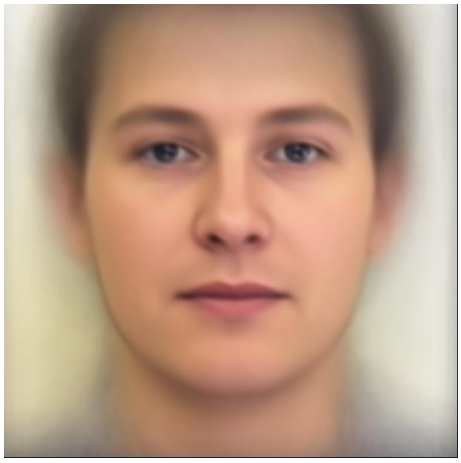
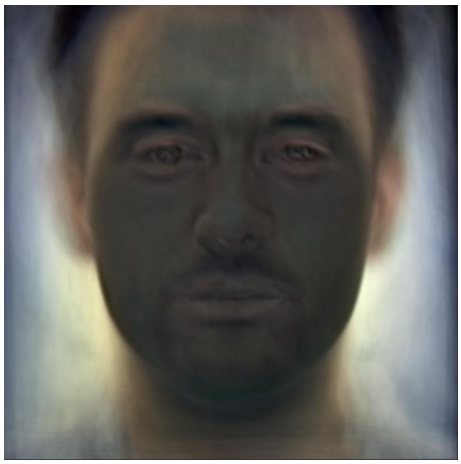
學號：B04701232　系級：　工管三　姓名：陳柔安

A. PCA of colored faces

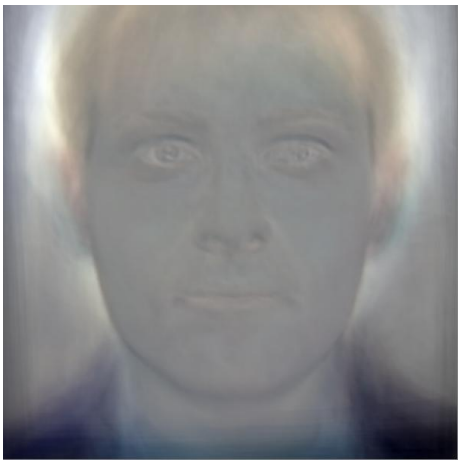
1. (.5%) 請畫出所有臉的平均。



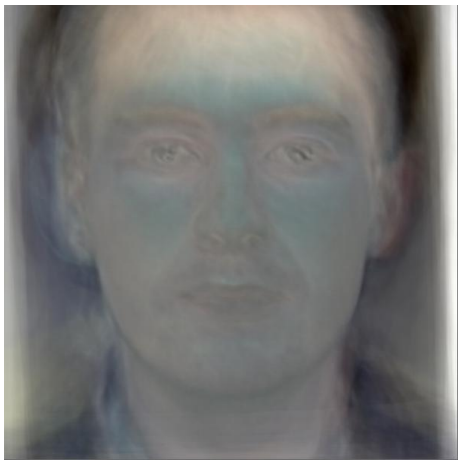
2. (.5%) 請畫出前四個 Eigenfaces，也就是對應到前四大 Eigenvalues 的 Eigenvectors。



