A GENERAL PROBABILISTIC FORMULATION FOR NEURAL CLASSIFIERS

Tülay Adalı
Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County
Baltimore, MD 21250, USA

Abstract- We use partial likelihood (PL) theory to introduce a general probabilistic framework for the design and analysis of neural classifiers. The formulation allows for the training samples used in the design to have correlations in time, and for use of a wide range of neural network probability models including recurrent structures. We use PL theory to establish a fundamental information-theoretic connection, show the equivalence of likelihood maximization and relative entropy minimization, *without* making the common assumptions of independent training samples and true distribution information. Large sample optimality properties of PL can also be established under mild regularity conditions which allows adaptive-structure and robust classifier designs by using modified likelihood functions and information-theoretic criteria.

I. INTRODUCTION

The probabilistic view of a neural network classifier such that the network outputs are associated with posterior class probabilities is quite attractive for a number of reasons. Among others, this view offers advantages both in understanding the properties of learning in neural networks and in developing new approaches for learning.

The commonly used mean square error criterion can be shown to approximate the posterior class probabilities in a neural classifier [5] which are the main objects of interest in a classification problem. A more attractive approach, however, is to train the network with an objective that directly aims at estimating the probabilities. Information theoretic criteria such as the relative entropy [7] (Kullback-Leibler distance) satisfy this requirement. Relative entropy measures the information theoretic distance between two target distributions, the true and the estimated distributions, in a supervised learning setting. Maximum likelihood (ML) estimator as the parameter point for which the observed data is most likely is intuitively a very reasonable choice and it possesses nice large sample optimality properties such as consistency, asymptotic normality, and asymptotic efficiency as well as invariance

0-7803-5060-X/98/$10.00 © 1998 IEEE

with respect to functions of the parameters. Both objectives, the ML and the RE, let learning in a neural classifier to be viewed as statistical estimation of a probability model parametrized by a neural network structure. The equivalence of maximum likelihood, the foundation for parametric statistical inference, and minimum relative entropy learning can easily be established under two key *assumptions* [2, 5, 12]:

(i) The observations are independent and identically distributed (i.i.d.)

(ii) The *true* class probabilities given the observed data are known.

The first assumption is required to be able to characterize the likelihood, and the second for learning on the relative entropic cost. Obviously, both assumptions lack plausibility. Reference [9] notes the unrealism in expecting the correlations required to make the classes distinguishable not to be reflected in correlations between the observations. However, in [9], it is also added that a classifier designed with such an objective (likelihood expressed as product of independent and identical distributions) usually leads to good classification results. The problems associated with the independence assumption is more obvious in a time series application where the observation vector is defined as the last L observations and each consecutive vector has $L-1$ common elements (assuming scalar observations at each time instant.) The second assumption given above requires that, for each observation, we have the true probability distribution over class labels. Though not realistic, the assumption is necessary to be able to train the classifier using the relative entropy criterion [2] as the measure requires the true conditional distribution and this assumption provides a convenient way to map true class labels to probabilities.

In this paper, we view learning in a neural network classifier as statistical estimation of a parametrized probability model and use partial likelihood (PL) theory [3], to present a general probabilistic framework for designing neural classifiers for problems in which ordering of training pairs is conveniently defined or is essential as in time series problems. We establish the equivalence of relative entropy minimization and likelihood maximization under two regularity conditions. These are general conditions that are functions of the selected probability model, i.e., the neural classifier structure. We note that these conditions are satisfied for the sigmoidal feedforward classifier that uses the logistic activation function. Hence, the result justifies the use of either the likelihood or the relative entropy formulation for estimating the distribution for the general case of dependent observations without requiring that the true conditional distribution is known. The conditions of the theorem can be shown to hold for other network structures as

well, such as the finite normal mixtures (FNM) model [10].

