COMPARISON OF ENTROPY AND MEAN SQUARE ERROR CRITERIA IN ADAPTIVE SYSTEM TRAINING USING HIGHER ORDER STATISTICS

Deniz Erdogmus and Jose C. Principe

Computational Neuro-Engineering Laboratory
Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611
E-mail: deniz@grove.ufl.edu and principe@cnel.ufl.edu

ABSTRACT

The error-entropy-minimization approach in adaptive system training is addressed in this paper. The effect of Parzen windowing on the location of the global minimum of entropy has been investigated. An analytical proof that shows the global minimum of the entropy is a local minimum, possibly the global minimum, of the non-parametrically estimated entropy using Parzen windowing with Gaussian kernels. The performances of error-entropy-minimization and the mean-square-error-minimization criteria are compared in short-term prediction of a chaotic time series. Statistical behavior of the estimation errors and the higher order central moments of the time series data and its predictions are utilized as the comparison criteria.

1. INTRODUCTION

Starting with the early work of Wiener [1] on adaptive filters, the mean square error (MSE) has been almost exclusively employed in the training of all adaptive systems including artificial neural networks. There are mainly two reasons lying behind this choice: Analytical simplicity, and the assumption that most real-life random phenomena may be expressed accurately by the Gaussian distribution. The probability density function (pdf) of the Gaussian is characterized by only its first and second order statistics. Hence, under these linearity and Gaussianity assumptions, MSE, which concentrates on second order statistics, would be able to extract all possible information from a data set whose statistics are solely defined by its mean and variance.

However, most real-life problems are governed by nonlinear equations and most random phenomena are far from being normally distributed. Therefore, for the training of adaptive systems, a criterion that not only considers the second order statistics but that also takes into account the higher-order statistical behavior of the systems is required.

The entropy of a given probability distribution function, introduced by Shannon [2], is a scalar quantity that provides a measure of the average information contained in the distribution. By definition, information is a function of the pdf itself hence the entropy is related to the pdf rather than any particular statistics of it.

Application of the entropy criterion to the system identification problem is conceptually quite straightforward. Given a time series produced by an unknown system to be used as the training data, the entropy of the estimation error over the training data set must be minimized [3]. The interpretation of this is as follows. When entropy of the error is minimized, the expected information contained in the estimation error is minimized; hence the adaptive system is trained optimally in the sense that the mutual information between the time series and the model output is maximized.

In practice, an analytical expression for the pdf of a random variable, which is necessary for the computation of the entropy, is not available in most cases. Therefore, it has to be estimated non-parametrically from the samples of the random variable. One way to approximate the pdf of a given sample distribution is to utilize Parzen windowing [3]. In Parzen windowing, the pdf is approximated by a sum of even, symmetric kernels whose centers are translated to the sample points. A suitable and commonly used kernel function is the Gaussian. The Gaussian function is preferable because it is continuously differentiable, and therefore the sum of Gaussians is continuously differentiable on the space of real vectors of any dimension.

The organization of the paper is as follows. First, the backpropogation training algorithms for both Shannon's and Renyi's entropy with parameter 2 are given for the

one-dimensional case. Second, an analytical proof showing us that the global minimum of the entropy is still a minimum of the Parzen window estimated entropy when Gaussian kernels are employed. Then, the results of a case study where estimation-error-entropy-minimization criterion is applied to the short-term prediction of a chaotic time series with a time-delay neural network (TDNN) are presented. The performances of MSE trained and entropy trained TDNNs are compared in terms of error power and higher order statistics.

2. BACKPROPOGATION FOR TDNN TRAINING: ENTROPY CRITERION

A typical system identification scheme with a TDNN is shown in Fig. 1. The training criterion is what characterizes the learning procedure and determines the overall performance of the final model.

