Homework 3 : Report

110024516 統研所 邱繼賢

1 Introduction

1.1 Abstract

本次作業使用 Finger signs dataset，建構一個以分類爲目的的 convolutional neural network (CNN) model，對每張圖片中伸出的手指個數 $(0 \sim 5)$，也就是 label 進行預測，並且希望在 testing data 上的預測精準度可以達到 93%。

1.2 Data

Finger signs dataset 可以再區分爲以下四個部分：

1. train image : 1080 張圖片，每張圖片維度： (64,64,3)，數值落在 $[0,255]$ 區間
2. train label : 1080 個落在 $[0,5]$ 的整數值，代表每張 train image 所對應的 label
3. test image : 120 張圖片，每張圖片維度： (64,64,3)，數值落在 $[0,255]$ 區間
4. test label : 120 個落在 $[0,5]$ 的整數值，代表每張 test image 所對應的 label

將 train & test image 的數值：$x \leftarrow 2(x/255) - 1$ 映射到 $[-1,1]$ 區間，並再將 train & test label 利用 One-hot encoding 的方式從向量轉換成行數爲 6 的 matrix。

1.3 Cross-Validation

將 (train image , train label) 隨機的分割成五等分，在以下的模型訓練中取其中的四份作爲 training set，另一份作爲 validation set，每一次都選擇不同份作爲 validation set，重複此步驟五次，即稱爲 5-fold cross-validation。

利用 cross-validation 決定最終模型有以下優點：

1. 所有的 training data 都一樣次數的被當作 train 或是 validation 使用，平均上來說我們最終決定的 model 有使用到所有 training data 的資訊；相對來說，只切一次 validation set 的方式則會遺失部分資訊來 train model。
2. 5-fold cross-validation 因爲重複做了五次，並用五次的結果平均來決定最終 model，可以降低某一個 validation set 切得相對極端造成對 model 的影響。

2 Model

2.1 Architecture

模型架構如 Figure 1 所示，先將 (64,64,3) 的 tensor 接 convolutional neural network (將 Conv2d + BatchNorm + ReLU + MaxPool 四個動作視爲一層 CNN layer)，然後轉換成向量後再接 fully connected neural network，最後 output layer 爲一個長度爲 6 的向量，利用 softmax function 計算出六類別的 predicted probabilities，取其中數值最大者即爲 predicted label。會選擇先利用 CNN 來接 image tensor 是因爲圖片是一種 local dependence 很強的資料類型，如果一開始就直接將 tensor 拉成向量來處理的話，就會失去此性質。

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 64, 64]	1,792
BatchNorm2d-2	[-1, 64, 64, 64]	128
ReLU-3	[-1, 64, 64, 64]	0
MaxPool2d-4	[-1, 64, 32, 32]	0
Conv2d-5	[-1, 128, 32, 32]	73,856
BatchNorm2d-6	[-1, 128, 32, 32]	256
ReLU-7	[-1, 128, 32, 32]	0
MaxPool2d-8	[-1, 128, 16, 16]	0
Conv2d-9	[-1, 256, 16, 16]	295,168
BatchNorm2d-10	[-1, 256, 16, 16]	512
ReLU-11	[-1, 256, 16, 16]	0
MaxPool2d-12	[-1, 256, 8, 8]	0
Conv2d-13	[-1, 512, 8, 8]	1,180,160
BatchNorm2d-14	[-1, 512, 8, 8]	1,024
ReLU-15	[-1, 512, 8, 8]	0
MaxPool2d-16	[-1, 512, 4, 4]	0
Conv2d-17	[-1, 1024, 4, 4]	4,719,616
BatchNorm2d-18	[-1, 1024, 4, 4]	2,048
ReLU-19	[-1, 1024, 4, 4]	0
MaxPool2d-20	[-1, 1024, 2, 2]	0
Conv2d-21	[-1, 2048, 2, 2]	18,876,416
BatchNorm2d-22	[-1, 2048, 2, 2]	4,096
ReLU-23	[-1, 2048, 2, 2]	0
MaxPool2d-24	[-1, 2048, 1, 1]	0
Linear-25	[-1, 1024]	2,098,176
ReLU-26	[-1, 1024]	0
Linear-27	[-1, 512]	524,800
ReLU-28	[-1, 512]	0
Linear-29	[-1, 6]	3,078

