Pattern Recognition 44 (2011) 877–885

Contents lists available at ScienceDirect

Pattern Recognition

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

A linear discriminant analysis method based on mutual information maximization

Haihong Zhang [a,*], Cuntai Guan [a], Yuanqing Li [b]

[a] Institute for Infocomm Research, A*STAR, Singapore 138632, Singapore
[b] School of Automation Science and Technology, South China University of Technology, Guangzhou 510460, China

ARTICLE INFO

Article history:
Received 9 February 2010
Received in revised form
6 September 2010
Accepted 7 November 2010

Keywords:
Discriminant analysis
Mutual information
Feature extraction

ABSTRACT

We present a new linear discriminant analysis method based on information theory, where the mutual information between linearly transformed input data and the class labels is maximized. First, we introduce a kernel-based estimate of mutual information with a variable kernel size. Furthermore, we devise a learning algorithm that maximizes the mutual information w.r.t. the linear transformation. Two experiments are conducted: the first one uses a toy problem to visualize and compare the transformation vectors in the original input space; the second one evaluates the performance of the method for classification by employing cross-validation tests on four datasets from the UCI repository. Various classifiers are investigated. Our results show that this method can significantly boost class separability over conventional methods, especially for nonlinear classification.

1. Introduction

Discriminant analysis (DA) (also referred to as discriminant feature extraction) aims to find a transformation of input variables into latent variables (features) with maximum class separability [1]. It comprises one of the key subjects in pattern recognition, and differs largely from feature selection (e.g. [2,3]) in that, while feature selection ranks and selects the input variables according to their predictive significance, feature extraction transforms the variables by, e.g. a linear combination. In this work, we focus on linear DA in favor of its simplicity and possibility for nonlinear extension via the kernel trick [4].

The most widely used DA method is known as Fisher linear discriminant analysis (LDA) [5,6], based on Fisher's criterion that the ratio of inter-class scatter over intra-class scatter is maximized. It is designed for 2-class problems under the *homoscedastic* condition that all classes share one Gaussian covariance matrix. The non-optimality or sub-optimality of LDA is well recognized in the literature (see [7–9]): neither is it able to deal with heteroscedastic data (i.e. classes do not have equal covariance matrices) in a proper way, nor it is Bayes optimum for more than 2-class problems.

A number of alternative LDA techniques have been proposed in the past to address the problems. For multi-class problems, non-parametric scatter matrices were proposed in the so-called non-parametric discriminant analysis (NDA) [10]. The matrices were generally of full rank, allowing more features than the class number to be extracted. Besides, the non-parametric methodology allowed DA to work well even for non-Gaussian datasets. Another extension of LDA to multi-class was reported in [11], where the approximate Pairwise Accuracy Criteria (aPAC) replaced Fisher's criterion. Specifically, aPAC weighted the contribution of individual class pairs in terms of Bayes error (a similar weighting scheme was reported in [12]). More recently, a minimum Bayes error method was reported for dealing with multi-class homoscedastic data [7].

In the heteroscedastic discriminant analysis (HDA) [13], all the classes were allowed to have different covariance matrices. It was derived in a maximum-likelihood framework to handle heteroscedastic data. Without closed-form solution, the method resorted to numerical optimization. Another heteroscedastic extension of LDA (HELDA) was presented in [14] that utilized the *Chernoff* criterion to handle multi-class, heteroscedastic data. Favorably, the Chernoff criterion led to a closed-form solution.

Generally, these methods are limited with their unimodal Gaussian assumptions. Furthermore, theoretical analysis [8] has concluded that generalized eigen-based linear equations (widely used in many methods above) may not work whether or not the data are homoscedastic or heteroscedastic.

An alternative is to use the *maximum mutual information* (MMI) criterion. Stemmed from information theory, mutual information [15] basically measures how much knowing the features reduces the uncertainty about the class labels. In [9], the authors studied the relationship between MMI and the criteria of LDA and its heteroscedastic extensions. Importantly, the study has shown that MMI is Bayesian optimum under more general conditions than that for earlier criteria. Besides, MMI can connect to minimum Bayes error via lower and upper bound [16]. The criterion has also been applied to feature selection [2,3,17,18].

In view of its superior capability for handling complex data distributions, MMI-based discriminant analysis [19] or blind source

* Corresponding author.
E-mail addresses: hhzhang@i2r.a-star.edu.sg, gnohiah@gmail.com (H. Zhang), ctguan@i2r.a-star.edu.sg (C. Guan), auyqli@scut.edu.cn (Y. Li).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.11.001

separation [20] was promoted recently. Especially in [19], the method "MeRMaID_SIG" used Renyi's quadratic entropy, in favor of its lower computational complexity, to replace Shannon's entropy for the mutual information formulation. However, the quadratic entropy generally diverges from Shannon's entropy which is the fundamental of information theory behind the MMI approach. Furthermore, that method is limited to predefined or annealed [21] kernel size, while, as we will show later at the end of Section 2, a kernel size as an intrinsic function of the transformation parameters is preferable for making MMI a consistent measure of separability.

In this paper we propose a new MMI-based DA method and demonstrate its superiority. In particular, we introduce a non-parametric, kernel-based estimate of mutual information based on Shannon's entropy. Particularly, the kernel size is defined as a function of the feature distributions, and the estimate is invariant against of dilatation/contraction of output dimensions. In other words, the kernel size becomes a function of the DA transformation matrix. Furthermore, we derive from the estimate a gradient-based learning algorithm.

We investigate the method using a toy problem firstly. The transformation vectors are visualized in the original 2D input space. We then evaluate the method using cross-validation on four datasets from the UCI repository, while comparing it with existing DA methods including aPAC, HELDA and MeRMaID_SIG. For assessing the class separability of features generated by different methods, we employ a linear and a nonlinear support vector machines (SVMs) in addition to a Parzen window classifier ([1, Section 6.1]).

The remainder of the paper is organized as follows. Section 2 presents a robust mutual information estimate for the objective function of linear discriminant analysis. Section 3 describes a gradient-based learning algorithm, followed by experimental results in Section 4. Section 5 presents discussions, and Section 6 finally concludes the paper.

2. Mutual information estimate

Let a variable in the original space be $\mathbf{x}\in\mathbb{R}^n$. It is linearly transformed into a latent variable (i.e. a feature vector) $\mathbf{y}\in\mathbb{R}^m$ by

$$\mathbf{y}=\mathbf{W}\mathbf{x}, \tag{1}$$

where $\mathbf{W}\in\mathbb{R}^{m\times n}$ is a projection matrix that comprises n_w column vectors.

