Information Theoretic Feature Selection and Projection

Deniz Erdogmus[1 2], Umut Ozertem[1] and Tian Lan[2]

[1] CSEE Department, Oregon Health and Science University, Portland, OR
[2] BME Department, Oregon Health and Science University, Portland, OR

1 Introduction

In pattern recognition, a classifier is trained solve the multiple hypotheses testing problem in which a particular input feature vector's membership to one of the classes is assessed. Given a finite number of training examples, feature dimensionality reduction to improve generalization and optimal exploitation of the information content in the feature vector regarding class labels is essential. Such dimensionality reduction enables the classifier to achieve improved generalization through: (*i*) eliminating redundant dimensions that do not convey reliable statistical information for classification, (*ii*) determining a manifold on which projections of the original high dimensional feature vector exhibit maximal information about the class label, and (*iii*) reducing the complexity of the classifier to help avoid over-fitting. In other words, feature dimensionality reduction through projections with various constraints can exploit salient features and eliminate irrelevant feature fluctuations by representing the discriminative information in a lower dimensional manifold embedded in the original Euclidean feature space.

In principle, the maximally discriminative dimensionality reduction solution is typically nonlinear - in fact, the minimum-risk Bayesian classifier can be interpreted as the optimal nonlinear dimensionality reduction mapping. However, given finite amount of training data, arbitrary robust nonlinear projections are challenging to obtain, thus there is wide interest in the literature in finding regularized nonlinear projections as well as simple linear projections. Furthermore, in some situations, such as real-time brain computer interfaces, it is even desirable to completely eliminate the computational and hardware requirements of evaluating the values of certain features, in which case feature selection - a special case of linear projections constrained to sparse orthonormal matrices - is required. Existing approaches for feature dimensionality reduction can be classified into the so-called *wrapper* and *filter* categories.

The wrapper approach aims to identify the optimal feature dimensionality reduction solution for a particular classifier, therefore involves repeatedly ad-

justing the projection network based on the cross validation performance of a corresponding trained classifier with the particular topology. This approach is theoretically the optimal procedure to find the optimal feature dimensionality reduction given a particular training set, a specific classifier topology, and a particular projection family. However, repeated training of the classifier and iterative learning of the projection network under this framework is computationally very expensive and could easily become unfeasible when the original feature dimensionality is very high and the number of training samples is large. The complexity is further increased by the cross-validation procedure and repeated training of the classifier to ensure global optimization, if particular topologies that are prone to local optima, such as neural networks, are selected. It is also widely accepted, and is intuitive, that some classification algorithms, such as decision tree, multi-layer perceptron neural networks have inherent ability to focus on relevant features and ignore irrelevant ones, when properly trained [1].

The filter approach provides a more flexible and computationally attractive approach at the expense of not identifying the optimal feature-classifier combination. This technique relies on training a feature projection network through the optimization of a suitable optimality criterion that is relevant to classification error/risk. Since the filter approach decouples the feature projection optimization from the following classifier-training step, this approach enables the designer to conveniently compare the performance of various classifier topologies on the reduced dimensionality features obtained through the filter approach. In this chapter, we will propose linear and nonlinear feature projection network training methodologies based on the filter approach. These projections will be optimized to approximately maximize the mutual information between the reduced dimensionality feature vector and the class labels. The selection of mutual information is motivated by information theoretic bounds relating Bayes probability of error for a particular feature vector and its mutual information with the class labels. Specifically, Fano's lower bound [2, 3, 4] provides a performance bound on the classifiers and more importantly the Hellman-Raviv bound [2, 3, 4], expressed as $p_e \leq (H(C) - I(\mathbf{x}; C))/2$ where C are the class labels corresponding to feature vectors $\mathbf{x}$, p_e denotes the minimum probability of error, which is obtained with a Bayes classifier, and H and I denote Shannon's entropy and mutual information, respectively. The entropy of the class labels depends only on the class priors. Consequently, maximizing the mutual information between the (projected) features and the class labels results in a dimensionality reduction that minimizes this tight bound on Bayes error. The maximum mutual information principle outlined here can be utilized to solve the following three general problems: determining optimal (*i*) feature ranking and selection, (*ii*) linear feature projections, and (*iii*) nonlinear feature projections that contain maximal discriminative information, minimal redundancy, and irrelevant statistical variations. Earlier work on utilizing mutual information to select input features for a classifier includes [5, 6, 7, 8].

Many feature selection and projection methods have been developed in the past years [9, 10, 11, 12, 13, 14, 15, 16, 17, 18]. Guyon and Elisseeff also reviewed several approaches used in the context of machine learning [19]. 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 technique is linear principle components analysis (PCA) [9, 10]. Although this technique is widely used, its shortcomings for pattern recognition are well known. A generalization to nonlinear projections, Kernel PCA [20], 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 [21]. Besides statistical independence, source sparsity and nonnegativity is also utilized as a statistical assumption in achieving dimensionality reduction through sparse bases, a technique called nonnegative matrix factorization (NMF) [22]. 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) [23]. Extensions of this approach to supervised local linear embeddings that consider class label information also exist [24]. Linear Discriminant Analysis (LDA) attempts to eliminate the shortcoming of PCA by finding linear projections that maximize class separability under the Gaussian distribution assumption [11]. 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 [25], 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. Nevertheless, empirical evidence suggests that robust nonparametric solutions to nonlinear problems in pattern recognition can be obtained by first projecting the data into a higher dimensional space (possibly infinite) determined by the eigenfunctions of the selected interpolation kernel. The regularization of the solution is achieved by the proper selection of the kernel. Torkkola [14] proposes utilizing quadratic density distance measures to evaluate an approximate mutual information measure to find linear and parametric nonlinear projections, based on the early work on information theoretic learning [26], and Hild et al [18] propose optimizing similar projections using a discriminability measure based on Renyi's entropy. The latter two proposals are based on utilizing the nonpara-

metric kernel density estimation (KDE) technique (also referred to as Parzen windowing) [27].

Estimating mutual information requires assuming a pdf estimate explicitly or implicitly. Since the data pdf might take complex forms, in many applications determining a suitable parametric family becomes a nontrivial task. Therefore, mutual information is typically estimated more accurately using nonparametric techniques [28, 29]. 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 multidimensional entropies of continuous random vectors. The 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 [29, 30, 31, 32, 33]. On the other hand, entropy estimators based on kernel density estimation (KDE) provide a differentiable alternative [28, 33, 34].

In this chapter we derive and demonstrate computationally efficient algorithms for estimating and optimizing mutual information, specifically for the purpose of learning optimal feature dimensionality reduction solutions in the context of pattern recognition. Nevertheless, the estimators could be utilized in other contexts. These techniques and algorithms will be applied to the classification of multichannel EEG signals for brain computer interface design, as well as sonar imagery for target detection. Results will be compared with widely used benchmark alternatives such as LDA and kernel LDA.

