MAXIMALLY DISCRIMINATIVE SPECTRAL FEATURE PROJECTIONS USING MUTUAL INFORMATION

Umut Ozertem, Deniz Erdogmus
CSEE Department, OGI, Oregon Health & Science University, Portland, OR 97006, USA

Abstract—**Determining the optimal subspace projections, which maintains the best representation of the original data, is an important problem in machine learning and pattern recognition. In this paper, we propose a nonparametric nonlinear subspace projection technique that employs kernel density estimation based information theoretic methods and kernel machines, in order to maintain class separability maximally under the Shannon mutual information criterion.**

I. INTRODUCTION

Dimensionality reduction is an important step in a variety of applications including pattern recognition, data compression, and exploratory data analysis. This results from the fact that the relevant information of the data can be represented in lower dimensions, which not only reduces the computational complexity but also provides a generalization of the data, leading to a robust solution.

Projection can be achieved either by a feature transformation or a feature selection. 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. Besides, since feature selection is a special case of feature transformations, we are mainly interested in feature 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.

Principle components analysis (PCA) is historically the first dimensionality reduction technique [1]. PCA and its nonlinear extension to nonlinear projections, Kernel PCA [2,3], exhibit the same shortcoming, namely, the projected features are not necessarily useful for classification.

Linear Discriminant Analysis (LDA) attempts to tackle this shortcoming of PCA by searching for linear projections that maximizes class separability under Gaussianity assumption. LDA projections are optimized based on the means and covariance matrices of classes, which are not descriptive for an arbitrary probability density function. Its nonlinear extension Kernel LDA [4], generalizes this assumption by first projecting the data to a hypothetical high dimensional space where the Gaussianity condition is assumed to be satisfied. However, the kernel functions used in practice do not necessarily validate this assumption.

Second-order statistical measures have found widespread application in many areas of machine learning and pattern recognition. However, the insufficiency of only 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, and proven to be superior to the traditional second-order measures. 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 mutual information (MI) between the projected features and the class labels, which is motivated by lower and upper bounds in information theory that relate this MI to probability of error. In principle, MI measures nonlinear dependencies between a set of random variables taking the higher order statistical structures existing in the data into account, as opposed to linear and second-order statistical measures such as correlation and covariance [7].

Mutual information can be estimated nonparametrically from the training samples [8]. Since the class label vector is discrete-valued, the problem reduces to just estimating entropies of continuous random vectors. The multi-dimensional entropy can be estimated nonparametrically using a number of techniques. However, techniques based on sample spacing are not differentiable, hence not suitable for adaptive learning of feature projections [9]. On the other hand, entropy estimators based on kernel density estimation (KDE) provide a differentiable alternative [8,10].

In this paper, we propose a method for determining optimal nonlinear feature projections that maximize the Shannon mutual information 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 methods are computationally prohibitive for large training sets. We propose to avoid this complication by exploiting the kernel induced feature (KIF) transformation to obtain an algorithm that has $O(N)$ complexity. Further computational savings are achieved by employing the deflation procedure in the KIF space to determine each projection sequentially rather than simultaneously.

II. THEORETICAL BACKGROUND

The aim of the feature subspace projections is to establish a generalization of the data in order to improve the classifier robustness as well as reducing the computational complexity of the classifiers. On the other hand, the classifier performance must not be compromised during the projections by losing information about the data by throwing away some useful components. Theoretically, the optimal subspace

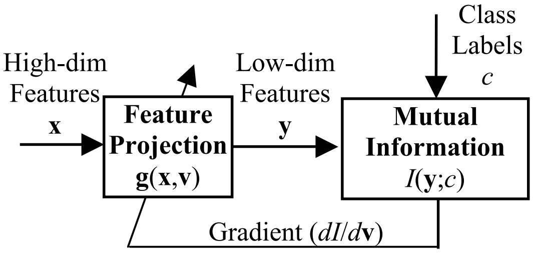


Figure 1. Determining optimal feature subspace projections using mutual information.

projections should minimize the Bayesian risk, and since it is a widely used and accepted risk function, we will use the probability of error as Bayesian risk function.

The average probability of error has been shown to be related to MI between the feature vectors and the class labels. Specifically, Fano's and Hellman & 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 [11,12]. Maximizing this MI reduces both bounds, therefore, forces the probability of error to decrease, leading to an improved classifier performance [6].