The mutual information between the variable $\mathcal{Y}$ of $\mathbf{y}$ and the class variable $\mathcal{C}$ is known as

$$I(\mathcal{Y},\mathcal{C})=H(\mathcal{Y})-H(\mathcal{Y}|\mathcal{C})=H(\mathcal{Y})-\sum_{c\in\mathcal{C}}H(\mathcal{Y}|c)P(c), \tag{2}$$

where c is a particular class label. The entropy $H(\mathcal{Y})$ is determined by probability density function $p_y(\mathbf{y})$:

$$H(\mathcal{Y})=-\int_{\mathbf{y}}p_y(\mathbf{y})\log(p_y(\mathbf{y}))\,d\mathbf{y}. \tag{3}$$

Now we show that the mutual information is invariant under nonsingular linear transformation of the feature space. Consider two linear transformation matrices $\mathbf{W}$ and $\mathbf{W}'$, each transforms the original data vector $\mathbf{x}$ to $\mathbf{y}$ and $\mathbf{y}'$, respectively. Consider the feature vector $\mathbf{y}$ being transformed by a nonsingular (i.e. full rank) square matrix G and a constant vector $\mathbf{g}_0$.

$$\mathbf{y}'=G\mathbf{y}+\mathbf{g}_0,\quad \text{i.e. } \mathbf{W}'\mathbf{x}=G\mathbf{W}\mathbf{x}+\mathbf{g}_0. \tag{4}$$

We would like to note that the transformation as Eq. (4) between two feature spaces ($\mathbf{y}$ and $\mathbf{y}'$) is equivalent to a transformation between the two linear methods' matrices $\mathbf{W}$ and $\mathbf{W}'=G\mathbf{W}$ plus a constant additive vector $\mathbf{g}_0$. We would like to emphasize that the nonsingular transformation G refers to the relationship between two feature spaces. And, it should not be confused with the usually non-square transformation matrix $\mathbf{W}$.

The mutual information becomes

$$H(\mathcal{Y}')=H(G\mathcal{Y}+\mathbf{g}_0)=-\int_{-\infty}^{\infty}p_{y'}(G\mathbf{y}+\mathbf{g}_0)\log(p_{y'}(G\mathbf{y}+\mathbf{g}_0))d(G\mathbf{y}+\mathbf{g}_0). \tag{5}$$

Since G is a full-rank square matrix, we note

$$\left|\det\left(\frac{d(G\mathbf{y}+\mathbf{g}_0)}{d\mathbf{y}}\right)\right|=|\det(G)|, \tag{6}$$

and write the following:

$$\begin{aligned}H(\mathcal{Y}')&=-|\det(G)|\int_{-\infty}^{\infty}\frac{p_y(\mathbf{y})}{|\det(G)|}\log\left(\frac{p_y(\mathbf{y})}{|\det(G)|}\right)d\mathbf{y}\\&=\log(|\det(G)|)-\int_{-\infty}^{\infty}p_y(\mathbf{y})\log(p_y(\mathbf{y}))\,d\mathbf{y}=H(\mathcal{Y})+\log(|\det(G)|).\end{aligned} \tag{7}$$

Therefore, the entropy is changed by $\log(|\det(G)|)$. But the change is cancelled out in the mutual information:

$$\begin{aligned}I(\mathcal{Y}',\mathcal{C})&=H(\mathcal{Y}')-\sum_{c\in\mathcal{C}}P_c(c)H(\mathcal{Y}'|c)\\&=H(\mathcal{Y})+\log(|\det(G)|)-\sum_{c\in\mathcal{C}}P_c(c)\left[H(\mathcal{Y}|c)+\log(|\det(G)|)\right]\\&=H(\mathcal{Y})-\sum_{c\in\mathcal{C}}P_c(c)H(\mathcal{Y}|c)=I(\mathcal{Y},\mathcal{C}).\end{aligned} \tag{8}$$

Hence, the mutual information is invariant against nonsingular linear transformation like Eq. (4). This is an important property required for a metric of feature extraction: since nonsingular linear transformation is invertible, it shall have no effect in the class separability of the features under transformation.

Shannon's mutual information is a functional of underlying probability distributions, thus it has no analytical form for computation. Instead, like in [22,18,3] we resort to a Monte Carlo approximation described below.

Suppose there are l samples of data: $\{\mathbf{x}_i\}$, $i=1,\ldots,l$, transformed to $\{\mathbf{y}_i\}$ by Eq. (1). Now consider how to estimate the mutual information of $\{\mathbf{y}_i\}$ with the corresponding class labels. First of all, we note that the entropy of the feature vector $\mathcal{Y}$ can be expressed as the expectation of $\log(p_y(\mathbf{y}))$

$$H(\mathcal{Y})=-E[\log(p_{\mathbf{y}}(\mathbf{y}))]\cong-\frac{1}{l}\sum_{i=1}^{l}\log(p_{\mathbf{y}}(\mathbf{y}_i)). \tag{9}$$

Subsequently, $p_y(\mathbf{y})$ can be estimated with kernel density estimation [23]:

$$p_{\mathbf{y}}(\mathbf{y})\cong\frac{1}{l}\sum_{i=1}^{l}\varphi(\mathbf{y}-\mathbf{y}_i), \tag{10}$$

where φ usually takes a Gaussian form:

$$\varphi(\mathbf{y}-\mathbf{y}_i)=\alpha\exp(-\tfrac{1}{2}(\mathbf{y}-\mathbf{y}_i)^T\boldsymbol{\Psi}^{-1}(\mathbf{y}-\mathbf{y}_i)). \tag{11}$$

Here the factor α shall make the integral of $\hat{p}_{\mathbf{y}}(\mathbf{y})$ equal to 1, as required for probability density function. In this work we consider a constant α. The kernel size matrix $\boldsymbol{\Psi}$ is diagonal, and each diagonal element is determined by

$$\psi_{k,k}=\zeta\frac{1}{l-1}\sum_{i=1}^{l}(y_{ik}-\overline{y_k})^2, \tag{12}$$

where $\overline{\mathbf{y}}_k$ is the empirical mean of $\mathbf{y}_k$, and we set the coefficient $\zeta=(4/3l)^{0.1}$ according to the normal optimal smoothing strategy [24].

By substituting Eq. (10) into Eq. (9), the entropy of the feature vector can be estimated using

$$\hat{H}(\mathcal{Y})=-\frac{1}{l}\sum_{i=1}^{l}\log\left\{\frac{1}{l}\sum_{j=1}^{l}\varphi(\mathbf{y}_i-\mathbf{y}_j)\right\}, \tag{13}$$

and the conditional intra-class entropy $\hat{H}(\mathcal{Y}|c)$ can be estimated similarly by using class c samples only.

The mutual information estimate becomes

$$\hat{I}(\mathcal{Y},\mathcal{C}) = \hat{H}(\mathcal{Y}) - \sum_{c} P(c)\hat{H}(\mathcal{Y}|c). \tag{14}$$

Importantly, we show in below that the mutual information estimate is invariant against the following transformation:

$$\mathbf{y}' = F\mathbf{y} + \mathbf{g}_0, \tag{15}$$

where F is a full-rank diagonal matrix whose k-th diagonal element is denoted by $f_{k,k}$, and $\mathbf{g}_0$ is a translation vector. In a geometric sense, the transformation is equivalent to dilating (if $f_{k,k} > 1$)/contracting (if $f_{k,k} < 1$) the dimensions of the latent variable $\mathbf{y}$, in addition to translation. Therefore, such a transformation is invertible, and the class separability remains the same after the transformation.