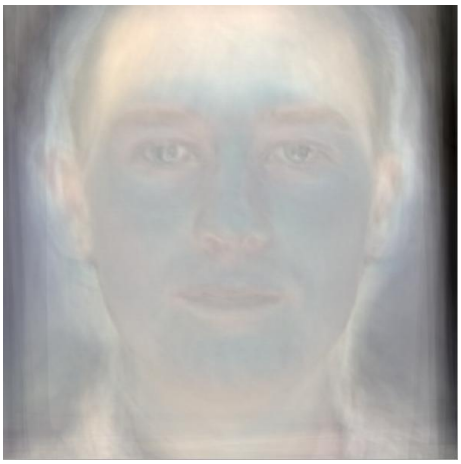
eigenface_1



eigenface_2

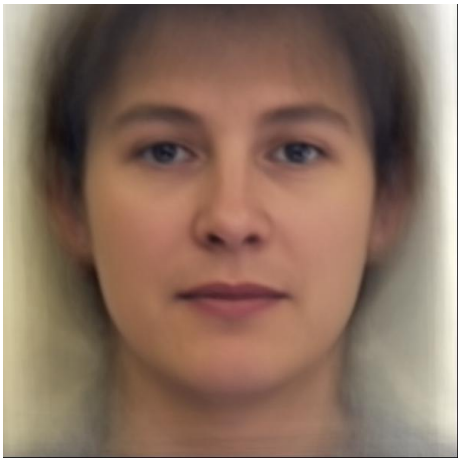


eigenface_3

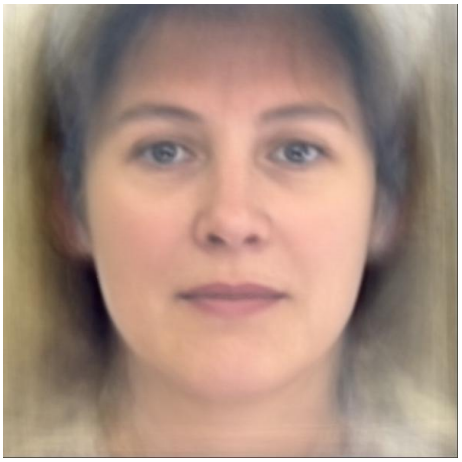


eigenface_4

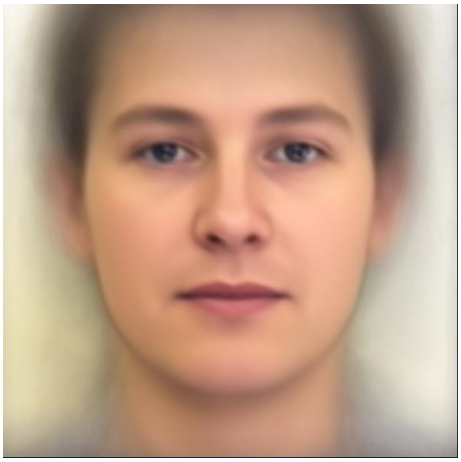
3. (.5%) 請從數據集中挑出任意四個圖片，並用前四大 Eigenfaces 進行 reconstruction，並畫出結果。



reconstruct_0



reconstruct_4



reconstruct_7

reconstruct_9

4. (.5%) 請寫出前四大 Eigenfaces 各自所佔的比重，請用百分比表示並四捨五入到小數點後一位。
   4.1%, 2.9%, 2.4%, 2.2%

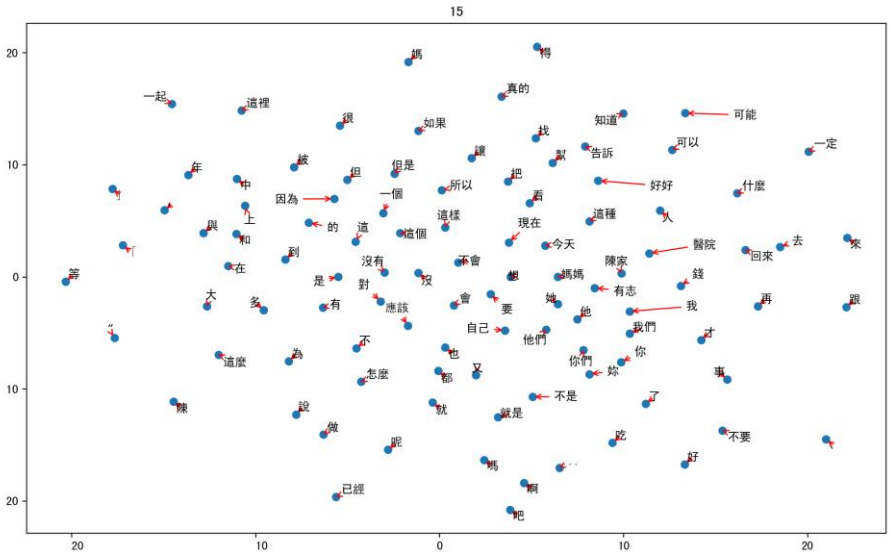
Visualization of Chinese word embedding

1. (.5%) 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義。

   使用 gensim 中的 word2vec，size=100(一個詞 100 維代表)，

   min_count=4500(在語料中要出現 4500 次以上才會被計算詞向量)

2. (.5%) 請在 Report 上放上你 visualization 的結果。



3. (.5%) 請討論你從 visualization 的結果觀察到什麼。

相近意思的詞彙會在比較相近的地方，例如圖片右方的來、去、回來等皆維來去動詞，意思相近；而較靠近下面的嗎、吧、啊等是結尾驚嘆詞，意思也相近。

C. Image clustering

1. (.5%) 請比較至少兩種不同的 feature extraction 及其結果。(不同的降維方法或不同的 cluster 方法都可以算是不同的方法)