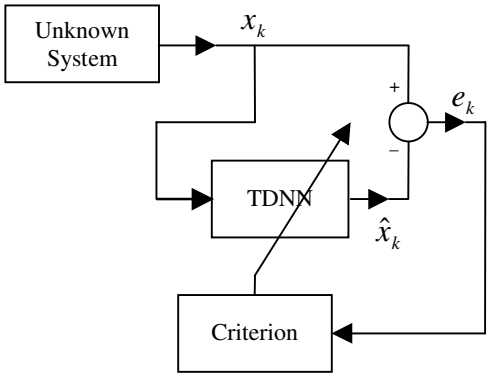

Figure 1: System Identification Scheme

The purpose of this scheme is to find the TDNN weights that optimize the criterion. If the adaptation criterion is chosen to be the minimization of the mean square error, and the optimization procedure is fixed to be steepest descent, then the originating training algorithm is the well-known backpropogation algorithm [4]. However, due to the reasons stated before, if the adaptation criterion is picked to be the minimization of Shannon's entropy of the error with steepest descent, the training algorithm becomes

$$w(n+1) = w(n) - \eta \frac{\partial \hat{H}_S(e)}{\partial w} \tag{1}$$

Here, Shannon's entropy [2] is given by (2), where the error pdf is approximated by Parzen windowing with Gaussian kernels of zero mean and variance σ^2 from its N samples as shown in (3).

$$H_S(e) = -\int_{-\infty}^{\infty} f_e(\xi) \log f_e(\xi) d\xi \tag{2}$$

$$\hat{f}_e(\xi) = \frac{1}{N} \sum_{i=1}^{N} \kappa(\xi - e_i, \sigma^2) \tag{3}$$

The gradient to be used in the training algorithm is therefore

$$\frac{\partial \hat{H}_S(e)}{\partial w} = \frac{1}{N\sigma^2} \sum_{i=1}^{N} \frac{\partial \hat{x}_i}{\partial w} \cdot \int_{-\infty}^{\infty} (\xi - e_i) \kappa(\xi - e_i) \log \hat{f}_e(\xi) d\xi \tag{4}$$

Here, $\partial \hat{x}_i / \partial w$ can be computed as in standard backpropogation. Note that this algorithm requires the numerical evaluation of a complicated integral over the real line. Therefore, this algorithm is extremely slow and computationally inefficient. This problem can be solved by employing Renyi's entropy with $\alpha = 2$ [6]. Renyi's entropy with parameter α is given by [7]

$$H_{R\alpha}(e) = \frac{1}{1-\alpha} \log \int f_e^{\alpha}(\xi) d\xi \tag{5}$$

Note that, for $\alpha = 2$, this expression becomes as in (6) and minimization of this reduces the maximization of the information potential [6]

$$H_{R2}(e) = -\log \int_{-\infty}^{\infty} f_e^2(\xi) d\xi \tag{6}$$

$$V(e) = \int_{-\infty}^{\infty} f_e^2(\xi) d\xi \tag{7}$$

In practice, the pdf of error is again estimated using Parzen windowing as shown in (3), and the gradient vector to be used in the steepest ascent algorithm for the maximization of the information potential is found as

$$\frac{\partial \hat{V}(e)}{\partial w} = \frac{1}{2N^2\sigma^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (e_j - e_i) \cdot \kappa(e_i - e_j, 2\sigma^2) \left[\frac{\partial \hat{x}_j}{\partial w} - \frac{\partial \hat{x}_i}{\partial w} \right] \tag{8}$$

Notice that the sum of integrals in the Shannon's entropy case is replaced by a double summation in this algorithm with a doubling in the standard deviation of the kernel to be used. The property that made this possible is that the integral of two Gaussians with equal standard deviations is a Gaussian with twice the standard deviation. This algorithm is much faster compared to the previous one due to this elimination of the integral. This derivation may be easily extended to the multi-variable case.

One important point to be mentioned in training with entropy is that, the entropy does not change with the mean of the error over the data set. This can be easily shown by a simple change of variables of the integration, i.e. $\zeta = \xi - \mu_e$. Due to this property of the entropy, the algorithm will converge, with a probability of one, to a set of optimal weights, which do not yield zero error-mean. However, this can be corrected by modifying the bias weight of the output neuron properly to yield zero mean error over the training data set just after training ends. It must also be noted that the entropy is a cost function with many local minima. The nonlinear nature of the optimization was experimentally verified.

3. PROOF OF MINIMA EQUIVALENCE

Now we proceed with proving that the global minimum of the entropy is still a minimum of the non-parametrically estimated entropy for both Shannon's entropy definition and Renyi's entropy, when Parzen windowing with Gaussian kernels is utilized.

