Entropy Manipulation of Arbitrary Nonlinear Mappings

John W. Fisher III **José C. Principe**

Computational NeuroEngineering Laboratory
EB, #33, PO Box 116130
University of Floridaa
Gainesville, FL 32611-6130
jwf@ecl.ufl.edu
principe@cnel.ufl.edu

Abstract

We discuss an unsupervised learning method which is driven by an information theoretic based criterion. Information theoretic based learning has been examined by several authors Linsker [2, 3], Bell and Sejnowski [5], Deco and Obradovic [1], and Viola *et al* [6]. The method we discuss differs from previous work in that it is extensible to a feed-forward multi-layer perceptron with an arbitrary number of layers and makes no assumption about the underlying PDF of the input space. We show a simple unsupervised method by which multi-dimensional signals can be nonlinearly transformed onto a maximum entropy feature space resulting in statistically independent features.

1.0 INTRODUCTION

Our goal is to develop mappings that yield statistically independent features. We present here a nonlinear adaptive method of feature extraction. It is based on concepts from information theory, namely mutual information and maximum cross-entropy. The adaptation is unsupervised in the sense that the mapping is determined without assigning an explicit target output, à priori, to each exemplar. It is driven, instead, by a global property of the output: cross entropy.

There are many mappings by which statistically independent outputs can be obtained. At issue is the usefulness of the derived features. Towards this goal we apply Linsker's *Principle of Information Maximization* which seeks to transfer maximum *information* about the input signal to the output features. It is also shown that the resulting adaptation rule fits naturally into the back-propagation method for training multi-layer perceptrons.

Previous methods [1] have optimized entropy at the output of the mapping by considering the underlying distribution at the input. This represents a complex problem for general nonlinear mappings. The method presented here, by contrast, is more directly related to the technique of Bell and Sejnowski [5] in which we manipulate

0-7803-4256-9/97/$10.00 ©1997 IEEE

entropy through observation at the output of the mapping. Specifically, we exploit a property of entropy coupled with a saturating nonlinearity which results in a method for entropy manipulation that is extensible to feed-forward multi-layer perceptrons (MLP). The technique can be used for an MLP with an arbitrary number of hidden layers. As mutual information is a function of two entropy terms, the method can be applied to the manipulation of mutual information as well.

In section 2 we discuss the concepts upon which our feature extraction method is based. We derive the adaptation method which results in statistically independent features in section 3. An example result is presented in section 4, while our conclusions and observations appear in section 5.

2.0 BACKGROUND

The method we describe here combines cross entropy maximization with Parzen window probability density function estimation. These concepts are reviewed.

2.1 Maximum Entropy as a Self-organizing Principle

Maximum entropy techniques have been applied to a host of problems (e.g. blind separation, parameter estimation, coding theory, etc.). Linsker [2] proposed maximum entropy as a self-organizing principle for neural systems. The basic premise being that any mapping of a signal through a neural network should be accomplished so as to maximize the amount of information preserved. Linsker demonstrates this *principle of maximum information preservation* for several problems including a deterministic signal corrupted by gaussian noise. Mathematically Linsker's principle is stated

$$I(x, y) = h_Y(y) - h_{Y|X}(y|x) \tag{1}$$

where $I(x, y)$ is the mutual information of the RVs X and Y, and $h([\ ])$ is the continuous entropy measure [4]. Given the RV (random vector), $Y \in \Re^N$, the continuous entropy is defined as

$$h_Y(u) = -\int_{-\infty}^{\infty} \log(f_Y(u)) f_Y(u) du, \tag{2}$$

where $f_Y(u)$ is the probability density function of the RV, the base of the logarithm is arbitrary, and the integral is N-fold. Several properties of the continuous entropy measure are of interest.

1. If the RV is restricted to a finite range in $\Re^N$ the continuous entropy measure is maximized for the *uniform* distribution.
2. If the covariance matrix is held constant the measure is maximized for the *normal* distribution.