Besides its attractiveness for studying and understanding the properties of the neural classifier, the probabilistic formulation provides considerable advantages for deriving new learning procedures. The performance advantages of using the relative entropic (or partial likelihood) cost are presented in [1] for a binary classification example. This is a direct consequence of the *well-formed* property [13] of the relative entropic cost which guarantees the recovery of steepest descent learning from convergence at the wrong extreme, a property not satisfied by the mean square error (MSE) cost function. The information theoretic connection we present also allows derivation of new adaptive algorithms based on information-geometric alternating projections [4] as well as a methodology to understand the properties of various estimation/learning schemes as discussed in [10].

II. NEURAL NETWORKS AS ESTIMATOR OF MULTI-CLASS POSTERIOR PROBABILITIES

Assume that we have a training set $\mathcal{T}$ of N related input and output pairs $\mathcal{T} = \{x_n, \mathbf{y}_n\}_{n=1}^N$ and the problem is to train the classifier such that for a given observation $\mathbf{y}_k$, $\mathbf{x}_k$ will be assigned to one of m classes $C_1, C_2, \cdots, C_m$, such that $\mathbf{x}_k$ takes a value from a finite alphabet $\mathcal{S} = \{\mathbf{a}_0, \mathbf{a}_1, ..., \mathbf{a}_m\}$. The actual value that the random variable $\mathbf{x}_k$ takes, i.e., the value $\mathbf{a}_i$ is of consequence only in the case of training using relative errors, e.g. when using the MSE cost.

Given the posterior class probabilities $P(C_i|\mathbf{y})$ for $i = 1, 2, \cdots, m$, Bayes classifier will assign $\mathbf{y}$ to class C_i if $P(C_i|\mathbf{y}) > P(C_j|\mathbf{y})$ $\forall j \neq i$, a choice which minimizes the classification error probability. Note that, the total distribution information, rather than that of the most likely class only, can be useful depending on the particular application. Since, the goal is the estimation of the probabilities $P(C_i|\mathbf{y})$ $\forall \mathbf{a}_i \in \mathcal{S}$, we can use a feedforward neural network probability model such that

$$P_{\theta}(C_i|\mathbf{y}) = f_i(\boldsymbol{\theta}, \mathbf{y}) \tag{1}$$

where $\boldsymbol{\theta}$ is the vector of network parameters which we can estimate/learn using the appropriate criterion. It is important to remember that the probabilistic formulation brings the additional constraint that the network outputs lie in the range [0,1] and that $\sum_{i=1}^m f_i(\boldsymbol{\theta}, \mathbf{y}) = 1$.

For the binary case, $\mathcal{S} = \{0, 1\}$, we only need to estimate $P_{\theta}(C_1|\mathbf{y})$ as $P_{\theta}(C_2|\mathbf{y}) = 1 - P_{\theta}(C_1|\mathbf{y})$. For example, we can use the posterior

probability model:

$$P_\theta(C_1|\mathbf{y}) = f(\boldsymbol{\theta}, \mathbf{y}) = g\left(\sum_{i=1}^{q} h\left(\mathbf{y}_n{}^T \mathbf{w}^i\right) v^i\right) \tag{2}$$

where $\mathbf{w}^i \in \mathbf{R}^{L\times 1}$ is the weight vector between the input layer and the hidden node i, ($i = 1, \cdots, q$, where q is the number of hidden nodes), $\mathbf{y}_n \in \mathbf{R}^{L\times 1}$, and v^i is the weight between the hidden node i and the output node. We can represent the entire set of weights by $\boldsymbol{\theta} = [\mathbf{W}, \mathbf{v}] \in \mathbf{R}^{q\times(L+1)}$ where $\mathbf{W} = [\mathbf{w}^1, \mathbf{w}^2, \cdots, \mathbf{w}^q]^T \in \mathbf{R}^{q\times L}$ and $\mathbf{v} = [v^1, v^2, \cdots, v^q]^T \in \mathbf{R}^{q\times 1}$. The hidden node activation function $h(\cdot)$ can be chosen to ensure network approximation capabilities [12], e.g. it can be chosen as the familiar logistic or the radial basis function. However, for learning parameters by gradient optimization, $g(\cdot)$ has to be chosen such that $g'(\cdot) > 0$.

