Neurocomputing 277 (2018) 101–107

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Spectral regression based marginal Fisher analysis dimensionality reduction algorithm

Bing Liu[a,b], Yong Zhou[a,*], Zhan-guo Xia[a,*], Peng Liu[c,*], Qiu-yan Yan[a], Hui Xu[a,*]

[a] School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, Jiangsu, China
[b] Insititute of Electrics, Chinese Academy of Sciences, Beijing 100190, China
[c] Internet of Things Perception Mine Research Center, China, University of Mining and Technology, Xuzhou 221008, Jiangsu, China

ARTICLE INFO

Article history:
Received 30 September 2016
Revised 4 May 2017
Accepted 9 May 2017
Available online 24 August 2017

Keywords:
Marginal Fisher analysis
Spectral regression
Extreme learning machine (ELM)
Dimensionality reduction

ABSTRACT

Traditional nonlinear dimensionality reduction methods, such as multiple kernel dimensionality reduction and nonlinear spectral regression (SR), are generally regarded as extended versions of linear discriminant analysis (LDA) in the supervised case. As is well known, LDA has the restrictive assumption that the data of each class is of a Gaussian distribution. Thus, the performance of these methods will be degraded if such an assumption is not hold. Although some methods based on marginal Fisher analysis are proposed to overcome the drawback of LDA, they have to solve the problem of dense metrics generalized eigenvalue decomposition, which is very time-consuming. To address these issues, in this paper, marginal Fisher analysis criterion based on extreme learning machine (ELM) is proposed to improve spectral regression and kernel marginal Fisher analysis. It is proved that the proposed marginal Fisher analysis is a special case of traditional kernel marginal Fisher analysis. Based on the proposed criterion, a novel supervised dimensionality reduction algorithm is presented by virtue of ELM and spectral regression. Experimental results on benchmark datasets validate that the proposed algorithm outperforms the state-of-the-art nonlinear dimensionality reduction methods in supervised scenarios.

1. Introduction

Recently, nonlinear dimensionality reduction is an active research subject in machine learning and pattern recognition [1–3]. A family of multiple kernel dimensionality reduction methods, such as MKL-DR [4], MKL-TR [5], MKL-SR [6] and MKL-SRTR [7], has been proposed to automatically construct new kernels using existing base kernels instead of using only one specific kernel. Since these methods can be unified under the graph embedding framework and be regarded as multiple kernel versions of linear discriminant analysis (LDA), which has the assumption that the distribution of each class is considered to be a unimodal Gaussian. This property often does not exist in real-world applications and separability of the different classes cannot be well characterized by interclass scatter.

Marginal Fisher analysis (MFA) effectively overcomes the limitations of the traditional linear discriminant analysis algorithm due to data distribution assumptions and available projection directions. Kernel marginal Fisher analysis (KMFA), which can be regarded as an extension of MFA [8], has been proposed for supervised nonlinear dimensionality reduction. KMFA is formulated as a ratio-trace optimization problem based on marginal Fisher criterion, which not only solves the problem of the restrictive assumption by using an intrinsic graph and another penalty graph, but can better characterize the separability of different classes than the interclass variance in linear discriminant analysis (LDA). But, KMFA has to solve the problem of dense metrics generalized eigenvalue decomposition (GEVD), which is very time-consuming. In addition, it is difficult to specify the optimal parameters of kernel functions in real applications. Extreme spectral regression (ESR) algorithm has been presented to reduce human interventions, which incorporates ELM into spectral regression to speed up its learning speed [9]. But, this method is still based on graph embedding. In the supervised case, it is equivalent to LDA and has to obey the data distribution hypothesis.

To address these issues, we take advantage of MFA, SR and ELM to construct a novel supervised dimensionality reduction model, which not only effectively overcomes the limitations of LDA, but has fast learning speed. First, Marginal Fisher Criterion based on extreme learning machine (ELM) is proposed to speed up kernel marginal Fisher analysis. It is proved that the improved marginal

* Corresponding authors.
E-mail addresses: yzhou@cumt.edu.cn (Y. Zhou), xiazg@cumt.edu.cn (Z.-g. Xia), liupeng@cumt.edu.cn (P. Liu), xuhui@cumt.edu.cn (H. Xu).

http://dx.doi.org/10.1016/j.neucom.2017.05.097
0925-2312/

Table 1
Notations.

Notations	Descriptions
$\mathbb{R}^d$	the input d-dimensional Euclidean space
n	the number of total training data points
c	the number of classes that the samples belong to
$\boldsymbol{X}$	$\boldsymbol{X}=[\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n]\in\mathbb{R}^{d\times n}$ is the training data set
$k(\boldsymbol{x}_i,\boldsymbol{x}_j)$	Kernel function of variables $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$
$\boldsymbol{K}$	Kernel matrix $\boldsymbol{K}=\{k(\boldsymbol{x}_i,\boldsymbol{x}_j)\}\in\mathbb{R}^{n\times n}$
$\|\ \|$	norm in the Hilbert space $\mathcal{H}$
$\boldsymbol{H}$	the hidden-layer output matrix
$\boldsymbol{\beta}$	the vector of the output weights between the hidden layer of nodes and the output node
$\boldsymbol{L}$	$\boldsymbol{L}$ is the graph Laplacian matrix of the intraclass compactness graph $\boldsymbol{W}$
$\boldsymbol{L^p}$	$\boldsymbol{L^p}$ is the Laplacian matrix of the interclass separability graph $\boldsymbol{W^p}$

Fisher analysis is a special case of kernel marginal Fisher analysis. Second, a novel supervised dimensionality reduction algorithm is presented by virtue of spectral regression based on this criterion. Finally, experimental results on benchmark datasets validate that the proposed algorithm outperforms the state-of-the-art supervised dimensionality reduction methods.

The paper is structured as follows. In Section 2, we briefly introduce the related work. The proposed model and algorithm are introduced in Section 3. In Section 4, the experimental results are presented and validate the effectiveness of the proposed method. Finally, we give the related conclusions in Section 5. In order to avoid confusion, we give a list of the main notations used in this paper in Table 1.