Now we study the impact of the transformation on the mutual information estimate. After the transformation, the diagonal matrix Ψ (Eq. (12)) becomes a diagonal matrix Ψ' whose elements are given by

$$\psi'_{k,k} = f_{k,k}^2 \psi_{k,k}. \tag{16}$$

With the new diagonal matrix Ψ', the Gaussian-kernel function by Eq. (11) becomes

$$\begin{aligned}\hat{\varphi}(\mathbf{y}'_i - \mathbf{y}'_j) &= \alpha \exp\left(-\frac{1}{2}(\mathbf{y}'_i - \mathbf{y}'_j)^T \Psi'^{-1}(\mathbf{y}'_i - \mathbf{y}'_j)\right) \\ &= \alpha \exp\left(-\frac{1}{2}\sum_{k=1}^{d_y}(f_{k,k}y_{ik} + f_{0k} - f_{k,k}y_{ik} - f_{0k})^2 f_{k,k}^{-2}\psi_{k,k}\right) \\ &= \alpha \exp\left(-\frac{1}{2}(\mathbf{y} - \mathbf{y}_i)^T \psi^{-1}(\mathbf{y} - \mathbf{y}_i)\right) = \varphi(\mathbf{y} - \mathbf{y}_i).\end{aligned} \tag{17}$$

Therefore, the transformation in Eq. (15) does not change the Gaussian kernel function output, thus has no effect on the entropy estimate in Eq. (10) as well as on the mutual information estimate in Eq. (2).

The above property is important for the mutual information estimate as the metric (criterion) for class separability in discriminant analysis. It is known that Bayes error is the best criterion for class separability, but it is just too complex and useless as an analytical tool ([1, Chapter 10]). Therefore, other criteria are needed in practice that shall be as consistent as possible with the Bayes error. If there is a nonsingular transformation between two linear analysis transforms, they are equivalent in Bayes error [25] since no classification information is lost under the transformation. Accordingly, criteria for discriminant analysis shall also be invariant under such a transformation, at least under dilating/contracting. In this sense, the mutual information estimate is favorable compared with prior arts: for example, the criterion used in [19] employed a fixed kernel size and would not be invariant under dilating/contracting. On the other hand, we can see from the above development that the key point of the proposed estimate is in its kernel size being described as an appropriate function of the features (i.e. as a function of the linear projection matrix W).

Therefore, we propose to use the mutual information estimate $\hat{I}(\mathcal{Y},\mathcal{C})$ as the objective function (i.e. criterion) for discriminant analysis, and derive in the following an algorithm to learn the optimum W which produces maximum mutual information.

3. Learning algorithm

The objective of learning is therefore to find the optimum transformation matrix W that maximizes the mutual information estimate $\hat{I}(\mathcal{Y},\mathcal{C})$. However, the mutual information estimate expressed by Eqs. (13) and (14) takes a rather complicated form, and there is no closed-form solution to maximization. (The estimate can be viewed as combinations of a number of Gaussian functions, which are in turn determined by the transformation matrix W for the discriminant analysis.)

Here we propose a numerical solution by employing a gradient-based optimization algorithm. To this end, we first consider each projection vector, e.g. the k-th projection vector $\mathbf{w}_k$ in the linear transformation matrix, and note that the gradient of mutual information estimate with respect to the projection vector is

$$\nabla_{\mathbf{w}_k} I(\mathcal{Y},\mathcal{C}) = \nabla_{\mathbf{w}_k} H(\mathcal{Y}) - \sum_{c \in \mathcal{C}} P(c)\nabla_{\mathbf{w}_k} H(\mathcal{Y}|c). \tag{18}$$

From Eq. (13), we have

$$\nabla_{\mathbf{w}_k} H(\mathcal{Y}) = -\frac{1}{l}\sum_{i=1}^{l}\beta_i \frac{1}{l}\sum_{j=1}^{l}\frac{\partial\varphi(\mathbf{y}_i - \mathbf{y}_j)}{\partial \mathbf{w}_k}, \tag{19}$$

where

$$\beta_i = \left(\frac{1}{l}\sum_{j=1}^{l}\varphi(\mathbf{y}_i - \mathbf{y}_j)\right)^{-1}. \tag{20}$$

From Eq. (11), we have

$$\frac{\partial\varphi(\mathbf{y}_i - \mathbf{y}_j)}{\partial \mathbf{w}_k} = -\frac{1}{2}\varphi(\mathbf{y}_i - \mathbf{y}_j)\frac{\partial(\mathbf{y}_i - \mathbf{y}_j)^T \Psi^{-1}(\mathbf{y}_i - \mathbf{y}_j)}{\partial \mathbf{w}_k}. \tag{21}$$

Let us denote the quadratic function $(\mathbf{y}_i - \mathbf{y}_j)^T \Psi^{-1}(\mathbf{y}_i - \mathbf{y}_j)$ by ϑ_{ij}. And, ϑ_{ij} can be decomposed as below:

$$\vartheta_{ij} = \sum_{k_1=1}^{d_o}\sum_{k_2=1}^{d_o}\psi^{-1}_{k_1k_2}(y_{ik_1} - y_{jk_1})(y_{ik_2} - y_{jk_2}). \tag{22}$$

The gradient of ϑ_{ij} is

$$\frac{\partial\vartheta_{ij}}{\partial \mathbf{w}_k} = \sum_{k_1=1}^{d_o}\sum_{k_2=1}^{d_o}\left[\frac{\partial\psi^{-1}_{k_1k_2}}{\partial \mathbf{w}_k}(y_{ik_1} - y_{jk_1})(y_{ik_2} - y_{jk_2}) + \psi^{-1}_{k_1k_2}\frac{\partial(y_{ik_1} - y_{jk_1})(y_{ik_2} - y_{jk_2})}{\partial \mathbf{w}_k}\right]. \tag{23}$$

Consider that $(y_{ik_1} - y_{jk_2})^2$ is a function of $\mathbf{w}_k$ if and only if $k_1 = k$ and/or $k_2 = k$, and $\psi^{-1}_{k_1k_2}$ is a function of $\mathbf{w}_k$ if and only if $k_1 = k_2 = k$. Furthermore, $\psi^{-1}_{k_1k_2} = 0$ if $k_1 \neq k$ or $k_2 \neq k$. The expression of the gradient above can be written as

$$\frac{\partial\vartheta_{ij}}{\partial \mathbf{w}_k} = \frac{\partial\psi^{-1}_{kk}}{\partial \mathbf{w}_k}(y_{ik} - y_{jk})^2 + \psi^{-1}_{kk_1}\frac{\partial(y_{ik} - y_{jk})^2}{\partial \mathbf{w}_k}. \tag{24}$$

