Weighted Generalized Kernel Discriminant Analysis Using Fuzzy Memberships

JING YANG
Liaocheng University
School of Mathematics Sciences
Hunan Road 1, 252059, Liaocheng
P.R. CHINA
yangjing860204@163.com

LIYA FAN
Liaocheng University
School of Mathematics Sciences
Hunan Road 1, 252059, Liaocheng
P.R. CHINA
fanliya63@126.com

Abstract: Linear discriminant analysis (LDA) is a classical approach for dimensionality reduction. However, LDA has limitations in that one of the scatter matrices is required to be nonsingular and the nonlinearly clustered structure is not easily captured. In order to overcome these problems, in this paper, we present several generalizations of kernel fuzzy discriminant analysis (KFDA) which include KFDA based on generalized singular value decomposition (KFDA/GSVD), pseudo-inverse KFDA (PIKFDA) and range space KFDA (RSKFDA). These KFDA-based algorithms adopts kernel methods to accommodate nonlinearly separable cases. In order to remedy the problem that KFDA-based algorithms fail to consider that different contribution of each pair of class to the discrimination, weighted schemes are incorporated into KFDA extensions in this paper and called them weighted generalized KFDA algorithms. Experiments on three real-world data sets are performed to test and evaluate the effectiveness of the proposed algorithms and the effect of weights on classification accuracy. The results show that the effect of weighted schemes is very significantly.

Key–Words: Kernel fuzzy discriminant analysis; fuzzy membership; undersampled problem; weighting function; classification accuracy

1 Introduction

Feature extraction process is an important part of pattern recognition and machine learning, which can results in computation cost decreasing and classification performance increasing. An appropriate representation of data from all features is an important problem in machine learning and data mining problems. All original features can not always beneficial for classification or regression tasks. Some features are irrelevant or redundant in distribution of data set. These features can decrease the classification performance. In order to increase the classification performance and to reduce computation cost of classifier, the feature selection process should be used in classification or regression problems [1].

Linear discriminant analysis (LDA) [2] is one of the most popular linear projection techniques for feature extraction, it aims to maximize between-class scatter and minimize within-class scatter, thus maximize the class discriminant. But due to its limitation of linearly, LDA is difficult to capture nonlinear relationships with a linear mapping. To overcome this problem of LDA, the methods which based on the so-called kernel trick have been proposed over the last few years. The essence of kernel discriminant analysis (KDA) is to perform LDA in an implicit high-dimensional feature space[3-4]. In this paper, we extend fuzzy discriminant analysis (FDA) to a nonlinear model, called kernel fuzzy discriminant analysis (KFDA), to deal with nonlinear separable problem. However, similar to the KDA-based methods, the KFDA usually encounters two problems. One is the singularity problem caused by the undersampled problems. We briefly present several KFDA extensions to deal with this problem, such as, KFDA/GSVD, pseudo-inverse KFDA (PIKFDA), range space KFDA (RSKFDA). KFDA/GSVD is one of generalizations of FDA based on kernel functions and GSVD, it overcomes the singularity of the scatter matrices by applying the GSVD to solve the generalized eigenvalue problem in the feature space. The KFDA solution is a special case of KFDA/GSVD method. In PIKFDA, the inverse of the kernel scatter matrix is replaced by the pseudo-inverse. The range space KFDA is a method based on the transformation by a basis of the range space of the within-class scatter matrix of the kernel matrix to handle undersampled problems.

Another drawback of the KFDA method is that it fails to consider that different contribution of each pair of class to the discrimination. A promising solution to this problem is to introduce weighted schemes into the criteria. Motivated by the form of the weighted between-class scatter matrix proposed by Loog et al. [5]. In this paper, we reformulate the KFDA-based methods in the weighted forms, we call them weighted generalized KFDA algorithms, where the weight aims to emphasize the different roles of the individual class pairs in the discrimination. By using the weighted form of the between-class scatter matrix, we can make use of the advantage of the class membership. Moreover, we present weighted versions of KFDA/GSVD, PIKFDA and range space KFDA with five weighting functions for each weighted scheme, where the K-Nearest neighbors (KNN) method [6] is used for a classifier. We apply the Euclidean distance $d_{il} = \|m_i - m_l\|$ between the means of classes i and l in weighting function $w(d_{il})$. A weighting function is generally a monotonically decreasing function because classes that are closer to one another are likely to have a greater confusion and should be given a greater weightage. For weighting functions, we first apply two special cases of the weighting function $w(d_{il}) = (d_{il})^{-p}$ proposed by Lotlikar et al. [7] with $p = 1$ and $p = 3$, and then an improved version of weighting function $w(d_{il}) = \frac{1}{2d_{il}^2} erf(\frac{d_{il}}{2\sqrt{2}})$ is presented by Loog [5] where the Mahanalobis distance is replaced by the Euclidean distance. In addition, according to the feature of weighting functions, we present two new weighting functions. In this paper, we focus on study the effectiveness of the proposed algorithms and the effect of weights on the dimensional algorithms. From recent research in weighted methods, we can see an appropriate choice of the weight on the criterion plays a crucial role in the performance delivered. To further study the effect of weighting functions, we choice three different kernel functions and three real-world data sets in our experiments. Extensive comparisons of different weighting functions on KFDA methods are conducted.

The rest of the paper is organized as follows. In Section 2, we briefly review FDA algorithm. The KFDA and generalized KFDA algorithms are presented in Section 3. Weighted versions of generalized KFDA algorithms and weighting functions are introduced in Section 4. Extensive experiments with proposed algorithms have been performed in Section 5, the results demonstrate the effectiveness of the proposed algorithms. Conclusion follows in Section 6.

2 Related works

2.1 LDA

In this section, we brief review the classical LDA. Given a data matrix $X = [x_1, \cdots, x_N] \in \mathbf{R}^{L\times N}$, where $x_i \in \mathbf{R}^L, i = 1, \cdots N$. Assume the original data is already clustered and partitioned into r classes. Let $X = [X_1, \cdots, X_r]$, where $X_i \in \mathbf{R}^{L\times N_i}$ is the data matrix belonged to the i-th class and $\sum_{i=1}^r N_i = N$. Let N_i be the set of indices of i-th class, i.e., x_j belongs to the i-th class for $j \in N_i$.