2. Related work

2.1. Extreme learning machine

The output function of ELM for generalized SLFNs in the case of one output node is [10–15]

$$f_L(\boldsymbol{x})=\sum_{i=1}^{L}\beta_i h_i(\boldsymbol{x})=\boldsymbol{h}(\boldsymbol{x})\boldsymbol{\beta}, \tag{1}$$

where $\boldsymbol{\beta}=[\beta_1,\ \ldots\ ,\beta_L]^T$ is the vector of the output weights between the hidden layer of L nodes and the output node, and $\boldsymbol{h}(\boldsymbol{x})=[h_1(\boldsymbol{x}),\ \ldots,h_L(\boldsymbol{x})]$ is the output (row) vector of the hidden layer with respect to the input $\boldsymbol{x}$. $\boldsymbol{H}$ is the hidden-layer output matrix denoted by

$$\boldsymbol{H}=\begin{bmatrix}h(\boldsymbol{x}_1)\\ h(\boldsymbol{x}_2)\\ \vdots\\ h(\boldsymbol{x}_n)\end{bmatrix}=\begin{bmatrix}h_1(\boldsymbol{x}_1)&\ldots&h_L(\boldsymbol{x}_1)\\ h_1(\boldsymbol{x}_2)&\ldots&h_L(\boldsymbol{x}_2)\\ &\vdots\vdots\vdots&\\ h_1(\boldsymbol{x}_n)&\ldots&h_L(\boldsymbol{x}_n)\end{bmatrix}. \tag{2}$$

For completeness, we briefly introduce the multiclass classifiers of ELM.

(1) *Multiclass classifier with single output:* ELM can approximate any target continuous functions and the output of the ELM classifier $\boldsymbol{h}(\boldsymbol{x})\boldsymbol{\beta}$ can be as close to the class labels in the corresponding regions as possible. Thus the classification problem for ELM with a single-output node can be formulated as [10–15]:

$$\text{Minimize}:\ L_{\text{ELM}}=\frac{1}{2}\|\boldsymbol{\beta}\|^2+\frac{C}{2}\sum_{i=1}^{n}\varepsilon_i^2 \tag{3}$$

$$\text{Subject to}:\ \boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\beta}=t_i-\varepsilon_i,\ \ i=1,\ldots,n$$

For the binary classification case, ELM only has one output node and the decision function of ELM classifier is

$$f(\boldsymbol{x})=\boldsymbol{h}(\boldsymbol{x})\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{H}^T\boldsymbol{H}\right)^{-1}\boldsymbol{H}^T\boldsymbol{T} \tag{4}$$

where $\boldsymbol{T}=\begin{bmatrix}t_{11}&\ldots&t_{1m}\\ t_{21}&\ldots&t_{2m}\\ &\vdots\vdots\vdots&\\ t_{n1}&\ldots&t_{nm}\end{bmatrix}$. Generally, Eq. (10) is applied to large-scale data sets or moderate data sets. The decision function applied to small-scale training samples is

$$f(\boldsymbol{x})=\boldsymbol{h}(\boldsymbol{x})\boldsymbol{H}^T\left(\frac{\boldsymbol{I}}{C}+\boldsymbol{H}\boldsymbol{H}^T\right)^{-1}\boldsymbol{T} \tag{5}$$

(2) *Multiclass classifier with multioutputs:* If ELM has multioutput nodes, an m-class classifier is corresponding to m output nodes. If the original class label is l, the expected output vector of the m output nodes is $\boldsymbol{t_i}=[\overbrace{0,\ldots,0}^{\boldsymbol{l}},1,0,\ldots,0]^{\boldsymbol{T}}$. That is, the lth element of $\boldsymbol{t_i}=[t_{i,1},\ldots,t_{i,m}]^{\boldsymbol{T}}$ is one and the rest of the elements are zero. The classification problem for ELM with multioutput nodes is [10–15]:

$$\text{Minimize}:\ L_{\text{ELM}}=\frac{1}{2}\|\boldsymbol{\beta}\|^2+\frac{C}{2}\sum_{i=1}^{n}\|\varepsilon_i\|^2$$

$$\text{Subject to}:\ \boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\beta}=\boldsymbol{t}_i^T-\varepsilon_i^T,\ \ i=1,\ldots,n. \tag{6}$$

where $\varepsilon_i=[\varepsilon_{i1},\ldots,\varepsilon_{im}]^{\boldsymbol{T}}$ is the training error vector of the m output nodes with respect to the training sample $\boldsymbol{x_i}$. In this case, the predicted class label of sample $\boldsymbol{x}$ is

$$\text{label}(\boldsymbol{x})=\text{argmax}_{i\in\{1,2,\ldots,m\}}\{f_i(\boldsymbol{x})\}. \tag{7}$$

where $f(\boldsymbol{x})=[f_1(\boldsymbol{x}),\ldots,f_m(\boldsymbol{x})]^T$.

2.2. Spectral regression algorithm

The SR algorithm transforms the optimization problem of dimensionality reduction into a regression framework, which avoids eigen-decomposition of dense matrices. In addition, it takes advantage of regularization terms to improve the performance of dimensionality reduction. Given a training set with l labeled samples $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, ..., $\boldsymbol{x}_l$ and u unlabeled samples $\boldsymbol{x}_{l+1}$, $\boldsymbol{x}_{l+2}$, ..., $\boldsymbol{x}_{l+u}$, where the sample $\boldsymbol{x}_i\in\boldsymbol{R}^d$ belongs to one of c classes. The SR algorithm is summarized as follows [16]:

Step 1: Constructing the weight matrix $\boldsymbol{W}$ using binary weights or heat kernel weights w_{ij}.

Step 2: Let $\boldsymbol{D}$ be the $n\times n$ diagonal matrix, where $\boldsymbol{D}_{ii}=\sum_{j=1}^{l+u}\boldsymbol{w}_{ij}$. Find the largest c generalized eigenvectors y_0, y_1, ..., y_{c-1} of the following eigen-problem:

$$\boldsymbol{W}\boldsymbol{y}=\lambda\boldsymbol{D}\boldsymbol{y}, \tag{8}$$

Step 3: Choosing a linear projection $y_i=f(\boldsymbol{x}_i)=\boldsymbol{a}^T\boldsymbol{x}_i$, Eq. (6) can be rewritten as:

$$XWX^T\boldsymbol{a}=\lambda XDX^T\boldsymbol{a}. \tag{9}$$

Alternatively, if $y_i = f(\boldsymbol{x}_i) = \sum_{j=1}^{n} \alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, Eq. (6) can be rewritten as:

$$\boldsymbol{KWK\alpha} = \lambda \boldsymbol{KDK\alpha}. \tag{10}$$

where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]^T$. The optimal $\boldsymbol{\alpha}$'s are the eigenvectors corresponding to the maximum eigenvalue of the eigen-problem (10).

