Learning from Examples with Information Theoretic Criteria

Jose C. Principe, Dongxin Xu, Qun Zhao, John W. Fisher III

Computational NeuroEngineering Laboratory,

University of Florida, Gainesville, FL 32611

principe@cnel.ufl.edu

Abstract

This paper discusses a framework for learning based on information theoretic criteria. A novel algorithm based on Renyi's quadratic entropy is used to train, directly from a data set, linear or nonlinear mappers for entropy maximization or minimization. We provide an intriguing analogy between the computation and an information potential measuring the interactions among the data samples. We also propose two approximations to the Kulback-Leibler divergence based on quadratic distances (Cauchy-Schwartz inequality and Euclidean distance). These distances can still be computed using the information potential. We test the newly proposed distances in blind source separation (unsupervised learning) and in feature extraction for classification (supervised learning). In blind source separation our algorithm is capable of separating instantaneously mixed sources, and for classification the performance of our classifier is comparable to the support vector machines (SVMs).

1 Introduction

Learning theory develops models from data in an inductive framework. It is therefore no surprise that one of the critical issues of learning is generalization. But before generalizing the machine must learn from the data. How an agent learns from the real world is far from being totally understood. Our most developed framework to study learning is perhaps statistical learning theory [32], where the goal of the learning machine is to approximate the (unknown) a posteriori probability of the targets given a set of exemplars (Figure 1). But there are many learning scenarios that do not fit this model (such as learning without a teacher). Instead we can think that the agent is exposed to sources of information from the external world, and explores and exploits redundancies from one or more sources. This alternate view of learning shifts the problem to the quantification of redundancy and ways to manipulate it. Since redundancy is intrinsically related to the mathematical concept of information, information theory becomes the natural framework to study machine learning. Barlow [2] was one of the pioneers to bring the mathematical concept of information to biologically plausible information processing. His work motivated others to reduce redundancy in learning [11], and it is one of the basis of the work on sparse representations in vision [23]. Linsker proposed the maximization of mutual information between the input to the output of a systems as a principle for self-organization [21].

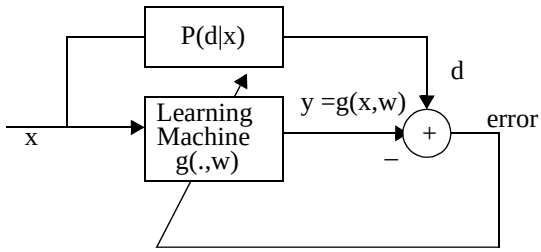

Figure 1: Machine learning according to statistical learning theory. The parameters w are adapted to minimize a measure of the discrepancy betwwen y and d.

Information theory proposed by Claude Shannon [30] has served a crucial role in communication theory [4], but its application to pattern recognition and learning theory has been less pivotal [5]. At the core lies the difficulty that pattern recognition is a discipline based on the learning by example metaphor, while information theory principles require an analytic form for the probability density function (pdf). One possibility is to postulate the form of the pdfs (e.g. a Gaussian distribution) and estimate from the data their parameters (mean and variance for Gaussian). This has been exactly the way Linsker [21] applied his principle of maximum information preservation (InfoMax). The analytic tractability has also restricted most of the work to linear models [5], [21] [25].

Recently, we have shown that these restrictions are no longer necessary [9], [37]. We developed a nonparametric estimator of entropy for a set of data (based on the Parzen window pdf estimator with appropriate entropy measures) and formulated entropy manipulation as seeking extrema of a cost function. Hence any mapper (linear or nonlinear) can be trained with our scheme. We have shown that the method although computationally demanding ($O(N^2)$, N is the number of data points in the training set) is robust and extracts more information from the input data than the mean square error criterion (which only captures second order information from the data and can be regarded as a specific case of our scheme). We have applied the technique to blind source separation [35] and pose estimation [36] with very good results.

This paper clarifies and extends the algorithm for entropy estimation to the important case of mutual information. The mutual information of two random vectors is a very useful principle for designing information processing systems as InfoMax clearly shows. We will start by briefly reviewing information theoretic learning and its unifying role for learning with or without a teacher. We then proceed by presenting an algorithm that can train arbitrary learning machines to maximize (or minimize) mutual information between its input and output. We will conclude the paper by presenting two applications, one for blind source separation and the other to classification of vehicles in synthetic aperture radar (SAR) imagery.

2 Information Theoretic Learning

We define information theoretic learning (ITL) as the procedure to adapt the free parameters *w* of a learning machine *g(.,w)* using an information theoretic criterion (Figure 2). Information theoretic learning seems the natural way to train the parameters of a learning machine because the ultimate goal of learning is to transfer the information contained in the external data (input and or desired response) onto the parametric adaptive system. We envisage two basic criteria for ITL: entropy (maximization or minimization) and mutual information (maximization or minimization). Both work in the output space of the learning system, but each has its own domain of application: entropy is a function of one variable and it is dependent upon the specific coordinate system utilized to represent the data. Hence, entropy manipulation is intrinsically an unsupervised learning paradigm. Entropy maximization is formally an extension of maximizing output energy in linear adaptive systems with the MSE criterion (which leads to the well known principal component analysis), and has been used for blind source separation [3]. Entropy minimization has been utilized for redundancy reduction [2], prediction [10], and can be potentially used in clustering.

Mutual information relies on the estimation of a divergence measure [19] between probability density functions of two random variables and is independent of the coordinate system. Potentially it is the information measure more useful for engineering applications because it involves sets of random variables. Depending on the nature of these variables mutual information criteria can fall either under supervised or unsupervised learning as we will see below. Mutual information has been utilized in independent component analysis [23], blind source separation [1], and we show applications to feature extraction [36], classification [37], and suggest its general role to extend adaptive linear filtering towards information filtering.

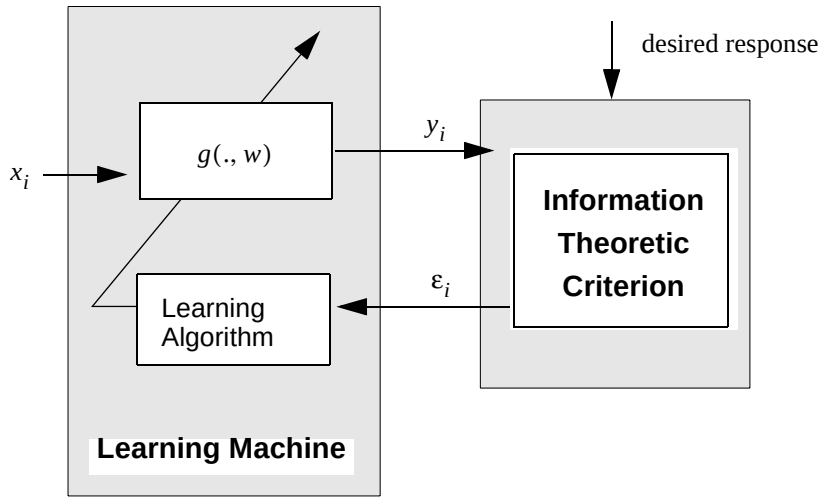

Figure 2: Training a learning machine (linear or nonlinear) with ITL

2.1 Entropy Criterion and its Applications

Let us define the amount of information associated with the measurement of a discrete event x_k which occurs with probability p_k as $I(p_k) = \log\frac{1}{p_k}$, which is Hartley's amount of information [19]. Shannon's entropy H_S is the expectation of Hartley's measure, i.e.

$$H_S(x) = \sum_{k=1}^{n} p_k I(p_k) \qquad \sum_{k=1}^{n} p_k = 1 \qquad p_k \geq 0 \tag{1}$$

Entropy has been extended to continuous random variables $x \in C$ as the differential entropy [4]

$$H_S(x) = \int_C p(x)\log\frac{1}{p(x)}dx \tag{2}$$

The entropy criterion leads to the maximum entropy principle (MaxEnt) enunciated by Jaynes [18], one of the most powerful optimization principles with large applications in statistical mechanics, physical sciences, economics and engineering [19]. The goal of MaxEnt is to maximize uncertainty about what is unknown about the problem constraints. Jaynes shows that most of the distributions used in statistical physics can be derived with MaxEnt.

