International Journal of Pattern Recognition
and Artificial Intelligence
Vol. 26, No. 3 (2012) 1256005 (23 pages)

DOI: 10.1142/S0218001412560058

World Scientific
www.worldscientific.com

IMPACT OF FULL RANK PRINCIPAL COMPONENT ANALYSIS ON CLASSIFICATION ALGORITHMS FOR FACE RECOGNITION

FENGXI SONG[*,†,§], JANE YOU[*], DAVID ZHANG[*] and YONG XU[‡]

[*]*Department of Computing*
Hong Kong Polytechnic University, Hong Kong

[†]*Department of Automation*
New Star Research Inst. of Applied Tech. in Hefei City Hefei, P. R. China

[‡]*Shenzhen Graduate School*
Harbin Institute of Technology, Shenzhen, P. R. China
[§]*songfengxi@yahoo.com*

Received 24 March 2011
Accepted 9 February 2012
Published 15 August 2012

Full rank principal component analysis (FR-PCA) is a special form of principal component analysis (PCA) which retains all nonzero components of PCA. Generally speaking, it is hard to estimate how the accuracy of a classifier will change after data are compressed by PCA. However, this paper reveals an interesting fact that the transformation by FR-PCA does not change the accuracy of many well-known classification algorithms. It predicates that people can safely use FR-PCA as a preprocessing tool to compress high-dimensional data without deteriorating the accuracies of these classifiers. The main contribution of the paper is that it theoretically proves that the transformation by FR-PCA does not change accuracies of the k nearest neighbor, the minimum distance, support vector machine, large margin linear projection, and maximum scatter difference classifiers. In addition, through extensive experimental studies conducted on several benchmark face image databases, this paper demonstrates that FR-PCA can greatly promote the efficiencies of above-mentioned five classification algorithms in appearance-based face recognition.

Keywords: Pattern classification; principal component analysis; dimension reduction; face recognition.

1. Introduction

Full rank principal component analysis (FR-PCA)[12,26,33] is a special form of principal component analysis (PCA) which retains all nonzero components of PCA. Unlike PCA which has widely been studied in the field of appearance-based face recognition[3,5,8,9,13,14,16–18,22,23,29–32] FR-PCA only arouses attentions of a few researchers in recent years. To avoid the computational difficulty of a well-known facial feature extraction algorithm, N-LDA[4] when applied to high-dimensional face

image data, Huang *et al.*[12] suggested that before applying N-LDA, people should use FR-PCA to reduce the dimensionality of the data at first. What is important is that Zhao *et al.*[33] had theoretically proven that the transformation by N-LDA is equivalent to the transformation by FR-PCA plus N-LDA.

By using the generalized scatter difference instead of the generalized Rayleigh quotient as a class separability measure, the feature extraction algorithm — multiple maximum scatter difference (MMSD)[26] readily avoids the singularity problem which conventional linear discriminant analysis methods such as Fisher linear discriminant analysis and Foley−Sammon discriminant analysis usually encounter in face recognition. It will be very time-expensive or even impossible to perform MMSD directly on high-dimensional face image data. To deal with the computational difficulty, Song *et al.*[26] embedded a FR-PCA stage into the facial feature extraction algorithm and strictly proved that the transformation by FR-PCA plus MMSD is equivalent to the transformation by MMSD.

These two papers well demonstrate that FR-PCA, as a data preprocessing tool, cannot only greatly save computational times consumed by these two feature extraction algorithms but also retain their effectiveness in face recognition. However, as a data preprocessing tool, how FR-PCA will affect the recognition accuracy of a classification algorithm has so far not been seriously studied.

Generally speaking, it is hard to estimate how the accuracy of a classifier will change after data are compressed by PCA. We find an interesting fact that the transformation by FR-PCA does not change the recognition accuracy of many well-known classification algorithms. It predicates that we can use FR-PCA to compress data without deteriorating the accuracies of these classifiers.

Lee and Landgrebe[15] ever introduced two interesting concepts: discriminantly redundant feature and discriminantly informative feature. In other words, a feature is discriminantly redundant if an error of the Bayesian classifier remains unchanged when it is taken out of the feature set. Otherwise, it is discriminantly informative. Since we cannot train a truly Bayesian classifier in practice, whether a feature is discriminantly redundant or discriminantly informative is classifier-dependent. Thus, we extend the concept as follows: A feature is discriminantly redundant for a given classifier if the recognition accuracy of the classifier remains unchanged when it is taken out of the feature set. Although it is hard to judge whether a given feature is discriminantly redundant or not in general, we can usually judge whether a given feature is discriminantly redundant or not for a particular classifier. In fact, our theoretical studies presented in this paper show that each component of PCA with zero variance is a discriminantly redundant feature for the k nearest neighbor (k-NN),[33] the minimum distance (MD, also named as the centroid),[6] support vector machine (SVM),[27] large margin linear projection (LMLP),[24] and maximum scatter difference (MSD)[25] classifiers.

The contribution of this paper is twofold: first, it reveals a series of theoretical properties of FR-PCA which are especially helpful for high-dimensional data classification by rigorous mathematical proofs; second, through extensive experimental studies conducted on several benchmark face image databases, it demonstrates that

FR-PCA can greatly promote the efficiencies of k-NN, MD, SVM, LMLP, and MSD classification algorithms in appearance-based face recognition.

The rest of the paper is organized as follows: in Sec. 2, we will introduce the concept of FR-PCA. In Sec. 3, we will study the impact of FR-PCA on accuracies of two multi-category classification algorithms. In Sec. 4, the impact of FR-PCA on accuracies of three binary classification algorithms is studied. In Sec. 5, we evaluate the impact of FR-PCA on efficiencies of aforementioned five classifiers through extensive experimental studies conducted on several benchmark face image databases. Finally, in Sec. 6, we will offer a brief conclusion and several suggestions for future work.

2. The Full Rank Principal Component Analysis

Assume $\mathbf{u}_1, \ldots, \mathbf{u}_N \in R^d$ to be a set of d-dimensional training samples. The total scatter matrix of these samples is defined as follows:

$$S_T = \sum_{i=1}^{N} (\mathbf{u}_i - \mathbf{m})(\mathbf{u}_i - \mathbf{m})^T \in R^{d \times d}, \tag{1}$$

where

$$\mathbf{m} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{u}_i \tag{2}$$

is the average sample.

Since S_T is a positive semi-definite matrix, its eigenvalues are all non-negative real numbers. Let $\lambda_1, \ldots, \lambda_d$ be all of the eigenvalues of S_T in decreasing order and $\varphi_1, \ldots, \varphi_d$ their corresponding orthonormal eigenvectors.

On account of the fact that $\varphi_1, \ldots, \varphi_d$ is an orthonormal basis of R^d, each vector $\mathbf{x} \in R^d$ can be represented as $\mathbf{x} = \sum_{i=1}^{d} (\varphi_i^T \mathbf{x}) \varphi_i$, i.e. the linear combination of all of its components. By choosing part principal components, say, retaining the first n principal components and omitting the rest, we obtain an approximation of $\mathbf{x}$ as $\sum_{i=1}^{n} (\varphi_i^T \mathbf{x}) \varphi_i$. The matrix $\Phi_n = [\varphi_1, \ldots \varphi_n] \in R^{d \times n}$ is called the transformation matrix of PCA, which can be used to compress a d-dimensional sample vector $\mathbf{x}$ to a n-dimensional feature vector $\Phi_n^T \mathbf{x}$. Since n (a parameter chosen by the user) is usually smaller than d (the dimension of the input space), PCA is often used as a feature extraction or dimension reduction technique.

If the rank of S_T is r ($r \leq \min(d, N-1)$), then all eigenvalues of S_T are zeros except for the first r ones. Denote $\Phi = [\varphi_1, \ldots, \varphi_r]$ to be the matrix composed of the first r orthonormal eigenvectors which correspond to positive eigenvalues. We call the PCA which uses the matrix Φ as the transformation matrix, the full rank PCA, abbreviated as FR-PCA. FR-PCA uses $\Phi \Phi^T \mathbf{x} = \sum_{i=1}^{r} (\varphi_i^T \mathbf{x}) \varphi_i$ as the representation of a vector $\mathbf{x}$.