Step 4: Calculating the transform matrix $\boldsymbol{A}$, for the linear projection $y_i = f(\boldsymbol{x}_i) = \boldsymbol{a}^T \boldsymbol{x}_i$, $\boldsymbol{A} = [\boldsymbol{a}_1 \ldots, \boldsymbol{a}_{c-1}] \in \mathbb{R}^{d \times c-1}$. $\boldsymbol{a}_k$ $(k = 1, \ldots, c-1)$ is the solution of regularized least square problem:

$$\boldsymbol{a}_k = \arg\min_{\boldsymbol{a}} \left(\sum_{i=1}^{n} \left(\boldsymbol{a}^T \boldsymbol{x}_i - y_i^k \right)^2 + \gamma \|\boldsymbol{a}\|^2 \right), \tag{11}$$

where y_i^k is the ith element of $\boldsymbol{y}^k$. For the nonlinear projection $y_i = f(\boldsymbol{x}_i) = \sum_{j=1}^{n} \alpha_j k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, Eq. (11) can be transformed into

$$\min_{\boldsymbol{\alpha}_k} \sum_{i=1}^{n} \left(f(\boldsymbol{x}_i) - y_i^k \right)^2 + \alpha \|f\|_K^2, \tag{12}$$

where $\boldsymbol{\alpha}_k$ $(k = 1, \ldots, c-1)$ is the solution of the following linear equations system:

$$(\boldsymbol{K} + \alpha I)\boldsymbol{\alpha}_k = \boldsymbol{y}_k, \tag{13}$$

where $\boldsymbol{K}$ is $n \times n$ gram matrix and $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.

3. Marginal Fisher analysis dimensionality reduction via SR and ELM

3.1. Marginal Fisher analysis

Marginal Fisher analysis (MFA) aims to overcome the limitations of LDA, which designs new criterion that characterizes the intra-class compactness and the inter-class separability.

Given the input data point $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $\boldsymbol{y}_i$ is the class label of $\boldsymbol{x}_i$. Denote $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d \times n}$ as the training data matrix. Constructing the intraclass compactness graph $\boldsymbol{W}$ and interclass separability graph $\boldsymbol{W}^p$, where

$$W_{ij} = \begin{cases} 1, & if\ i \in N_{k_1}^+(j)\ or\ j \in N_{k_1}^+(i) \\ 0, & else. \end{cases} \tag{14}$$

Here, $N_{k_1}^+(i)$ indicates the index set of the k_1 nearest neighbors of the sample $\boldsymbol{x}_i$ in the same class.

$$W_{ij}{}^p = \begin{cases} 1, & if\ (i,j) \in P_{k_2}(c_i)\ \ or\ \ (i,j) \in P_{k_2}(c_j) \\ 0, & else. \end{cases} \tag{15}$$

Here, $P_{k_2}(c)$ is a set of data pairs that are the k_2 nearest pairs among the set $\{(i,j),\ i \in \pi_c,\ j \notin \pi_c\}$.

Marginal Fisher Criterion can be expressed as follows:

$$\begin{aligned} \boldsymbol{y}^* &= \operatorname{argmin} \frac{\sum_i \sum_{i \in N_{k_1}^+(j) \text{ or } j \in N_{k_1}^+(i)} \left\| y_i - y_j \right\|^2 W_{ij}}{\sum_i \sum_{(i,j) \in P_{k_2}(c_i) \text{ or } (i,j) \in P_{k_2}(c_j)} \left\| y_i - y_j \right\|^2 W_{ij}{}^p} \\ &= \operatorname{argmin} \frac{\boldsymbol{y}^T (\boldsymbol{D} - \boldsymbol{W}) \boldsymbol{y}}{\boldsymbol{y}^T (\boldsymbol{D}^p - \boldsymbol{W}^p) \boldsymbol{y}} \end{aligned} \tag{16}$$

where $\boldsymbol{y} = [y_1, y_2, \ldots, y_n]^T$ and y_i denotes the low-dimensional representation of original sample x_i in high-dimensional space. The optimal $\boldsymbol{y}$'s in Eq. (16) can be obtained by solving the generalized eigenvalue problem:

$$\boldsymbol{Ly} = \lambda \boldsymbol{L}^p \boldsymbol{y} \tag{17}$$

where $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ and $\boldsymbol{L}^p = \boldsymbol{D}^p - \boldsymbol{W}^p$ are the Laplacian matrices of the intraclass compactness graph $\boldsymbol{W}$ and interclass separability graph $\boldsymbol{W}^p$, respectively.

3.2. ELM-based MFA

If a linear projection $\boldsymbol{y} = \boldsymbol{w}^T \boldsymbol{x}$ is introduced, Eq. (16) can be rewritten as:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \frac{\boldsymbol{w}^T \boldsymbol{XLX}^T \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{XL}^p \boldsymbol{X}^T \boldsymbol{w}} \tag{18}$$

which is referred to as marginal Fisher analysis (MFA) and can be transformed into the following generalized eigenvalue problem:

$$\boldsymbol{XLX}^T \boldsymbol{w} = \boldsymbol{XL}^p \boldsymbol{X}^T \boldsymbol{w} \tag{19}$$

Assume that a nonlinear projection $\boldsymbol{y} = \boldsymbol{K\alpha}$ is used, where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]^T$, Eq. (16) can be rewritten as:

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \frac{\boldsymbol{\alpha}^T \boldsymbol{KLK\alpha}}{\boldsymbol{\alpha}^T \boldsymbol{KL}^p \boldsymbol{K\alpha}} \tag{20}$$

which is referred to as kernel marginal Fisher analysis and can be further transformed into the following generalized eigenvalue problem:

$$\boldsymbol{KLK\alpha} = \boldsymbol{KL}^p \boldsymbol{K\alpha} \tag{21}$$

We extend the marginal Fisher analysis based on ELM (termed as EMFA) in the following way:

First, intraclass compactness is characterized from the intrinsic graph by the term

$$\begin{aligned} S_c &= \sum_i \sum_{i \in N_{k_1}^+(j) \text{or } j \in N_{k_1}^+(i)} \left\| \boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\beta} - \boldsymbol{h}(\boldsymbol{x}_j)\boldsymbol{\beta} \right\|^2 W_{ij} \\ &= 2\boldsymbol{\beta}^T \boldsymbol{H}^T (\boldsymbol{D} - \boldsymbol{W}) \boldsymbol{H\beta}, \end{aligned} \tag{22}$$

where, $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_L]^T \in \mathbb{R}^L$ and $\boldsymbol{D} = \operatorname{diag}(\boldsymbol{D}_1, \ldots, \boldsymbol{D}_n)$ is a diagonal matrix with the diagonal elements defined as $\boldsymbol{D}_i = \sum_{j=1}^{n} \boldsymbol{w}_{ij}$.