The aim of LDA is to find a linear transformation $G \in \mathbf{R}^{L\times m}$ that maps each x_i to $y_i \in \mathbf{R}^m$ by $y_i = G^T x_i$ and optimally preserves the cluster structure in the reduced-dimensional space. Let the between-class, within-class and total scatter matrices are defined as

$$S_b = \sum_{i=1}^{r} n_i(c_i - c)(c_i - c)^T,$$
$$S_w = \sum_{i=1}^{r} \sum_{j\in N_i} (x_j - c_i)(x_j - c_i)^T,$$
$$S_t = \sum_{j=1}^{n} (x_j - c)(x_j - c)^T,$$

where $c_i = (1/N_i)\sum_{j\in N_i} x_j$ and $c = (1/N)\sum_{j=1}^N x_j$ are class means and the global mean, respectively. We can easily see that the trace of S_w measures the within-class closeness and the trace of S_b measures the between-class separation. In the lower-dimensional space obtained from the transformation G, three scatter matrices above become $S_w^L = G^T S_w G, S_b^L = G^T S_b G$ and $S_t^L = G^T S_t G$. An optimal transformation G would maximize trace (S_b^L) and minimize trace (S_w^L), simultaneously. In classical LDA which requires S_w or S_b is nonsingular, common optimizations include

$$\max_G\{\mathrm{trace}((S_w^L)^{-1}S_b^L)\}$$

and

$$\min_G\{\mathrm{trace}((S_b^L)^{-1}S_w^L)\}.$$

The problems above are equivalent to finding the eigenvectors corresponding to the $r-1$ largest eigenvectors for the generalized eigenvalue problem

$$S_b x = \lambda S_w x, \quad \lambda \neq 0.$$

2.2 Fuzzy discriminant analysis

Fuzzy discriminant analysis (FDA) [8-9] is an extension of LDA using fuzzy memberships. Let $X = \{x_1, \cdots, x_N\}$ be a set of samples with fuzzy memberships $U = \{u_{ij} | i = 1, \cdots, r, j = 1, \cdots, N\}$, where r is the number of classes and N is the number of samples. With the FKNN algorithm [8], the computations of the fuzzy membership degrees can be realized through a sequence of steps:

- Step 1: compute the Euclidean distance matrix between pairs of features vectors in the training set.

- Step 2: set diagonal elements of this matrix to infinity (practically place large numeric values there).
- Step 3: sort the distance matrix (treat each of its columns separately) in an ascending order. Collect the corresponding class labels of the pattern located in the closest neighborhood of the pattern under consideration (as we are concerned with 'k' neighbors, this returns a list of 'k' integers).
- Step 4: Compute the membership degree to class 'i' for jth pattern using the expression proposed in the literature [10]:

$$u_{ij} = \begin{cases} \gamma + (1-\gamma)(n_{ij}/k), & i,j \text{ belong to the same class}, \\ (1-\gamma)(n_{ij}/k), & \text{otherwise}, \end{cases}$$

where $\gamma = \frac{N-r}{2^h N}, n_{ij}$ stands for the number of the neighbors of the jth pattern that belongs to the ith class, h and γ are constants that ultimately control the value of u_{ij} and satisfy the constrains $h \in (0,1)$ and $\gamma \in (0,1)$.

The membership degrees should satisfy

$$0 \le u_{ij} \le 1,$$
$$0 < \sum_{i=1}^{r} u_{ij} \le r,$$
$$0 < \sum_{j=1}^{N} u_{ij} \le N,$$

which are relaxed constrains of the ones generally required for fuzzy methods having the sum-to-one constraint:

$$\sum_{i=1}^{r} u_{ij} = 1 \quad (1)$$

The relaxed constrains make it possible for FDA to accommodate measures that do not satisfy the sum-to-one constrain in Eq. (1). The fuzzy number of elements in class i and in the total data set can be defined as:

$$N_i = \sum_{j=1}^{N} u_{ij}^g,$$
$$N_F = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g = \sum_{i=1}^{r} N_i$$

It should be noted that N_F is not equal to N due to the fuzzifier constant and relaxed constraints. Taking into account the fuzzy membership degree, the mean vectors of class i and the total data set are

$$\mu_i = \frac{1}{N_i}\sum_{j=1}^{N} u_{ij}^g x_j$$
$$\mu = \frac{1}{N_F}\sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g x_j = \frac{1}{N_F}\sum_{i=1}^{r} N_i \mu_i.$$

Using the definitions above, the fuzzy between-class, fuzzy within-class and fuzzy total scatter matrices can be defined as

$$S_{fb} = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g (\mu_i - \mu)(\mu_i - \mu)^T,$$
$$S_{fw} = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g (x_j - \mu_i)(x_j - \mu_i)^T,$$
$$S_{ft} = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g (x_j - \mu)(x_j - \mu)^T,$$

and we can easily show $S_{ft} = S_{fb} + S_{fw}$. The discriminant vector can be obtained by solving either of the following generalized eigen-problems:

$$S_{fb}v = \lambda S_{fw}v,$$
$$S_{fb}v = \lambda S_{ft}v.$$

3 Kernel fuzzy discriminant analysis

FDA is linear learning algorithm and it cannot deal with nonlinear problem. In real world, the nonlinear problem always exists. To solve this problem, we introduce kernel methods into FDA to obtain kernel fuzzy discriminant analysis (KFDA). Let $k : \mathbf{R}^L \times \mathbf{R}^L \to \mathbf{R}$ be a kernel function, $\mathcal{F}$ be the reproducing kernel Hilbert space of the kernel k and $\Phi : \mathbf{R}^L \to \mathcal{F}$ be the corresponding feature mapping. Let $X_\Phi = \{\Phi(x_1), \cdots, \Phi(x_N)\}$ be a set of mapped samples in $\mathcal{F}$. Kernel fuzzy between-class, within-class and total scatter matrices can be respectively defined as

$$S_{fb}^{\Phi} = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g (\mu_i^{\Phi} - \mu^{\Phi})(\mu_i^{\Phi} - \mu^{\Phi})^T,$$
$$S_{fw}^{\Phi} = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g (\Phi(x_j) - \mu_i^{\Phi})(\Phi(x_j) - \mu_i^{\Phi})^T,$$
$$S_{ft}^{\Phi} = \sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g (\Phi(x_j) - \mu^{\Phi})(\Phi(x_j) - \mu^{\Phi})^T,$$

where

$$\mu_i^{\Phi} = \frac{1}{N_i}\sum_{j=1}^{N} u_{ij}^g \Phi(x_j),$$
$$\mu^{\Phi} = \frac{1}{N_F}\sum_{i=1}^{r}\sum_{j=1}^{N} u_{ij}^g \Phi(x_j) = \frac{1}{N_F}\sum_{i=1}^{r} N_i \mu_i^{\Phi}.$$

