THE CHINESE UNIVERSITY
OF HONG KONG (SHENZHEN)
School of Science and Engineering

Tutorial - 9

Fundamental of Artificial Intelligence **CSC3180**

1. Answer the following questions:

(a) Why do we need to reduce the dimension of the data? (L10: P4-7)

- Many dimensions are related to each other.
- The number of training data should be very large if the dimension is relatively high.
- Some valuable information exists in the latent sub-space.

(b) What is the scatter matrix (L10: P17)? Understand what about eigenvector and eigenvalue as well as their functions? (L10: P18)

- In multivariate statistics and probability theory, the **scatter matrix** is a statistic that is used to make **estimates of the covariance matrix**, for instance of the multivariate normal distribution.

Given *n* samples of *m*-dimensional data, represented as the m-by-n matrix, $X = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$, the sample mean is

$$\overline{\mathbf{x}} = \frac{1}{n}\sum_{j=1}^{n}\mathbf{x}_j$$

where $\mathbf{x}_j$ is the *j*-th column of X.

The **scatter matrix** is the *m*-by-*m* positive semi-definite matrix

$$S = \sum_{j=1}^{n}(\mathbf{x}_j - \overline{\mathbf{x}})(\mathbf{x}_j - \overline{\mathbf{x}})^T = \sum_{j=1}^{n}(\mathbf{x}_j - \overline{\mathbf{x}}) \otimes (\mathbf{x}_j - \overline{\mathbf{x}}) = \left(\sum_{j=1}^{n}\mathbf{x}_j\mathbf{x}_j^T\right) - n\overline{\mathbf{x}}\overline{\mathbf{x}}^T$$

From https://en.wikipedia.org/wiki/Scatter_matrix

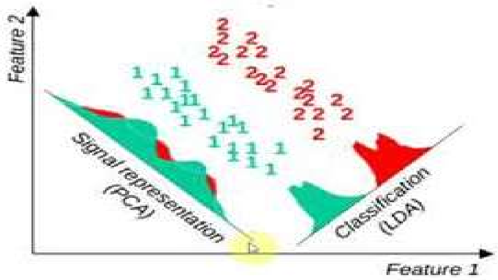
- The eigenvalue and eigenvectors is specified to the **scatter matrix of** the training data. The meaning of eigenvalue and eigenvectors to **PCA** is that The eigenvectors that have the largest eigenvalues are corresponding to the most significant variation of the data, and can be the most significant bases to efficiently represent the original data.

(c) Understand the PCA application to facial recognition: Eigenface. (L10: P22-26)

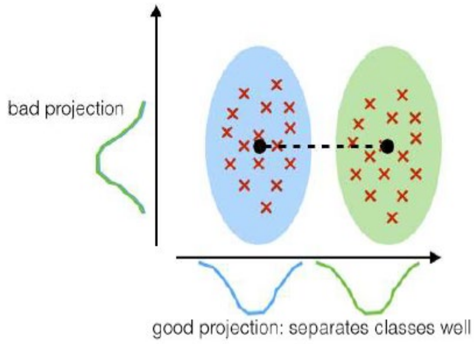
Eigenfaces: The (principle) eigenvectors of the covariance matrix are viewed as images.

(d) Compare two techniques PCA and LDA, and point out their main difference. (L10: P34-37)

PCA: component axes that maximize the variance. PCA is mainly used for feature extraction, as well dimensional reduction.



LDA: maximizing the component axes for class-separation. LDA is mainly used for classification

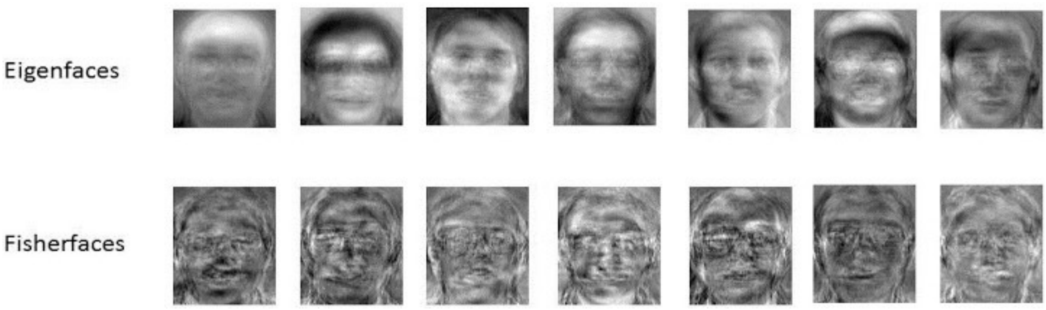


Eigenface:

Eigenfaces attempt to maximise the scatter of the training images in face space.

