

Homework 4 Report

學號：r06521605

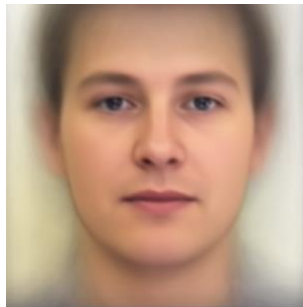
系級：土木所電輔組碩一

姓名：許舜翔

A. PCA of colored faces

A.1.

以下為所有臉的圖片平均：



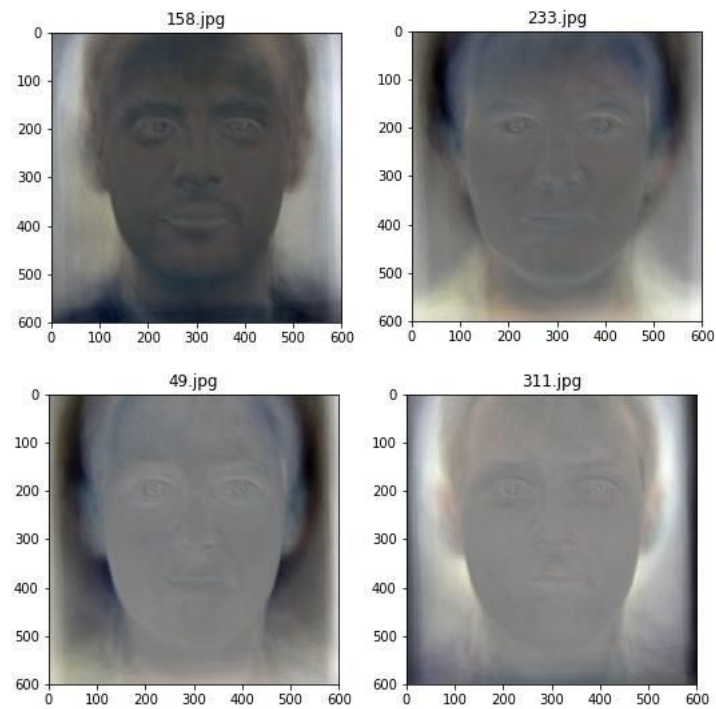
A.2.

以下為前四大 eigenvalues 對應之 eigenvectors 所畫出的圖像：



A.3.

以下為隨機挑選四張照片，並用前四大 eigenfaces 進行 reconstruction 的結果：



A.4.

以下依序為前四大的 eigenfaces 所佔的比重：

```
1 | ratio: 4.1%
2 | ratio: 2.9%
3 | ratio: 2.4%
4 | ratio: 2.2%
```

B. Image clustering

B.1.

降維方法	分類方式	Public score	Private score
Pca (取前五個 eigenvector 進行降 維)	Kmeans (n_cluster = 7)	0.93348	0.93332
Auto encoder (sample code)	Kmeans (n_cluster = 2)	0.97406	0.96862

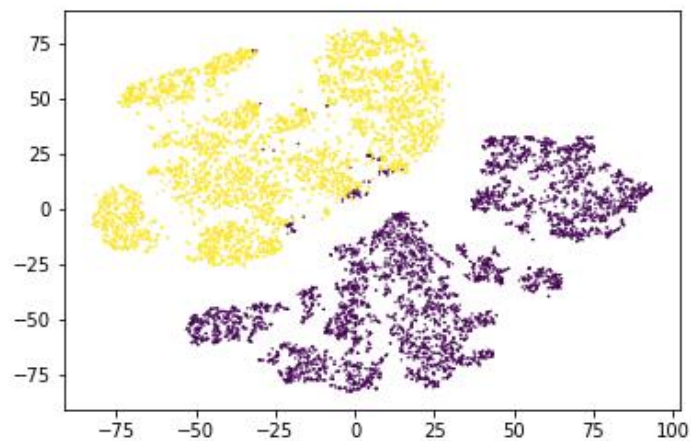
Auto encoder (sample code 的架構, lr = 1e-4, epochs = 60)	Kmeans (n_cluster = 2)	0.69336	0.69275
----------------------------------------------------------------	---------------------------	---------	---------

以 row2 & row3 可以看出 learning rate 的值設定太小導致 training loss 還未下降到最低點。

以 row1 & row2 推論說 autoencoder (784->32) 比起 pca 前五個 eigenvector 所取得的 feature 要來得更佳，以下為兩種方法對同一張圖 reconstruct 的結果：

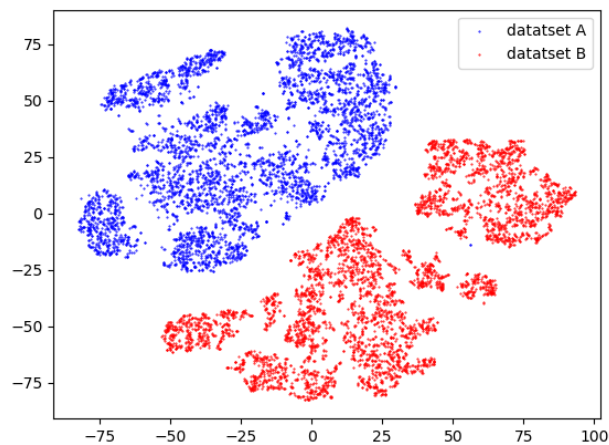
B.2.

以下為透過 kmeans 及 autoencoder 的方式預測出的結果，並以 TSNE 投影到二維平面上：



B.3.

