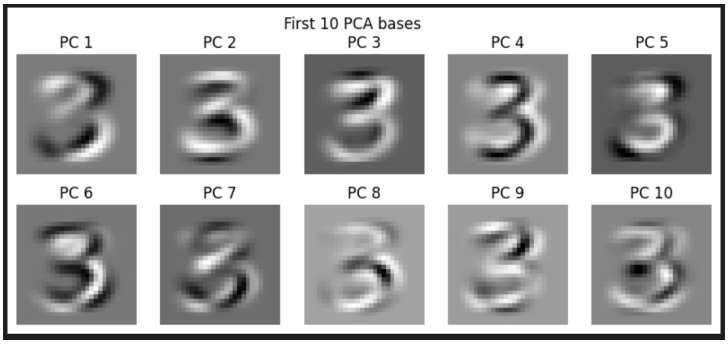
Pattern Recognition Computer Experiment 3

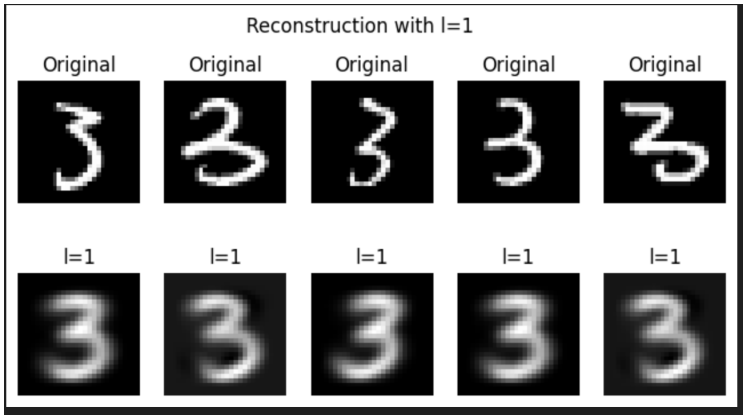
113062589 洪聖祥

1. MNIST PCA dimension reduction

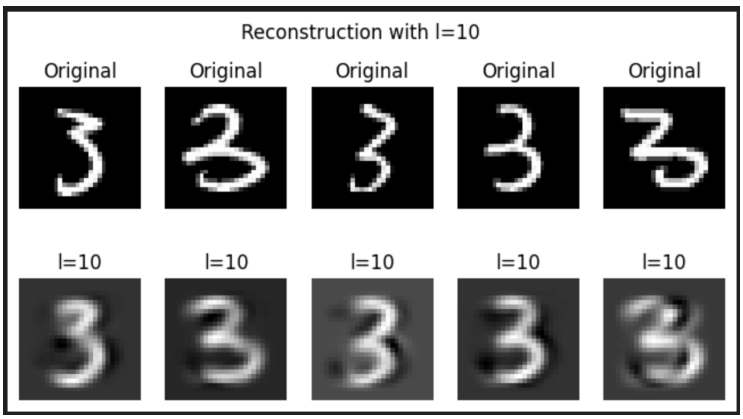
   (b) Show PCA basis and dimension-reduced reconstructions



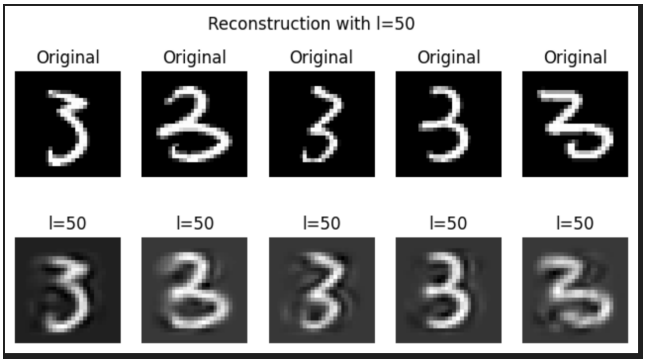
10 Principal components of digit “3”