Fisherfaces:

Fisherfaces attempt to maximise the between class scatter, while minimising the within class scatter. Recognition Using Class Specific Linear Projection



(e) Point out the difference between PCA and 2D-PCA. (L10: P52)

2DPCA has the following advantages comparing with PCA.

- 2DPCA is based on the image matrix, it is simpler and more straightforward.
- 2DPCA is better than PCA in recognition accuracy.
- 2DPCA is computationally more efficient than PCA.
- 2DPCA is more suitable for small sample size problems.
- 2DPCA can evaluate the covariance matrix more accurately.

In summary, the 2DPCA method is superior to PCA in terms **of recognition accuracy** and **computational efficiency**.

2. **From L10: P15-21, PCA method given by using image data is defined, which projects an image space with 644 dimension into 6 dimension eigenvector space. Please understand each step.**

Solution:

Step 1: Define training set X and test set Y. There are k classes in X.

Step 2: The total mean of X is m, the mean values of each class $C_1, C_2 ... C_k$ is $m_1, m_2 ... m_k$, E is the mathematical expectation.

$$m_1 = E(X^{(C_1)})$$

$$m_2 = E(X^{(C_2)})$$

…………..

$$m_k = E(X^{(C_k)})$$

$$m = E(m_1, m_2, ..., m_k)$$

Step 3: Compute the total scatter matrix S_t:

$$S_t = E[(X - m)(X - m)^T]$$

Step 4: Compute the eigenvalue and eigenvectors of S_t by solving the function:

$$\lambda_i \varphi_i = S_t \varphi_i, \quad i = 1, 2, 3, ...D$$

where λ_i and φ_i are the i^{th} eigenvalue and eigenvector. $D = width \times height$ is the number of pixels of an sample image.

Step 5: Suppose there are n nonzero eigenvalues among λ_i, say $\lambda_1, \lambda_2, ..., \lambda_n$ in descending order.

Compute and find the minimum n^* ($n^* \le n \le D$) which satisfies:

$$r_\lambda = \frac{\sum_{i=1}^{n^*} \lambda_i}{\sum_{j=1}^{D} \lambda_j} > threshold$$

where r_λ is the ratio of the eigenvalue sum of selected components to the total sum. A high *threshold* will achieve high accuracy of classification; while a low *threshold* can obtain fewer dimensions after PCA.

Step 6: The PCA projection transform W is composed by:

$$W = (\varphi_1, \varphi_2, ..., \varphi_{n^*})$$

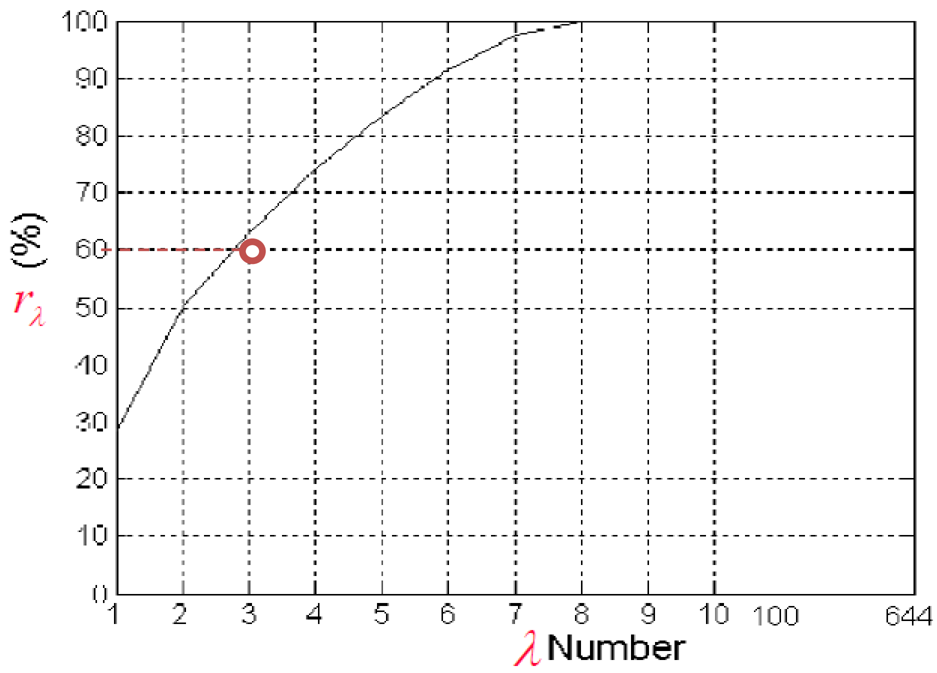
The transformed feature sets of X and Y is:

$$X^{'} = (X - m) * W$$

$$Y^{'} = (Y - m) * W$$

3. **According to the figure in P19, how to find its minimum λ if we hope to get r_λ > 60? ($\lambda = 3$).**