For the general case of multiple classes, to ensure that the network outputs are valid probabilities (i.e., they sum upto one), there is usually a second normalization stage that is cascaded to the feedforward structure. The exponential normalization, the so-called *softmax* function [2] has been the most popular for multi-class learning with the likelihood cost. We can introduce softmax normalization for a single hidden layer feedforward neural classifier with logistic activation function $h(\cdot)$ as

$$f_i(\boldsymbol{\theta}, \mathbf{y}_n) = \frac{\exp(\phi_i)}{\sum_{j=1}^{m} \exp(\phi_j)} \tag{3}$$

where $\phi_j = \sum_{i=1}^{q} h\left(\mathbf{y}_n{}^T \mathbf{w}^i\right) v^{ij}$ with $\mathbf{w}^i \in \mathbf{R}^{L\times 1}$ and $\mathbf{y}_n \in \mathbf{R}^{L\times 1}$ defined similar to (2), and v^{ij} as the weight between the hidden node i and the jth output node. In [6], a modified softmax normalization is used that eliminates the inherent redundancy in the standard softmax and the training is achieved by a Gauss-Newton scheme which is shown to increase convergence rate considerably compared to learning by the Robbins-Monro procedure.

Note that, rather than modeling the probability mass function (pmf), we can also choose probability models with continuous outputs, i.e., can model the probability density function (pdf) as a direct consequence of the Bayesian formula as shown in [10].

III. LIKELIHOOD FOR INDEPENDENT OBSERVATIONS AND THE INFORMATION-THEORETIC VIEW

Assuming N *independently* drawn observations $\mathbf{y}_j$, we can write the likelihood of the network parameters as

$$\mathcal{L}_N(\theta) = \prod_{j=1}^{N} \prod_{i=1}^{m} f_i(\boldsymbol{\theta}, \mathbf{y}_j)^{T_{i,j}} \tag{4}$$

where the indicator index $T_{i,j}$ is defined as $T_{i,j} = 1$ if $\mathbf{x}_j \in C_i$ and 0 otherwise. Given a selected network structure, maximum likelihood (ML) selects the network parameter $\boldsymbol{\theta}_0$ which results in a distribution that best matches the observed data. In a problem where the ultimate aim is determining the probability distribution, such as the general classification problem we consider, this is intuitively a very reasonable choice. Also, with nice large sample optimality properties such as consistency, asymptotic normality, and asymptotic efficiency as well as invariance with respect to functions of the parameters, ML estimation is a very desirable general statistical framework in which to pose classification problems.

The relative entropy (RE), or the Kullback-Leibler distance, [7], on the other hand, is a fundamental information-theoretic measure of how accurate the estimated probability distribution p_θ is an approximation to the true probability distribution p and is given by $D(p\|p_\theta) = E\left\{\log \frac{p}{p_\theta}\right\}$ where the expectation is with respect to the true distribution p. The RE is always nonnegative and is zero only when the two distributions match, $p = p_\theta$. We can write the RE for our classification problem as

$$D_j(p\|p_\theta) = \sum_{i=1}^{m} P(C_i|\mathbf{y}_j) \log \frac{P(C_i|\mathbf{y}_j)}{f_i(\boldsymbol{\theta}, \mathbf{y}_j)}. \tag{5}$$

and define the *accumulated* relative entropy (ARE) as the total Kullback-Leibler discriminatory information contained in the training set $\mathcal{T}$ as

$$\mathcal{I}_N(\theta) = \sum_{j=1}^{N} D_j(p\|p_\theta) \tag{6}$$

Assuming that the *true* probability distribution over labels $P(C_i|\mathbf{y}_n)$ is available and is given by $T_{i,j}$ defined above, i.e., letting $P(C_i|\mathbf{y}_j) \approx T_{i,j}$ $\forall \mathbf{a}_i \in \mathcal{S}$, we can rewrite ARE as

$$\mathcal{I}_N(\theta) = -\sum_{j=1}^{N} H(T_j) - \sum_{j=1}^{N} \sum_{i=1}^{m} T_{i,j} \log f_i(\mathbf{y}, \boldsymbol{\theta}). \tag{7}$$