We can see that kernel fuzzy scatter matrices also satisfy the relationship $S_{ft}^{\Phi} = S_{fb}^{\Phi} + S_{fw}^{\Phi}$. The optimal projection directions for KFDA can be obtained by maximizing either of the two criterions in $\mathcal{F}$:

$$J_1(v) = \frac{v^T S_{fb}^{\Phi} v}{v^T S_{fw}^{\Phi} v}, \quad (2)$$

$$J_2(v) = \frac{v^T S_{fb}^{\Phi} v}{v^T S_{ft}^{\Phi} v}. \quad (3)$$

We do not solve this optimization problem directly due to the high or even infinite dimensionality of $\mathcal{F}$. Fortunately, we can show that the eigenvector v must lie in a space spanned by $\{\Phi(x_j)\}_{j=1}^N$ in $\mathcal{F}$ and thus it can be expressed in the form of the following linear expansion :

$$v = \sum_{j=1}^{N} w_j \Phi(x_j). \quad (4)$$

Substituting Eq. (4) into the numerators and denominators of Eq. (2) and (3), we obtain

$$v^T S_{fb}^{\Phi} v = w^T K_{fb} w,$$
$$v^T S_{fw}^{\Phi} v = w^T K_{fw} w,$$
$$v^T S_{ft}^{\Phi} v = w^T K_{ft} w,$$

then we can rewrite the criterions in Eq. (2) and (3) as:

$$J_1'(w) = \frac{w^T K_{fb} w}{w^T K_{fw} w}. \quad (5)$$

$$J_2'(w) = \frac{w^T K_{fb} w}{w^T K_{ft} w}. \quad (6)$$

where

$$w = (w_1, w_2, \cdots, w_N)^T,$$
$$K_{fb} = \sum_{i=1}^{r} \sum_{j=1}^{N} u_{ij}^g (m_i - m)(m_i - m)^T,$$
$$K_{fw} = \sum_{i=1}^{r} \sum_{j=1}^{N} u_{ij}^g (k_j - m_i)(k_j - m_i)^T,$$
$$K_{ft} = \sum_{i=1}^{r} \sum_{j=1}^{N} u_{ij}^g (k_j - m)(k_j - m)^T,$$

with

$$m = \Big(\frac{1}{N_F} \sum_{i=1}^{r} \sum_{j=1}^{N} u_{ij}^g k(x_j, x_1), \frac{1}{N_F} \sum_{i=1}^{r} \sum_{j=1}^{N} u_{ij}^g k(x_j, x_2), \cdots, \frac{1}{N_F} \sum_{i=1}^{r} \sum_{j=1}^{N} u_{ij}^g k(x_j, x_N)\Big)^T,$$

$$m_i = \Big(\frac{1}{N_i} \sum_{j=1}^{N} u_{ij}^g k(x_j, x_1), \frac{1}{N_i} \sum_{j=1}^{N} u_{ij}^g k(x_j, x_2), \cdots, \frac{1}{N_i} \sum_{j=1}^{N} u_{ij}^g k(x_j, x_N)\Big)^T,$$

$$k_j = (k(x_j, x_1), k(x_j, x_2), \cdots, k(x_j, x_N))^T.$$

K_{fb}, K_{fw} and K_{ft} can be seen the scatter matrices of the kernel matrix $K = [k(x_i, x_j)]_{N \times N}$ when each column in K is considered as a point in the N dimensional space, we can easily show that the total scatter matrix $K_{ft} = K_{fb} + K_{fw}$. As a result, the solution to Eq. (2) and can be the m leading eigenvectors $w_1, \cdots, w_m$ of the matrix $K_{fw}^{-1} K_{fb}$.

For any input vector x, its low-dimensional feature representation $y = (y_1, \cdots, y_m)^T$ can then be obtained as

$$y = (w_1, \cdots, w_m)^T (k(x_1, x), \cdots, k(x_N, x))^T.$$

Note that the solution above is based on the assumption that the scatter matrix K_{fw} or K_{ft} is invertible. However, for many real-world applications, this assumption is almost always invalid due to the undersampled problems as discussed above. In order to solve this problem, we can apply several generalizations of FDA algorithms, obtaining nonlinear discriminant analysis methods.

3.1 KFDA based on GSVD

Howland et al.[11] proposed LDA/GSVD, which is an extension of LDA based on GSVD. It overcomes the singularity of the scatter matrices by applying the GSVD to solve the generalized eigenvalue problem. An efficient algorithm for LDA/GSVD was presented in [12]. Note that K_{fb}, K_{fw} and K_{ft} are both singular and KFDA cannot be applied, so as in [13], we can present a generalization of KFDA based on the GSVD, denoted KFDA/GSVD. Similarly to the efficient algorithm for LDA/GSVD, we can present an efficient algorithm for KFDA/GSVD as follows.

Algorithm 3.1. KFDA/GSVD

1. Compute the EVD of K_{ft}:
   $$K_{ft} = [\ U_1\ U_2\ ] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix}.$$
2. Put $\tilde{K}_{fb} = \Sigma_1^{-1/2} U_1^T K_{fb} U_1 \Sigma_1^{-1/2}$ and compute V from the EVD of $\tilde{K}_{fb}$: $\tilde{K}_{fb} = V \Gamma_b^T \Gamma_b V^T$.
3. Assign the first $r - 1$ columns of $U_1 \Sigma_1^{-1/2} V$ to G_h.
4. For any input vector x, its low-dimensional feature representation z can thus be given by $z = G_h^T (k(x_1, x), \cdots, k(x_N, x))^T$.

3.2 Pseudo-inverse KFDA

As discussed the KFDA, we know that K_{fb} and K_{fw} are both singular and the KFDA cannot be applied. One simple method for solving this problem is to use the pseudo-inverse of K_{fw} instead, let us call this method as pseudo-inverse KFDA (PIKFDA).

We can see from [14] that in pseudo-inverse L-DA the inverse of a scatter matrix is replaced by the pseudo-inverse, and LDA/GSVD is a special case of pseudo-inverse LDA. So, similar to Algorithm 3.1, we can derive an efficient algorithm for the pseudo-inverse KFDA as follows.

Algorithm 3.2. Pseudo-inverse KFDA

1. Compute the EVD of K_{ft}:
$K_{ft} = [\ U_1\ U_2\ ] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^T \\ U_2^T \end{bmatrix}$.
2. Put $\tilde{K}_{fb} = \Sigma_1^{-1/2} U_1^T K_{fb} U_1 \Sigma_1^{-1/2}$ and compute V from the EVD of $\tilde{K}_{fb}$: $\tilde{K}_{fb} = V \Gamma_b^T \Gamma_b V^T$.
3. Assign the first $r-1$ columns of $U_1 \Sigma_1^{-1/2} V$ to X_δ.
4. Put $G = X_\delta M$, where M is any nonsingular matrix.
5. For any input vector x, its low-dimensional feature representation z can thus be given by $z = G^T(k(x_1, x), \cdots, k(x_N, x))^T$.