2 Nonparametric Estimators for Entropy and Mutual Information with a Discrete Variable

The probability distribution of a feature vector $\mathbf{x} \in \Re^n$ is a mixture of class distributions conditioned on the class label c :$p(\mathbf{x}) = \sum_c p_c p(\mathbf{x}|c)$. Maximizing mutual information between the projected features and the class labels requires implicitly or explicitly estimating this quantity. Shannon's mutual information between the continuous feature vector and the discrete class label can be expressed in terms of the overall Shannon entropy of the features and their average class conditional entropy: $I(\mathbf{x};c) = H(c) - \sum_c p_c H(\mathbf{x}|c)$. Specifically, the Shannon joint entropy for a random vector x and this vector conditioned on a discrete label c are defined as:

$$H(\mathbf{x}) = -\int p(\mathbf{x}) log(p(\mathbf{x}))d\mathbf{x}, \quad H(\mathbf{x}|c) = -\int p(\mathbf{x}|c) log(p(\mathbf{x}|c))d\mathbf{x} \qquad (1)$$

This demonstrates that estimating the mutual information between a continuous valued random feature vector and a discrete valued class label is relatively easy since only multiple joint entropy estimates (including conditionals)

with the dimensionality of the feature vector of interest need to be obtained. Given a finite labeled training data set $\{(\mathbf{x}_1, c_1), \ldots, (\mathbf{x}_N, c_N)\}$, where c_i takes one of the class labels as its value, the multidimensional entropy terms in (1) can be estimated nonparametrically with most convenience, although parametric and semiparametric estimates are also possible to obtain. The parametric approach assumes a family of distributions for the class conditional densities and utilizes Bayesian model fitting techniques, such as maximum likelihood. For iterative optimization of mutual information, this procedure is computationally very expensive, therefore not always feasible. Semiparametric approaches utilize a suitable truncated series expansion to approximate these distributions around a reference density (for instance, Hermite polynomial expansion around a multivariate Gaussian is commonly utilized in the independent component analysis literature and leads to the well known kurtosis criterion in that context) [21]. These estimates are accurate provided that the reference density is selected appropriately and the series converges fast, so that low-order truncations yield accurate approximations.

In this chapter, we will place the most emphasis on nonparametric estimators of entropy and mutual information due to their computational simplicity and versatile approximation capabilities. Various approaches that depend on pairwise sample distances and order statistics (such as ranked samples and minimum spanning trees). All of these approaches can in fact be explained as special cases of kernel density estimator based plug-in entropy estimation. Specifically, variable kernel size selection results in highly accurate density representations, therefore entropy estimates. For a sample set $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ the kernel density estimate with variable kernel size is:

$$p(\mathbf{x}) \approx N^{-1} \sum_{k=1}^{N} K_{\Sigma_k}(\mathbf{x} - \mathbf{x}_k) \tag{2}$$

where typical kernel functions in the literature are uniform (leads to K-nearest-neighbor sliding histogram density estimates) and Gaussian. Especially important is the latter, since the Gaussian kernel is commonly utilized in many kernel machine solutions to nonlinear regression, projection, and classification problems in machine learning:

$$G_{\Sigma}(\xi) = exp(-\xi^T \Sigma^{-1} \xi / 2) / \sqrt{(2\pi)^n |\Sigma|} \tag{3}$$

where n is the dimensionality of the kernel function.

In this chapter, parametric and semiparametric approaches will be briefly reviewed and sufficient detail will be provided for the reader to understand the basic formulations. In addition, detailed expressions for various nonparametric estimators will be provided, their theoretical properties will be presented, and their application to the determiniation of maximally informative and discriminative feature projections will be illustrated with many real datasets including EEG classification in the context of brain interfaces, neuron spike

detection from microelectrode recordings, and target detection in synthetic aperture radar imagery.

2.1 Parametric Entropy Estimation

The parametric approach relies on assuming a family of distributions (such as the Gaussian, Beta, Exponential, mixture of Gaussians, etc.) that parametrically describes each candidate distribution. The *optimal* pdf estimate for the data, is then determined using maximum likelihood (ML) or maximum *a posteriori* (MAP) statistical model fitting using appropriate regularization through model order selection criteria. For example, the ML estimate yields $p(x;\Theta_{ML})$ as the density estimate by solving

$$\Theta_{ML} = arg\max_{\Theta} \sum_{k=1}^{N} log\ p(\mathbf{x}_k;\Theta) \tag{4}$$

where $p(\mathbf{x}_k;\Theta)$ is the selected parametric family of distributions. The modeling capability of the parametric approach can be enhanced by allowing mixture models (such as mixture of Gaussians), in which case, the family of distributions is in the form:

$$p(\mathbf{x};\{\alpha_k,\Theta_k\} = \sum_{k=1}^{M} \alpha_k\ G(\mathbf{x},\Theta_k) \tag{5}$$

Once the density estimate is optimized as described, it is plugged-in to the entropy expression to yield the sample estimator for entropy, which is obtained in two-stages, since analytical expressions for the entropy of most parametric families is not available. Using the sample mean approximation for expectation, we obtain:

$$H(\mathbf{x}) \approx -\frac{1}{N}\sum_{j=1}^{N} logp(\mathbf{x}_j;\Theta_{ML}) \tag{6}$$

The difficulty with parametric methods in learning optimal feature projections is that it requires solving an optimization problem (namely ML or other model fitting procedure) within the main projection optimization problem. Another drawback of the parametric approach is the insufficiency of parametric models for general-purpose data modeling tasks. As the features in an arbitrary pattern recognition problem might exhibit very complicated structures, the selected parametric family for modeling might remain too simplistic to be able to accurately model all possible variations of the data distribution during learning. It can be shown that the ML density estimate asymptotically converges to the member of the parametric family that minimizes the KLD with the true underlying density [35].

An alternative approach to parametric estimation involves exploiting the maximum entropy principle. Consider the following constrained optimization problem that seeks the maximum entropy distribution given some nonlinear moments:

$$\max_{p(\mathbf{X})} = -\int p(\mathbf{x}) log\ p(\mathbf{x}) d\mathbf{x}\ sub.\ to \int p(\mathbf{x}) f_k(x) d\mathbf{x} = \alpha_k\ k = 0, 1, \ldots, m \quad (7)$$

where $f_0(\mathbf{x}) = 1$ and $\alpha_0 = 1$ 1 in order to guarantee normality. The solution to this maximum entropy density problem is in the form of an exponential: $p(\mathbf{x}) = exp(\lambda_0 + \sum_{k=1}^{m} \lambda_k f_k(\mathbf{x}))$, where the Lagrange multipliers can be approximated well by solving a linear system of equations if the true data distribution is close to the maximum entropy density with the same moments [36]. Further refinement can be obtained using these estimates as initial condition in a fixed-point equation solver for the constraints in (7). In this form, given the Lagrange multipliers, the entropy is easy to calculate:

$$H(\mathbf{x}) = -\int p(\mathbf{x}) \lambda_0 + \sum_{k=1}^{m} \lambda_k f_k(\mathbf{x}) d\mathbf{x} = -\lambda_0 - \sum_{k=1}^{m}\ \lambda_k \alpha_k \quad (8)$$

The nonlinear moment functions can be selected to be various polynomial series, such as Taylor, Legendre, or Hermite (the first one is the usual polynomials and the latter two are explained below).

2.2 Semiparametric Entropy Estimation

The semiparametric approach provides some additional flexibility over the parametric approaches as they are based on using a parametric density model as the reference point (in the Taylor series expansion) and additional flexibility is introduced in terms of additional series coefficients. Below, we present a few series expansion models for univariate density estimation including Legendre and Gram-Charlier series. Multidimensional versions become computationally infeasible due to the combinatorial expansion of cross terms and their associated coefficients [21].