In signal processing, entropy can also be utilized to extend many of the established methodologies based on second order moments, e.g. variance and correlation functions. Recall that the moments of an i.i.d. random variable $E\{x^n\}$ completely describe the distribution. When the random variable is Gaussian distributed, only the first moment (the mean) and the second moment (the variance) are different from zero. Since the Gaussian assumption is pervasive in engineering models, this explains why many important figures of merit are based on mean and variance. For instance, the well-known concept of signal-to-noise ratio (SNR) evolved from the need to quantify the deterministic versus the stochastic part of real world signals. SNR can be defined as the ratio between the mean and the variance of signal-plus-noise, since normally the signal is deterministic (the mean) and the noise is a wideband (white) zero-mean random variable. If the noise is Gaussian, SNR characterizes adequately the relation between the energy in the mean and in the higher order moments of the measured signal. However, if the noise is not Gaussian, the variance should be replaced by the entropy in the SNR definition.

Output variance maximization is a well established (biological plausible) principle of self-organization described by Hebb [6]. It also gives rise to important signal processing operations known as maximum eigenfiltering (or matched filtering) obtained by maximizing the Rayleigh quotient [16]

$$J(w) = \frac{w^T S w}{w^T w} \tag{3}$$

where S is the autocorrelation of the input and w is the system's weight vector. If the input noise is white, Eq. 3 is really a SNR since the noise autocorrelation function is an identity matrix and $w^T I w = w^T w$. We have shown that maximizing the entropy at the output of a nonlinear system yields substantially more information about the data distribution than eigenfiltering [27]. An alternative, albeit less well-known, criterion for SNR maximization at the output of an adaptive system with input x and noise n is [37]

$$J_H = H(w^T x) - H(w^T n) \tag{4}$$

Eq. 4 is a much broader definition of SNR, because now instead of working with the second order statistics of the signal and noise we use their entropies. This definition is embedded in Linsker's work on maximum information preservation (InfoMax) [21]. Under certain mild conditions, maximizing the transfer of information between the input and output of a system defaults to maximizing the output entropy. Maximization of the output entropy was utilized by Bell and Sejnowski in their well-known method of blind separation of sources [3]. An extension of Eq. 3 to multiple outputs gives rise to the well known principal component analysis (PCA) [6]. Substituting covariances by entropies may lead to a principled way of computing principal curves.

More generally, we can extend all these concepts to stochastic time series. Variances give rise to time autocorrelation functions which are the second order moments of the random process. Time autocorrelation functions play a central role in adaptive filtering theory [16], eigendecompositions (Karhunen-Loeve transforms) and neural network learning [15], so we expect that an entropy based criterion will impact all these applications. As stated by Plumbey [25], the challenge is to develop computational algorithms to extend entropy manipulation to the general case of nonlinear systems and non-Gaussian signals.

2.2 Mutual Information Criterion and its Applications

Mutual information manipulation is more useful than entropy for learning because it involves the estimation of a distance between pdfs. Many information theoretic distance measures between two pdfs have been proposed and studied in the literature [20], but the most widely known is the Kullback-Leibler (K-L) divergence. The distance between two functions *f(x)* and *g(x)* of the same random variable *x* can be defined as the K-L divergence between the two pdfs [4], i.e.

$$D(f||g) = \int_C f(x) \log \frac{f(x)}{g(x)} dx \tag{5}$$

The K-L divergence can be regarded as an "asymmetric distance" between the pdfs. One can show that it is always nonnegative and zero only if f(x) = g(x) [4]. The Kulback-Leibler divergence is at the center of the other well known information theoretic criterion, which was enunciated by Kullback and is called principle of minimum crossentropy (MinxEnt). The goal of MinxEnt is to find a probability distribution that is as close as possible to another distribution. For the special case that f(x) is the joint probability of two random variables X_1 and X_2 $f(x) = f_{X_1 X_2}(x_1, x_2)$ and g(x) is the product of the corresponding marginal variables $g(x) = f_{X_1}(x_1) f_{X_2}(x_2)$, the Kulback-Leibler divergence becomes the mutual information between X_1 and X_2, that is,

$$I(X_1, X_2) = \iint f_{X_1 X_2}(x_1, x_2) \frac{f_{X_1 X_2}(x_1, x_2)}{f_{X_1}(x_1) f_{X_2}(x_2)} dx_1 dx_2 \tag{6}$$

Mutual information can also be thought as a distance between the joint density and the product of the marginals since it is always greater or equal to zero (mutual information is even symmetric) [4]. The minimum is obtained when the variables are independent.

Mutual information gives rise to either unsupervised or supervised learning rules depending upon how the problem is formulated. Figure 3 shows a block diagram of a unifying scheme for learning based on the same ITL criterion of mutual information. The only difference is the signal source which is shown as a switch with 3 positions.

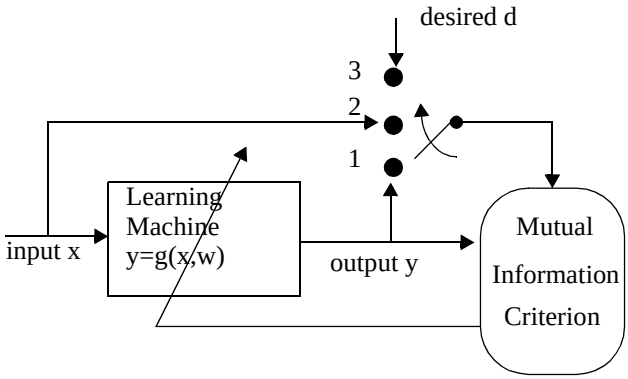

Figure 3: Unifying learning models with the mutual information criterion.

When the switch is in position 1 or 2, learning belongs to the unsupervised type and corresponds to manipulating the mutual information at the output of the learning system or between its input and output. A practical example with switch in position 1 is the on-going work on independent component analysis (ICA) or blind source separation (BSS) where the goal is to minimize the mutual information among the output of a mapper to yield independent components [15], [1]. An example of the block diagram with switch in position 2 is Linsker's InfoMax criterion [21] where the goal is to transfer as much information between the input and output of a mapper by maximizing the joint input-output mutual information. Note that this is a direct implementation of InfoMax, unlike the approach that maximizes output entropy discussed above.

However, if the goal is to maximize the mutual information between the output of a mapper and an external desired response, then learning becomes supervised. This is achieved by setting the switch to position 3, and now the desired response appears as one of the marginal pdfs in the mutual information criterion. The system is solving a feature extraction problem because it finds an input projection relevant to approximate, in an information theoretic sense, the external desired response. The two important cases belong both to function approximation: first, if the desired response is a set of indicator functions, the task is feature extraction for classification. We are performing feature selection based on information theoretic criterion. Note also that here the desired data is always quantified by means of its pdf, not by deriving a sample by sample error. Therefore we can think of this case as supervised learning without numeric targets, as we illustrate later in section 6.2 [37]. Second, if the desired response data is a continuous function we named the application information filtering [26]. This name came from the realization that the learning machine is seeking a projection of the input space that best approximates in an information sense the desired response. Information filtering extends Wiener filtering [16] where the adaptive system is restricted to be a linear filter and the criterion is minimization of the error variance (second order moment).

2.3 How Appropriate is the Mutual Information Criterion for Learning?

Due to the novelty of this approach, we do not have yet many arguments to theoretically justify the use of mutual information criterion for learning theory. The solid foundation for the use of information theory stems from communication theory [30], [4], [8], and from statistical mechanics [18]. But in learning theory two of the fundamental problems are inference and statistical estimation [32]. For instance in parameter estimation, we know today that the variance of any unbiased estimator is bounded from below by the Cramer-Rao bound [5]. Similarly, it is important to ask how appropriate it is to use mutual information based criteria for minimizing the Bayes error in classification.

Assume that the goal is to estimate a variable *x* with a discrete pdf *p(x)* by calculating an estimate $\hat{x}$ from another random variable *y* characterized by *p(x|y)*. Under mild conditions, Fano showed that [8]

$$P(x \neq \hat{x}) \geq \frac{H_S(x|y) - 1}{\log(\Theta(x))}$$

where Θ(x) means the number of possible instances of *x*. This equation shows that the probability of error is lower bounded by the conditional entropy of x given y. Substituting the definition of mutual information $I(x, y)$ we obtain

$$P(x \neq \hat{x}) \geq \frac{H_S(x) - I(x, y) - 1}{\log(\Theta(x))} \tag{7}$$

Notice that we have no control over the entropy of *x* nor the number of possible instances of *x*. *Therefore to improve the lower bound on the achievable probability of error, we should maximize the mutual information between x and y.* Since the goal is to minimize the probability of error, we may think that Eq. 7 is not very useful result because it does not provide an upper bound. But exactly like the Cramer-Rao bound, Eq. 7 talks about the achievable lower bound, while the upper bound depends upon the particular estimator we choose. A theoretical upper bound for the Bayes error given as a function of the conditional entropy is given in [17].