3.3 Range space KFDA

In this subsection, we present a method, in which the kernel trick is incorporated into FDA in the range space of fuzzy within-class scatter matrix, and denote it by RSKFDA for short. This method transforms the feature space by using a basis of range(K_{fw}). Let the EVD of $K_{fw} \in \mathbf{R}^{N\times N}$ be

$$K_{fw} = [\ U_{w1}\ U_{w2}\ ] \begin{bmatrix} \Sigma_{w1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_{w1}^T \\ U_{w2}^T \end{bmatrix},$$

where $s_1 = \text{rank}(K_{fw}), U_{w1} \in \mathbf{R}^{N\times s_1}$ and Σ_{w1} is a diagonal matrix. We can easily show that

$$\text{range}(K_{fw}) = \text{span}(U_{w1})$$

and the transformation by $V_y = U_{w1}\Sigma_{w1}^{-1/2}$ projects the data in the feature space $\mathcal{F}$ to range(K_{fw}). The within-class scatter matrix $\tilde{K}_{fw}$ in the transformed space is $\tilde{K}_{fw} = V_y^T K_{fw} V_y = I_{s_1}$. Let the EVD of $\tilde{K}_{fb} \equiv V_y^T K_{fb} V_y$ be

$$\tilde{K}_{fb} = [\ \tilde{U}_{b1}\ \tilde{U}_{b2}\ ] \begin{bmatrix} \tilde{\Sigma}_{b1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{U}_{b1}^T \\ \tilde{U}_{b2}^T \end{bmatrix},$$

where $s_3 = \text{rank}(\tilde{K}_{fb}), \tilde{U}_{b1} \in \mathbf{R}^{s_1\times s_3}$ and $\tilde{\Sigma}_{b1}$ is a diagonal matrix. In RSKFDA, the optimal transformation matrix G_y is obtained by

$$G_y = V_y \tilde{U}_{b1} = U_{w1}\Sigma_{w1}^{-1/2}\tilde{U}_{b1}.$$

Hence, for any input vector x, its low-dimensional feature representation z can be obtained by $z = G_y^T(k(x_1, x), \cdots, k(x_N, x))^T$.

4 Weighted versions and weighting functions

Similar to the case of LDA-based methods, KFDA-based algorithms also fail to consider that different contribution of each pair of class to the discrimination. If two of the class means are far away from each other, which means that they are well separated, then their contributions to the discrimination task is minor. However, if two of the class means are close together, which means that they are not well separated, then finding the discriminant vectors that can better separate them is important to improve the discriminant performance. So in order to control the contribution of each pair of class to the discrimination, a commonly method is to incorporate a weighting function into the criterion by using a weighted between-class scatter matrix in place of the ordinary between-class scatter matrix. We can rewrite the kernel fuzzy between-class scatter matrix as

$$S_{fb}^\Phi = \sum_{i=1}^{r} N_i(\mu_i^\Phi - \mu^\Phi)(\mu_i^\Phi - \mu^\Phi)^T.$$

We define weighted between-class scatter matrix in $\mathcal{F}$ as follows (see [15-17]):

$$S_{FB}^\Phi = \sum_{i=1}^{r-1}\sum_{l=i+1}^{r} \frac{N_i N_l}{N_F} w(d_{il})(\mu_i^\Phi - \mu_l^\Phi)(\mu_i^\Phi - \mu_l^\Phi)^T, \quad (7)$$

where N_F is same as in Section 2 and $w(d_{il})$ is a weighting function, which is a monotonically decreasing function of the Euclidean distance $d_{il} = \|\mu_i^\Phi - \mu_l^\Phi\|$. Apparently, the weighted kernel fuzzy between-class scatter matrix S_{FB}^Φ degenerates to the kernel fuzzy between-class scatter matrix S_{fb}^Φ if the weighting function in (7) gives a constant weight value. In addition, it is clear that d_{ij} in $\mathcal{F}$ can be calculated by the kernel trick as follows:

$$d_{il} = \left\|\mu_i^\Phi - \mu_l^\Phi\right\| = \sqrt{d_1 - d_2 - d_3 + d_4},$$

where

$$\mu_i^\Phi = \frac{1}{N_i}\sum_{j=1}^{N} u_{ij}^g \Phi(x_j),$$
$$\mu_l^\Phi = \frac{1}{N_l}\sum_{j=1}^{N} u_{lj}^g \Phi(x_j),$$
$$d_1 = \frac{1}{N_i^2}\sum_{j_1=1}^{N}\sum_{j_2=1}^{N} u_{ij_1}^g u_{ij_2}^g k(x_{j_1}, x_{j_2}),$$
$$d_2 = \frac{1}{N_i N_l}\sum_{j_1=1}^{N}\sum_{j_2=1}^{N} u_{ij_1}^g u_{lj_2}^g k(x_{j_1}, x_{j_2}),$$
$$d_3 = \frac{1}{N_l N_i}\sum_{j_1=1}^{N}\sum_{j_2=1}^{N} u_{lj_1}^g u_{ij_2}^g k(x_{j_1}, x_{j_2}),$$
$$d_4 = \frac{1}{N_l^2}\sum_{j_1=1}^{N}\sum_{j_2=1}^{N} u_{lj_1}^g u_{lj_2}^g k(x_{j_1}, x_{j_2}).$$

Consequently, we have the following generalized eigen-problem:

$$S_{FB}^{\Phi}v = \lambda S_{fw}^{\Phi}v. \quad (8)$$

By the theory of reproducing kernel, we can express the solution v of Eq. (8) as $v = \sum_{i=1}^{N} w_i\Phi(x_i)$ and then rewrite Eq. (8) as (see [18]):

$$K_{FB}w = \lambda K_{fw}w, \quad (9)$$

where $w = (w_1, \cdots, w_N)^T$, K_{fw} is same as in Section 3 and

$$K_{FB} = \sum_{i=1}^{r-1}\sum_{l=i+1}^{r} \tfrac{N_iN_l}{N_F} w(d_{il})(m_i - m_l)(m_i - m_l)^T,$$

with

$$m_i = (\tfrac{1}{N_i}\sum_{j=1}^{N} u_{ij}^g k(x_j, x_1), \tfrac{1}{N_i}\sum_{j=1}^{N} u_{ij}^g k(x_j, x_2), \cdots, \tfrac{1}{N_i}\sum_{j=1}^{N} u_{ij}^g k(x_j, x_N))^T,$$
$$m_l = (\tfrac{1}{N_l}\sum_{j=1}^{N} u_{lj}^g k(x_j, x_1), \tfrac{1}{N_l}\sum_{j=1}^{N} u_{lj}^g k(x_j, x_2), \cdots, \tfrac{1}{N_l}\sum_{j=1}^{N} u_{lj}^g k(x_j, x_N))^T.$$

By solving Eq. (9), we can get m eigenvectors $w_1, \cdots, w_m$. Consequently, for any input vector x, its low-dimensional feature representation z can be obtained by

$$z = (w_1, \cdots, w_m)^T(k(x_1, x), \cdots, k(x_N, x))^T.$$

If the kernel fuzzy between-class scatter matrix S_{fb}^{Φ} is replaced by the weighted kernel fuzzy between-class scatter matrix S_{FB}^{Φ}, by means of the algorithms obtained in Section 3, we can get three weighted generalized KFDA methods.

