A RECURSIVE RENYI'S ENTROPY ESTIMATOR

Deniz Erdogmus[1], Jose C. Principe[1], Sung-Phil Kim[1], Justin C. Sanchez[2]

Computational NeuroEngineering Laboratory,
[1]Electrical & Computer Engineering Department
[2]Biomedical Engineering Department
University of Florida, Gainesville, FL 32611.
[deniz,principe,phil,justin]@cnel.ufl.edu

Abstract. Estimating the entropy of a sample set is required in solving numerous learning scenarios involving information theoretic optimization criteria. A number of entropy estimators are available in the literature; however, these require a batch of samples to operate on in order to yield an estimate. In this paper, we derive a recursive formula to estimate Renyi's quadratic entropy on-line, using each new sample to update the entropy estimate to obtain more accurate results in stationary situations or to track the changing entropy of a signal in nonstationary situations.

INTRODUCTION

Entropy, defined as the average information content of the outcomes of a random event, is first introduced by Shannon [1] in the context of digital communications. Although Shannon himself did not originally refer to his work as *information theory*, the mathematical elegance of his contribution attracted the attention of numerous researchers, which helped build this celebrated theory. The implications of information theory are far reaching; it is not merely of interest to researchers working on digital communications or to pure mathematicians. In the recent decades, information theory has found applications in many different areas of science ranging from social sciences to physical sciences and applied sciences, including numerous areas of engineering.

In the last decade, the literature of adaptive systems research has also taken its toll from this increased interest on information theory. Successful solutions to important practical engineering problems involving the training of adaptive filters and neural networks have been developed through the use of information theoretic optimization criteria. Blind source separation and blind deconvolution are possibly the most frequently studied problems in this regard. Estimating the entropy of a signal is essential in determining the solution of many other adaptive learning problems as well. The design of information theoretically optimal state estimators in control system design [2], information theoretic supervised learning of neural networks [3,4], information theoretic subspace projections [5] and feature extraction [6,7] are examples of these problems. Information theory plays a role in the self-organization of adaptive systems as well, as demonstrated by Linsker [8].

We have recently introduced a nonparametric estimator for Renyi's entropy, which is used successfully in the information theoretic training of linear and nonlinear adaptive systems [9,10]. The main drawback of (off-line) batch training was partially overcome by the introduction of the stochastic information gradient [11]. However, as with all stochastic training algorithms, the major problem of this latter was the misadjustment in the vicinity of the desired optimal solution.

Since there is a need to achieve smooth and fast convergence to the optimal solution using low-complexity learning rules in on-line adaptation, we desire to come up with a recursive entropy estimator. This estimator, possibly using a forgetting factor, will preserve some information from the past, while updating the entropy estimate based on the newly acquired samples. The gradient of this recursive entropy estimator could then be utilized in training adaptive systems on-line.

BATCH QUADRATIC ENTROPY ESTIMATOR

It is possible to derive two different recursive entropy estimators, one for stationary environments, one for nonstationary environments. In order to derive the entropy recursion for stationary situations and also to form a basis for reference, we first present the batch estimator for Renyi's quadratic entropy. For a random variable X with probability distribution function (pdf) $f_X(.)$, Renyi's entropy of order-α is defined as [12]

$$H_\alpha(X) = \frac{1}{1-\alpha}\log\int_{-\infty}^{\infty} f_X^\alpha(x)dx = \frac{1}{1-\alpha}\log E_X[f_X^{\alpha-1}(X)] \tag{1}$$

Using Parzen windowing with kernel function $k_s(.)$ to estimate the pdf from its samples $\{x_1,\ldots,x_N\}$ [13], and approximating the expectation operator with sample mean, we obtain the following estimator for Renyi's entropy [4].

$$\hat{H}_\alpha(X) = \frac{1}{1-\alpha}\log\left[\frac{1}{N^\alpha}\sum_{j=1}^{N}\left(\sum_{i=1}^{N}k_s(x_j - x_i)\right)^{\alpha-1}\right] \tag{2}$$

Specifically for the choice of α=2, we obtain the quadratic entropy, which is given and estimated by

$$H_2(X) = -\log E_X[f_X(X)] \cong -\log\frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N}k_s(x_j - x_i) \tag{3}$$

The argument of the 'log' is named the quadratic information potential due to resemblance between this quantity and the physical potential energy of an ensemble of particles [14]. Under the entropy-based training rules, the samples start behaving similar to physical particles; therefore under the same analogy, they were named *information particles*.

