Available online at www.sciencedirect.com
SCIENCE DIRECT
Pattern Recognition 39 (2006) 1241 – 1252
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/patcog

Spectral feature projections that maximize Shannon mutual information with class labels

Umut Ozertem[a,*], Deniz Erdogmus[a], Robert Jenssen[b]

[a]CSEE Department, Oregon Graduate Institute, OHSU, 20000 NW Walker Road, Beaverton, Portland, OR 97006, USA
[b]Department of Physics, University of Tromso, N-9037, Tromso, Norway

Received 5 August 2005; received in revised form 23 November 2005; accepted 17 January 2006

Abstract

Determining optimal subspace projections that can maintain task-relevant information in the data is an important problem in machine learning and pattern recognition. In this paper, we propose a nonparametric nonlinear subspace projection technique that maintains class separability maximally under the Shannon mutual information (MI) criterion. Employing kernel density estimates for nonparametric estimation of MI makes possible an interesting marriage of kernel density estimation-based information theoretic methods and kernel machines, which have the ability to determine nonparametric nonlinear solutions for difficult problems in machine learning. Significant computational savings are achieved by translating the definition of the desired projection into the kernel-induced feature space, which leads to obtain analytical solution.

Keywords: Feature extraction; Mutual information; Optimal subspace projection

1. Introduction

Dimensionality reduction is an important step in a variety of applications including pattern recognition, data compression, and exploratory data analysis. In practice, the relevant information about the data structure can often be represented by a lower dimensional manifold embedded in the original Euclidian data space. Specifically, in pattern recognition, a high-dimensional feature vector is available, but usually the classification task can be achieved equally well by a feature vector of reduced dimensionality. In practice, reducing the number of features will also help the classifier learn a more robust solution and achieve a better generalization performance. This is due to the fact that irrelevant feature components are eliminated by the *optimal* subspace projection. Recent developments in kernel machines indicate that robust solutions to nonlinear problems in pattern recognition can be obtained by first projecting the data into a higher dimensional space (possibly infinite). The regularization of the solution is achieved by the proper selection of the kernel. In this paper, we develop a technique based on kernel mutual information (MI) estimation for finding nonlinear projections by first projecting the data to such a space and then projecting it down to a much lower dimensionality.

Subspace projection is typically achieved either by feature selection or by feature transformation. Optimal feature selection coupled with a specific classifier topology, namely the wrapper approach, results in a combinatorial computational requirement, thus, is unsuitable for adaptive learning of feature projections. In addition, feature selection is a special case of feature transformations; therefore, we will focus on the general case of determining optimal nonlinear transformations.

Adaptive learning of nonlinear feature transformations, namely the filter approach, is achieved by optimizing a suitable criterion. The possibility of learning the optimal feature projections sequentially decreases the computational requirements making the filter approach especially attractive. Perhaps, historically the first dimensionality reduction

* Corresponding author. Tel.: +1 503 7481564; fax: +1 503 7481548.
E-mail address: ozertemu@csee.ogi.edu (U. Ozertem).

0031-3203/$30.00
doi:10.1016/j.patcog.2006.01.014

technique is linear principle components analysis (PCA) [1,2]. Although this technique is widely used, its shortcomings for pattern recognition are well known. A generalization to nonlinear projections, Kernel PCA [3], still exhibits the same shortcoming; the projected features are not necessarily useful for classification. Another unsupervised (i.e., ignorant of class labels) projection method is independent component analysis (ICA), a modification of the uncorrelatedness condition in PCA to independence, in order to account for higher order statistical dependencies in non-Gaussian distributions [4]. Besides statistical independence, source sparsity and nonnegativity are also utilized as a statistical assumption in achieving dimensionality reduction through sparse bases, a technique called nonnegative matrix factorization (NMF) [5]. These methods, however, are linear and restricted in their ability to generate versatile projections for curved data distributions. Local linear projections is an obvious method to achieve globally nonlinear yet locally linear dimensionality reduction. One such method that aims to achieve dimensionality reduction while preserving neighborhood topologies is local linear embedding (LLE) [6]. Extensions of this approach to supervised local linear embeddings that consider class label information also exist [7].

Linear discriminant analysis (LDA) attempts to eliminate the shortcoming of PCA by finding linear projections that maximize class separability under the Gaussian distribution assumption [8]. The LDA projections are optimized based on the means and covariance matrix of each class, which are not descriptive of an arbitrary probability density function (pdf). In addition, only linear projections are considered. Kernel LDA [9], generalizes this principle to finding nonlinear projections under the assumption that the kernel function induces a nonlinear transformation (dependent on the eigenfunctions of the kernel) that first projects the data to a hypothetical high-dimensional space where the Gaussianity assumption is satisfied. However, the kernel functions used in practice do not necessarily guarantee the validity of this assumption.

Traditionally, second-order statistical methods have found widespread application in adaptive signal processing, machine learning, and pattern recognition, as we can observe from the literature easily. In more contemporary approaches, many researchers have realized the importance of exploiting the additional freedom that nonlinear systems give over convenient linear systems (such as linear projections in the feature subspace projection context). In addition, the insufficiency of mere second-order statistics in many application areas have been discovered and more advanced concepts including higher-order statistics, especially those stemming from information theory are now being studied and applied in many contexts by researchers in machine learning and signal processing. The value of information theoretic approaches combined with nonlinear topologies have been demonstrated in many applications, Torkkola's recent work on quadratic MI-based linear projections, which is built on early work on information theoretic learning [10] is one of the most prominent [11]. Unfortunately, despite these recent advances in the understanding of the role of nonlinear topologies and information theoretic concepts in pattern recognition, the use of traditional second-order statistical linear rules such as PCA and LDA (or their variations) continue to find widespread use possibly due to the delay in the dissemination of recent results in information theoretic learning in the scientific community.

In the filter approach, it is important to optimize a criterion that is relevant to Bayes risk, which is typically measured by the probability of error. A suitable criterion is MI between the projected features and the class labels, which is motivated by lower and upper bounds in information theory that relate this quantity to probability of error. In principle, MI measures nonlinear dependencies between a set of random variables taking into account higher order statistical structures existing in the data, as opposed to linear and second-order statistical measures such as correlation and covariance [12].

Evaluating the MI between two scalar random variables (one being the discrete class labels) is relatively easy as compared to estimating it for random vectors. Consequently, MI-based feature selection is widely recognized as a powerful method in the literature [13–16]. Since features are generally mutually dependent, feature selection in this manner is typically suboptimal in the sense of maximum MI.

MI is defined in terms of the probability density of the data; hence, requires a pdf estimate. Since the data pdf might take complex forms, in practice, in many applications determining a suitable parametric family becomes a nontrivial task. Therefore, MI should be estimated nonparametrically from the training samples [17,18]. Although this is a challenging problem for two continuous-valued random vectors, in the feature transformation setting the class labels are discrete-valued. This reduces the problem to simply estimating entropies of continuous random vectors. The multi-dimensional entropy can be estimated nonparametrically using a number of techniques. Entropy estimators based on sample spacing, such as the minimum spanning tree, are not differentiable making them unsuitable for adaptive learning of feature projections [18–22]. On the other hand, entropy estimators based on kernel density estimation (KDE) provide a differentiable alternative [17,22,23]. Torkkola recently proposed utilizing a quadratic MI measure, estimated using KDE [10], to determine optimal linear feature projections [11].

In this paper, we propose a method for determining optimal nonlinear feature projections that maximize the Shannon MI between the projections and the class labels. Nonparametric entropy estimation using KDE results in $O(N^2)$ complexity, where N is the number of training samples. Therefore, gradient-based adaptation is computationally prohibitive for large training sets, especially with local optima problems in training nonlinear topologies. We propose to avoid this complication by exploiting the

kernel-induced feature (KIF) transformation to obtain an analytical solution for the optimal nonlinear multi-dimensional projections that can be expressed in terms of the eigenvectors and eigenvalues of the *kernel matrix*.

2. Theoretical background

The goal of feature subspace projections is to improve classifier robustness by reducing data dimensionality in order to facilitate better generalization, as well as reducing the learning and operating complexity of the classifiers. While doing so, classification performance must not be compromised by throwing away components that provide useful information regarding the class labels. Theoretically, optimal feature projections should minimize the Bayes risk function for the given problem; the average probability of error is a widely used and accepted risk function and merits special attention.[1]

The average probability of error has been shown to be related to MI between the feature vectors and the class labels. Specifically, Fano's, Hellman and Raviv's bounds demonstrate that probability of error is bounded from below and above by quantities that depend on the Shannon MI between these variables [24,25]. Maximizing this MI reduces both bounds, therefore, forces the probability of error to decrease. A similar result was also obtained by Erdogmus and Principe using Renyi's MI; a parametric family of lower and upper bounds for the probability of error was provided [17,26]. Specifically, Hellman and Raviv showed that the probability of error for a C-class problem is bounded by $P(error) \leqslant (H_S(c) - I_S(\mathbf{y}, c))/2$, where $H_S(c)$ is the Shannon entropy of the a priori probabilities of the classes and $I_S(\mathbf{y}, c)$ is the Shannon MI between the continuous-valued feature vectors and the discrete-valued class labels. Consequently, maximizing the MI between the projected features and the class labels potentially improves classification performance, and therefore, has drawn much attention [11,14,15,27].