Legendre Series Expansion: Consider the Legendre polynomials defined recursively as follows using the initial polynomials $P_0(x) = 1$ and $P_1(x) = x$:

$$P_k(x) = \frac{1}{k}[(2k-1)x\ P_{k-1}(x) - (k-1)P_(k-2)(x)]\ \ k \geq 2 \quad (9)$$

Any pdf (satisfying certain continuity and smoothness conditions) can expressed in terms of the Legendre polynomials and in terms of polynomial statistics of the data.

$$q(x) = \sum_{k=0}^{\infty} (k + \frac{1}{2}) E[P_k(X)] P_k(x) \quad (10)$$

The expectations can be approximated by sample mean approximations to obtain an approximation from a finite sample set.

Gram-Charlier Series Expansion: The characteristic function of a distribution $q(x)$, denoted by $\Phi_q(w)$, can be expressed in terms of characteristic function $\Phi_r(w)$ of an *arbitrary* reference distribution $r(x)$ as

$$\Phi_q(w) = exp(\sum_{k=1}^{\infty}(c_{q,k} - c_{r,k})\frac{(jw)^k}{k!})\Phi_r(w) \tag{11}$$

where $c_{q,k}$ and $c_{r,k}$ cumulants of $q(x)$ and $r(x)$,), respectively. The cumulants are expressed in terms of the moments using the Taylor series expansion of the cumulant generating function, defined as $\Psi(w) = log\Phi(w)$.

For the special case of a zero-mean and unit-variance Gaussian distribution as the reference pdf (denoted by $G(x)$ below), expanding the series and collecting the coefficients of same order derivatives together leads to the following expression in terms of the Hermite polynomials and polynomial moments as before.

$$q(x) = G(x)\sum_{k=0}^{\infty}\frac{E[H_k(X)]}{k!}H_k(x) \tag{12}$$

The Hermite polynomials are obtained using the initial polynomials $H_0(x) = 1$, $H_1(x) = x$, and the recursion

$$H_k(x) = x\ H_{k-1}(x) - (k-1)H_{k-2}(x) \quad k \geq 2 \tag{13}$$

Expansion on Complete Bases: All pdfs (satisfying the usual continuity and smoothness conditions) can be approximated by a truncated linear combination of basis functions (preferably orthonormal). Given infinitely many orthonormal bases $\{b_1(x), b_2(x), \ldots\}$, it is possible to express an arbitrary pdf in terms of the following linear combination.

$$q(x) = \sum_{k=1}^{\infty}E[b_k(X)]b_k(x) \tag{14}$$

An abundance of orthonormal bases for Hilbert spaces can be found in the function approximation literature (the eigenfunctions of any reproducing kernel forms a bases for the pdf space. A drawback of the series approximations is that the truncation leads to pdf estimates that do not satisfy the two basic conditions for a function to be a probability distribution: nonnegativity and integration to unity [37].

2.3 Nonparametric Entropy Estimates

In the remainder of this chapter we will focus on nonparametric estimates, especially the following two that emerge naturally from plug-in entropy estimation utilizing a variable-size kernel density estimate; namely, sample-spacing

estimate and Parzen estimate. For illustration purposes, the former will be presented for univariate entropy estimation and extensions to multidimensional cases will be discussed.

The most straightforward nonparametric approach in entropy estimation, usually leading to poor estimates, yet surprisingly utilized frequently in the, is to consider a histogram approximation for the underlying distribution. Fixed-bin histograms lack the flexibility of sliding histograms, where the windows are placed on every sample. A generalization of sliding histograms is obtained by relaxing the rectangular window and assuming smoother forms that are continuous and differentiable (and preferably symmetric and unimodal) pdfs. This generalization is referred to as kernel density estimation (KDE) and is shown in (2). Another generalization of histograms is obtained by letting the bin-size vary in accordance with local data distribution. In the case of rectangular windows, this corresponds to nearest neighbor density estimation [38], and for KDE this means variable kernel size [38, 39]. The corresponding entropy estimates are presented below.

Entropy Estimation Based on Sample Spacing: Suppose that the ordered samples $\{x_1 < x_2 < \ldots < x_N\}$ drawn from $q(x)$ are provided. We assume that the distribution is piecewise constant in m-neighborhoods of samples [31], leading to the following approximation:

$$p(x) = (N+1)^{-1}(x_{i+1} - x_i), i = 0, \ldots, N \tag{15}$$

Denoting the corresponding empirical cdf by $P(x)$, for ordered statistics, it is known that

$$E[P(x_{i+m} - P(x_i))] = \frac{m}{N+m}, \quad i = 1, \ldots, N-m \tag{16}$$

where the expectation is evaluated with respect to the joint data distribution $q(x_1), \ldots, q(x_N)$, assuming iid samples. Substituting this in entropy, we obtain the m-spacing estimator as

$$H(x) \approx -\frac{1}{N-m} \sum_{i=1}^{N-m} log((N+1)(x_{i+m} - x_i)/m) \tag{17}$$

The spacing interval m is chosen to be a slower-than-linear increasing function of N in order to guarantee asymptotic consistency and efficiency. Typically, $m = N^{1/2}$ is preferred in practice due to its simplicity, but other roots are viable. A difficulty with the sample spacing approach is its generalization to higher dimensionalities. Perhaps the most popular extension of sample-spacing estimators to multidimensional random vectors is the one based on the minimum spanning tree recently popularized in the signal processing community by Hero [30]. This estimator relies on the fact that the integral in Renyi's definition [40] of entropy is related to the sum of the lengths of the edges in the minimum spanning tree with useful asymptotic convergence guarantees. One drawback of this approach is that it only applies to entropy orders of

$0 < \alpha < 1$ (Shannon entropy is the limiting case as a approaches 1). Another drawback is that finding the minimum spanning tree itself is a computationally cumbersome task that is also prone to local minima due to the heuristic selection of a neighborhood search radius by the user in order to speed-up.

Another generalization of sample spacing estimates to multi-dimensional entropy estimation has relied on the L1-norm as the distance measure between the samples instead of the usual Euclidean norm [39]. This technique can, in principle, be generalized to arbitrary norm definitions. The drawback of this method is its nondifferentiability, which renders it next to useless for traditional iterative gradient-based adaptation, but could be useful for feature ranking. This approach essentially corresponds to extending (15) to the multidimensional case as data-dependent variable-volume hyperrectangles. One could easily make this latter approach differentiable through the use of smooth kernels rather than rectangular volumes. Such modification will also form the connection between the sample-spacing methods and kernel based methods described next.

Parzen Windowing Based Entropy Estimation: Kernel density estimation is a well-understood and useful nonparametric technique that can be employed for entropy estimation in the plug-in estimation framework [33]. For a given set of iid samples $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ the variable size KDE is given in (2). To simplify computational requirements, fized size isotropic kernels could be utilized by assuming $\Sigma_k = \sigma^2\mathbf{I}$ for all samples. The kernel function and its size can be optimized in accordance with the ML principle [42, 43] or other rules-of-thumb could be employed to obtain approximate optimal parameter selections [41, 39]. Given a suitable kernel function and size, the corresponding plug-in entropy estimate is easily obtained to be:

$$H(\mathbf{x}) = -\frac{1}{N}\sum_{j=1}^{N} log\frac{1}{N}\sum_{i=1}^{N} K_{\Sigma_i}(\mathbf{x}_j - \mathbf{x}_i) \tag{18}$$

Next, we demonstrate how these estimators can be utilized to design maximally discriminative linear and nonlinear feature projections via maximization of mutual information based on nonparametric estimators.