3. If the RV is transformed by a mapping $g:\Re^N \rightarrow \Re^N$ then the entropy of the new RV, $y = g(x)$, satisfies the inequality

$$h_Y(y) \leq h_X(x) + E\{\ln(|J_{XY}|)\}, \tag{3}$$

with equality if and only if the mapping has a unique inverse, where J_{XY} is the Jacobian of the mapping from X to Y.

Regarding the first two properties we note that for either case each element of the RV is *statistically independent* from the other elements.

Examination of (3) implies that by transforming a RV we can increase the amount of information. This is a consequence of working with continuous RVs. In general the continuous entropy measure is used to compare the relative entropies of several RVs. We can see from (3), that if two RVs are mapped by the same invertible *linear* transformation their relative entropies (as measured by the difference) remains unchanged. However, if the mapping is nonlinear, in which case the second term of (3), is a function of the random variable, it is possible to change relative information of two random variables. From the perspective of classification this is an important point. If the mapping is *topological* (in which case it has a unique inverse), there is no increase, theoretically, in the ability to separate classes. That is, we can always reflect a discriminant function in the transformed space as a warping of another discriminant function in the original space. However, finding the discriminant function is a different problem altogether. By changing the relative information, the form of the discriminant function may be simpler.

This is not true, however, for a mapping onto a subspace. Our implicit assumption here is that we are unable to reliably determine a discriminant function in the full input space. As a consequence we seek a subspace mapping that is in some measure optimal for classification. We cannot avoid the loss of information (and hence some ability to discriminate classes) when using a subspace mapping. However, if the criterion used for adapting the mapping, is entropy based, we can perhaps minimize this loss. It should be mentioned that in *all* classification problems there is an implicit assumption that the classes to be discriminated do indeed lie in a subspace.

2.2 Nonparametric Pdf Estimation

One difficulty in applying the continuous entropy measure with continuous RVs is that it requires some knowledge of the underlying PDF (probability distribution function). Unless assumptions are made about the form of the density function it is very difficult to use the measure directly. A nonparametric kernel-based method for estimating the PDF is the Parzen window method [7]. The Parzen window estimate of the probability distribution, $\hat{f}_Y(u)$, of a random vector $Y \in \Re^N$ at a point u is defined as

$$\hat{f}_Y(u) = \left(\frac{1}{N_y}\right) \sum_{i=1}^{N_y} \kappa(y_i - u). \tag{4}$$

The vectors $y_i \in \Re^N$ are observations of the random vector and $\kappa([\ ])$ is a kernel function which itself satisfies the properties of PDFs (i.e. $\kappa(u) > 0$ and $\int \kappa(u) du = 1$). Since we wish to make a local estimate of the PDF, the kernel function should also be localized (i.e. uni-modal, decaying to zero). In the method we describe we will also require that $\kappa([\ ])$ be differentiable everywhere. In the multidimensional case the form of the kernel is typically gaussian or uniform. As a result of the differentiability requirement, the gaussian kernel is most suitable here. The computational complexity of the estimator increases with dimension, however, we will be estimating the PDF in the output space of our multi-layer perceptron where the dimensionality can be controlled.

3.0 DERIVATION OF LEARNING ALGORITHM

As we stated our goal is to find statistically independent features; features that jointly posses minimum mutual information or maximum cross entropy.

Suppose we have a mapping $g : \Re^N \rightarrow \Re^M$; $M < N$, of a random vector $X \in \Re^N$, which is described by the following equation

$$Y = g(\alpha, X) \tag{5}$$

How do we adapt the parameters α such that the mapping results in a maximum cross-entropy random variable? If we have a desired target distribution then we can use the Parzen windows estimate to minimize the "distance" between the observed distribution and the desired distribution. If the mapping has a restricted range (as does the output of an MLP using sigmoidal nonlinearities), the uniform distribution (which has maximum entropy for restricted range) can be used as the target distribution. If we adapt the parameters, α, of our mapping such that the output distribution is uniform, then we will have achieved statistically independent features regardless of the underlying input distribution.