Solution:



4. **From P44-49, 2DPCA method given by using image data is defined. Please understand each step. Also, learn the PCA flowchart in P42, please draw a 2DPCA flowchart.**

Solution:

Step1: Denote the training set and testing set as $X = [X_1, X_i, \cdots, X_n], X_i \in R^{d\times g}$ **and** $Y = [Y_1, Y_i, \cdots, Y_m], Y_i \in R^{d\times g}$ **.**

Step2: Compute their mathematical expectation E

$$M = E(X_i) = \frac{1}{n}\sum X_i$$

Step3: Compute the total scatter matrix S_t

$$S_t = \frac{1}{n}\sum (X_i - M)^T (X_i - M)$$

Step4: Compute the eigenvalue and eigenvectors of S_t

$$\lambda_i \varphi_i = S_t \varphi_i,\ i = 1, \cdots, g$$

where λ_i **and** φ_i **are the i-th eigenvalue and eigenvector.**

Step5: Suppose there are g nonzero eigenvalues among λ_i , say $\lambda_1, \lambda_2, \ldots, \lambda_g$ in descending order.

Compute and find the minimum g^* which satisfies:

$$r_\lambda = \frac{\sum_{i=1}^{g^*} \lambda_i}{\sum_{j=1}^{g} \lambda_j} > threshold$$

Step6: The 2DPCA projection transform W is composed by

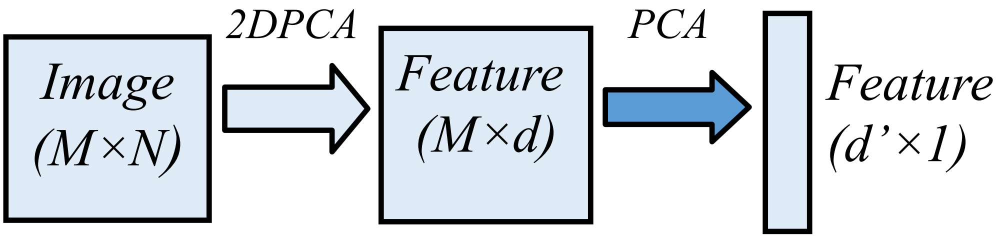
$$W = \left(\varphi_1, \varphi_2, \cdots, \varphi_{g^*}\right)$$

Then, for X and Y, we obtain their transformed feature sets X' and Y'

$X' = (X - M)W$, $Y' = (Y - M)W$

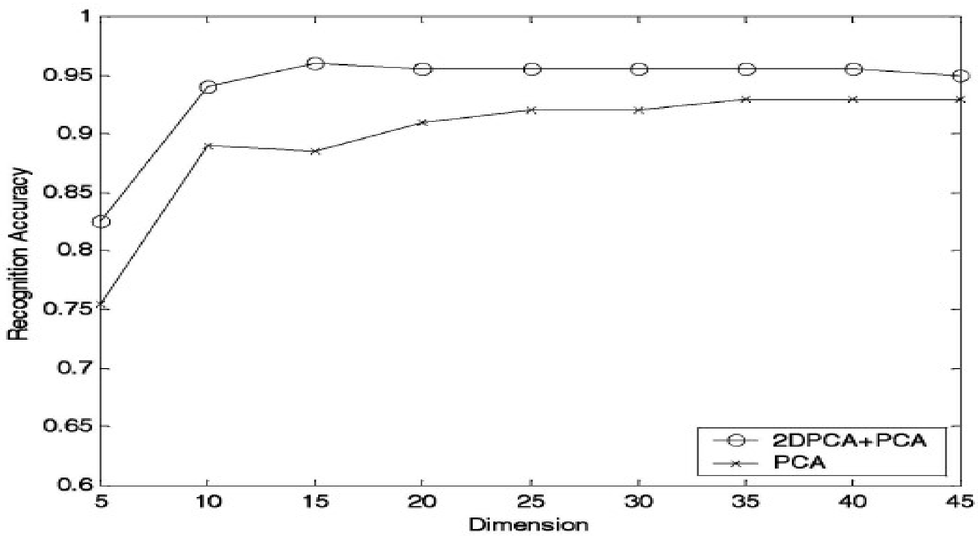
Then, we can apply any classifier to do the classification, e.g. nearest neighbor classifier.

5. **Following flowchart is about 2DPCA+PCA, could you explain why we should put them together and understand its input/output for each stage?**



Solution:

One disadvantage of 2DPCA (compared to PCA) is that the feature dimension of 2DPCA is much higher than PCA. Solution: 2DPCA + PCA.



The performance of 2DPCA+PCA is still better than that of PCA.