EXACT RECURSION FOR QUADRATIC ENTROPY

Investigating the structure of the nonparametric estimator for quadratic information potential in (3), we notice that it is possible to obtain a recursive formula to update the information potential estimate when a new sample is acquired. Suppose that at time k, when we already have k samples, the quadratic information potential is estimated to be (dropping '$_2$' from this point on)

$$\hat{V}_k(X) = \frac{1}{k^2}\sum_{j=1}^{k}\sum_{i=1}^{k} k_s(x_j - x_i) \tag{4}$$

Suppose at time $k+1$ we get a new sample x_{k+1} and we wish to update our estimate. Assuming that the kernel function is selected to be an even-symmetric pdf,

$$\begin{aligned}\hat{V}_{k+1}(X) &= \frac{1}{(k+1)^2}\sum_{j=1}^{k+1}\sum_{i=1}^{k+1} k_s(x_j - x_i) \\ &= \frac{k^2}{(k+1)^2}\hat{V}_k + \frac{1}{(k+1)^2}\left[2\sum_{i=1}^{k} k_s(x_{k+1} - x_i) + k_s(0)\right]\end{aligned} \tag{5}$$

Once the information potential estimate is updated, the new entropy estimate can be obtained by simply evaluating $\hat{H}_{k+1}(X) = -\log\hat{V}_{k+1}(X)$. Since this recursion yields exactly the same estimate as the batch estimator in (3) at every time instance, we will call this the *exact recursive entropy estimator*. This exact recursion is useful for estimating the entropy of stationary signals, however, due to its increasing memory depth, it is not suitable for nonstationary environments. Therefore, we will employ the fixed forgetting factor approach to derive one that would serve satisfactorily in such situations.

FORGETTING RECURSION FOR QUADRATIC ENTROPY

We start by defining a recursive Parzen window estimate. Suppose that at time k, we already have the pdf estimate $f_k(x)$ for $f_X(x)$. Using the new sample x_{k+1}, we update this pdf estimate according to

$$f_{k+1}(x) = (1-\lambda) f_k(x) + \lambda k_s(x - x_{k+1}) \tag{6}$$

The initial pdf estimate could be selected as $f_1(x) = k_s(x - x_1)$. Substituting the recursive pdf estimate in (6) for the actual pdf in the definition given in (3), we obtain the recursion for the quadratic information potential.

$$\begin{aligned}\overline{V}_{k+1} &= E_X[f_{k+1}(X)] = (1-\lambda)E_X[f_k(X)] + \lambda E_X[k_s(X - x_{k+1})] \\ &\cong (1-\lambda)\overline{V}_k + \frac{\lambda}{L}\sum_{i=k-L+1}^{k} k_s(x_i - x_{k+1})\end{aligned} \tag{7}$$

The recursion in (7) is named as the *forgetting recursive entropy estimator.* The parameters λ, L, and s are called the forgetting factor, window length, and kernel size, respectively. These free design parameters have an effect on the convergence

properties of this recursive entropy estimator. These will be investigated in the following sections.

An interesting relationship between the exact and forgetting recursive entropy estimators of (5) and (7) is that, if we replace the fixed memory depth and the fixed window length of (7) with dynamic ones, the two recursions asymptotically converge to the same value. In order to see this, we set $l = 1 - k^2/(k+1)^2$ and $L=k$. Then take the limit of the difference between (5) and (7) as k goes to infinity.

$$\lim_{k\to\infty}(\hat{V}_{k+1} - \overline{V}_{k+1}) = \lim_{k\to\infty}\left[\begin{array}{l} \dfrac{k^2}{(k+1)^2}\hat{V}_k - (1-l)\overline{V}_k + \dfrac{1}{(k+1)^2}k_s(0) \\ + \dfrac{2}{(k+1)^2}\displaystyle\sum_{i=1}^{k} k_s(x_{k+1} - x_i) - \frac{l}{k}\sum_{i=1}^{k} k_s(x_i - x_{k+1}) \end{array}\right] = 0 \qquad (8)$$

The practically important property of this recursive estimator is that it reduces the computational complexity from $O(N^2)$ to $O(L)$. This is a drastic reduction in the computation power requirements. The forgetting recursive entropy estimator also enjoys a reduced memory requirement compared to the exact recursion and the batch formula.

