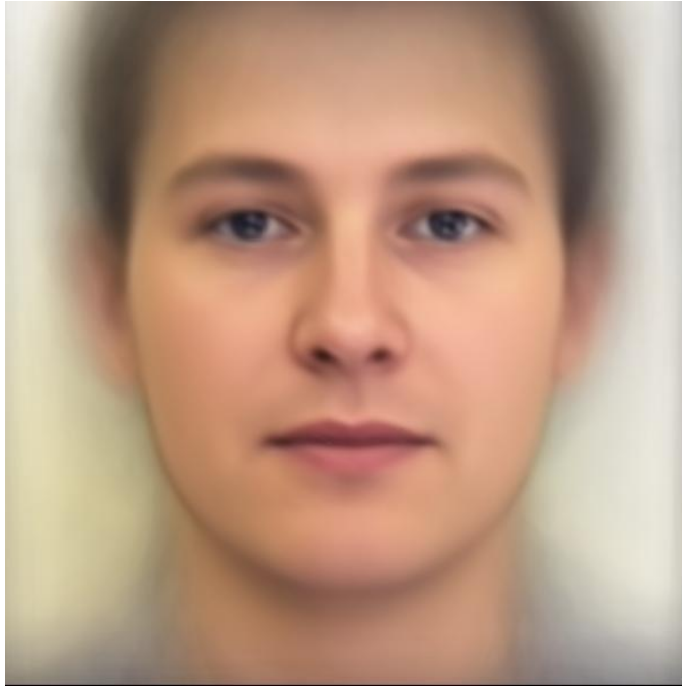


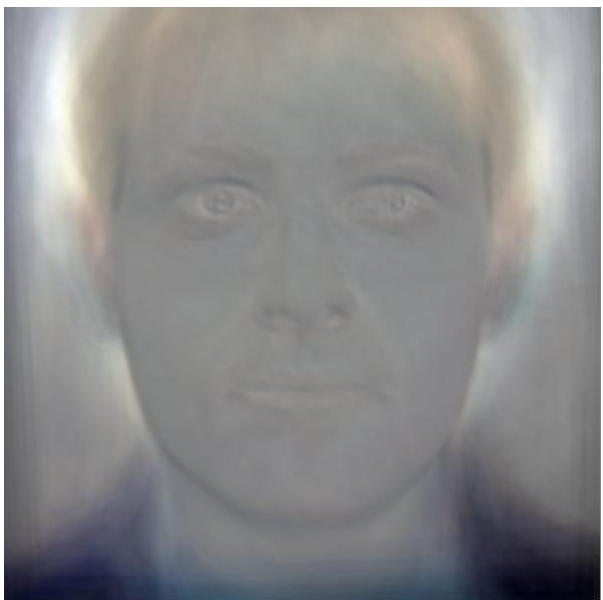
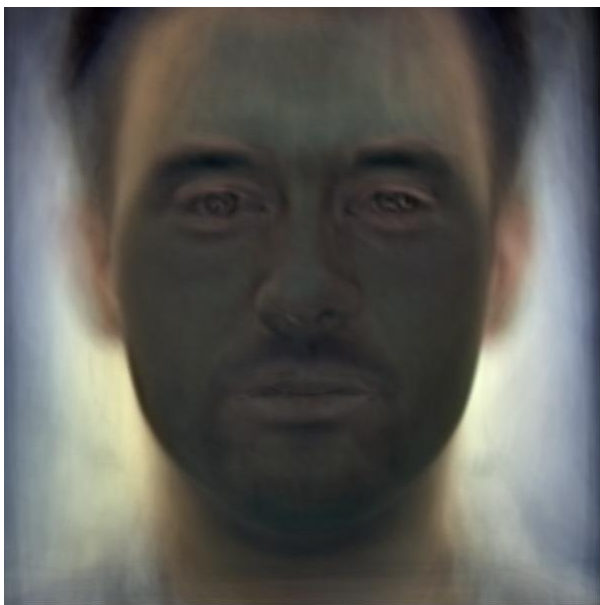
A. PCA of colored faces (Collaborators:無)