Viola *et al* [6] has taken a very similar approach to entropy manipulation, although that work differs in that it does not address nonlinear mappings directly, the gradient method is estimated stochastically, and entropy is worked with explicitly. By our choice of topology (MLP) and distance metric we are able to work with entropy indirectly and fit the approach naturally into a back-propagation learning paradigm.

As our minimization criterion we use integrated squared error between our estimate and the desired distribution, which we approximate with a summation.

$$\begin{aligned} J &= \frac{1}{2}\int_{\Omega_Y} (f_Y(u) - \hat{f}_Y(u, y))^2 du \\ &\approx \sum_j \frac{1}{2}(f_Y(u_j) - \hat{f}_Y(u_j, y))^2 \Delta u \qquad y = \{y_1 \ldots y_{N_Y}\} \end{aligned} \tag{6}$$

In (6), Ω_Y indicates the nonzero region (a hypercube for the uniform distribution) over which the M-fold integration is evaluated. The criterion above exploits the fact that the MLP with saturating nonlinearities has finite support at the output. This fact coupled with property 1 (i.e. as the integrated squared error between the observed output distribution and the uniform distribution is minimized, entropy is maximized) makes the criterion suitable for entropy manipulation.

Assuming the output distribution is sampled adequately, we can approximate this integral with a summation in which $u_j \in \Re^M$ are samples in M-space and Δu is represents a volume.

The gradient of the criterion function with respect to the mapping parameters is determined via the chain rule as

$$\begin{aligned}\frac{\partial J}{\partial \alpha} &= \left(\frac{\partial J}{\partial \hat{f}}\right)\left(\frac{\partial \hat{f}}{\partial g}\right)\left(\frac{\partial g}{\partial \alpha}\right) \\ &= \left(\frac{\Delta u}{N_u}\right)\sum_j (f_Y(u_j) - \hat{f}_Y(u_j, y))\left(\frac{\partial \hat{f}}{\partial g}\right)\left(\frac{\partial g}{\partial \alpha}\right), \\ &= \left(\frac{\Delta u}{N_u}\right)\sum_j \varepsilon_Y(u_j, y)\left(\frac{\partial \hat{f}}{\partial g}\right)\left(\frac{\partial g}{\partial \alpha}\right)\end{aligned} \tag{7}$$

where $\varepsilon_Y(u_j, y)$ is the computed distribution error over all observations y. The last term in (7), $\partial g / \partial \alpha$, is recognized as the sensitivity of our mapping to the parameters α. Since our mapping is a feed-forward MLP (α represents the weights and bias terms of the neural network), this term can be computed using standard backpropagation. The remaining partial derivative, $\partial \hat{f} \ / \partial g$, is

$$\begin{aligned}\frac{\partial \hat{f}}{\partial g} &= \left(\frac{1}{N_Y}\right)\sum_{i=1}^{N_Y} \kappa'(y_i - u_j) \\ &= \left(\frac{1}{N_Y}\right)\sum_{i=1}^{N_Y} \kappa'(g(\alpha, x_i) - u_j)\end{aligned} \tag{8}$$

Substituting (8) into (7) yields

$$\frac{\partial J}{\partial \alpha} = \left(\frac{\Delta u}{N_u N_Y}\right)\sum_j \sum_i \varepsilon_Y(u_j, y_i)\kappa'(g(\alpha, x_i) - u_j)\left(\frac{\partial}{\partial \alpha} g(\alpha, x_i)\right) \tag{9}$$

The terms in (9), excluding the mapping sensitivities, become the new error term in our backpropagation algorithm. This adaptation scheme is depicted in figure 1, which shows that this adaptation scheme fits neatly into the backpropagation paradigm.

