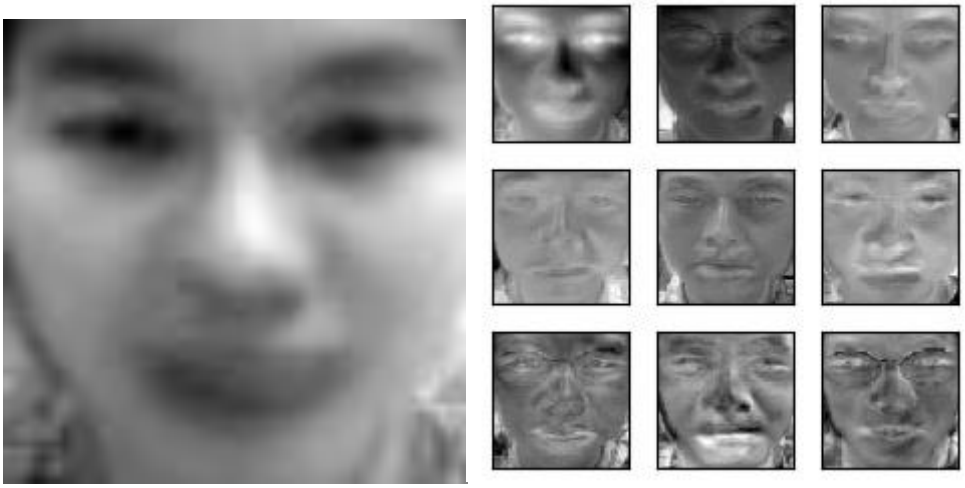
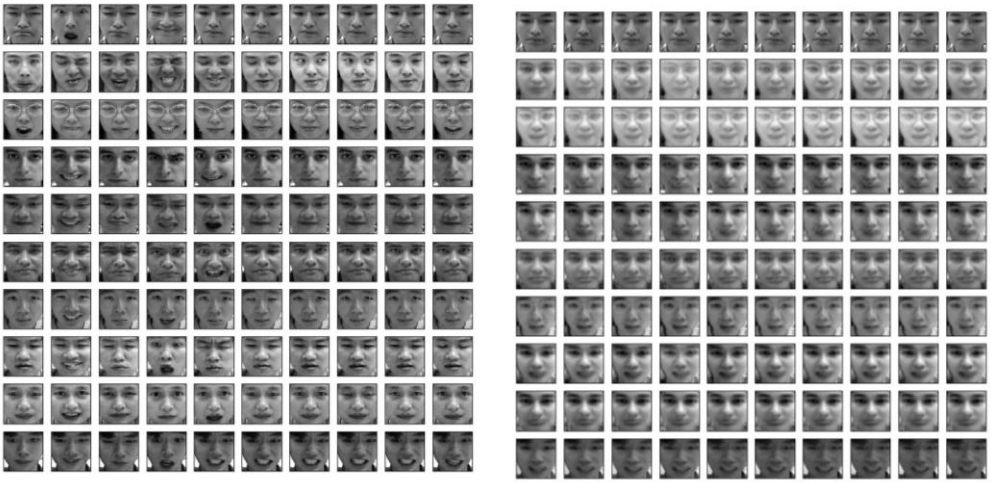
學號：b04902053 系級：資工二 姓名：鄭淵仁