Since the first term in (7), the entropy of T distribution, is not a function of the network parameters, it is easy to see that maximization of log-likelihood is equivalent to minimization of relative entropy. Hence at least one $\arg\min_\theta \mathcal{I}_N(\boldsymbol{\theta})$ tends to one $\arg\max_\theta \bar{\mathcal{L}}_N(\boldsymbol{\theta})$. Thus the optimal model parameters $\boldsymbol{\theta}_0$ have the fundamental information theoretic interpretation that they minimize the Kullback-Leibler information given a probability model. Thus viewing learning as related to Kullback-Leibler information minimization in this way implies that learning is a *maximum likelihood* statistical estimation procedure. However, note that this is true under the two key assumptions of i.i.d. observations and the information of true distribution over labels.

IV. PARTIAL LIKELIHOOD FORMULATION AND THE INFORMATION-THEORETIC VIEW

We use a recent extension of maximum likelihood, the *partial likelihood* (PL) theory [3] to develop a general probabilistic framework for neural classifiers which is particularly suitable for application to problems in which time-ordering is essential (e.g. time-series problems), or can be conveniently defined. There are many cases where the process is generated according to a stochastic model that provides some correlation between samples at different times, and obviously most signal processing problems are of this nature.

Introduced by Cox in 1975 [3] as a generalization of the ideas of conditional and marginal likelihood, partial likelihood, has originated from the work on survival analysis, and aims at reduction of the parameter dimension by eliminating nuisance parameters. It is defined as a certain factorization of the full likelihood which is obtained by throwing away the part of the factorization that involves the nuisance parameters. Although this definition indicates some loss of information about the parameters of interest too, PL can yield estimators that are highly efficient [14]. However, this view of PL is quite limiting since such factorizations are very difficult to obtain. Therefore, we adopt an alternate definition of PL which has resulted from the counting process approach to survival analysis to develop a unified theory of neural classifiers based on the likelihood theory. In this definition of PL, time ordering is the key element since it depends on nested conditioning and does not require complete parametric specification, i.e., can also bypass the specification of any nuisance parameters. Hence, by this definition, PL bypasses major difficulties in the characterization of likelihood and can offer a suitable framework for design and analysis of neural classifiers.

Three aspects of maximum PL estimation are unique as compared to other extensions of maximum likelihood: (i) PL can easily be characterized for dependent observations, (ii) it can tolerate missing data, and (iii) in its characterization, it allows for sequential processing, hence is a suitable formulation for real-time applications. To write the PL, consider the same training set $\mathcal{T}$ defined before but note that now the ordering of the related input and output pairs $\{x_n, \mathbf{y}_n\}_{n=1}^N$ is important. Define $\mathcal{F}_k$ as the σ-field generated by the past x_i, $i \leq k-1$, and the outputs (past covariate information) $\mathbf{y}_i$, $i \leq k-1$. It can also include the current output value $\mathbf{y}_k$. Hence, $\mathcal{F}_k$ is a collection of all relevant events upto discrete (time) instant k, i.e., represents the history at k and $\mathcal{F}_{k-1} \subset \mathcal{F}_k$, i.e, $\mathcal{F}_k$ is an increasing sequence of σ-fields.

In a time series problem where the observation vector is defined as $\mathbf{y}_n = [y_n, y_{n-1}, \cdots, y_{n-L+1}]$ and a new sample is shifted in at each new time instant, the condition $\mathcal{F}_{n-1} \subset \mathcal{F}_n$ is easily satisfied. The filtration requirement on the sigma-fields can easily accommodate missing data problems.