Second, interclass separability is characterized by a penalty graph with the term

$$\begin{aligned} S_p &= \sum_i \sum_{(i,j) \in P_{k_2}(c_i) \text{ or } (i,j) \in P_{k_2}(c_j)} \left\| \boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\beta} - \boldsymbol{h}(\boldsymbol{x}_j)\boldsymbol{\beta} \right\|^2 W_{ij}{}^p \\ &= 2\boldsymbol{\beta}^T \boldsymbol{H}^T (\boldsymbol{D}^p - \boldsymbol{W}^p) \boldsymbol{H\beta}, \end{aligned} \tag{23}$$

where $\boldsymbol{D}^p$ is the diagonal matrix of $\boldsymbol{W}^p$.

Finally, Marginal Fisher Criterion based on ELM can be denoted as follows:

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} \frac{\boldsymbol{\beta}^T \boldsymbol{H}^T (\boldsymbol{D} - \boldsymbol{W}) \boldsymbol{H\beta}}{\boldsymbol{\beta}^T \boldsymbol{H}^T (\boldsymbol{D}^p - \boldsymbol{W}^p) \boldsymbol{H\beta}} \tag{24}$$

The optimal $\boldsymbol{\beta}$'s are the eigenvectors corresponding to the minimum eigenvalue of the eigen-problem:

$$\boldsymbol{H}^T (\boldsymbol{D} - \boldsymbol{W}) \boldsymbol{H\beta} = \lambda \boldsymbol{H}^T (\boldsymbol{D}^p - \boldsymbol{W}^p) \boldsymbol{H\beta}. \tag{25}$$

In order to avoid eigen-decomposition of dense matrices in Eq. (25), we first present the following theorem:

Theorem 1. *Let $\boldsymbol{y}$ be the eigenvector of the eigen-problem in Eq. (17) with eigenvalue λ. If $\boldsymbol{y} = \boldsymbol{H\beta}$, then $\boldsymbol{\beta}$ is the eigenvector of the eigen-problem in Eq. (25) with the same eigenvalue λ. If $\boldsymbol{y} = \boldsymbol{w}^T \boldsymbol{x}$, then $\boldsymbol{w}$ is the eigenvector of the eigen-problem in Eq. (19) with the same eigenvalue λ. If $\boldsymbol{y} = \boldsymbol{K\alpha}$, then $\boldsymbol{\alpha}$ is the eigenvector of the eigen-problem in Eq. (21) with the same eigenvalue λ.*

Proof. At the left side of Eq. (17), replacing $\boldsymbol{y}$ by $\boldsymbol{H\beta}$, we have

$$\boldsymbol{H}^T (\boldsymbol{D} - \boldsymbol{W}) \boldsymbol{H\beta} = \boldsymbol{H}^T (\boldsymbol{D} - \boldsymbol{W}) \boldsymbol{y} = \lambda \boldsymbol{H}^T (\boldsymbol{D}^p - \boldsymbol{W}^p) \boldsymbol{y} = \lambda \boldsymbol{H}^T (\boldsymbol{D}^p - \boldsymbol{W}^p) \boldsymbol{H\beta}$$

Thus, $\boldsymbol{\beta}$ is the eigenvector of the eigen-problem Eq. (25) with the same eigenvalue λ. If $\boldsymbol{y} = \boldsymbol{w}^T \boldsymbol{x}$ or $\boldsymbol{y} = \boldsymbol{K\alpha}$, the conclusions of Theorem 1 can be proved using the same method. □

According to Theorem 1, the transformation matrices can be obtained through two steps:

Table 2
The MFA and KMFA algorithms using spectral regression.

Input: Labeled samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$, the number of classes c, the number of the nearest k_1 and k_2, parameters γ.
Output: Transformation matrix and embedding results.

Step 1: Constructing the intraclass compactness graph $\boldsymbol{W}$ and interclass separability graph $\boldsymbol{W}^p$.
Step 2: Calculate Laplacian matrices $(\boldsymbol{D}^p - \boldsymbol{W}^p)$ and $(\boldsymbol{D} - \boldsymbol{W})$.
Step 3: Find the smallest c generalized eigenvectors y_0, y_1, ..., y_{c-1} of the eigen-problem (17).
Step 4: For KMFA, find $\boldsymbol{\alpha}_k (k = 1 \dots c)$ by solving the least square problem (27). For MFA, find $\boldsymbol{w}_k (k = 1 \dots c)$ by solving the least square problem (28).
Step 5: Output the transformation matrix $\Theta = [\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_c]$ for KMFA and $\Theta = [\boldsymbol{w}_1, \dots, \boldsymbol{w}_c]$ for MFA.
Step 6: The unseen samples can be embedded into c dimensional subspace by $\Theta^T \boldsymbol{x}$ for MFA and $\Theta^T \boldsymbol{K}$ **for** KMFA.

Table 3
The proposed EMFA algorithm.

Input: Labeled samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$, the number of classes c, the number of the nearest k_1 and k_2, parameters γ and L.
Output: Transformation matrix $\Theta = [\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_c]$ and embedding results.

Step 7: Constructing the intraclass compactness graph $\boldsymbol{W}$ and interclass separability graph $\boldsymbol{W}^p$.
Step 8: Calculate Laplacian matrices $(\boldsymbol{D}^p - \boldsymbol{W}^p)$ and $(\boldsymbol{D} - \boldsymbol{W})$.
Step 9: Find the smallest c generalized eigenvectors y_0, y_1, ..., y_{c-1} of the eigen-problem (17).
Step 10: Randomly generate $\{(a_i, b_i)\}_{i=1}^L$ from any continuous probability distribution.
Step 11: Find $\boldsymbol{\beta}_k (k = 1 \dots c)$ by solving the least square problem (26).
Step 12: Output the transformation matrix $\Theta = [\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_c]$.
Step 13: The unseen samples can be embedded into c dimensional subspace by $\Theta^T \boldsymbol{h}(\boldsymbol{x})$, where $\boldsymbol{h}(\boldsymbol{x}) = [\, h_1(\boldsymbol{x}), \ \dots, h_L(\boldsymbol{x})]$.

1. Since $(\boldsymbol{D}^p - \boldsymbol{W}^p)$ and $(\boldsymbol{D} - \boldsymbol{W})$ are all sparse matrices, it is easy to solve the eigen-problem in Eq. (17) to get $\boldsymbol{y}$. Generally,
2. For EMFA, calculate c vectors $\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_c \in \mathbb{R}^L$. $\boldsymbol{\beta}_k (k = 1, \dots, c)$ is the solution of regularized least square problem:

$$\boldsymbol{\beta}_k = \int \arg\min_{\boldsymbol{\beta}} \left(\sum_{i=1}^{n} \left(\boldsymbol{h}(\boldsymbol{x}_i)\boldsymbol{\beta} - y_i^k \right)^2 + \gamma \left\| \boldsymbol{\beta} \right\|^2 \right), \tag{26}$$

where y_i^k is the ith element of $\boldsymbol{y}^k$.