Obviously, the representation derived by FR-PCA is also an approximation of the original sample. While compressing d-dimensional original sample vectors into

(a) (b)

Fig. 1. (a) An original sample from the ORL and (b) the approximation by FR-PCA.

r-dimensional feature vectors, FR-PCA might lose some information present in original samples. Figure 1(a) is an original sample image from the ORL[11] face image database and Fig. 1(b) is a two-dimensional display of the compressed representation derived by FR-PCA for the same sample. The transformation matrix is calculated by using all 400 samples of the database as training samples.

In the sense of data compression, FR-PCA is lossy because it loses a lot of detailed information of the original sample. However, as a dimension reduction or data preprocessing method, we will show in the following sections that FR-PCA retains all discriminantly informative features.

In undersampled classification of high-dimensional data such as face recognition, the dimension of original samples, d, is much larger than the size of the training samples N. Moreover, the transformation matrix Φ of FR-PCA can be readily calculated by using the singular value decomposition theorem as in Ref. 1. Thus, in general, FR-PCA plus a classification algorithm is much more efficient than the classification algorithm itself. This is also confirmed by our experimental results presented in Sec. 5.

The detailed description for FR-PCA is presented in Algorithm 1.

3. Impact of FR-PCA on Multi-Category Classification Algorithms

3.1. *Lemmas and corollaries*

Suppose $S \subset R^d$ is an n-dimensional ($n < d$) subspace, and $\mathbf{w}_1, \ldots, \mathbf{w}_n \in S$ an orthonormal basis of S. Let symbol $W \in R^{d \times n}$ denote the matrix $[\mathbf{w}_1, \ldots, \mathbf{w}_n]$. Since $\text{rank}(WW^T) = \text{rank}(W) = n$, so $WW^T \neq I_d$, where I_d is the identity matrix of order d. Therefore, the equation $WW^T\mathbf{x} = \mathbf{x}$ is not true for $\mathbf{x} \in R^d$ in general. However, the following lemma tells us that if $\mathbf{x} \in S$, we can always gain it by reversing its projection $W^T\mathbf{x}$.

Algorithm 1: Algorithm to compute the transformation matrix of FR-PCA

Input: The training samples, $\mathbf{u}_1, \ldots, \mathbf{u}_N \in R^d$
Output: The transformation matrix of FR-PCA, Φ

1. Calculate the average sample, $\mathbf{m}$ using Eq. (2).
2. Compute the precursor of the total scatter matrix, H_t using the following formula:

$$H_t \leftarrow [\mathbf{u}_1 - \mathbf{m}, \ldots, \mathbf{u}_N - \mathbf{m}].$$

3. Perform eigen decomposition to $H_t^T H_t$ as $H_t^T H_t = V^T D V$.
4. Work out the eigenvector matrix of $S_T = H_t H_t^T$ using the formula

$$\Phi \leftarrow H_t V_r D_r^{-1/2}.$$

// Where D_r is a diagonal matrix with all nonzero eigenvalues, V_r the corresponding eigenvector matrix,

// and r the rank of H_t.

Lemma 1. *For any vector* $\mathbf{x} \in S$, *the equation* $WW^T\mathbf{x} = \mathbf{x}$ *holds.*

Proof. Since $\mathbf{x} \in S$ and $\mathbf{w}_1, \ldots, \mathbf{w}_n$ is an orthonormal basis of S, there are real numbers $x_1, \ldots, x_n$ such that $\mathbf{x} = \sum_{i=1}^n x_i \mathbf{w}_i$.

Therefore,

$$\begin{aligned} WW^T\mathbf{x} &= [\mathbf{w}_1, \ldots, \mathbf{w}_n] \begin{bmatrix} \mathbf{w}_1^T \\ \vdots \\ \mathbf{w}_n^T \end{bmatrix} \sum_{i=1}^n x_i \mathbf{w}_i \\ &= [\mathbf{w}_1, \ldots, \mathbf{w}_n] \begin{bmatrix} \sum_{i=1}^n x_i \mathbf{w}_1^T \mathbf{w}_i \\ \vdots \\ \sum_{i=1}^n x_i \mathbf{w}_n^T \mathbf{w}_i \end{bmatrix} \\ &= [\mathbf{w}_1, \ldots, \mathbf{w}_n] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \mathbf{x}. \end{aligned}$$

This lemma is very important and we will repeatedly appeal to it when we prove corollaries, theorems and other lemmas in this paper.

Corollary 1 immediately follows Lemma 1.

Corollary 1. *For any vector* $\mathbf{x} \in S$*, it follows that* $\|W^T\mathbf{x}\| = \|\mathbf{x}\|$*, where* $\|\cdot\|$ *is the Euclidean norm.*

Proof. $\|W^T\mathbf{x}\| = \sqrt{(W^T\mathbf{x})^T(W^T\mathbf{x})} = \sqrt{\mathbf{x}^T(WW^T\mathbf{x})} \overset{\text{Lemma1}}{=} \sqrt{\mathbf{x}^T\mathbf{x}} = \|\mathbf{x}\|$.

This corollary tells us that the linear transformation W whose column vectors are orthonormal does not change the length of an arbitrary vector in the subspace S spanned by it.

Lemma 2. *Let* $\mathbf{u}_1, \ldots, \mathbf{u}_N \in R^d$ *be a set of training samples as before, and* $\mathbf{m}$*, the global mean as defined in* (2)*. Denote* $\mathbf{v}_i = \mathbf{u}_i - \mathbf{m}$*,* $i = 1, \ldots, N$*, and* $S_N = \text{span}(\{\mathbf{v}_1, \ldots, \mathbf{v}_N\})$*, the subspace generated by the vector set* $\{\mathbf{v}_1, \ldots, \mathbf{v}_N\}$*. Then, the column vectors of the transformation matrix* Φ *of FR-PCA is an orthonormal basis of the subspace* S_N*.*

Proof. We first prove that each eigenvector of S_T which corresponds to a positive eigenvalue belongs to the subspace S_N.

Denote φ to be the eigenvector of S_T which corresponds to the eigenvalue $\lambda(\lambda > 0)$, then we have

$$\varphi = \frac{1}{\lambda} S_T\varphi = \frac{1}{\lambda} \sum_{i=1}^{N} (\mathbf{u}_i - \mathbf{m})(\mathbf{u}_i - \mathbf{m})^T\varphi = \frac{1}{\lambda} \sum_{i=1}^{N} \mathbf{v}_i\mathbf{v}_i^T\varphi. \tag{3}$$

Let

$$a_i = \frac{1}{\lambda}\mathbf{v}_i^T\varphi, \quad i = 1, \ldots, N. \tag{4}$$

Substitute (4) into (3) and it follows that $\varphi = \sum_{i=1}^{N} a_i\mathbf{v}_i \in S_N$.

This indicates that $S \subset S_N$, where S is the subspace spanned by all eigenvectors of S_T which correspond to positive eigenvalues.

On the other hand, we have

$$\begin{aligned} \dim(S) &= \text{rank}(S_T) = \text{rank}\left(\sum_{i=1}^{N} (u_i - m)(u_i - m)^T\right) \\ &= \text{rank}(H_t H_t^T) = \text{rank}(H_t) = \dim(S_N), \end{aligned}$$

where $H_t = [\mathbf{u}_1 - \mathbf{m}, \ldots, \mathbf{u}_N - \mathbf{m}] = [\mathbf{v}_1, \ldots, \mathbf{v}_N]$ is the precursor of the total scatter matrix.

Thus, we can conclude that $S = S_N$.

In view of the fact that the column vectors of Φ is an orthonormal basis of the subspace S, we complete the proof of the lemma.

3.2. *Two theorems for FR-PCA transformation*

Theorem 1. *The Euclidean distance between each pair of training samples is equal to the Euclidean distance between their projections under an FR-PCA transformation.*

Proof. For two arbitrary training samples $\mathbf{x}_i, \mathbf{x}_j \in \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$, we have

$$\mathbf{x}_i - \mathbf{x}_j = (\mathbf{x}_i - \mathbf{m}) - (\mathbf{x}_j - \mathbf{m}) = \mathbf{v}_i - \mathbf{v}_j \in S_N.$$

According to Corollary 1 and Lemma 2, it follows that

$$d(\Phi^T\mathbf{x}_i, \Phi^T\mathbf{x}_j) = \|\Phi^T(\mathbf{x}_i - \mathbf{x}_j)\| = \|\mathbf{x}_i - \mathbf{x}_j\| = d(\mathbf{x}_i, \mathbf{x}_j).$$

Theorem 1 indicates that the FR-PCA transformation preserves the Euclidean distance between each pair of training samples.