The PL is written as the product

$$\mathcal{L}_N^p(\boldsymbol{\theta}) = \prod_{j=1}^{N} \prod_{i=1}^{m} f_i(\theta, \mathcal{F}_j)^{T_{i,j}}. \tag{8}$$

Note that to write the PL, we defined a new probability $P_\theta(C_i|\mathcal{F}_n)$ which is conditioned on all the past information available at the current instant n rather than the current output $\mathbf{y}_n$. Hence PL provides a formulation suitable for use of recurrent network probability models $f_i(\boldsymbol{\theta}, \mathcal{F}_j)$ and the PL theory can be used to study properties of recurrent networks as well.

With this new definition of probabilities, the ARE can be written as

$$\mathcal{I}_N(\theta) = \sum_{j=1}^{N} E\{\log \frac{P(C_i|\mathcal{F}_j)}{f_i(\boldsymbol{\theta}, \mathcal{F}_j)} | \mathcal{F}_j\} \tag{9}$$

We assume that for $\boldsymbol{\theta}_0$, $f(\cdot)$ defined in (1) achieves the true probability distribution and define $r_j(\boldsymbol{\theta}) \equiv \log \frac{P_{\theta_0}(C_i|\mathcal{F}_j)}{f_i(\boldsymbol{\theta}, \mathcal{F}_j)}$ which allows us to write the ARE as $\mathcal{I}_N(\theta) = \sum_{j=1}^{N} i_j(\boldsymbol{\theta})$ where $i_j(\boldsymbol{\theta}) = E\{r_j(\boldsymbol{\theta})|\mathcal{F}_j\}$ and define $\mathcal{J}_n(\boldsymbol{\theta}) = \sum_{k=1}^{N} j_k(\boldsymbol{\theta})$ where $j_k(\boldsymbol{\theta}) \equiv Var\{r_k(\boldsymbol{\theta})|\mathcal{F}_k\}$. The expectations in the above definitions are with respect to the true distribution $P_{\theta_0}(C_i|\mathcal{F}_j)$ $\forall \mathbf{a}_i \in \mathcal{S}$. Based on these definitions, we establish the relationship between PL maximization and ARE minimization for the general case of dependent observations by the following theorem:

Theorem: Given continuous functions $f_i(\cdot)\ \forall \mathbf{a}_i \in \mathcal{S}$, if, for each $\boldsymbol{\theta} \neq \boldsymbol{\theta}_0$, there exists a constant $\delta > 0$ such that, as $N \to \infty$,

$$P(\mathcal{I}_N(\boldsymbol{\theta})/N > \delta) \longrightarrow 1 \tag{10}$$

and

$$\mathcal{J}_N(\boldsymbol{\theta})/N^2 \longrightarrow 0 \ \text{ in probability} \tag{11}$$

then at least one $\arg\min_\theta \mathcal{I}_N(\boldsymbol{\theta})$ tends to one $\arg\max_\theta \bar{\mathcal{L}}_N^p(\boldsymbol{\theta})$ almost surely on $\Omega = \{\boldsymbol{\theta} \,|\, \mathcal{I}_N(\boldsymbol{\theta}) \uparrow \infty, \sum_{i=1}^N j_i(\boldsymbol{\theta})/\mathcal{I}_i^2(\boldsymbol{\theta}) < \infty\}$ where $\bar{\mathcal{L}}_N^p(\boldsymbol{\theta}) \equiv \ln \mathcal{L}_N^p(\boldsymbol{\theta})$.

Proof: Define $\mathcal{R}_N(\boldsymbol{\theta}) = \sum_{i=1}^N r_i(\boldsymbol{\theta})$. Lemma 2B in [14] states that, if the conditions (10) and (11) are satisfied, then, for each $\boldsymbol{\theta} \neq \boldsymbol{\theta}_0$, as $N \to \infty$, $(\mathcal{R}_N(\boldsymbol{\theta}) - \mathcal{I}_N(\boldsymbol{\theta}))/\mathcal{I}_N(\boldsymbol{\theta}) \longrightarrow 0$ almost surely on the set

$$\Omega = \left\{ \boldsymbol{\theta} | \mathcal{I}_N(\boldsymbol{\theta}) \uparrow \infty, \sum_{i=1}^N j_i(\boldsymbol{\theta})/\mathcal{I}_i^2(\boldsymbol{\theta}) < \infty \right\}. \tag{12}$$

Therefore, for any $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, and $\forall \epsilon > 0$, there exists an M such that,

$$\mathcal{I}_N(\boldsymbol{\theta})(1-\epsilon) < \mathcal{R}_N(\boldsymbol{\theta}) < \mathcal{I}_N(\boldsymbol{\theta})(1+\epsilon) \tag{13}$$

if $N > M$.