MI was first introduced by Shannon in the context of digital communications between discrete random variables and was generalized to continuous random variables. In feature extraction, we are interested in the MI between the continuous-valued feature vector $\mathbf{y}$ and the discrete-valued class labels c. Shannon MI between $\mathbf{y}$ and c is defined in terms of the entropies of the overall data and the individual classes as [12]

$$I_S(\mathbf{y}; c) = H_S(\mathbf{y}) - \sum_c p_c H_S(\mathbf{y}|c), \tag{1}$$

[1] For different risk functions, the following results can easily be modified.

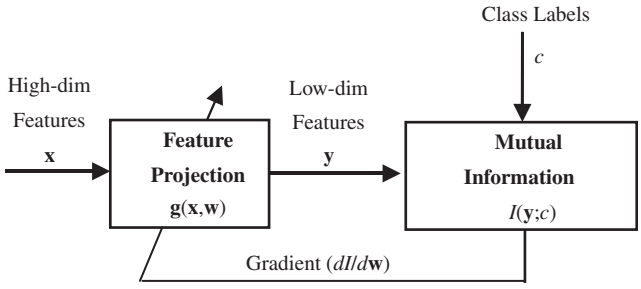

Fig. 1. Determining optimal feature subspace projections using mutual information.

where p_c are the prior class probabilities. The Shannon entropy is given by

$$H_S(\mathbf{y}) = -\int p(\mathbf{y}) \log\ p(\mathbf{y})\, \mathrm{d}\mathbf{y},$$
$$H_S(\mathbf{y}|c) = -\int p(\mathbf{y}|c) \log p(\mathbf{y}|c)\, \mathrm{d}\mathbf{y}, \tag{2}$$

where $p(\mathbf{y}|c)$ are the class conditional distributions and the overall data distribution is

$$p(\mathbf{y}) = \sum_c p_c p(\mathbf{y}|c). \tag{3}$$

Under the framework of *optimal feature subspace projections that maximize MI with class labels*, the adaptive learning procedure to find these optimal projections follows the block diagram shown in Fig. 1. In the most general case, a high-dimensional feature vector is projected to a lower dimensional vector by a nonlinear parametric function (such as a neural network), whose weights (denoted by $\mathbf{w}$) are optimized to maximize the MI criterion [10,11,27]. Since learning rules based on MI measures are typically computationally intensive, nonlinear projections are avoided and one resorts to linear projections of the form $\mathbf{y} = \mathbf{W}\mathbf{x}$ [11,27].

As seen in Eq. (1), in order to estimate MI we need to estimate the conditional class entropies as well as the overall data entropy. As mentioned earlier, entropy estimators based on sample spacing are not suitable for gradient-based adaptation. A feasible alternative is the KDE-based plug-in estimator [17,22,23]. Given a set of independent and identically distributed (iid) samples $\{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$, which can be partitioned into subsets corresponding to each class as $\{\mathbf{y}_1^c, \ldots, \mathbf{y}_{N_c}^c\}$, the entropies in Eq. (1) can be estimated by [22]

$$H_S(\mathbf{y}) = -\frac{1}{N}\sum_{j=1}^{N} \log \frac{1}{N}\sum_{i=1}^{N} K(\mathbf{y}_j - \mathbf{y}_i),$$
$$H_S(\mathbf{y}|c) = -\frac{1}{N_c}\sum_{j=1}^{N_c} \log \frac{1}{N_c}\sum_{i=1}^{N_c} K(\mathbf{y}_j^c - \mathbf{y}_i^c). \tag{4}$$

Clearly, optimizing a nonlinear topology to maximize (1) using the estimators in Eq. (4) will be computationally expensive as N increases. In the next section, we propose a

nonparametric nonlinear topology that stems from the theory of reproducing kernels in Hilbert spaces.

3. Spectral transformations and maximally separable projections

We are given a set of features $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and their corresponding class labels $\{c_1, c_2, \ldots, c_N\}$. The number of samples in each class is denoted by N_c and the total number of classes is C. We are interested in finding a nonlinear subspace projection $\mathbf{y} = g(\mathbf{x})$ such that Shannon MI between the projection and the class labels, i.e. $I_S(\mathbf{y}, c)$, is maximized.

According to the theory of reproducing kernels for Hilbert spaces (RKHS), the eigenfunctions $\{\bar{\varphi}_1(\mathbf{x}), \bar{\varphi}_2(\mathbf{x}), \ldots\}$ collected in vector notation as $\bar{\boldsymbol{\varphi}}(\mathbf{x})$, of a kernel function K that satisfy the Mercer conditions [28] form a basis for the Hilbert space of finite power nonlinear functions [29,30].[2] Therefore, every finite-L_2-norm nonlinear transformation $g_d(\mathbf{x})$ can be expressed as a linear combination of these bases:

$$y_d = g_d(\mathbf{x}) = \mathbf{v}_d^{\mathrm{T}} \bar{\boldsymbol{\varphi}}(\mathbf{x}), \tag{5}$$

where y_d is the dth component of the projection vector $\mathbf{y}$. As we will show next, such linear combinations of nonlinear basis functions arise naturally from the KDE-based nonparametric estimates of MI in the context of feature subspace projections. Having the KDE-based MI estimate, one can define the projection in the KIFS to obtain an objective function that is to be optimized. Interestingly, the optimizer of this objective function is given by an analytical solution; hence, no optimization method is required.

To grasp an intuition of the approach before the detailed derivation, one should first consider the characteristics of the data mapping into KIFS. KIFS is a potentially infinite dimensional space and the *kernel trick* defines a transformation from the original data space to a hyper-sphere in this space. For a symmetric translation invariant nonnegative (since we will connect to density estimation later) kernel, we can write

$$K(\mathbf{x} - \mathbf{x}') = \sum_{k=1}^{\infty} \bar{\lambda}_k \bar{\varphi}_k(\mathbf{x}) \bar{\varphi}_k(\mathbf{x}') = \bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}) \bar{\boldsymbol{\Lambda}} \bar{\boldsymbol{\varphi}}(\mathbf{x}') \geqslant 0. \tag{6}$$

Notice that for a nonnegative kernel, KIFS transformation maps all the data points into the same half of this hyper-sphere; i.e., the *angles* between all transformed data pairs are less than π radians.

In the following, we will demonstrate how the nonlinear projections in Eq. (5) can be optimally determined for maximal MI. The determination of the optimal solution is much easier in the KIFS, therefore, we will start with the kernel MI estimator in the original data space and employ the kernel trick in Eq. (6) to express the problem equivalently in the KIFS. Following the nonlinear projection to the very high-dimensional kernel space, we will perform a subspace projection to determine the low-dimensional overall nonlinear projections.

3.1. Estimating the Shannon mutual information nonparametrically using kernel density estimates

Consider the Shannon MI between the high-dimensional original feature vectors and the class labels,

$$\begin{aligned} I_S(\mathbf{x}; c) &= \sum_c \int p_{\mathbf{x}c}(\mathbf{x}, c) \log \frac{p_{\mathbf{x}c}(\mathbf{x}, c)}{p_{\mathbf{x}}(\mathbf{x}) p_c} \, \mathrm{d}\mathbf{x} \\ &= \sum_c \int p_{\mathbf{x}|c}(\mathbf{x}|c) p_c \log \frac{p_{\mathbf{x}|c}(\mathbf{x}|c) p_c}{p_{\mathbf{x}}(\mathbf{x}) p_c} \, \mathrm{d}\mathbf{x} \\ &= \sum_c p_c \int p_{\mathbf{x}|c}(\mathbf{x}|c) \log \frac{p_{\mathbf{x}|c}(\mathbf{x}|c)}{p_{\mathbf{x}}(\mathbf{x})} \, \mathrm{d}\mathbf{x} \\ &= \sum_c p_c E_{\mathbf{x}|c} \left[\log \frac{p_{\mathbf{x}|c}(\mathbf{x}|c)}{p_{\mathbf{x}}(\mathbf{x})} \right]. \end{aligned} \tag{7}$$