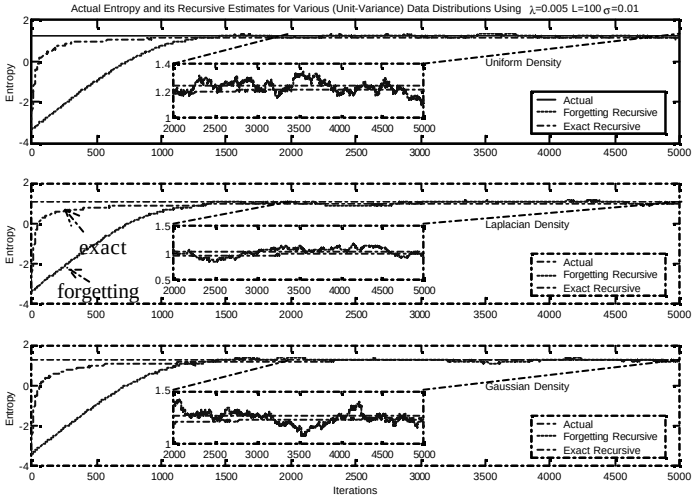

Figure 1. Actual entropy and its exact and forgetting recursive estimates for uniform, Laplacian and Gaussian densities.

CASE STUDIES

In this section, we investigate the performance of the recursive entropy estimators proposed above. We start by demonstrating the convergence properties

of both estimators to the true entropy value of the pdf underlying the data that is being presented. In these simulations, we have utilized 5000 samples generated by zero-mean, unit-variance uniform, Laplacian, and Gaussian distributions. For these density functions, both the exact and forgetting recursions are evaluated over the samples. The estimated entropy values using a Gaussian kernel with size $s = 0.01$ as well as the actual entropy of the true pdf of the data are shown in Fig. 1. For the forgetting recursion, the forgetting factor is selected to be 0.005 and the window length is chosen as 100.

In our second set of simulations, we investigate the effect of the forgetting factor on the convergence time and the convergence accuracy (variance after convergence) of the forgetting estimator in (7). For this purpose, we have utilized this recursion on a uniform density for 10000 iterations. Three different values are used for the forgetting factor: 0.001, 0.003, and 0.01. The convergence plots of the estimates are shown in Fig. 2. Starting from the same initial estimate, the three recursions converge after approximately 8000, 2500, and 1000 iterations. As expected, the faster the convergence, the larger the estimation variance is. When we evaluate the variances of the estimated entropy values over the last 1000 samples of each convergence curve, we see that larger forgetting factors result in larger variance; the variances are respectively, $1.1\text{x}10^{-4}$, $9.5\text{x}10^{-4}$, and $2.7\text{x}10^{-3}$. In these runs, we have used L=100 and $s = 0.01$. This result conforms to the well-known general behavior of the forgetting factor in recursive estimates. There is an intrinsic trade-off between speed and variance, which the designer must consider in selecting the forgetting factor.

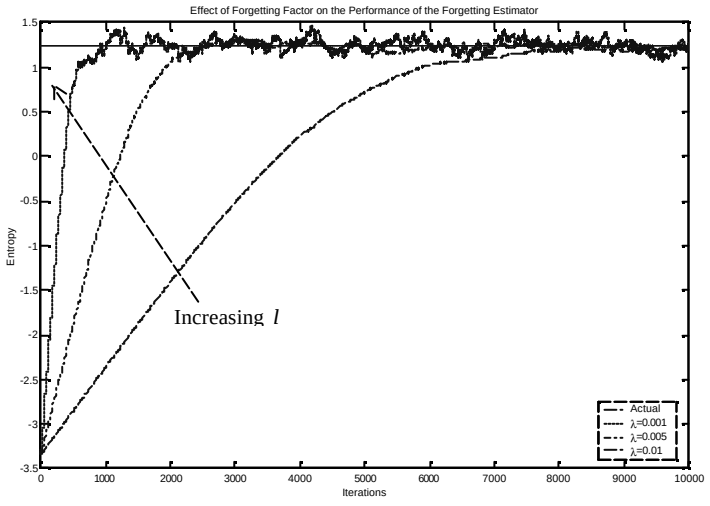

Figure 2. Comparison of the convergence properties of the forgetting estimator for different values of the forgetting factor.