A.1. (.5%) 請畫出所有臉的平均。



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

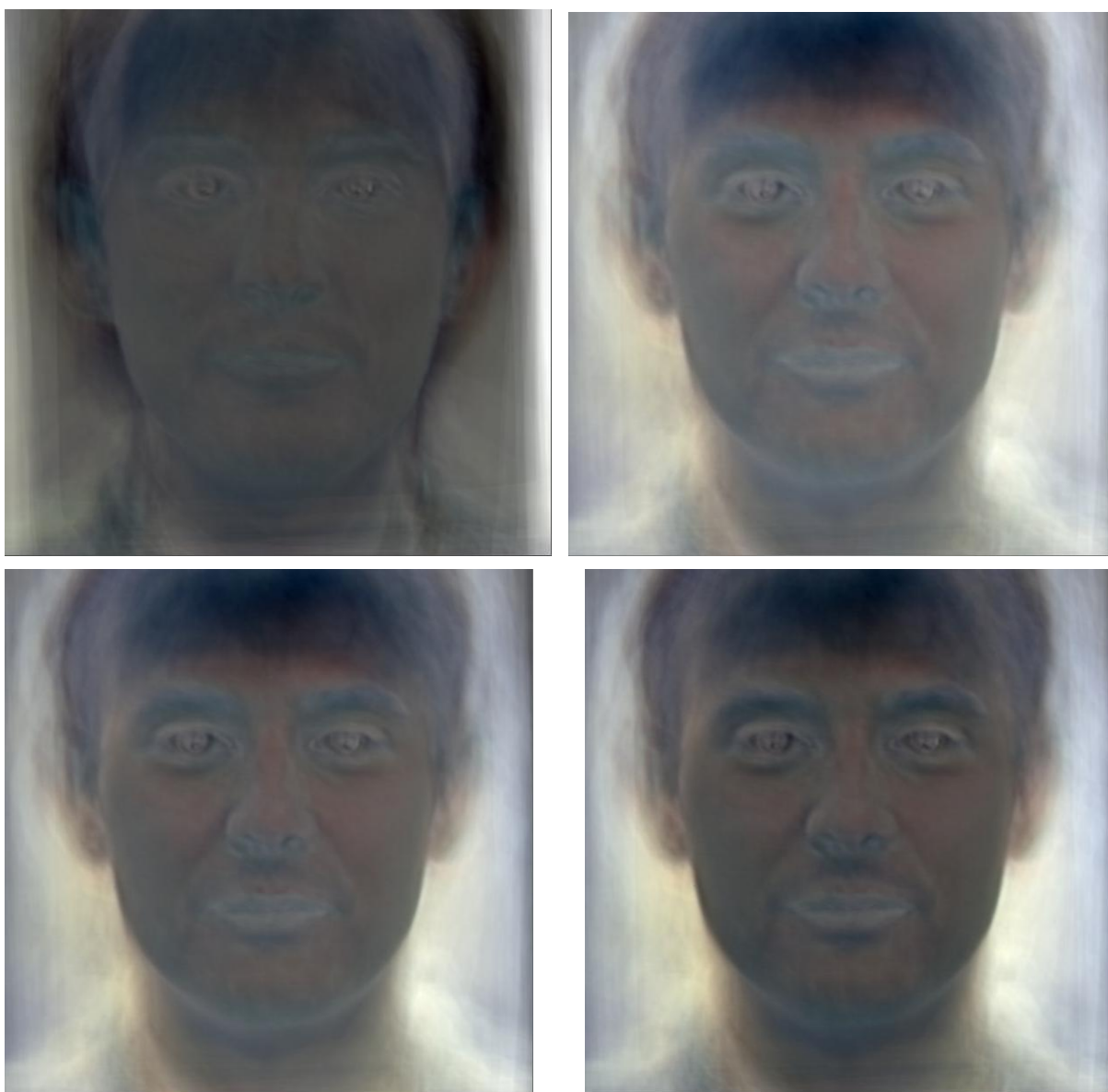
由左至右，由上至下分別為前四個 Eigenfaces。





A.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。

以下由左至右，由上至下分別為 0/50/100/150.jpg



- A.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。

