Available online at www.sciencedirect.com
ScienceDirect
Neurocomputing 71 (2008) 1345–1358
NEUROCOMPUTING
www.elsevier.com/locate/neucom

A sequential multi-category classifier using radial basis function networks

S. Suresh*, N. Sundararajan, P. Saratchandran

School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore

Received 31 August 2006; received in revised form 30 May 2007; accepted 10 June 2007
Communicated by K. Li
Available online 6 July 2007

Abstract

This paper presents a new sequential multi-category classifier using radial basis function (SMC-RBF) network for real-world classification problems. The classification algorithm processes the training data one by one and builds the RBF network starting with zero hidden neuron. The growth criterion uses the misclassification error, the approximation error to the true decision boundary and a distance measure between the current sample and the nearest neuron belonging to the same class. SMC-RBF uses the hinge loss function (instead of the mean square loss function) for a more accurate estimate of the posterior probability. For network parameter updates, a decoupled extended Kalman filter is used to reduce the computational overhead. Performance of the proposed algorithm is evaluated using three benchmark problems, viz., image segmentation, vehicle and glass from the UCI machine learning repository. In addition, performance comparison has also been done on two real-world problems in the areas of remote sensing and bio-informatics. The performance of the proposed SMC-RBF classifier is also compared with the other RBF sequential learning algorithms like MRAN, GAP-RBFN, OS-ELM and the well-known batch classification algorithm SVM. The results indicate that SMC-RBF produces a higher classification accuracy with a more compact network. Also, the study indicates that using a function approximation algorithm for classification problems may not work well when the classes are not well separated and the training data is not uniformly distributed among the classes.

Keywords: RBF network; Hinge loss function; Sequential learning; Multi-category classification; Decoupled extended Kalman filter

1. Introduction

In many real-world classification problems, such as medical diagnosis applications, the complete set of training data are not available at the same time; rather they become available only over a period of time. For example, in a cancer detection problem involving micro-array gene expression data, the tumor marker records are available initially only from a small number of patients. However, the size of the database increases with more tumor patients over a period of time. Also, tumor marker labels may change after clinical surgeries requiring modifications in the existing database. Both the addition of new samples and changes of class labels in the existing database normally requires retraining the designed classifiers. This retraining process requires a large amount of computational time and memory. A learning scheme that handles these sequential data without retraining on the entire data will be quite useful for various applications. In this paper, we present such a scheme for multi-category classification problems.

Among various classification methods, neural network based classifiers have been successfully used in a number of applications such as pattern recognition [6], remote sensing [11] and in biomedical areas [8]. Neural networks are able to construct complex classification boundaries [12] without any prior assumption on the input data statistics. Radial basis function networks (RBFN) have been widely used in all these applications due to its approximation ability as well as its simple architecture.

RBFN is a three layered feed forward network with a radially symmetric Gaussian function as an activation

*Corresponding author.
E-mail address: ssundaram@ntu.edu.sg (S. Suresh).

0925-2312/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2007.06.003

function in the hidden layer [19]. Let $\mathbf{X}$ be an n-dimensional input pattern vector ($\mathbf{X} = [x_1, x_2, \ldots, x_n]$). The RBFN with m outputs and K Gaussian neurons is given by

$$\hat{y}_i = \sum_{j=1}^{K} w_{ij} \exp\left(-\frac{\|\mathbf{X} - \boldsymbol{\mu}_j\|}{2\sigma_j^2}\right),$$
$$\boldsymbol{\mu}_j \in \Re^n, \ \sigma_j^2 \in \Re^+, \ w_{ij} \in \Re, \ i = 1, 2, \ldots, m, \quad (1)$$

where y_i is the ith output, $\boldsymbol{\mu}_j$ and σ_j are the center and width of the Gaussian basis function of the jth neuron, respectively, and w_{ij} is the interconnection weight between the ith output neuron and the jth Gaussian neuron. $\Re^+$ indicates the set of all positive real numbers.