Our third set of simulations study the effect of the window length, which approximates the expectation operator. For this purpose, we have fixed the forgetting factor to 0.002, and the kernel size to 0.01 in (7). Three values of L are tried: 10, 100, and 1000. The results of the recursive estimation using these three different window lengths are shown in Fig. 3. As expected, the speed of convergence is not affected by the variations in this parameter. Only, the estimation variance after convergence is greatly affected. Specifically, the variance of the estimates for these three cases over the last 1000 iterations of the recursion are 6.7×10^{-3}, 7.1×10^{-4}, and 2.2×10^{-5}. This conforms with the general behavior of the sample mean approximation for expectation: The more samples used, the smaller the variance gets. The trade-off in the selection of this parameter is between the accuracy after convergence and the memory requirement. The larger L gets, the more storage space is required for holding previous samples in memory; on the other hand, estimation variance is decreased.

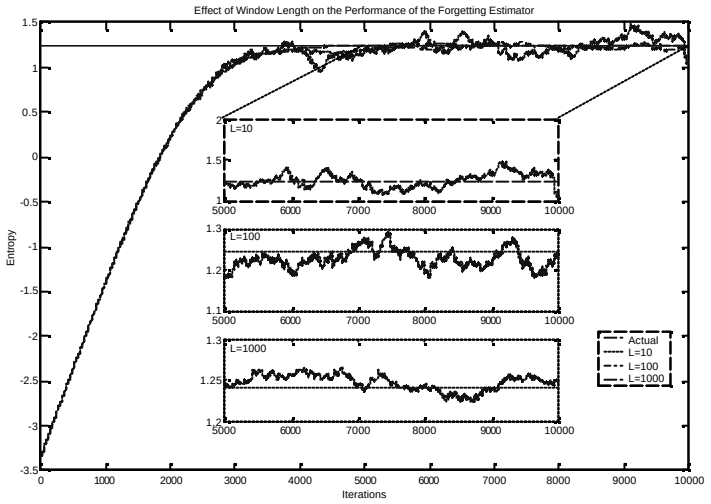

Figure 3. Comparison of the convergence properties of the forgetting estimator for different values of the window length.

Our fourth set of simulations investigates the effect of kernel size on the variance and bias of the forgetting recursive estimator. As we know, Parzen windowing has a bias that increases with larger kernel sizes, whereas its variance increases with smaller kernel sizes. In accordance with this property of Parzen windowing, we expect our non-parametric estimator to exhibit similar behavior under the variations of kernel size. The convergence plots of the recursions for various values of the kernel size are shown for a uniformly distributed data set in Fig. 4. In all runs, the forgetting factor was fixed to 0.002 and the window length was taken as 100. For the Gaussian kernel function with sizes of 0.001, 0.01, 0.1, and 1, the bias over the last 1000 samples of the recursions turned out to be

$5.1x10^{-2}$, $2.2x10^{-2}$, $1.3x10^{-2}$, and $2.4x10^{-1}$; the variances were also computed and found to be $3.9x10^{-3}$, $1.6x10^{-4}$, $2.9x10^{-5}$, and $3.4x10^{-5}$. As expected, the smallest kernel size resulted in the largest variance and the largest kernel size resulted in the largest bias.