For KMFA, Calculate c vectors $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_c \in \mathbb{R}^n$. $\boldsymbol{\alpha}_k (k = 1, \dots, c)$ is the solution of regularized least square problem:

$$\boldsymbol{\alpha}_k = \arg\min_{\boldsymbol{\alpha}} \left(\sum_{i=1}^{n} \left(\boldsymbol{f}(\boldsymbol{x}_i) - y_i^k \right)^2 + \gamma \left\| \boldsymbol{f} \right\|^2 \right), \tag{27}$$

where $\boldsymbol{f}(\boldsymbol{x}) = \sum_{j=1}^n \alpha_j k(\boldsymbol{x}, \boldsymbol{x}_j)$ according to the classical representer theorem [17].

For MFA, calculate c vectors $\boldsymbol{w}_1, \dots, \boldsymbol{w}_c \in \mathbb{R}^d$. $\boldsymbol{w}_k (k = 1, \dots, c)$ is the solution of regularized least square problem:

$$\boldsymbol{w}_k = \arg\min_{\boldsymbol{w}} \left(\sum_{i=1}^{n} \left(\boldsymbol{w}^T \boldsymbol{x}_i - y_i^k \right)^2 + \gamma \left\| \boldsymbol{w} \right\|^2 \right), \tag{28}$$

where y_i^k is the ith element of $\boldsymbol{y}^k$.

Theorem 2 demonstrates the relationship between EMFA and KMFA.

Theorem 2. *Assume* $\boldsymbol{H} = [\boldsymbol{h}(\boldsymbol{x}_1), \boldsymbol{h}(\boldsymbol{x}_2), \dots, \boldsymbol{h}(\boldsymbol{x}_n)]^T$, *where* $\boldsymbol{h}(\boldsymbol{x})$ *is* $\frac{1}{\sqrt{L}}(k_{b_1}(\boldsymbol{x}, \boldsymbol{a}_1), k_{b_2}(\boldsymbol{x}, \boldsymbol{a}_2), \dots, k_{b_L}(\boldsymbol{x}, \boldsymbol{a}_L))$, $\{(\boldsymbol{a}_i, b_i)\}_{i=1}^L$ *are randomly generated from any continuous probability distribution and L is the number of hidden nodes in ELM. Then EMFA is a special case of KMFA.*

Proof. For KMFA, $\boldsymbol{\alpha}$ can be obtained by the following equation:

$$\boldsymbol{\alpha}^{\mathrm{T}} = (\boldsymbol{K} + \gamma \mathbf{I})^{-1} \boldsymbol{y}^{\mathrm{T}} \tag{29}$$

If $\boldsymbol{K} = \boldsymbol{H}\boldsymbol{H}^T$, where $\boldsymbol{H} = [\boldsymbol{h}(\boldsymbol{x}_1), \boldsymbol{h}(\boldsymbol{x}_2), \dots, \boldsymbol{h}(\boldsymbol{x}_n)]^T$ and $\boldsymbol{h}(\boldsymbol{x}) = \frac{1}{\sqrt{L}}(k_{b_1}(\boldsymbol{x}, \boldsymbol{a}_1), k_{b_2}(\boldsymbol{x}, \boldsymbol{a}_2), \dots, k_{b_L}(\boldsymbol{x}, \boldsymbol{a}_L))$, the embedding function of KMFA can be derived as follows:

$$\boldsymbol{f}^*(\boldsymbol{x})^{\mathrm{T}} = \boldsymbol{K}(\boldsymbol{K} + \gamma \mathbf{I})^{-1} \boldsymbol{y} = h(\boldsymbol{x}) \boldsymbol{H}^T \left(\boldsymbol{H}\boldsymbol{H}^T + \gamma \mathbf{I} \right)^{-1} \boldsymbol{y} = h(\boldsymbol{x}) \boldsymbol{\beta},$$

which is exactly the embedding function of EMFA. This completes the proof of Theorem 2. □

The proposed algorithms are described in Tables 2 and 3, respectively. In the supervised case, constructing the graph Laplacian matrix takes $O(n \log n)$ time by using the cover tree structure. For EMFA, the computational complexity of the regularized least squares problem is $O(\min\{n, L\}^3)$. Thus, the total computational complexity of EMFA is $O(\min\{n, L\}^3 + n \log n)$. Correspondingly, the total computational complexity of MFA and KMFA is $O(\min\{n, d\}^3 + n \log n)$ and $O(n^3 + n \log n)$, respectively.

Table 4
Description of benchmark datasets.

Datasets	Dimensions	# of samples	# of classes
Ionosphere	33	351	2
Sonar	60	208	2
USPS	256	3000	10
Isolet	617	900	3
MINIST	784	600	3
Extended Yale B	1024	1900	38
PIE	1024	10,200	68
COIL-20	1024	1440	20

4. Experiments

We carry out all algorithms on UCI datasets (Sonar, Ionosphere and Isolet), face recognition datasets (Extended Yale, PIE), digits recognition datasets (USPS and MNIST) and object recognition datasets (COIL-20). The basic information of datasets is shown in Table 4. For all datasets, we first normalize the values of elements to the range [0, 1]. All the experiments have been performed in MATLAB R2013a running in a 3.10 GHZ Intel Core i5-2400 with 8-GB RAM.

In this experiment, for fair comparison, we mainly compared the proposed methods with the following approaches based on spectral regression: ESR [9], MKL-SR [6] and MKL-SRTR [7] in supervised settings. The maximum number of iterations for all multiple kernel methods is initialized as 10 and 10 RBF base kernels are used and their σ values are set as 0.10, 0.20, 0.40, 0.80, 1.60, 3.20, 6.40, 12.80, 25.60 and 51.20 respectively. For MKL-SR and MKL-SRTR, the affinity matrix $\boldsymbol{W} = [w_{ij}]$ is defined as

$$w_{ij} = \begin{cases} 1/n_{y_i}, & if\ y_i = y_j, \\ 0, & otherwise. \end{cases} \tag{30}$$

For MKL-SR and MKL-SRTR, another affinity matrix $\boldsymbol{W}' = [w'_{ij}]$, where $w'_{ij} = 1/N$. The final reduced dimension is c for all methods.

Table 5
Classification accuracy of different DR methods.