Theorem 2. *For two arbitrary training samples* $\mathbf{x}_i, \mathbf{x}_j \in \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$, *and any test sample* $\mathbf{x} \in R^d$, *it follows that* $d(\mathbf{x}, \mathbf{x}_1) < d(\mathbf{x}, \mathbf{x}_2)$ *if and only if* $d(\Phi^T\mathbf{x}, \Phi^T\mathbf{x}_1) < d(\Phi^T\mathbf{x}, \Phi^T\mathbf{x}_2)$.

Proof. We rewrite the vector $\mathbf{x}$ as the sum of two vectors, i.e. $\mathbf{x} = \alpha + \beta$, where $\alpha \in S_N, \beta \in S_N^{\perp}$, and $S_N^{\perp}$ is the orthogonal complementary of the subspace S_N.

Since

$$\begin{aligned} d(\mathbf{x}, \mathbf{x}_i) &= \|\mathbf{x} - \mathbf{x}_i\| = \|\alpha - (\mathbf{x}_i - \mathbf{m}) + \beta - \mathbf{m}\| \\ &= \sqrt{\|\alpha - (\mathbf{x}_i - \mathbf{m})\|^2 + \|\beta - \mathbf{m}\|^2 + 2(\beta - \mathbf{m})^T[\alpha - (\mathbf{x}_i - \mathbf{m})]}, \end{aligned}$$

it follows that

$$\begin{aligned} d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j) &\iff \|\alpha - (\mathbf{x}_i - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T\mathbf{x}_i \\ &< \|\alpha - (\mathbf{x}_j - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T\mathbf{x}_j. \end{aligned}$$

On the other hand, since

$$\begin{aligned} &d(\Phi^T\mathbf{x}, \Phi^T\mathbf{x}_i) \\ &\quad = \|\Phi^T(\mathbf{x} - \mathbf{x}_i)\| = \|\Phi^T[\alpha - (\mathbf{x}_i - \mathbf{m}) + \beta - \mathbf{m}]\| \\ &\quad = \sqrt{\|\Phi^T[\alpha - (\mathbf{x}_i - \mathbf{m})]\|^2 + \|\Phi^T(\beta - \mathbf{m})\|^2 + 2(\beta - \mathbf{m})^T\Phi\Phi^T[\alpha - (\mathbf{x}_i - \mathbf{m})]}, \end{aligned}$$

it follows that

$$\begin{aligned} &d(\Phi^T\mathbf{x}, \Phi^T\mathbf{x}_i) < d(\Phi^T\mathbf{x}, \Phi^T\mathbf{x}_j) \\ &\quad \iff \|\Phi^T[\alpha - (\mathbf{x}_i - \mathbf{m})]\|^2 - 2(\beta - \mathbf{m})^T\Phi\Phi^T\mathbf{x}_i \\ &\quad < \|\Phi^T[\alpha - (\mathbf{x}_j - \mathbf{m})]\|^2 - 2(\beta - \mathbf{m})^T\Phi\Phi^T\mathbf{x}_j \\ &\quad \overset{\text{Lemma 2, Corollary 1}}{\iff} \|\alpha - (\mathbf{x}_i - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T\Phi\Phi^T(\mathbf{x}_i - \mathbf{m}) \\ &\quad < \|\alpha - (\mathbf{x}_j - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T\Phi\Phi^T(\mathbf{x}_j - \mathbf{m}) \\ &\quad \overset{\text{Lemmas 1 \& 2}}{\iff} \|\alpha - (\mathbf{x}_i - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T(\mathbf{x}_i - \mathbf{m}) \\ &\quad < \|\alpha - (\mathbf{x}_j - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T(\mathbf{x}_j - \mathbf{m}) \\ &\quad \iff \|\alpha - (\mathbf{x}_i - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T\mathbf{x}_i < \|\alpha - (\mathbf{x}_j - \mathbf{m})\|^2 - 2(\beta - \mathbf{m})^T\mathbf{x}_j \\ &\quad \iff d(\mathbf{x}, \mathbf{x}_i) < d(\mathbf{x}, \mathbf{x}_j). \end{aligned}$$

Thus, we complete the proof of the theorem.

Theorem 2 tells us that the FR-PCA transformation does not change the relative spatial distributions of all samples (including both training and test samples).

3.3. *Impact of FR-PCA on the recognition accuracy of k-NN*

It is well known that the k-NN algorithm classifies a test sample by a majority vote of its neighbors, with the test sample being assigned to the class most common amongst its k nearest training samples. According to Theorem 2, it is easy to conclude that training samples $\mathbf{x}_1, \ldots, \mathbf{x}_k$ are the k nearest neighbors of a test sample $\mathbf{x}$ if and only if $\Phi^T\mathbf{x}_1, \ldots, \Phi^T\mathbf{x}_k$ are the k nearest neighbors of $\Phi^T\mathbf{x}$. This indicates that for a given positive integer k, the class label of $\Phi^T\mathbf{x}$ is always the same as the one of $\mathbf{x}$ assigned by the k-NN classifier.

Since the NN (nearest neighbor) classifier is a special instance of k-NN classifiers, based on above discussion, we know that the FR-PCA transformation does not change the recognition accuracy of an NN, which is the most popular classifier used in face recognition.

3.4. *The impact of FR-PCA on the recognition accuracy of MD*

The MD classifier assigns a test sample $\mathbf{x}$ to the class whose centroid (i.e. the mean of samples from the class) is nearest to $\mathbf{x}$.

Assume $\mathbf{u}_1, \ldots, \mathbf{u}_N \in R^d$ to be all training samples, and $\mathbf{u}_1^{(i)}, \ldots, \mathbf{u}_{N_i}^{(i)}$ and $\mathbf{u}_1^{(j)}, \ldots, \mathbf{u}_{N_j}^{(j)}$ to be all training samples from the ith and the jth classes, respectively. Denote $\mathbf{m}_i = \frac{1}{N_i}\sum_{k=1}^{N_i}\mathbf{u}_k^{(i)}$ and $\mathbf{m}_j = \frac{1}{N_j}\sum_{k=1}^{N_j}\mathbf{u}_k^{(j)}$ to be the centroids of the ith and jth classes.

Since $\mathbf{m}_i - \mathbf{m} = \frac{1}{N_i}\sum_{k=1}^{N_i}(\mathbf{u}_k^{(i)} - \mathbf{m}) \in S_N$ and $\mathbf{m}_j - \mathbf{m} \in S_N$, it is obvious that

$$\begin{aligned} d(\mathbf{x}, \mathbf{m}_i) < d(\mathbf{x}, \mathbf{m}_j) \quad &\Longleftrightarrow \quad d(\mathbf{x} - \mathbf{m}, \mathbf{m}_i - \mathbf{m}) < d(\mathbf{x} - \mathbf{m}, \mathbf{m}_j - \mathbf{m}) \\ &\overset{\text{Theorem 2}}{\Longleftrightarrow} \quad d(\Phi^T(\mathbf{x} - \mathbf{m}), \Phi^T(\mathbf{m}_i - \mathbf{m})) \\ &< \quad d(\Phi^T(\mathbf{x} - \mathbf{m}), \Phi^T(\mathbf{m}_j - \mathbf{m})) \\ &\Longleftrightarrow \quad d(\Phi^T\mathbf{x}, \Phi^T\mathbf{m}_i) < d(\Phi^T\mathbf{x}, \Phi^T\mathbf{m}_j). \end{aligned}$$

This indicates that the class label of $\Phi^T\mathbf{x}$ is always the same as the one of $\mathbf{x}$ assigned by the MD classifier. Thus, the FR-PCA transformation does not change the recognition accuracy of MD.

4. Impact of FR-PCA on Binary Classification Algorithms

Unlike k-NN and MD which can be directly applied to multi-category classification, SVM, LMLP, and MSD are all binary classifiers in nature. There are three steps to apply a binary classification algorithm, say, SVM to a multi-category classification problem. At first, we have to break a multi-category classification problem into a series of binary ones. There are mainly three decomposition strategies: "one-vs-one", "one-vs-rest", and "DAG".[10] Then, we train an SVM classifier for each of these

binary classification problems. Finally, we combine class labels assigned by all of these SVM classifiers into one for a given test sample. If we can prove that for a given test sample in each binary classification problem, the class label assigned by the SVM classifier trained on original samples is the same as the one assigned by the SVM classifier trained on FR-PCA transformed samples, we can conclude that FR-PCA has no impact on the recognition accuracy of SVM regardless of the decomposition strategy used.