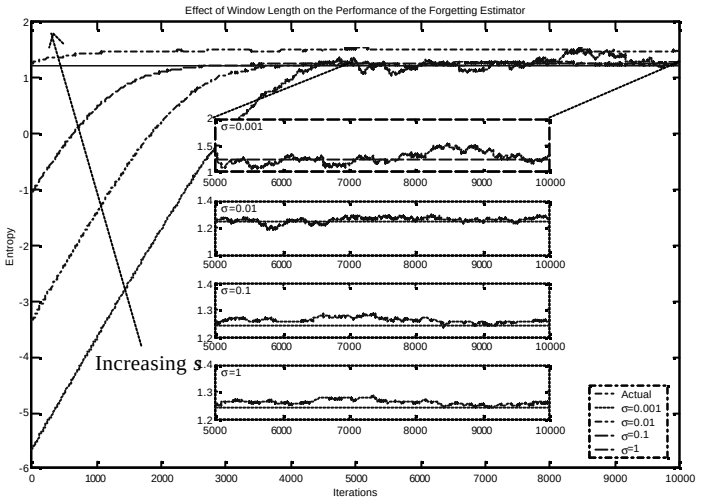

Figure 4. Comparison of the convergence properties of the forgetting estimator for different values of the kernel size.

Our fifth simulation demonstrates the tracking capability of the forgetting estimator in (7). For this simulation, we have utilized a forgetting factor of 0.002, a window length of 100, and a base kernel size of 0.01. The recursion is initialized to the entropy of the kernel function. In order to enhance the differences between the entropies of the uniform, Laplacian, and Gaussian pdfs, we have scaled their standard deviations by the coefficients 1, 5, and 0.2 respectively. In estimating the entropy of a scaled sample sequence using the presented estimators, it is also necessary to scale up or down the kernel size according to the standard deviation of the samples [15]. The base kernel size is selected to suit a unit-variance data pdf. Since, in general, the variance of the data pdf is unknown, we employ a recursive estimator to estimate this parameter as well.

$$\mathrm{var}(x)_{k+1} = (1-l)\,\mathrm{var}(x)_k + l\ x_k^2 \tag{9}$$

This recursive variance estimator assumes the same forgetting factor value of 0.002. The algorithm is presented with a sequence of 30000 random samples generated by zero-mean uniform, Laplacian, and Gaussian distributions with standard deviations 10, 1, and 30 respectively. For a comparison, we also present the entropy tracking results where these actual scale factors are utilized to scale up/down the kernel size at the instants of switching between pdfs. The initial scale factor estimate in (9), i.e. the estimate of the standard deviation of the pdf

underlying the samples, is set to 1. We observe from Fig. 5 that even though the scale estimates are not initially accurate, the entropy estimates converge towards the actual entropy value and as soon as the scale factor estimate converges, the difference between the two entropy estimates that use the estimated and actual values of the scale factors drop back to a negligible level.

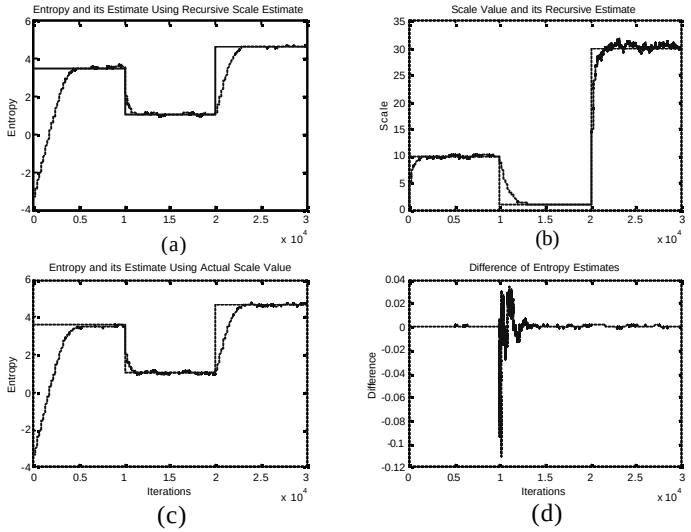

Figure 5. Comparison of entropy estimates using the actual and estimated values of the scale factor a) entropy estimate using the estimated scale factor b) scale factor estimate c) entropy estimate using the actual scale factor d) difference between the two entropy estimates using the actual and estimated values of the scale factor.