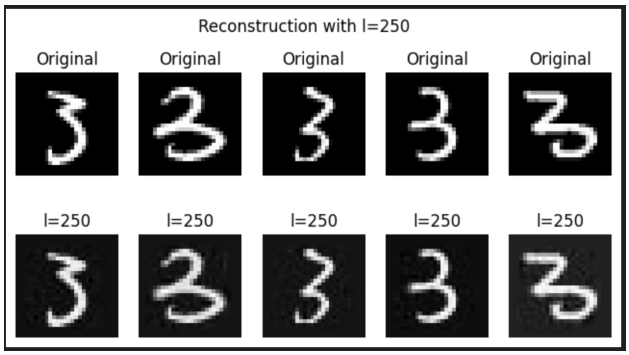
Dimension = 1, reconstruction



Dimension = 10, reconstruction



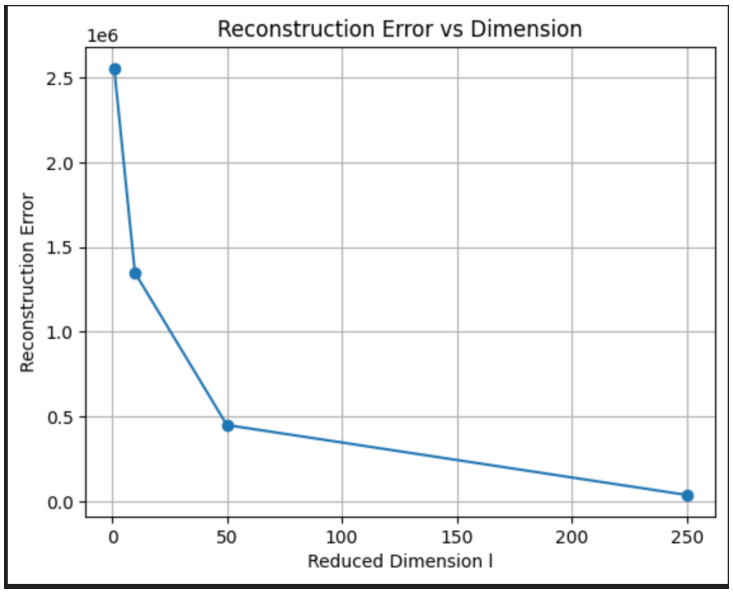
Dimension = 50, reconstruction



Dimension = 250 reconstruction

(c) Evaluate error with dimension l = 1, 10, 50, 250 and conclude

To evaluation error with different dimensions, I draw a reconstruction error versus dimension plot. From the below plot, we can conclude that as the dimension increases, the reconstruction error decreases.



2. EM algorithm for Gaussian mixtures

(a) Experiment settings for EM algorithm

Initialization:

- Initialize means: random choose K (K=3 in this exp) data as its means
- Initialize covariances: set K (K=3 in this exp) covariance as the data matrix's covariance matrix
- Initialize weights: set K (K=3 in this exp) weights as 1 / K
- Stopping criterion: meets the total iterations or the log likelihood changes less a small threshold (1e-4 in my exp)

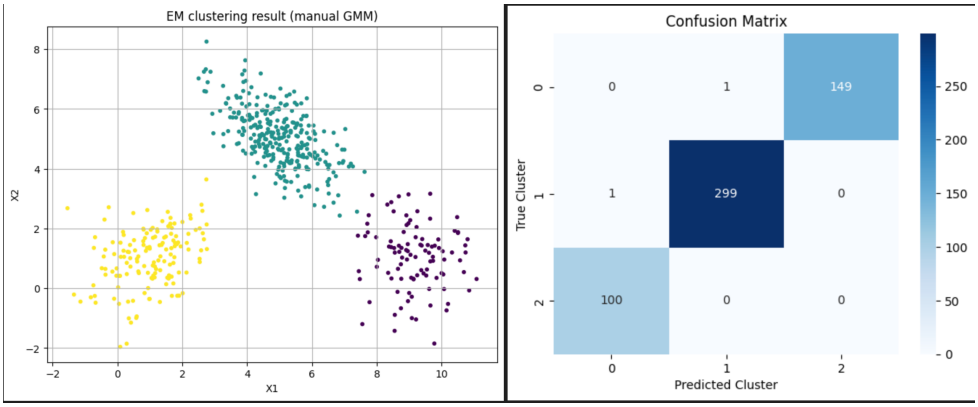
(b) Run EM algorithm

(c) Compare the results and conclude

In my experiment, the final parameters estimated are:

- Means: (0.9952, 1.0275) / (4.989, 5.035) / (9.1665, 1.024)
- Covariance:
- $\begin{bmatrix} 0.87680951 & 0.31068167 \\ 0.31068167 & 0.94162033 \end{bmatrix}$
- $\begin{bmatrix} 0.98183192 & -0.58101926 \\ -0.58101926 & 0.95643721 \end{bmatrix}$
- $\begin{bmatrix} 0.74386815 & -0.07181958 \\ -0.07181958 & 1.12053933 \end{bmatrix}$
- Weights: 0.27098514 / 0.54634109 / 0.18267377

By comparing confusion matrix and 2D visualization, we observe that EM algorithm has learned the distribution.



3. K-means algorithm

From the dataset generate from Problem2, we can also cluster the dataset using K-means. By setting the K value as 2, 3 and 4, we can generate the below scatter plot. From the below plot, if we set a low K (K=2), K-means cluster two large groups but misclassifies one of the actual three clusters. By setting a proper K (K=3), K-means do a good job on separating 3 true clusters. When we set a large K (K=4), K-means split one of the real clusters into two, resulting in over-segmenting. From this experiment, we can conclude that K-means is highly sensitive to the predefined cluster count.