學號：R05943110 系級：電子碩二 姓名：蕭堯

A. PCA of colored faces

(.5%) 請畫出所有臉的平均。 詳見 **Fig.1**

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

(.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。詳見 **Fig.3 Fig.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。 1st:**4.1%** 2nd: **2.9%** 3rd: **2.4%** 4th: **2.2%**

Fig.1

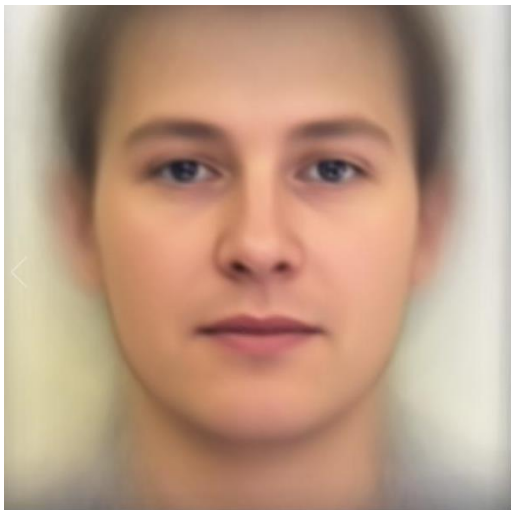


Fig.2

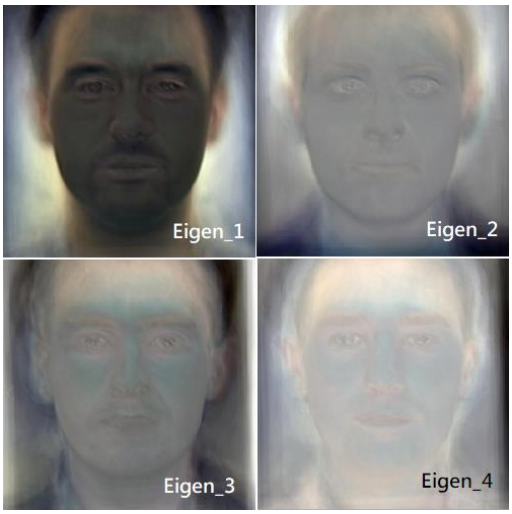
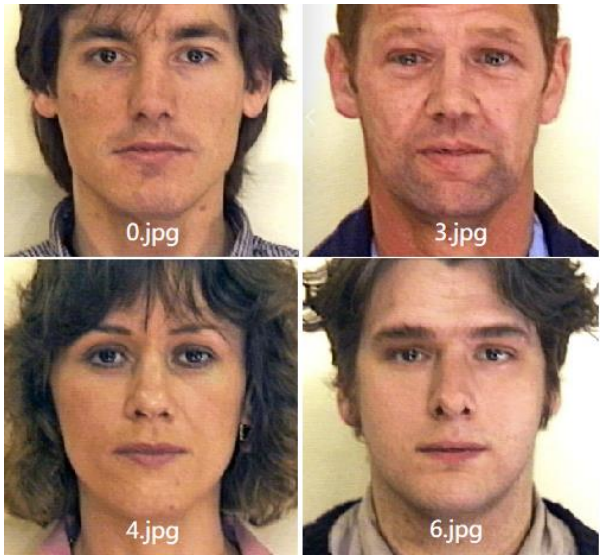
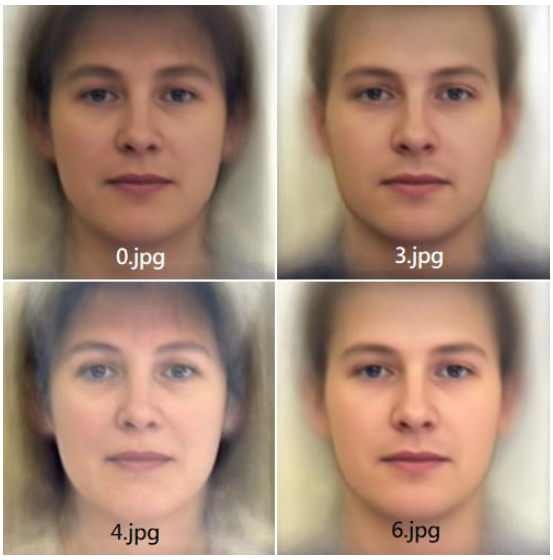