If we assume that $\mathcal{I}_N(\boldsymbol{\theta})$ achieves its minimum on $\Omega_{\mathcal{I}}^{(N)} \subset \Omega$ and $\mathcal{R}_N(\boldsymbol{\theta})$ on $\Omega_{\mathcal{R}}^{(N)} \subset \Omega$, then for $\boldsymbol{\theta}_N^* \in \Omega_{\mathcal{I}}^{(N)}$ and $\bar{\boldsymbol{\theta}}_N^* \in \Omega_{\mathcal{R}}^{(N)}$, we can write

$$\mathcal{I}_N(\boldsymbol{\theta}_N^*)(1-\epsilon) \leq \mathcal{I}_N(\bar{\boldsymbol{\theta}}_N^*)(1-\epsilon) < \mathcal{R}_N(\bar{\boldsymbol{\theta}}_N^*) \tag{14}$$

and

$$\mathcal{R}_N(\bar{\boldsymbol{\theta}}_N^*) \leq \mathcal{R}_N(\boldsymbol{\theta}_N^*) < \mathcal{I}_N(\boldsymbol{\theta}_N^*)(1+\epsilon), \tag{15}$$

or equivalently,

$$\mathcal{I}_N(\boldsymbol{\theta}_N^*)(1-\epsilon) < \mathcal{R}_N(\bar{\boldsymbol{\theta}}_N^*) < \mathcal{I}_N(\boldsymbol{\theta}_N^*)(1+\epsilon). \tag{16}$$

We can also write (13) for $\boldsymbol{\theta}_N^*$ as

$$\mathcal{I}_N(\boldsymbol{\theta}_N^*)(1-\epsilon) < \mathcal{R}_N(\boldsymbol{\theta}_N^*) < \mathcal{I}_N(\boldsymbol{\theta}_N^*)(1+\epsilon). \tag{17}$$

Hence, for sufficiently large N, we have

$$\mathcal{R}_N(\bar{\boldsymbol{\theta}}_N^*)/\mathcal{I}_N(\boldsymbol{\theta}_N^*) \longrightarrow 1 \tag{18}$$

by (16), and

$$\mathcal{R}_N(\boldsymbol{\theta}_N^*)/\mathcal{I}_N(\boldsymbol{\theta}_N^*) \longrightarrow 1 \tag{19}$$

by (17) almost surely on Ω. Hence, by (18) and (19), at least a point in $\Omega_{\mathcal{I}}^{(N)}$ tends to a point in $\Omega_{\mathcal{R}}^{(N)}$ almost surely.

Since we can express $\mathcal{R}_N(\boldsymbol{\theta}_N^*)$ in terms of the log PL $\bar{\mathcal{L}}_N^p(\boldsymbol{\theta}_N)$ as

$$\mathcal{R}_N(\boldsymbol{\theta}_N^*) = \bar{\mathcal{L}}_N^p(\boldsymbol{\theta}_0) - \bar{\mathcal{L}}_N^p(\boldsymbol{\theta}_N^*) \tag{20}$$

where the first term $\bar{\mathcal{L}}_N^p(\boldsymbol{\theta}_0)$ is constant, the conclusion of the theorem follows. □

The theorem establishes the equivalence of PL maximization and ARE minimization under two regularity conditions. The first condition of the theorem, (10), represents the rate by which the Kullback-Leibler information accumulates with N, and guarantees that for each $\boldsymbol{\theta} \neq \boldsymbol{\theta}_0$, $\mathcal{I}_N(\boldsymbol{\theta}) \to \infty$ as $N \to \infty$, i.e. the information continues to accumulate. The second condition, (11), on the other hand implies asymptotical stability of variance. The conditions can be shown to be satisfied for the perceptron probability model [1] and the FNM model [10] for the binary case. Using the softmax normalization representation given in (3), the result can easily be extended to the multi-class case by following a procedure similar to the one in [1] since softmax provides a convenient exponential formulation similar to the one presented in [1, 14].