Shannon's Entropy:
Shannon's entropy is given by (2). Clearly, the global minimum of Shannon's entropy is achieved when the pdf of error is the Dirac-delta function.

$$\begin{aligned}
\frac{\partial H_S}{\partial e_j} &= -\frac{\partial}{\partial e_j}\int_{-\infty}^{\infty} f_e(\xi)\log f_e(\xi)d\xi \\
&= -\int_{-\infty}^{\infty}\left[\frac{\partial f_e}{\partial e_j}\log f_e + f_e\frac{\partial f_e / \partial e_j}{f_e}\right]d\xi \\
&= -\int_{-\infty}^{\infty}\frac{\partial f_e}{\partial e_j}\log f_e d\xi - \frac{\partial}{\partial e_j}\int_{-\infty}^{\infty} f_e d\xi \\
&= -\int_{-\infty}^{\infty}\frac{1}{N\sigma^2}(\xi - e_j)\kappa(\xi - e_j)\log f_e d\xi
\end{aligned} \tag{9}$$

Since entropy is independent of the mean of the error, we can concentrate on the case where mean of e is zero without loss of generality. The gradient of the entropy for Gaussian kernels is given in (9). Evaluating this gradient at $e = [e_1 \quad \dots \quad e_N] = 0$,

$$\left.\frac{\partial H_S}{\partial e_j}\right|_{e=0} = -\int_{-\infty}^{\infty}\frac{1}{N\sigma^2}\xi\kappa(\xi)\log\kappa(\xi)d\xi = 0 \tag{10}$$

since we are integrating an odd function. Hence $e = 0$ is a stationary point of $H_S(e)$. Now we continue with the computation of the Hessian to see if it is furthermore a minimum. Using the same approach as above, the diagonal and off-diagonal entries of the Hessian are found to be

$$\left.\frac{\partial^2 H_S}{\partial e_j^2}\right|_{e=0} = \frac{\partial}{\partial e_j}\left(\frac{\partial H_S}{\partial e_j}\right)_{e=0} = \frac{N-1}{N^2\sigma^2} \tag{11}$$

$$\left.\frac{\partial^2 H_S}{\partial e_k \partial e_j}\right|_{e=0} = \frac{\partial}{\partial e_k}\left(\frac{\partial H_S}{\partial e_j}\right)_{e=0} = \frac{-1}{N^2\sigma^2} \tag{12}$$

The eigenvalues of the Hessian matrix then can be computed to be $\lambda_0 = 0$ with multiplicity 1 and $\lambda_i = 1/(N\sigma^2)$ with multiplicity (N-1), hence the Hessian is positive semi definite. The value of the entropy may decrease in the direction of the eigenvector that corresponds to the 0 eigenvalue, which is found as

$$\vec{e}_0 = [1 \quad 1 \quad \dots \quad 1]^T \tag{13}$$

This means, if the error changes along this direction it will still have constant entries. However, we have shown above that, the mean of e does not change the value of the entropy. Therefore, when e changes along the direction of $\vec{e}_0$, the value of the entropy remains constant. So we conclude that Shannon's entropy approximated by Parzen windowing with Gaussian kernels has a minimum at the point where error is completely constant over the whole data set.

Renyi's Entropy:
Renyi's entropy is defined by (5), and is known to approach Shannon's entropy as α goes to 1 [7]. It is also independent of the mean of e.

The gradient of Renyi's entropy in the case of Gaussian kernels evaluated at $e = 0$ is found to be

$$\frac{\partial H_{R\alpha}}{\partial e_j} = \frac{\alpha}{N\sigma^2(1-\alpha)\int_{-\infty}^{\infty}\kappa^{\alpha}d\xi}\int_{-\infty}^{\infty}\xi\kappa^{\alpha}d\xi = 0 \qquad (14)$$

Hence $e = 0$ is a stationary point of Renyi's entropy approximated by Parzen windowing with Gaussian kernels. After some steps similar to those followed in the Shannon's entropy case, the diagonal and off-diagonal elements of the Hessian matrix evaluated at $e = 0$ are found as

$$\left.\frac{\partial^2 H_{R\alpha}}{\partial e_j^2}\right|_{e=0} = \frac{N-1}{N^2\sigma^2} \qquad (15)$$

$$\left.\frac{\partial^2 H_{R\alpha}}{\partial e_k \partial e_j}\right|_{e=0} = \frac{-1}{N^2\sigma^2} \qquad (16)$$

