On Learning and Computational Complexity of FIR Radial Basis Function Networks, Part I: Learning of FIR RBFN's

Kayvan Najarian
Computer Science Department, University of North Carolina at Charlotte
9201 University City Blvd., Charlotte, NC 28223, U.S.A.
E-mail: knajaria@uncc.edu

Abstract— **Recently, the complexity control of dynamic neural models has gained significant attention from signal processing community. The performance of such a process depends highly on the applied definition of "model complexity", i.e. complexity models that give simpler networks with better model accuracy and reliability are preferred. The learning theory creates a framework to assess the learning properties of models. These properties include the required size of the training samples as well as the statistical confidence over the model. In this paper, we apply the learning properties of two families of FIR Radial Basis Function Networks (RBFN's) to introduce new complexity measures that reflect the learning properties of such neural model. Then, based on these complexity term, we define cost functions, which provide a balance between the training and testing performances of the model, and give desirable levels of accuracy and confidence.**

Keywords— **Radial Basis Function Networks, Computational Complexity, Computational Learning Theory, PAC Learning, Finite Impulse Response (FIR) Models**

I. Introduction

When linear models are unable to address the complexity of a signal (or a system), nonlinear models prove to be useful. Neural networks are known as powerful nonlinear tools in signal and image processing. The fact that neural networks can be easily and efficiently implemented using VLSI has made these models more appealing.

It is well-known that despite the popularity of neural networks, the difference between the training and testing behaviour of the neural networks has remained as the main concern in using such models. The major negative effect of an undesirable difference between the testing and training performances is a phenomenon known as "overfitting". Overfitting occurs when the trained model successfully fits the training data and fails to do so in the case of the testing data. It is well known that the difference between the testing and training performances of a model depends on how complex the model is. Therefore, in order to maintain a reasonable degree of the testing-training balance, one has to minimize the complexity of the model.

The Probably Approximately Correct (PAC) learning theory, proposed by Valiant [1], deals with the accuracy and confidence of the above-mentioned modeling task. PAC learning and other similar learning schemes allow quantitative evaluation of the learning properties of modeling procedures in which the data are independently and identically distributed (i.i.d.) in accordance to a probability measure P. PAC learning theory explores the difference between the testing and training performances of different types of function sets and evaluates the reliability of the developed models. Recently, new learning schemes are introduced that extend the results of the PAC learning to non-i.i.d. cases, and provide us with frameworks to assess learning properties of the dynamic modeling applications [2]. Since many dynamic systems can be efficiently modeled using "Nonlinear Finite Impulse Response" (Nonlinear FIR) models, the main focus of the present paper will be "neural FIR models".

The paper is organized as follows: Section II describes the basic definitions of the learning theory and is followed by Section III which gives the recent results on the learning properties of FIR modeling using two families of Radial Basis Function Networks (RBFN's). In Section IV, the results of the previous section is dis-

cussed and one possible application of the developed theory is introduced. Section V concludes the paper. The Part II of the paper [3], based on the results of the first part, new complexity measures for the foresaid family of neural models are presented and the corresponding cost functions to be minimized during the training procedure are built.

II. Basic Definitions

In this section, some of the basic concepts of statistical learning theory, including learning with m-dependency data, are reviewed. We start this section with the definition of m-dependent random variables (r.v.s):

Definition II.1: A set of r.v.s $\{Y_i\}_{i=1}^n$ is said to be m-dependent iff for all j and k, r.v.s Y_j *and* Y_k are independent if $|j-k| > m$. In other words, in a set of m-dependent r.v.s, the radius of dependency is limited to the integer m.

Now, suppose $X = [\alpha, \beta]^d \subset \mathcal{R}^d$ is an arbitrary set. Also suppose that $\mathcal{S}$ and P denote a σ-algebra of subsets of X and a probability measure on $(X, \mathcal{S})$, respectively. A function set $\mathcal{F}$ is defined as a set of measurable functions $f : X \to [-1/2, 1/2]$. There is nothing special about the interval [-1/2,1/2] and it can be replaced throughout by any bounded interval.

In a typical modeling task, an unknown function $f \in \mathcal{F}$ is to be estimated. In order to perform the estimation, a set of training data has to be generated as: $z_n = \{(x_i, f(x_i))\}_{i=1}^n$. Also, assume that each x_i's are m-dependent random variables identically distributed according to the probability measure P. An algorithm A, based on the training data z_n, generates a function $h \in \mathcal{F}$ as an approximator of f. At this point, we can define the concept of learning as follows:

Definition II.2: it Suppose that based on z_n, where $(x_1, \ldots, x_n)$ is a sequence of m-dependent r.v.s marginally-distributed according to the probability P. An approximation task is to be performed as described above. Then a function set $\mathcal{F}$ is said to be PAC learnable iff an algorithm "A" can be found based on which for any ϵ and δ, there exists "n" such that:

$$\sup_{f \in \mathcal{F}} \Pr\{d_P(f, h) \leq \epsilon\} \geq (1 - \delta) \qquad (1)$$

where $d_P(f, h)$ is a distance between f and h defined in terms of the probability distribution P, and Pr represents the probability of an outcome.