**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)**



上圖中，左圖是平均臉，右圖是 PCA 得到的前 9 個 eigenfaces。

**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)**



上圖中，左圖是原始的圖片，右圖是 reconstruct 之後的圖片。

**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)**

k 是 60。

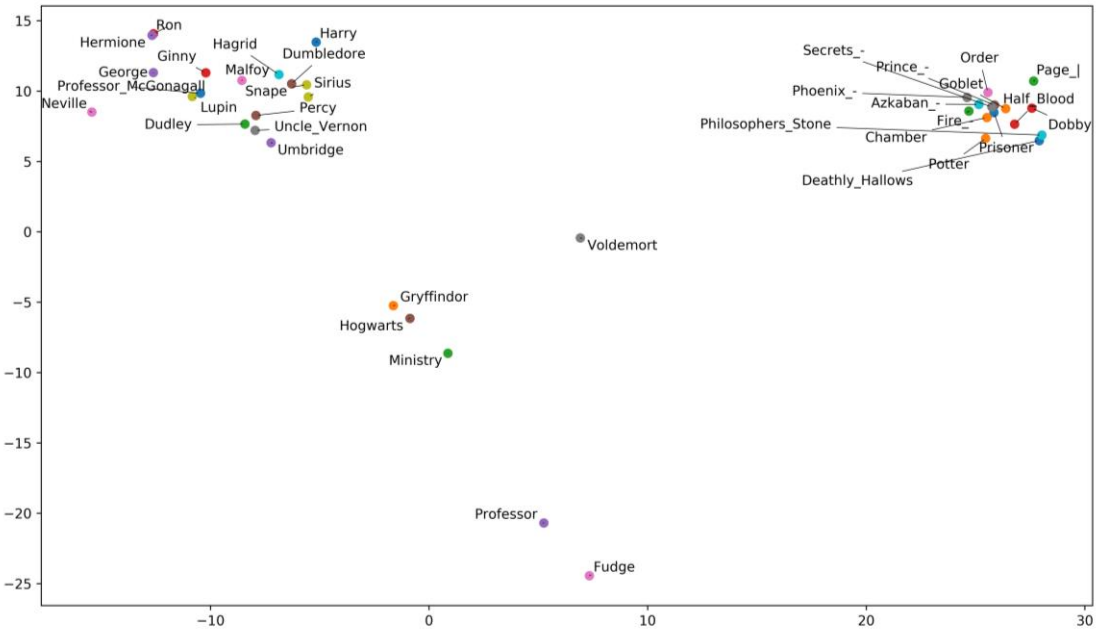
**2.1. 使用 word2vec toolkit 的各個參數的值與其意義：**

答：

| 使用的參數 | 值 | 意義 |
|---|---|---|
| **train** | ./all-phrases | train 的 data 的路徑 |
| **output** | ./all.bin | 要把 train 好的 word vector 的結果存到的路徑 |
| **size** | default(100) | word vector 的大小 |
| **window** | 25 | word 跟 word 之間最大跳躍的長度 |
| **sample** | default(0) | 設一個詞出現次數的 threshold。出現太多次的詞會被隨機的 down-sample |
| **hs** | default(1) | 使用 Hierarchical Softmax |
| **negative** | 5 | negative sample 的數量，待補 |
| **min_count** | 100 | 把出現次數少於 min_count 次的字去掉 |
| **alpha** | default(0.25) | starting learning rate |
| **cbow** | default(1) | 使用 continuous back of words model |

**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：

我發現比較相關的詞會出現在一起，例如 Gryffindor、Hogwarts、Ministry 會是很接近的一群；而其他如 Ron 和 Hermione 幾乎疊在一起了，可見他們在文本裡面關係很密切；Sirius、Snape、Percy、Malfoy 也是很接近的一群。

**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：

- 估計的方法：

  我使用 **DNN** 來估計原始維度。使用的方法如下：

  **1.** 生成訓練資料：
    a) 使用題目提供的生成資料的 python script 生成不同維度的 data point，共 300,000 筆資料。
    b) 使用 PCA 把每一個 data point 都降維成 100 維的矩陣，並且只存下最大的前 60 維的數字作為 training data。
    c) 把各個 data point 的維度取 log 後作為 training data 的 label。

  **2.** 生成測試資料：
    a) 把要測試的 200 筆資料都使用 PCA 降維成 100 維的矩陣，並且只取求 60 維的資料作為 testing data。

  **3. Train and Predict**：
    a) 使用 DNN 訓練出很多組 model。
    b) 把不同 model predict 出來的值取平均，輸出到 csv。

- 原理：

  由於 ELU 的運算對於資料的原始維度影響不大，所以使用 PCA 後得出來的值，只要設好 threshold，應該都能預測出正確的維度。但是正確的 threshold 可能會因為原始維度的不同、資料數量、或資料的值的大小差異而有不同的值。所以就使用 DNN 訓練出一個，可以從經過 PCA 轉換後的資料算出原始資料維度的 function，再使用這個 function 計算出結果。

- 結果&合理性：

  在 validation set 上面的 RMSE 是 0.050711，在 kaggle 上面，public data 的 RMSE 是 0.05280，private data 的 RMSE 是 0.04020。

  預測出來的值沒經過調整就都在 1~60 之間，以值的大小而言，很合理。

- 通用性：

  由於我的訓練資料是基於題目提供的生成資料的 python script 生成出來的，所以只會對使用相同方法生成出來的資料有效，在其他情況下的資料會是沒有效果的。

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

**答：**

計算出來的維度是 exp(-468961)。

結果很明顯是**不合理**的，維度並不會有負值，而且負的值還那麼大。

會產生這樣的結果的原因就如同上一小題所述：我的訓練資料是基於題目提供的生成資料的 python script 生成出來的，所以只會對使用相同方法生成出來的資料有效。但是 hand rotation 這些圖很明顯不是由題目提供的 python script 生成出來的，所以結果會差很多而且不合理。