Fig.3



Origin

Fig.4



Reconstruct

B. Visualization of Chinese word embedding

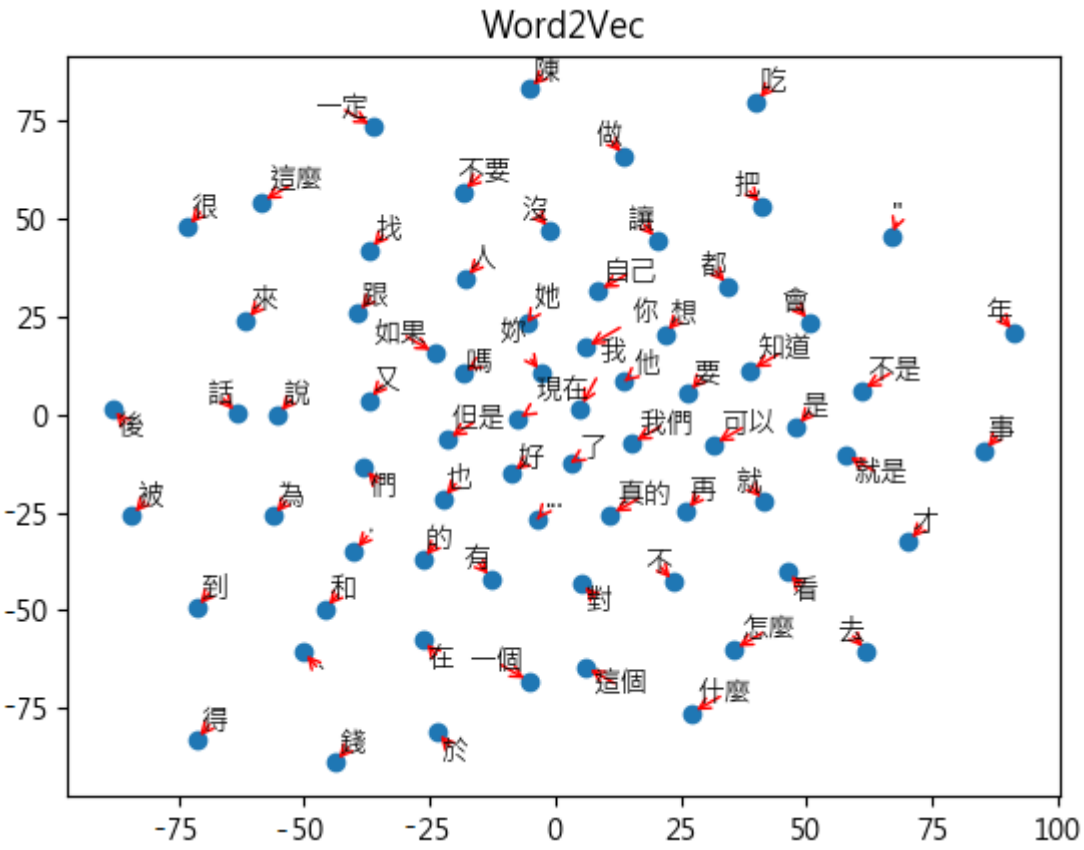
(.5%) 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義。

使用套件: genism

向量維度: 300 (每個詞都用 300 維的向量表示)

Window: 5 (目前的字詞與預測的字詞的最大距離為 5)

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



(.5%) 請討論你從 visualization 的結果觀察到什麼。

類似性質的字會被分到附近，例如：

A. 我、你、他、她、妳、我們　B. 怎麼、什麼　C. 話、說

D. 是、不是、可是、就是、可以

至於向量之間的差值具有的類比關係好像觀察不出來，我想也許是因為使用 TSNE 非線性降維會破壞掉這種線性關係吧？

C. Image clustering

(.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)

Dimension Reduction

我採用 CNN auto-encoder，從原本的(28,28,1)降到(8,4,4)維

Cluster

K-mean

效果很差，kaggle 只有 0.28，用 TSNE 可視化後發現以 center 的距離來判斷 class 效果的確不是很好

Semi-supervised

我把前 100 筆圖像以人工的方式先自動分類，做為 label data，再把降維後的向量每筆都跟這 100 筆算 distance，這 100 筆中最近的 label data 的 class 就成為該筆 data 的 predict class，的 kaggle 有變好到 0.43