The pdfs $p_{\mathbf{x}|c}$ and $p_{\mathbf{x}}$ in Eq. (7) are estimated using KDE with $K(.)$ as the kernel. The conditional expectation can be approximated by a sample mean over the appropriate samples.[3] This leads to

$$\begin{aligned} I_S(\mathbf{x}; c) &\approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \frac{p_{\mathbf{x}|c}(\mathbf{x}_j^c|c)}{p_{\mathbf{x}}(\mathbf{x}_j^c)} \\ &\approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \frac{(1/N_c) \sum_{i=1}^{N_c} K(\mathbf{x}_j^c - \mathbf{x}_i^c)}{(1/N) \sum_{i=1}^{N} K(\mathbf{x}_j^c - \mathbf{x}_i)}. \end{aligned} \tag{8}$$

Assuming that K is a reproducing kernel with an eigendecomposition as in Eq. (6), the MI estimate becomes

$$I_S(\mathbf{x}; c) \approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \left[\frac{N \bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}_j^c) \bar{\boldsymbol{\Lambda}} \bar{\boldsymbol{\Phi}}_{\mathbf{x}} \mathbf{m}_c}{N_c \bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}_j^c) \bar{\boldsymbol{\Lambda}} \bar{\boldsymbol{\Phi}}_{\mathbf{x}} \mathbf{1}} \right], \tag{9}$$

where we define the membership vector $\mathbf{m}_c$ for each class c, such that $\mathbf{m}_{ci} = 1$ if $c_i = c$, 0 otherwise, and the vectors $\mathbf{e}_i$ whose ith entry is 1 and all others are zeros, as well as a vector of ones, denoted by $\mathbf{1}$. The class priors are estimated using sample counts from the training data, i.e., $p_c = N_c/N$. In addition, we introduced the matrix $\bar{\boldsymbol{\Phi}}_{\mathbf{x}} = [\bar{\boldsymbol{\varphi}}(\mathbf{x}_1) \ldots \bar{\boldsymbol{\varphi}}(\mathbf{x}_N)]$, where $N = N_1 + \cdots + N_C$. Defining the average vectors of the transformed features for each class and for the whole training set as $\bar{\boldsymbol{\mu}}_c = (1/N_c) \bar{\boldsymbol{\Phi}}_{\mathbf{x}} \mathbf{m}_c$ (for the feature vectors from class c) and $\bar{\boldsymbol{\mu}} = (1/N) \bar{\boldsymbol{\Phi}}_{\mathbf{x}} \mathbf{1}$ (for the whole data set), we equivalently obtain:

$$I_S(\mathbf{x}; c) \approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \left[\frac{\bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}_j) \bar{\boldsymbol{\Lambda}} \bar{\boldsymbol{\mu}}_c}{\bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}_j) \bar{\boldsymbol{\Lambda}} \bar{\boldsymbol{\mu}}} \right]. \tag{10}$$

[2] The true eigenfunctions and eigenvalues of the reproducing kernel will be denoted using variables with a bar. This will help to distinguish quantities related to the continuous kernel function from the equivalent quantities related to the kernel matrix.

[3] Note that this estimation technique maintains certain consistency requirements, such as the Bayes relationship between the estimated overall density $p(\mathbf{x})$ and the class-conditional densities $p(\mathbf{x}|c)$.

Note that so far we have only utilized the true eigenfunctions and the eigenvalues of the kernel function; however, the true eigenfunctions cannot be obtained analytically in general. In the next subsection we will estimate these eigenfunctions, and rewrite the projection scheme given in Eq. (5) using these approximations.

3.2. Spectral transformations that the maximize Shannon mutual information in the kernel-induced feature space

According to the projection model in Eq. (5), the projection is accomplished in the kernel-induced $\boldsymbol{\varphi}$-space. If the target reduced dimensionality is D, we have $\mathbf{y} = \mathbf{V}^{\mathrm{T}}\bar{\boldsymbol{\varphi}}(\mathbf{x})$, where $\mathbf{V} = [\mathbf{v}_1 \dots \mathbf{v}_D]$ consists of orthonormal columns $\mathbf{v}_d$. The normality constraint helps reduce redundancy in the representation of these nonlinear projections since the scale of the projection does not carry information relevant to classification (samples from all classes are scaled), and orthogonality ensures efficient subspace representation and uncorrelated projections. The back-projection of $\mathbf{y}$ to the KIFS is given by

$$\bar{\boldsymbol{\varphi}}(\mathbf{y}) = \mathbf{V}\mathbf{V}^{\mathrm{T}}\bar{\boldsymbol{\varphi}}(\mathbf{x}). \tag{11}$$

This leads to the following cost function that needs to be maximized by optimizing $\mathbf{V}$:

$$J(\mathbf{V}) = \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log\left[\frac{\bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}_j)\mathbf{V}\mathbf{V}^{\mathrm{T}}\bar{\boldsymbol{\Lambda}}\mathbf{V}\mathbf{V}^{\mathrm{T}}\bar{\boldsymbol{\mu}}_c}{\bar{\boldsymbol{\varphi}}^{\mathrm{T}}(\mathbf{x}_j)\mathbf{V}\mathbf{V}^{\mathrm{T}}\bar{\boldsymbol{\Lambda}}\mathbf{V}\mathbf{V}^{\mathrm{T}}\bar{\boldsymbol{\mu}}}\right]. \tag{12}$$

In practice, analytical expressions for the (infinitely many) eigenfunctions of the kernel function are not available. Therefore, these must be approximated using the available training samples. Spectral methods provide the necessary tools to achieve this. Following the common procedure in spectral methods, and using all training samples in pairs as $\mathbf{K}_{ij} = K(\mathbf{x}_i - \mathbf{x}_j)$, we define the symmetric kernel matrix $\mathbf{K}$ (also called the affinity matrix). The matrix $\mathbf{K}$ can be decomposed into its eigenvalues and eigenvectors as $\mathbf{K} = \boldsymbol{\Phi}_{\mathbf{x}}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{\Phi}_{\mathbf{x}}$, which are essentially approximations of the sought eigenfunctions and eigenvalues of the kernel function. Specifically, according to the Nystrom routine [31], the eigenfunctions can be approximated using the eigendecomposition of the affinity matrix $\mathbf{K}$ as follows:

$$\bar{\boldsymbol{\varphi}}(\mathbf{x}) \approx \boldsymbol{\varphi}(\mathbf{x}) = \sqrt{N}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{k}(\mathbf{x}), \tag{13}$$

where $\mathbf{k}(\mathbf{x}) = [K(\mathbf{x} - \mathbf{x}_1), \dots, K(\mathbf{x} - \mathbf{x}_N)]^{\mathrm{T}}$. With this substitution, the nonlinear feature transformations become $\mathbf{y} = \mathbf{V}^{\mathrm{T}}\boldsymbol{\varphi}(\mathbf{x})$ and the approximation for the criterion in Eq. (12) becomes

$$J(\mathbf{V}) = \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log\left[\frac{\boldsymbol{\varphi}^{\mathrm{T}}(\mathbf{x}_j)\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\mu}_c}{\boldsymbol{\varphi}^{\mathrm{T}}(\mathbf{x}_j)\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{V}\mathbf{V}^{\mathrm{T}}\boldsymbol{\mu}}\right], \tag{14}$$

where $\boldsymbol{\mu}_c = (1/N_c)\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{m}_c$ and $\boldsymbol{\mu} = (1/N)\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{1}$ are the class and overall mean vectors of the data in the $\boldsymbol{\varphi}$-space. Note that $\boldsymbol{\mu} = p_1\boldsymbol{\mu}_1 + \cdots + p_C\boldsymbol{\mu}_C$.

3.3. Analytical solution for C-1 and lower dimensional projections

Observing the numerator and the denominator of the argument of the logarithm in Eq. (14), one can notice that this criterion can be maximized by selecting $\mathbf{V}$ such that its columns span the intersection of the subspace orthogonal to the mean vector $\boldsymbol{\mu}$, and the subspace spanned by the set of class mean vectors $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_C\}$.[4] The subspace spanned by the columns of $\mathbf{V}$, by construction, can be at most C-1 dimensional and is uniquely defined by the class structure of the data. In fact, all the lower dimensional optimal subspace projections are also contained in this subspace, and the analytical solution for these projections can be easily determined as we will show next.