In feature extraction, we are interested in the MI between the continuous-valued feature vector **y** and the discrete-valued class labels *c*. We formulate the problem using Renyi's generalized definition of MI between **y** and *c* with respect to α is defined in terms of the overall data and the individual class distributions as [7]

$$\begin{aligned} I_\alpha(\mathbf{y};c) &= \frac{1}{\alpha-1}\log\sum\nolimits_c \int p_{\mathbf{y}c}^{\alpha}(\mathbf{y},c)\,p_{\mathbf{y}}^{1-\alpha}(\mathbf{y})\,p_c^{1-\alpha}\;d\mathbf{y} \\ &= \frac{1}{\alpha-1}\log\sum\nolimits_c p_c \int p_{\mathbf{y}|c}^{\alpha}(\mathbf{y}\,|\,c)\,p_{\mathbf{y}}^{1-\alpha}(\mathbf{y})\;d\mathbf{y} \end{aligned} \tag{1}$$

where p_c are the prior class probabilities, The overall data distribution in terms of class conditional distributions $p(\mathbf{y}|c)$ is given as,

$$p(\mathbf{y}) = \sum\nolimits_c p_c\, p(\mathbf{y}\,|\,c) \tag{2}$$

As seen in (1), in order to estimate MI we need to estimate the conditional class probability distributions as well as the overall data distribution. Density estimators based on sample spacing are not suitable for gradient-based adaptation, and a feasible alternative is the KDE-based plug-in estimator [8]. Clearly, optimizing a nonlinear topology to maximize (1) using the KDE-based estimators 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.

Under the framework of *optimal feature subspace projections that maximize mutual information with class labels*, the adaptive learning procedure to find these optimal projections follows the block diagram shown in Fig. 1. A high dimensional feature vector is projected to a lower dimensional vector by a nonlinear topology (such as a neural network), whose weights (denoted by **v**) are optimized to maximize the MI criterion [5,6].

III. 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 $\{c_1,c_2,\ldots,c_N\}$, where 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 such that the MI between the projection and the class labels, namely $I_S(\mathbf{y},c)$, 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 [13] form a basis for the Hilbert space of finite power nonlinear functions [14].[1] Therefore, every finite-L_2-norm nonlinear transformation $g_d(\mathbf{x})$ can be expressed as a linear combination of these bases:

$$y_d = g_d(\mathbf{x}) = \mathbf{v}_d^T\bar{\boldsymbol{\varphi}}(\mathbf{x}) \tag{3}$$

where y_d is the d^{th} component of the projection vector **y**. As we will show next, such linear combinations of nonlinear basis functions arise naturally from the KDE-based nonparametric estimates of mutual information in the context of feature subspace projections.

A. Estimating the MI Nonparametrically Using KDE

Consider the Renyi's MI between the high-dimensional original feature vectors and the class labels.

$$I_\alpha(\mathbf{x};c) = \frac{1}{\alpha-1}\log\sum\nolimits_c p_c E_{\mathbf{x}|c}\left[\left(\frac{p_{\mathbf{x}|c}(\mathbf{x}\,|\,c)}{p_{\mathbf{x}}(\mathbf{x})}\right)^{\alpha-1}\right] \tag{4}$$

Estimating the pdf's using a KDE estimator with kernel *K*(.) and approximating the conditional expectation by a sample mean we obtain

$$I_\alpha(\mathbf{x};c) \approx \frac{1}{\alpha-1}\log\sum_c \frac{p_c}{N_c}\sum_{j=1}^{N_c}\left(\frac{(1/N_c)\sum_{i=1}^{N_c}K(\mathbf{x}_j^c-\mathbf{x}_i^c)}{(1/N)\sum_{i=1}^{N}K(\mathbf{x}_j^c-\mathbf{x}_i)}\right)^{\alpha-1} \tag{5}$$

Assuming that *K* is a Mercer kernel we can write, $K(\mathbf{x}-\mathbf{x}') = \bar{\boldsymbol{\varphi}}^T(\mathbf{x})\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\varphi}}(\mathbf{x}')$. Hence, the MI estimate becomes

$$I_\alpha(\mathbf{x};c) \approx \frac{1}{\alpha-1}\log\sum_c \frac{p_c}{N_c}\sum_{j=1}^{N_c}\left(\frac{N\bar{\boldsymbol{\varphi}}^T(\mathbf{x}_j^c)\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\Phi}}_{\mathbf{x}}\mathbf{m}_c}{N_c\bar{\boldsymbol{\varphi}}^T(\mathbf{x}_j^c)\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\Phi}}_{\mathbf{x}}\mathbf{1}}\right)^{\alpha-1} \tag{6}$$