With all these nice properties of information measures, the reader may be wondering why information theory has not been widely applied in machine learning. The answer lies in the difficulty of estimating entropy and mutual information directly from data. Next we will provide an estimator for entropy based on an alternative definition of entropy proposed by the Hungarian mathematician Alfred Renyi [28].

3 Renyi's entropy

Shannon's entropy was defined in Eq. 1 as the expectation of Hartley's amount of information, but there are alternate definitions of entropy. In the general theory of means, the mean of the real numbers $x_1, \ldots, x_n$ with weights $p_1, \ldots, p_n$ has the form:

$$\bar{x} = \varphi^{-1}\left(\sum_{k=1}^{n} p_k \varphi(x_k)\right) \tag{8}$$

where $\varphi(x)$ is a Kolmogorov-Nagumo function, which is an arbitrary continuous and strictly monotonic function defined on the real numbers. In general, an entropy measure obeys the relation:

$$H = \varphi^{-1}\left(\sum_{k=1}^{n} p_k \varphi(I(p_k))\right) \tag{9}$$

As an information measure, φ( ) can not be arbitrary since information is "additive". To meet the additivity condition, φ( ) can be either $\varphi(x) = x$ or $\varphi(x) = 2^{(1-\alpha)x}$. If the former is used, Eq. 9 will become Shannon's entropy. If $\varphi(x) = 2^{(1-\alpha)x}$, Eq. 9 becomes Renyi's entropy with order α [28] which we will denote by $H_{R\alpha}$

$$H_{R\alpha} = \frac{1}{1-\alpha}\log\left(\sum_{k=1}^{n} p_k^{\alpha}\right) \qquad \alpha > 0, \alpha \neq 1 \tag{10}$$

When $\alpha = 2$, Eq. 10 becomes $H_{R_2} = -\log \sum_{k=1}^{n} p_k^2$ and it will be called Quadratic Entropy. According to Figure 3, we are interested in manipulating entropy and mutual information at the output of a system, hence we will be using *y* as our random variable to denote exactly this fact.

For the continuous random variable Y with pdf $f_Y(y)$, we can obtain the differential version for these two types of entropy following a similar route to the Shannon differential entropy [30]:

$$\begin{cases} H_{R\alpha}(Y) = \dfrac{1}{1-\alpha}\log\left(\displaystyle\int_{-\infty}^{+\infty} f_Y(y)^{\alpha} dy\right) \\ H_{R_2}(Y) = -\log\left(\displaystyle\int_{-\infty}^{+\infty} f_Y(y)^{2} dy\right) \end{cases} \tag{11}$$

From the point of view of estimation, Renyi's entropy is very appealing since it involves the integral of a power of the pdf, which is simpler to estimate than Shannon's entropy. Renyi's entropy also brings a different view to the problem of entropy estimation. Let us consider the probability distribution $P= (p_1,p_2,...,p_N)$ as a point in a N-dimensional space. Due to the conditions on the probability measure ($p_k \geq 0$, $\sum_{k=1}^{N} p_k = 1$) P always lies in the first quadrant of an hyperplane in N dimensions intersecting each coordinate axis at the point 1 (Fig. 4). The distance of P to the origin is

$$\|P\|_{\alpha} = \sqrt[\alpha]{\sum_{k=1}^{N} p_k^{\alpha}} = \sqrt[\alpha]{V_{\alpha}} \tag{12}$$

and is called the α-norm of the probability distribution. Renyi's entropy (Eq. 10) can be written as a function of V_α

$$H_{R\alpha} = \frac{1}{1-\alpha}\log V_{\alpha} \tag{13}$$

When different values of α are selected in the Renyi's family, the end result is to select different α-norms. Shannon entropy can be considered as the limiting case $\alpha \to 1$ of the probability distribution norm. Other values of α will measure the distance to the origin in different ways, very much like the selection of the norm of the error in the learning criterion [15]. We settled on $\alpha = 2$ because in the nonlinear dynamics literature Renyi's entropy has also been used to estimate attractor's dimension from experimental data with very good results [13]. In general, higher α increases the robustness of the estimation in areas with low sample density, but the algorithmic complexity increases exponentially with α, so $\alpha = 2$ is a good compromise.

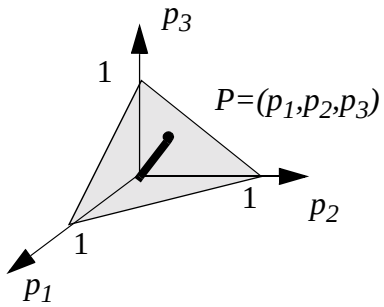

Figure 4: Geometric interpretation of entropy for N=3. The distance of P to the origin is related to the α-norm.

It is important to discuss the implications of this development. Shannon's entropy definition has been intrinsically related to the estimation of the pdf of the random variable. Attempts of using it have either assumed an analytical model for the pdf [4], [19] or have used nonparametric pdf estimators [34], which perform poorly in large dimension-

ality spaces. Renyi's definition alternatively shows that entropy is related to the norm of the pdf in probability spaces. The norm of a vector is a much easier quantity to estimate in high dimensional spaces, in particular if the order of the norm is low (such as the 2-norm).

4 Quadratic entropy and its nonparametric estimator

We will be working with Renyi's quadratic entropy because there is a straight forward way of estimating the 2-norm of the pdf using the well known Parzen window estimator [24]. Let $y_i \in R^k$, $i = 1, \ldots, N$, be a set of samples from a random variable $Y \in R^k$ in k-dimensional space which can be the output of a nonlinear mapper such as a multilayer perceptron (MLP). How can we estimate the 2-norm of this set of data samples? One answer lies in the estimation of the data pdf by the Parzen window method using a Gaussian kernel:

$$f_Y(y) = \frac{1}{N} \sum_{i=1}^{N} G(y - y_i, \sigma^2) \tag{14}$$

where $G(y, \Sigma)$ is the Gaussian kernel in k dimensional space, and Σ is the covariance matrix (for simplicity spherically symmetric kernels $\Sigma = \sigma^2 I$ are utilized here). We just need to substitute Eq. 14 in Eq. 11 to yield immediately:

$$\begin{aligned} H(\{y_i\}) &= -\log\left(\int_{-\infty}^{+\infty} f_Y(y)^2 dy \right) = -\log V(\{y_i\}) \\ V(\{y_i\}) &= \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} G(y_i - y_j, 2\sigma^2) \end{aligned} \tag{15}$$

Note how simple this result is. The integral computation was replaced by the evaluation of the kernel at the location y_i-y_j. Instead of $H(\{y_i\})$ we will be manipulating $V(\{y_i\})$ since it is simpler and the logarithm does not affect the optimization. Making the analogy between data samples and "physical particles", $V(\{y_i\})$ can be regarded as an overall potential energy of the data set since $G(y_i - y_j, 2\sigma^2)$ can be taken as the potential energy of data sample y_i in the potential field of data sample y_j, or vice versa. We will call this potential energy an *information potential*, where the data samples have a correspondence to physical particles and the information potential to a potential field. Therefore, maximizing Renyi's quadratic entropy is equivalent to minimizing information potential. Our estimator for quadratic Renyi's entropy (Eq. 15) only suffers from the approximation inherent to the pdf estimation.

Just like in mechanics, the derivative of the potential energy is a force, in this case an *information force*. The information force moves the data samples in the output space to achieve an equilibrium state dictated by our criterion. Therefore,

$$\frac{\partial}{\partial y_i} G(y_i - y_j, 2\sigma^2) = G(y_i - y_j, 2\sigma^2)(y_j - y_i)/(2\sigma^2) \tag{16}$$

can be regarded as the force that the data sample y_j impinges upon y_i. If all the data samples are free to move in a certain region of the space, then the information forces between each pair of data will drive all the data samples to a state with minimum information potential.