A very important observation is that the class mean vectors in the KIFS are orthogonal to each other with their individual norms equal to $p_c^{-1/2}$, p_c being the class prior probability. We introduce the following matrix consisting of the class mean vectors in its columns:

$$\mathbf{M} = [\boldsymbol{\mu}_1 \dots \boldsymbol{\mu}_C], \tag{15}$$

where $\mathbf{M}$ satisfies $\mathbf{M}^{\mathrm{T}}\mathbf{M} = \mathbf{P}^{-1}$ (see Appendix B), with $\mathbf{p} = [p_1, \dots, p_C]$ and $\mathbf{P} = \mathrm{diag}(\mathbf{p})$. The overall data mean vector is then $\boldsymbol{\mu} = \mathbf{M}\mathbf{p}$. These identities easily lead to the conclusion that $\boldsymbol{\mu}$ is unit-norm. The columns of the matrix $\mathbf{V}$ defined below spans the desired solution subspace:

$$\mathbf{V} = \mathbf{M} - \boldsymbol{\mu}(\boldsymbol{\mu}^{\mathrm{T}}\mathbf{M}) = \mathbf{M} - \boldsymbol{\mu}(\mathbf{p}^{\mathrm{T}}\mathbf{M}^{\mathrm{T}}\mathbf{M}) = \mathbf{M} - \boldsymbol{\mu}\mathbf{1}^{\mathrm{T}}, \tag{16}$$

where $\mathbf{1}$ denotes a vector of ones.

3.4. Algorithm for determining optimal projections to fewer dimensions

In this section, we generalize the intuition developed in the previous section about determining the optimal projections by finding orthogonal directions to the mean vector $\boldsymbol{\mu}$. To this end, a procedure based on Gram–Schmidt orthogonalization will be employed. Note that the deflation will be implemented through the class mean vectors $\boldsymbol{\mu}_c$, therefore, the computational complexity of this algorithm is relatively low.

We start by constructing the matrix $\mathbf{M} = [\boldsymbol{\mu}_1 \dots \boldsymbol{\mu}_C]$. Consequently, all columns lie in one half of the vector space. This matrix is renamed as $\mathbf{M}^C$ to denote that its column rank is C. We introduce the sign vector $\mathbf{s}^C = [1, \dots, 1]^{\mathrm{T}}$ for reasons that will become clear shortly. Using the elementwise multiplication operator $\cdot$, we calculate $\mathbf{r}^C = \mathbf{s}^C \cdot \mathbf{p}$. The overall mean vector $\boldsymbol{\mu}^C$ is then given by $\boldsymbol{\mu}^C = \mathbf{M}^C\mathbf{r}^C$.

[4] Since $\boldsymbol{\mu}$ also lies in the span of the set $\{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_C\}$, and is not equal to any of the members of this set for nonzero prior probabilities, choosing $\mathbf{V}$ in the suggested manner will lead the numerator to remain always finite.

Table 1
The overall algorithm

Outline of the algorithm:
- Given a set of training data $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and their corresponding class labels $\{c_1, c_2, \ldots, c_N\}$, determine the kernel size (for Gaussian kernels according to Silverman's rule of thumb):

$$\sigma^2 = \frac{1}{n}\mathrm{tr}(\boldsymbol{\Sigma}_{\mathbf{x}})(4/((2n+1)N))^{2/(n+4)}$$

- Construct the kernel matrix $\mathbf{K}$, where $\mathbf{K}_{ij} = K(\mathbf{x}_i - \mathbf{x}_j)$
- Decompose $\mathbf{K}$ into its eigenvectors and eigenvalues such that $\mathbf{K} = \boldsymbol{\Phi}_{\mathbf{x}}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{\Phi}_{\mathbf{x}}$
- For the training data, calculate the kernel induced feature transformations as follows:

$$\boldsymbol{\varphi}(\mathbf{x}_j) = \sqrt{N}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{k}(\mathbf{x}_j)$$

- Determine the class means and the overall mean using $\boldsymbol{\mu}_c = (1/N_c)\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{m}_c$ and $\boldsymbol{\mu} = (1/N)\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{1}$
- Perform the following deflation procedure until the desired projection dimensionality is reached:
  1. Set $\mathbf{s}^{C\text{-}d} = [1, \ldots, 1]^{\mathrm{T}}$ in the first step, or according to Appendix B in the following steps
  2. Calculate $\mathbf{r}^{C\text{-}d} = \mathbf{s}^{C\text{-}d} \cdot \mathbf{p}$ and determine the *new* overall mean vector $\boldsymbol{\mu}^{C\text{-}d}$ by $\boldsymbol{\mu}^{C\text{-}d} = \mathbf{M}^{C\text{-}d}\mathbf{r}^{C\text{-}d}$ (The symbol $\cdot$ denotes elementwise vector product.)
  3. Construct the matrix $\mathbf{M}^{C\text{-}d} = [\boldsymbol{\mu}_1^{C\text{-}d} \ldots \boldsymbol{\mu}_C^{C\text{-}d}]$. If C-d is the desired projection dimension, determine the eigenvectors of $\mathbf{M}^{C\text{-}d}\mathbf{M}^{C\text{-}d,\mathrm{T}}$ that correspond to the C-d nonzero eigenvalues. Assign these eigenvectors to $\mathbf{V}$
  4. Otherwise, perform the following deflation operation and go back to the first step:

$$\mathbf{M}^{C\text{-}1} = \left(\mathbf{I}_N - \frac{\boldsymbol{\mu}^C\boldsymbol{\mu}^{CT}}{\|\boldsymbol{\mu}^C\|^2}\right)\mathbf{M}^C$$

The optimal projection of the data to C-1 dimensions is determined by the C-1 dimensional subspace orthogonal to $\boldsymbol{\mu}^C$; therefore, $\mathbf{M}^C$ is deflated as

$$\mathbf{M}^{C\text{-}1} = \left(\mathbf{I}_N - \frac{\boldsymbol{\mu}^C\boldsymbol{\mu}^{C\mathrm{T}}}{\|\boldsymbol{\mu}^C\|^2}\right)\mathbf{M}^C. \tag{17}$$

Any orthonormal bases that span the same space as the columns of the deflated matrix $\mathbf{M}^{C\text{-}1}$ is a valid candidate for the projection matrix $\mathbf{V}$ with C-1 orthonormal columns. A possible method to obtain these bases is to employ Gram–Schmidt orthonormalization to the columns of $\mathbf{M}^{C\text{-}1}$ and determine the eigenvectors of $\mathbf{M}^{C\text{-}1}\mathbf{M}^{C\text{-}1,\mathrm{T}}$ that correspond to the C-1 nonzero eigenvalues (which could be achieved sequentially). An efficient algorithm for sequential determination of these eigenvectors is provided in Appendix A. In the latter case, for example, the determined eigenvectors can be immediately assigned as $\mathbf{V}$.

The procedure continues similarly for reducing dimensionality further. The vector $\mathbf{s}^{C\text{-}1}$ is constructed (see Appendix C for the procedure for the construction of $\mathbf{s}^{C\text{-}d}$, since for $d > 0$ this step requires some care). The mean vector in the deflated space is calculated using $\boldsymbol{\mu}^{C\text{-}1} = \mathbf{M}^{C\text{-}1}\mathbf{r}^{C\text{-}1}$. The class means matrix is deflated using

$$\mathbf{M}^{C\text{-}2} = \left(\mathbf{I}_N - \frac{\boldsymbol{\mu}^{C\text{-}1}\boldsymbol{\mu}^{C\text{-}1,\mathrm{T}}}{\|\boldsymbol{\mu}^{C\text{-}1}\|^2}\right)\mathbf{M}^{C\text{-}1}. \tag{18}$$

As before, the orthonormal projection matrix $\mathbf{V}$ to C-2 dimensions is determined by finding the nonzero eigenvectors of $\mathbf{M}^{C\text{-}2}\mathbf{M}^{C\text{-}2,\mathrm{T}}$. The procedure is carried out in this manner until deflation down to the desired number of dimensions is achieved.

Once the column-orthonormal projection matrix $\mathbf{V}$, which is $N \times D$, is obtained previously unseen test samples can be transformed using

$$\boldsymbol{\varphi}(\mathbf{y}) = \sqrt{N}\mathbf{V}^{\mathrm{T}}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{k}(\mathbf{x}). \tag{19}$$

The overall algorithm is summarized in Table 1.

Note that the procedure described here requires determining the eigenvectors of an $N \times N$ kernel matrix. Unless certain simplifications are introduced, this process can potentially become $O(N^3)$. It is possible to avoid this level of complexity by determining the required eigenvectors sequentially using an algorithm as the one described in Appendix A. Nevertheless, such algorithms still require $O(N^2)$ calculations per eigenvector per iteration. Due to the iterative nature, the overall complexity might easily exceed analytical methods, such as those based on factorization techniques [32]. Alternatively, the eigendecomposition of the kernel matrix could be performed on smaller data matrices using representative subsets, and the Lanczos method or the Nystrom routine could be employed [31,32]. In fact, such an approach using a balanced number of samples from each class to determine the eigenfunctions could become preferable, as the prior class probabilities become more unbalanced, the eigenfunction estimates will become more biased towards emphasizing the stronger classes, thus yielding high-variance projection solutions.