3 Linear and Nonlinear Feature Projection Design via Maximum Mutual Information

In this section we present two different techniques for determining linear and nonlinear projections respectively. For finding optimal linear projections, including feature selection, we will employ ICA decomposition in conjunction with the m-spacing estimator given in (17). For determining optimal nonlinear projections, we will employ the more general KDE-based plug-in estimate given in (18). In both cases, we assume that independent and identically distributed (iid) training data of the form $\{(\mathbf{x}_1, c_1), \ldots, (\mathbf{x}_N, c_N)\}$, where $x_i \in \Re^n$ is available.

3.1 Linear Feature Projections

Given the training data, we seek to determine a maximally informative linear feature projection $\mathbf{y}=\mathbf{A}\mathbf{x}$ from n to m dimensions that is characterized by a projection matrix $A \in \Re^{m \times n}$. Recently, methods based on optimizing this matrix via direct maximization of an approximation to the mutual information $I(\mathbf{y}, c)$ [14, 18]. These methods are based on slow iterative updates of the matrix due to the fact that at every update the gradient or other suitable update for the matrix must be computed using double-sum pairwise averages of samples due to the form in (18) that arises from the plug-in formalism. Stochastic gradient updates are a feasible tool to improve speed by reducing the computational load at each iteration, yet they may still not be sufficiently fast for very high dimensional scenarios.

We have recently proposed an approximation to this procedure by assuming that the class-mixture and class-conditional distributions of the given features obey the linear-ICA generative statistical model. Under this assumption, each class distribution as well as the mixture data density can be linearly brought to a separable form consisting of *independent* feature coordinates. This assumption is realistic for circularly symmetric class distributions, as well as elliptically symmetric class distributions where the independent axes of different classes are aligned, such that all classes can be separated into independent components simultaneously. In other cases, the assumption will fail and result in a biased estimate of the mutual information and therefore the optimal projection.

Under the assumption of linear separability of class distributions (note that this is not traditional linear separability of data points), we can obtain an independent linearly transformed feature coordinate system: $\mathbf{y}=\mathbf{W}\mathbf{x}$, where $\mathbf{W}$ is optimized using a suitable linear ICA algorithm on the training data $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ [21]. Under the assumption of overall and class-conditional independence, the mutual information of $\mathbf{y}$ with c, can be decomposed into the sum of mutual informations between each marginal of $\mathbf{y}$ and c:

$$I(\mathbf{y}; c) \approx \sum_{d=1}^{n} I(y_d; c) \tag{19}$$

Each *independent* projection can now be ranked in terms of its individual contribution to the total discriminative information by estimating $I(y_d; c)$. From the training data, one obtains: $\mathbf{y}_i = \mathbf{W}\mathbf{x}_i$ and the samples y_d are$\{\mathbf{y}_{d1}, \ldots, \mathbf{y}_{dN}\}$. Employing the m-spacing estimator in (17), we obtain:

$$\begin{array}{c} H(y_d) \approx -\frac{1}{N-m} \sum_{i=1}^{N-m} log((N+1)(y_{d(i+m)} - y_{di})/m) \\ H(y_d|c) \approx -\frac{1}{N_c-m_c} \sum_{i=1}^{N_c-m_c} log((N_c+1)(y_{d(i+m_c)} - y_{di})/m_c) \\ I(y_d; c) = H(y_d) - \sum_c p_c H(y_d|c) \end{array} \tag{20}$$

Suppose that the following ranking is obtained: $I(y_(1); c) > I(y_(2); c) > \ldots > I(y_(n); c)$. Then the rows of $\mathbf{W}$ corresponding to the top m marginals $y_(1), \ldots, y_(m)$ are retained as the linear projection matrix.

3.2 Feature Subset Selection

Consider the mixture density of the features: $p(\mathbf{z}) = \sum_c p_c p(\mathbf{z}|c)$. Assuming that $\mathbf{W}$ and $\mathbf{W}^c$ are the linear ICA solutions (separation matrices) for $p(\mathbf{z})$ and $p(\mathbf{z}|c)$ respectively, let $\mathbf{y}$=$\mathbf{W}\mathbf{z}$ and $\mathbf{y}^c = \mathbf{W}^c\ \mathbf{z}^c$, where z^c is a random vector distributed according to $p(\mathbf{z}|c)$ and $\mathbf{z}$ is a random vector drawn from $p(\mathbf{z})$. It can be shown that the following identities hold:

$$H(\mathbf{z}) = \sum_{d=1}^{m} H(y_d) - log|\mathbf{W}| - I(\mathbf{y}) \quad H(\mathbf{z}|c) = \sum_{d=1}^{m} H(y_d^c) - log\|\mathbf{W}^c\| - I(\mathbf{y}^c) \tag{21}$$

The residual mutual information due to imperfect ICA solutions are denoted by $I(\mathbf{y})$ and $I(\mathbf{y}^c)$. Under the assumption of linear separability (in the ICA sense mentioned above rather than in the traditional meaning), these residual mutual information terms will (be assumed to) become zero. Since mutual information is decomposed into class-conditionals entropies as $I(\mathbf{z};c) = H(\mathbf{z}) - \sum_c p_c H(\mathbf{z}|c)$, given linear ICA solutions $\mathbf{W}$ and $\mathbf{W}_c$, we obtain the following decomposition of mutual information:

$$I(\mathbf{z};c) = \sum_{d=1}^{m}(H(y_d) - \sum_c p_c H(y_d^c)) + (log|\mathbf{W}| - \sum_c p_c log|\mathbf{W}^c|) + (I(y) - \sum_c p_c I(\mathbf{y}^c)) \tag{22}$$

Given any subset of features selected from the components (marginals) of $\mathbf{x}$, the high-dimensional feature vectors that needs to be reduced by selection, (22) can be utilized to estimate the mutual information of this subset, denoted by $\mathbf{z}$, with the class labels, by assuming that linear ICA solutions $\mathbf{W}$ and $\mathbf{W}^c$ is obtained using a suitable algorithm [21] and the training samples corresponding to each class (or the whole set). Further, the bias is assumed to be zero: $I(\mathbf{y}) - \sum_c p_c I(\mathbf{y}^c) = 0$.[3] The feature subset can be performed by evaluating the mutual information between all possible subsets (there are 2^n of them) or utilizing a heuristic/greedy ranking algorithm to avoid the combinatorial explosion of subsets to consider. For ranking, forward (additive), backward (subtractive), or forward/backward strategies can be employed. In the purely forward strategy we perform the following additive ranking iteration:

Initialization: Let the Unranked Feature Set (UFS) be $\{x_1, \ldots, x_n\}$ and the *Ranked Feature Set* be empty. *Ranking iterations*: For d from 1 to n perform the following. Let *Candidate Set* i (CSi) be the union of RFS and x_i, evaluate the MI $I(CSi;C)$ between the features in the candidate set and the class labels for every x_i in UFS. Label the feature x_i with the highest

[3] Optionally, computational complexity can be further reduced by assuming that all class-conditional and class mixture densities can be linearly separated by the same ICA solution $\mathbf{W}$, such that $\mathbf{W} = \mathbf{W}^{(c)}$ for all c. This additional assumption would also eliminate need to include the correction terms depending on the separation matrices.