Note that the Hessian matrix for the Renyi's entropy is independent of α, and it is identical to the Shannon's entropy. Hence, it has the same eigenvalues as the Hessian matrix for Shannon's entropy. Similarly, the eigenvector corresponding to the zero eigenvalue is equal as well and therefore all the arguments related are also valid for Renyi's entropy. Thus we conclude that Renyi's entropy approximated by Parzen windowing with Gaussian kernels has a minimum at the point where the error is completely constant over the whole data set.

4. SHORT TERM PREDICTION OF CHAOTIC TIME SERIES

We have seen in the previous sections that the entropy criterion does not discriminate between pdfs that have distinct means but exactly the same higher order central moments. Therefore, when a TDNN is trained with the entropy criterion, the expected value of the error over the training data set will not converge to zero. We have mentioned that, this problem can easily be solved by adjusting the bias weight of the last layer to make the mean of estimation error zero over the training set after the learning finishes. If the output neuron in the TDNN is chosen to be a linear one, this modification is a simple addition of the mean of the current error to the bias weight of the output neuron.

As a case study, the short-term prediction of the Mackey-Glass chaotic time series [8] with parameter $\tau = 30$ using both MSE trained and (Renyi's) entropy trained TDNNs is presented here. The time-delay TDNN has a 3-tap input, 5 neurons in the hidden layer and a single linear output neuron. The embedding dimension is chosen to be 3 here. This is less than the embedding dimension suggested by Taken's Embedding Theorem, namely 5, for the Mackey-Glass series [9]. For this size of the reconstruction space the difficulty level of the prediction problem increases. Increased difficulty is desired since we know that even the MSE criterion performs quite well for this time series when a 5-tap input is employed.

The TDNNs are trained over a training data set of length 200 starting from 100 randomly chosen initial weights, so that hopefully the global optimal solution is one of the solutions suggested by this Monte Carlo type training approach. In this sequence, the weights of the MSE trained TDNN is iterated 100 times for each initial set of weights whereas those of the entropy trained TDNN are iterated for 30 times according to their corresponding backpropogation algorithms using the conjugate gradient algorithm [10]. At the end of the mentioned Monte Carlo training approach, the best set of weights obtained by each of the criteria are taken and checked for further improvement by employing a small constant step size. For this specific case, these extra iterations did not improve the cost function for either of the criteria. Finally, the bias weight of the output neuron of both artificial neural networks are adjusted to give zero mean error over the training data set after training had finished.

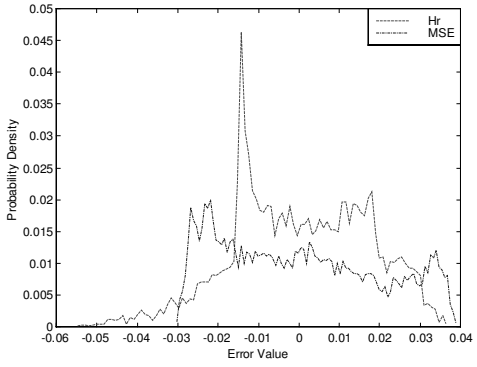

Figure 2: The probability distribution functions for the estimation errors of MSE and entropy trained TDNNs.

The trained networks are tested on an independently created test data set of length 10000. We do not present the error plots of the two TDNNs over the test data set here because they do not bear any information when presented in that form. Rather, we will concentrate on the statistical properties of the error signals. The pdfs of

these two error distributions approximated by a histogram of equally spaced 100 bins are shown in Fig. 2. As observed from this plot, the error distribution of the entropy trained TDNN is more concentrated around zero whereas the error distribution of the MSE trained TDNN is more uniformly distributed over its support.

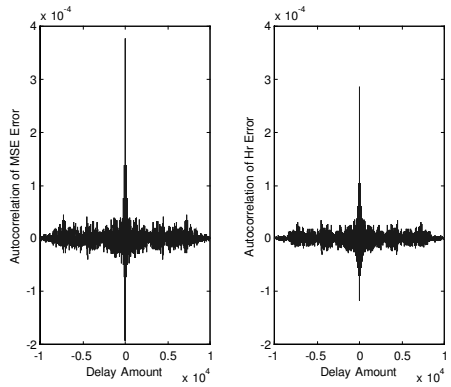

Figure 3: The autocorrelation functions of the estimation errors; MSE trained and entropy trained TDNNs respectively.