Ensemble

這次的 predict 大部分都為 0 只有很少部分是 1，所以我用了上述 Semi-supervised 的方法 train 了三個 model(同一個 code 跑了三次)，並且讓這三個 model 去預測，如果三個 model 都預測為 1，那最後的 predict 才是 1，反之為 0，kaggle 可以到 0.91，比原本的 0.43 好上許多

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

詳見 Fig.5

(.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。**詳見 Fig.6**

因為我的 kaggle best 採用 Ensemble + Semi-supervised 的方法，所以我會用同組 code 跑 3 次(model_1, model_2, model_3) ，每次都做一張圖來比較。而且我也會從 label data 裡找出 50 筆 images，作為 Semi-supervised 的 label data 去找 cluster，以下是我的結果：

Fig.5

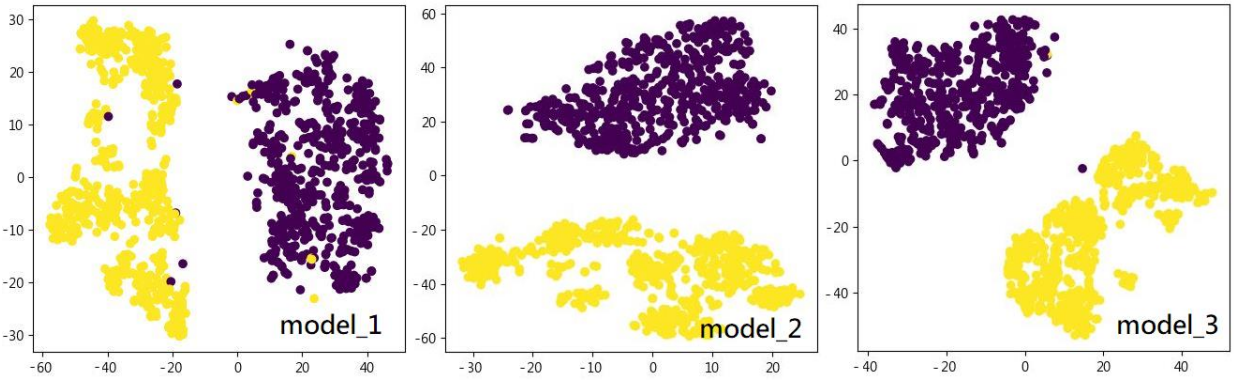
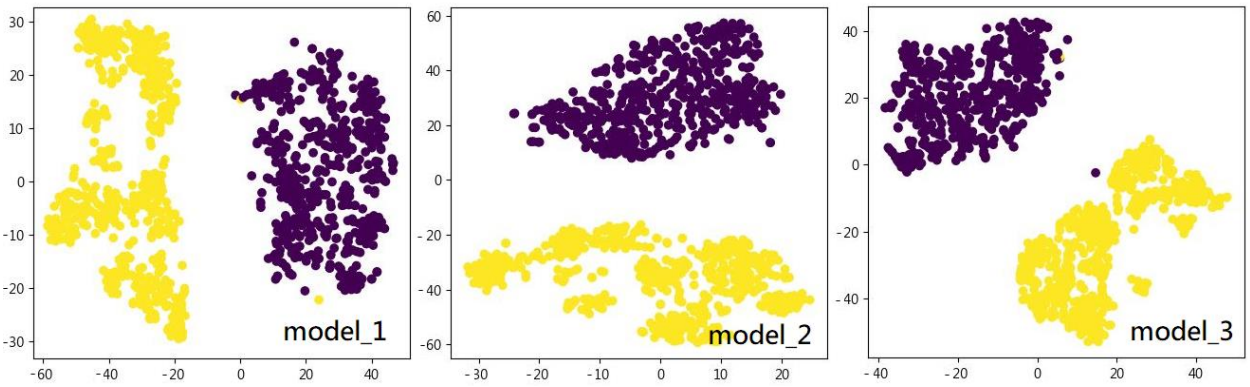


Fig.6



三次跑出的結果都有點差異，model_2 和 model_3 的 predict 和 label 的完全一樣，只有 model_1 和 label 有些不同。這也說明為什麼 ensemble 能夠大大改善 kaggle 的分數，另外，在 Fig.6 裡可以看出來還有些點沒有分得很對，代表我的 auto-encoder 還有改進的空間