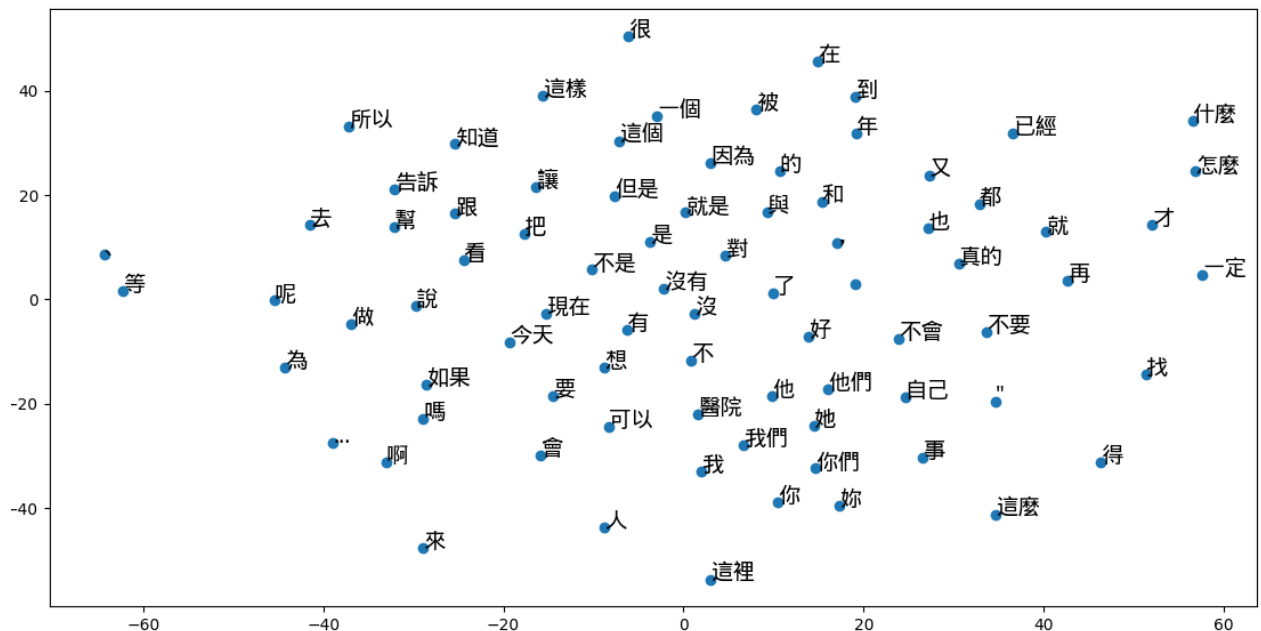
Index	比重
1	4.1%
2	3.0%
3	2.4%
4	2.2%

B. Visualization of Chinese word embedding (Collaborators:無)

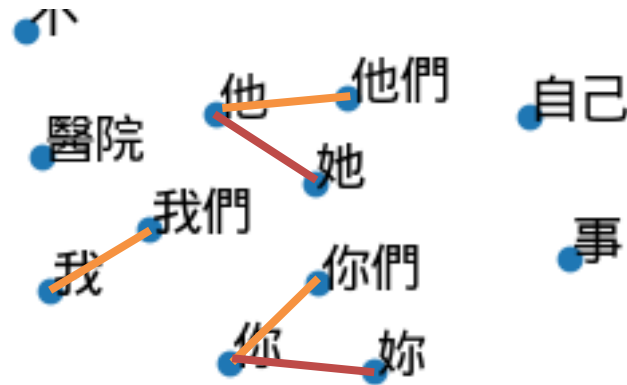
- B.1. (.5%) 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義。

使用 gensim(3.2.0)套件，size=64，min_count=6000。size 為轉換後向量的維度，min_count 設定該文字重複出現次數的門檻。