For the computation of $\partial\psi^{-1}_{kk}/\partial \mathbf{w}_k$, we first note from Eq. (12) that ψ_{kk} can be expressed as a direct function of $\mathbf{w}_k$:

$$\psi_{k,k} = \zeta\frac{1}{l-1}\sum_{i=1}^{l}\mathbf{w}_k^T(\mathbf{x}_i - \overline{\mathbf{x}}_i)(\mathbf{x}_i - \overline{\mathbf{x}}_i)^T\mathbf{w}_k \equiv \zeta\mathbf{w}_k^T\Phi\mathbf{w}_k, \tag{25}$$

where $\overline{\mathbf{x}}$ is the empirical mean of $\mathbf{x}$, and Φ denotes the empirical covariance matrix of $\mathbf{x}$. It then follows that

$$\frac{\partial\psi^{-1}_{kk}}{\partial \mathbf{w}_k} = \frac{\eta}{2}\frac{\partial(\mathbf{w}_k^T\Phi\mathbf{w}_k)}{\partial \mathbf{w}_k} = \eta\Phi\mathbf{w}_k, \tag{26}$$

where for simplicity we denote

$$\eta = -2\zeta^{-1}(\mathbf{w}_k^T\Phi\mathbf{w}_k)^{-2}. \tag{27}$$

Furthermore,

$$\frac{\partial(y_{ik} - y_{jk})^2}{\partial \mathbf{w}_k} = 2(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T\mathbf{w}_k. \tag{28}$$

With the above equations, we can write the gradient of the entropy estimate $\hat{H}(\mathcal{Y})$ as

$$\nabla_{\mathbf{w}_k}\hat{H}(\mathcal{Y}) = \mathbf{A}\mathbf{w}_k, \tag{29}$$

where

$$\mathbf{A}=\frac{1}{2l^2}\sum_{i=1}^{l}\beta_i\sum_{j=1}^{l}\varphi(\mathbf{y}_i-\mathbf{y}_j)[\eta\Phi(y_{ik}-y_{jk})^2+2\psi_{kk}^{-1}(\mathbf{x}_i-\mathbf{x}_j)(\mathbf{x}_i-\mathbf{x}_j)^T]. \tag{30}$$

Similarly for each within-class entropy, we have $\mathbf{A}_c$. Therefore, the gradient of the mutual information is given by

$$\nabla_{\mathbf{w}_k}I(\mathcal{Y},\mathcal{C})=\left(\mathbf{A}-\sum_c P(c)\mathbf{A}_c\right)\mathbf{w}_k. \tag{31}$$

Note that because the multiplier $(\mathbf{A}-\sum_c P(c)\mathbf{A}_c)$ contains rather complicated functions of $\mathbf{W}$, the gradient in the above equation is indeed a nonlinear function of all $\mathbf{w}_k$. Using the above equations to compute the gradient for each projection vector $\mathbf{w}_k$, we employ an iterative optimization procedure to update the projection vectors by

$$\mathbf{w}_k^{(iter+1)}=\mathbf{w}_k^{(iter)}+\lambda\nabla_{\mathbf{w}_k}I^{(iter)}(\mathcal{Y},\mathcal{C}), \tag{32}$$

where λ is the step size. In this work, we perform a local line search procedure to determine the step size. The method tries a number (tentatively 15) of λ in the range of [0 0.01] to update $\mathbf{W}$, and checks if a local maximum of mutual information exists within the range excluding the boundary points. If a local maximum exists, the method uses that λ as the final step size in this iteration. Otherwise, it increases the range (tentatively by a factor of 1.5) and repeats the line search procedure.

The pseudocode of the optimization procedure is described in Fig. 1. Since the above method is a deterministic process, it is important to set an appropriate initial value for $\mathbf{W}$. In the work, we consider selecting initial value among a randomly generated set. In the following studies, we generate 50 samples for the initial value, and choose the one that produces the largest mutual information estimate.

4. Experimental results

4.1. Toy problem

We used a toy problem to investigate the proposed method, by visualizing the generated transformation vectors in the original space. Unlike the real world datasets to be used later that have high dimensionality ($\gg 3$), the toy problem consists of bivariate input samples, allowing visualization in an explicit form.

Fig. 2 illustrates the problem and the result. The two classes were not linearly separable: while the positive class (in red color) has a unimodal Gaussian distribution, the negative class (in blue color) is randomly scattered around a half-circle that surrounds the positive class. After applying various methods including aPAC, PCA, MeRMaID_SIG and the proposed method (MMILA), we plotted the resultant transformation vectors as lines starting from the zero point (note that the data were zero-meaned beforehand).

The near-horizontal line by aPAC can be reversed without affecting the learning and classification system. Hence, the three existing methods created quite similar transformation vectors. Furthermore, the vectors were almost parallel to the two axes, implying that none of the three methods can well explore the data structure.

The proposed method MMILA produced fairly different results. Interestingly, the two transformation vectors can be viewed as piece-wise linear approximations to the arc-shape separation between the two classes. Therefore, it seems that the MMILA features can better describe the discriminative data structure.

The methods were further tested under strong additive noise. As in subfigure (b), the sample distributions of the two classes were

Input: A set of training samples $\{x_i\}_{i=1}^{l}$ with corresponding class labels $\{c_i\}_{i=1}^{l}$.

Output: A linear transformation matrix W.

Algorithm:

1. Initialization: iteration step $iter = 0$, generate a number (e.g. 50 in the present work) of random W, and choose the one as W (0) which produces the largest mutual information estimate;
2. Compute the gradient of W $^{(iter)}$ according to Eq. 31 and related equations;
3. Perform a linear search that intuitively seeks a local maximum mutual information along the gradient direction. In the present work we simply try a range of the step size (λ in Eq. 32, see also Section 3 below Eq.32) to update W $^{(iter)}$ and check the mutual information estimate respectively, if no local maximum exists in the range, increase the search range and repeat the search until a local maximum is found and set it as W $^{(iter+1)}$;
4. Iteration step $iter = iter + 1$;
5. Check termination condition: compute the gain of mutual information estimate in this interation; terminate learning if the gain is smaller than a preset small threshold value (1e-4 in the present work), otherwise proceed to Step 2;

Fig. 1. The learning algorithm for the proposed maximum mutual information linear discriminant analysis.

quite overlapped due to noise. Nevertheless, MMILA still produced similar and reasonable results like in subfigure (a).

4.2. Real-world datasets

We used four real-world classification datasets from the UCI repository, include both 2-class data and multi-class data. The nature of the data is largely different among the sets. Details about the datasets are summarized in Table 1.