$I(CSi, C)$ as $x_{(d)}$, redefine RFS as the union of RFS and $x_{(d)}$, and remove the corresponding feature from UFS.

Alternatively, a purely backward strategy would iteratively remove the least informative features from UFS and include in the RFS as the worst features that should be eliminated. A better alternative to both approaches is to initialize as above and allow both additive and subtractive (replacing) operations to the RFS such that earlier ranking mistakes can be potentially corrected at future iterations.

The main drawback of the linear ICA approach presented in these sections is the high possibility of the feature distributions not following the main underlying assumption of linear separability. To address this issue, linear ICA can be replaced with nonlinear ICA, especially local linear ICA, which has the flexibility of nonlinear ICA in modelling independent nonlinear coordinates in a piecewise linear manner, and the simplicity of linear ICA in solving for optimal separation solutions. The application of local linear ICA essentially follows the same procedures, except the whole process is initialized by a suitable partitioning of the data space using some vector quantization or clustering algorithm (for instance k-means or mean shift clustering could be utilized).

Due to the additivity of Shannon entropy, if the data space is partitioned into P nonoverlapping bur complementary regions, the entropies and mutual information become additive over these regions:

$$\begin{aligned} H(\mathbf{z}) &= \textstyle\sum_p q_p H(z^{(p)}) \\ H(\mathbf{z}|c) &= \textstyle\sum_p q_p H(\mathbf{z}^{(p)}|c) \\ I(\mathbf{z};c) &= \textstyle\sum_p q_p I(z^{(p)};c) \end{aligned} \tag{23}$$

where q_p is the probability mass of partition p (number of samples in partition p divided by the total number of samples). Within each partition, we still have $I(\mathbf{z}^{(p)};c) = H(\mathbf{z}^{(p)}) - \sum_c q_{pc} H(\mathbf{z}^{(p)}|c)$, thus (21) and (22) can be employed on each partition separately and summed up to determine the total mutual information.

3.3 Smooth Nonlinear Feature Projections

In order to derive a nonparametric nonlinear feature projection, consider the following equivalent definition of Shannon's mutual information and the KDE plug-in estimate with some positive semidefinite kernel $K(.)$:

$$I_S(\mathbf{z};c) = \sum_c p_c E_{\mathbf{z}|c}\left[log\frac{p_{\mathbf{z}|c}(\mathbf{z}|c)}{p_{\mathbf{z}}(\mathbf{z})}\right] \approx \sum_c \frac{p_c}{N_c}\sum_{j=1}^{N_c} log\frac{(1/N_c)\sum_{i=1}^{N} K(\mathbf{z}_j^c - \mathbf{z}_i^c)}{(1/N_c)\sum_{i=1}^{N} K(\mathbf{z}_j^c - \mathbf{z}_i)} \tag{24}$$

where $\mathbf{z}_1, \ldots, \mathbf{z}_N$ is the training set and its subset corresponding to class c is $\mathbf{z}_1^c, \ldots, \mathbf{z}_N^c$. According to the theory of reproducing kernels for Hilbert spaces

(RKHS), the eigenfunctions $\overline{\varphi}_1(\mathbf{z}), \overline{\varphi}_2(\mathbf{z}), \ldots$ collected in vector notation as $\overline{\boldsymbol{\varphi}}(\mathbf{z})$, of a kernel function K that satisfy the Mercer conditions [44] form a basis for the Hilbert space of square-integrable continuous and differentiable nonlinear functions [45, 46]. Therefore, every smooth nonlinear transformation $g_d(\mathbf{x})$ in this Hilbert space can be expressed as a linear combination of these bases:

$$y_d = g_d(\mathbf{z}) = \mathbf{v}_d^T \, \overline{\boldsymbol{\varphi}}(\mathbf{z}) \tag{25}$$

where y_d is the d^{th} component of the projection vector $\mathbf{y}$. For a symmetric positive semidefinite, translation invariant, and nonnegative (since we will establish a connection to KDE) kernel, we can write

$$K(\mathbf{z}-\mathbf{z}') = \sum_{k=1}^{\infty} \overline{\lambda}_k \overline{\varphi}_k(\mathbf{z}) \overline{\varphi}_k(\mathbf{z}') = \overline{\boldsymbol{\varphi}}^T(\mathbf{z}) \overline{\boldsymbol{\Lambda}} \overline{\boldsymbol{\varphi}}^T(\mathbf{z}') \geq 0 \tag{26}$$

Notice that for a nonnegative kernel, kernel induced feature space (KIFS) defined by the $\overline{\boldsymbol{\varphi}}(\mathbf{z})$ 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 p radians. This is a crucial observation for the proper geometrical interpretation of what follows. substituting (26) into (24), we get:

$$I_S(\mathbf{z};c) \approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} log \left[\frac{N \overline{\boldsymbol{\varphi}}^T(\mathbf{z}_j^c) \overline{\boldsymbol{\Lambda}} \boldsymbol{\Phi}_{\mathbf{Z}} \mathbf{m}_c}{N \overline{\boldsymbol{\varphi}}^T(\mathbf{z}_j^c) \overline{\boldsymbol{\Lambda}} \boldsymbol{\Phi}_{\mathbf{Z}} \mathbf{1}} \right] \tag{27}$$

where $\mathbf{m}_c i = 1 \; if \; c_i = c, \; 0 \; otherwise$, $\mathbf{1}$ is the vector of ones, $N = N_1 + \ldots + N_C$, and $p_c = N_c/N$. The matrix $\boldsymbol{\Phi}_{\mathbf{Z}} = [\overline{\boldsymbol{\varphi}}(\mathbf{z}_1) \cdots \overline{\boldsymbol{\varphi}}(\mathbf{z}_N)]$. The class-average vectors in the KIFS are $\overline{\boldsymbol{\mu}}_c = (1/N_c) \overline{\boldsymbol{\Phi}}_{\mathbf{Z}} \mathbf{m}_c$ and for the whole data it is $\overline{\boldsymbol{\mu}} = (1/N_c) \overline{\boldsymbol{\Phi}}_{\mathbf{Z}} \mathbf{1}$. Substituting these:

$$I_S(\mathbf{z};c) \approx \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} log \left[\frac{N \overline{\boldsymbol{\varphi}}^T(\mathbf{z}_j^c) \overline{\boldsymbol{\Lambda}} \boldsymbol{\mu}}{N \overline{\boldsymbol{\varphi}}^T(\mathbf{z}_j^c) \overline{\boldsymbol{\Lambda}} \boldsymbol{\mu}} \right] \tag{28}$$

Consider a projection dimensionality of m; we have $\mathbf{y} = \mathbf{V}_T \overline{\boldsymbol{\varphi}}(\mathbf{x})$, where $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$ consists of orthonormal columns $\mathbf{v}_d$. Note that the orthonormality constraint of any linear projection is reasonable because any full rank linear transformation can be written as the product of an orthonormal matrix times an invertible arbitrary linear transformation (due to the existence of singular value decomposition with nonzero eigenvalues). The arbitrary transformation does not change the information content of the projection, thus can be omitted. The back-projection of y to the KIFS is

$$\widetilde{\boldsymbol{\varphi}}(\mathbf{z}) = \mathbf{V}\mathbf{V}^T \overline{\boldsymbol{\varphi}}(\mathbf{z}) \tag{29}$$