Suppose the data samples are the outputs of a parametric adaptive system, for example an MLP. If we want to adapt the MLP with an iterative algorithm such that the system maximizes the output entropy $H(\{y(n)\})$, the problem is equivalent to finding the parameters of the MLP so that the information potential $V(\{y(n)\})$ is minimized. There-

fore, the information forces applied to each data sample can be back-propagated to the parameters using the chain rule [29]. As an example, the following gradient can be interpreted as force back-propagation:

$$\frac{\partial}{\partial w_{ij}} V(\{y(n)\}) = \sum_{n=1}^{N} \sum_{p=1}^{k} \frac{\partial}{\partial y_p(n)} V(\{y(n)\}) \frac{\partial}{\partial w_{ij}} y_p(n) \tag{17}$$

where $y(n) = (y_1(n), \ldots, y_k(n))^T$, and w_{ij} is one of the weights in MLP. The quantity

$$\frac{\partial}{\partial y(n)} V(\{y(n)\}) = \left(\frac{\partial}{\partial y_1(n)} V(\{y(n)\}), \ldots, \frac{\partial}{\partial y_k(n)} V(\{y(n)\}) \right)^T \tag{18}$$

is the information force that the data sample $y(n)$ is subject to. Notice that the sensitivity of the output with respect to a MLP parameter $\frac{\partial}{\partial w_{ij}} y_p(n)$ is the transmission mechanism through which information forces are back-propagated to the parameter. From the analogy with the backpropagation formalism we conclude that information forces take the place of the injected error. *Hence, we obtain a general, nonparametric, and sample-based methodology to adapt arbitrary nonlinear systems (with smooth nonlinearities) for entropy manipulation.*

The user has to select only two parameters in the training algorithm: the learning rate and the kernel size. We suggest an adaptive stepsize algorithm for faster convergence [9]. The kernel size requires more attention. First, we normalize to one the slope of Eq. 16 at $y_i = y_j$ to provide a force that is independent of the kernel size. From the understanding of the information potential, it is straight forward to conclude that the samples have to interact with each other. Therefore for entropy minimization we set the kernel size such that the two furthest samples still interact. Since the samples change position during learning, this distance should be updated during training (but infrequently to avoid adding another dynamics to the learning process). For entropy maximization the goal is to produce evenly distributed samples in the output space. Hence the kernel size can be estimated as $D\sqrt{\frac{y_{max} - y_{min}}{N}}$, where D is the dimension of the output space and N the number of samples. We also suggest to slowly anneal the kernel size, as done in Kohonen training. We verified experimentally that the kernel size needs to be in the correct range, but does not need to be finely tuned. In [27] we present a more principled approach to set the kernel size based on cross-validation.

5 Quadratic Mutual Information and Cross-Information Potential

For two random variables Y_1 and Y_2 (with marginal pdfs $f_{Y_1}(y_1)$, $f_{Y_2}(y_2)$ and joint pdf $f_{Y_1 Y_2}(y_1, y_2)$), mutual information can be estimated using the Kullback-Leibler divergence between the joint probability and the factored marginals [5]. But the problem is that the K-L divergence is very difficult to estimate nonparametrically in high dimensional spaces. From the literature on information theoretic measures we saw that there are other distances proposed that provide very similar results to the K-L divergence. In learning this situation is even more favorable due to the fact that we are maximizing or minimizing mutual information (or entropy), therefore as long as our criterion has extrema that coincides with the K-L divergence extrema the results will be indistinguishable. This is the great advantage of a learning framework, which implies that there is considerable freedom in selecting criteria for information theoretic learning. Inspired by this reasoning and constrained by quadratic forms of pdfs, we propose the following two information theoretic distance measures to estimate mutual information: The first is based on the Cauchy-Schwartz inequality:

$$I_{CS}(Y_1, Y_2) = \log \frac{\left(\iint f_{Y_1 Y_2}(y_1, y_2)^2 dy_1 dy_2 \right) \left(\iint f_{Y_1}(y_1)^2 f_{Y_2}(y_2)^2 dy_1 dy_2 \right)}{\left(\iint f_{Y_1 Y_2}(y_1, y_2) f_{Y_1}(y_1) f_{Y_2}(y_2) dy_1 dy_2 \right)^2} \tag{19}$$

which we called Cauchy-Schwartz quadratic mutual information (CS-QMI) [27]. The CS-QMI can be thought as a generalization of the correlation coefficient, which measures the angle between the joint pdf and the product of the

marginals in probability space. It is obvious that $I_{CS}(Y_1, Y_2) \geq 0$ and the equality holds true if and only if Y_1 and Y_2 are statistically independent, i.e. $f_{Y_1Y_2}(y_1, y_2) = f_{Y_1}(y_1)f_{Y_2}(y_2)$. So, $I_{CS}(Y_1, Y_2)$ is appropriate to measure the independence of two variables (minimization of mutual information). We also have experimental evidence that $I_{CS}(Y_1, Y_2)$ is an appropriate measure for dependence of two variables (maximization of mutual information) and has an upper bound due to the normalization. In [37] we proposed a second alternative definition for quadratic mutual information based on the Euclidean distance between the joint entropy and the product of their marginals, i.e.

$$I_{ED}(Y_1, Y_2) = \iint (f_{Y_1Y_2}(y_1, y_2) - (f_{Y_1}(y_1))f_{Y_2}(y_2))^2 dy_1 dy_2 \tag{20}$$

which was named Euclidean distance quadratic mutual information (ED-QMI). The integrated square error (ISE) between kernel based density estimates of multivariate pdfs has been studied in the statistical literature, and applied to experimentally measure the distance between chaotic attractors with good results (the Diks test) [7]. We were the first to apply the concept of ISE in learning applications [9]. Here we merely extend it to measure the distance between the joint and the product of the marginals. ED-QMI is also a distance and is zero when the variables are statistically independent.

For learning, what is essential is that the minima and maxima of the newly defined CS-QMI and ED-QMI coincide with the extrema of I(Y_1, Y_2). We have derived the relationships between CS-QMI, ED-QMI and mutual information for the case of Gaussian random variables [37], and concluded that they have the same maxima and minima. In [27] we show a case of a simple probability mass function to illustrate that the extrema between CS-QMI, ED-QMI and mutual information also coincide. For more general pdfs we only have experimental evidence that the quadratic mutual information criteria are able to find solutions that produce good results. Here we will present an algorithm to estimate CS-QMI directly from the data (see [27] for a full treatment).

Suppose that we observe a set of data samples $\{y_{i1}, i = 1, \ldots, N\}$ for the variable Y_1, $\{y_{i2}, i = 1, \ldots, N\}$ for the variable Y_2. Let $y_i = (y_{i1}, y_{i2})^T$. Then $\{y_i, i = 1, \ldots, N\}$ are data samples for the joint variable $(Y_1, Y_2)^T$. Based on the Parzen window method, the joint pdf and marginal pdf can be estimated as:

$$\begin{cases} f_{Y_1Y_2}(y_1, y_2) = \frac{1}{N_i}\sum_{i=1}^{N} G(y_1 - y_{i1}, \sigma^2)G(y_2 - y_{i2}, \sigma^2) \\ f_{Y_1}(y_1) = \frac{1}{N_i}\sum_{i=1}^{N} G(y_1 - y_{i1}, \sigma^2) \\ f_{Y_2}(y_2) = \frac{1}{N_i}\sum_{i=1}^{N} G(y_2 - y_{i2}, \sigma^2) \end{cases} \tag{21}$$

Combining (19), (21) and using (15), we obtain the following expressions for the CS-QMI based on a set of data samples:

$$\begin{cases} I_{CS}((Y_1, Y_2)\,|\,\{y_i\}) = \log \dfrac{V(\{y_i\})V_1(\{y_{i1}\})V_2(\{y_{i2}\})}{V_{nc}(\{y_i\})^2} \\ V(\{y_i\}) = \dfrac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} G(y_i - y_j, 2\sigma^2) = \dfrac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\prod_{l=1}^{2} G(y_{il} - y_{jl}, 2\sigma^2)\right) \\ V_l(y_j, \{y_{il}\}) = \dfrac{1}{N}\sum_{i=1}^{N} G(y_{jl} - y_{il}, 2\sigma^2), \quad l = 1, 2 \\ V_l(\{y_{il}\}) = \dfrac{1}{N}\sum_{j=1}^{N} V_l(y_j, \{y_{il}\}), \qquad l = 1, 2 \\ V_{nc}(\{y_i\}) = \dfrac{1}{N}\sum_{j=1}^{N}\left(\prod_{l=1}^{2} V_l(y_j, \{y_{il}\})\right) \end{cases} \tag{22}$$

These expressions can be interpreted in terms of information potentials and extended for the case of multiple variables [37], but we do not have space to elaborate on the interpretation.