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, as well as a vector of ones, denoted by **1**. Besides, we introduced the matrix $\bar{\boldsymbol{\Phi}}_{\mathbf{x}} = [\bar{\boldsymbol{\varphi}}(\mathbf{x}_1)\cdots\bar{\boldsymbol{\varphi}}(\mathbf{x}_N)]$, where $N=N_1+\ldots+N_C$. Defining the mean vectors in the transformed domain as $\bar{\boldsymbol{\mu}}_c = (1/N_c)\bar{\boldsymbol{\Phi}}_{\mathbf{x}}\mathbf{m}_c$ and $\bar{\boldsymbol{\mu}} = (1/N)\bar{\boldsymbol{\Phi}}_{\mathbf{x}}\mathbf{1}$ for class *c* and whole data set respectively, we obtain

$$I_\alpha(\mathbf{x};c) \approx \frac{1}{\alpha-1}\log\sum_c \frac{p_c}{N_c}\sum_{j=1}^{N_c}\left(\frac{\bar{\boldsymbol{\varphi}}^T(\mathbf{x}_j)\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\mu}}_c}{\bar{\boldsymbol{\varphi}}^T(\mathbf{x}_j)\bar{\boldsymbol{\Lambda}}\bar{\boldsymbol{\mu}}}\right)^{\alpha-1} \tag{7}$$

[1] The bar denotes true eigenfunctions and eigenvalues of the kernel.

As also seen from (7), we can obtain different cost functions for different values of α. The robustness and the performance of the projection results strictly depend on the choice of α. As an example, one can easily see that for increasing values of α the MI estimator is becoming to be less dependent to the outliers in the data. For the limiting case as α approaches to 1, Renyi's MI converges to Shannon's MI definition, which is widely used and merits special attention. At this point, we will use the Shannon's MI by taking the limit as $\alpha \to 1$, leaving the dependency on α to be studied later.

$$I_S(\mathbf{x};c) = \lim_{\alpha \to 1} I_R(\mathbf{x};c) \approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \left| \frac{\overline{\boldsymbol{\varphi}}^T(\mathbf{x}_j)\overline{\boldsymbol{\Lambda}}\overline{\boldsymbol{\mu}}_c}{\overline{\boldsymbol{\varphi}}^T(\mathbf{x}_j)\overline{\boldsymbol{\Lambda}}\overline{\boldsymbol{\mu}}} \right| \quad (7)$$

Note that so far we have only utilized the true eigenfunctions and the eigenvectors of the kernel function.

*B. Spectral Transformations that Maximize Shannon Mutual Information in the Kernel-Induced Feature Spac***e**

According to our projection model in (3), effectively, the projection is accomplished in the kernel-induced φ-space. If the target reduced dimensionality is D, we have $\mathbf{y} = \mathbf{V}^T \overline{\boldsymbol{\varphi}}(\mathbf{x})$, where $\mathbf{V}=[\mathbf{v}_1 \ldots \mathbf{v}_D]$ consists of orthonormal columns $\mathbf{v}_d$. Therefore, the best L_2-orthogonal approximation for $\overline{\boldsymbol{\varphi}}(\mathbf{x})$ is

$$\overline{\boldsymbol{\varphi}}(\mathbf{y}) = \mathbf{V}\mathbf{V}^T \overline{\boldsymbol{\varphi}}(\mathbf{x}) \quad (8)$$

This leads to the following cost function that needs to be maximized by optimizing **V**:

$$J(\mathbf{V}) = \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \left| \frac{\overline{\boldsymbol{\varphi}}^T(\mathbf{x}_j)\mathbf{V}\mathbf{V}^T\overline{\boldsymbol{\Lambda}}\mathbf{V}\mathbf{V}^T\overline{\boldsymbol{\mu}}_c}{\overline{\boldsymbol{\varphi}}^T(\mathbf{x}_j)\mathbf{V}\mathbf{V}^T\overline{\boldsymbol{\Lambda}}\mathbf{V}\mathbf{V}^T\overline{\boldsymbol{\mu}}} \right| \quad (9)$$