The eigenfunctions of the kernel are not explicitly known in practice typically, therefore, we employ the common Nystrom approximation [47], $\overline{\boldsymbol{\varphi}}(\mathbf{z}) \approx \boldsymbol{\varphi}(\mathbf{z}) = \sqrt{N} \boldsymbol{\Lambda}^{-1} \boldsymbol{\Phi}_{\mathbf{Z}} \mathbf{k}(\mathbf{z})$, where $\mathbf{k}(\mathbf{z}) = [K(\mathbf{z}-\mathbf{z}_1), \ldots, K(\mathbf{z}-\mathbf{z}_N)]^T$ and the

eigendecomposition $K = \boldsymbol{\Phi}_{\mathbf{Z}}^T \boldsymbol{\Lambda} \boldsymbol{\Phi}_{\mathbf{Z}}$ of the data affinity matrix whose entries are $K_{ij} = K(\mathbf{z}_i - \mathbf{z}_j)$ provide the other necessary terms. Combining (29) with this approximation and substituting in (28) leads to the following cost function that needs to be maximized by optimizing an orthonormal $\mathbf{V} \in \Re^{N \times m}$:

$$J(\mathbf{V}) = \sum_c \frac{p_c}{N_c} \sum_{j=1}^{N_c} log \left[\frac{\boldsymbol{\varphi}^T(\mathbf{x}_j) \mathbf{V}\mathbf{V}^T \boldsymbol{\Lambda} \mathbf{V}\mathbf{V}^T \boldsymbol{\mu}_c}{\boldsymbol{\varphi}^T(\mathbf{x}_j) \mathbf{V}\mathbf{V}^T \boldsymbol{\Lambda} \mathbf{V}\mathbf{V}^T \boldsymbol{\mu}} \right] \tag{30}$$

where $\boldsymbol{\mu}_c = (1/N_c)\boldsymbol{\Phi}_{\mathbf{Z}}\mathbf{m}_c$ and $\boldsymbol{\mu} = (1/N)\boldsymbol{\Phi}_{\mathbf{Z}}\mathbf{1}$ are the class and overall mean vectors of the data in the $\boldsymbol{\Phi}$-space. Note that $\boldsymbol{\mu} = p_1\boldsymbol{\mu}_1 + \ldots + p_C\boldsymbol{\mu}_C$ and with the approximation, we have $\mathbf{y} = \mathbf{V}^T\boldsymbol{\varphi}(\mathbf{x})$.

By observation, (30) is seen to be maximized by any orthonormal matrix $\mathbf{V}$ whose columns span the intersection of the subspace orthogonal to $\boldsymbol{\mu}$ and $span(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_C)$ [48]. This also points out the fact that any projection that leads to a reduced dimensionality of more than *C-1*, where C is the number of classes, is redundant. It is also possible to find the analytical solution for the optimal projection to *C-1* dimensions or less. Let $\mathbf{M} = [\boldsymbol{\mu}_1 \ldots \boldsymbol{\mu}_C]$, note that $\mathbf{M}^T\mathbf{M} = \mathbf{P}^{-1}$ with $\mathbf{p} = [\mathbf{p}_1, \ldots, \mathbf{p}_C]$ and $\mathbf{P} = diag(\mathbf{p})$. We observe that $\boldsymbol{\mu} = \mathbf{M}\mathbf{p}$ is unit-norm and

$$\mathbf{V} = \mathbf{M} - \boldsymbol{\mu}(\boldsymbol{\mu}^T\mathbf{M}) = \mathbf{M} - \boldsymbol{\mu}(\mathbf{p}^T\mathbf{M}^T\mathbf{M}) = \mathbf{M} - \boldsymbol{\mu}\mathbf{1}^T \tag{31}$$

for a *C-1* dimensional projection. For lower dimensional projections, deflation can be utilized and the procedure can be found in [48].

Special case of 2-classes: We illustrate the analytical solution for the case of projection to a single dimension in the case of two classes. We parameterize the projection vector as $\mathbf{v} = \mathbf{M}\mathbf{P}^{-1/2}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}^T\boldsymbol{\alpha} = 1$ (so that $\mathbf{v}^T\mathbf{v} = 1$). It can be found that the optimal solution is provided by $\boldsymbol{\alpha} = [-p_2^{1/2}, p_1^{1/2}]^T$ (and its negative yields a projection equivalent in discriminability where the two class projections are flipped in sign). For the projection, the natural threshold is zero since the data mean is projected to this point (due to the fact that v is orthogonal to $\boldsymbol{\mu}$).

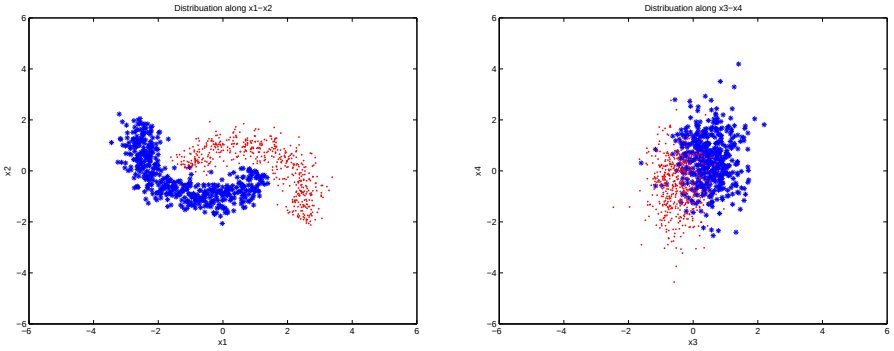
4 Experimental Evaluation and Comparisons

In this section, we present the experimental evaluation for the methods explained above and we also provide comparison with LDA and kernel LDA. The first example is an illustrative toy example, and the preceding two experiments are challenging problems performed on real data.

4.1 Synthetic Dataset

This dataset consists of four features: x_i $i = 1, \ldots, 4$, where x_1 and x_2 are nonlinearly related, x_3 and x_4 are independent from the first two features and

are linearly correlated Gaussian-distributed with different mean and variance. There are two classes in this dataset represented with different markers in Figure 1a and 1b. Forming an almost separable distribution with a nonlinear separation boundary in the x_1 and x_2 plane, and overlapping in the x_3 and x_4 plane, this dataset forms a good example to compare linear and nonlinear methods.