The cross-information potential (the argument of the log of I_{CS} in Eq. 22) is more complex than the information potential of Eq. 15. Three different potentials (joint potential *V(.)*, marginal potentials $V_l(.)$, unnormalized cross-potential $V_{nc}(.)$) contribute to the cross-information potential. Hence, the force applied to each data sample y_p comes from three independent sources (the marginal components). The q marginal force (marginal space indexed by q) that the data point y_p receives can be calculated according to the following formulas:

$$\begin{aligned} \frac{\partial}{\partial y_{pq}} V(\{y_i\}) &= \frac{1}{N^2}\sum_{i=1}^{N}\left(\prod_{l=1}^{k} G(y_{iq} - y_{pq}, 2\sigma^2)\right)\frac{y_{iq} - y_{pq}}{\sigma^2} \\ \frac{\partial}{\partial y_{pq}} V_q(\{y_{iq}\}) &= \frac{1}{N^2}\sum_{i=1}^{N} G(y_{iq} - y_{pq}, 2\sigma^2)\frac{y_{iq} - y_{pq}}{\sigma^2} \\ \frac{\partial}{\partial y_{pq}} V_{nc}(\{y_i\}) &= \frac{1}{N^2}\sum_{j=1}^{N}\frac{1}{2}(B_j) G(y_{jq} - y_{pq}, 2\sigma^2)\frac{y_{jq} - y_{pq}}{\sigma^2} \end{aligned} \tag{23}$$

where $B_j = \prod_{l \neq q} V_l(y_j, \{y_{il}\}) + \prod_{l \neq q} V_l(y_p, \{y_{il}\})$. The overall marginal force that the data point y_p receives is:

$$\frac{\partial}{\partial y_{pq}} I_{CS}((Y_1, Y_2)\,|\,\{y_i\}) =$$
$$= \frac{1}{V(\{y_i\})}\frac{\partial}{\partial y_{pq}} V(\{y_i\}) + \frac{1}{V_q(\{y_{iq}\})}\frac{\partial}{\partial y_{pq}} V_q(\{y_{iq}\}) - 2\frac{1}{V_{nc}(\{y_i\})}\frac{\partial}{\partial y_{pq}} V_{nc}(\{y_i\})$$

Notice that the force from different sources are normalized by their corresponding information potentials to balance them out. This is a very nice feature of the CS-QMI. Once the forces that each data point receives are calculated, these forces become the injected error, and can again be back-propagated to all the parameters of the learning machine so that the adaptation takes the system state to the extremum of the criterion (minimum or maximum depending on the sign of the error).

6 Experimental results

In order to demonstrate the use of ITL in realistic problems, we will present here an example of blind source separation and classification. Other tests of this methodology have been reported [35],[31], [36].

6.1 Blind Source Separation

Blind source separation can be formulated in the following way. The observed data $\boldsymbol{X} = \boldsymbol{AS}$ is a linear mixture ($\boldsymbol{A} \in R^{m \times m}$ is non-singular) of independent source signals $\boldsymbol{S} = (\boldsymbol{S}_1, \ldots, \boldsymbol{S}_m)^T$. There is no further information about the sources and the mixing matrix, hence the denomination "Blind". The problem is to find a projection $\boldsymbol{W} \in R^{m \times m}$, so that $\boldsymbol{Y} = \boldsymbol{WX}$ will become $\boldsymbol{Y} = \boldsymbol{S}$ up to a permutation and scaling.

We present below the results of a linear de-mixing system trained with the Cauchy-Schwartz quadratic mutual information (CS-QMI) criterion. From this point of view, the problem can be re-stated as finding a projection $\boldsymbol{W} \in R^{m \times m}$, $\boldsymbol{Y} = \boldsymbol{WX}$ so that the CS-QMI among all the components of $\boldsymbol{Y}$ is minimized, that is all the output signals are independent of each other. This methodology is intrinsically nonparametric, unlike the mainstream work in BSS [1], [15], so it is independent of the special of the kurtosis. For ease of illustration, only 2-source-2-sensor problem is tested.

There are two experiments presented: Experiment 1 tests the performance of the method on a very sparse data set which was instantaneously mixed in the computer with a mixing matrix [2, 0.5; 1, 0.6]. Two, 2-D, different colored Gaussian noise segments are used as sources, with 30 data points for each segment (sparse data case). The two segments were concatenated and shuffled. Fig. 5 (left panel) shows the source density in the joint space (each axis is one source signal). As Fig. 5 shows, the mixing produces a mixture with both long and short "tails" which is difficult to separate (middle panel). Whitening is first performed on the mixtures to facilitate de-mixing. The data distributions for the recovered signals are plotted in Fig. 5 (right panel). As we can observe the original source density is obtained with high fidelity. Fig. 5 also contains the evolution of the SNR of de-mixing-mixing product matrix ($\boldsymbol{WA}$) during training as a function of batch iterations. The adaptation approaches a final SNR of 36.73 dB in less than 700 batch iterations.

Experiment 2 uses two speech signals from the TIMIT database as source signals (Fig. 6). The mixing matrix is [1, 3.5; 0.8, 2.6] where the two mixing direction [1, 3.5] and [0.8, 2.6] are similar. An on-line implementation is tried in this experiment, in which a short-time window (200 samples) slides over the speech data (e.g. 10 samples/step). In each window position, the speech data within the window is used to calculate the information potentials, information forces and back-propagated forces all using batch learning to adjust the de-mixing matrix. As the window slides at 10 samples/step the demixing matrix keeps being updated. The training curve (SNR vs. sliding index) is shown in Fig. 6 which tells us that the method converges within 40,000 samples of speech and achieves a SNR approaching 49.15 dB, which is comparable to other methods for this mixing condition. The large spikes in the training curve shows the occasional almost perfect demixing matrix estimation while the algorithm is still adapting (notice that during adaptation the algorithm can estimate one of the directions very well although it is still far away from the optimal solution).

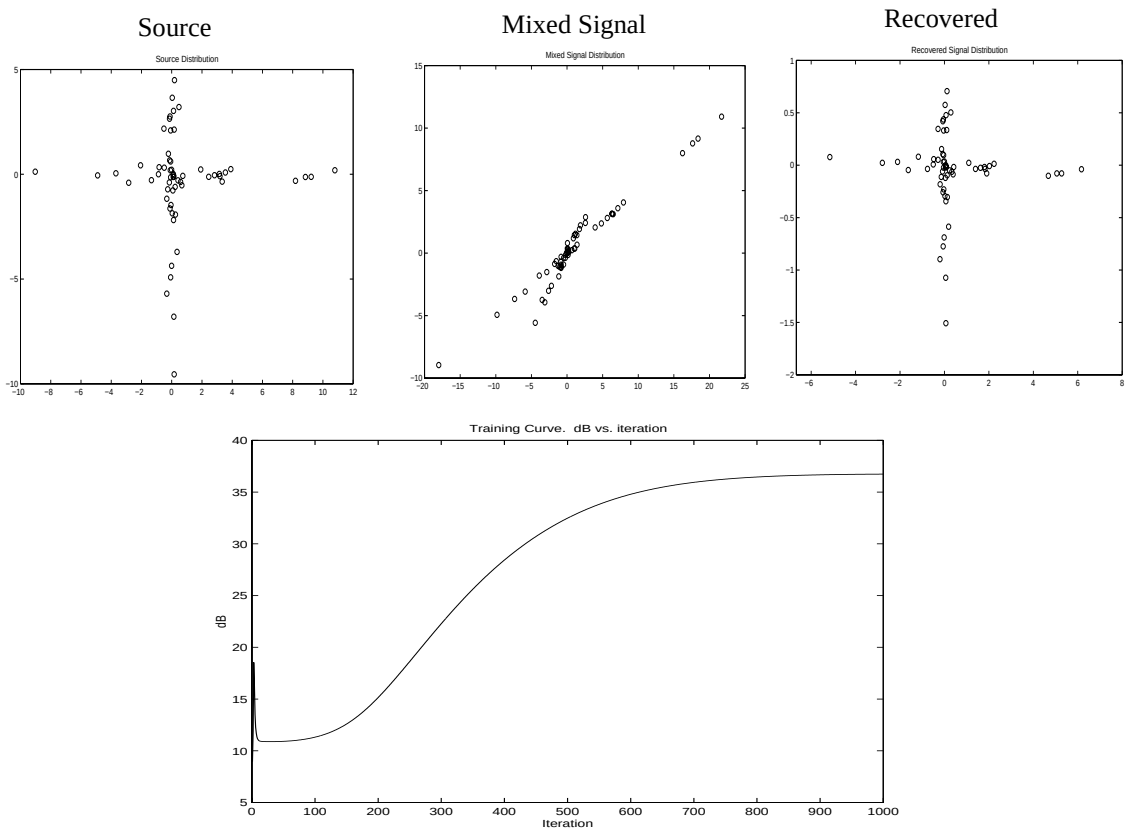

Figure 5: Data distributions for the sources (left), mixed (middle) and demixed with the proposed method (right). Learning curve on bottom plotting the product WA in dB as a function of batch iterations. Notice the final 36 dB of SNR.