Analytical expressions for the eigenfunctions of the kernel $\overline{\boldsymbol{\varphi}}(\mathbf{x})$ are not available. However, spectral methods provide necessary tools to approximate these from the training samples. Following the common procedure in spectral methods, using all training samples in pairs as $\mathbf{K}_{ij} = K(\mathbf{x}_i - \mathbf{x}_j)$, we define the symmetric kernel matrix **K** (also called the affinity matrix). The matrix **K** can be decomposed into its eigenvalues and eigenvectors as $\mathbf{K} = \boldsymbol{\Phi}_\mathbf{x}^T \boldsymbol{\Lambda} \boldsymbol{\Phi}_\mathbf{x}$, which are essentially approximations of the sought eigenfunctions and eigenvalues of the kernel function. Hence the eigenfunctions can be approximated using the eigendecomposition of the affinity matrix **K** as follows:

$$\boldsymbol{\varphi}(\mathbf{x}) = \sqrt{N}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_\mathbf{x}\mathbf{k}(\mathbf{x}) \quad (10)$$

where $\mathbf{k}(\mathbf{x})=[K(\mathbf{x}-\mathbf{x}_1),\ldots, K(\mathbf{x}-\mathbf{x}_N)]^T$. Substituting this, the transformations become $\mathbf{y} = \mathbf{V}^T \boldsymbol{\varphi}(\mathbf{x})$ and (9) becomes,

$$J(\mathbf{V}) = \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} \log \left| \frac{\boldsymbol{\varphi}^T(\mathbf{x}_j)\mathbf{V}\mathbf{V}^T\boldsymbol{\Lambda}\mathbf{V}\mathbf{V}^T\boldsymbol{\mu}_c}{\boldsymbol{\varphi}^T(\mathbf{x}_j)\mathbf{V}\mathbf{V}^T\boldsymbol{\Lambda}\mathbf{V}\mathbf{V}^T\boldsymbol{\mu}} \right| \quad (11)$$

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 φ-space. It is important to note that the class priors p_c are estimated from the training data by N_c/N and $\boldsymbol{\mu}=p_1\boldsymbol{\mu}_1+\ldots+p_C\boldsymbol{\mu}_C$.

A critical issue affecting the performance of the subspace projections is the suitable selection of the kernel function. A practical consideration in selecting the kernel function is the selection of the functional form of the kernel as well as the width of the kernel. Typically, this problem is tackled by trying to optimize the parameters for a family of kernels of some specific type. The connection to kernel density estimation, presented in (5), clearly indicates that the kernel function should be selected to match the distribution of the data as much as possible. 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 that gives the *optimal* kernel size for the data set assuming that a Gaussian distribution underlies [16]:

$$\sigma^2 = \frac{1}{n} tr(\boldsymbol{\Sigma}_\mathbf{x}) \left(\frac{4}{(2n+1)N} \right)^{2/(n+4)} \quad (12)$$

where n is the dimensionality of the data **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, however, such modifications will be studied in future publications, since the goal of this paper is to demonstrate the concept, rather than optimizing every little implementation detail.

C. Projections to a Single Dimension

For illustration, first we 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 after optimizing each projection vector, yielding the optimal projection directions sequentially, which results in lower computational load as compared to searching for all the projections simultaneously.

Imposing the constraint $\mathbf{v}^T\mathbf{v}=1$, we need to maximize

$$J(\mathbf{v}) = \sum_c p_c \log \left| \frac{\mathbf{v}^T\boldsymbol{\mu}_c}{\mathbf{v}^T\boldsymbol{\mu}} \right| \quad (13)$$

A very important observation is that these 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 is due to the fact that the data transformations are calculated using (10) for both training and testing data. This leads to the following:

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{j=1}^{N_c} \sqrt{N}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_\mathbf{x}\mathbf{k}(\mathbf{x}_j^c) \approx \frac{\sqrt{N}}{N_c}\boldsymbol{\Phi}_\mathbf{x}\mathbf{m}_c \quad (14)$$

Now consider the inner product between two mean vectors:

$$\boldsymbol{\mu}_c^T\boldsymbol{\mu}_d = \frac{N}{N_c N_d}\mathbf{m}_c^T\mathbf{m}_d = \begin{cases} N/N_c & \text{if } c = d \\ 0 & \text{if } c \neq d \end{cases} \quad (15)$$