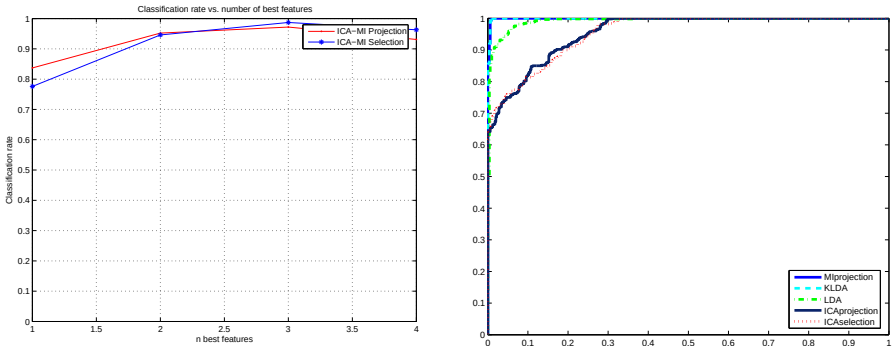
(a) x_1 vs x_2 (b) x_3 vs x_4

Fig. 1. The synthetic dataset

For ICA feature projection and selection, we use Support Vector Machine (SVM) to classify them. For the SVM, we use Chang and Lin's library toolbox. Based on the experiment results, we select the parameter of SVM as: penalty parameter c=10, and kernel size g=10. We apply ICA-MI feature projection and ICA-MI feature selection, nonlinear MI projection methods on the dataset, as well as LDA and kernel LDA. Each class contains 500 samples and we divide the dataset into five equal parts, four of which are used as training simples, one of which is used as testing samples. Figure 2a shows the classification accuracy vs. number of n best features, the Figure 2b presents a comparison of five above mentioned methods. To present a fair comparison, while comparing with other methods, we consider one dimensional projections for ICA based methods.

4.2 Brain Computer Interface Dataset

In this experiment, we apply the same five methods on Brain Computer Interfaces Competition III dataset V. This dataset contains human brain EEG data from 3 subjects during 4 non-feedback sessions. The subject is sitting on a chair, relaxed arms resting on their legs and executed one of the three tasks:



(a) classification performance vs dimensions

(b) Performances regarding to different methods

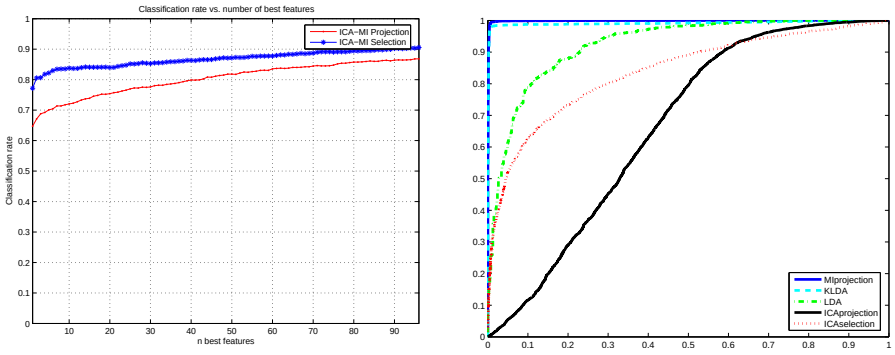
Fig. 2. The performance evaluation on synthetic dataset

imagination of left hand movements; imagination of right hand movements; generation of words beginning with the same random letter. The EEG data were collected during the sessions. The data of all 4 sessions of a given subject were collected on the same day, each lasting 4 minutes with 5-10 minutes breaks in between. We want to classify one of the three tasks from the EEG data. The raw EEG data contains 32 channels at 512 Hz sampling rate. The raw EEG potentials were first spatially filtered by means of a surface Laplacian. Then, every 62.5 ms, the power spectral density (PSD) in the band 8-30 Hz was estimated over the last second of data with a frequency resolution of 2 Hz for the 8 centro-parietal channels C3, Cz, C4, CP1, CP2, P3, Pz, and P4. As a result, an EEG sample is a 96-dimensional vector (8 channels times 12 frequency components).

To be able to present the results in a ROC curve we only use the first the classes in the dataset. We also mix the data from all sessions together, then and use five-fold cross validation as in the previous experiment. The classification performance vs. number of selected features, and comparison of one dimensional projections by different methods are presented in Figure 3a, and 3b, respectively.

4.3 Sonar Mine Detection

The mine detection dataset consists of sonar signals bounced off either a metal cylinder or a roughly cylindrical rock. Each sonar reflection is represented by a 60-dimensional vector, where each dimension represents the energy that falls within a particular energy band, integrated over a certain period of time. There are 208 60-dimensional sonar signals in this dataset, 111 of them belongs



(a) classification performance vs dimensions

(b) Performances regarding to different methods

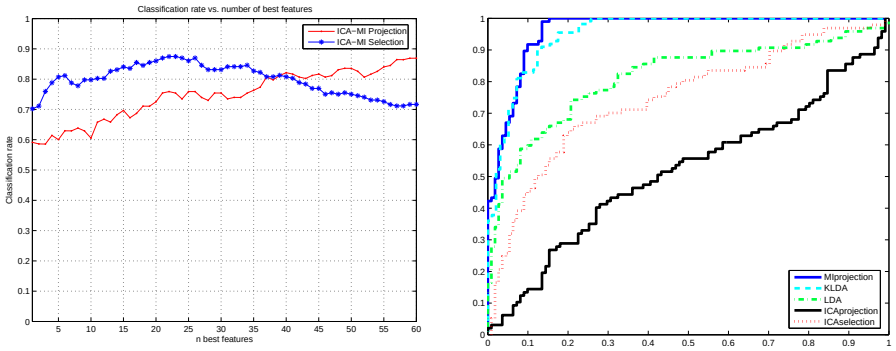
Fig. 3. The performance evaluation on BCI dataset

to mines and 97 of them obtained by bouncing sonar signals from cylindrical rocks under similar conditions. These sonar signals are collected from a variety of different aspect angles, and this dataset was originally used by Gorman and Sejnowski in their study of sonar signal classification [49]. The dataset is available in UCI machine learning repository [50].

As in the previous experiments, here we compare five different methods: MI Projections, ICA feature selection, ICA feature projection, LDA, and Kernel LDA. As in the previous experiments, for all these methods, we present the results of five-fold cross validation, where the class a priori probabilities in the bins are selected according to the a priori probabilities in the dataset. The results for projections into a single dimension is presented with a ROC curve, whereas the performance increase of the ICA based linear methods for different dimensions are presented separately. Due to the nonlinear structure of the optimal class separation boundary, nonlinear methods show superior performance in this experiment. Figure 4a presents the performance of ICA based methods for different number of dimensions, and a comparison of one dimensional projections with all five methods is presented in Figure 4b.

5 Conclusions

We presented and compared several information theoretic feature selection an projection methods. Selection and projection methods based on ICA are either linear or locally linear methods, which are simply analyzable. As seen from the original feature space, the mutual information projection method is nonlinear and not easy to analyze. Although it is hard in the original input space,



(a) classification performance vs dimensions

(b) Performances regarding to different methods

Fig. 4. The performance evaluation on sonar mine detection dataset

this method is also simple to analyze in the KIFS, where the MI projection becomes a linear method with the use of kernel trick.

MI projection method also provided similar -sometimes slightly better- performance as compared to widely used KLDA method. At this point, note that KLDA is known to be numerically unstable if there are not enough data points, and MI projection method provides same/better performance with an analytical solution. Among the information theoretic methods presented here, MI projection methods provided the best performance, and it also does not suffer from the input data dimensionality, whereas methods based on ICA transformation severely lose accuracy with increasing input dimensionality in the estimation of the inverse mixing matrix.

6 Acknowledgements

This work is partially supported by NSF grants ECS-0524835, and ECS-0622239.