3.5. The special case of projections to a single dimension

For illustration, we first focus on finding a one-dimensional nonlinear projection that maximizes MI with the class labels. For multi-dimensional projections the deflation procedure can be employed, yielding the optimal projection directions sequentially. The case of projections into an arbitrary number of dimensions will be discussed in the following section.

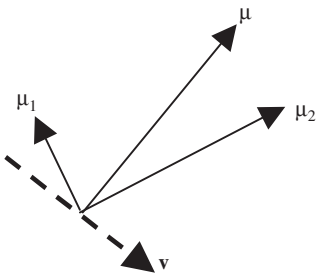

Fig. 2. The optimal subspace projection into one dimension in a two-class case, where $\boldsymbol{\mu}$ denotes the overall data mean, $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ denote class means and $\mathbf{v}$ is the projection direction.

Since $\mathbf{M}$ spans the subspace that the single dimensional projection vector $\mathbf{v}$ resides in, we express it as

$$\mathbf{v} = \mathbf{M}\mathbf{P}^{1/2}\boldsymbol{\alpha}, \tag{20}$$

where $\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\alpha} = 1$, and the projections of a data to one dimension under this methodology can be completely determined by choosing $\boldsymbol{\alpha}$ as composed of the entries of the following set $\{p_1^{1/2}, \ldots, p_C^{1/2}\}$, by shuffling them and modifying their signs as necessary (and perhaps replacing some with as determined by the appropriate rotation matrix).

In the case of two classes ($C = 2$), the two solutions are $\boldsymbol{\alpha} = [-p_2^{1/2}, p_1^{1/2}]^{\mathrm{T}}$ and its negative, which is an equivalent solution from the aspect of projection. In the case of three classes, the three distinct solutions are given by $\boldsymbol{\alpha} = [-p_2^{1/2}, p_1^{1/2}, 0]^{\mathrm{T}}$, $\boldsymbol{\alpha} = [-p_3^{1/2}, 0, p_1^{1/2}]^{\mathrm{T}}$, $\boldsymbol{\alpha} = [0, -p_3^{1/2}, p_2^{1/2}]^{\mathrm{T}}$. These solutions differ in their ordering of the projected classes on the projection axis and in general. The deflation procedure selects the solution that utilizes the two larger class probabilities, placing the class with the smallest probability in the center on the projection axis.

Similar analytical expressions could be derived for candidate projections in the case of more than three classes, but the general iterative procedure proposed in the previous section already considers these issues and constructs the solution without the need to go through all possible solutions (local maxima). Nevertheless, for cases with few classes, these direct analytical solutions are very practical. A geometric interpretation of the one-dimensional projection solution for the two-class case is shown in Fig. 2. Here $\boldsymbol{\mu}$ denotes the overall data mean, $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ denote class means and $\mathbf{v}$ is the projection direction. Since $\mathbf{v}$ can be defined only by the class mean vectors and class a priori probabilities, the computational load of the projection is mostly due to the eigenvector decomposition of the kernel matrix.

A practical consideration in selecting the kernel function in all spectral methods is the selection of the functional form of the kernel as well as the width of the kernel. Due to the kernel density estimation connection, it is natural to select this parameter based on the accuracy of the density estimate. This is discussed in Appendix D.

4. Experiments

In order to illustrate how the proposed nonparametric nonlinear projection scheme works, simulations using a synthetic data set—the crescent data set—and real data sets from the UCI database. Comparisons with Kernel LDA will be shown. These experiments demonstrate the effectiveness of the nonlinear projections obtained through this methodology in determining nonparametric projections to separate classes with nonlinear discriminant boundaries.

4.1. Crescent data set

This data set consists of two crescent-shaped classes with a nonlinear class boundary. For each class, 300

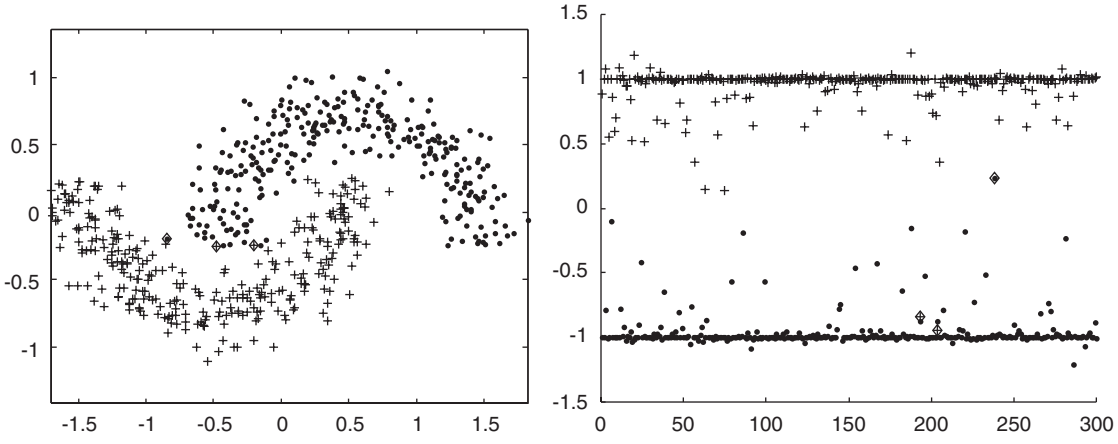

Fig. 3. The original samples for both classes indicated by + and · signs are shown at the left (a). At the right (b), the values of the one-dimensional projection are shown for both classes with the same signs. The ◊ symbols in both the plots indicate the classification errors made using a threshold on the projections values.

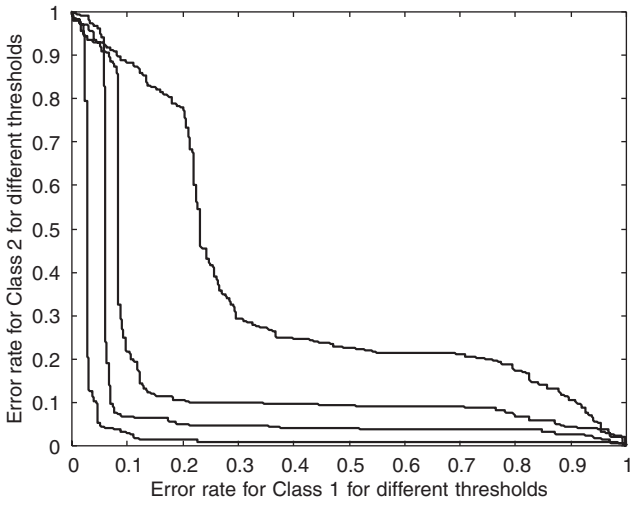

Fig. 4. The Prob(decide 1|true 2) vs Prob(decide2|true 1) curves for the crescent data set for various degrees of overlap (Gaussian radius standard deviations of 0.2, 0.3, 0.4, and 0.5).

two-dimensional samples are generated by uniformly selecting the angle in a π-radian arc and perturbing the radius with Gaussian distributed random values. The centers of the semicircles describing the classes are also shifted to create the nonlinear separation boundary. The class centers are selected to eliminate the possibility of having a linear projection direction on which the classes become easily separable. Therefore, nonlinear projections are required.

A sample simulation result using the crescent classes is presented in Fig. 3. The original data are shown in Fig. 3a and the values of the one-dimensional projection are presented in Fig. 3b. In both subfigures, the errors based on the optimal threshold on the nonlinear projection values are also indicated by diamonds. Note that the errors occur at the samples that are also visually separated well from their true classes.