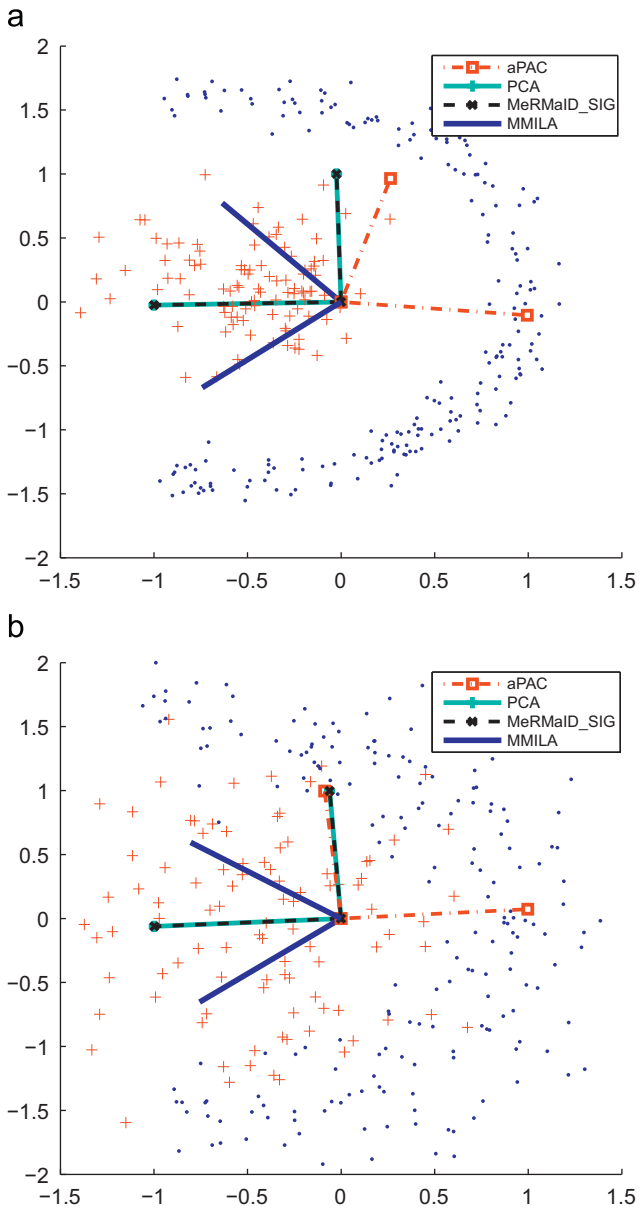

Fig. 2. Toy problem for discriminative feature extraction. (2-class problem) The two class samples are plotted as red circles or blue dots in (a), and (2-class problem with strong noise) as red crosses or blue dots in (b). See Subsection 4.1 for further description. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

The experiment was conducted in MATLAB, where every attribute in the data was linearly normalized to the range of [0 1] beforehand. The general objectives of the experiment are: to study the convergence of the iterative optimization method; to assess the performance of the MMI-based discriminant analysis method (referred to MMILA hereafter), in terms of class separability of the features.

We evaluated the class separability in terms of classification accuracy using randomized cross-validation and different classifiers. A 5×5-fold cross-validation is conducted using the Matlab function "crossvalind" from the MATLAB bioinformatics toolbox to generate random cross-validation partitions of data. Each cross-validation test was initialized with different random seeds. We used a linear support vector machine (SVM-L) a Gaussian-kernel support vector machine (SVM-G) (using the LIBSVM toolbox [26]), and a Parzen window classifier (referred to as Parzen hereafter) [1] that shared the same kernel size with MMILA. Each classifier learned from the training set to classify the test set samples during cross-validation.

The proposed method were compared against three existing methods, namely, aPAC, HELDA, PCA and MeRMaID_SIG. The randomized cross-validation settings were consistent across different DA methods, output dimensions, and the classifier choice.

The statistics of the classification result is summarized in Tables 2 and 3. Consider each combination of dataset and dimension as a particular case (e.g. "Musk" and dimension $==1$). Among all the 15 cases, MMILA yielded highest mean accuracy rate in 11 cases, aPAC in two cases, HELDA in two cases and PCA in one case only.

Since the classification performance varies with the output dimension of features [27,28], we plot the classification accuracy as a function of the output dimension in Fig. 3 so as to facilitate the analysis of the results. In the "Musk" and the "Glass" datasets, MMILA improved classification accuracy significantly over other discriminant analysis methods in the >2 dimensions. In "Yeast" and "Vehicle", MMILA also yielded the highest accuracy rates in most cases.

To further examine the statistical significance of the MMILA results compared with others, we ran t-tests of the hypothesis that MMILA produced a higher mean classification accuracy. Paired t-test using cross-validation results was performed using the Matlab function "ttest". Particularly, we compared the results of MMILA against those by aPAC (which provided overall the best accuracy among the existing methods) and those by MeRMaID_SIG, the state-of-the-art mutual information-based feature extraction method.

Fig. 4 further summarizes the comparison results in terms of statistics. Specifically, it illustrates how likely MMILA would outperform significantly (p-value < 0.05) the two existing methods of aPAC and MeRMaID_SIG. Compared with aPAC, MMILA tends to produce significant improvement in nonlinear classification and higher dimension. Compared with MeRMaID_SIG, MMILA also outperformed in nonlinear classification.

Table 1
Datasets used for evaluation. Only predictive attributes are considered.

Name	#instance	#attribute	#class	Remark
Musk	476	166	2	*Real-value attributes*; *class distribution*: 43.5%, 56.5%
Yeast	1484	8	10	*Real-value attributes*; *class distribution*: 16.4%, 28.9%, 31.2%, 29.6%, 23.6%, 34.4%, 11.0%, 2.0%, 1.4%, 0.3%
Glass	214	9	6	*Real-value attributes*; *the null class* (*i.e. no samples*) *from the original seven classes is removed*; *class distribution*: 32.7%, 35.5%, 7.9%, 6.1%, 4.2%, 13.6%
Vehicle	846	18	4	*Integer attributes in* [0 1018] *range*; *class distribution*: 23.5%, 25.7%, 25.8%, 25.1%

Table 2
Classification accuracy MEAN(STD) on datasets "Musk" and "Yeast" from randomized five runs of fivefold cross-validation. Each entry here shows the statistics of the accuracy rate samples (i.e. 25 samples from 5-by-5 cross-validation) in form of MEAN(STD).

Dataset	Classifier	Dimension	DA methods				
			aPAC	HELDA	MeRMaID_SIG	PCA	MMILA
Musk	SVM-L	1	0.800(0.035)	0.662(0.067)	0.758(0.043)	0.565(0.003)	0.806(0.032)
		2	0.803(0.031)	0.709(0.051)	0.752(0.040)	0.565(0.003)	0.806(0.031)
		4	0.804(0.030)	0.747(0.034)	0.778(0.025)	0.626(0.050)	0.803(0.029)
		8	0.800(0.033)	0.759(0.040)	0.769(0.031)	0.680(0.044)	0.765(0.045)
		12	0.800(0.033)	0.766(0.038)	0.769(0.029)	0.724(0.034)	0.770(0.045)
	SVM-G	1	0.804(0.034)	0.697(0.042)	0.766(0.038)	0.595(0.032)	0.805(0.032)
		2	0.803(0.032)	0.754(0.037)	0.781(0.030)	0.647(0.034)	**0.835**(0.043)
		4	0.812(0.033)	0.792(0.044)	0.811(0.031)	0.684(0.037)	**0.872**(0.043)
		8	0.821(0.038)	0.800(0.042)	0.815(0.025)	0.768(0.045)	**0.884**(0.037)
		12	0.818(0.041)	0.795(0.037)	0.816(0.025)	0.874(0.029)	**0.889(**0.037)
	Parzen	1	0.800(0.033)	0.690(0.038)	0.763(0.036)	0.565(0.003)	**0.806**(0.032)
		2	0.802(0.029)	0.748(0.035)	0.771(0.036)	0.620(0.039)	0.823(0.042)
		4	0.810(0.033)	0.797(0.031)	0.806(0.029)	0.687(0.043)	0.860(0.048)
		8	0.818(0.035)	0.804(0.037)	0.808(0.025)	0.771(0.037)	0.878(0.039)
		12	0.818(0.038)	0.810(0.030)	0.806(0.028)	0.868(0.034)	0.883(0.039)
Yeast	SVM-L	1	0.419(0.017)	0.311(0.002)	0.453(0.031)	0.390(0.033)	0.418(0.021)
		2	0.524(0.026)	0.395(0.018)	0.516(0.033)	0.457(0.033)	0.523(0.034)
		3	0.548(0.030)	0.506(0.047)	0.532(0.018)	0.504(0.029)	0.549(0.031)
		4	0.567(0.023)	0.541(0.028)	0.540(0.018)	0.559(0.023)	0.560(0.026)
		5	0.579(0.023)	0.565(0.019)	0.544(0.020)	0.562(0.019)	0.565(0.030)
	SVM-G	1	**0.423**(0.019)	0.311(0.002)	0.472(0.027)	0.406(0.032)	0.417(0.019)
		2	0.535(0.022)	0.403(0.029)	0.523(0.031)	0.474(0.038)	**0.539**(0.031)
		3	0.556(0.029)	0.513(0.040)	0.542(0.014)	0.538(0.032)	**0.566**(0.030)
		4	0.572(0.026)	0.558(0.023)	0.548(0.017)	**0.577**(0.027)	0.574(0.026)
		5	0.592(0.025)	**0.583**(0.024)	0.555(0.019)	0.582(0.027)	0.578(0.027)
	Parzen	1	0.409(0.023)	0.311(0.002)	0.450(0.023)	0.381(0.022)	0.398(0.021)
		2	0.511(0.024)	0.395(0.022)	0.508(0.032)	0.435(0.028)	0.513(0.023)
		3	0.534(0.026)	0.491(0.041)	0.522(0.015)	0.456(0.024)	0.539(0.019)
		4	0.559(0.024)	0.531(0.025)	0.536(0.020)	0.508(0.019)	0.565(0.026)
		5	0.570(0.024)	0.554(0.024)	0.539(0.021)	0.537(0.030)	0.573(0.025)

