Available online at www.sciencedirect.com
SCIENCE DIRECT®

Neural Networks 18 (2005) 951–957

Neural Networks

www.elsevier.com/locate/neunet

# Sensitivity analysis applied to the construction of radial basis function networks

D. Shi[a,*], D.S. Yeung[b], J. Gao[c]

[a]School of Computer Engineering, Nanyang Technological University, Singapore 639798, Singapore
[b]Department of Computing, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
[c]School of Mathematics, Statistics and Computer Science, University of New England, Armidale, NSW, Australia

Received 29 March 2004; revised 25 February 2005; accepted 25 February 2005

**Abstract**

Conventionally, a radial basis function (RBF) network is constructed by obtaining cluster centers of basis function by maximum likelihood learning. This paper proposes a novel learning algorithm for the construction of radial basis function using sensitivity analysis. In training, the number of hidden neurons and the centers of their radial basis functions are determined by the maximization of the output's sensitivity to the training data. In classification, the minimal number of such hidden neurons with the maximal sensitivity will be the most generalizable to unknown data. Our experimental results show that our proposed sensitivity-based RBF classifier outperforms the conventional RBFs and is as accurate as support vector machine (SVM). Hence, sensitivity analysis is expected to be a new alternative way to the construction of RBF networks.


*Keywords:* Sensitivity analysis; Radial basis function neural network; Orthogonal least square learning; Network pruning

## 1. Introduction

As one of the most popular neural network models, radial basis function (RBF) network attracts lots of attentions on the improvement of its approximate ability as well as the construction of its architecture. Bishop (1991) concluded that an RBF network can provide a fast, linear algorithm capable of representing complex non-linear mappings. Park and Sandberg (1993) further showed that RBF network can approximate any regular function. In a statistical sense, the approximate ability is a special case of statistical consistency. Hence, Xu, Krzyzak, and Yuille (1994) presented upper bounds for the convergence rates of the approximation error of RBF networks, and proved constructively the existence of a consistent estimator point-wise and $L_2$ convergence rates of the best consistent estimator for RBF networks. Their results can be a guide to optimize the construction of an RBF network, which includes the determination of the total number of radial basis functions along with their centers and widths.

There are three ways to construct an RBF network, namely, clustering, pruning and critical vector learning. Bishop (1991) and Xu (1998) follow the clustering method, in which the training examples are grouped and then each neuron is assigned to a cluster. The pruning method, such as Chen, Crown, and Grant (1991) and Mao (2002), creates a neuron for each training example and then to prune the hidden neurons by example selection. The critical vector learning method, exemplified by Scholkopf, Sung, Burges, Girosi, Niyogi, and Poggio (1997) constructs an RBF with the critical vectors, rather than cluster centers.

Moody and Darken (1989) located optimal set of centers using both the k-means clustering algorithm and learning vector quantization. The drawback of this method is that it considers only the distribution of the training inputs, yet the output values influence the positioning of the centers. Bishop (1991) introduced the Expectation–Maximization (EM) algorithm to optimize the cluster centers with two steps: obtaining initial centers by clustering and optimization of the basis functions by applying the EM algorithm. Such a treatment actually

* Corresponding author. Tel.: +65 6790 6245; fax: +65 6792 6559.
*E-mail address:* asdmshi@ntu.edu.sg (D. Shi).

0893-6080/$ - see front matter 
doi:10.1016/j.neunet.2005.02.006

does not perform a maximum likelihood learning but a suboptimal approximation. Xu (1998) extended the model for mixture of experts to estimate basis functions, output neurons and the number of basis functions all together. The maximum likelihood learning and regularization mechanism can be further unified to his established Bayesian Ying Yang (BYY) learning framework (Xu, 2004a–c), in which any problem can be decomposed into Ying space or invisible domain (e.g., the hidden neurons in RBFs), and Yang space or visible domain (e.g., the training examples in RBFs), and the invisible/unknown parameters can be estimated through harmony learning between these two domains.