The optimal classification threshold in the projection domain can be determined using the ROC curves[5] over a validation set with a data-oriented design philosophy, however, theoretically, the optimal threshold for a one-dimensional

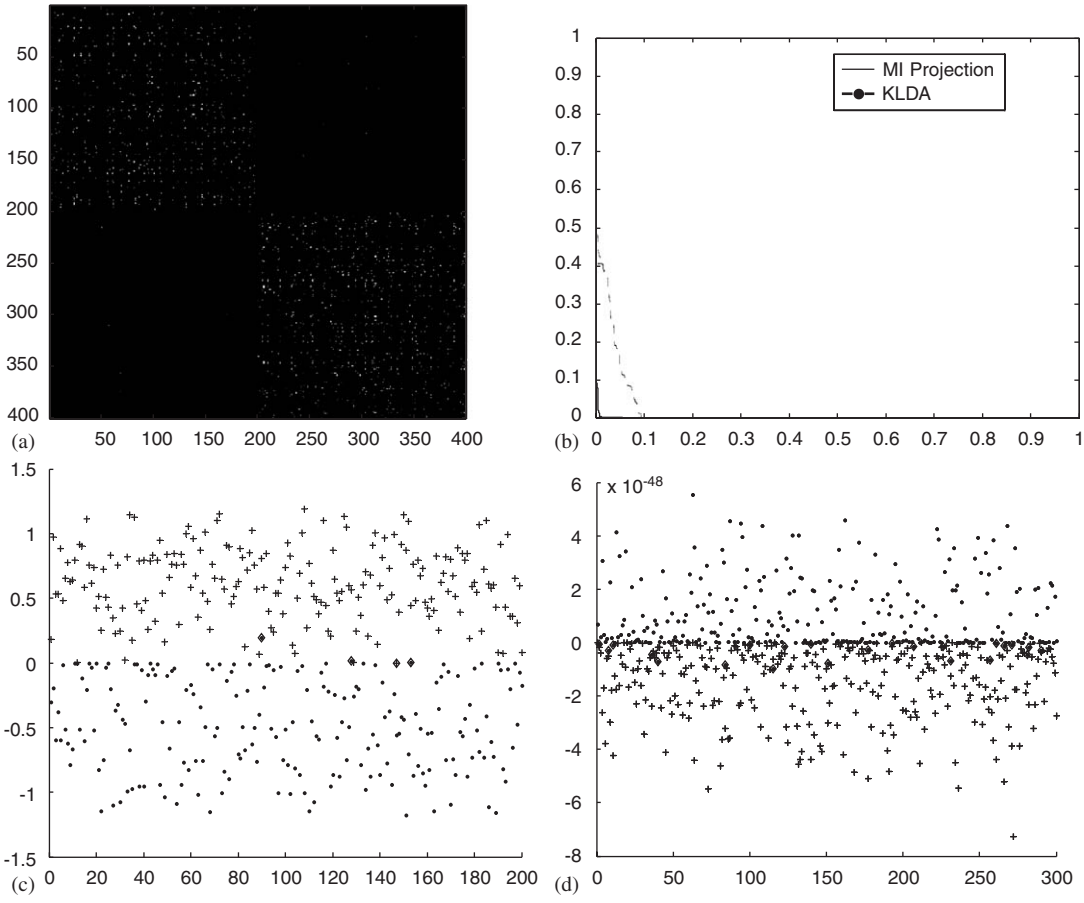

Fig. 5. The kernel matrix constructed using Silverman's rule and a spherically symmetric kernel is presented at the top left (a). The ROC curves for Kernel LDA and the proposed method are presented at the top right (b)—the proposed method outperforms KLDA significantly. The projection results of the proposed method for both classes indicated by + and · signs are shown at the bottom (c), where the ◊ symbols indicate the classification errors made using a zero threshold on the projections. The results for Kernel LDA are presented in (d) in the same manner.

[5] The ROC stands for *receiver operating characteristics*.

projection is zero. For various degrees of overlap (controlled by the variance of the Gaussian radial perturbation), the ROC curves for the crescent data set are depicted in Fig. 4. As expected, increasing the overlap results in *worse* ROC curves. The optimal threshold for a given data set is determined by the intersection point of the ROC curve with the line passing through the origin with slope p_1/p_2. Theoretically, this optimal threshold is zero, as also seen from Fig. 3b.

In the case of two-classes, there are two equivalent optimal solutions corresponding to the projection shown in Fig. 2b and its negative.

4.2. Comparison with Kernel LDA

To provide comparison with an existing similar benchmark nonlinear subspace projection method, the results of the proposed MI based projection scheme is compared with those of Kernel LDA. The comparisons are performed over three benchmark data sets on UCI database [33], namely handwritten digit recognition data set, Wisconsin breast cancer data set and ionosphere data set.

Handwritten digit classification database contains 250 samples from 44 subjects. Although the original database contains 10 digits, for ease of illustration, we utilize only the digits one and two. Being 16-dimensional, the original data are impossible to present in a figure even with a suitable two-dimensional subspace projection. For this data set, the original kernel matrix $\mathbf{K}$ constructed for the same kernel size with the one that has been used in our algorithm is presented along with the projection results into one dimension in Fig. 5. The optimal threshold of zero is assumed (since in the two-class case, the overall data mean vector in the KIFS determines the linear separation boundary, thus, the projections to its orthogonal must be separated by the zero-threshold). For Kernel LDA, the projection results corresponding to the optimal threshold value are demonstrated, and to generalize the performance comparison, ROC curves of these two methods are employed.

Similar experiments are performed using Wisconsin breast cancer data set and ionosphere data set. Preserving the a priori class probabilities in the training and testing sets, one-third of the data set is used for the training and the testing results corresponding to the remaining part are presented in Fig. 6. Gaussian kernels with Silverman's kernel size are used for the experiments and ROC curves corresponding to demonstrated for Wisconsin breast cancer data set and ionosphere data set in Fig. 6a and b, respectively.

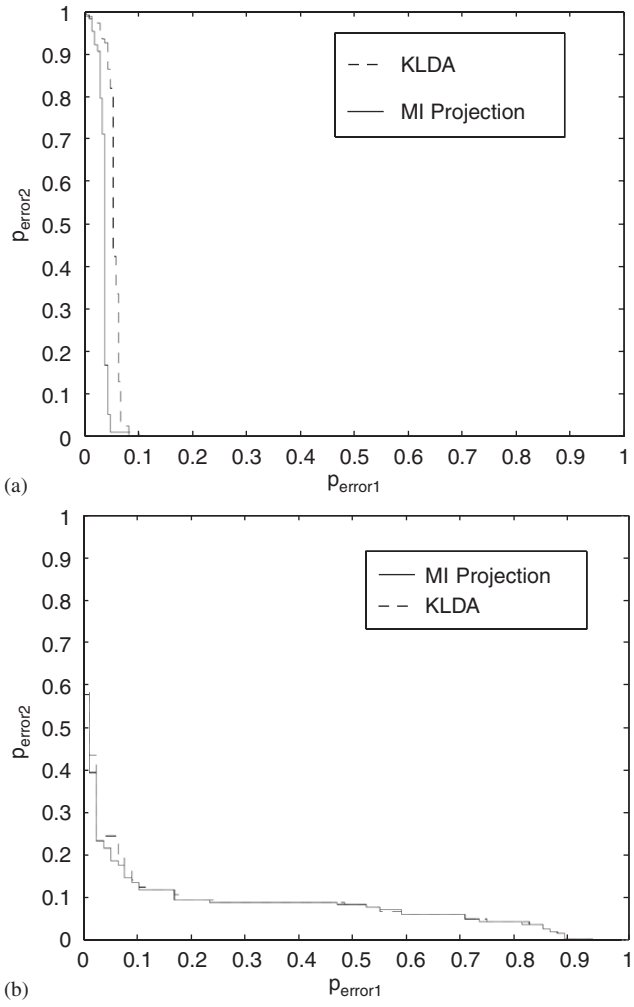

Fig. 6. ROC curves for (a) Wisconsin breast cancer data set and (b) ionosphere data set. Classification errors are shown with solid lines for MI projections and with dashed lines for Kernel LDA.

4.3. Landsat data set

Another real-data illustration is presented in this section for the Landsat data, which can be found in the UCI database [33]. This is a 36-dimensional data set with six classes, and 200 samples of these data points for each class are used for both training and testing. Results for the testing set performance are presented in Fig. 7. Being sorted according to their class labels, 1200 testing samples are projected to the $\boldsymbol{\varphi}$-space and the matrix consisting of the cosine of the angle between each data pair is shown in Fig. 7a as an image. Fig. 7b shows the pairwise cosine-angle image for the six-dimensional projections (in general, for a C-class data set, up to C projections can be considered with both the proposed and KLDA methods before the projection covariance matrices become rank-zero). The samples from the same class are expected to have a small angle between them, and larger angles are expected for interclass pairs. Note that for a given kernel selection, which is given by the Silverman rule of thumb here, the classification errors that one would make with the full dimensional data naturally leads to errors in the projections. This is due to the fact that if the original class distributions overlap then the projection cannot resolve this overlap. The class structure is more pronounced in the projection image than in the original data image in Fig. 7.