Thus, the mean vectors of each class in the φ-space create an orthogonal (but not normal) basis for the space in which our optimization variable **v** lies in. Defining a basis matrix $\mathbf{M} = [\boldsymbol{\mu}_1 \ldots \boldsymbol{\mu}_C]$ —which satisfies $\mathbf{M}^T\mathbf{M}=\mathbf{P}^{-1}$, where is $\mathbf{P}=diag(p_1,\ldots,p_C)$— we can express **v** as

$$\mathbf{v} = \mathbf{M}\mathbf{P}^{1/2}\boldsymbol{\alpha} \tag{16}$$

where $\boldsymbol{\alpha}^T\boldsymbol{\alpha}=1$. Using (16), and the identities $\boldsymbol{\mu} = \mathbf{M}\mathbf{p}$ and $\mathbf{M}^T\boldsymbol{\mu}_c = p_c^{-1}\mathbf{e}_c$, where **p** is the vector of class priors and $\mathbf{e}_c$ is the canonical unit vector in direction, the maximization problem in (13) can be converted to a problem in terms $\boldsymbol{\alpha}$ subject to $\boldsymbol{\alpha}^T\boldsymbol{\alpha}=1$ as:

$$\begin{aligned}\max_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}) &= \sum_{c=1}^{C} p_c \log \frac{\left|\boldsymbol{\alpha}^T\mathbf{P}^{1/2}\mathbf{e}_c / p_c\right|}{\left|\sum_{d=1}^{C} p_d \boldsymbol{\alpha}^T\mathbf{P}^{1/2}\mathbf{e}_d / p_d\right|} \\ &= \sum_{c=1}^{C} p_c \log \frac{\left|\alpha_c\right| p_c^{1/2}}{\left|\sum_{d=1}^{C} \alpha_d p_d^{1/2}\right|} - \sum_{c=1}^{C} p_c \log p_c\end{aligned} \tag{17}$$

Notice that, due to the constraint $\boldsymbol{\alpha}^T\boldsymbol{\alpha}=1$, we can express all feasible solutions of $\boldsymbol{\alpha}$ in terms of rotations of a unit norm vector. For convenience, consider rotations of the form $\boldsymbol{\alpha}=\mathbf{R}\mathbf{q}$, where **q** is a vector consisting of entries $q_c=p_c^{1/2}$. With this substitution, we can rewrite (17) as shown in (18), where D_{KL} denotes the Kullback-Leibler divergence measure. Clearly, the first term is an inconsequential constant in the optimization problem, and a rotation matrix that achieves $\mathbf{q}^T\mathbf{R}\mathbf{q}=0$ maximizes the criterion. Note that this is equivalent to selecting **Rq** orthogonal to **q**. Since the coordinates of the mean vector $\boldsymbol{\mu}$ in terms of the bases given by the normalized class means $\boldsymbol{\mu}_c p_c^{1/2}$ is also $p_c^{1/2}=q_c$, this solution coincides with the observation that the optimal projection should be orthogonal to the overall data mean vector in the $\boldsymbol{\varphi}$-space.

$$\begin{aligned}\max_{\mathbf{R}} J(\mathbf{R}) &= \sum_{c=1}^{C} p_c \log \frac{\left|\mathbf{R}_{c:}\mathbf{q}\right| q_c}{\left|\mathbf{q}^T\mathbf{R}\mathbf{q}\right|} - \sum_{c=1}^{C} p_c \log p_c \\ &= -D_{KL}(\mathbf{p} \,\|\, \mathbf{q}) + E_{\mathbf{p}}\left[\log \frac{\left|\mathbf{R}_{c:}\mathbf{q}\right|}{\left|\mathbf{q}^T\mathbf{R}\mathbf{q}\right|}\right]\end{aligned} \tag{18}$$

In general, rotation matrices corresponding to orthogonal transformations of the vector consist of 0's and ±1's (the cosine and sine of $\pm\pi/2$). Therefore, the projections of a data to one dimension under this methodology can be completely determined by the entries of **q**, i.e., $\{p_1^{1/2},\ldots,p_C^{1/2}\}$, by shuffling them and modifying their signs as necessary (and perhaps replacing some with as determined by the appropriate rotation matrix). For example, in the case of 2 classes (C=2), the two solutions are $\boldsymbol{\alpha}=[-p_2^{1/2},p_1^{1/2}]^T$ and its negative, which is an equivalent solution. In the case of C=3, the three distinct solutions are given by $\boldsymbol{\alpha}=[-p_2^{1/2},p_1^{1/2},0]^T$, $\boldsymbol{\alpha}=[-p_3^{1/2},0, p_1^{1/2}]^T$, $\boldsymbol{\alpha}=[0,-p_3^{1/2},p_2^{1/2}]^T$. These solutions differ in their ordering of the projected classes on the projection axis and in general, the solution that also maximizes the numerator of the first term in (17) is preferable. The reason for this will become apparent in the next section.