In order to obtain a stable result the learning rate is linearly reduced through training. Although whitening is done before CS-QMI learning, we believe that the whitening process can also be incorporated into the ITL algorithm.

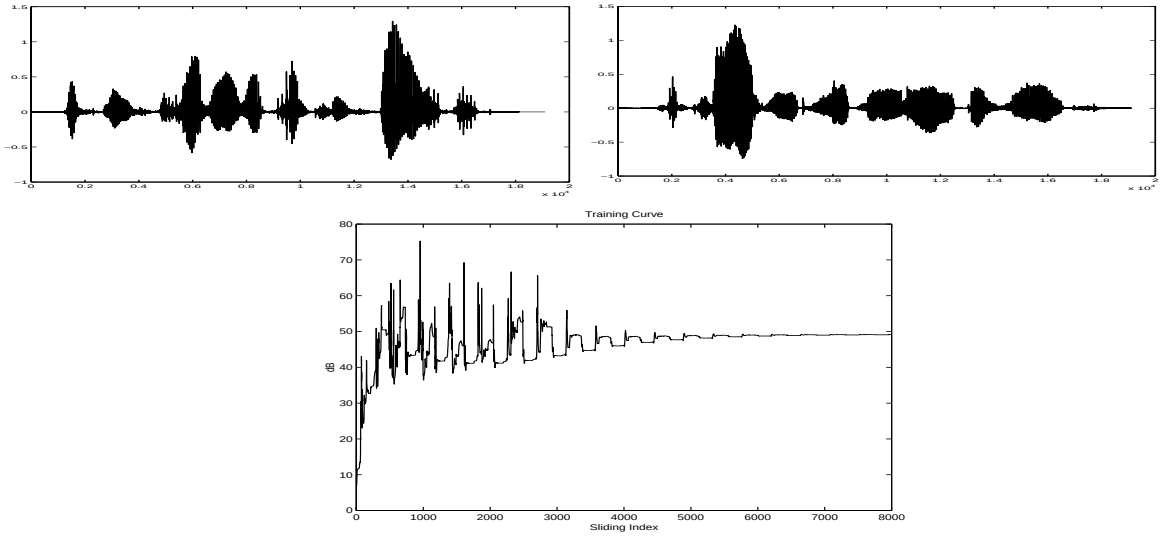

Figure 6: Two speech signals from TIMIT that were mixed, and resulting training curve plotting WA in dB versus the sliding window index. Final SNR is around 50 dB.

6.2 Feature Extraction for Classification

This example is part of an on-going effort in our laboratory to develop classifiers for automatic target recognition using synthetic aperture radar (SAR/ATR) imagery. Synthetic aperture radar (SAR) automatic target recognition (ATR) experiments were performed using the MSTAR database to classify three targets and reject confusers. The data are 80 by 80 SAR images drawn from three types of ground vehicles: the T72, BTR70, and BMP2 as shown in Figure 7. These images are a subset of the 9/95 MSTAR Public Release Data [22]. The poses (aspect angles) of the vehicles lie between 0 to 180 degrees as shown in Figure 7.

A SAR image is the amplitude of the FFT (fast Fourier transform) of the radar return properly mapped from time to space. The images are very noisy due to the image formation and lack of resolution due to the radar wavelength, which makes the classification of SAR vehicles a non-trivial problem [33]. Unlike optical images, the SAR images of the same target taken at different aspect angles are not correlated with each other which precludes the existence of a rotation invariant transform. This results from the fact that a SAR image reflects the fine target structure (point scatter distribution on the target surface) at a certain pose. Parts of the target structure will be occluded when illuminated by the radar, which results in dramatic differences from image to image with angular increments as small as 10 degrees. Thus a classifier should be trained with each pose for better results.

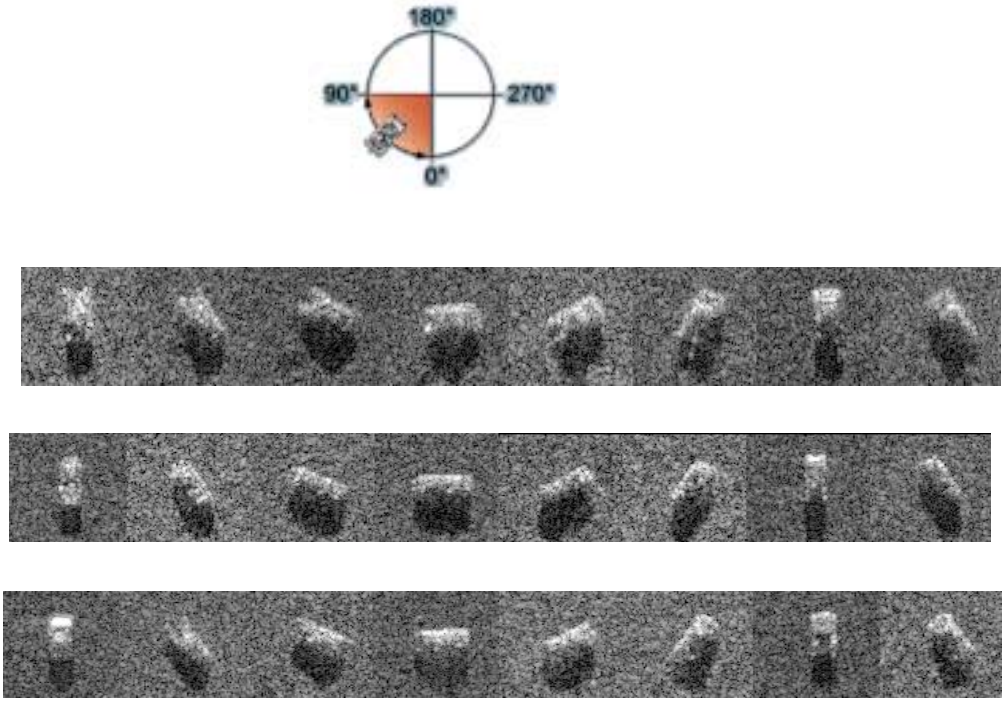

Figure 7: Examples of the SAR training set for 3 vehicles. Notice the difficulty of the task both in terms of the variability and noise in the images.

In these experiments we have created 6 classifiers each covering 30 degrees of aspect such that vehicles appearing at poses between 0-180 degrees can be classified accurately. We have further compared three classifiers: a support vector machine (SVM) using a Gaussian kernel [32], an optimal separation hyperplane (OH) classifier [38] and the classifier based on the mutual information criterion ED-QMI of Eq. 20. We finally compare them with the perceptron trained with the delta rule to gauge the level of performance with more conventional methods.

The training set contained SAR images taken at a depression angle of seventeen degrees, while the testing set depression angle is fifteen degrees. Hence, the SAR images between the training and the testing sets for the same vehicle at the same pose are different, which helps to test the classifier generalization. Variants (different serial number) of the three targets were also used in the testing set. The size of training and testing sets is 406 and 724, respectively.

Two types of experiments were conducted. One is the conventional classification task, and the other is the more challenging recognition task. In the recognition task confuser vehicles, i.e. other vehicles not used in the training where presented to the classifiers and the rejection rate was computed for a detection probability of Pd=0.9.

The SVM and OH classifiers where trained with the Adatron algorithm [12]. The difference between these two classifiers is that the OH does the classification in the input space, while the SVM does the classification in feature space. For this problem nearly all the inputs are support vectors so the classification with the SVM is in fact done in a 400 dimensional space. Since the Adatron algorithm is applied to a single output perceptron, we trained sequentially one class versus the other two. Further details can be found in [38].

