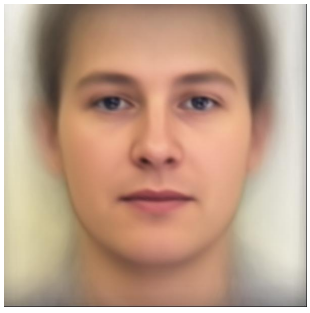
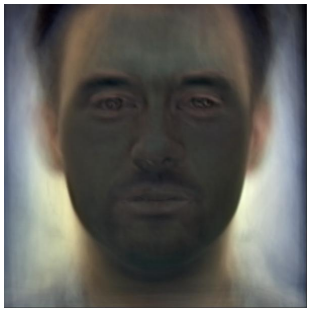
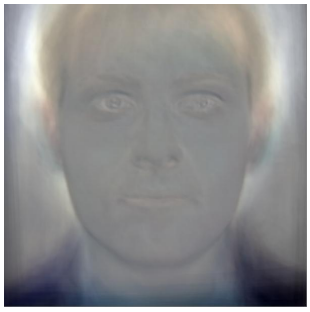
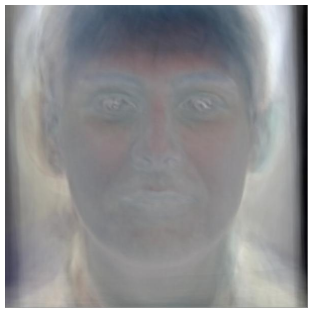
學號：**B03202006** 系級： 物理四 姓名：陳昱蓁

A. PCA of colored faces

A.1. (.5%) 請畫出所有臉的平均。

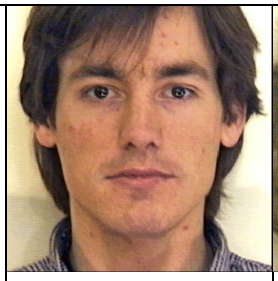
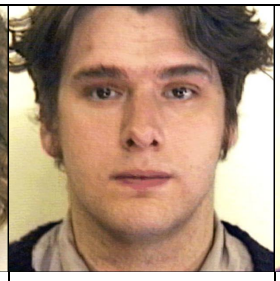
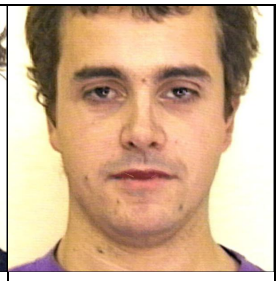
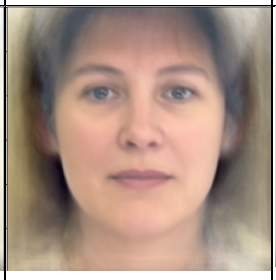
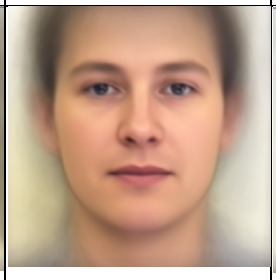
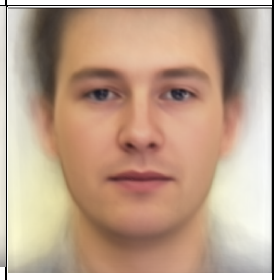


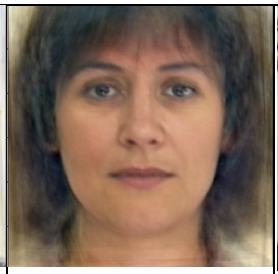
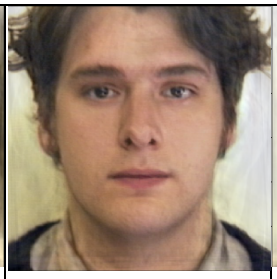
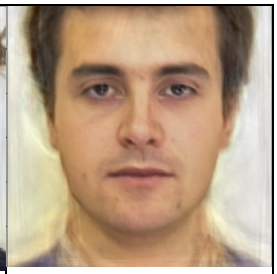
A.2. (.5%) 請畫出前四個 Eigenfaces，也就是對應到前四大 Eigenvalues 的 Eigenvectors。


E0 E1 E2 E3

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。

原圖				
用前 4 個 eigenfaces 重建				

用前 80 個 eigenfaces 重建				

僅用前 4 個 eigenfaces 的結果，四張照片並無顯著差別，都長得很像平均臉，但是用了前 80 個 eigenfaces 時，四個人的差異就出來了，而且幾乎已經很像原圖。