PL, hence, provides a general probabilistic framework to pose classification problems for the cases of both *dependent* and *independent* observations. Since it allows characterization of the likelihood without any assumptions on the dependence structure of the data and includes all past history as part of its formulation, it makes a wider range of probability models available for use, such as the recurrent neural networks. A very important property of the PL is that its score function is a Martingale. We can use this property to establish the large sample properties of the PL estimator by generalizing the results given in [1] to the multi-class case. The asymptotic properties of likelihood allows one to incorporate network architecture selection into the design by using information-theoretic criteria [8, 11] and the selection of robust classifiers by defining modified likelihood functions [6, 8].

REFERENCES

[1] T. Adalı, X. Liu, and M. K. Sönmez, "Conditional distribution learning with neural networks and its application to channel equalization," *IEEE Trans. Signal Processing*, vol. 45, no. 4, pp. 1051-1064, Apr. 1997.

[2] J. S. Bridle, "Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition," *NATO ASI Series, vol. F68, Neurocomputing*, pp. 227-236.

[3] D.R. Cox, "Partial likelihood," *Biometrika*, vol. 62, pp. 69-72, 1975.

[4] I. Csiszár and G. Tusnády, "Information geometry and alternating minimization procedure," *Statistics and Decisions*, Supplementary issue, No. 1, (E. Dedewicz *et al.*, eds.), pp. 205-237, Munich, Oldenburg Verlag, 1984.

[5] H. Gish, "A probabilistic approach to the understanding and training of neural network classifiers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Albuquerque, NM, April 1990, pp. 1361-1364.

[6] M. Hintz-Madsen, M.W. Pedersen, L.K. Hansen, and J. Larsen, "Design and Evaluation of Neural Classifiers," in *Proc. IEEE Workshop on Neural Networks for Signal Processing VI*, S. Usui, Y. Tohkura, S. Katagiri and E. Wilson (eds.), Piscataway, New Jersey: IEEE, pp. 223-232, 1996.

[7] L. Kullback, and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics 22*, pp. 79-86, 1951.

[8] Jan Larsen, Lars Nonboe Andersen, Mads Hintz-Madsen and Lars Kai Hansen "Design of Robust Neural Network Classifiers," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Seattle, WA, May 1998, pp. 1205-1208.

[9] R. Rohwer and M. Morciniec, "The theoretical and experimental status of the n-tuple classifier," *Neural Networks*, vol. 11, no. 1, pp. 1-14, 1998.

[10] B. Wang, T. Adalı, X. Liu, and J. Xuan, "Partial likelihood for real-time signal processing using finite normal mixtures," *Proc. IEEE Workshop on Neural Networks for Signal Processing (NNSP)*, Cambridge, UK, Sep. 1998.

[11] Y. Wang, T. Adalı, S.-Y. Kung, and Z. Szabo, "Quantification and segmentation of brain tissue from MR images: A probabilistic neural network approach," to appear *IEEE Trans. Image Processing*, Special Issue on Applications of Neural Networks to Image Processing, August 1998.

[12] H. White, "Learning in artificial neural networks: A statistical perspective," *Neural Computation,* vol. 1, pp. 425-464, 1989.

[13] B. S. Wittner and J. S. Denker, "Strategies for teaching layered networks classification tasks," *Neural Info. Proc. Systems* (Denver, CO), 1988, pp. 850-859.

[14] W. H. Wong,"Theory of partial likelihood," *Ann. Statist.*, 14, pp. 88-123, 1986.