Similar analytical expressions could be derived for candidate projections in the case of more than 3 classes, but the general iterative procedure proposed in the next section already considers these issues and constructs the solution without having to go through all possible rotations that result in orthogonal vectors in the C dimensional space. Nevertheless, for cases with few classes, these analytical solutions are very practical, since it only takes evaluating a portion of (17) for all candidate solutions and selecting the one that yields the maximum value. The function to be evaluated is specifically

$$\sum_{c=1}^{C} p_c \log\left|\alpha_c\right| p_c^{1/2} \tag{19}$$

D. Algorithm for Determining Optimal Projections to C or 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 complexity of this algorithm is relatively low.

We start by constructing the matrix $\mathbf{M} = [\boldsymbol{\mu}_1 \ldots \boldsymbol{\mu}_C]$, where the mean vector norms satisfy (15). 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 $\mathbf{s}^C=[1,\ldots,1]^T$ (for reasons that will become clear shortly). Using the elementwise multiplication operator •, we calculate $\mathbf{r}^C=\mathbf{s}^C \bullet \mathbf{p}$. The overall mean vector $\boldsymbol{\mu}^C$ is then given by $\boldsymbol{\mu}^C=\mathbf{M}^C\mathbf{r}^C$. 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:

$$\mathbf{M}^{C-1} = \left(\mathbf{I}_N - \boldsymbol{\mu}^C \boldsymbol{\mu}^{CT} / \| \boldsymbol{\mu}^C \|^2\right)\mathbf{M}^C \tag{20}$$

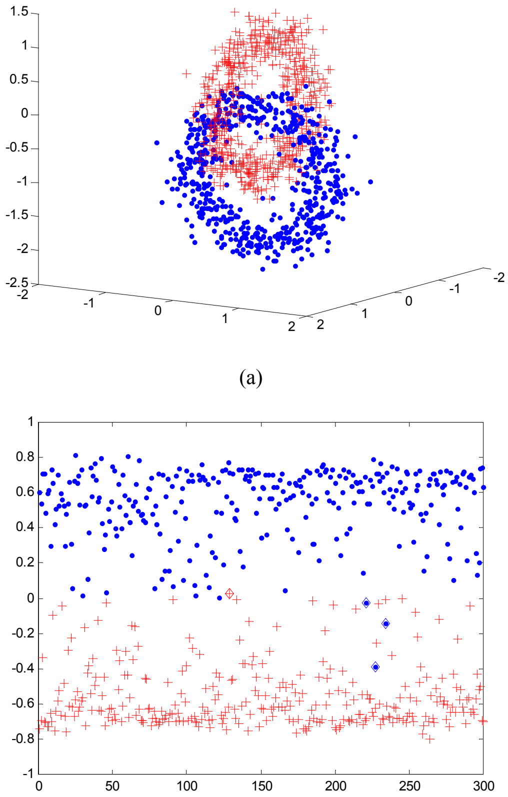
Any orthonormal bases that span the same space as the columns of the deflated matrix $\mathbf{M}^{C\text{-}1}$ is a valid candidate for the projection matrix **V** with C-1 orthonormal columns. Possible methods to obtain these bases is to employ Gram-Schmidt orthonormalization to the columns of $\mathbf{M}^{C\text{-}1}$ and determining the eigenvectors of $\mathbf{M}^{C\text{-}1}\mathbf{M}^{C\text{-}1,T}$ that correspond to the C-1 nonzero eigenvalues (which could be achieved sequentially). In the latter case, for example, the determined eigenvectors can be immediately assigned as **V**.