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



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



從上圖擷取部分可以發現：

1. 我-我們/你-你們/他-他們之間的關係非常類似
2. 你-妳/他-她之間的關係非常類似

C. Image clustering (Collaborators:無)

C.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)

1. 使用 PCA 降維至 400 維，再使用 KMeans 分為兩類

Kaggle 成績：1.00000

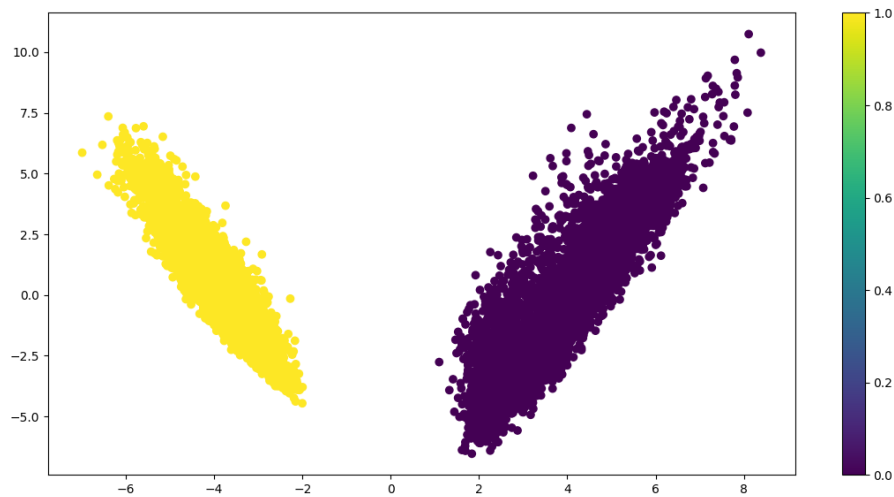
```
pca=PCA(n_components=400,whiten=True).fit_transform(x)
predict=KMeans(n_clusters=2).fit(pca).labels_
```

2. 使用 Auto Encoder 降至 64 維，再使用 KMeans 分為兩類

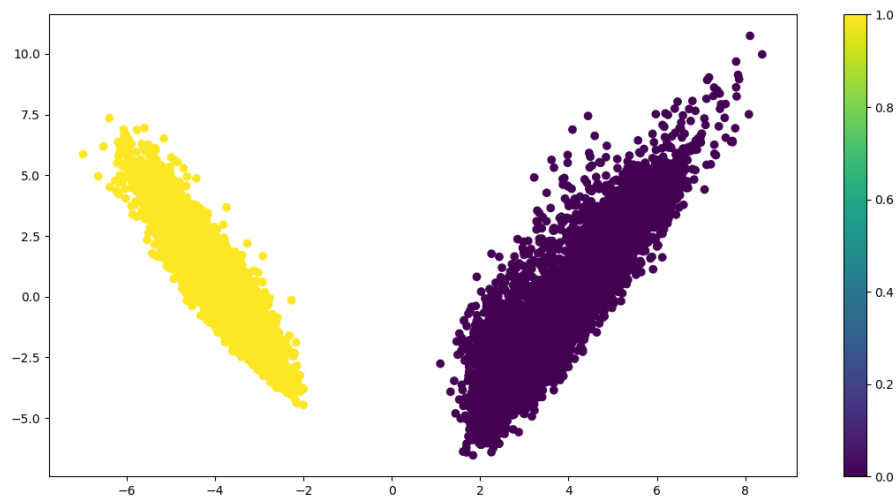
Kaggle 成績：1.00000

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 784)	0
dense_1 (Dense)	(None, 256)	200960
dense_2 (Dense)	(None, 128)	32896
dense_3 (Dense)	(None, 64)	8256
dense_4 (Dense)	(None, 128)	8320
dense_5 (Dense)	(None, 256)	33024
dense_6 (Dense)	(None, 784)	201488
Total params: 484,944		
Trainable params: 484,944		
Non-trainable params: 0		

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



C.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。



與預測的結果相同，原先 Model 的準確度為 100%。