方法一：autoencoding(dim=64) + kmeans

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 784)               0
_________________________________________________________________
dense_2 (Dense)              (None, 256)               200960
_________________________________________________________________
dense_3 (Dense)              (None, 128)               32896
_________________________________________________________________
dense_4 (Dense)              (None, 64)                8256
_________________________________________________________________
dense_5 (Dense)              (None, 64)                4160
=================================================================
Total params: 246,272
Trainable params: 246,272
Non-trainable params: 0
_________________________________________________________________
```

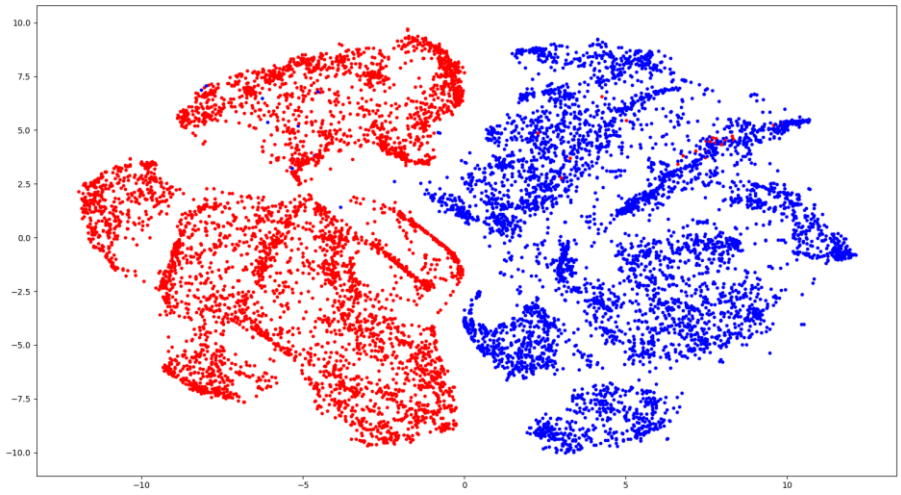
上圖為 encoder 的 model summary, decoder 是將 encoder 的 network 反過來。

準確率(kaggle)：0.82844/0.82880 (public/private)

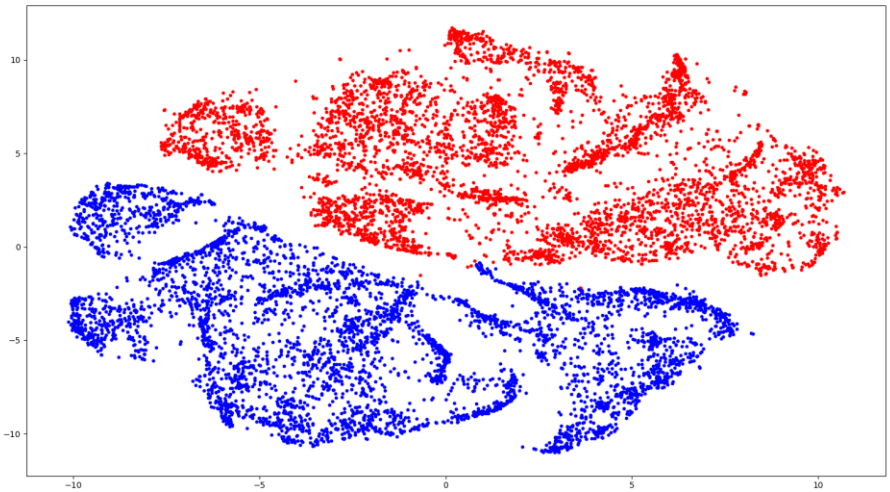
方法二：PCA(dim=64) + kmeans
準確率(kaggle)：0.02055/0.02045 (public/private)

方法一之表現遠好於方法二

2. (.5%) 預測 visualization.npy 中的 label，在二維平面上視覺化 label 的分佈。



3. (.5%) visualization.npy 中前 5000 個 images 跟後 5000 個 images 來自不同 dataset。請根據這個資訊，在二維平面上視覺化 label 的分佈，接著比較和自己預測的 label 之間有何不同。



由這兩題結果可發現雖然預測能有效地將相近的分成同一群，但兩張圖片的點

的分布有些許不同、分線也十分不同，推測可能是因為繪圖時做兩次 TSNE，投影上有些誤差。預測的結果藍色區域會有少許紅點點，紅色區域也有藍點點，代表分群誤差的部分，而實際的 label 就沒有這個問題。