4.1 Weighted generalized KFDA

Similar to Algorithms 3.1 and 3.2, we can derive weighted KFDA/GSVD and weighted PIKFDA algorithms.

Algorithm 4.1. Weighted KFDA/GSVD

1. Compute the EVD of K_{ft} :
$K_{ft} = [\ U_{t1}\ U_{t2}\ ] \begin{bmatrix} \Sigma_{t1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_{t1}^T \\ U_{t2}^T \end{bmatrix};$
2. Compute the weighted between-class scatter matrix K_{FB};
3. Put $\check{K}_{FB} = \Sigma_{t1}^{-1/2}U_{t1}^T K_{FB} U_{t1}\Sigma_{t1}^{-1/2}$ and compute V from the EVD of $\check{K}_{FB}$:
$\check{K}_{FB} = V\Gamma_b^T\Gamma_b V^T$;
4. Assign the first $r-1$ columns of $U_{t1}\Sigma_{t1}^{-1/2}V$ to $\hat{G}_h$;
5. For any input vector x, its low-dimensional feature representation z can thus be given by $z = \hat{G}_h^T(k(x_1, x), \cdots, k(x_N, x))^T$.

Algorithm 4.2. Weighted PIKFDA

1. Compute the EVD of K_{ft} :
$K_{ft} = [\ U_{t1}\ U_{t2}\ ] \begin{bmatrix} \Sigma_{t1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_{t1}^T \\ U_{t2}^T \end{bmatrix};$
2. Compute the weighted between-class scatter matrix K_{FB};
3. Put $\check{K}_{FB} = \Sigma_{t1}^{-1/2}U_{t1}^T K_{FB} U_{t1}\Sigma_{t1}^{-1/2}$ and compute V from the EVD of $\check{K}_{FB}$:
$\check{K}_{FB} = V\Gamma_b^T\Gamma_b V^T$;
4. Assign the first $r-1$ columns of $U_{t1}\Sigma_{t1}^{-1/2}V$ to $\hat{X}_\delta$;
5. $\hat{G} = \hat{X}_\delta M$, where M is any nonsingular matrix;
6. For any input vector x, its low-dimensional feature representation z can thus be given by $z = \hat{G}^T(k(x_1, x), \cdots, k(x_N, x))^T$.

From Algorithms 4.1 and 4.2, we can see that weighted KFDA/GSVD is a special case of weighted PIKFDA with M being the identity matrix.

In addition, we can also obtain weighted RSKFDA.

Algorithm 4.3. Weighted RSKFDA

1. Compute the EVD of K_{fw} :
$K_{fw} = [\ U_{w1}\ U_{w2}\ ] \begin{bmatrix} \Sigma_{w1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_{w1}^T \\ U_{w2}^T \end{bmatrix};$
2. Compute the weighted between-class scatter matrix K_{FB};
3. Put $\tilde{K}_{FB} = \Sigma_{w1}^{-1/2}U_{w1}^T K_{FB} U_{w1}\Sigma_{w1}^{-1/2}$ and compute the EVD of $\tilde{K}_{FB}$:
$$\tilde{K}_{FB} = [\ \tilde{U}_{b1}\ \tilde{U}_{b2}\ ] \begin{bmatrix} \tilde{\Sigma}_{b1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{U}_{b1}^T \\ \tilde{U}_{b2}^T \end{bmatrix};$$
4. Put $\hat{G}_y = U_{w1}\Sigma_{w1}^{-1/2}\tilde{U}_{b1}$;
5. For any input vector x, its low-dimensional feature representation z can thus be given by $z = \hat{G}_y^T(k(x_1, x), \cdots, k(x_N, x))^T$.

4.2 Weighting functions

We can see from [5,7,19] that weighting functions have close relationships with classification accuracy. Different weighting functions can product different classification error for weighted generalized KFDA methods. Selecting suitable weighting function can increase classification accuracy. In this paper, we consider three weighted schemes Algorithms 4.1 - 4.3 with five weighting functions for each weighted scheme. We apply the Euclidean distance $d_{il} = \|\mu_i - \mu_l\|$ between the means of classes i and l in weighting functions $w(d_{il})$. A weighting function is generally a monotonically decreasing function because classes that are closer to one another are likely to have a greater confusion and should be given a greater weightage.

According to the fractional-step LDA procedure in [7], the weighting function should drop faster than the Euclidean distance between the class means for X_i and X_l in $\mathcal{F}$. We first apply two special cases of the weighting function $w(d_{il}) = (d_{il})^{-p}$ proposed by Lotlikar et al. [7] with $p = 1$ and $p = 3$, and then an improved version of weighting function $w(d_{il}) = \frac{1}{2d_{il}^2} erf(\frac{d_{il}}{2\sqrt{2}})$ presented by Loog [5], where the Mahanalobis distance is replaced by the Euclidean distance. In addition, according to the feature of weighting functions mentioned above, we present two new weighting functions. They are listed below:

$$\begin{aligned}
w_1 &: w(d_{il}) = (d_{il})^{-3}, \\
w_2 &: w(d_{il}) = \tfrac{1}{2d_{il}^2} erf(\tfrac{d_{il}}{2\sqrt{2}}), \\
w_3 &: w(d_{il}) = (d_{il})^{-1}, \\
w_4 &: w(d_{il}) = e^{\frac{1}{d_{il}}}, \\
w_5 &: w(d_{il}) = \tfrac{1}{e^{d_{il}}}.
\end{aligned}$$

5 Experiments and analysis

In this section, in order to explain the effective of the proposed methods and illustrate the effect of weighting functions, we conduct a series of experiments with 3 different data sets and 3 kernel functions. The 3 data sets are respectively Dermatology and Irrs databases taken from the UCI Machine Learning Repository [20] and the Yale database [21]. The kernel functions are respectively Gaussian RBF kernel, non-normal Gaussian RBF kernel and polynomial kernel [6]:

$$\begin{aligned}
&k_{RBF}(x, y) = \exp(-\|x - y\|^2/\sigma^2), \sigma \in \mathbf{R}, \\
&k_{NN-RBF}(x, y) = \exp(-\|x-y\|^d/\sigma^2), d \geq 0, d \neq 2, \\
&k_{poly}(x, y) = (r_1 x^T \cdot y - r_2)^d, d \in \mathbf{N}, d \geq 1,
\end{aligned}$$

In our experiments, the KNN classifier with $k = 5$ is used for the classification. For simplicity, the parameter of fuzzy scatter matrices and kernel fuzzy scatter matrices is fixed as $g = 1$. Moreover, we randomly generate 5 matrices for M and compute the misclassification rates by using the optimal transformation matrices produced in PIKFDA for each data set. For each method, we randomly split the data to the training and test set of equal size and repeat it 10 times to obtain mean prediction misclassification rates.