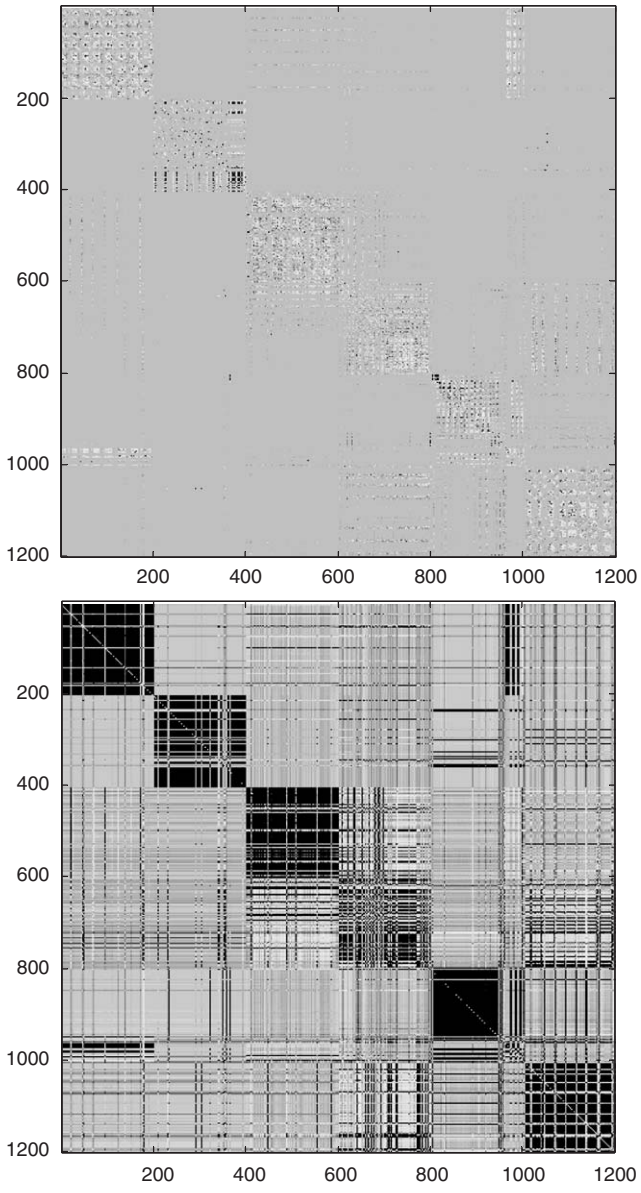

Fig. 7. The kernel matrix constructed using Silverman's rule and a spherically symmetric kernel is presented at the top. The bottom image shows the cosine of the angles between the six-dimensional projections determined by the proposed algorithm. In both images, the diagonal entries are zeroed and all other values are scaled linearly to the unit interval to maximize visual contrast.

5. Conclusions

Subspace projections are important tools in pattern recognition, as they can potentially improve classifier accuracy and generalization performance by eliminating redundant features and reducing dimensionality to allow for simpler classifier topologies, which in turn helps generalization. In addition, lower dimensional inputs to the classifiers help reduce computational complexity, therefore, can increase throughput.

Most traditional techniques are based on simple linear topologies (such as PCA and LDA) or parametric nonlinear topologies (such as radial basis functions or multi-layer perceptrons), which can be trained to optimize certain suitable class separability criteria. For example, LDA uses a separability criterion based on the assumption of Gaussian class conditional distributions. Linear or nonlinear parametric subspace projection topologies can be trained using more advanced separability criteria, such as MI between the projected features and the class labels.

In this paper, we have proposed a nonparametric nonlinear subspace projection methodology based on maximizing the Shannon MI between the projections and the class labels. Interpreting the nonparametric kernel estimator for MI as a nonparametric kernel-machine, we are able to determine nonlinear projections that maintain class separability nonparametrically. The proposed method lays out an interesting framework under which nonparametric kernel-density estimates of information theoretic optimality criteria can be linked to nonparametric nonlinear kernel-machines. The proposed approach first maps the original data to a very high-dimensional KIFS and then determines an optimal mapping from this space to a much lower dimensional space. Theoretically, projections of dimensionality more than the number of classes are not necessary.

The most important feature of the proposed approach is that the kernel calculations are done only once for the training data in order to determine the optimal nonlinear projection. Once the original data are projected to the KIFS (through the eigenfunctions of the kernel function used in the density estimation phase), the procedure reduces to a possibly analytical optimization routine, which only depends on the class priors and the simple lower-order statistics of the data (in this case, the mean vectors of the classes in the KIFS). In contrast, more traditional parametric projection algorithms based on optimizing the same nonparametric MI estimate would have to rely on gradient updates of the weights, which requires the $O(N^2)$ kernel matrix calculations at every iteration of the gradient algorithm. Consequently, the proposed method not only eliminates the unnecessary kernel evaluations introduced by such algorithms, but also allows us to determine the optimal solution in the KIFS analytically.

Future work will focus on reducing the memory and computational requirements of the nonparametric projection by determining accurate approximations to the spectral projection to a higher dimensional space and improving performance by incorporating variable kernel size density estimation to the presented framework.

Appendix A

Here, we present a simple algorithm to determine the largest eigenvectors sequentially. The algorithm makes use of the Rayleigh quotient. For a symmetric matrix **R**, from

the eigenvector equation, $\mathbf{Rv}=\lambda\mathbf{v}$, a unit variance eigenvector $\mathbf{v}$, $\lambda=\mathbf{v}^{\mathrm{T}}\mathbf{Rv}$. Therefore, the unit-norm eigenvector is a fixed point of the following iteration:

$$\mathbf{v}_{k+1}=T\mathbf{v}_k+(1-T)\frac{\mathbf{Rv}_k}{\mathbf{v}_k^{\mathrm{T}}\mathbf{Rv}_k}. \tag{A.1}$$

The stepsize T is introduce to eliminate the limit cycle that arises from the dynamics behavior of the fixed point algorithm with $T=0$. For $T\in(0,1)$, the iterations in Eq. (A.1) can be shown to converge to the largest eigenvector of the symmetric positive definite matrix $\mathbf{R}$.

In order to obtain the subsequent large eigenvectors, deflation can be employed after the convergence of Eq. (A.1). This is accomplished by replacing $\mathbf{R}$ with its deflated version:

$$\mathbf{R}=(\mathbf{I}-(\mathbf{v}_\infty^{\mathrm{T}}\mathbf{Rv}_\infty)\mathbf{v}_\infty\mathbf{v}_\infty^{\mathrm{T}})\mathbf{R}. \tag{A.2}$$

For the deflated $\mathbf{R}$, the iterations in Eq. (A.1) will converge to the second largest eigenvector of the original $\mathbf{R}$ and so on.

Appendix B

Since the data transformations are calculated using (13) for both training and testing data, the class mean vectors are orthogonal to each other with their individual norms equal to $p_c^{-1/2}$, p_c being the class prior probability. This leads to the following:

$$\begin{aligned}\boldsymbol{\mu}_c&=\frac{1}{N_c}\sum_{j=1}^{N_c}\sqrt{N}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{k}(\mathbf{x}_j^c)\\&\approx\frac{\sqrt{N}}{N_c}\tilde{\boldsymbol{\Lambda}}^{-1}\boldsymbol{\Phi}_{\mathbf{x}}\sum_{j=1}^{N_c}\boldsymbol{\Phi}_{\mathbf{x}}^{\mathrm{T}}\boldsymbol{\Lambda}\bar{\boldsymbol{\varphi}}(\mathbf{x}_j^c)\\&=\frac{\sqrt{N}}{N_c}\sum_{j=1}^{N_c}\bar{\boldsymbol{\varphi}}(\mathbf{x}_j^c)\\&\approx\frac{\sqrt{N}}{N_c}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{m}_c.\end{aligned} \tag{B.1}$$

Now consider the inner product between two mean vectors:

$$\begin{aligned}\boldsymbol{\mu}_c^{\mathrm{T}}\boldsymbol{\mu}_d&=\frac{N}{N_cN_d}\mathbf{m}_c{}^{\mathrm{T}}\boldsymbol{\Phi}_{\mathbf{x}}^{\mathrm{T}}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{m}_d\\&=\frac{N}{N_cN_d}\mathbf{m}_c^{\mathrm{T}}\mathbf{m}_d=\begin{cases}N/N_c & \text{if } c=d,\\0 & \text{if } c\neq d.\end{cases}\end{aligned} \tag{B.2}$$

Thus, the mean vectors of each class in the $\boldsymbol{\varphi}$-space create an orthogonal (but not normal) basis for the space in which our optimization variable $\mathbf{V}$ resides.