The procedure continues similarly for reducing dimensionality further: Construct $\mathbf{s}^{C\text{-}1}$, Calculate the mean vector in the deflated space using $\boldsymbol{\mu}^{C\text{-}1}=\mathbf{M}^{C\text{-}1}\mathbf{r}^{C\text{-}1}$, deflate the class means matrix using

$$\mathbf{M}^{C-2} = \left(\mathbf{I}_N - \boldsymbol{\mu}^{C-1} \boldsymbol{\mu}^{C-1,T} / \| \boldsymbol{\mu}^{C-1} \|^2\right)\mathbf{M}^{C-1} \tag{21}$$

As before, the orthonormal projection matrix **V** to C-2 dimensions is determined by finding the nonzero eigenvectors of $\mathbf{M}^{C\text{-}2}\mathbf{M}^{C\text{-}2,T}$. 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 **V**, which is N×D, is obtained previously unseen test samples can be transformed using



(a)

(b)

Figure 2. The original samples for both classes indicated by + and • signs are shown in (a). In (b), the values of the one-dimensional projection are shown for both classes with the same signs. The ◊ symbols indicate the classification errors made using a threshold on the projections values.

$$\boldsymbol{\varphi}(\mathbf{y}) = \sqrt{N}\mathbf{V}^T\boldsymbol{\Lambda}^{-1}\boldsymbol{\Phi}_{\mathbf{x}}\mathbf{k}(\mathbf{x}) \qquad (22)$$

Note that the procedure described here requires determining the larger eigenvectors of an $N\times N$ symmetric matrix at every step of the deflation process. 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 a suitable algorithm. 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 [17]. 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 [15,17]. In fact, in practice, 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.

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): $$\sigma^2 = \frac{1}{n}tr(\boldsymbol{\Sigma}_{\mathbf{x}})\left(4/((2n+1)N)\right)^{2/(n+4)}$$
- Construct the kernel matrix **K**, where $\mathbf{K}_{ij} = K(\mathbf{x}_i - \mathbf{x}_j)$.
- Decompose **K** into its eigenvectors and eigenvalues such that $\mathbf{K} = \boldsymbol{\Phi}_{\mathbf{x}}^T\boldsymbol{\Lambda}\boldsymbol{\Phi}_{\mathbf{x}}$.
- 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 $\mathbf{s}^{C\text{-}d}$=$[1,\ldots,1]^T$ in the first step, $s'^{C\text{-}d}_j$=sign($\boldsymbol{\mu}_j^T\mathbf{u}^{C\text{-}d}$). in the following steps.
  2. Calculate $\mathbf{r}^{C\text{-}d}$=$\mathbf{s}^{C\text{-}d}\bullet\mathbf{p}$ 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 • denotes elementwise vector product.)
  3. Construct the matrix $\mathbf{M}^{C-d} = [\boldsymbol{\mu}_1^{C-d} \ldots \boldsymbol{\mu}_C^{C-d}]$. If $C\text{-}d$ is the desired projection dimension, determine the eigenvectors of $\mathbf{M}^{C\text{-}d}\mathbf{M}^{C\text{-}d,T}$ that correspond to the $C\text{-}d$ nonzero eigenvalues. Assign these eigenvectors to **V**.
  4. Otherwise, perform the following deflation operation and go back to the first step: $$\mathbf{M}^{C-1} = \left(\mathbf{I}_N - \boldsymbol{\mu}^C\boldsymbol{\mu}^{CT} / \|\boldsymbol{\mu}^C\|^2\right)\mathbf{M}^C$$

IV. Experiments

In order to illustrate how the proposed nonparametric nonlinear projection scheme works, simulations using two datasets will be presented. The chain dataset is selected to demonstrate the effectiveness of the nonlinear projections obtained through this methodology in determining nonparametric projections to separate classes with nonlinear discriminant boundaries, and the matched filter example is chosen to motivate the use of these techniques as nonlinear filters.

Chain Dataset: Chain dataset consists of two interlocked and circular shaped classes with 300 three-dimensional samples for each class, uniformly distributed around the circle and perturbed around the circle with Gaussian distributed random values. This dataset is generated such that there is a nonlinear decision boundary between the classes, in order to eliminate the possibility of having a linear projection direction on which the classes become easily separable; hence, nonlinear projections are required here.

Sample simulation results using the chain dataset are presented in Fig. 2. The original data set is shown in Fig. 2a, and the values of the one-dimensional projection are presented in Fig. 2b. The errors based on the optimal threshold are indicated by diamonds.