4.1. *The impact of FR-PCA on the recognition accuracy of SVM*

To clearly discuss SVM[2,27] and other binary classification algorithms, we need to introduce a new set of symbols.

Suppose $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$ are training samples for a certain binary classification problem. Their class labels are $y_1, \ldots, y_n \in \{-1, 1\}$, respectively. The trained SVM classifier is defined as $f(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + b)$ which assigns a test sample $\mathbf{x}$ to class 1 if $\mathbf{w}^T\mathbf{x} + b > 0$ or class -1 if $\mathbf{w}^T\mathbf{x} + b \leq 0$. The weight $\mathbf{w}$ and the bias b constitute the unique optimal solution of the following quadratic programming problem:

Minimize:

$$\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i, \tag{5}$$

subject to:

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, i = 1, \ldots, n, \tag{6}$$

where C is a parameter which balances the training error against the margin between two trained separating hyperplanes.

Solving the optimization model is equivalent to solving the following Wolfe dual of the Lagrangian formulation of the quadratic programming problem[2]:

Maximize:

$$\sum_i \alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \mathbf{x}_i^T\mathbf{x}_j, \tag{7}$$

subject to:

$$0 \leq \alpha_i \leq C, \tag{8}$$

$$\sum_i \alpha_i y_i = 0. \tag{9}$$

The solution is given by

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i, \tag{10}$$

and

$$b = y_j - \mathbf{w}^T \mathbf{x}_j, \tag{11}$$

for any j, if $0 < \alpha_j < C$.

Similarly, training an SVM classifier on reduced samples $\Phi^T \mathbf{x}_1, \ldots, \Phi^T \mathbf{x}_n$ is equivalent to solving the following quadratic programming problem:
Maximize:

$$\sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j (\Phi^T \mathbf{x}_i)^T (\Phi^T \mathbf{x}_j), \tag{12}$$

subject to constraints (8) and (9).

The solution is given by

$$\mathbf{w}_1 = \sum_i \alpha_i y_i \Phi^T \mathbf{x}_i, \tag{13}$$

and

$$b_1 = y_j - \mathbf{w}_1^T \Phi^T \mathbf{x}_j, \tag{14}$$

for any j, if $0 < \alpha_j < C$.

Since

$$\begin{aligned}
&\sum_{i,j} \alpha_i \alpha_j y_i y_j (\Phi^T \mathbf{x}_i)^T (\Phi^T \mathbf{x}_j) \\
&\quad = \left(\sum_i \alpha_i y_i \Phi^T \mathbf{x}_i \right)^T \left(\sum_j \alpha_j y_j \Phi^T \mathbf{x}_j \right) \\
&\quad = \left[\sum_i \alpha_i y_i \Phi^T (\mathbf{x}_i - \mathbf{m}) + \left(\sum_i \alpha_i y_i \right) \Phi^T \mathbf{m} \right]^T \\
&\qquad \times \left[\sum_j \alpha_j y_j \Phi^T (\mathbf{x}_j - \mathbf{m}) + \left(\sum_j \alpha_j y_j \right) \Phi^T \mathbf{m} \right] \\
&\quad \overset{(9)}{=} \left[\sum_i \alpha_i y_i \Phi^T (\mathbf{x}_i - \mathbf{m}) \right]^T \left[\sum_j \alpha_j y_j \Phi^T (\mathbf{x}_j - \mathbf{m}) \right] \\
&\quad = \sum_{i,j} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i - \mathbf{m})^T \Phi \Phi^T (\mathbf{x}_j - \mathbf{m}) \\
&\quad \overset{\text{Lemma 1}}{=} \sum_{i,j} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i - \mathbf{m})^T (\mathbf{x}_j - \mathbf{m}) \\
&\quad = \left[\sum_i \alpha_i y_i \mathbf{x}_i - \left(\sum_i \alpha_i y_i \right) \mathbf{m} \right]^T \left[\sum_j \alpha_j y_j \mathbf{x}_j - \left(\sum_j \alpha_j y_j \right) \mathbf{m} \right] \\
&\quad \overset{(9)}{=} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j
\end{aligned}$$

thus we can rewrite (12) as (7). This indicates that the objective functions of these two Wolfe duals are identical.

In consideration of the fact that they are subjected to the same constraints, they must share a unique optimal solution $\alpha_1, \ldots, \alpha_n$.

Now, suppose that $\mathbf{w}_0$ and b_0 are the weight and bias trained by SVM on original training samples and $\alpha_1, \ldots, \alpha_n$ the optimal solution of the Wolfe dual of its Lagrangian formulation. Then, we have $\mathbf{w}_0 = \sum_i \alpha_i y_i \mathbf{x}_i$ and $b_0 = y_j - \mathbf{w}^T \mathbf{x}_j$ for some j, if $0 < \alpha_j < C$.

Since $\Phi^T \mathbf{w}_0 = \sum_i \alpha_i y_i \Phi^T \mathbf{x}_i$, thus, $\Phi^T \mathbf{w}_0$ is just the weight trained by the SVM on samples $\Phi^T \mathbf{x}_1, \ldots, \Phi^T \mathbf{x}_n$.

According to (14), b_1 can be computed as follows:

$$\begin{aligned} b_1 &= y_j - \mathbf{w}_1^T \Phi^T \mathbf{x}_j = y_j - (\Phi^T \mathbf{w}_0) \Phi^T \mathbf{x}_j = y_j - \left(\sum_i \alpha_i y_i \Phi^T x_i \right)^T \Phi^T \mathbf{x}_j \\ &= y_j - \sum_i \alpha_i y_i \mathbf{x}_i^T \Phi \Phi^T \mathbf{x}_j \overset{(9)}{=} y_j - \sum_i \alpha_i y_i (\mathbf{x}_i - \mathbf{m})^T \Phi \Phi^T \mathbf{x}_j \\ &\overset{\text{Lemma 1}}{=} y_j - \sum_i \alpha_i y_i (\mathbf{x}_i - \mathbf{m})^T \mathbf{x}_j \overset{(9)}{=} y_j - \sum_i \alpha_i y_i \mathbf{x}_i^T \mathbf{x}_j = y_j - \mathbf{w}_0^T \mathbf{x}_j = b_0. \end{aligned}$$

To sum up the above discussions, we find that if $f_0(\mathbf{x}) = \text{sign}(\mathbf{w}_0^T \mathbf{x} + b_0)$ is the classifier trained by SVM on the original samples, $f_1(\mathbf{x}) = \text{sign}(\mathbf{w}_0^T \Phi \mathbf{x} + b_0)$ must be the classifier trained by SVM on the FR-PCA transformed samples.

Furthermore, it is not difficult to prove that if $f_1(\mathbf{x}) = \text{sign}(\mathbf{w}_1^T \mathbf{x} + b_1)$ is the classifier trained by SVM on the transformed samples by FR-PCA, $f_0(\mathbf{x}) = \text{sign}(\mathbf{w}_1^T \Phi^T \mathbf{x} + b_1)$ must be the classifier trained by SVM on the original samples.

Finally, to prove that the FR-PCA transformation does not change the recognition accuracy of SVM, we only need to prove that

$$f_0(\mathbf{x}) = \text{sign}(\mathbf{w}_0^T \mathbf{x} + b_0) = \text{sign}(\mathbf{w}_0^T \Phi \Phi^T \mathbf{x} + b_0) = f_1(\Phi^T \mathbf{x}). \tag{15}$$

Since

$$\begin{aligned} \mathbf{w}_0^T \Phi \Phi^T \mathbf{x} &= \left(\sum_i \alpha_i y_i \mathbf{x}_i \right)^T \Phi \Phi^T \mathbf{x} \\ &= \sum_i \alpha_i y_i \mathbf{x}_i^T \Phi \Phi^T \mathbf{x} \\ &\overset{(9)}{=} \sum_i \alpha_i y_i (\mathbf{x}_i - \mathbf{m})^T \Phi \Phi^T \mathbf{x} \\ &\overset{\text{Lemma 1}}{=} \sum_i \alpha_i y_i (\mathbf{x}_i - \mathbf{m})^T \mathbf{x} \\ &\overset{(9)}{=} \sum_i \alpha_i y_i \mathbf{x}_i^T \mathbf{x} = \mathbf{w}_0^T \mathbf{x}. \end{aligned}$$

So (15) always holds.