Datasets	MKL-SR	ESR	MKL-SRTR	KMFA	EMFA
Ionosphere	89.53 ± 3.67	91.56 ± 0.86	93.16 ± 0.72	93.64 ± 1.47	**95.84 ± 1.56**
Sonar	80.37 ± 4.35	84.89 ± 4.13	86.75 ± 2.57	86.43 ± 3.21	**87.65 ± 2.39**
USPS	90.68 ± 0.77	93.86 ± 0.43	94.31 ± 0.51	95.26 ± 0.93	**96.69 ± 0.45**
Isolet	95.78 ± 0.22	96.90 ± 0.11	**97.61 ± 0.12**	96.92 ± 0.35	97.25 ± 0.22
MNIST	93.25 ± 0.62	93.71 ± 0.82	93.87 ± 0.75	94.57 ± 1.12	**95.35 ± 0.95**
COIL-20	93.26 ± 0.89	94.63 ± 0.35	95.70 ± 0.26	96.48 ± 0.26	**97.47 ± 0.15**

Table 6
Recognition accuracy rates on PIE (mean ± std-dev%).

Train size	MKL-SR	MKL-SRTR	ESR	KMFA	EMFA
5 × 68	69.1 ± 1.5	73.8 ± 1.6	74.7 ± 0.9	75.6 ± 1.5	**77.1 ± 1.2**
10 × 68	86.4 ± 0.8	87.3 ± 1.7	88.4 ± 0.6	88.9 ± 1.1	**90.7 ± 0.8**
20 × 68	90.3 ± 0.7	91.2 ± 0.8	93.5 ± 0.4	94.3 ± 0.8	**94.9 ± 0.9**
30 × 68	92.2 ± 0.7	93.4 ± 0.8	95.8 ± 0.4	96.0 ± 0.9	**96.3 ± 0.7**
40 × 68	94.9 ± 0.8	96.2 ± 0.7	96.7 ± 0.3	97.2 ± 0.6	**97.9 ± 0.6**

Table 7
Recognition accuracy rates on Extended Yale B (mean ± std-dev%).

Train size	MKL-SR	MKL-SRTR	ESR	KMFA	EMFA
10 × 38	72.6 ± 1.5	74.2 ± 1.5	74.9 ± 1.4	77.3 ± 1.8	**78.8 ± 1.5**
15 × 38	84.8 ± 0.7	86.7 ± 0.6	88.1 ± 0.7	88.7 ± 0.6	**90.2 ± 0.4**
20 × 38	91.5 ± 0.7	93.9 ± 0.4	95.2 ± 0.4	95.8 ± 0.6	**96.5 ± 0.3**
25 × 38	93.9 ± 0.5	94.6 ± 0.3	96.8 ± 0.3	96.6 ± 0.4	**97.8 ± 0.2**

In each experiment, we select randomly samples to form training and testing sets with ratio 1:1. All experiments have been repeated 30 times, and Table 5 summarizes the mean classification accuracies and the standard deviations of different algorithms. To evaluate the performance of these algorithms, we performed the ELM classification algorithm in each learned subspace. For ESR and EMFA, the dimension of the subspace is c, where c is the number of categories, and the regularization parameter γ was tuned by a 10-fold cross-validation on the training data over the range of $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$. Specifically, the Gaussian function was selected and the number of hidden nodes L was tuned by a 10-fold cross-validation over the range of {100, 150, 200, …, 1000}. For EMFA, k_1 and k_2 are set as 10.

As can be seen from Table 5, EMFA evidently outperforms ESR, MKL-SR, MKL-SRTR and KMFA in most datasets, which achieves five best recognition rates among six datasets. This is due to the fact that EMFA utilizes the penalty graph to characterize the interclass marginal point adjacency relationship. Without prior information on data distributions, the interclass margin can better characterize the separability of different classes than the interclass variances in other algorithms. For KMFA, it is difficult to specify the optimal parameters of kernel functions, which generally results in locally optimal solutions. It demonstrates that EMFA makes good use of SR, MFA and ELM to achieve the outstanding discriminant analysis power and yields the best nonlinear low-dimensional representations.

For each individual of PIE, l (= 5, 10, 20, 30, 40) images are randomly selected for training and the rest are used for testing. The classification accuracy rates of each method on PIE are shown in Table 6. For each given l, the results are averaged over 30 random splits and report the mean as well as the standard deviation. As can be seen from Table 6, MKL-SR and MKL-SRTR cannot achieve the satisfactory results, which is due to the fact that these methods have to utilize iterative optimization methods to obtain local optima. In addition, ESR, MKL-SR and MKL-SRTR have the assumption that the distribution of each class is considered to be a unimodal Gaussian and separability of the different classes cannot be well characterized by interclass scatter. Thus,

Table 8
Computation time of different classification methods on PIE(s).

Train size	ESR		MFA		KMFA		EMFA	
	Training time	Testing time	Training time	Testing time	Training time	Testing time	Training time	Testing time
5 × 68	0.43	0.27	0.26	0.27	0.57	0.28	0.45	0.26
10 × 68	0.66	0.25	0.31	0.26	0.92	0.26	0.83	0.24
20 × 68	1.23	0.24	0.36	0.25	1.63	0.27	1.55	0.23
30 × 68	1.36	0.22	0.43	0.23	2.54	0.23	1.76	0.22
40 × 68	1.44	0.21	0.51	0.21	3.26	0.22	2.03	0.21