The classifier based on the ED-QMI is a perceptron with a 80x80 input layer and 2 outputs (i.e. creating a two dimensional feature space). Due to the large input dimension, a one hidden layer MLP produced virtually the same results, so it will not be further considered. The idea is to find a projection that will preserve the most information jointly contained in the output and the desired response. Therefore, one should maximize our measure of mutual information in the criterion (ED-QMI). The training progresses smoothly and is over in 200 batch iterations. Figure 8 depicts three

snapshots of training in the beginning of training, half way and at the end of training. In the left panels we show the samples and the information forces being exerted on each output sample. In the right panels we zoom in the output space to have a clearer view of the separation between clusters. Notice that in the beginning of training the images of each input are mixed in the output space indicating bad discrimination. Half way through the training we see the clusters separating, and the information forces are large and centrifugal, i.e. separating the clusters. We can also observe a smaller dispersion in each cluster because information forces among samples of different clusters repel while the samples of each class attract. At the end of training the information forces are almost zero and the clusters are well separated, and very compact (almost a point). Clearly this will provide easy discrimination among the classes (at least for the training set). Note that the ED-QMI information force in this particular case can be interpreted as repulsion among the samples with different class labels, and attraction with each other among the samples within the same class.

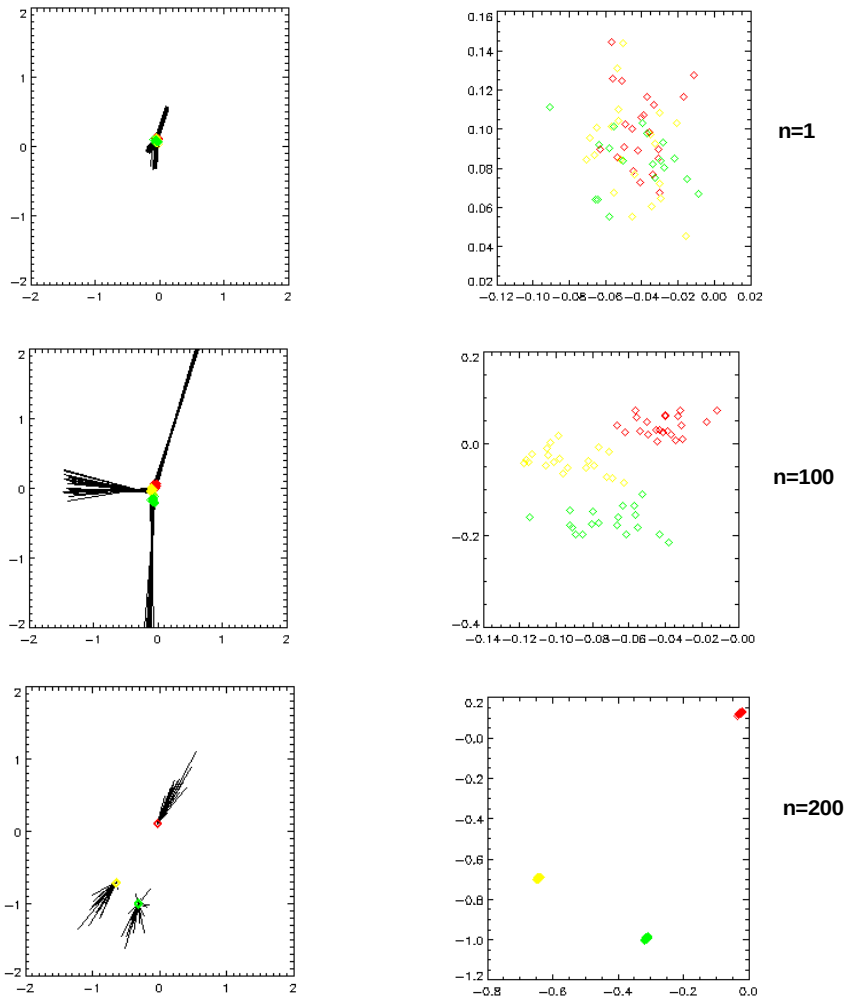

Figure 8: Three snapshots of the 2D output space of the classifier during learning. Left panels show the output clusters (color coded per class) and the information forces, while the right panels zoom in the output space to see each individual output sample.

The input data projected in the feature space (Figure 8) is the natural starting point to design the classifier. The estimation of the joint pdf of the class labels and the mapper output $\hat{f}_{CY}(y, c)$ using the ED-QMI is given by [37]

$$\hat{f}_{CY}(y, c) = \frac{1}{N}\sum_{i=1}^{N} G(y - y_i, 2\sigma^2)\delta(c - c_i) \tag{24}$$

where σ^2 is the variance for Gaussian kernel function for the feature variable y, and $\delta(c - c_i)$ is the Kronecker delta function

$$\delta(c - c_i) = \begin{cases} 1 & c = c_i \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

Notice that the class labels appear only as a selector in the calculations of Eq. 24, so effectively we are not using the numerical value of the targets (as is done in supervised learning). Based on the joint pdf $\hat{f}_{CY}(y, c)$, the Bayes classifier can be built up as

$$\arg\max_{c} \hat{f}_{CY}(y, c) \qquad y = g(x, w) \qquad c = 1, 2, 3 \tag{26}$$

Since the class identity variable C is discrete, the search for the maximum can be simply implemented by comparing each value of $\hat{f}_{CY}(y, c)$ across the classes.

Table I shows the results for classification using the OH, the SVM, the ED-QMI and the perceptron.

Table I. Misclassification error (%)

	BMP2	**BTR70**	**T72**	**Average**
OH	6.45	1.87	5.28	5.25
SVM	7.74	0.93	4.56	5.39
ED-QMI	6.77	0.93	4.23	5.11
Perceptron	9.35	2.80	11.4	8.98

We see that the classifier trained with ED-QMI performs at the same level as the other two classifiers. This is very rewarding since the SVMs are known for their extremely good performance. It is interesting to analyze the principles behind each classifier. The OH is creating discriminant functions in the input space (6,400 dimensions), while the SVM is creating discriminant functions in a space of dimensionality given by the support vectors. This decoupling between the input space and the feature space dimensionality is a distinct feature of the SVMs. In our case this yields a smaller 400 dimensional space. The information theoretic feature extraction using ED-QMI first projects the data to a low space (here 2D) from where a Bayes classifier can be directly designed (Eq. 26). From Table I we see that the ED-QMI result is slightly better (although the differences are probably not significant), which means that the ED-QMI based projection is also well tuned to the structure of the data clusters. To compare these results with "conventional" classifiers we also implemented a perceptron and trained it with MSE, weight decay and early stopping (for details see [15]). As we can observe from Table I the perceptron produces almost twice the misclassification error of any of the other classifiers.

Table II shows the results for the recognition task. Two different vehicles (275 different examples) were added to the test set creating what is called the confuser class [38]. Now the problem becomes much more difficult because we are measuring how well the discriminant function represents the class in an extended operating environment. With the conventional test set data we check the generalization performance in areas of the input space close the classes, but this still leaves out many unexplored regions of the input space where the classifier will provide a class assignment. Confusers test exactly the performance in areas of the input space away from the class centers. Ideally the response of the classifier to other vehicles not present in the training set, and which reside away from the training data, should be close to zero. But the conventional training does not enforce this. The problem becomes a blend of detection and classification, which has been named recognition. We have to establish a detection threshold for the comparison (in fact a receiver operating characteristic would be more appropriate [38]). Here the realistic probability of detection of Pd = 0.9 is chosen. The results are presented in Table II. In this task, a good classifier will produce low misclassification error and reject as many confusers as possible. The SVM outperforms the OH for confuser rejection by a large margin

Table II. Misclassification error (%) and confuser rejection (%) for Pd=0.9

	BMP2	BTR70	T72	Average	Confuser
OH	3.87	0.93	2.28	2.76	48
SVM	3.55	0.93	0.98	2.07	68
ED-QMI	3.95	0.75	0.95	1.88	64
Perceptron	3.87	1.87	6.19	4.56	22

(68% versus 48%). We see that the perceptron trained with ED-QMI has comparable performance to the SVM machine. The average classifier error rate is slightly better than the SVM but the rejection rate to confusers is slightly worse (64 versus 68%). The perceptron classifier has a very poor performance for confuser rejection.