Thus, we have completely proven that as a data preprocessing method, FR-PCA has no impact on the recognition accuracy of SVM algorithms.

4.2. *Impact of FR-PCA on the recognition accuracy of LMLP*

An LMLP[24] algorithm is specially designed for undersampled classification problems. Before a formal discussion, we need to introduce several notations.

Let ω_1 and ω_2 be the set of training samples with positive and negative class labels, respectively, and $\mathbf{m}_1$ and $\mathbf{m}_2$ the mean vectors for samples from the positive class ω_1 and negative class ω_2, respectively. The between-class scatter matrix (S_b) and within-class scatter matrix (S_w) are defined as:

$$S_b = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T, \tag{16}$$

$$S_w = \sum_{i=1}^{2} \sum_{\mathbf{x} \in \omega_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T. \tag{17}$$

The weight $\mathbf{w}_0$ trained by LMLP is the unique optimal solution of the following optimization model.

Maximize:

$$\frac{\mathbf{w}^T S_b \mathbf{w}}{\mathbf{w}^T \mathbf{w}}, \tag{18}$$

subject to:

$$\mathbf{w}^T S_w \mathbf{w} = 0. \tag{19}$$

Once the weight w_0 is derived, the bias b_0 is calculated by using the formula

$$b_0 = -\mathbf{w}_0^T (\mathbf{m}_1 + \mathbf{m}_2). \tag{20}$$

Now, by mapping training samples to their projections by using the transformation matrix of FR-PCA, we obtain a training sample $\Phi^T \mathbf{x}_1, \ldots, \Phi^T \mathbf{x}_n$ in the FR-PCA transformed space.

The weight $\mathbf{w}_1$ trained by LMLP in the FR-PCA transformed space is a unique optimal solution of the following optimization model.

Maximize:

$$\frac{\mathbf{w}^T S_b^{\Phi} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}, \tag{21}$$

subject to:

$$\mathbf{w}^T S_{\mathbf{w}}^{\Phi} \mathbf{w} = 0. \tag{22}$$

Once the weight $\mathbf{w}_1$ is derived, the bias b_1 is calculated by using the formula

$$b_1 = -\mathbf{w}_1^T (\mathbf{m}_1^{\Phi} + \mathbf{m}_2^{\Phi}), \tag{23}$$

where S_b^{Φ}, S_w^{Φ}, $\mathbf{m}_1^{\Phi}$, and $\mathbf{m}_2^{\Phi}$ are the between-class scatter matrix, within-class scatter matrix, mean vectors for samples from the positive class, and mean vector for samples from the negative class, respectively, in the FR-PCA transformed space.

It is well known that[6]

$$S_b^{\Phi} = \Phi^T S_b \Phi, \quad S_w^{\Phi} = \Phi^T S_w \Phi, \tag{24}$$

and

$$\mathbf{m}_{i1}^{\Phi} = \Phi^T \mathbf{m}_i, \quad i = 1, 2. \tag{25}$$

In the following paragraphs, we will prove that $\mathbf{w}_0^T \mathbf{x} + b_0 > 0$ if and only if $\mathbf{w}_1^T \Phi^T \mathbf{x} + b_1 > 0$.

First, we will prove three lemmas.

Lemma 3. $\Phi\Phi^T S_b = S_b, \Phi\Phi^T S_w = S_w$.

Proof.

$$\begin{aligned}
\Phi\Phi^T S_b &= \Phi\Phi^T(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \\
&= \Phi\Phi^T[(\mathbf{m}_1 - \mathbf{m}) - (\mathbf{m}_2 - \mathbf{m})](\mathbf{m}_1 - \mathbf{m}_2)^T \\
&= [\Phi\Phi^T(\mathbf{m}_1 - \mathbf{m}) - \Phi\Phi^T(\mathbf{m}_2 - \mathbf{m})](\mathbf{m}_1 - \mathbf{m}_2)^T \\
&\overset{\text{Lemma 1}}{=} [(\mathbf{m}_1 - \mathbf{m}) - (\mathbf{m}_2 - \mathbf{m})](\mathbf{m}_1 - \mathbf{m}_2)^T \\
&= (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T = S_b,
\end{aligned}$$

$$\begin{aligned}
\Phi\Phi^T S_w &= \Phi\Phi^T \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \\
&= \Phi\Phi^T \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} [(\mathbf{x} - \mathbf{m}) - (\mathbf{m}_i - \mathbf{m})](\mathbf{x} - \mathbf{m}_i)^T \\
&= \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} [\Phi\Phi^T(\mathbf{x} - \mathbf{m}) - \Phi\Phi^T(\mathbf{m}_i - \mathbf{m})](\mathbf{x} - \mathbf{m}_i)^T \\
&\overset{\text{Lemma 1}}{=} \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} [(\mathbf{x} - \mathbf{m}) - (\mathbf{m}_i - \mathbf{m})](\mathbf{x} - \mathbf{m}_i)^T \\
&= \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T = S_w.
\end{aligned}$$

Lemma 4. $S_b\Phi\Phi^T = S_b$, $S_w\Phi\Phi^T = S_w$.

Proof. Based on Lemma 3 and the symmetrical properties of S_b and S_w, the conclusion is obvious.

Lemma 5. *For any* $\mathbf{w} \in R^d$, *if* $\mathbf{w}^T S_w \mathbf{w} = 0$, *then* $\Phi\Phi^T \mathbf{w} = \mathbf{w}$.

Proof. Based on (17) we know that

$$\mathbf{w}^T S_w \mathbf{w} = 0 \Longleftrightarrow \forall\ i \in \{1,2\}, \forall\ \mathbf{x} \in \omega_i, \mathbf{w}^T(\mathbf{x} - \mathbf{m}_i) = 0,$$

since

$$\mathbf{x} - \mathbf{m}_i = (\mathbf{x} - \mathbf{m}) - (\mathbf{m}_i - \mathbf{m}) \in S_N.$$

Then, from the condition $\mathbf{w}^T S_w \mathbf{w} = 0$, we can conclude that $\mathbf{w}$ is a linear combination of $\mathbf{x} - \mathbf{m}_i, \mathbf{x} \in \omega_i, i = 1, 2$. Thus, $\mathbf{w} \in S_N$.

According to Lemma 1, it is obvious that $\Phi\Phi^T\mathbf{w} = \mathbf{w}$.

Now, we begin to prove that $\mathbf{w}_1 = \Phi^T\mathbf{w}_0$.
Since

$$\begin{aligned}
\mathbf{w}^T S_w \mathbf{w} = 0 &\overset{\text{Lemma 3}}{\Longleftrightarrow} \mathbf{w}^T\Phi\Phi^T S_w \mathbf{w} = 0 \\
&\overset{\text{Lemma 4}}{\Longleftrightarrow} \mathbf{w}^T\Phi\Phi^T S_w \Phi\Phi^T \mathbf{w} = 0 \\
&\Longleftrightarrow (\Phi^T\mathbf{w})^T(\Phi^T S_w \Phi)(\Phi^T\mathbf{w}) = 0 \\
&\overset{(24)}{\Longleftrightarrow} (\Phi^T\mathbf{w})^T S_w^{\Phi}(\Phi^T\mathbf{w}) = 0
\end{aligned}$$

and for any $\mathbf{w} \in R^d$ which satisfies $\mathbf{w}^T S_w \mathbf{w} = 0$, we have

$$\begin{aligned}
\frac{\mathbf{w}^T S_b \mathbf{w}}{\mathbf{w}^T\mathbf{w}} &\overset{\text{Lemma 3}}{=} \frac{\mathbf{w}^T\Phi\Phi^T S_b \mathbf{w}}{\mathbf{w}^T\mathbf{w}} \\
&\overset{\text{Lemma 4}}{=} \frac{\mathbf{w}^T\Phi\Phi^T S_b \Phi\Phi^T \mathbf{w}}{\mathbf{w}^T\mathbf{w}} \\
&\overset{\text{Lemma 5}}{=} \frac{\mathbf{w}^T\Phi\Phi^T S_b \Phi\Phi^T \mathbf{w}}{\mathbf{w}^T\Phi\Phi^T\mathbf{w}} \\
&= \frac{(\Phi^T\mathbf{w})^T(\Phi^T S_b \Phi)(\Phi^T\mathbf{w})}{(\Phi^T\mathbf{w})^T(\Phi^T\mathbf{w})} \\
&\overset{(24)}{=} \frac{(\Phi^T\mathbf{w})^T S_b^{\Phi}(\Phi^T\mathbf{w})}{(\Phi^T\mathbf{w})^T(\Phi^T\mathbf{w})}.
\end{aligned}$$

This indicates that $\mathbf{w}_0$ is the optimal solution of the model in (18) and (19) if and only if $\Phi^T\mathbf{w}_0$ is the optimal solution of the model in (21) and (22). Thus, we have $\mathbf{w}_1 = \Phi^T\mathbf{w}_0$.