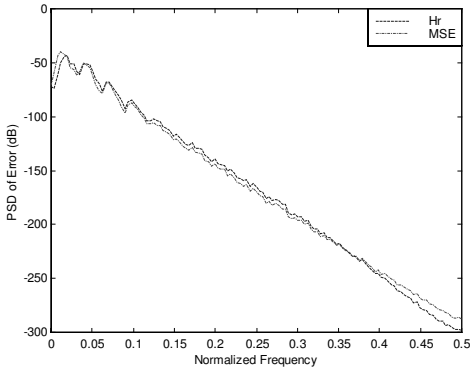

Figure 5: The PSD estimates of the estimation errors.

It is observed from Fig. 3 that, the autocorrelation function of entropy trained TDNN's error contains higher frequency components compared to that of the MSE trained TDNN. This is also evident from the PSD estimates of the two random processes. The PSD of the error of entropy trained TDNN decays slower than that of the MSE trained one in the lower frequencies.

For completeness, we present here the data with its estimations superimposed. The data window is chosen to include the point where the entropy trained TDNN makes its maximum error, namely the sample 3438.

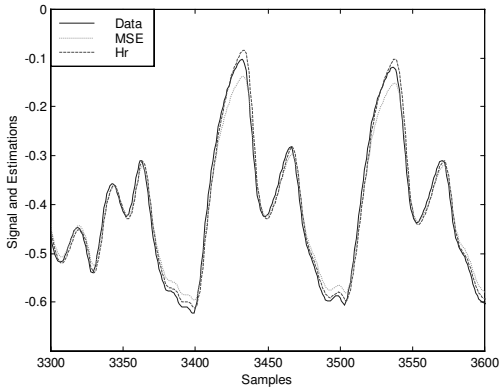

Figure 6: The short-term prediction of Mackey-Glass time series by MSE and entropy trained TDNNs.

We investigate the central moments of the two error distributions for a more quantitative comparison. The obvious ideal solution should have a Dirac-delta distributed error. All central moments of this desired error distribution are zero. In other words, we would like the central moments of the error distribution as close to zero as possible. The following table lists the first 5 central moments of the error distributions for the two training criteria.

n	$E[(e^{MSE}-\mu_e^{MSE})^n]$	$E[(e^{Ent}-\mu_e^{Ent})^n]$
1	0	0
2	$0.377\text{x}10^{-3}$	$0.285\text{x}10^{-3}$
3	$0.227\text{x}10^{-5}$	$-0.859\text{x}10^{-6}$
4	$0.271\text{x}10^{-6}$	$0.208\text{x}10^{-6}$
5	$0.372\text{x}10^{-8}$	$-0.27\ \text{x}10^{-8}$

Table 1: Central Moments of the error distributions.

Another way of looking at the same problem is to compare the central moments of the data distribution and the predicted time series distribution. What we are essentially trying to do is to match the probability distribution of the estimated time series to that of the data samples. In other words, we would like the central moments of the predictions to be as close to those of the test data set as possible. The central moments of the test data set and the predictions by the MSE and entropy trained TDNNs are given next.

n	$E[(x-\mu_x)^n]$	$E[(\hat{x}^{MSE}-\mu_{\hat{x}}^{MSE})^n]$	$E[(\hat{x}^{Ent}-\mu_{\hat{x}}^{Ent})^n]$
1	0	0	0
2	$1.585\text{x}10^{-2}$	$1.248\text{x}10^{-2}$	$1.6\text{x}10^{-2}$
3	$1.085\text{x}10^{-3}$	$0.785\text{x}10^{-3}$	$1.383\text{x}10^{-3}$
4	$6.186\text{x}10^{-4}$	$3.845\text{x}10^{-4}$	$6.838\text{x}10^{-4}$
5	$8.998\text{x}10^{-5}$	$5.125\text{x}10^{-5}$	$12.197\text{x}10^{-5}$

Table 2: Central Moments of the desired data samples and the prediction samples.