Examination of the gaussian kernel and its differential in two dimension illustrates some of the practical issues of implementing this method of feature extraction as well as providing an intuitive understanding of what is happening during the adaptation process. The N-dimensional gaussian kernel evaluated at some u is (simplified for two dimensions)

$$\begin{aligned} \kappa(y_i - u) &= \frac{1}{(2\pi)^{N/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(y_i - u)^{\dagger}\Sigma^{-1}(y_i - u)\right) \\ &= \frac{1}{2\pi\sigma^2}\exp\left(-\frac{1}{2\sigma^2}y_i - u^{\dagger}y_i - u\right) \quad ; \ \Sigma = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}, N = 2 \end{aligned} \tag{10}$$

The partial derivative of the kernel (also simplified for the two-dimensional case) with respect to the input y_i as observed at the output of the MLP is

$$\begin{aligned} \frac{\partial\kappa}{\partial y_i} &= -\left(\frac{\exp\left(-\frac{1}{2}(y_i - u)^{\dagger}\Sigma^{-1}(y_i - u)\right)}{(2\pi)^{N/2}|\Sigma|^{1/2}}\right)\Sigma^{-1}(y_i - u) \\ &= \kappa(y_i - u)\Sigma^{-1}(u - y_i) \\ &= \left(\frac{\exp\left(-\frac{1}{2\sigma^2}(y_i - u)^{\dagger}(y_i - u)\right)}{2\pi\sigma^4}\right)(u - y_i) \quad ; \ \Sigma = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}, N = 2 \end{aligned} \tag{11}$$

These functions are shown in figure 2. The contour of the gaussian kernel is useful in that it shows that output samples, y_i, greater than two standard deviations from the center, u, of the kernel (in the feature space) do not significantly impact the estimate of the output PDF at that sample point. Likewise, the gradient term, is not significant for output samples exceeding two standard deviations from the kernel center. Consequently sample points for the PDF estimate should not exceed a distance of two standard deviations from each other, otherwise, samples caught "in between" do not contribute significantly to the estimate of the PDF. A large number of such samples can cause very slow adaptation.

Recall that the terms in (9) replace the standard error term in the backpropagation algorithm. This term is plotted as a surface in figure 2 minus the PDF error. From this plot we see that the kernels act as either local attractors or repellors depending on whether the computed PDF error is negative (repellor) or positive (attractor). In this way the adaptation procedure operates in the feature space locally from a globally derived measure of the output space (PDF estimate).

4.0 EXPERIMENTAL RESULTS

We have conducted experiments using this method on millimeter-wave ISAR (inverse synthetic aperture radar) images (64 x 64 pixels). The mapping structure we use in our experiment is a multi-layer perceptron with a single hidden layer (4096 input nodes,4 hidden nodes, 2 output nodes). Using the adaptation method described, we trained the network on two vehicle types with ISAR images from 180 degrees of aspect. The projection of the training images (and between aspect testing images) is shown in figure 3 (where adjacent aspect training images are connected). As can be some significant class separation is exhibited (without prior labeling of the classes). We also note that the points where the classes overlap correspond to the cardinal aspect angles, which are, in general, difficult aspect angles to separate on similar vehicles in this type of imagery.

5.0 CONCLUSIONS

We have presented what we believe to be a new method of unsupervised learning. This method unlike previous methods is not limited to linear topologies [3] nor unimodal PDFs [5]. In effect, we achieve features which are statistically independent from each other and yet are still, clearly, structurally related to the input structure as exhibited by the results of our example. This property bears similarity to Kohonen's discrete SOFM, however our map exists in a continuous output space. We are pursuing in our research more rigorous analysis in the comparison of the resulting feature maps to the Kohonen type. We are utilizing this method as a preprocessing for classification in our continuing research, although other applications certainly exist (e.g. blind separation).

Acknowledgments

This work was partially supported by DARPA grant F33615-97-1-1019.

REFERENCES

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

[2] R. Linsker, 1988, "Self-organization in a perceptual system.", *Computer,* vol. 21, pp. 105-117.

[3] R. Linsker, 1990, "How to generate ordered maps by maximizing the mutual information between input and output signals.", *Neural Computation,* 1, 402-411.

