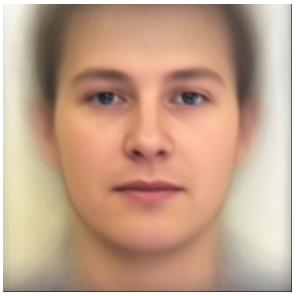
學號：R06922006 系級： 資工碩一 姓名：劉宏國

==========PCA of colored faces==========

1. 請畫出所有臉的平均。(collaborator:自己)



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

上圖由左到右分別對應第一大、第二大、第三及第四大的 Eigenvectors。

3. 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。(collaborator:自己)

上圖為前四張 reconstruction 的結果和原圖得比較。

4. 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。(collaborator:自己)

[0.04144625 0.02948732 0.02387711 0.02207842] => 4.1% 2.9% 2.4% 2.2%

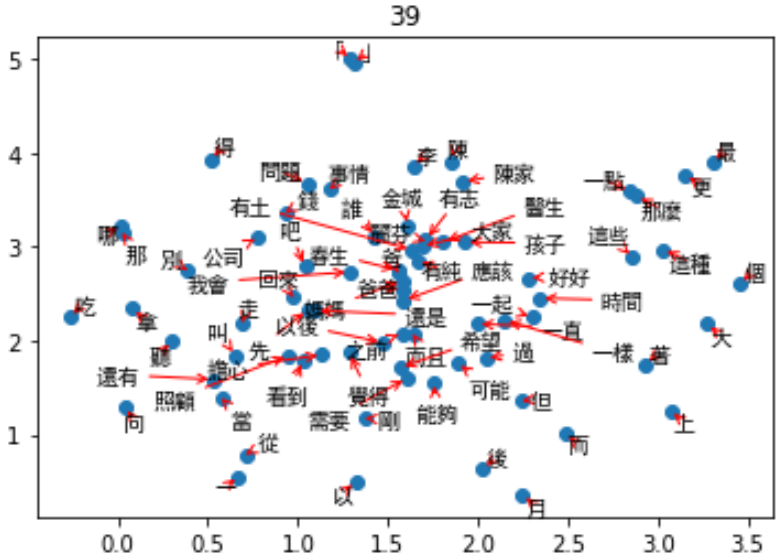
==========Visualization of Chinese word embedding==========

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。(collaborator:自己)

我是使用 gensim.models.Word2Vec，其中參數部分 min_word_count = 0（針對 term 出現次數小於 min_word_count 者不列入 w2v 訓練，這裡我設成 0，把全部 term 都加入訓練）；num_workers = 4 (Number of threads to run in parallel)；context = 3 (Context window size)；embedding_size = 300 (每一個 term 向量的維度)。

6. 請在 Report 上放上你 visualization 的結果。(collaborator:自己)

下圖的 term 出現次數都介於 3000 到 6000 之間。



7. 請討論你從 visualization 的結果觀察到什麼。(collaborator:自己)

由上圖所示，相近的 term 向量，指向的點會相近，例如：「 和 」，他們個別指向的點非常的相近，表示他們個別所代表的 w2v 向量也相近，從一段句子來說，「 和 」***都是成對出現，因此他們的向量會非常相近。***

==========Image clustering==========

8. 請比較至少兩種不同的 feature extraction 及其結果。(collaborator:自己)

方法一：PCA + Kmeans

首先經過 PCA 達到降維的效果，從原本的 784 維降至 400 維，再接去進入 Kmeans 分群標示圖片 label，這邊我只分成兩群，取得 label 後再用 test 的 index 找兩張圖片的 label 是否相同，相同就來自於同一個 dataset。

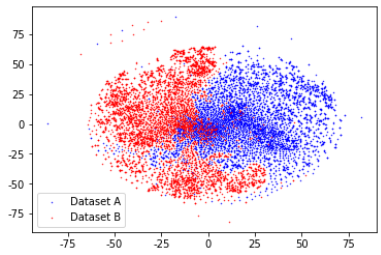
最終 F1-Score 可達：1.0。

方法二：auto encoder + Kmeans

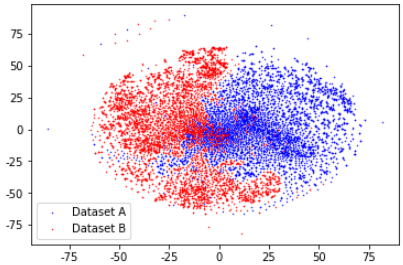
第二種方法是透過 auto encoder 達到降維的效果，這邊我是只有一層的 encoder 及 decoder，從原本的 784 維降至 300 維，訓練 epochs=50、batch_size=256 及有 shuffle。訓練完後再用 encoder 去 predict 降維的結果，之後再套入 Kmeans 和上述相同。

最終 F1-Score 可達：0.71657。

9. 預測 visualization.npy 中的 label，在二維平面上視覺化 label 的分佈。(collaborator:自己)



10. 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。(collaborator:自己)



因為預測出來上傳 Kaggle 的成績為 1.0，因此和 predict 出來的圖片差異不大，但可能因為 TSNE 參數沒調好，導致還是有一區比較雜，不是確切的兩個分類。