Appendix C

In the algorithm provided in Section 3.4, at every deflation step, some of the class mean vectors must be flipped in order to ensure that all class mean vectors lie in one half of the C-d dimensional space at the dth step of deflation. This process guarantees that the argument of the logarithm in the criterion remains positive. Initially, we have C mean vectors in the C-dimensional space, and in each deflation step the dimensionality (rank) is reduced by one. In the dth step of deflation, the d classes with the lowest a priori probabilities may need to be flipped, i.e., their corresponding $\mathbf{s}^{C\text{-}d}$ entry is selected to be -1. The remaining C-d class means are not changed. In order to determine which mean vectors (with small probabilities will be flipped, after sorting the vectors according to their decreasing a priori class probabilities as $\{\boldsymbol{\mu}'_1,\ldots,\boldsymbol{\mu}'_C\}$, using the normalized versions of the mean vectors $\{\boldsymbol{\mu}'_1,\ldots,\boldsymbol{\mu}'_{C\text{-}d}\}$ we calculate the vector $\mathbf{u}^{C\text{-}d}=\boldsymbol{\mu}'_1/\|\boldsymbol{\mu}'_1\|+\cdots+\boldsymbol{\mu}'_{C\text{-}d}/\|\boldsymbol{\mu}'_{C\text{-}d}\|$. The corresponding $\mathbf{s}^{C\text{-}d}$ entries of the vectors $\{\boldsymbol{\mu}'_{C\text{-}d+1},\ldots,\boldsymbol{\mu}'_C\}$ are set to $s'^{\,C\text{-}d}_j=\mathrm{sign}(\boldsymbol{\mu}'_j{}^{\mathrm{T}}\mathbf{u}^{C\text{-}d})$. Although this is not the only possible selection of $\mathbf{s}^{C\text{-}d}$, flipping the class means with lowest prior probabilities leads us to the best solution, which maximizes the separation between the classes that have the highest a priori probabilities.

Appendix D

Typically, the problem is simplified by assuming a parametric family of kernels and trying to optimize the parameters based on the quality of the solutions obtained [34]. Even when the functional form of the kernel is fixed, there seems to be no principled way of setting the kernel size, in the literature, prior to solving the problem. In fact, often the kernel size is varied and the one that gives the *best* solution is selected. This is definitely an unacceptable and unnecessary computational load on all spectral algorithms.

The connection to density estimation, presented in Eq. (8), clearly indicates that the kernel function should be selected to match the distribution of the data as much as possible. There is a wide literature on how to select kernel sizes for kernel density estimates, including methods that range from heuristics to principled Bayesian approaches such as maximum likelihood [35–37]. For simplicity, in the following experiments, a circular Gaussian kernel is assumed and its width parameter (variance) is determined utilizing the rule of thumb by Silverman [38]:

$$\sigma^2=\frac{1}{n}\mathrm{tr}(\boldsymbol{\Sigma}_{\mathbf{x}})\left(\frac{4}{(2n+1)N}\right)^{2/(n+4)}, \tag{D.1}$$

where n is the dimensionality of the data $\mathbf{x}$, N is the number of samples, and $\boldsymbol{\Sigma}_{\mathbf{x}}$ is the sample covariance of the training set. Clearly, certain obvious improvements include

utilizing a different kernel size for each class or even each data point itself, and allowing anisotropic covariances as kernel width, as well as using kernel size optimization procedures that do not assume Gaussianity [39]. For now, we leave these discussions to be studied as a future work, since the goal of this paper is to demonstrate the concept, rather than optimizing every little implementation detail.

References

[1] E. Oja, Subspace Methods of Pattern Recognition, Wiley, New York, 1983.

[2] P.A. Devijver, J. Kittler, Pattern Recognition: A Statistical Approach, Prentice-Hall, London, 1982.

[3] B. Scholkopf, A. Smola, K.R. Muller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (1998) 1299–1319.

[4] A. Hyvarinen, J. Karhunen, E. Oja, Independent Component Analysis, Wiley, New York, 2001.

[5] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (1999) 788–791.

[6] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.

[7] J. Costa, A.O. Hero, Classification constrained dimensionality reduction, Proceedings of ICASSP, vol. 5, 2005, pp. 1077–1080.

[8] K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, New York, 1990.

[9] G. Baudat, F. Anouar, Generalized discriminant analysis using a kernel approach, Neural Comput. 12 (2000) 2385–2404.

[10] J.C. Principe, J.W. Fisher, D. Xu, Information theoretic learning, in: S. Haykin (Ed.), Unsupervised Adaptive Filtering, Wiley, New York, 2000, pp. 265–319.

[11] K. Torkkola, Feature extraction by non-parametric mutual information maximization, J. Mach. Learn. Res. 3 (2003) 1415–1438.

[12] T. Cover, J. Thomas, Elements of Information Theory, Wiley, New York, 1991.

[13] D. Koller, M. Sahami, Toward optimal feature selection, Proceedings of the International Conference on Machine Learning, Bari, Italy, 1996, pp. 284–292.

[14] R. Battiti, Using mutual information for selecting features in supervised neural net learning, Neural Networks 5 (4) (1994) 537–550.

[15] B.V. Bonnlander, A.S. Weigend, Selecting input variables using mutual information and nonparametric density estimation, Proceedings of International Symposium on Artificial Neural Networks, Tainan, Taiwan, 1994, pp. 42–50.

[16] H. Yang, J. Moody, Data visualization and feature selection: new algorithms for nonGaussian data, Adv. Neural Inf. Process. Syst. (2000) 687–693.

[17] D. Erdogmus, Information theoretic learning: Renyi's entropy and its applications to adaptive system training, Ph.D. Dissertation, University of Florida, Gainesville, Florida, 2002.

[18] A. Kraskov, H. Stoegbauer, P. Grassberger, Estimating mutual information, Phys. Rev. E 69 (2004) 066138.

[19] E.G. Learned-Miller, J.W. Fisher III, ICA using spacings estimates of entropy, J. Mach. Learn. Res. 4 (2003) 1271–1295.

[20] O. Vasicek, A test for normality based on sample entropy, J. R. Stat. Soc. B 38 (1) (1976) 54–59.

[21] A.O. Hero III, B. Ma, O.J.J. Michel, J. Gorman, Applications of entropic spanning graphs, IEEE Signal Process. Mag. 19 (5) (2002) 85–95.

[22] J. Beirlant, E.J. Dudewicz, L. Gyorfi, E.C. van der Meulen, Nonparametric entropy estimation: an overview, Int. J. Math. Stat. Sci. 6 (1) (1997) 17–39.

[23] D. Erdogmus, J.C. Principe, An error-entropy minimization algorithm for supervised training of nonlinear adaptive systems, IEEE Trans. Signal Process. 50 (7) (2002) 1780–1786.

[24] R.M. Fano, Transmission of Information: A Statistical Theory of Communications, MIT Press, New York, 1961.

[25] M.E. Hellman, J. Raviv, Probability of error, equivocation and the Chernoff bound, IEEE Trans. Inf. Theory 16 (1970) 368–372.

[26] D. Erdogmus, J.C. Principe, Lower and upper bounds for misclassification probability based on Renyi's information, J. VLSI Signal Process. Syst. 37 (2/3) (2004) 305–317.

[27] K.D. Bollacker, J. Ghosh, Linear feature extractors based on mutual information, Proceedings of International Conference on Pattern Recognition, Vienna, Austria, 1996, pp. 720–724.

[28] J. Mercer, Functions of positive and negative type, and their connection with the theory of integral equations, Trans. London Philos. Soc. A 209 (1909) 415–446.

[29] G. Wahba, Spline Models for Observational Data, SIAM, Philadelphia, PA, 1990.

[30] H. Weinert (Ed.), Reproducing Kernel Hilbert Spaces: Applications in Statistical Signal Processing, Hutchinson Ross Pub. Co., Stroudsburg, PA, 1982.

[31] C. Fowlkes, S. Belongie, F. Chung, J. Malik, Spectral grouping using the Nystrom method, IEEE Trans. Pattern Anal. Mach. Intell. 23 (2004) 298–305.

[32] G.H. Golub, C.F. van Loan, Matrix Computations, third ed., Johns Hopkins University Press, Baltimore, MA, 1996.

[33] UCI Machine Learning Repository, ⟨http://www.ics.uci.edu/~mlearn/MLSummary.html⟩.

[34] M. Seeger, Gaussian processes for machine learning, Int. J. Neural Syst. 14 (2) (2004) 69–106.

[35] R.P.W. Duin, On the choice of the smoothing parameters for Parzen estimators of probability density functions, IEEE Trans. Comput. 25 (11) (1976) 1175–1179.

[36] L. Devroye, G. Lugosi, Combinatorial Methods in Density Estimation, Springer, New York, 2001.

[37] N.N. Schraudolph, Gradient-based manipulation of nonparametric entropy estimates, IEEE Trans. Neural Networks 15 (4) (2004) 828–837.

[38] B.W. Silverman, Density Estimation for Statistics and Data Analysis, Chapman & Hall, London, 1986.

[39] N. Kumar, A.G. Andreou, Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition, Speech Commun. 26 (4) (1998) 283–297.