5.1 Experiments on Dermatology database

This data set contains 34 attributes, 33 of which are linear valued and one of them is nominal, it includes six classes: psoriasis, seboreic dermatitis, ichen planus, pityriasis rosea , cronic dermatitis, pityriasis rubra pilaris, there are 366 instances and we choose 358 samples which not include "?" for our experiments.

In our experiment, the Gaussian RBF kernel

$$k(x, y) = \exp(-\|x - y\|^2/10^2),$$

non-normal Gaussian RBF kernel

$$k(x, y) = \exp(-\|x - y\|/10^2)$$

and polynomial kernel

$$k(x, y) = 10^{(-8)} x^T \cdot y + 1$$

are used, respectively. The FKNN parameter k is set as $k = 5$. The parameter of kernel fuzzy scatter matrices is fixed as $h = 0.3$. The average misclassification rates are listed in Table 1, where w_0 indicates no weighting function is introduced.

Table 1 shows that w_4 and w_5 produce good overall results on the Gaussian RBF kernel, they are better than other weighting functions. From Figure 1, we can see expect for a case, the weighting function w_2 and w_3 produce the same results, they produce the classification accuracy 2.5% higher than w_0 for M_4, but they produce the classification accuracy 3.3889% lower than w_0 for M_3. For the non-normal Gaussian RBF, the five weighting functions produce similar overall results, however, they can not improve the performance of the weighted generalized methods as shown in Figure 2. The results of the polynomial kernel are shown in Figure 3, the weighting function w_5 produces the results are near to w_0. The weighting functions w_2 and w_3 produce the same results, the misclassification rates are lower than w_0 for M_2 and M_4. The weighting function w_4 can not be applied on the polynomial kernel. The weighting function w_1 produces the classification accuracy 96.0556% for M_4, the result is 0.9445% higher than w_0. From Table 1, we can see the five weighting functions improve the performance of the weighted generalized methods rather limit, but the 1% difference still makes the weighted KFDA algorithms outperform the KFDA methods.

Table 1: Misclassification rate (%) on data set Dermatology

kernel	$w(d_{ij})$	W-KFDA/GSVD	W-PIKFDA					W-RSKFDA
			M_1	M_2	M_3	M_4	M_5	
RBF	w_0	4.6667	7.0556	6.5000	5.3889	8.3889	6.0000	10.8889
	w_1	4.6111	5.4444	5.6667	7.3333	8.4444	6.2778	12.4444
	w_2	4.8333	5.2222	6.1667	8.7778	5.8889	6.6111	10.3333
	w_3	4.8333	5.2222	6.1667	8.7778	5.8889	6.6111	12.3333
	w_4	4.6111	5.2778	6.1667	7.3889	7.7222	5.9444	11.5556
	w_5	4.8889	5.7222	6.3889	6.6667	6.0000	6.9444	10.8333
NN-RBF	w_0	2.7222	2.6667	2.6667	2.6667	2.7222	2.8333	2.7222
	w_1	2.8889	2.9444	2.8333	2.7222	2.7222	2.8333	2.8333
	w_2	2.8889	3.2222	2.8333	2.7222	2.8333	2.8889	2.6111
	w_3	2.8889	3.2222	2.8333	2.7222	2.8333	2.8889	2.6111
	w_4	2.7778	2.8333	2.7778	2.8333	2.7778	2.7778	3.3333
	w_5	2.8333	2.6111	2.6667	2.7778	2.7778	2.7222	2.7222
Poly	w_0	4.4444	4.3889	5.0556	3.8889	4.8889	3.9444	4.6111
	w_1	4.1667	4.3889	5.3333	4.4444	3.9444	4.3889	4.6111
	w_2	4.5556	4.4444	4.6667	5.3889	4.3333	4.5000	4.6667
	w_3	4.5556	4.4444	4.6667	5.3889	4.3333	4.5000	4.6667
	w_4	-	-	-	-	-	-	-
	w_5	4.4444	4.3889	5.1111	3.8889	4.8889	3.9444	4.5556

Table 2: Misclassification rate (%) on data set Irrs

kernel	$w(d_{ij})$	W-KFDA/GSVD	W-PIKFDA					W-RSKFDA
			M_1	M_2	M_3	M_4	M_5	
RBF	w_0	3.7333	4.1333	4.6667	4.4000	4.5333	4.2667	4.0000
	w_1	4.5333	4.8000	4.5333	4.2667	4.4000	4.2667	3.8667
	w_2	4.6667	4.6667	4.1333	3.8667	4.4000	4.2667	3.4667
	w_3	4.6667	4.6667	4.1333	3.8667	4.4000	4.2667	3.7333
	w_4	-	-	-	-	-	-	-
	w_5	3.7333	4.1333	4.6667	4.4000	4.5333	4.2667	3.7333
NN-RBF	w_0	4.4000	4.4000	4.4000	4.2667	4.2667	4.5333	4.5333
	w_1	4.4000	4.4000	4.1333	4.1333	4.4000	4.2667	4.5333
	w_2	4.4000	4.4000	4.2667	4.1333	4.4000	4.1333	4.5333
	w_3	4.4000	4.4000	4.2667	4.1333	4.4000	4.1333	4.5333
	w_4	-	-	-	-	-	-	-
	w_5	4.4000	4.4000	4.4000	4.2667	4.2667	4.5333	4.5333
Poly	w_0	8.6667	7.4667	8.6667	10.4000	7.6000	9.4667	9.4667
	w_1	9.4667	9.8667	10.0000	8.0000	11.3333	8.4000	8.5333
	w_2	5.7333	8.0000	9.7333	9.7333	9.7333	9.7333	10.9333
	w_3	5.4667	8.1333	8.8000	10.0000	9.7333	9.7333	9.6000
	w_4	6.1333	8.1333	8.8000	9.8667	9.6000	9.7333	8.8000
	w_5	5.6000	8.2667	9.4667	9.4667	9.7333	9.7333	9.2000