In classical batch learning approach for RBFN (i.e., repeated presentation of training samples), the number of hidden neurons are fixed a priori. Usually, the centers and widths of the Gaussian neurons are calculated using the K-means clustering method and the weights are estimated using the least square method [16,17]. The generalization ability [18] and compactness [1,3,13] of RBFNs have been addressed in batch scenarios by many researchers. In the sequential learning scenario, the RBFN parameters are updated sequentially based on the error computed using the new training sample [7,10,15,26]. Since, the sequential learning algorithm does not require retraining of network whenever new data arrives, it is preferred over classical batch algorithms for many practical applications. In a sequential learning algorithm, the training samples are presented one by one such that the algorithm does not require prior information about the total number of training samples. Hence, the sequential learning scheme requires less computational effort and minimal storage space over the batch learning scheme [10,15,19,25].

Recently Huang et al. [7] have developed a sequential algorithm called growing and pruning RBFN (GAP-RBFN) and have presented its better performance on a number of function approximation and classification problems. The GAP-RBFN algorithm introduced the concept of neuron significance for growing and pruning of the hidden neurons and does not require fixing the number of neurons a priori.

In GAP-RBFN algorithm [7], the criterion for growing, updating the parameters and pruning are given by

- *Growing criterion*: If $\|\mathbf{X}_n - \boldsymbol{\mu}_{nr}\| > \varepsilon_n$ and $E_{\text{sig}}(nr) > e_{\min}$ then add a new neuron to the network. Here, nr is the nearest neighbor neuron to the new training sample (i.e., neuron closest to sample $\mathbf{X}_n$) and E_{sig} is the significance of nrth neuron defined as its average output over all the input samples it has received so far [7]. The control parameters $e_{\min}$ and ε_n are thresholds chosen a priori by trial and error.
- *Parameter update*: If the growth criterion is not satisfied then adjust the network parameters only for the nearest neighbor neuron (nr) using an extended Kalman filter (EKF) algorithm [7].
- *Pruning criterion*: If the significance of the nearest neighbor neuron is less than the expected accuracy ($E_{\text{sig}}(nr) < e_{\min}$) then the nrth hidden neuron can be removed from the network.

In GAP-RBFN, the criteria for growing, pruning and updating parameters are carried out for the nearest (to the current input sample) neuron. This is similar to the other sequential algorithm known as extended minimal resource allocation network (EMRAN) [25].

The performance of MRAN and GAP-RBFN algorithms have been evaluated [7,15,26] using various benchmark function approximation and classification problems. For function approximation problems, the generalization performance of these sequential algorithms has been shown to be better than the batch algorithms. For classification problems, the generalization performance depends heavily on the optimal selection of control parameters of the learning algorithms and the input data distribution as they use the basic function approximation approach for classification. Hence, sequential learning algorithms available in the literature may not work well for classification problems as they use criteria based on only concepts of function approximation and not classification.

To show this clearly, a motivational example is presented below to indicate how the growing/pruning criteria developed in GAP-RBFN, which is good for function approximation problems, is inadequate for classification problems. The motivational example also shows that one has to consider the Euclidian distance measure within the class and the misclassification error in the criteria for better performance.

1.1. Motivational example

Let us consider a simple two class problem as shown in Fig. 1(a). Here, each input pattern has two input features, namely x_1 and x_2. If $x_1 < 0.5$ and $x_2 < 0.5$ then the sample is assigned to class C_1. Similarly, if $x_1 > 0.5$ and $x_2 > 0.5$ then the sample is assigned to class C_2. The class label C_1 and C_2 are coded as 1 and -1, respectively. The training samples are generated randomly. Next, we use the GAP-RBFN learning algorithm to illustrate our point that the criteria appropriate for function approximation are inadequate for classification.

The objective here is to approximate the ideal decision surface shown in Fig. 1(b) with an approximation error ($e_{\min} = 1.0 \times 10^{-3}$). It should be noted here that in this figure and subsequent figures the scale for x_2 reads from right (0) to left (1) to make the visualization of the decision surface clear.

The threshold for distance growing criterion ε_n is fixed at 0.2. The training samples are presented sequentially to the GAP-RBFN classifier. The first six samples presented to the network satisfies the growing criterion. Hence, six hidden neurons are added to the network. The centers of the hidden neurons are nothing but the actual sample