

1.1. Dataset 中前 10 個人的前 10 張照片的平均臉和 PCA 得到的前 9 個 eigenfaces:

答：(左圖平均臉，右圖為 3x3 格狀 eigenfaces, 順序為 左到右再上到下)



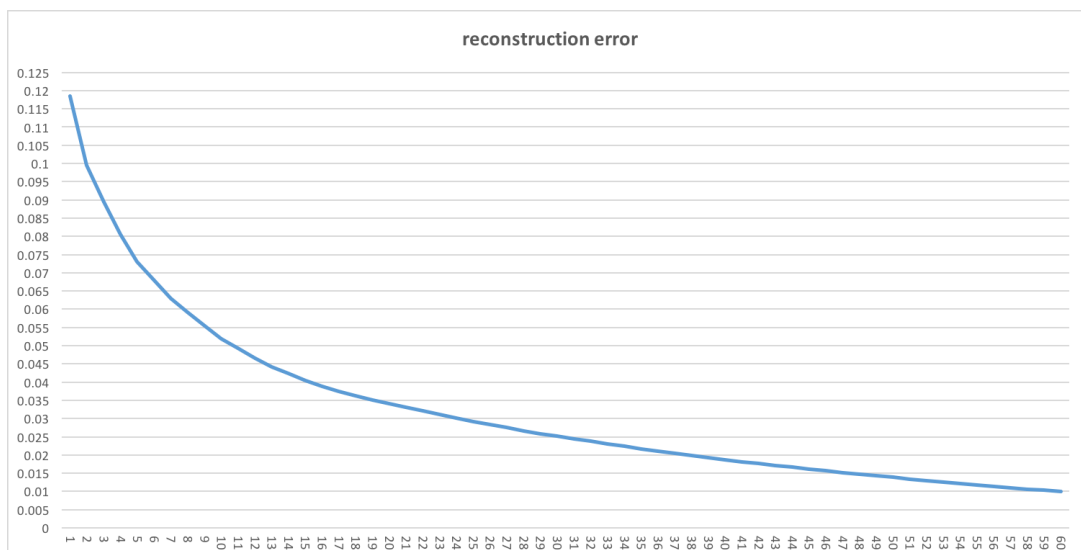
1.2. Dataset 中前 10 個人的前 10 張照片的原始圖片和 reconstruct 圖 (用前 5 個 eigenfaces):

答：(左右各為 10x10 格狀的圖, 順序一樣是左到右再上到下)



1.3. Dataset 中前 10 個人的前 10 張照片投影到 top k eigenfaces 時就可以達到 < 1% 的 reconstruction error.

答：(回答 k 是多少)



當 k 等於 59 時, reconstruction error 為 0.0102, 當 k 為 60 時, reconstruction error 會低於 0.01(1%) 達到 0.0099。

2.1. 使用 word2vec toolkit 的各個參數的值與其意義:

答:

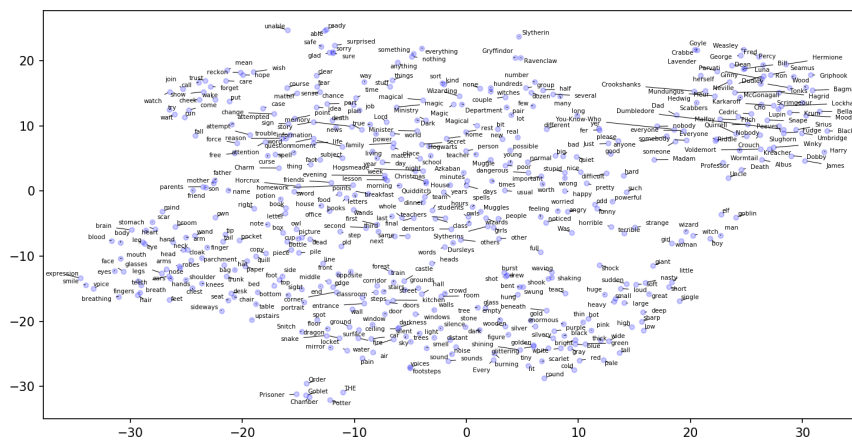
```
word2vec.word2phrase('all.txt', 'word2phrase.txt', verbose = False)
word2vec.word2vec('word2phrase.txt', 'word2vec.bin', alpha = 0.087, hs = 1, size = 100, verbose = False)
```

verbose 為不輸出 training 的過程、alpha 為 learning rate、size 為 vector 的長度、hs 為是否要使用 Hierarchical Softmax, 此外一些較重要的參數例如:cbow 為是否使用 cbow 算法、sample 為單詞頻率超過此 threshold 則會 down-sampled、window 則代表單詞之間可以 skip 的最大值, 下圖為 word2vec 各參數之預設值。

```
def word2vec(train, output, size=100, window=5, sample='1e-3', hs=0,
             negative=5, threads=12, iter_=5, min_count=5, alpha=0.025,
             debug=2, binary=1, cbow=1, save_vocab=None, read_vocab=None,
             verbose=False):
    """
    word2vec execution
```

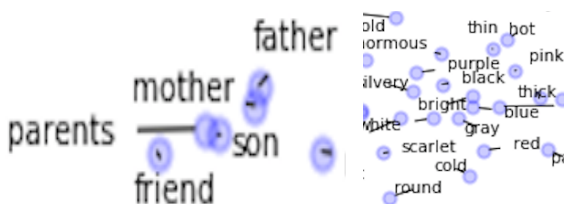
2.2. 將 word2vec 的結果投影到 2 維的圖:

答: (圖)



2.3. 從上題視覺化的圖中觀察到了什麼?

答:



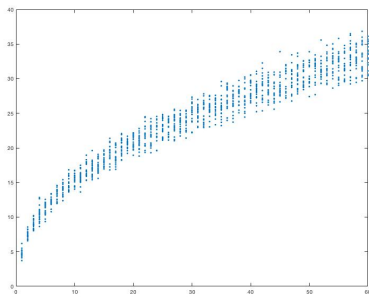
可以觀察到一些關聯的字詞會聚集在一起 (如上圖所示)。

3.1. 請詳加解釋你估計原始維度的原理、合理性, 這方法的通用性如何?

答:

透過助教提供產生資料的程式碼自己模擬從原始 dim1-dim60 經過轉換得到最終 dim100 的資料。可以從上述過程中發現原始 dim1 變成 dim100 後資料的標準差會較小、dim2 變成 dim100 後資料的標準差會比 dim1 變成 dim100 大、dim3 變成 dim100 後資料的標準差會比 dim2 變成 dim100 大、以此類推... 為了驗證結果我將每個 dim1-60 驗證 n 次 (也就是說 dim1-> dim100 跑 n 次, dim2 -> dim 100 跑 n 次...), 最後畫出標準差的變化圖即可驗證, 最後再將測試資料讀入計算標準差, 找標準差差異最小的維度即可。

PS: 上述的標準差是指將 100 維的資料每維各別算標準差, 最後平均這 100 筆標準差



由於驗證過這方法個人認為很合理，且在 kaggle 上結果也算不錯，但該法的通用性不會很好，因為是透過模擬，但在現實世界中是不太可能有這種模擬的機會，故我認為其通用性不會太好。

3.2. 將你的方法做在 hand rotation sequence dataset 上得到什麼結果？合理嗎？請討論之。

答：

因為我的方法需要觀察原始維度轉換成新維度的過程，故這題無法實作。