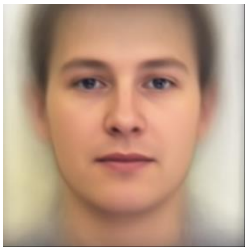
學號：R06725005 系級： 資管碩一 姓名：郝思喬

A. PCA of colored faces

A.1. (.5%) 請畫出所有臉的平均。

將全部圖片的 RGB 分別相加除以平均，得到下圖。



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

下圖從左到右依序為最大的四個 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。

挑選的圖片為 1, 101, 201, 301 四張，結果為下圖。

A.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。

前四大 Eigenfaces 佔的比例為 4.1%、2.9%、2.4%、2.2%。

B. Visualization of Chinese word embedding

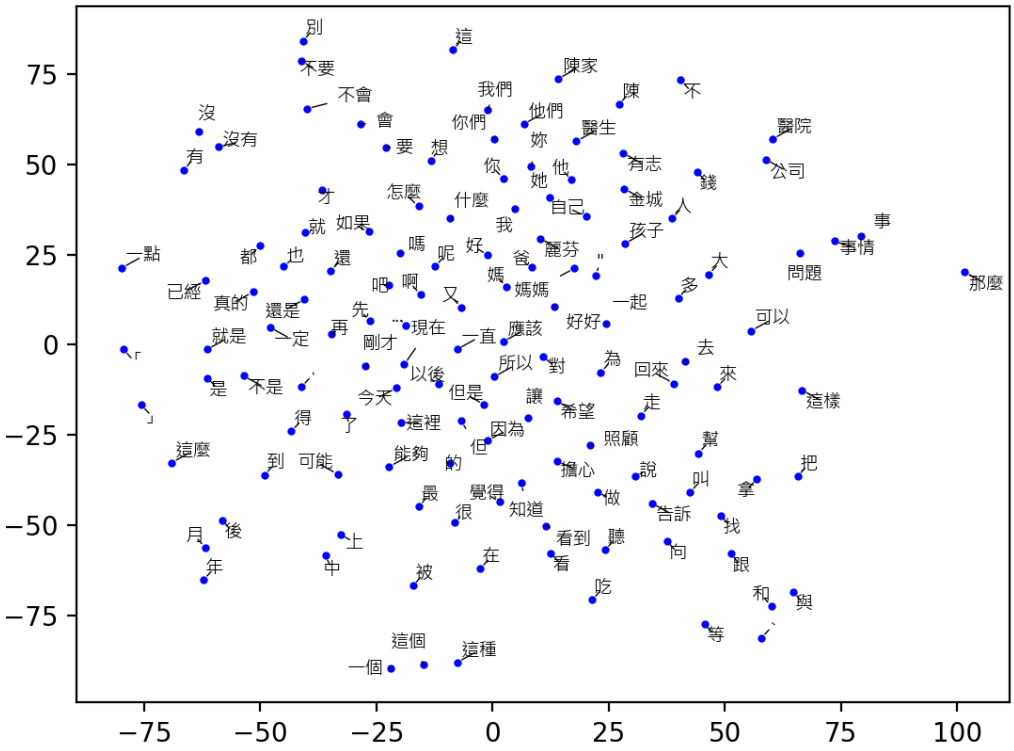
B.1. (.5%) 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義。

使用 jieba 做 tokenize 之後再用 gensim 的 Word2Vec。

size=200: 表示做出來的每個 token 的維度有 200 維。

window=5: 為預設值，表示與目前這個 word 最大的預測距離。
min_count=5: 為預設值，表示 frequency 低於五次的 token 不計算。

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。
在右下角的區塊較多動詞，左下方的部分為量詞，正上方的部分為人物的名詞或是代名詞，左方的部分為程度副詞。

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)
我使用的兩種方法為：

1. PCA 降維+K-means 分群
   先利用 sklearn 的 PCA 將原本 784 維的圖再降成 784 維的 feature，設定 svd_solver 為 randomized、whiten 為 True，再將結果丟入 sklearn 的 k-means 分為兩群。

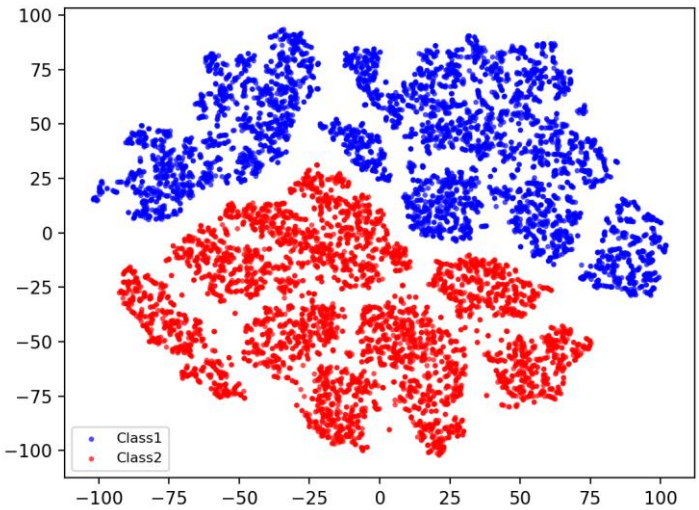
2. Deep CNN Autoencoder+K-means 分群
   將 input 接 4 層 Conv2D，除第三層之外使用 MaxPooling2D，再接上 3 層 Dense 後得到 64 維的 feature。接著再使用 sklearn 的 k-means 分為兩群。

兩種方法的兩群個數以及準確率為下：

方法	兩群數量	F1 Score
PCA	69994 / 70006	0.99165
Deep CNN Autoencoder	69995 / 70005	0.99451

從結果來看使用 PCA 以及 autoencoder 的方法皆能將 feature 精準的抽取出來。但是我在實驗 PCA 的各種維度時，與 autoencoder 一樣 64 維的 PCA 的 feature 並無法完整表達原本 data，預測出來的分群分別為約 30000/90000。因此可以得知在將原本 data 降為程某一特定的小維度時，autoencoder 能夠更有效的表達原本 data 的特性。

C.2. (.5%) 預測 visualization.npy 中的 label，在二維平面上視覺化 label 的分佈。
根據預測的結果，兩個 class 皆為 5000 筆資料，畫出來的圖為以下：



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。
根據 true label，得到的圖為以下，與自己預測的圖相同：