CONCLUSIONS

Estimating the entropy of a sample sequence or a signal has numerous practical applications in signal processing and adaptive system training. We have previously introduced a robust and data-efficient estimator for Renyi's entropy, which proved to result in superior learning algorithms in terms of convergence speed and data efficiency in solving problems such as blind source separation, blind deconvolution and other supervised and unsupervised training applications. In this paper, we have targeted the major drawback of batch computation requirement that this entropy estimator enforced. We have derived two recursive formulations to extend the estimation flexibility of our nonparametric Renyi's entropy estimator. These two recursions, named exact recursive entropy estimator and forgetting recursive entropy estimator, allow on-line treatment of the entropy of signals for computationally simple and fast manipulation of the adaptive system parameters. The drastic reduction from $O(N^2)$, where N is the batch size, to $O(L)$, where L is the window length, in computational complexity, makes the forgetting recursive

entropy estimator attractive for use in on-line information theoretic learning scenarios, where an entropy-based cost function is to be optimized.

We have investigated the convergence properties of these recursive estimators in simulations and studied the effect of design parameters, namely the forgetting factor, the window length, and the kernel size, on the convergence speed and final estimation variance through simulations. The results were in accordance with our expectations. In short, increased forgetting factor resulted in faster convergence and larger variance, increased window length resulted in smaller variance and had no effect on convergence time, and finally, increased kernel size resulted in smaller variance and larger estimation bias.

Acknowledgments: This work is partially supported by the grants NSF-ECS-9900394 and ONR-N00014-01-1-0405.

REFERENCES

[1] C.E. Shannon, "A mathematical theory of communication," Bell Sys. Tech. J., vol. 27, pp. 379-423,623-653, 1948.

[2] X. Feng, K.A. Loparo, Y. Fang, "Optimal State Estimation for Stochastic Systems: An Information Theoretic Approach," IEEE Transactions on Automatic Control, vol. 42, no. 6, pp. 771-785, 1997.

[3] D. Erdogmus, J.C. Principe, "An Entropy Minimization Algorithm for Supervised Training of Nonlinear Adaptive Systems," to appear in IEEE Transactions on Signal Processing 2002.

[4] D. Erdogmus, J.C. Principe, "Generalized Information Potential Criterion for Adaptive System Training," to appear in IEEE Trans. on Neural Networks, 2002.

[5] J.W. Fisher, Nonlinear Extensions to the Minimum Average Correlation Energy Filter, PhD Dissertation, University of Florida, 1997.

[6] J. Atick, "Could Information Theory Provide an Ecological Theory of Sensory Processing?" Network, vol. 3, pp. 213-251, 1992.

[7] J. Atick, A. Redlich, "Convergent Algorithms for Sensory Receptive Field Development," Neural Computation, vol. 5, pp. 45-60, 1993.

[8] R. Linsker, "An Application of the Principle of Maximum Information Preservation to Linear Systems," in D.S. Tourezky (ed.), Morgan-Kauffman, San Francisco, 1988.

[9] D. Erdogmus, J.C. Principe, "Convergence Analysis of the Information Potential Criterion in Adaline Training," Proceedings of Neural Networks for Signal Processing 2001 (NNSP XI), pp. 123-132, Falmouth, MA, Sep 2001.

[10] D. Erdogmus, D. Rende, J.C. Principe, T.F. Wong, "Nonlinear Channel Equalization Using Multilayer Perceptrons with Information-Theoretic Criterion," Proceedings of Neural Networks for Signal Processing 2001 (NNSP XI), pp. 443-451, Falmouth, MA, Sep 2001.

[11] D. Erdogmus, J.C. Principe, "An On-Line Adaptation Algorithm for Adaptive System Training with Minimum Error Entropy: Stochastic Information Gradient," in Proc. of Independent Component Analysis 2001 (ICA'01), 2001.

[12] A. Renyi, Probability Theory., American Elsevier Pub. Co., New York, 1970.

[13] E. Parzen, "On Estimation of a Probability Density Function and Mode,". in Time Series Analysis Papers, Holden-Day, Inc., California, 1967.

[14] J.C. Principe, D. Xu, J. Fisher, "Information Theoretic Learning," in Unsupervised Adaptive Filtering, vol I, S. Haykin (ed.), Wiley, New York, 2000.

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