[4] A. Papoulis, 1991, *Probability, Random Variables, and Stochastic Processes*, 3rd Ed, pp. 533-602, McGraw-Hill.

[5] T. Bell and J. Sejnowski, 1995, "An Information-Maximization Approach to Blind Separation and Blind Deconvolution", *Neural Computation* 7, 1129-1159.

[6] P. Viola, N. Schraudolph, and J. Sejnowski, 1996, "Empirical Entropy Manipulation for Real-World Problems", *Neural Information Processing Systems* 8, MIT Press, 1996.

[7] E. Parzen, 1962, "On the estimation of a probability density function and the mode.", *Ann. Math. Stat.,* 33, pp. 1065-1076.

[8] T. Kohonen, 1988, *Self-Organization and Associative Memory* (1st ed.), Springer Series in Information Sciences, vol. 8, Springer-Verlag.

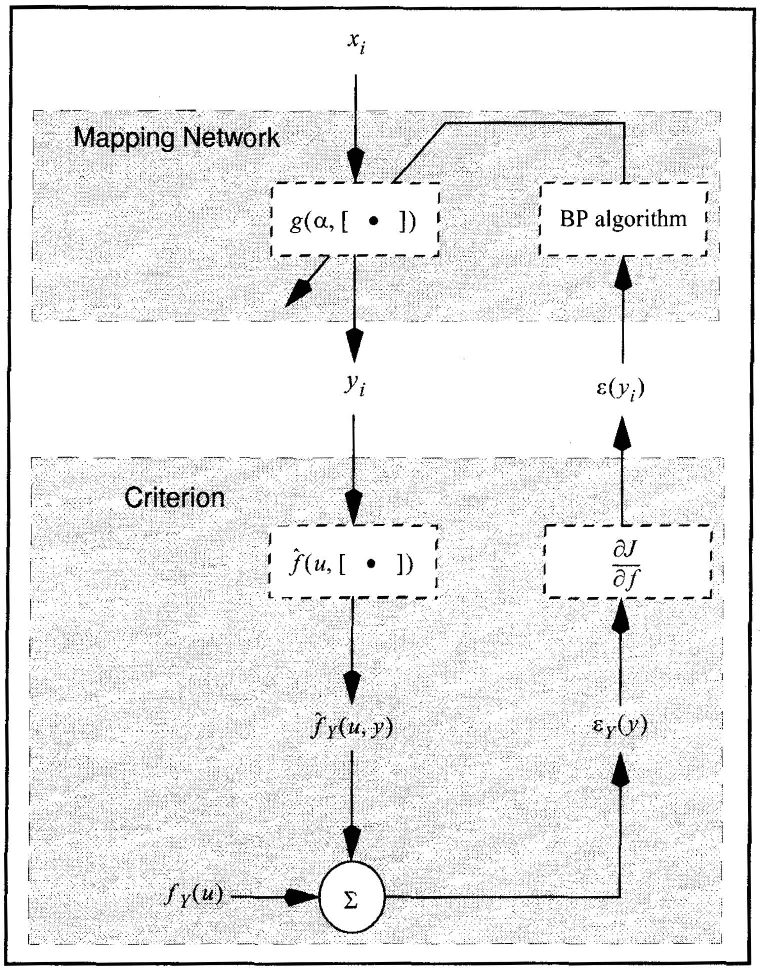

Figure 1 Block diagram of PDF driven adaptation scheme