The rejection to confusers is highly dependent upon the type of discriminant function that the network topology creates. We [38] (and others [14]) have shown that the most suitable discriminant function for the task of rejection is a local discriminant function. Global discriminant functions such as hyperplanes produce with high probability large responses in areas of the input space away from the class clusters, while local discriminant functions naturally bound the class. This partially explains the difference between the OH and the SVM since they are trained with the same algorithm, except that one creates linear discriminant functions in the input space (OH) while SVMs create local discriminant in pattern space. The ED-QMI implements a Bayesian classifier in the projected space and it is difficult to say how it is projected back to the input space, but its performance is much closer to the SVM than to the OH classifier. Hence, we conclude that the mutual information training is creating discriminant functions that fit tightly the class cluster, comparable to the best classifiers. As Table II clearly shows, the perceptron trained with the delta rule totally breaks down for the task of recognition (it can only reject 22% of the confuser vehicles). This shows that MSE training places discriminant functions to meet the training set criterion but do not guarantee a good match to the data clusters in the input space.

7 Conclusion

We develop in this paper a nonparametric framework for information theoretic learning. Hence, the learning machine is able to learn directly from the data just like the conventional MSE criterion, but now utilizing information contained in the probability density function instead of only second order statistics about the error. Under this framework, we can manipulate entropy and mutual information at the output of any linear or nonlinear systems. We show that the mutual information criterion can be utilized without any modifications in both supervised and unsupervised learning, unifying one of the most well established taxonomic distinctions in neural network learning. We utilize the concept of Renyi's Quadratic entropy to develop an easily implementable entropy estimator based on the information potential. Although the Parzen estimator is utilized in our algorithm, we note that the important quantity is the integral of the pdf which is much easier to estimate from data. With this estimator of entropy applied to the output space of a parametric system the parameters can be adapted with information force backpropagation. Using the Cauchy-Schwartz and the Euclidean distances instead of Kulback-Leibler divergence we are able to extend the method to estimate dis-

tances between pdfs. We illustrate the performance of the novel algorithm in two applications: blind source separation (an unsupervised problem) and automatic target recognition in SAR (a supervised problem). The performance in BSS is similar to other algorithms [35]. In classification the features obtained by maximizing the mutual information between the output of the nonlinear system and the desired response yielded classifiers that rival SVMs. This is very important because SVMs utilize a very different mechanism to design classifiers (projection to higher dimensional spaces). Similar performance means that information theoretic projections onto a reduced space found equally discriminant features for classification. This shows the potential of the new technique not only for classification but also for information filtering. Present work is addressing details in the training such as the kernel size selection, and the effect on performance of the number of output space dimensions. Generalization is also being investigated and compared with that of the MLP and SVMs. We are also studying the statistical properties of the new estimators for entropy and mutual information. The algorithms developed here are $O(N^2)$ where N is the number of samples in the training set. This seems to be an intrinsic limitation since Renyi's quadratic entropy is computed from the interactions of pairs of data samples. On one hand this criterion uses more information about the input data (a data set with N samples has Nx(N-1)/2 different pairs) but it takes longer to compute.

Acknowledgments: This work was partially supported by a DARPA-Air Force grant F33615-97-1019 and NSF ECS-9900394.

References

[1] Amari S., Chichocki A., Yang H., "A new learning algorithm for blind source separation", In Advances of Information Processing Systems 8, pp 757-763, 1996.

[2] Barlow H., Unsupervised learning, Neural Computation, vol 1, 295-311, 1989.

[3] Bell A. and Sejnowski T." An information-maximization approach to blind separation and blind deconvolution", Neural Computation, 7:1129-1159, 1995.

[4] Cover T. and Thomas J., "Elements of Information Theory", Wiley, 1991.

[5] Deco G. and Obradovic D., "An Information-Theoretic Approach to Neural Computing", New York, Springer, 1996

[6] Diamantaras K., and Kung S., "Principal Component Neural Networks: Theory and Applications, Wiley, 1996.

[7] Diks C., Zwet W., Takens F., DeGoede J., Detecting differences between delay vector distributions", Physical Rev E, vol 53, #3, 2169-2176, 1996.

[8] Fano R., "Transmission of information", MIT Press, 1961.

[9] Fisher J. W. III "Nonlinear Extensions to the Minimum Average Correlation Energy Filter" Ph.D dissertation, Dept. of ECE, University of Florida, 1997.

[10] Fisher J., Ihler A., Viola P., "Learning informative statistics: a nonparametric approach", Proc. of Neural Information Proc. Systems, vol 12 (in press).

[11] Foldiak P., "Adaptive network for optimal linear feature extraction", IEEE Int. Joint Conf. Neural Net., vol 1, 401-405, 1989.

[12] Friess T., Support vector neural networks: the kernel Adatron with bias and soft margin", Research report, U. of Sheffield, UK, 1998.

[13] Grassberger I., and Proccacia I., "Measuring the strangeness of strange attractors", Physica D, vol 9, 189-208, 1983.

[14] Gori M and Scarselli F., "Are multilayer perceptrons adequate for pattern recognition and verification?, IEEE Trans. Pattern Analysis and Machine. Intell. 20(11):1121-1132, 1998.

[15] Haykin S., “Neural Networks, A Comprehensive Foundation”, Macmillan Publishing Company, 1998.

[16] Haykin S., “Adaptive Filter Theory”, Prentice Hall, 1986.

[17] Hellman M., Raviv J., “Probability of error, equivocation and the Chernoff bound”, IEEE Trans. Inform. Theory, vol IT-16, #4, 368-372, 1970.

[18] Jaynes E., “In formation theory and statistical mechanics”, Physical Review, vol 106, 620-630, 1957.

[19] Kapur, J.N. “Measures of Information and Their Applications”. John Wiley & Sons, 1994.

[20] Lin J., “Divergence measures based on Shannon entropy”, IEEE Trans. Inform. Theory, vol 37, #1, 145-151, 1991.

[21] Linsker R. “An application of the principle of maximum information preservation to linear systems”, in Advances in Neural Information Processing Systems 1, Morgan-Kaufman, pp 485-494, 1988.

[22] MSTAR (public) Targets, CDROM, Veda Inc. Ohio, 1997.

[23] Olshausen B. and Fields D., “Sparse coding with an overcomplete basis set: a strategy employed by V1”, Vision research, vol 37, 3311-3325, 1997.

[24] Parzen, E. “On the estimation of a probability density function and the mode”, Ann. Math. Stat. 33, p1065, 1962.

[25] Plumbley M., Fallside F., “An information theoretic approach to unsupervised networks”, Int. J. Conf. on Neural Nets, vol 2, p 598, Washington, DC, 1989.

[26] Principe J., “From linear adaptive to information filtering”, Key note address, IEEE Workshop Neural Nets for Sig. Proc., Cambridge, England, August 1998.

[27] Principe J., Xu D., Fisher J., “Information theoretic learning”, in Unsupervised Adaptive Filtering, Ed. Haykin, Wiley, 2000 (in press).

[28] Renyi, A. “Some Fundamental Questions of Information Theory”, Selected Papers of Alfred Renyi, Vol.2, Akademic Kiado, Budapest, 1976.

[29] Rumelhart, D.E., Hinton, G.E. and Williams, J.R. “Learning representations by back-propagating errors”, Nature (London), 323, pp533-536, 1986.

[30] Shannon C. and Weaver W., “The mathematical theory of communication”, University of Illinois Press, 1949.

[31] Wu H-C, principe J., Novel Quadratic Entropy measures and their application to blind source separation/extraction, accepted in IEEE Workshop Neural Networks Sig. Proc. 1999

[32] Vapnik V., “Statistical Learning theory”, Wiley, 1998.

[33] Velten V., Ross T. Mossing J., Worrell S., Bryant M., “standard SAR/ATR evaluation experiments using the MSTAR public release data set”, Research Report, Wright State U., 1998.

[34] Viola P., Schraudolph N., Sejnowski T., “Empirical entropy manipulation for real-world problems”, Proc. Neural Info. Proc. Sys. (NIPS 8) Conf., 851-857, 1995.

[35] Xu D., Principe J., Fisher J. and Wu H-C. “A Novel Measure for Independent Component Analysis (ICA)” Proc. ICASSP’98, vol II, 1161-1164, 1998.

[36] Xu D., Fisher J., Principe J., “Mutual Information approach to pose estimation”, Proc. SPIE, vol 3370, Algorithms for synthetic aperture radar imagery V, 218-229, 1998.

[37] Xu D., “Energy, Entropy and Information Potential for Neural Computation”, Ph.D. dissertation, U. of Florida, 1999.

[38] Zhao Q. and J. Principe, “From hyperplanes to large margin classifiers: Appllications to SAR/ATR”, In Proc. SPIE 13th Annual Int. Sym. Aerospace/Defense Sensing, Simulation and Control, Vol 3718, 1999.