References

1. Duch W, Wieczorek T, Biesiada J, Blachnik M (2004) Comparison of feature ranking methods based on information entropy, Proc. of International Joint Conference on Neural Networks, 1415–1420
2. Erdogmus D, Principe J C (2004) Lower and Upper Bounds for Misclassification Probability Based on Renyi's Information, Journal of VLSI Signal Processing Systems, 37:305–317

3. Fano R M (1961) Transmission of Information: A Statistical Theory of Communications. Wiley, New York
4. Hellman M E, Raviv J, (1970) Probability of Error, Equivocation and the Chernoff Bound, IEEE Transactions on Information Theory, 16:368–372
5. Koller D, Sahami M (1996) Toward Optimal Feature Selection, Proceedings of the International Conference on Machine Learning, 284–292
6. Battiti R (1994) Using Mutual Information for Selecting Features in Supervised Neural Net Learning, Neural Networks, 5:537–550
7. Bonnlander B V, Weigend A S (1994) Selecting Input Variables Using Mutual Information and Nonparametric Density Estimation, Proceedings of International Symposium on Artificial Neural Networks, 42–50
8. Yang H, Moody J (2000) Data Visualization and Feature Selection: New Algorithms for Nongaussian Data, Advances in Neural Information Processing Systems, 687–693
9. Oja E (1983) Subspace Methods of Pattern Recognition, Wiley, New York
10. Devijver P A, Kittler J (1982), Pattern Recognition: A Statistical Approach, Prentice Hall, London
11. Fukunaga K, (1990) Introduction to Statistical Pattern Recognition, Academic Press, New York
12. Everson R, Roberts S (2003) Independent Component Analysis: A Flexible Nonlinearity and Decorrelating Manifold Approach, Neural Computation,11:1957–1983
13. Hyvrinen A, Oja E, Hoyer P, Hurri J (1998) Image Feature Extraction by Sparse coding and Independent Component Analysis, Proceedings of ICPR, 1268–1273
14. Torkkola K, (2003) Feature Extraction by Non-Parametric Mutual Information Maximization, Journal of Machine Learning Research, 3:1415–1438
15. Battiti R, (1994) Using Mutual Information for Selecting Features in Supervised Neural Net Training, IEEE Transaction Neural Networks, 5:537–550
16. Kira K, Rendell L. (1992) The feature selection problem: Traditional methods and a new algorithm, Proceedings of Conference on Artificial Intelligence, 129–134
17. John G H, Kohavi R, Pfleger K, (1994) Irrelevant features and the subset selection problem, Proceedings of Conference on Machine Learning, 121–129
18. Hild II K E, Erdogmus D, Torkkola K, Principe J C (2006) Feature Extraction Using Information-Theoretic Learning, IEEE Transactions on Pattern Analysis and Machine Intelligence, 28:1385–1392
19. Guyon I, Elisseeff A (2000) An Introduction to Variable and Feature Selection, Journal of Machine Learning Research (Special Issue on Variable and Feature Selection)
20. Scholkopf B, Smola A, Muller K R (1998) Nonlinear Component Analysis as a Kernel Eigenvalue Problem, Neural Computation, 10:1299–1319
21. Hyvarinen A, Karhunen J, Oja E (2001) Independent Component Analysis, Wiley
22. Lee D D, Seung H S (1999) Learning the parts of objects by non-negative matrix factorization, Nature 401, 788–791
23. Roweis S, Saul L, (2000) Nonlinear dimensionality reduction by locally linear embedding, Science,290:2323–2326
24. Costa J, Hero A O (2005) Classification constrained dimensionality reduction, Proceedings of ICASSP, 5:1077–1080

25. Baudat G, Anouar F (2000) Generalized Discriminant Analysis Using a Kernel Approach, Neural Computation, 12:2385–2404
26. Principe J C, Fisher J W, Xu D, (2000) Information Theoretic Learning, Unsupervised Adaptive Filtering, S. Haykin Editor, Wiley, New York, 265–319
27. Parzen E (1967) On Estimation of a Probability Density Function and Mode, Time Series Analysis Papers, Holden-Day, Inc., San Diego, California
28. Erdogmus D, (2002) Information Theoretic Learning: Renyi's Entropy and its Applications to Adaptive System Training, PhD Dissertation, University of Florida, Gainesville, Florida
29. Kraskov A, Stoegbauer H, Grassberger P (2004) Estimating Mutual Information, Physical Review E, 69:066138
30. Learned-Miller E G, Fisher J W (2003) ICA Using Spacings Estimates of Entropy, Journal of Machine Learning Research,4:1271–1295
31. Vasicek O, (1976) A Test for Normality Based on Sample Entropy, Journal of the Royal Statistical Society B,38:54–59
32. Hero A O, Ma B, Michel O J J, Gorman J (2002) Applications of Entropic Spanning Graphs, IEEE Signal Processing Magazine, 19:85–95
33. Beirlant J, Dudewicz E J, Gyorfi L, Van Der Meulen E C (1997) Nonparametric Entropy Estimation: An Overview, International Journal of Mathematical and Statistical Sciences, 6:17-39
34. Erdogmus D, Principe J C (2002) An Error-Entropy Minimization Algorithm for Supervised Training of Nonlinear Adaptive Systems, IEEE Transactions on Signal Processing, 50:1780–1786
35. Erdogmus D, Principe J C (2006) From Linear Adaptive Filtering to Nonlinear Information Processing, to appear in IEEE Signal Processing Magazine
36. Erdogmus D, Hild II K E, Rao Y N, Principe J C (2004) Minimax Mutual Information Approach for Independent Components Analysis, Neural Computation, 16:1235–1252
37. Girolami M, Orthogonal Series Density Estimation and the Kernel Eigenvalue Problem, Neural Computation, MIT Press, 14:669–688
38. Duda R O, Hart P E, Stork D G (2000) Pattern Classification, 2nd ed., Wiley
39. Devroye L, Lugosi G (2001) Combinatorial Methods in Density Estimation, Springer, New York
40. Renyi A (1970) Probability Theory, North-Holland, Amsterdam
41. Silverman B W, (1986) Density Estimation for Statistics and Data Analysis, Chapman and Hall, London
42. Duin R P W (1976) On the Choice of the Smoothing Parameters for Parzen Estimators of Probability Density Functions, IEEE Transactions on Computers, 25:1175–1179
43. Schraudolph N (2004) Gradient-Based Manipulation of Nonparametric Entropy Estimates, IEEE Transactions on Neural Networks, 15:828–837
44. Mercer J (1909) Functions of Positive and Negative Type, and Their Connection with the Theory of Integral Equations, Transactions of the London Philosophical Society A, 209:415–446
45. Wahba G (1990) Spline Models for Observational Data, SIAM, Philedelphia, Pennsylvania
46. Weinert H (ed.) (1982) Reproducing Kernel Hilbert Spaces: Applications in Statistical Signal Processing, Hutchinson Ross Pub. Co., Stroudsburg, Pennsylvania

47. Fowlkes C, Belongie S, Chung F, Malik J (2004) Spectral Grouping Using the Nystrom Method, IEEE Transactions on Pattern Analysis and Machine Intelligence, 23:298–305
48. Ozertem U, Erdogmus D, Jenssen R (2006) Spectral Feature Projections That Maximize Shannon Mutual Information with Class Labels, Pattern Recognition, 39:1241–1252
49. Gorman R. P., Sejnowski T. J. (1988) Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets, Neural Networks 1:75–79
50. http://www.ics.uci.edu/ mlearn/MLRepository.html