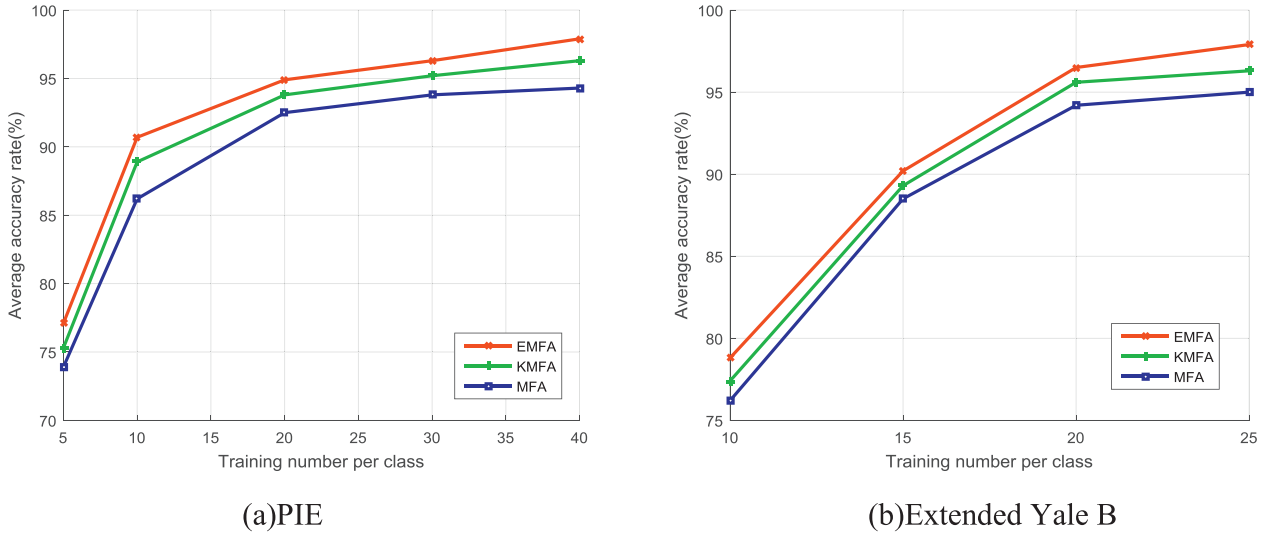

Fig. 1. Face recognition accuracies of EMFA, KMFA and MFA on PIE and Extended Yale B. (a) Results on PIE; (b) results on Extended Yale B.

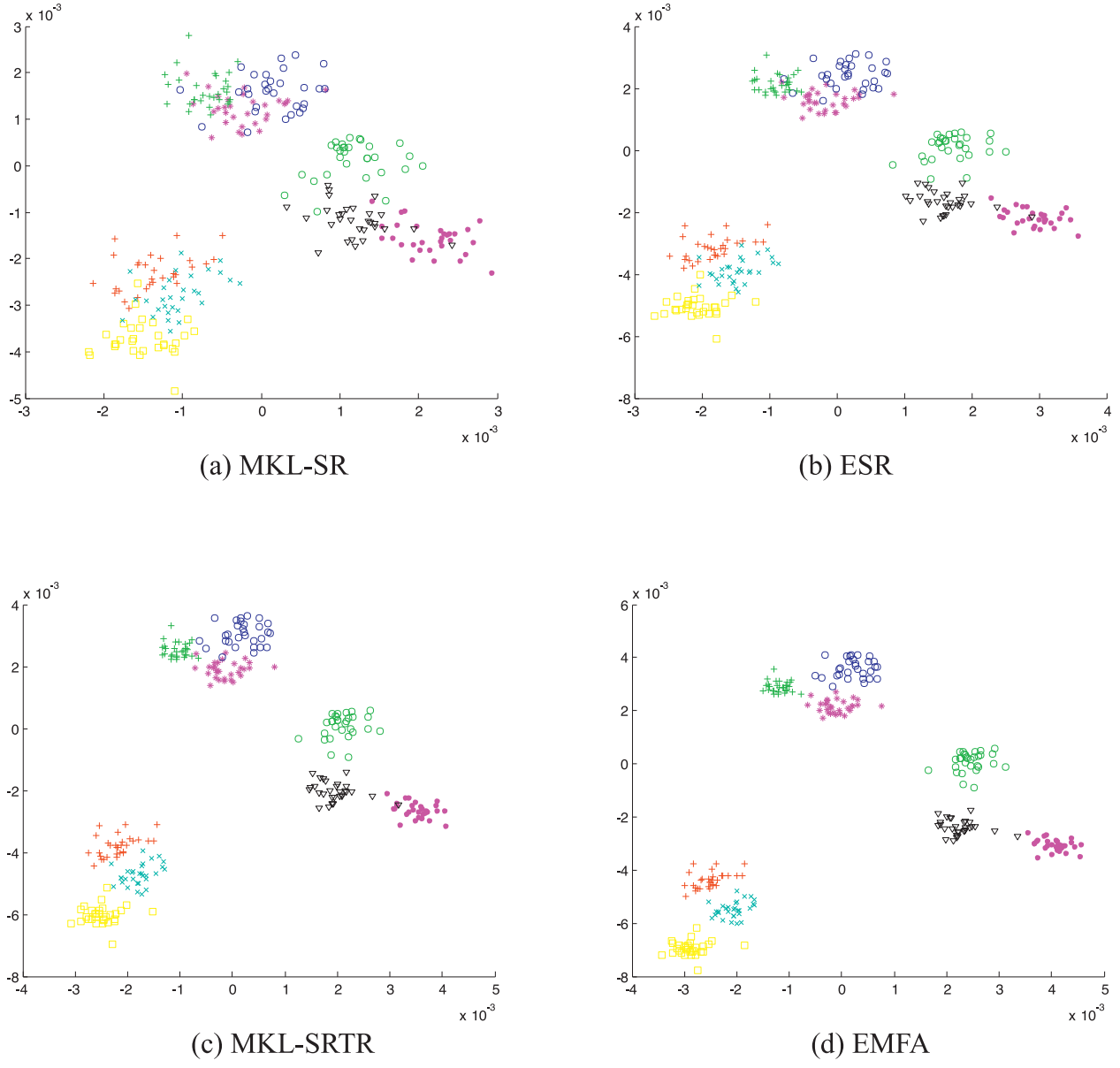

Fig. 2. Comparison of two-dimensional embedded results obtained by different algorithms on the first 9 classes of PIE. (a) Projection of training data with supervised MKL-SR; (b) projection of training data with supervised ESR; (c) projection of training data with supervised MKL-SRTR; (d) projection of training data with supervised EMFA.

they cannot solve the over-fitting problem in the small sample size case.

For the Extended Yale B face data set, we selected 50 images for each class. A random subset with l (= 10, 15, 20, 25) images per individual was first taken to form the training set and the rest of the data set was used to be the testing set. The experiments are repeated over 30 random splits. Table 7 reports the mean classification accuracy rates of each method as well as the standard deviation. From Table 7, we can observe that the proposed EMFA algorithm achieves all best results for each l, which shows that EMFA effectively combines MFA with SR to overcome the limitations of LDA.

We further report the training and testing time of ELM based on ESR and the proposed method in Table 8, where the training time includes the computational time of the dimension reduction for the testing data and the training time of the ELM algorithm. As can be seen from Table 8, ESR and EMFA, based on ELM, have close time cost. Since EMFA is a special case of KMFA, its learning speed is faster than that of KMFA. MFA runs much faster than other algorithms due to its linearity. Overall, compared with other algorithms, the proposed algorithm can achieve better performance at much faster learning speed, which is consistent with the theoretical analysis.