Another useful concept in function learning is an ϵ-cover of a function set which is defined as a set of functions $\{g_i\}_{i=1}^q$ in $\mathcal{F}$ such that for any function $f \in \mathcal{F}$, there is a g_j where: $d_P(f, g_j) < \epsilon$. An ϵ-cover with minimal size is called a minimal ϵ-cover, and its size is denoted as $N(\epsilon, \mathcal{F}, d_P)$. The generic algorithm used in the PAC learning theory is "empirical risk minimization algorithm" [4], which is used to compare learning with different function sets. A brief definition of this algorithm is given below:

Definition II.3: Let $\epsilon > 0$ be specified, and let $\{g_i\}_{i=1}^q$ be an $\epsilon/2$-cover (not necessarily minimal) of $\mathcal{F}$ with respect to d_P where d_P is defined above.

Then the empirical risk minimization algorithm is as follows: Draw a set of samples $(x_1, \ldots, x_n) \in X^n$, distributed in accordance with P. Define the cost functions: $\hat{J}_i = \frac{1}{n} \sum_{j=1}^n |f(x_j) - g_i(x_j)|$, $i = 1, \ldots, q$. Now the output of the algorithm is a function $h = g_l$ such that: $\hat{J}_l = \min_{1 \leq i \leq q} \hat{J}_i$.

III. Learning Properties of RBFN's

Since in many practical applications, the sequence of input data is generated based on a uniform distribution, in this paper we restrict ourselves to the modeling with uniformly-distributed m-dependent data. In such a training environment, it has been shown that the empirical risk minimization algorithm performed over different families of neural networks is learnable, and the upper bounds on the sample complexity (the minimum number of data points required for training) for such methods have been presented ([2] and [5]). An RBFN is defined as follows:

$$f(x) = \sum_{i=1}^l a_i \phi_i(r_i)$$

where: l is the number of neurons (basis functions), $a = (a_1, \ldots, a_l)$ forms the weight vector of the network with $|a_i| < M_a < \infty$ for all i, $\phi_i(.)$'s are the bounded differentiable radial basis functions in which $r_i = \|x - c_i\|$, and c_i is the center of the ith basis function.

Here, we review the available results on the learning of two families of RBFN's, i.e Gaussian and RMQ RBFN's. In a Gaussian RBFN:

$$\phi_i(r_i) = \exp(-b_i r_i^2) - \exp(-b_i \|c_i\|^2) \tag{2}$$

where $0 < b_i < \infty$ is the width (or scattering) parameter of the ith basis function. The second term normalizes each basis function and guarantees that $f(\mathbf{0}) = 0$. The second type of RBFN's to be considered here is the Reciprocal Multi-Quadratic RBFN's (RMQ-RBFN's). In this type of RBFN's, the basis functions are defined as:

$$\phi_i(r_i) = \frac{1}{\sqrt{1 + b_i r_i^2}} - \frac{1}{\sqrt{1 + b_i \|c_i\|^2}} \tag{3}$$

where $0 < b_i < \infty$ is the width (or scattering) parameter of the ith basis function. Similar to Gaussian functions, the second term normalizes each basis function so that $f(\mathbf{0}) = 0$.

For a RBDFN's network, the following theorem is proved in [6].

Theorem III.1: Consider the Gaussian and RMQ RBFN's introduced above and suppose that $\phi_i(r_i)$'s are given as (2). Forming b as:

$$b = (b_1,\ b_2,\ \ldots,\ b_l)$$

define:

$$A_{rbfn} = \sup_{a,b} \sum_{i=1}^{l} |a_i| \sqrt{b_i}\ .$$

Then, the empirical risk minimization algorithm with m-dependent data performed over a minimal $\epsilon/2$-cover results in the PAC learning with m-dependency. Moreover, in the case of Gaussian RBFN's, the sample complexity of the algorithm is given by:

$$n \geq \frac{8(m+1)}{\epsilon^2} \times \left\{ \left[\frac{2\sqrt{2} A_{rbfn} d(\beta - \alpha)}{\epsilon \sqrt{e}} \right]^d \ln 2 + \ln \frac{(m+1)}{\delta} \right\} \tag{4}$$

or equivalently:

$$\delta \geq 2^{\left[\frac{2\sqrt{2} A_{rbfn} d(\beta-\alpha)}{\epsilon\sqrt{e}} \right]^d} (m+1) \times \exp\left[-n\epsilon^2 / 8(m+1) \right]\ . \tag{5}$$