Chen et al. (1991) proposed orthogonal least square (OLS) learning to determine the optimal centers. The OLS combines the orthogonal transform with the forward regression procedure to select model terms from a large candidate term set. The advantage of employing orthogonal transform is that the responses of the hidden layer neurons are decorrelated so that the contribution of individual candidate neurons to the approximation error reduction can be evaluated independently. However, the original OLS learning algorithm lacks generalization and global optimization abilities. Mao (2002) employed OLS to decouple the correlations among the responses of the hidden units so that the class separability provided by individual RBF neurons can be evaluated independently. This method can select a parsimonious network architecture as well as centers providing large class separation.

The common feature of all the above methods is that the radial basis function centers are a set of the optimal cluster centers of the training examples. Schokopf et al. (1997) calculated support vectors using a support vector machine (SVM), and then used these support vectors as radial basis function centers. Their experimental results showed that the support-vector-based RBF outperforms conventional RBFs. Although the motivation of these researchers was to demonstrate the superior performance of a full support vector machine over either conventional or support-vector-based RBFs, their idea of critical vector learning is worth borrowing.

This paper proposes a novel approach to determining the centers of RBF networks based on sensitivity analysis. The remainder of this paper is organized as follows: In Section 2, we describe the concepts of sensitivity analysis. In Section 3, the most critical vectors are obtained by OLS in terms of sensitivity analysis. Section 4 contains our experiments and Section 5 offers our conclusions.

## 2. Sensitivity analysis on neural networks

Sensitivity is initially investigated for the construction of a network prior to its design, since problems (such as weight perturbation, which is caused by machine imprecision and noisy input) significantly affect network training and generalization (Widrow, 1960). Stevenson, Winter, and Widrow (1990) established sensitivity analysis to weight error and derive an analytical expression for the probability of error in Madaline. Typically, one can simulate hardware imprecision by introducing perturbation on weight and input to measure the sensitivity. Zurada, Malinowski, and Usui (1997) extended this idea of sensitivity analysis to network pruning.

There are two different methods to measure sensitivity, one is noise-to-signal ratio, the other is expectation of output error. Sensitivity analysis is conducted by measuring the response of the network when parameter perturbations are introduced intentionally.

Treating all network inputs, weights, input perturbations, and weight perturbations as random variables, Piche (1995) defined sensitivity as the noise-to-signal-ratio (NSR) of the output layer:

$$\mathrm{NSR} = \frac{\sigma^2_{\Delta y}}{\sigma^2_y} = \frac{4}{\pi}\sqrt{\frac{\sigma^2_{\Delta x}}{\sigma^2_x} + \frac{\sigma^2_{\Delta w}}{\sigma^2_w}}, \tag{1}$$

where $\sigma^2_y$, $\sigma^2_x$, $\sigma^2_w$, $\sigma^2_{\Delta y}$, $\sigma^2_{\Delta x}$ and $\sigma^2_{\Delta w}$ refer to the variances of output $y$, inputs $x$, weights w, output error $\Delta y$, input perturbation $\Delta x$ and weight perturbation $\Delta w$, respectively. Piche's stochastic model is not generally valid because: (1) All neurons in the same layer are assumed to have the same activation function, but this is not the case in some network models. (2) To satisfy the central limit theorem, the number of neurons in hidden layers is assumed to be large. (3) Weight perturbations are assumed to be very small, but this would be too restrictive for network training. To address these problems, Yeung and Sun (2002) generalized Piche (1995)'s work in two significant ways: (1) No restriction on input and output perturbation, which widens the application areas of sensitivity analysis; (2) The commonly used activation functions are approximated by a general function expression whose coefficient will be involved in the sensitivity analysis. This treatment provides a way to sensitivity analysis on activation functions.

Zeng and Yeung (2001, 2003) proposed a quantified measure and its computation for the sensitivity of the MLP to its input perturbation. The sensitivity $s^l_i$ of a single neuron $i$ in layer $l$ is defined as the mathematical expectation of the absolute value of its output deviation caused by the perturbation $\Delta X^l$:

$$s^l_i = E\left(\left|f\left((X^l + \overline{\Delta X^l})\, W^l_i\right) - f(X^l \times W^l_i)\right|\right) \tag{2}$$

A bottom-up approach was adopted. After the sensitivities of single neurons are calculated, the sensitivity of the entire MLP network will be computed. Some applications of the MLP, such as improving error tolerance, measuring generalization ability, and pruning the network architecture, would benefit from their theoretical study. However, this