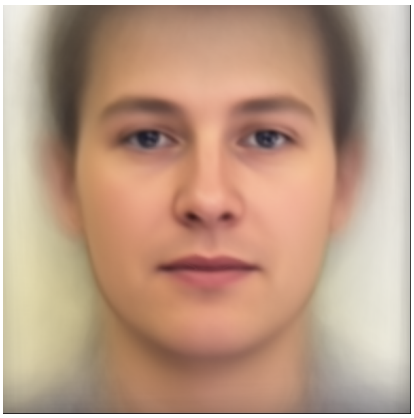
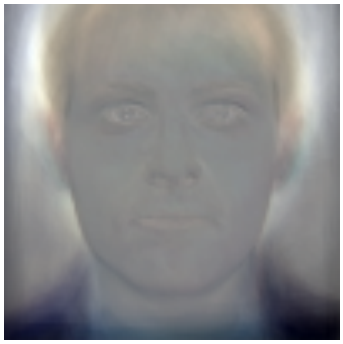
學號：B04901117 系級：電機三 姓名：毛弘仁

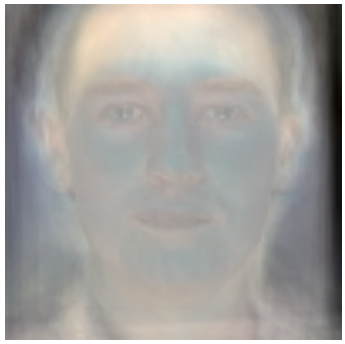
A. PCA of colored faces

1. **(.5%) 請畫出所有臉的平均。**

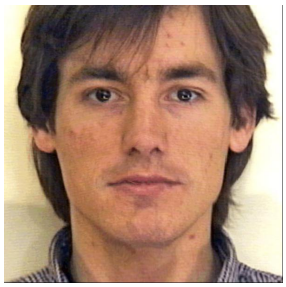
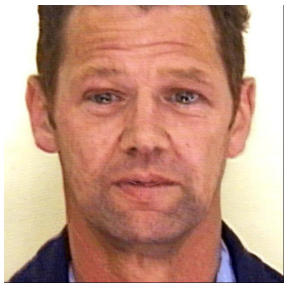


2. **(.5%) 請畫出前四個 Eigenfaces，也就是對應到前四大 Eigenvalues 的 Eigenvectors。**

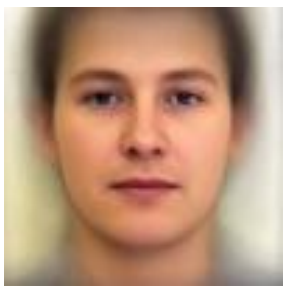


3. **(.5%) 請從數據集中挑出任意四個圖片，並用前四大 Eigenfaces 進行 reconstruction，並畫出結果。**







4. **(.5%) 請寫出前四大 Eigenfaces 各自所佔的比重 (explained variance ratio)，請四捨五入到小數點後一位。**

   0.042、0.030、0.024、0.022。

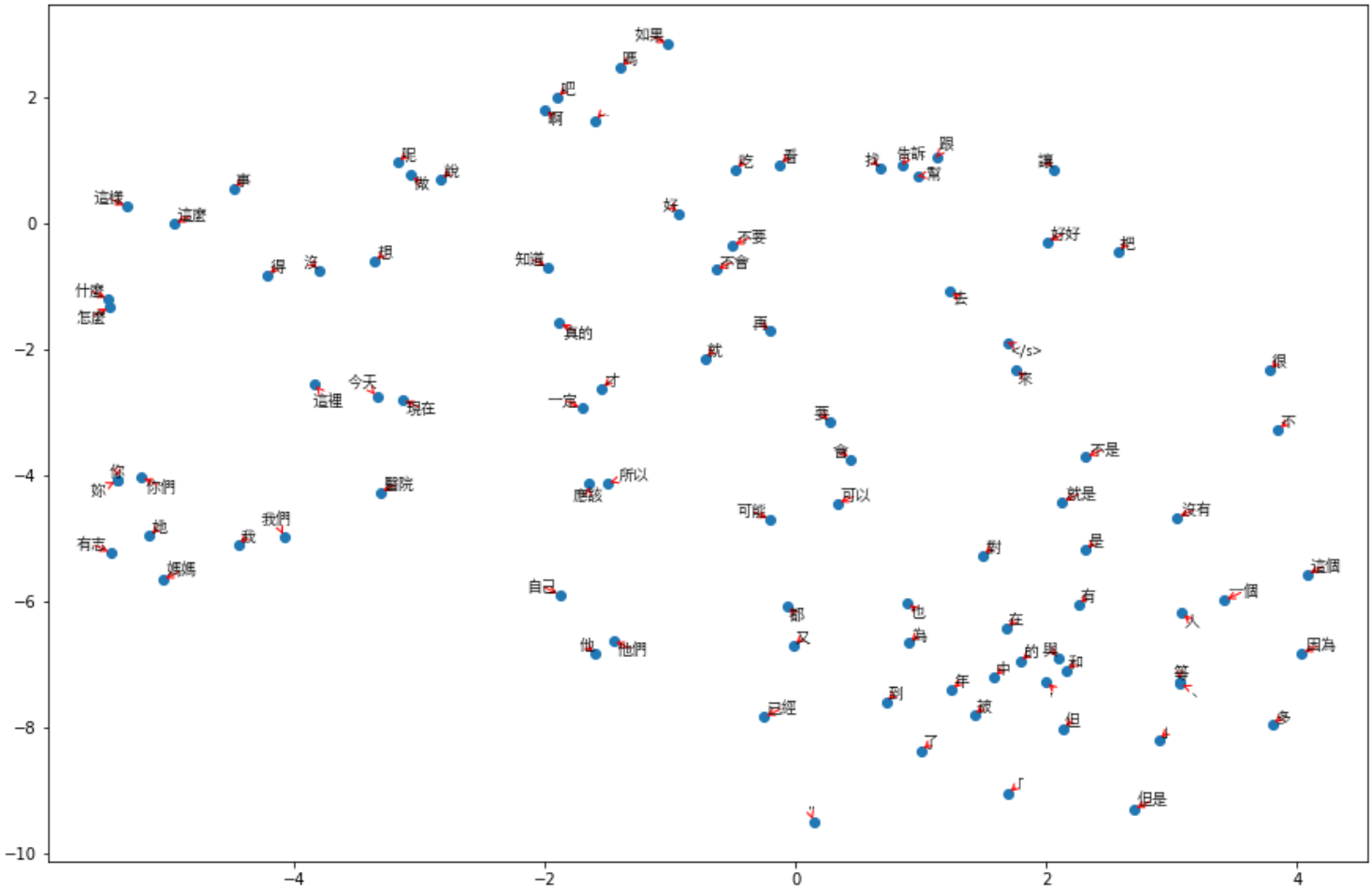
B. Visualization of Chinese word embedding