Table 3
Classification accuracy MEAN(STD) on datasets "Glass" and "Vehicle" from randomized five runs of fivefold cross-validation. Each entry here shows the statistics of the accuracy rate samples (i.e. 25 samples from 5-by-5 cross-validation) in form of MEAN(STD).

Dataset	Classifier	Dimension	DA methods				
			aPAC	HELDA	MeRMaID_SIG	PCA	MMILA
Glass	SVM-L	1	0.555(0.066)	0.553(0.056)	0.436(0.080)	0.365(0.028)	0.544(0.077)
		2	0.568(0.088)	0.563(0.068)	0.546(0.070)	0.518(0.086)	0.556(0.063)
		3	0.617(0.082)	0.566(0.067)	0.576(0.055)	0.586(0.081)	0.589(0.070)
		4	0.621(0.077)	0.582(0.061)	0.581(0.056)	0.582(0.071)	0.596(0.084)
		5	0.620(0.084)	0.588(0.053)	0.582(0.058)	0.591(0.048)	0.617(0.079)
	SVM-G	1	**0.571**(0.070)	0.549(0.066)	0.496(0.061)	0.525(0.056)	0.559(0.067)
		2	0.574(0.086)	0.558(0.062)	0.576(0.053)	**0.590**(0.075)	0.578(0.054)
		3	0.627(0.052)	0.564(0.066)	0.604(0.044)	**0.661**(0.058)	0.645(0.056)
		4	0.627(0.061)	0.567(0.053)	0.614(0.049)	0.671(0.057)	**0.675**(0.068)
		5	0.624(0.062)	0.590(0.050)	0.609(0.044)	0.686(0.058)	**0.690**(0.061)
	Parzen	1	0.540(0.060)	0.547(0.054)	0.469(0.054)	0.418(0.088)	0.535(0.083)
		2	0.550(0.084)	0.524(0.045)	0.545(0.056)	0.548(0.101)	0.564(0.065)
		3	0.578(0.055)	0.521(0.042)	0.572(0.043)	0.604(0.092)	0.623(0.064)
		4	0.605(0.067)	0.548(0.062)	0.581(0.058)	0.625(0.095)	0.639(0.064)
		5	0.598(0.073)	0.556(0.062)	0.589(0.052)	0.628(0.070)	0.656(0.065)
Vehicle	SVM-L	1	0.573(0.036)	0.490(0.032)	0.542(0.026)	0.313(0.026)	**0.593**(0.025)
		2	0.730(0.021)	0.731(0.026)	0.694(0.047)	0.433(0.029)	0.733(0.023)
		3	0.745(0.029)	0.745(0.027)	0.717(0.028)	0.452(0.028)	0.734(0.021)
		4	0.746(0.026)	0.757(0.028)	0.723(0.033)	0.448(0.039)	0.755(0.035)
		5	0.752(0.028)	0.763(0.032)	0.725(0.031)	0.575(0.029)	0.760(0.023)
	SVM-G	1	0.581(0.027)	0.498(0.027)	0.560(0.039)	0.441(0.028)	0.591(0.028)
		2	0.739(0.022)	0.730(0.025)	0.716(0.037)	0.524(0.028)	0.744(0.025)
		3	0.752(0.026)	**0.766**(0.027)	0.737(0.026)	0.539(0.033)	0.764(0.026)
		4	0.771(0.027)	0.786(0.025)	0.748(0.023)	0.531(0.023)	**0.793**(0.030)
		5	0.782(0.026)	0.789(0.026)	0.759(0.026)	0.625(0.031)	**0.815**(0.020)
	Parzen	1	0.580(0.035)	0.498(0.031)	0.553(0.033)	0.332(0.033)	0.582(0.028)
		2	0.743(0.024)	0.723(0.030)	0.709(0.036)	0.490(0.033)	**0.748**(0.026)
		3	0.745(0.032)	0.751(0.023)	0.727(0.025)	0.510(0.028)	0.753(0.020)
		4	0.768(0.032)	0.771(0.024)	0.745(0.024)	0.529(0.029)	0.785(0.030)
		5	0.773(0.031)	0.770(0.026)	0.747(0.025)	0.605(0.028)	0.796(0.019)