Based on this fact, (23), and (25), it is easy to derive that $b_1 = b_0$.

Thus, we have completely proven that as a data preprocessing method, FR-PCA has no impact on the recognition accuracy of LMLP algorithms.

4.3. *Impact of FR-PCA on the recognition accuracy of MSD*

In contrast to LMLP which can only be used in undersampled classification problems, MSD[25] can be applied to both small and large sample size recognition tasks.

The weight $\mathbf{w}_0$ trained by MSD is the unique optimal solution of the following optimization model.

Maximize:

$$\frac{\mathbf{w}^T(S_b - C \cdot S_w)\mathbf{w}}{\mathbf{w}^T\mathbf{w}}, \tag{26}$$

where the parameter C is a non-negative real number which balances the objective function of maximizing the between-class scatter and objective function of minimizing the within-class scatter.

In fact, $\mathbf{w}_0$ is the unitary eigenvector of the matrix $(S_b - C \cdot S_w)$ that corresponds to the largest eigenvalue λ_0.

Once $\mathbf{w}_0$ is calculated, the bias b_0 is computed by using (20).

The weight $\mathbf{w}_1$ trained by MSD in the FR-PCA transformed space is the unitary eigenvector of the matrix $(S_b^\Phi - C \cdot S_w^\Phi)$ that corresponds to the largest eigenvalue λ_1.

Once $\mathbf{w}_1$ is calculated, the bias b_1 is computed by using (23).

First, we prove the following lemma.

Lemma 6. *If* $\mathbf{w}$ *is an eigenvector of the matrix* $(S_b - C \cdot S_w)$ *that corresponds to a nonzero eigenvalue then* $\Phi\Phi^T\mathbf{w} = \mathbf{w}$.

Proof. We only need to prove that $\mathbf{w} \in S_N$.

Since $\mathbf{w}$ is an eigenvector of the matrix $(S_b - C \cdot S_w)$ that corresponds to a nonzero eigenvalue, there is a real number $\lambda \neq 0$ such that

$$(S_b - C \cdot S_w)\mathbf{w} = \lambda\mathbf{w}. \tag{27}$$

We divide the vector $\mathbf{w}$ into two parts, that is,

$$\mathbf{w} = \alpha + \beta, \tag{28}$$

where $\alpha \in S_N$, $\beta \in S_N^\perp$, $S_N^\perp$ is the orthogonal complimentary of the subspace S_N.

Substituting (28) into (27) and simplifying, it follows that

$$\beta = \frac{1}{\lambda}[(S_b - C \cdot S_w)\alpha - \lambda\alpha + (S_b - C \cdot S_w)\beta]. \tag{29}$$

Since

$$\begin{aligned}(S_b - C \cdot S_w)\alpha &= (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\alpha - C \\ &\quad \cdot \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T\alpha \\ &= [(\mathbf{m}_1 - \mathbf{m}) - (\mathbf{m}_2 - \mathbf{m})][(\mathbf{m}_1 - \mathbf{m}_2)^T\alpha] - C \\ &\quad \cdot \sum_{i=1}^{2} \sum_{\mathbf{x}\in\omega_i} [(\mathbf{x} - \mathbf{m}) - (\mathbf{m}_i - \mathbf{m})][(\mathbf{x} - \mathbf{m}_i)^T\alpha] \in S_N.\end{aligned}$$

In combination with (29), we can conclude that $\beta \in S_N$.

In consideration of the fact that $\beta \in S_N^\perp$, thus $\beta = 0$, and as a result $\mathbf{w} \in S_N$.

According to Lemma 1, the equation $\Phi\Phi^T\mathbf{w} = \mathbf{w}$ holds.

Now, we begin to prove that $\mathbf{w}_1 = \Phi^T\mathbf{w}_0$.

Suppose $\mathbf{w}$ is an eigenvector of the matrix $(S_b - C \cdot S_w)$ that corresponds to eigenvalue λ ($\lambda \neq 0$).

According to Lemmas 3, 4 and 6 we have

$$\begin{aligned}\lambda &= \frac{\mathbf{w}^T(S_b - C \cdot S_w)\mathbf{w}}{\mathbf{w}^T\mathbf{w}} \\ &= \frac{\mathbf{w}^T\Phi\Phi^T(S_b - C \cdot S_w)\Phi\Phi^T\mathbf{w}}{\mathbf{w}^T\Phi\Phi^T\mathbf{w}} \\ &\overset{(24)}{=} \frac{(\Phi^T\mathbf{w})^T(S_b^\Phi - C \cdot S_w^\Phi)(\Phi^T\mathbf{w})}{(\Phi^T\mathbf{w})^T(\Phi^T\mathbf{w})}.\end{aligned}$$

This indicates that $\Phi^T\mathbf{w}$ is an eigenvector of the matrix $(S_b^\Phi - C \cdot S_w^\Phi)$ that corresponds to eigenvalue λ. In contrary, if $\mathbf{w}$ is an eigenvector of the matrix $(S_b^\Phi - C \cdot S_w^\Phi)$ that corresponds to eigenvalue λ, it is obvious that $\Phi\mathbf{w}$ is an eigenvector of the matrix $(S_b - C \cdot S_w)$ that corresponds to eigenvalue λ.

So $\mathbf{w}_1 = \Phi^T\mathbf{w}_0$. Similar to the proof in Sec. 4.2, we have $b_1 = b_0$.

Thus, we have completely proven that as a data preprocessing method, FR-PCA has no impact on the recognition accuracy of MSD algorithms.

5. Impact of FR-PCA on Efficiencies of Classification Algorithms

In Secs. 3 and 4 we theoretically prove that the transformation by FR-PCA does not change the recognition accuracies of the k-NN, MD, SVM, LMLP, and MSD classification algorithms. In this section, we will demonstrate that FR-PCA can greatly promote the efficiencies of these five classifiers through a series of experimental studies conducted on ORL, AR, FERET, and Extended Yale B face image databases. ORL database is collected by AT&T Laboratories between April 1992 and April 1994. It has a total of 400 images, 10 different images for each of 40 individuals. All images are grayscale and normalized with a resolution of 112×92. To avoid the overflow problem encountered by MSD, we used a pixel grouping[4] technique to reduce the image resolution to 56×46 in our experiment. In each of the ten runs in the experiment, we used five images of each person for training and the remaining five for testing. The images of each person numbered 1 to 5, 2 to $6, \ldots, 10$ to 4 are used as training samples for the first, second, $\ldots$, and the tenth run, respectively.

Table 1 lists total computational times consumed by various classification algorithms (including times consumed by FR-PCA if applicable) for each of the ten runs on ORL database. In this experiment and the following other experiments, a one-vs-one decomposition strategy[10] is adopted. The parameter Cs for both SVM and MSD take 100. The SVM code used in the experiment comes from Ref. 19.

From Table 1, we find that FR-PCA can greatly promote the efficiencies of the NN, SVM, LMLP, and MSD classification algorithms on ORL face image database. The reason why FR-PCA fails to promote the efficiency of MD is that in this small database MD is so efficient that the time saved by FR-PCA cannot compensate for the time consumed by FR-PCA itself.

Table 1. Time consumed by various approaches for each of the ten runs on orl face image database (sec.).

