濾出直線橫線與斜線而已，並沒有像之前的 model 有一些 filter 看起來很不規則或是有光影變化。

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

我從 training data 中隨機選出 4000 筆 data 來當作 semi-supervised 的資料，相比於沒有做 semi-supervised 在 kaggle 上的正確率 66.425%，做了 semi-supervised 之後的準確率提高到了 66.871%，但這個提高並沒有十分顯著，因此我認為這不能代表 semi-supervised 的 training 可以提升準確率。我猜測若用全部的 training data 去訓練並將 testing data 用來做 semi-supervised learning 可能會提升一些準確率，但礙於時間因素，我會在上傳作業之後再來做這個測試。

註:

雖然應該只會運行 hw3_test.sh，若助教想要跑其他的.py 檔的話請將 train.csv 和 test.csv 放在 data 資料夾內(因為並沒有上傳 dataset)