Table 3: Misclassification rate (%) on data set Yale

kernel	$w(d_{ij})$	W-KFDA/GSVD	W-PIKFDA					W-RSKFDA
			M_1	M_2	M_3	M_4	M_5	
RBF	w_0	10.3333	11.1111	10.8889	10.2222	11.3333	11.1111	24.3333
	w_1	10.4444	10.6667	11.7778	11.2222	10.4444	10.8889	22.1111
	w_2	10.7778	10.6667	11.3333	10.8889	10.6667	10.5556	25.2222
	w_3	10.8889	10.6667	10.3333	11.4444	10.5556	10.7778	21.3333
	w_4	12.1111	10.3333	11.1111	11.4444	10.6667	10.3333	20.5556
	w_5	10.1111	11.7778	10.4444	9.0000	10.2222	11.0000	23.3333
NN-RBF	w_0	12.6667	11.4444	10.4444	11.2222	11.2222	12.3333	22.7778
	w_1	10.6667	11.2222	12.3333	12.6667	11.7778	11.6667	22.0000
	w_2	11.7778	12.4444	11.7778	11.8889	12.4444	12.7778	24.3333
	w_3	11.7778	12.8889	12.0000	11.7778	12.1111	12.1111	20.1111
	w_4	11.2222	11.1111	11.6667	12.2222	11.3333	10.7778	16.7778
	w_5	11.7778	11.0000	10.7778	10.8889	12.2222	10.8889	23.8889
Poly	w_0	10.3333	9.0000	9.5556	9.6667	10.2222	10.4444	21.1111
	w_1	11.8889	9.6667	11.5556	10.2222	9.8889	9.3333	23.2222
	w_2	9.7778	9.6667	9.5556	10.5556	10.8889	10.6667	23.6667
	w_3	10.2222	11.2222	10.2222	9.0000	10.8889	10.1111	21.1111
	w_4	10.2222	9.2222	9.3333	9.5556	10.5556	9.3333	21.1111
	w_5	24.8889	31.6667	29.6667	34.4444	31.0000	27.2222	17.0000

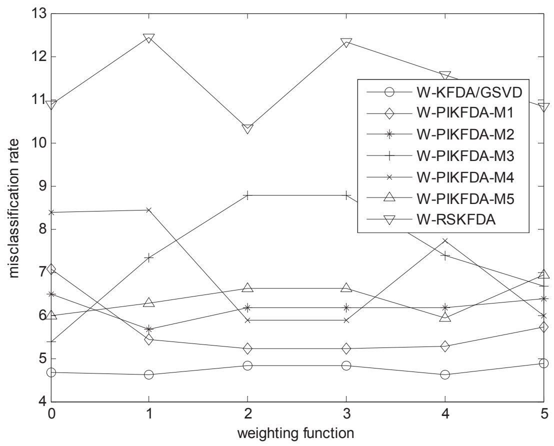

Figure 1: The misclassification rates (%) for the Dermatology and Gaussian RBF kernel

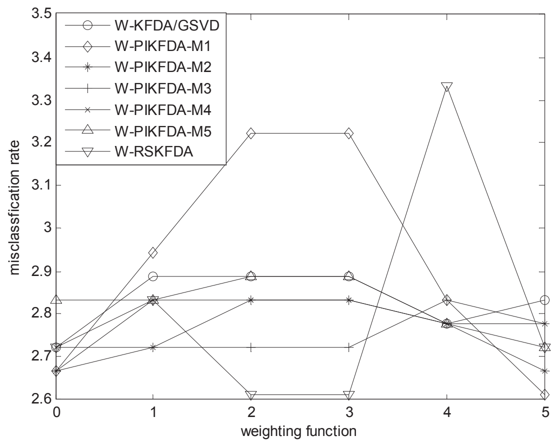

Figure 2: The misclassification rates (%) for the Dermatology and non-normal Gaussian RBF kernel

5.2 Experiments on Irrs database

The data set includes three classes: Setosa, Versicolour and Virginica, each class is in possession of four features: sepal length, sepal width, petal length, petal width, there are 50 instances in each class.

In this experiment, the Gaussian RBF kernel

$$k(x, y) = \exp(-\|x - y\|^2/10^8),$$

non-normal Gaussian RBF kernel

$$k(x, y) = \exp(-\|x - y\|^{0.9}/10^8)$$

and polynomial kernel

$$k(x, y) = x^T \cdot y - 0.51$$

are used, respectively. The FKNN parameter k is set as $k = 3$. The parameter of kernel fuzzy scatter ma-

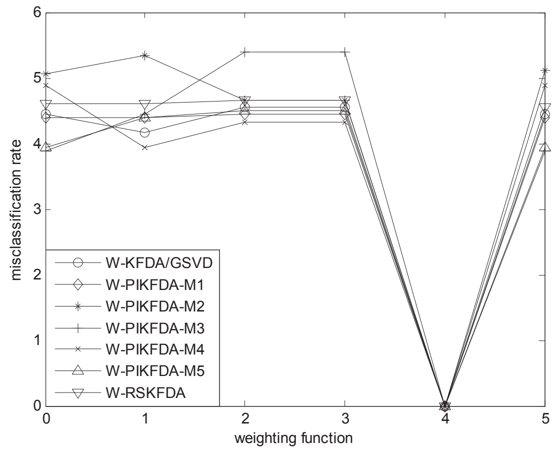

Figure 3: The misclassification rates (%) for the Dermatology and polynomial kernel

trices is fixed as $h = 0.6$. The average misclassification rates are listed in Table 2, where w_0 indicates no weighting function is introduced.

From Table 2, we can see that the weighting functions w_2, w_3 produce similar overall results on the Gaussian RBF kernel, they can improve the performance of the weighted generalized methods but the effects are not obvious, moreover, the weighting function w_4 can not be applied. Expect for a case, the weighting function w_5 produces the results are same to w_0. For the non-normal Gaussian RBF, the weighting function w_4 also can not be applied, the other four weighting functions produce similar overall results, but they can not improve the performance of the weighted generalized methods. The weighting function w_1 produces good overall results on the polynomial kernel, it is better than other weighting functions, however, it produces the classification accuracy 3.7333% lower than w_0 for M_4. The weighting function w_5 produces the best classification accuracy 94.4% for weighted KFDA/GSVD, the result is 3.0667% higher than w_0. Therefore, from Table 2, we find that our weighted dimensional methods can outperform KFDA algorithms.

5.3 Experiments on Yale database

The Yale face database [22] contains 165 face images of 15 individuals. There are 11 images per subject, and these 11 images are respectively under the following different facial expression or configuration: center-light, wearing glasses, happy, left-light, wearing no glasses, normal, right-light, sad, sleepy, surprised, and wink. In our experiment, the images are cropped to a size of 32x32, and the gray level values of all images are rescaled to [0 1].

In this experiment, the Gaussian RBF kernel

$$k(x, y) = \exp(-\|x - y\|^2/10^4),$$

non-normal Gaussian RBF kernel

$$k(x, y) = \exp(-\|x - y\|^{1.9}/10^4)$$

and polynomial kernel

$$k(x, y) = 10x^T \cdot y - 0.51$$

are used, respectively. The FKNN parameter k is set as $k = 5$, moreover, the optimal parameter of kernel fuzzy scatter matrices is fixed as $h = 0.3$. The average misclassification rates are listed in Table 3, where w_0 indicates no weighting function is introduced.