Note that the even moments of the entropy-trained predictor are very well matched to the original time series, while the odd moments are still in error, but above the true values. The moments of the MSE trained predictor are always smaller than the original, tending to a uniform error distribution.

These results point out clearly that the TDNN predictions approximate the statistical behavior of the Mackey-Glass chaotic attractor better when it is trained with the entropy criterion compared to the MSE criterion. The pdf estimations of the test data and the TDNN predictions for these data are presented in the following figure with a histogram of equally spaced 100 bins.

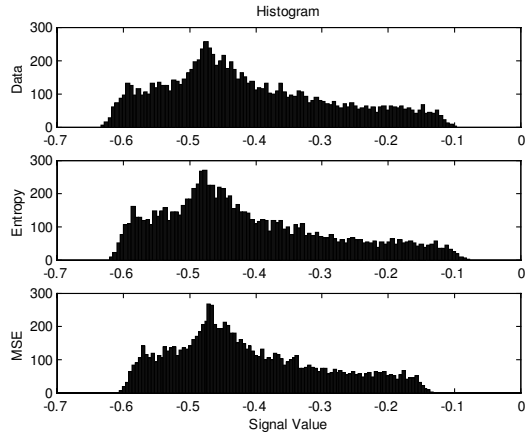

Figure 7: The histogram of the MG30 and its predictions by entropy-trained and MSE-trained TDNNs respectively.

5. CONCLUSIONS

In this paper, an information theoretical learning criterion for adaptive systems, namely estimation-error-entropy-minimization, has been investigated. An analytical proof, which shows that Parzen windowing approximation with Gaussian kernels of the probability distribution function to be used in entropy computation preserves the original global minimum of the entropy as one of the minima, possibly the global minimum, of the estimated entropy function. This proof enables us to use Parzen windowing with Gaussian kernels in entropy estimation safely from the entropy-minimization point of view.

A time-delay neural network has been trained for the short-term prediction of the Mackey-Glass chaotic time series using both the entropy and mean-square-error criteria. A Monte Carlo approach has been taken in terms of the initial weights of the time-delay neural network in order to avoid the local minimum solutions. However, it became evident that even with this approach, the global minimum of the MSE has not been achieved, since the entropy solution had smaller error power. The best solutions obtained by the mean-square-error and the entropy criteria were compared in terms of the statistical behavior of the estimation errors and the prediction values themselves. The comparison of central moments of the error distributions revealed the fact that the error of the entropy-trained time-delay neural network is closer to the ideal solution. The comparison of the central moments of test data samples and the prediction samples lead to the same conclusion. The predictions of the time-delay neural network trained with the entropy criterion approximate the statistical behavior of the actual output of the unknown system better than the one trained with the men-square-error criterion.

Acknowledgement: This work was partially supported by NSF grant ECS-9900394.

REFERENCES

[1] Haykin, S., *Introduction to Adaptive Filters*, MacMillan, NY, 1984.
[2] Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, 3rd ed., McGraw-Hill, Inc., USA, 1991.
[3] Fisher, J.W., *Nonlinear Extensions to the Minimum Average Correlation Energy Filter*, Ph.D. Dissertation, University of Florida, 1997.
[4] Parzen, E., "On Estimation of a Probability Density Function and Mode", in *Time Series Analysis Papers*, Holden-Day, Inc., CA, 1967.
[5] Rumelhart, D.E., Chauvin, Y., *Backpropogation: Theory, Architectures, and Applications*, Lawrence Erlbaum Associates, NJ, 1995.
[6] Renyi, A., *A Diary on Information Theory*, Wiley, NY, 1987.
[7] Principe, J., Xu, D., Fisher, J., Information Theoretic Learning, in *Unsupervised Adaptive Filtering*, Simon Haykin Editor, 265-319, Wiley, 2000.
[8] Kaplan, D., Glass, L., *Understanding Nonlinear Dynamics*, Springer-Verlag, NY, 1995.
[9] Kuo, J.M., *Nonlinear Dynamic Modeling With Artificial Neural Networks*, Ph.D. Dissertation, University of Florida, 1993.
[10] Luenberger, D.G., *Linear and Nonlinear Programming*, Addison-Wesley Pub. Co., MA, 1973.