

110590007 資工系大三 白宸安

1. problems you encountered:

遇上的主要是在實作 MY_PCA()和 MY_SparsePCA()這兩個 class，因為有較多的數學公式，要將它們理解並轉換成 code，花了不少時間。另外在 implement advanced part 時嘗試了不同的 component 數量，發現較大的值會保留更多資訊，能使 f1_score 更好。

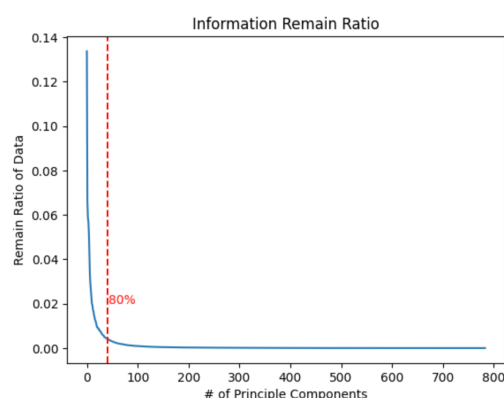
2. how you implement the basic PCA

- 一、 $x_{train_cent} = x_{train} - \text{mean_train}$ 來做 centralize data。
- 二、把 cent data 用 np.dot 來做 covariance matrix。
- 三、用 np.linalg.eigh(cov_mat)來算 eigen values 和 eigen vector
- 四、用 np.argsort(eigen_val)[::-1]來 sort eigen values
- 五、選擇前 n 個為 component：self.eigen_vec[:, :self.n_components]
- 六、Transform the data：data_pca = np.dot(centerized_data, self.principal_components)

3. Plot eigenvectors and the reconstruct image



4. Plot the distribution of eigenvalues



5. Introduce the preprocessing/PCA methods you implemented in advanced part.

在 advanced part 裡面，我調高了 component 的數量，調成 500，讓資訊能被保留的更多，這樣也能從 dataset 裡學到更多資訊，有發現增加 component 數量能使 f1_score 變高，但大約有個閥值，超過之後的 component 增加就沒有什麼效益了。