NN	3.922	2.922	3.016	2.938	3.031	3.000	3.031	2.969	3.078	2.938	**3.085**
FR-PCA+NN	1.094	1.047	1.078	1.047	1.047	1.063	1.047	1.047	1.063	1.078	**1.061**
MD	1.047	0.641	0.656	0.641	0.609	0.641	0.656	0.625	0.656	0.641	**0.681**
FR-PCA+MD	0.781	0.766	0.766	0.766	0.750	0.750	0.750	0.750	0.766	0.766	**0.761**
SVM	5.219	4.109	4.172	4.141	4.156	4.141	4.141	4.156	4.172	4.172	**4.258**
FR-PCA+SVM	1.141	1.094	1.109	1.109	1.094	1.094	1.109	1.094	1.109	1.109	**1.106**
LMLP	7.578	6.797	6.750	6.797	6.781	6.734	6.781	6.781	6.750	6.766	**6.852**
FR-PCA+LMLP	1.453	1.406	1.422	1.406	1.406	1.406	1.391	1.406	1.422	1.422	**1.414**
MSD	1097	1083	1082	1086	1078	1074	1076	1075	1074	1076	**1080**
FR-PCA+MSD	14.14	13.63	13.56	13.33	13.66	13.17	13.31	13.39	13.17	13.55	**13.49**

Note: The digits in the last column are the average times consumed by various approaches.

The AR[20] face image database contains over 4000 color face images of 126 people (70 men and 56 women), including frontal views of faces with different facial expressions, lighting conditions and occlusions. The pictures of most persons were taken in two sessions (separated by two weeks). Each session contains 13 color images and 120 individuals (65 men and 55 women) participated in both sessions. The images of these 120 individuals were selected and used in our experiment. Only the full facial images were considered here. That is, there are 1680 images in total, 14 different images for each of the 120 individuals. All images are grayscale and normalized with a resolution of 50×40 and preprocessed using histogram equalization. In the experiment, the seven images taken in the first session for each individual are used for training, the seven images taken in the second session are used for test.

Table 2 lists total computational times consumed by various classification algorithms (including times consumed by FR-PCA if applicable) on AR face image database.

From Table 2, we find that FR-PCA can promote the efficiencies of all of these five classification algorithms on AR face image database.

The subset of the FERET[21] face image database used in the experiment includes 1400 images of 200 individuals. There are seven different images of each person. All images are grayscale and normalized with a resolution of 40×40 and preprocessed using histogram equalization. The experiments use four images of each person for training and the remaining three images for test. The images of each person numbered 1 to 4, 2 to $5, \ldots, 7$ to 3 are used as training samples.

Table 3 lists total computational times consumed by various classification algorithms (including times consumed by FR-PCA if applicable) for each of the seven runs on the FERET face image database.

From Table 3, we find again that FR-PCA can greatly promote the efficiencies of all these five classification algorithms on FERET face image database.

The Extended Yale B database consists of 2414 frontal-face images of 38 individuals.[7] The cropped and normalized 192×168 face images were captured under various laboratory-controlled lighting conditions. To make the MSD algorithm applicable, we downsample the original images to 48×42. For each individual,

Table 2. Time consumed by various approaches on AR face image database (sec.).

NN	52.70
FR-PCA+NN	25.46
MD	6.770
FR-PCA+MD	3.338
SVM	59.23
FR-PCA+SVM	24.62
LMLP	275.9
FR-PCA+LMLP	233.1
MSD	4890
FR-PCA+MSD	1040

Table 3. Time consumed by various approaches for each of the seven runs on FERET face image database (sec.).

NN	100.9	91.17	87.53	87.86	87.94	87.81	88.16	**90.20**
FR-PCA+NN	20.81	17.91	17.92	17.91	17.94	17.89	25.50	**19.41**
MD	13.25	16.64	16.58	16.50	16.69	16.61	16.67	**16.13**
FR-PCA+MD	4.625	4.359	4.375	4.375	4.375	4.359	6.313	**4.683**
SVM	120.6	125.8	125.1	125.5	125.4	125.8	125.7	**124.8**
FR-PCA+SVM	71.56	66.14	66.23	66.42	66.69	66.53	113.7	**73.89**
LMLP	513.9	566.8	546.4	567.5	557.5	549.0	566.3	**552.5**
FR-PCA+LMLP	424.4	358.0	359.5	364.6	357.7	362.1	795.7	**431.7**
MSD	10580	10540	10540	10540	10540	10550	11550	**10690**
FR-PCA+MSD	3152	2954	2955	2945	2945	2951	3820	**3103**

Table 4. Time consumed by various approaches on Extended Yale B face image database (sec.).

NN	112.9
FR-PCA+NN	85.38
MD	3.167
FR-PCA+MD	2.418
SVM	24.90
FR-PCA+SVM	15.26
LMLP	36.06
FR-PCA+LMLP	32.11
MSD	17780
FR-PCA+MSD	7706

we randomly selected half of his/her images for training (i.e. about 32 images per individual) and the rest were left for testing. The total number of training samples is 1205, and the total number of test samples is 1209.

Table 4 lists total computational times consumed by various classification algorithms (including times consumed by FR-PCA if applicable) on Extended Yale B face image database.

From Table 4, we find that FR-PCA can promote the efficiencies of all these five classification algorithms on Extended Yale B face image database.

6. Conclusion and Further Work

Through the above rigorous mathematical deduction, we have proven that FR-PCA can be used as a data preprocessing approach without the risk of altering the recognition accuracy of the succeeding classification if a k-NN, MD, SVM, LMLP, or MSD algorithm is used. In addition, extensive experimental studies conducted on benchmark face image databases such as ORL, AR, FERET, and Extended Yale B demonstrate that FR-PCA can greatly promote the efficiencies of these five classification algorithms.

Similar to FR-PCA, (nonfull rank) PCA is commonly used as a data preprocessing tool to make certain classifiers applicable to high-dimensional data or to promote their efficiencies. Unlike FR-PCA, (nonfull rank) PCA might change the recognition accuracy of classifier used in the classification. The influence of nonfull rank PCA on the recognition accuracy of a classifier may be positive or negative. What important is that one cannot know how to select the parameter of (nonfull rank) PCA, i.e. the number of principal components such that the influence is positive. FR-PCA let us avoid the trouble of selecting the value of the parameter.

Along each component that corresponds to a zero eigenvalue, all training samples share the same projection, and thus the component makes no contribution in discriminating classes in terms of Euclidean distance. We have proved that FR-PCA does retain all discriminantly informative features for k-NN, MD, SVM, LMLP, and MSD classifiers. An immediate question is whether FR-PCA can retain all

discriminantly informative features for other Euclidean distance-based classifiers, such as quadratic Bayesian and neural networks. Although one might guess that FR-PCA does not change the recognition accuracy of any Euclidean distance-based classifiers, rigorous mathematical proofs for these guesses are needed.

Sparse representation-based classification (SRC)[28] is a completely new type of classification rule in face recognition. Reported experimental results in literature show that SRC outperforms many well-known facial classifiers such as NN and SVM. Another interesting question is whether FR-PCA will change the recognition accuracy of SRC.

Acknowledgments

This work is supported by the National Science Foundation of China under Grant No. 60975006.