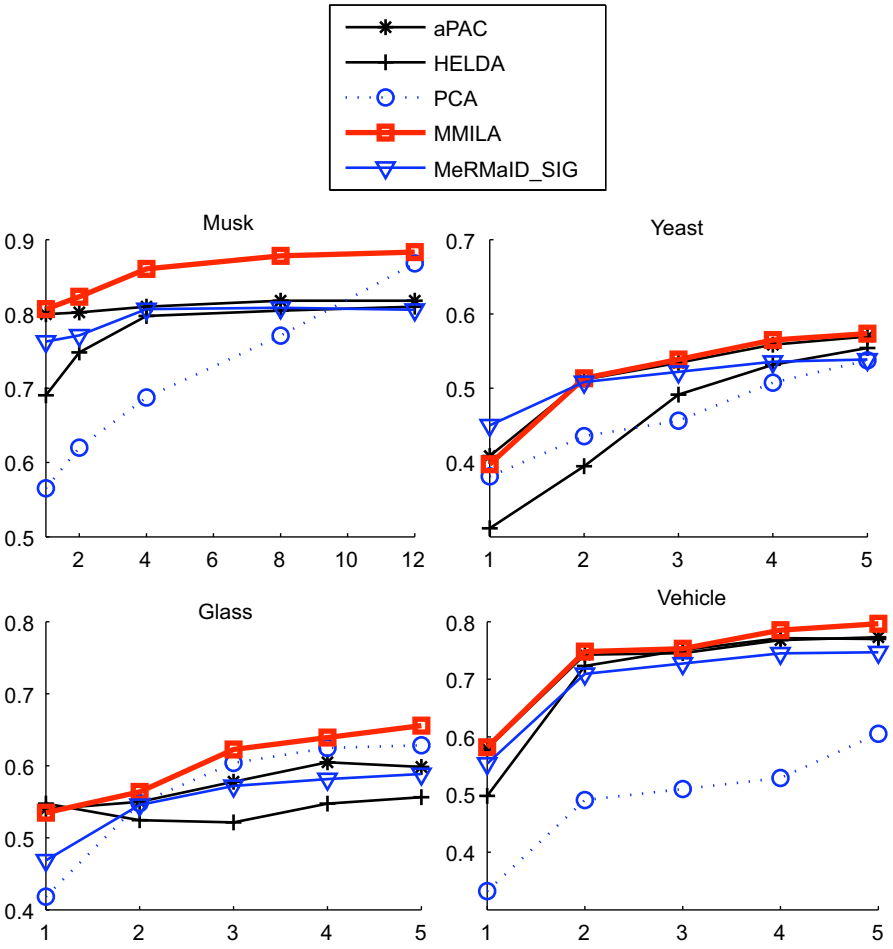

Fig. 3. Classification results using different feature extraction methods and a Parzen Window classifier (see Section 4.2). Here "MMILA" denotes the proposed method. The horizontal and the vertical axes denote the output dimension (i.e. number of features) and the classification accuracy, respectively.

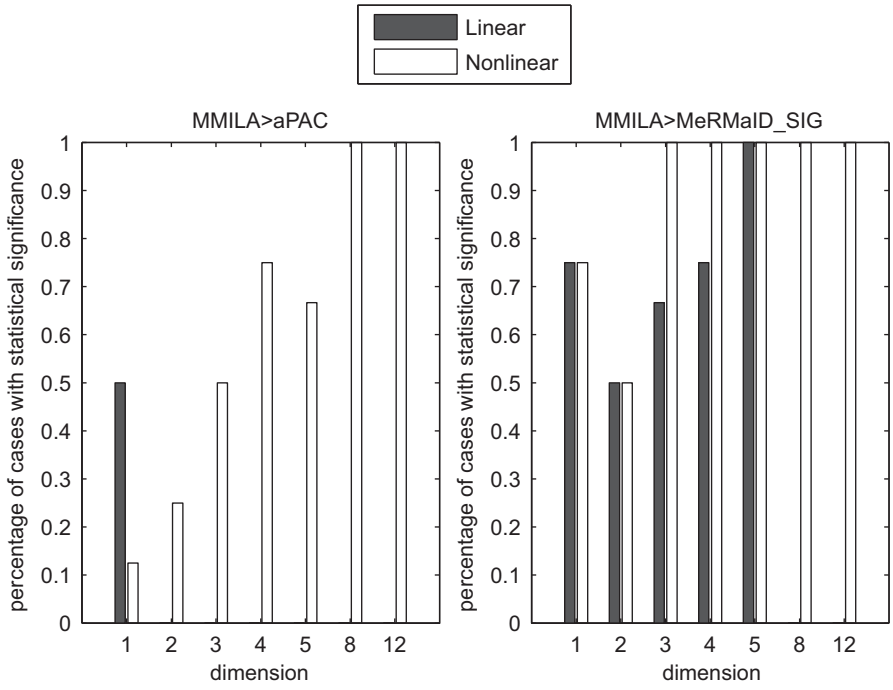

Fig. 4. Percentage of cases (see Section 4.2) in which the proposed method (MMILA) outperforms aPAC (left panel) or MeRMaID_SIG (right panel), with statistical significance (p-value < 0.05 in paired t-test). The black bars correspond to linear classification using SVM-L, while the blank bars to both nonlinear classifiers including SVM-G and Parzen.

5. Discussions

Result from the toy problem suggests that MMILA may produce fairly different linear transformation from those by existing methods. Particularly, it can better described the nonlinear separation between the two classes in hand. This may be attributed to the fact that the mutual information essentially measures nonlinear relationship between variables. Interestingly, this is also consistent with the finding in the real data classification results that MMILA tends to be more suitable for nonlinear classification than for linear classification.

As a generic feature extraction method, MMILA is also model-independent or hyper-parameter free, meaning that no assumption is made about the data structure. Therefore, the learning algorithm does not involve ad-hoc tuning of any hyper-parameters like learning ratio or kernel size. This also makes the implementation and numerical study straightforward.

It is worthwhile to emphasize that MMILA is a supervised learning method that utilizes the class labels of the data. Generally, it would lead to better classification results than unsupervised learning such as PCA, or its various extensions [29,27,30–32]. This can be verified in the classification results presented in Fig. 3, Tables 2 and 3.

Further development of the proposed method is intriguing. For example, we may look into more effective optimization procedure than the deterministic gradient-based optimization, since the deterministic gradient optimization can be prone to slow or premature convergence [33]. Therefore, it is interesting to investigate if global or stochastic optimization methods (such as [34]) can further improve MMILA.

Besides, while the present work was focused on linear analysis or linear feature extraction that is certainly useful for both pattern classification applications or visualization, its nonlinear extension can be interesting. An inspiring work was reported in [35], which performed mutual information-based learning (similar to [19]) in only the linear part of a radial basis function network. Besides, its extension to unsupervised learning and semi-supervised learning are also interesting. For example, active learning may be used to take into account the unlabelled samples [36]. Another possible direction, inspired by [37], would be to redesign the algorithm in a unified framework called graph embedding.

6. Conclusion

In this paper we have presented a MMI-based method for discriminative analysis or feature extraction. The method is based on a non-parametric mutual information estimate which measures the dependency between features and class labels. The kernel size for the mutual information estimate is taken as an intrinsic explicit function of the linear transformation matrix. And we have derived the expression of the gradient of the mutual information estimate with respect to the transformation matrix. We also devised a gradient-based learning algorithm using line search.

The toy problem study has indicated that, compared with existing methods, MMILA can produce fairly different linear transformations that better describe the nonlinear separation between the classes. We have also evaluated the method using four real world datasets from the UCI repository, while comparing it with existing DA methods including aPAC, HELDA and MeRMaID_SIG. The results demonstrated overall superiority of the proposed method (MMILA), which yielded best classification accuracy in 11 out of 15 cases. The results also indicate that the proposed method is more suitable for nonlinear classification than for linear classification.