Finally, we carry out MFA, KMFA and EMFA, which are all based on spectral regression, and compare face recognition accuracies of the proposed algorithms on PIE and Extended Yale B. Similar to the experimental settings in [8], the Gaussian Kernel $\exp\{-\|x-y\|^2/\delta^2\}$ is used and parameter δ is set as $\delta = 2^{(n-10)/2.5}\delta_0$, where δ_0 is the standard derivation of the training data set. The best results are reported using different kernel parameters. The average recognition results of each method vs. the number of training data are shown in Fig. 1. As can be seen in Fig. 1, EMFA and KMFA significantly have better clustering performance than MFA on these two face datasets. The performance of EMFA is best among these algorithms. This is due to the fact that EMFA and KMFA are all nonlinear dimensionality reduction methods based on spectral regression, which aim to learn nonlinear projections from high-dimensional spaces to low-dimensional

ones. Correspondingly, MFA is actually a kind of linear dimensionality reduction methods based on spectral regression, which is not applicable to nonlinear face datasets. Although EMFA is a special case of KMFA, it does not need to learn kernel parameters and randomly generates parameters of activation functions between input layers and hidden layers without human interventions. In contrast, KMFA has to empirically choose different kernel parameters to obtain satisfactory performance. But, it is hard to find a way to specify the optimal kernel parameter. Consequently, EMFA is the most cost-efficient algorithm among these supervised dimensionality reduction methods.

To visualize the supervised dimensionality reduction results, we selected training data from the first 9 classes of PIE and projected them into a two-dimensional subspace to generate a graphical representation, shown in Fig. 2. From Fig. 2, we can observe that the embedded results obtained by EMFA are separated from each other more clearly than other algorithms. The embedded data obtained by EMFA has the best separability, which further validates that the performance of EMFA is much better than that of other algorithms in the supervised case.

5. Conclusion

In this paper, we extend the Marginal Fisher Criterion based on ELM. Combined with SR and the proposed extended Marginal Fisher Criterion, a family of dimensionality reduction algorithms, are proposed for supervised nonlinear dimensionality reduction. By virtue of SR, we solve the out-of-sample extension problem. By means of ELM, our method not only introduces the nonlinear embedding functions, but also improves the efficiency of KMFA. Furthermore, it is more general for Fisher discriminant analysis. Experimental results on benchmark datasets validate the promising performance of the proposed method.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (2017XKQY082).

References

[1] A. Nazarpour, P. Adibi, Two-stage multiple kernel learning for supervised dimensionality reduction, Pattern Recognit. 48 (5) (2015) 1854–1862.
[2] X. Zhu, Z. Huang, Y. Yang, H.T. Shen, C. Xu, J. Luo, Self-taught dimensionality reduction on the high-dimensional small-sized data, Pattern Recognit. 46 (1) (2013) 215–229.
[3] X. Zhu, Z. Huang, H.T. Shen, J. Cheng, C. Xu, Dimensionality reduction by mixed kernel canonical correlation analysis, Pattern Recognit. 45 (8) (2012) 3003–3016.
[4] Y.-Y. Lin, T.-L. Liu, C.-S. Fuh, Multiple kernel learning for dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell. 33 (6) (2011) 1147–1160.
[5] W. Jiang, F.-L. Chung. A trace ratio maximization approach to multiple kernel-based dimensionality reduction, Neural Networks, 2014, 49, pp. 96–106.
[6] B. Liu, S. Xia, Y. Zhou, Multiple kernel spectral regression for dimensionality reduction, J. Appl. Math. 2013 (1) (2013) 1044–1065.
[7] M. Liu, W. Sun, B. Liu, Multiple kernel dimensionality reduction via spectral regression and trace ratio maximization, Knowl. Based Syst. 83 (1) (2015) 159–169.
[8] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell. 29 (1) (2007) 40–51.
[9] B Liu, S.X. Xia, F.R. Meng, et al., Extreme spectral regression for efficient regularized subspace learning, Neurocomputing 149 (PA) (2015) 171–179.
[10] J. Tang, C. Deng, G.-B. Huang, Extreme learning machine for multilayer perceptron, IEEE Trans. Neural Networks Learn. Syst. 27 (4) (2017) 809–821.
[11] G.-B. Huang, Z. Bai, L.L.C. Kasun, C.M. Vong, Local receptive fields based extreme learning machine, IEEE Comput. Intell. Mag. 10 (2) (2015) 18–29.
[12] L.L.C. Kasun, H. Zhou, G.-B. Huang, C.M. Vong, Representational learning with extreme learning machine for big data, IEEE Intell. Syst. 28 (December (6)) (2013) 31–34.
[13] Y. Miche, M. van Heeswijk, P. Bas, O. Simula, A. Lendasse, TROP-ELM: a double-regularized ELM using LARs and Tikhonov regularization, Neurocomputing 74 (16) (2011) 2413–2421.
[14] G.B. Huang, H.M Zhou, X.J. Ding, Extreme learning machine for regression and multiclass classification, IEEE Trans. Syst. Man Cybern. Part B Cybern. 42 (2) (2012) 513–529.
[15] G.B. Huang, Q.Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (1) (2006) 489–501.
[16] D. Cai, X. He, J. Han, SRDA: an efficient algorithm for large scale discriminant analysis, IEEE Trans. Knowl. Data Eng. 20 (1) (2008) 1–12.
[17] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: a geometric framework for learning from labeled and unlabeled examples, J. Mach. Learn. Res. 7 (1) (2006) 2399–2434.

Bing Liu received the B.Sc., M.Sc., and Ph.D. degrees in 2002, 2005, and 2013, respectively, from China University of Ming and Technology (CUMT), Xuzhou, China. He is currently an associate professor at School of Computer Science and Technology, CUMT. His current research interests include deep learning, nonlinear dimensionality reduction, feature extraction and selection, compressed sensing, and sparse machine learning methods.

Yong Zhou is currently a professor at School of Computer Science and Technology in China University of Ming and Technology (CUMT). He got the B.Sc., M.Sc., and Ph.D. degrees from Department of Electrical Engineering at CUMT in China. His research interests include machine learning, nonlinear dimensionality reduction and cognitive science.

Zhan-guo Xia is currently an associate professor at School of Computer Science and Technology in China University of Ming and Technology (CUMT). He got the B.Sc., M.Sc., and Ph.D. degrees at CUMT in China. His research interests include machine learning, pattern recognition.

Peng Liu is currently an associate professor at Internet of Things Perception Mine Research Center, China University of Mining and Technology. His research interests include machine learning, pattern recognition.

Qiu-yan Yan is currently an associate professor at School of Computer Science and Technology in China University of Mining and Technology (CUMT). She got the B.Sc., M.Sc., and Ph.D. degrees at CUMT in China. Her research interests include machine learning, pattern recognition.

Hui Xu is currently a lecturer at School of Computer Science and Technology in China University of Mining and Technology. Her research interests include intelligent computing, optimization method.