References

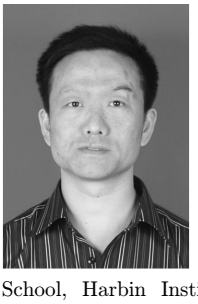
1. P. N. Belhumeur, J. P. Hespanha and D. J. Kriengman, Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection, *IEEE Trans. Pattern Anal. Machine Intell.* **19**(7) (1997) 711−720.
2. C. J. C. Burges, A tutorial on support vector machines for pattern recognition, *Data Min. Knowl. Discov.* **2**(2) (1998) 121−167.
3. S. K. Chen, B. C. Lovell and T. Shan, Robust adapted principal component analysis for face recognition, *Int. J. Pattern Recogn. Artif. Intell.* **23**(3) (2009) 491−520.
4. L. Chen, H. Liao, M. Ko, J. Lin and G. Yu, A new LDA-based face recognition system which can solve the small sample size problem, *Pattern Recogn.* **33**(10) (2000) 1713−1726.
5. X. Chen and J. Zhang, Maximum variance difference based embedding approach for facial feature extraction, *Int. J. Pattern Recogn. Artif. Intell.* **24**(7) (2010) 1047−1060.
6. R. O. Duda, P. E. Hart and D. G. Stork, *Pattern Classification* (John Wiley & Sons, 2001).
7. A. Georghiades, P. Belhumeur and D. Kriegman, From few to many: Illumination cone models for face recognition under variable lighting and pose, *IEEE Trans. Pattern Anal. Machine Intell.* **23**(6) (2001) 643−660.
8. M. Grgic, S. Shan, R. Lukac, H. Wechsler and M. S. Bartlett, Special issue: Facial image processing and analysis; Editorial, *Int. J. Pattern Recogn. Artif. Intell.* **23**(3) (2009) 355−358.
9. Z. He, X. You, Y. Tang, P. S. P. Wang and Y. Xue, Texture image retrieval using novel non-separable filter banks based on centrally symmetric matrices, in *Proc. ICPR2006* (Hong Kong, China, 2006), pp. 161−164.
10. C. W. Hsu and C. J. Lin, A comparison of methods for multi-class support vector machines, *IEEE Trans. Neural Netw.* **13**(2) (2002) 415−425.
11. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.
12. R. Huang, Q. S. Liu, H. Q. Lu and S. D. Ma, Solving the small sample size problem of LDA, in *Proc. Int. Conf. Pattern Recognition* (Quebec, Canada, 2002).
13. M. Kirby and L. Sirovich, Application of the Karhunen-Loeve procedure for the characterization of human faces, *IEEE Trans. Pattern Anal. Machine Intell.* **12**(1) (1990) 103−108.

14. K. V. Kumar and A. Negi, SubXPCA and a generalized feature partitioning approach to principal component analysis, *Pattern Recogn.* **41**(4) (2008) 1398−409.
15. C. Lee and D. A. Landgrebe, Feature extraction based on decision boundaries, *IEEE Trans. Pattern Anal. Machine Intell.* **15**(4) (1993) 388−400.
16. S. W. Lee, P. S. P. Wang, S. N. Yanushkevich and S. W. Lee, Noniterative 3D face reconstruction based on photometric stereo, *Int. J. Pattern Recogn. Artif. Intell.* **22**(3) (2008) 389−410.
17. C. J. Liu, Gabor-based kernel PCA with fractional power polynomial models for face recognition, *IEEE Trans. Pattern Anal. Machine Intell.* **26**(5) (2004) 572−581.
18. Y. Luo, M. L. Gavrilova and P. S. P. Wang, Facial metamorphosis using geometrical methods for biometric applications, *Int. J. Pattern Recogn. Artif. Intell.* **22**(3) (2008) 555−584.
19. J. Ma, Y. Zhao and S. Ahalt, OSU SVM Classifier Matlab Toolbox (ver 3.00), http://sourceforge.net/projects/svm/.
20. A. M. Martinez and R. Benavente, The AR face database, CVC Technical report, No. 24 (1998).
21. P. J. Phillips, H. Moon, S. A. Rizvi and P. J. Rauss, The FERET evaluation methodology for face-recognition algorithms, *IEEE Trans. Pattern Anal. Machine Intell.* **20**(10) (2000) 1090−1104.
22. F. Y. Shih, S. X. Cheng, C. F. Chuang and P. S. P. Wang, Extracting faces and facial features from color images, *Int. J. Pattern Recogn. Artif. Intell.* **22**(3) (2008) 515−534.
23. F. Y. Shih, C. F. Chuang and P. S. P. Wang, Performance comparisons of facial expression recognition in JAFFE database, *Int. J. Pattern Recogn. Artif. Intell.* **22**(3) (2008) 445−459.
24. F. X. Song, J. Y. Yang and S. H. Liu, Large margin linear projection and face recognition, *Pattern Recogn.* **37**(9) (2004) 1953−1955.
25. F. X. Song, D. Zhang, Q. L. Chen and J. Z. Wang, Face recognition based on a novel linear discriminant criterion, *Pattern Anal. Appl.* **10**(3) (2007) 165−174.
26. F. X. Song, D. Zhang, D. Y. Mei and Z. W. Guo, A multiple maximum scatter difference discriminant criterion for facial feature extraction, *IEEE Trans. Syst., Man, Cybern. B, Cybern.* **37**(6) (2007) 1599−1606.
27. V. N. Vapnik, *The Nature of Statistical Learning Theory* (Springer, 1995).
28. J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry and Y. Ma, Robust face recognition via sparse representation, *IEEE Trans. Pattern Anal. Machine Intell.* **31**(2) (2009) 210−227.
29. J. Yang, D. Zhang, A. F. Frangi and J. Y. Yang, Two-dimensional PCA: A new approach to appearance-based face representation and recognition, *IEEE Trans. Pattern Anal. Machine Intell.* **26**(1) (2004) 131−137.
30. X. You, Q. Chen, P. S. P. Wang and D. Zhang, Nontensor-product-wavelet-based facial feature representation, in *Image Pattern Recognition: Synthesis and Analysis in Biometrics*, eds. S. Yanushkevich, M. Gavrilova, P. S. P. Wang and S. Srihari (World Scientific Publishing Company, 2007), pp. 207−224.
31. X. You, D. Zhang, Q. Chen, P. S. P. Wang and Y. Tang, Face representation by using non-tensor product wavelets, in *Proc. ICPR2006* (Hong Kong, China, 2006), pp. 503−506.
32. H. T. Zhao, P. C. Yuen and J. T. Kwok, A novel incremental principal component analysis and its application for face recognition, *IEEE Trans. Syst., Man, Cybern. B, Cybern.* **36**(4) (2006) 873−886.
33. H. T. Zhao, P. C. Yuen and J. Y. Yang, Optimal subspace analysis for face recognition, *Int. J. Pattern Recogn. Artif. Intell.* **19**(3) (2005) 375−393.

Fengxi Song received the B.S. degree in Mathematics at Anhui University, China, in 1984. He received the M.S. degree in Applied Mathematics at Changsha Institute of Technology, China, in 1987 and the Ph.D. in Pattern Recognition and Intelligence Systems, at Nanjing University of Science and Technology, China, in 2004. Now, he is a Professor at New Star Research Institute of Applied Technology in Hefei City, China, and a research fellow in the Department of Computing, Hong Kong Polytechnic University. His research interests include computer vision, machine learning, and automatic text categorization.

Jane You obtained her B.Eng. in Electronic Engineering from Xi'an Jiaotong University in 1986 and Ph.D. in Computer Science from La Trobe University, Australia in 1992. She was a lecturer at the University of South Australia and Senior Lecturer at Griffith University from 1993 till 2002. Currently she is an Associate Professor at the Hong Kong Polytechnic University. Her research interests include image processing, pattern recognition, medical imaging, biometrics computing, multimedia systems and data mining.

David Zhang graduated in Computer Science from Peking University. He received his M.Sc in Computer Science in 1982 and his Ph.D. in 1985 from the Harbin Institute of Technology (HIT). From 1986 to 1988 he was a Postdoctoral Fellow at Tsinghua University and then an Associate Professor at the Academia Sinica, Beijing. In 1994 he received his second Ph.D. in Electrical and Computer Engineering from the University of Waterloo, Ontario, Canada. Currently, he is the Head, Department of Computing, and the Chair Professor at the Hong Kong Polytechnic University where he is the Founding Director of the Biometrics Technology Centre (UGC/CRC) supported by the Hong Kong SAR Government in 1998. He also serves as Visiting Chair Professor in Tsinghua University, and Adjunct Professor in Shanghai Jiao Tong University, Peking University, Harbin Institute of Technology, and the University of Waterloo. He is the Founder and Editor-in-Chief, International Journal of Image and Graphics (IJIG); Book Editor, Springer International Series on Biometrics (KISB); Organizer, the first International Conference on Biometrics Authentication (ICBA); Associate Editor of more than 10 international journals including IEEE Transactions and Pattern Recognition; Technical Committee Chair of IEEE CIS and the author of more than 10 books and 200 journal papers. Professor Zhang is a Croucher Senior Research Fellow, Distinguished Speaker of the IEEE Computer Society, and a Fellow of both IEEE and IAPR.



Yong Xu was born in Sichuan, China, in 1972. He received his B.S. degree and M.S. degree in 1994 and 1997, respectively. He received the Ph.D. in Pattern Recognition and Intelligence System at NUST (China) in 2005. Now he works at Shenzhen Graduate School, Harbin Institute of Technology. His current interests include feature extraction, biometric, face recognition, machine learning and image processing.