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。

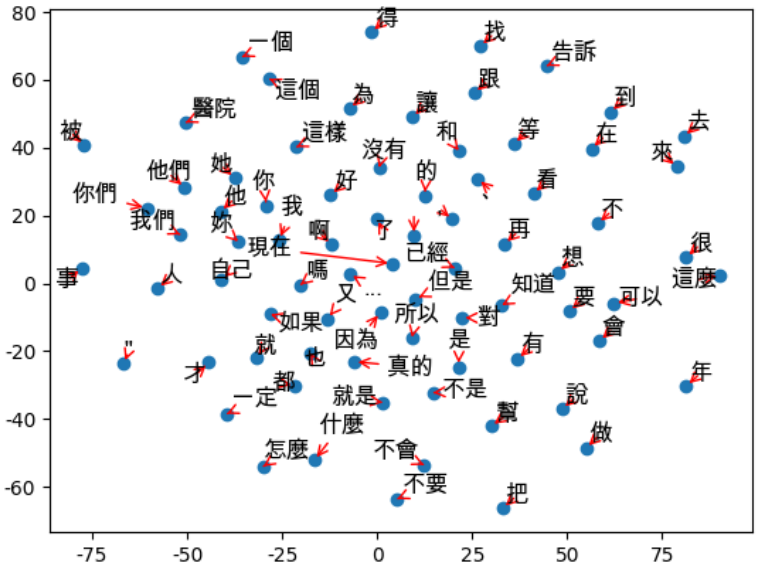
E0 (最大的 eigenvalues) : 4.1%
E1 : 2.9%
E2 : 2.4%
E3 : 2.2%

B. Visualization of Chinese word embedding

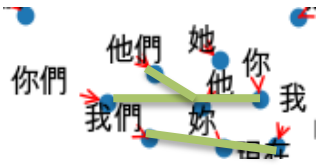
B.1. (.5%) 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義。

Word2vec 套件		gensim
參數調整	size = 80	word embedding 的 shape
	iter = 10	number of epochs
	hs = 1	使用 hierarchical softmax 做 training. (Default 是 negative sampling)

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

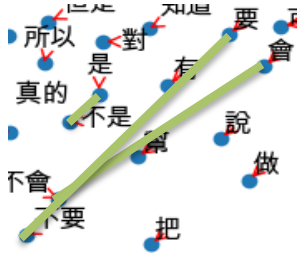


B.3. (.5%) 請討論你從 visualization 的結果觀察到什麼。



1. 代表人稱的詞彙均集中在一區。

人稱的“單數形”與“複數形”的相對位置具有高度關係。（複數偏左，單數偏右）



2. “否定”的詞彙偏下，“肯定”的詞彙偏上。

C. Image clustering

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)

Encoding 方法	細節描述	Training loss	Validation loss	Public score (F1)	Private score (F1)
CNN	兩層 convolution, 兩層 max pooling + 兩層 dense layer (encoding shape = 8)	0.0367	0.0365	0.02715	0.02701
DNN	三層 dense layer (encoding shape = 32)	0.0182	0.0182	0.94336	0.94085

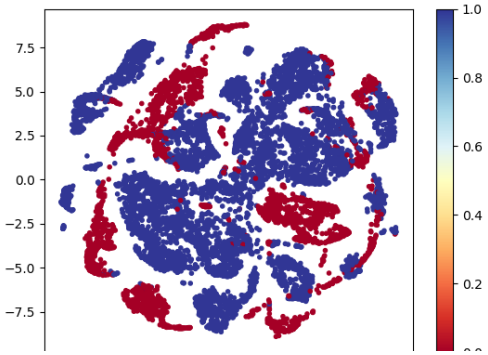
可以看出此問題中 CNN 的表現遠遠不及 DNN，主要是在於此問題 input 圖像較為簡單，CNN 的長處相較之下沒有特別優勢。而 CNN model structure 較為複雜，因此 loss 收斂的也比較慢，在 train 同樣的 epoch 之下（num_epoch = 11），DNN 的表現較好。

另外，也有可能是我做兩層 max pooling 使得圖像太小，最後 encoding shape 太小使得資訊不足，所以導致 performance 不佳。

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

Note:

使用 DNN 訓練 autoencoder (output shape = 32) ,再使用 K-means 的 cluster 方法，將資料分為兩群。



觀察預測的結果（將 encoding 做 t-SNE），此 autoencoder 預測的類群有大致的分界線（圓的外圍、內部），並非隨機分類。

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。

真正的解答，分類的法則果然是圓的外圍及內部。
比較之下可見我的 autoencoder 仍然不是分得很好，因為如圖 1 在圓的外圍（理應紅色）也有藍色的區塊（ex:左上角）。

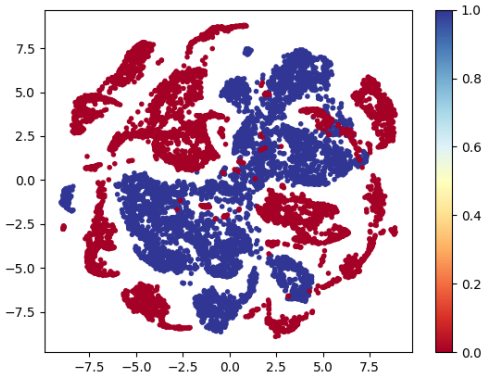


圖 2 true label