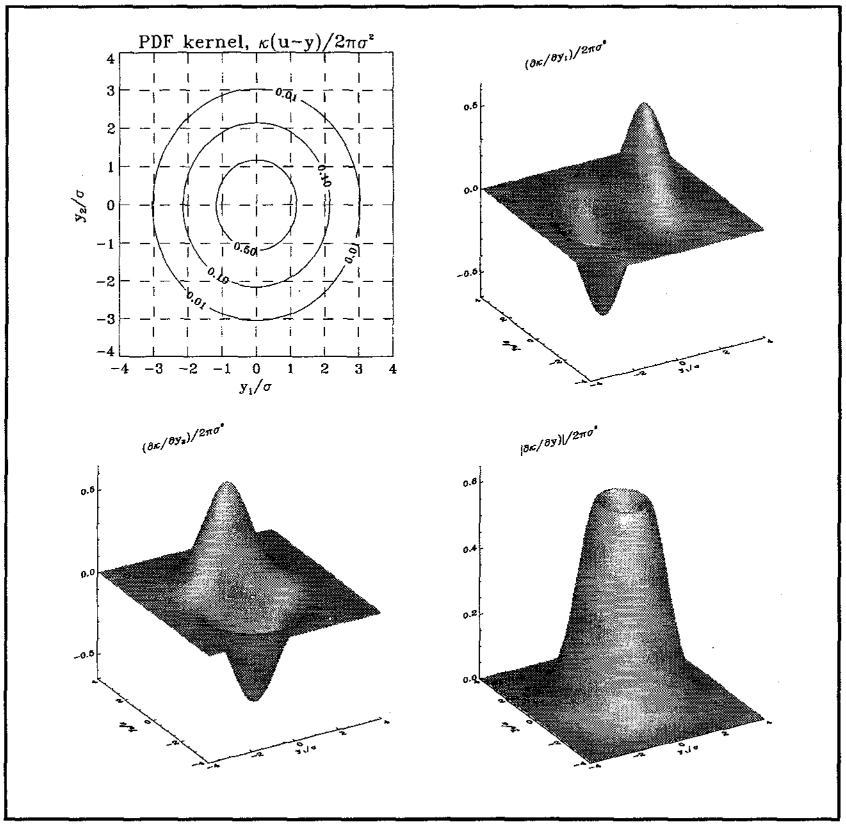

Figure 2 The plots above assume that we are using a two-dimensional gaussian kernel with a diagonal covariance matrix with σ^2 on the diagonals. Contour of the gaussian kernel (top left, normalized by σ), surface plots of the gradient terms with respect to y_1 (top right), y_2 (bottom left), and magnitude (bottom right) all normalized by σ^3. These terms are essentially zero at a distance of two standard deviations.

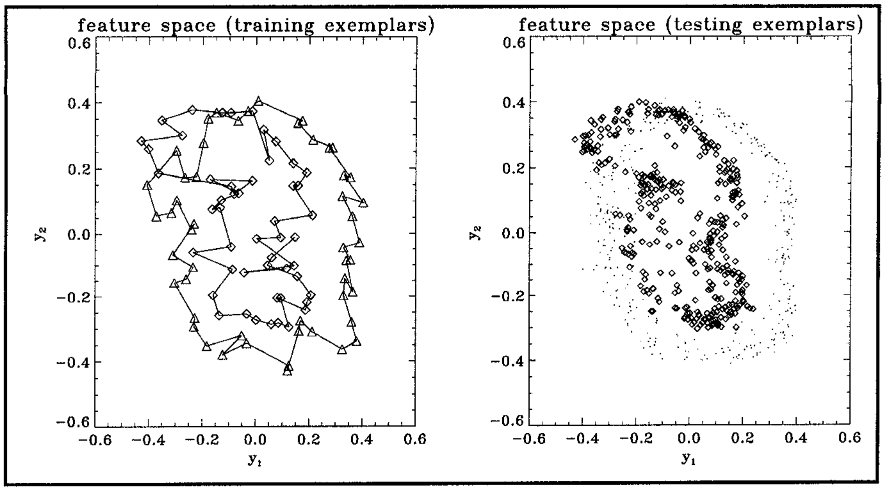

Figure 3 Example of training on ISAR images of two vehicles (aspect varying over 180 degrees). Over most of the aspect angles the vehicles are separated in the new feature space. Adjacent aspect angles are connected in the training set, evidence that topological neighborhoods were maintained. The mapping also generalizes to the testing set as well.