Similarly, in the case of RMQ-RBFN's, sample complexity is bounded by:

$$n \geq \frac{8(m+1)}{\epsilon^2} \times \left\{ \left[\frac{4 A_{rbfn} d(\beta - \alpha)}{3\sqrt{3}\ \epsilon} \right]^d \ln 2 + \ln \frac{(m+1)}{\delta} \right\} \tag{6}$$

or equivalently:

$$\delta \geq 2^{\left[\frac{4 A_{rbfn} d(\beta-\alpha)}{3\sqrt{3}\ \epsilon} \right]^d} (m+1) \times \exp\left[-n\epsilon^2 / 8(m+1) \right]\ . \tag{7}$$

Theorem III.1 provides a framework for learning of neural models using Gaussian and RMQ-RBFN's.

In the simplest form of modeling, the information regarding the structure of the network (such as the number of neurons and the size of the parameter space) is known, and the objective is to use a set of input-output samples to find the optimal values for the network's parameters. In other words, a modeling task based on the empirical risk minimization works only when the structure of the function set is known beforehand. In many practical applications, however, the exact prior information regarding the structure of the unknown network f (e.g. the number of neurons l) is not available. In such cases, a more sophisticated method should be used to provide us with not only the optimal set of parameters, but also the minimal structural complexity. One such method, introduced by Vapnik in [7], is known as "structural risk minimization method". Applying this procedure to RBFN's, one can search for the number of required neurons as well as the best set of parameters. In other words, the structural risk minimization uses the simplest function that provides learning with the pre-specified values of the accuracy, the confidence and the training data size.

IV. Discussion

The results given above evaluate the learning properties of two families of RBFN's. Using these bounds, the accuracy and confidence of the resulting models can be guaranteed. This in turn guarantees that unlike many practical use of neural networks, the neural models obtained with this learning-based algorithm avoid overfitting the data. The fact that overfitting is avoided encourages the use of neural networks in more sensitive applications (such as remote sensing and medical diagnostics, as described below), where the reliability of the model is most important.

One specific application of the developed theory is biomedical modeling and signal processing. Modeling and identification of biomedical signals and systems are highly sensitive processes in which based on the developed models, the biomedical signals or systems are diagnosed. Any incorrect prediction or diagnosis can result to great harms to the patient's health. Due to the sensitive nature of this matter, if physicians are to rely on the prediction or diagnosis made by an algorithm, they would like to have a quantitative guarantee (even with high probability) over the accuracy of the algorithms (models). As mentioned above, neural networks can be easily overfitted and this might result to neural models that are not reliable enough to be used for biomedical applications. Considering the fact that for many biomedical signals and systems there exist huge data repositories, if neural networks are trained with data sets that satisfy the (conservative) learning inequalities, they can create accurate models that give quantitative guarantees to physicians. Again, the conservative nature of the bounds developed here implies that the neural models can be trained by smaller training sets; however, since the huge data sets are available and the desirable processing capabilities of neural networks are required in biomedical applications, the developed bounds can still form reliable neural models of biomedical systems.

The conservative bounds created above can also be used to generate learning-based complexity measures that attempt to avoid the use of overcomplex models, which will be discussed in Part II of the paper [3]. By including the learning-based complexity term to the cost functions to be minimized during the optimization, one can limit the complexity of the model during the optimization phase.

V. Conclusions

The learning properties of FIR RBFN's are evaluated. These properties include the number of training data points that guarantee pre-specified values of accuracy and confidence. These results will be used in the second part of the paper to develop complexity terms that create a balance between the training and testing performance of the models and avoid overfitting.

References

[1] L.G. Valiant, "A theory of learnable," *Comm. ACM*, pp. pp. 1134–1142, 1984.

[2] K. Najarian, Guy A. Dumont, and Michael S. Davies, "PAC learning in Nonlinear FIR Models," *submitted to: Journal of Adaptive Control and Signal Processing*, To appear.

[3] K. Najarian, "On Learning and Computational Complexity of FIR Radial Basis Function Networks, Part II: Complexity Measures," *Submitted to ICASSP'2001*, May 2001.

[4] M. Vidyasagar, *A Theory of Learning and Generalization*, Springer, 1997.

[5] K. Najarian, G.A. Dumont and M.S. Davies, "A learning-theory-based training algorithm for variable-structure dynamic neural modeling," *Proc. Inter. Joint Conf. Neural Networks (IJCNN99)*, 1999.

[6] K. Najarian, *Appliation of learning theory in neural modeling of dynamic systems*, Ph.D. thesis, Dpartment of Electrical and Computer Engineering, University of British Columbia, 2000.

[7] V.N. Vapnik and A.Y. Chervonenkis, "On the uniform convergence of relative frequencies of events to their probabilities," *Theory of Probability and its Applications*, vol. 16, no. 2, pp. 264–280, 1971.