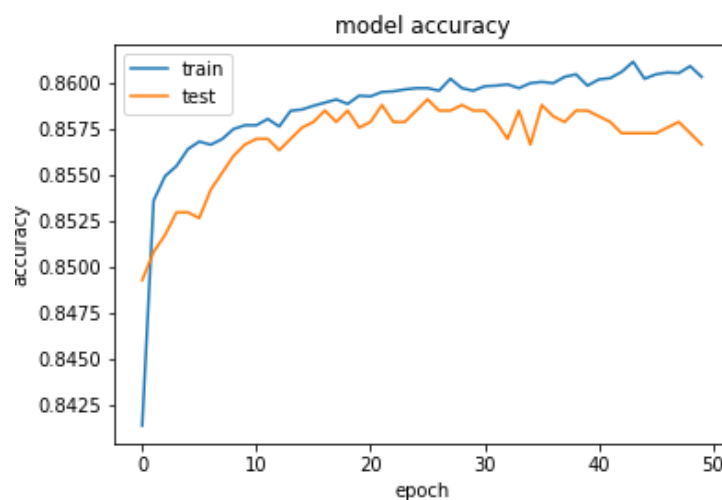
以下為前 5000 筆跟後 5000 筆分布的結果，同樣是利用 TSNE 的方法。可以發現給前題自行預測的結果，在邊界上會有一些混淆的部分，但也僅是少部分，與 Kaggle 上預測約 0.97 的精度相符。



C. Ensemble learning

C.1.

本題我選用 HW2 來實作 ensemble 的方法，當初模型 training 的 history 圖表如下：



可以發現 training 的準確率皆高於 validation 的準確率，故推論有 overfitting 的現象，故之後結合不同 model 結果時，選擇用 bagging 中最後將各模型結果取平均，在判斷屬於二元分類中的哪一類，這次使用了三種不同架構的模型（包含當初上傳的模型），分述如下：

1. Model_1（當初的模型，training histroy 如上）

```

=====
Layer (type)                 Output Shape              Param #
=====
dense_1 (Dense)              (None, 39)                1560
=====
dense_2 (Dense)              (None, 1)                 40
=====
Total params: 1,600
Trainable params: 1,600
Non-trainable params: 0
=====

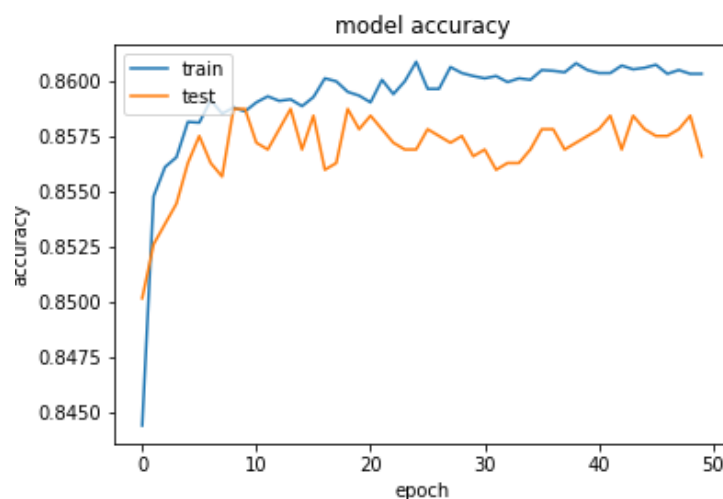
```

2. Model_2 (原先 DNN 結構多加一層)

```

=====
Layer (type)                 Output Shape              Param #
=====
dense_3 (Dense)              (None, 39)                1560
=====
dense_4 (Dense)              (None, 16)                640
=====
dense_5 (Dense)              (None, 1)                 17
=====
Total params: 2,217
Trainable params: 2,217
Non-trainable params: 0
=====

```

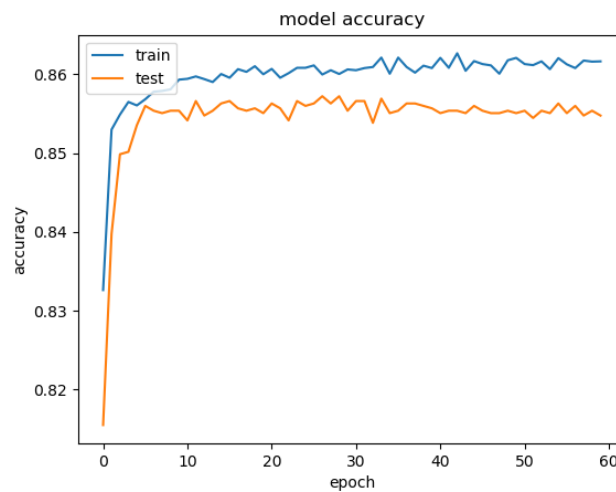


3. Model_3 (多加一層 BatchNormlization)

```

=====
Layer (type)                 Output Shape              Param #
=====
dense_6 (Dense)              (None, 39)                1560
=====
batch_normalization_1 (Batch (None, 39)          156
=====
dense_7 (Dense)              (None, 1)                 40
=====
Total params: 1,756
Trainable params: 1,678
Non-trainable params: 78
=====

```



最後 public/private 結果：

- 原本的 0.85933/0.85738
- Ensemble 後的 0.86031/0.85996

有些微的精度提升，並沒有顯著的成長，推測應為模型間結構類似，故透過 average output 也較難有效地解決 overfitting 的問題。