Total params: 27,781,126
Trainable params: 27,781,126
Non-trainable params: 0

Input size (MB): 0.05
Forward/backward pass size (MB): 12.82
Params size (MB): 105.98
Estimated Total Size (MB): 118.84

Figure 1: Architecture of image classification model

訓練此模型時，loss function 採用 Cross Entropy loss 用以比較 predicted probabilities 和 real label 之間的差異，optimization 的方式則是使用 Adam。

Figure 2 和 Figure 3 呈現出表現最好的模型 (setting of hyperparameter is shown in **2.2**) 在 50 epochs 下，5-fold CV 的 train and validation loss (accuracy) curve，可以看出在差不多 40 epoch 後 loss 和 accuracy 都趨於穩定，代表此時模型基本上已經收斂。

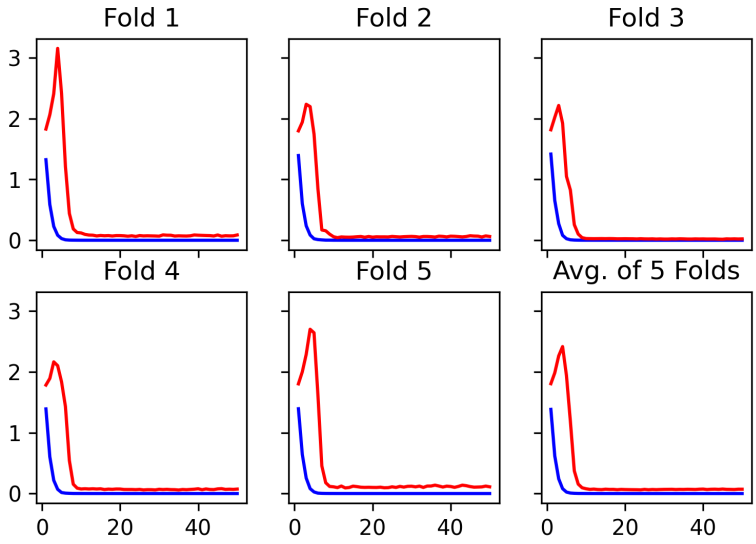

Figure 2: Train (blue) and Validation (red) loss of best model

2.2 Hyperparameter

Epoch, batch size, learning rate 等參數主要是影響 gradient loss 收斂的速度和穩定性，可以藉由觀察 train and validation loss curve 來做調整，對模型的整體表現影響不是那麼直接，以下主要探討 number of CNN layers, number of fully connected layers, kernel size, zero padding 等參數的改變對模型有何影響，並觀察模型在 5-fold CV 下 validation loss 的差異：

1. 首先觀察 number of CNN layers，結果如 Table 1，可以發現在 6 層的時候平均上表現最好，故決定 CNN 層數 = 6

2. 再來觀察 number of FC layers，結果如 Table 2 所示，可以發現在 3 層時平均上表現最好，再增加層數表現反而變差，可能有 overfitting 的狀況，故決定 FC 層數 = 3

3. 最後觀察 kernel size & zero padding，結果如 Table 3 所示，發現在 (3,1) 時表現最好，故決定 (kernel size , zero padding) = (3,1)