Nonlinear matched filter: Interpreting the matched filter problem as a two-class clustering problem, we can use the given algorithm in order to use a projection to one dimension and distinguish between two possible cases, namely $r=n$ or $r=s+n$, where r is the received signal, n is the channel noise, and s is the signal to be detected. Since the linear matched filter is optimal under quite restrictive conditions such as linearity and Gaussianity, the nonlinear matched filter is strongly superior to the linear matched filter in the absence of these restrictions. In order to simulate the case that the optimal threshold is unknown, ROC curves can be used in order to evaluate the system performance. Under the assumption that the signal suffers a nonlinear distortion in the channel, ROC curves for the nonlinear matched filter are depicted in Fig. 4 along with the traditional linear matched filter for a comparison. In consistency with the literature on digital communications, the channel nonlinearity in this example is taken to be a third order polynomial [18]. As expected, an increase in the overlap, hence a decrease in SNR, results in *worse* ROC curves. Given the ROC, the optimal threshold for a given data set can be easily determined using a line passing from (0,1), and whose slope is determined by the ratio of a priori class probabilities.

V. Conclusions

In this paper, we have proposed a nonparametric nonlinear subspace projection methodology based on maximizing the Shannon mutual information between the projections and the class labels. Interpreting the nonparametric kernel estimator for mutual information 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 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, in contrast with the traditional parametric projection algorithms based on optimizing the same nonparametric MI estimate that 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.

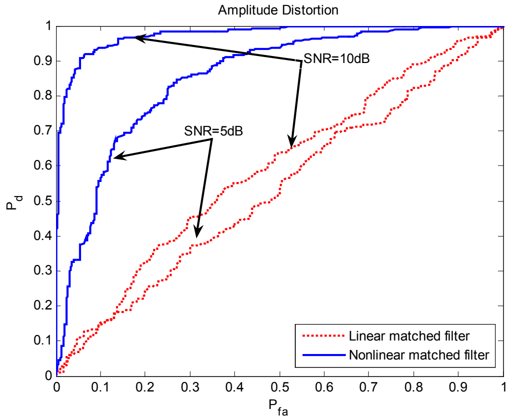


Figure 3. Performance comparison for signal detection in AWGN with nonlinear amplitude distortion. p_d and p_{fa} stand for probability of detection and probability of false alarm respectively.

References

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

[2] B. Scholkopf, A. Smola, K.R. Muller, "Nonlinear Component Analysis as a Kernel Eigenvalue Problem," Neural Computation, vol. 10, pp. 1299-1319, 1998.

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

[4] G. Baudat, F. Anouar, "Generalized Discriminant Analysis Using a Kernel Approach," Neural Computation, vol. 12, pp. 2385-2404, 2000.

[5] J.C. Principe, J.W. Fisher, D. Xu, "Information Theoretic Learning," in *Unsupervised Adaptive Filtering*, S. Haykin Editor, John Wiley & Sons, New York, 2000, pp.265-319.

[6] K. Torkkola, "Feature Extraction by Non-Parametric Mutual Information Maximization," Journal of Machine Learning Research, vol. 3, pp. 1415-1438, 2003.

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

[8] D. Erdogmus, *Information Theoretic Learning: Renyi's Entropy and its Applications to Adaptive System Training*, PhD Dissertation, University of Florida, Gainesville, Florida, 2002.

[9] E.G. Learned-Miller, J.W. Fisher III, "ICA Using Spacings Estimates of Entropy," Journal of Machine Learning Research, vol. 4, pp. 1271-1295, 2003.

[10] D. Erdogmus, J.C. Principe, "An Error-Entropy Minimization Algorithm for Supervised Training of Nonlinear Adaptive Systems," IEEE Transactions on Signal Processing, vol. 50, no. 7, pp. 1780-1786, 2002.

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

[12] M.E. Hellman, J. Raviv, "Probability of Error, Equivocation and the Chernoff Bound," IEEE Transactions on Information Theory, vol. 16, pp. 368-372, 1970.

[13] J. Mercer, "Functions of Positive and Negative Type, and Their Connection with the Theory of Integral Equations," Transactions of the London Philosophical Society A, vol. 209, pp. 415-446, 1909.

[14] H. Weinert (ed.), *Reproducing Kernel Hilbert Spaces: Applications in Statistical Signal Processing*, Hutchinson Ross Pub. Co., Stroudsburg, Pennsylvania, 1982.

[15] C. Fowlkes, S. Belongie, F. Chung, J. Malik, Spectral Grouping Using the Nystrom Method," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, pp. 298-305, 2004.

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

[17] G.H. Golub, C.F. van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, Maryland, 1996.

[18] X.N. Fernando, A.B. Sesay, "Nonlinear Channel Estimation Using Correlation Properties of PN Sequences," *Proc. IEEE Canadian Conference on Electrical and Computer Engineering*, pp. 469-474, 2001.