References

[1] K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, New York, 1990.
[2] M. Last, A. Kander, O. Maimon, Information-theoretic algorithm for feature selection, Pattern Recognition Letters 22 (2001) 799–811.
[3] J.M. Sotoca, F. Pla, Supervised feature selection by clustering using conditional mutual information-based distances, Pattern Recognition 43 (2010) 2068–2081.
[4] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, K.-R. Muller, Fisher discriminant analysis with kernels, in: IEEE Conference on Neural Networks for Signal Processing, 1999.
[5] R.A. Fisher, The use of multiple measurements in taxonomic problems, Annals of Eugenics 7 (1936) 179–188.
[6] C.R. Rao, The utilization of multiple measurements in problems of biological classification, Journal of the Royal Statistical Society, Series B 10 (1948) 159–203.
[7] O.S. Hamsici, A.M. Martinez, Bayes optimality in linear discriminant analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 30 (2008) 647–657.
[8] A.M. Martinez, M. Zhu, Where are linear feature extraction methods applicable, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (2005) 1934–1944.
[9] S. Petridis, S.J. Perantonis, On the relation between discriminant analysis and mutual information for supervised linear feature extraction, Pattern Recognition 37 (2004) 857–874.
[10] K. Fukunaga, J.M. Mantock, Nonparametric discriminant analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 5 (1983) 671–678.
[11] M. Loog, R.P.W. Duin, R. Haeb-Umbach, Multiclass linear dimension reduction by weighted pairwise fisher criteria, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (2001) 762–766.
[12] R. Lotlikar, R. Kothari, Fractional-step dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000) 623–627.
[13] N. Kumar, A.G. Andreou, Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition, Speech Communication 26 (1998) 283–297.
[14] R.P.W. Duin, M. Loog, Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2004) 732–739.
[15] T.M. Cover, J.A. Thomas, Elements of Information Theory, second ed., Wiley, New York, 2006.
[16] M. Ben-Bassat, User of distance measures, in: P. Krishnaiah, L. Kanal (Eds.), Handbook of Statistics, North-Holland, Amsterdam, 1982, pp. 773–791.
[17] P.A. Estevez, M. Tesmer, C.A. Perez, J.M. Zurada, Normalized mutual information feature selection, IEEE Transactions on Neural Networks 20 (2009) 189–201.
[18] N. Kwak, C.-H. Choi, Input feature selection by mutual information based on Parzen window, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 1667–1671.
[19] K.E. Hild, D. Erdogmus, K. Torkkola, J.C. Principe, Feature extraction using information-theoretic learning, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (2006) 1385–1392.
[20] K.E. Hild, D. Erdogmus, J.C. Principe, An analysis of entropy estimators for blind source separation, Signal Processing 86 (2006) 182–194.
[21] D. Erdogmus, J.C. Principe, Generalized information potential criterion for adaptive system training, IEEE Transactions on Neural Networks 13 (2002) 1035–1044.
[22] P. Viola, W.M. Wells III, Alignment by maximization of mutual information, International Journal of Computer Vision 24 (1997) 137–154.
[23] D.W. Scott, Multivariate Density Estimation: Theory, Practice, and Visualization, Wiley, 1992.
[24] A.W. Bowman, A. Azzalini, Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations, Oxford University Press, New York, 1997.
[25] M. Loog, On the equivalence of linear dimensionality-reducing transformations, Journal of Machine Learning Research 9 (2009) 2489–2490.
[26] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, 2001. 〈http://www.csie.ntu.edu.tw/∼cjlin/libsvm〉.
[27] J. Li, X. Li, D. Tao, KPCA for semantic object extraction in images, Pattern Recognition 41 (2008) 3244–3250.
[28] Y. Xu, D. Zhang, J.-Y. Yang, A feature extraction method for use with bimodal biometrics, Pattern Recognition 43 (2010) 1106–1115.
[29] B. Schölkopf, A. Smola, K.R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation 10 (1998) 1299–1319.
[30] Y. Xu, D. Zhang, J.-Y. Yang, A feature extraction method for use with bimodal biometrics, Pattern Recognition 43 (2010) 1106–1115.
[31] X. Li, Y. Pang, Deterministic column-based matrix decomposition, IEEE Transactions on Knowledge and Data Engineering 22 (2010) 145–149.
[32] Y. Yuan, X. Li, Y. Pang, X. Lu, D. Tao, Binary sparse nonnegative matrix factorization, IEEE Transactions on Circuits and Systems for Video Technology 19 (2009) 772–777.
[33] S. Amari, Natural gradient works efficiently in learning, Neural Computation 10 (1998) 251–276.

[34] M. Clerc, J. Kennedy, The particle swarm-explosion, stability, and convergence in a multidimensional complex space, IEEE Transactions on Evolutionary Computing 6 (2002) 58–73.
[35] K. Torkkola, Feature extraction by non-parametric mutual information maximization, Journal of Machine Learning Research 3 (2003) 1415–1438.
[36] X. He, Laplacian regularized d-optimal design for active learning and its application to image retrieval, IEEE Transactions on Image Processing 19 (2010) 254–263.
[37] Y. Yuan, Y. Pang, Discriminant adaptive edge weights for graph embedding, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2008.

Haihong Zhang received the B.E. degree in electronic engineering in 1997, and the M.E. degree in electronics and systems in 2000, respectively, from Hefei University of Technology, China, and University of Science and Technology of China. In 2005, he received the Ph.D. degree in computer science from National University of Singapore in 2005. He joined the Brain–Computer Interface Laboratory as a research fellow at Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR) in Singapore in 2004. His research interests include pattern recognition, brain–computer interface, EEG signal processing.

Cuntai Guan received the Ph.D. degree in electrical and electronic engineering from Southeast University, Nanjing, China, in 1993. From 1993 to 1994, he was at the Southeast University, where he was engaged in speech vocoder, speech recognition, and text-to-speech. During 1995, he was a Visiting Scientist at the Centre de Recherche en Informatique de Nancy (CRIN)/Centre National de la Recherche Scientique (CNRS) Institut National de Recherche en Informatique et en Automatique (INRIA), Paris, France, where he was involved in keyword spotting. From 1996 to 1997, he was with the City University of Hong Kong, Kowloon, Hong Kong, where he was engaged in developing robust speech recognition under noisy environment. From 1997 to 1999, he was with the Kent Ridge Digital Laboratories, Singapore, Singapore, where he was involved in multilingual, large vocabulary, continuous speech recognition. He was a Research Manager and the R&D Director for five years in industries, focusing on the development of spoken dialogue technologies. In 2003, he established the Brain–Computer Interface Group at the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore, where he is currently a Senior Scientist and a Program Manager. His current research interests include brain–computer interface, neural signal processing, machine learning, pattern classification, and statistical signal processing, with applications to assistive device, rehabilitation, and health monitoring.

Yuanqing Li received the B.S. degree in applied mathematics from Wuhan University, Wuhan, China, in 1988, and the M.S. degree in applied mathematics and the Ph.D. degree in control theory and applications from South China Normal University, Guangzhou, China, in 1994 and 1997, respectively. Since 1997, he has been with South China University of Technology, where he became a Full Professor in 2004. During 2002–2004, he was a Research Fellow at the Laboratory for Advanced Brain Signal Processing, RIKEN Brain Science Institute, Saitama, Japan. During 2004–2008, he was a Research Scientist at the Laboratory for Neural Signal Processing, Institute for Infocomm Research, Singapore. His current research interests include blind signal processing, sparse representation, machine learning, brain–computer interface, EEG, and fMRI data analysis. He is the author or coauthor of more than 60 scientific papers in journals and conference proceedings.