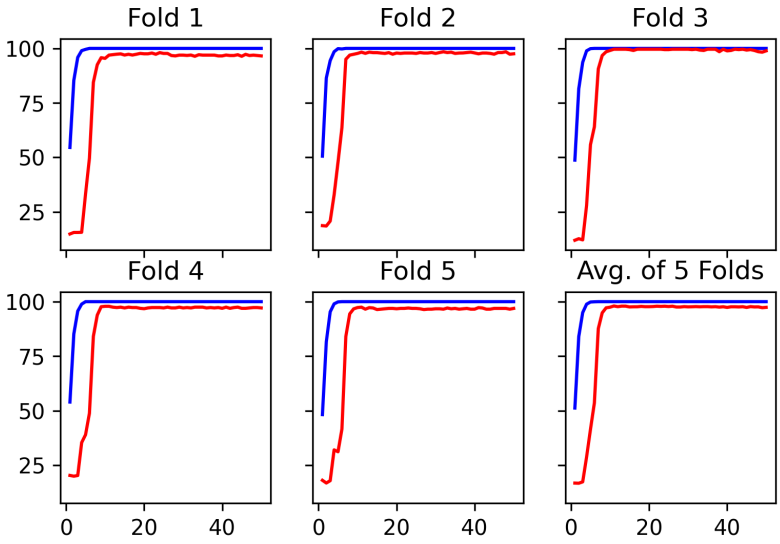

Figure 3: Train (blue) and Validation (red) accuracy of best model

	2	**3**	**4**	**5**	**6**
Fold 1	0.223	0.144	0.101	0.098	0.086
Fold 2	0.112	0.078	0.069	0.054	0.048
Fold 3	0.170	0.122	0.084	0.072	0.033
Fold 4	0.224	0.143	0.123	0.124	0.082
Fold 5	0.241	0.163	0.133	0.098	0.131
Average	0.194	0.130	0.102	0.089	0.076

Table 1: Validation loss in different number of CNN layers

	1	**2**	**3**	**4**	**5**
Fold 1	0.086	0.070	0.054	0.057	0.079
Fold 2	0.048	0.039	0.033	0.038	0.043
Fold 3	0.033	0.048	0.022	0.017	0.023
Fold 4	0.082	0.064	0.062	0.110	0.072
Fold 5	0.131	0.132	0.104	0.090	0.145
Average	0.076	0.071	0.055	0.063	0.072

Table 2: Validation loss in different number of FC layers

最後決定模型的各項 hyperparameters 如 Table 4 所示：

2.3 Final Model

在 **2.2** 中利用 5-fold cross-validation 決定出了各項 hyperparameters，此時我們

	(3,1)	**(5,2)**	**(7,3)**	**(9,4)**
Fold 1	0.054	0.046	0.083	0.051
Fold 2	0.033	0.030	0.038	0.048
Fold 3	0.022	0.033	0.052	0.101
Fold 4	0.062	0.097	0.118	0.123
Fold 5	0.104	0.086	0.087	0.074
Average	0.055	0.058	0.076	0.079

Table 3: Validation loss in different (kernel size , zero padding)

Num of CNN layers	6
Num of FC layers	3
kernel size	3
stride	1
zero padding	1
epoch	50
batch size	128
learning rate	1e-4
weight decay	1e-5

Table 4: Hyperparameter of best model

有五個模型，而要決定出最後的一個 final model 則是將所有的 training data 合併 (沒有切 validation)，然後利用全部的 training data 在 **2.1** 的架構和 **2.2** 的參數下重新再訓練一個模型，然後計算此模型在 testing data 上的 accuracy = 99.17% (可藉由執行 `test.py` 驗證)。

2.4 Pre-trained Model

接下來訓練 pre-trained model : AlexNet 來處理 image classification 的問題，testing accuracy = 96.67%，表現並不如前面我們自己寫的模型表現，但是 pre-train model 本身的泛化能力(generalized ability) 較佳，適合應對各種不同的資料，但對於特定的資料，如本題，表現可能就不如我們自己寫的模型，如果是處理更大量的資料或是需要建構參數更多的模型時，pre-train model 的優勢就會體現出來，而且 pre-train model feature extraction 的效果比較好，能夠提取出較好的特徵，再加上使用 pre-train model 比起 random initialization 的參數會來得更好。