From Table 3, it is interesting to note that the weighting function w_5 produces good overall results on Gaussian RBF kernel, it is better than other weighting functions and it can improve the face recognition accuracy, especially, we get the classification accuracy 1.2222% higher than that of w_0 for M_3. The weighting function of w_4 produces the classification accuracy 3.7777% higher than w_0 for weighted RSKFDA, however, it produces the classification accuracy 1.7778% lower than w_0 for weighted KFDA/GSVD. The weighting function w_4 produces good overall results on the non-normal Gaussian RBF kernel, it is the best one among the five weighting functions, we get the classification accuracy 6% higher than that of w_0 for weighted RSKFDA, w_1 produces the worst classification accuracy 1.8889% lower than w_0 for M_2. For the polynomial kernel, the weighting function w_4 produces good overall results, it can improve the performance of the weighted generalized KFDA methods but the effects are not obvious. Weighting function w_5 is the worst one among the five weighting functions, we get the classification accuracy 24.7777% lower than that of w_0 for M_3, however, it produces the misclassification rate 4.1111% lower than that of w_0 for weighted RSKFDA. Therefore, from Table 3, we can see that the weighted KFDA methods can outperform KFDA algorithms if we choose the appropriate weighting function.

In general, we can see the results that the weighted schemes in weighted generalized KFDA can bring about performance improvement over generalized KFDA.

6 Conclusion

In this paper, based on pseudo-inverse KFDA, KFDA/GSVD and range space KFDA, we propose weighted KFDA/GSVD, weighted pseudo-inverse

KFDA, and weighted range space KFDA. Not only can these methods deal with the singularity problem caused by the undersampled problems, they can also improves the KFDA methods by taking full advantage of the fuzzy membership of the training samples and the different contributions of the class means to the discrimination. In order to explain the effective of the proposed methods, we conduct a series of experiments on 3 different data sets from the UCI Machine Learning Repository and the Yale face database. Results show that different kernel functions and different weighting functions affect the classification accuracy of the proposed methods. Also the misclassification rates produced by each of the weighted dimension reduction methods for three kernel function on the five weighting functions are different. The differences show that not any weighting function may improve the performance of the weighted generalized KFDA. Hence, for the different weighted dimensional methods and the different kernel functions we must choose different weighting functions.

Acknowledgements: This work is supported by National Natural Science Foundation of China (10871226), Natural Science Foundation of Shandong Province (ZR2009AL006) and Young and Middle-Aged Scientists Research Foundation of Shandong Province(BS2010SF004), P.R. China.

References:

[1] Cao, Shen, Sun, Yang, Chen, Feature selection in a kernel space, *In International conference on machine learning (ICML) Oregon*, USA, June 20-24, pp. 121–128.

[2] K. Fukunaga, Introduction to Statistical Pattern Classification, *Academic Press*, San Diego, California, USA, 1990.

[3] J. Yang, Z. Jin, J. Yang, D. Zhang, The essence of kernel Fisher discriminant: KPCA plus LDA, *Pattern Recognition* 37, 2004, pp. 2097–2100.

[4] J. Yang, A. F. Frangi, J. Y. Yang, D. Zhang D, KPCA Plus LDA: a complete kernel fisher discriminant framework for feature extraction and recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 27, 2005, pp. 230–244.

[5] M. Loog, R. P. W. Duin, R. Haeb-Umbach, Multiclass linear dimension reduction by weighted pairwise Fisher criteria, *IEEE Trans. Pattern Anal. Mach. Intell.* 23 (7), 2001, pp. 762–766

[6] R. O. Duda, P. E. Hart and D. Stork, *Pattern Classification*, Wiley, 2000.

[7] R. Lotlikar, R. Kothari, Fractional-step dimensionality reduction, *IEEE Trans. Pattern Analysis and Machine Intelligence* 22 (6), 2000, pp. 623–627

[8] Z.-P. Chen, J.-H. Jiang, Y. Li, Y.-Z. Liang, R.-Q. Yu, Fuzzy linear discriminant analysis for chemical data sets, *Chemometrics and Intelligent Laboratory Systems* 45, 1999, pp. 295–302.

[9] W. Yang, H. Yan, J. Wang, J. Yang, Face recognition using complete fuzzy LDA, *Proceedings of the 2008 International Conference on Pattern Recognition*, 2008, pp. 1–4.

[10] Song, Yang, Yang, Wu, Zheng, Discriminant analysis approach using fuzzy fourfold subspaces model, *Neurocomputing* 73, 2010, pp. 2255–2265.

[11] P. Howland, M. Jeon, H. Park, Structure preserving dimension reduction for clustered text data based on the generalized singular value decomposition, *SIAM J. Matrix Anal. Appl.* 25 (1), 2003, pp. 165–179.

[12] C. H. Park, H. Park, A comparision of generalized linear discriminant analysis algorithms, *Pattern Recognition* 41, 2008, pp. 1083–1097.

[13] C. H. Park, H. Park, Nonlinear discriminant analysis using kernel functions and the generalized singular value decomposition, *SIAM J. Matrix Anal. Appl.* 27 (1), 2005, pp. 87–102.

[14] Ye, R. Janardan, C. H. Park and H. Park, An optimization criterion for generalized discriminant analysis on undersampled problems, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26 (8), 2004b, pp. 982–994.

[15] Y. X. Li, Y. Q. Gao, H. Erdogan, Weighted pairwise scatter to improve linear discriminant analysis, *Proceedings of the sixth International Conference on Spoken Language Processing*, 2000.

[16] G. Dai, Y. T. Qian, S. Jia, A kernel fractional-step nonlinear discriminant analysis for pattern recognition, *Proceedings of the 18th International Conference on Pattern Recognition* 2, August 2004, pp. 431–434.

[17] G. Dai, D. Y. Yeung, Nonlinear dimensionality reduction for classification using kernel weighted subspace method, *Proceedings of the IEEE International Conference on Image Processing*, September 2005, pp. 838–841.

[18] G. Dai, D. Y. Yeung, Y. T. Qian, Face recognition using a kernel fractional-step discriminant analysis algorithm, *Pattern recognition* 40, 2007, pp. 229–243.

[19] Y. Liang et al. Uncorrelated linear discriminant analysis based on weighted pairwise Fisher criterion, *Pattern Recognition* 40, 2007, pp. 3606–3615

[20] A. Asuncion, D. Newman. *UCI machine learning repository* <http://www.ics. uci.edu/ mlearn/MLRepository.html>, School of Information and Computer Science, University of California, Irvine, 2007.

[21] L. S. Qiao, S. C. Chen, X. Y. Tan, Sparsity preserving projections with applications to face Recognition, *Pattern Recognition* 43 (1), 2010, pp. 331–341.

[22] P. Belhumeur, J. Hepanha and D. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class specific linear projection, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (7), 1997, pp. 711–720.