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。**

   我使用的是助教投影片裡提到的 word2vec，它的 Github 說是 Google word2vec 的 Python interface。這個 interface 並沒有特別做很多 documentation，只給 sample code，因此要調整參數，需要自己從 word2vec 的 command line 指令訊息（會自動跟著 Python interface 一起安裝）摸索。

   我調整的參數是根據助教投影片，min_count（dictionary 裡面的字至少要在 corpus 裡出現幾次）設成 5000，這樣需要做 t-SNE 和 visualization 的 word vector 比較少，我的電腦也才跑得動。

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。**

可以發現有些類似的 word 會聚集在一起，像是「這樣」、「這麼」、「什麼」、「怎麼」；「這裡」、「今天」、「現在」。

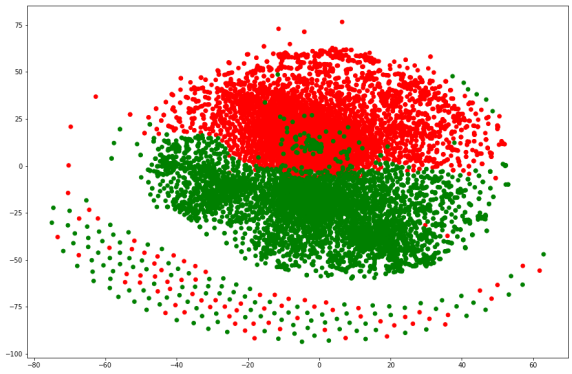
C. Image clustering

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)**

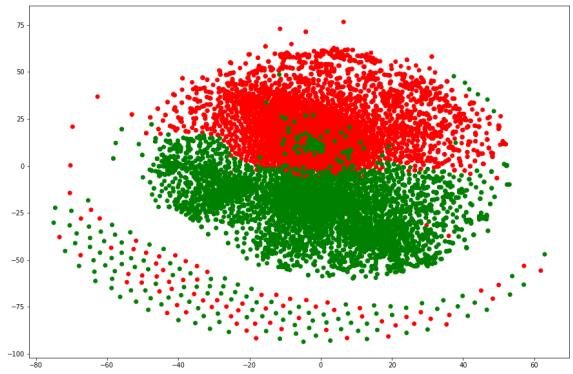
（一）看到圖片的兩類別是「手寫數字」和「衣物」，我想到一個 rule-based 的方法，就是看圖片當中，非黑色 pixel 的比例是否超過 0.3（也嘗試過使用 pixel value > 200 的比例，也就是「多白」，但是對於一些比較暗的衣物圖，像是襪子、鞋子，效果不好）。為了節省運算時間，我隨機抽圖片當中的 50 pixels 當統計樣本（降維），最後在 Kaggle 上得到 0.077 的 F1 score，不算太好。

（二）因為 rule-based 的降維方法不是很理想，所以我決定試著乖乖用 PCA 做做看。起初試了 n_components = 10, 50, 100，拿去做 k = 2 的 K-means clustering 效果都沒有很好，所以我直接跳到 n_components = 700（很接近 784 這個上限值），結果在 Kaggle 上竟然拿到 1.0 的 F1 score，實在很開心！

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



Prediction



Ground truth

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。**

這樣看起來，預測的 label 和 ground truth 應該是都一樣的，讓我 Kaggle 1.0 的分數又得到一些驗證。