Available online at www.sciencedirect.com
SciVerse ScienceDirect
Mathematics and Computers in Simulation 86 (2012) 2–9
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original article

Generalized recurrent neural network for ϵ-insensitive support vector regression

Yan Zhao [a], Qingshan Liu [b,*]

[a] Department of Basic Courses, Wannan Medical College, Wuhu 241000, China
[b] School of Automation, Southeast University, Nanjing 210096, China

Received 24 July 2010; received in revised form 17 March 2011; accepted 9 March 2012
Available online 23 August 2012

Abstract

In this paper, a generalized recurrent neural network is proposed for solving ϵ-insensitive support vector regression (ϵ-ISVR). The ϵ-ISVR is first formulated as a convex non-smooth programming problem, and then a generalize recurrent neural network with lower model complexity is designed for training the support vector machine. Furthermore, simulation results are given to demonstrate the effectiveness and performance of the proposed neural network.

Keywords: Non-smooth optimization; ϵ-Insensitive support vector regression; Generalized recurrent neural network; Global convergence

1. Introduction

Support vector machines (SVMs) are powerful tools for data classification and regression. In the recent years, many fast algorithms for SVMs have been developed [2]. Mangasarian [14] proposed the finite Newton algorithm for SVMs learning. Keerthi and DeCoste [5] introduced the modified finite Newton algorithm to speed up the finite Newton algorithm for fast solution of large scale linear SVMs.

More recently, as a software and hardware implementable approach, recurrent neural networks for solving linear and nonlinear optimization problems with their engineering applications have been widely developed [6,9,13,15,17]. Compared with traditional numerical optimization algorithms, the neural networks have fast convergence rate in real-time solutions. In 1986, Tank and Hopfield [15] proposed a recurrent neural network for solving the linear programming problems for the first time. In 1988, the dynamical canonical nonlinear programming circuit (NPC) was introduced by Kennedy and Chua [6] for optimization by utilizing a finite penalty parameter, which can generate the approximate optimal solutions. Wang and Xia [17] proposed a primal-dual neural network for solving the linear assignment problems. To get the optimal solutions of non-smooth optimization problems, Forti et al. [4] proposed and investigated the generalized NPC (G-NPC), which can be considered as a natural extension of NPC. In order to reduce the model complexity, some one-layer recurrent neural networks with lower model complexity have been constructed for solving linear and nonlinear programming problems [10,12]. This paper is concerned with a generalized recurrent neural network for the ϵ-insensitive support vector regression. The global convergence of the proposed recurrent neural

* Corresponding author.
E-mail address: qsliu@seu.edu.cn (Q. Liu).

0378-4754/$36.00
http://dx.doi.org/10.1016/j.matcom.2012.03.013

network is guaranteed using the Lyapunov-like method. Compared with the existing neural networks for support vector regression (SVR) learning, the proposed neural network herein has lower model complexity, but is efficient for SVR learning.

2. ϵ-Insensitive support vector regression (ϵ-ISVR)

Consider a given training data set $\mathcal{T} = \{(x_i, y_i) : x_i \in \mathbb{R}^n, y_i \in \mathbb{R}, i = 1, 2, \ldots, m\}$, where x_i is the input data and y_i is called the observation. we would like to find a linear or nonlinear regression function, $f(x)$, tolerating a small error in fitting the given data set. This can be achieved by utilizing the ϵ-insensitive loss function that sets an ϵ-insensitive "tube" around the data, within which errors are discarded. Disregarding the tiny errors that fall within some tolerance, say ϵ, that may lead to a better generalization ability by utilizing an ϵ-insensitive loss function. Also, applying the idea of SVMs [2,8,16], the function $f(x)$ is made as flat as possible in fitting the training data set by utilizing the ϵ-insensitive loss function.

We start with the case of linear function $f(x) = x^T w + b$, where $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$. To obtain a linear predictor, SVR solves the following convex optimization problem:

$$\begin{aligned} \min \quad & \frac{1}{2} w^T w + c \sum_{i=1}^{m} (\xi_i + \bar{\xi}_i), \\ \text{s.t.} \quad & w^T x_i + b - y_i \le \epsilon + \xi_i, \quad y_i - w^T x_i - b \le \epsilon + \bar{\xi}_i, \\ & \xi_i, \bar{\xi}_i \ge 0, \quad i = 1, 2, \ldots, m, \end{aligned} \tag{1}$$

where $c > 0$ determines the trade-off between the flatness of f and the amount up to which deviations larger than ϵ are tolerated.

Eliminating the slack variables ξ_i and $\bar{\xi}_i$, we get the following unconstrained non-smooth optimization problem:

$$\min \quad L_\epsilon(w, b) = \frac{1}{2}(w^T w + b^2) + c \sum_{i=1}^{m} |w^T x_i + b - y_i|_\epsilon, \tag{2}$$

where $|w^T x_i + b - y_i|_\epsilon = \max\{0, |w^T x_i + b - y_i| - \epsilon\}$ that represents the fitting errors.

Nonlinear SVR can be obtained by using a kernel function and an associated reproducing kernel Hilbert space H. According to the representor theory [7], the optimal function can be expressed as a linear combination of the kernel functions centered in the training samples,

$$f(x) = \sum_{i=1}^{m} \alpha_i K(x, x_i) + \beta. \tag{3}$$

The nonlinear support vector regression formulation can be written as follows:

$$\min \quad L_\epsilon(\alpha, \beta) = \frac{1}{2} \left(\sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j K(x_i, x_j) + \beta^2 \right) + c \sum_{i=1}^{m} \left| \sum_{j=1}^{m} \alpha_j K(x_i, x_j) + \beta - y_i \right|_\epsilon . \tag{4}$$

Assume the kernel matrix $K = \{K(x_i, x_j)\}_{m \times m}$ and K_i being the ith row of K, then (4) is rewritten as

$$\min \quad L_\epsilon(\alpha, \beta) = \frac{1}{2} \left(\alpha^T K \alpha + \beta^2 \right) + c \sum_{i=1}^{m} |K_i \alpha + \beta - y_i|_\epsilon . \tag{5}$$

Without loss of generality, we consider only the nonlinear SVR in this paper and the linear SVR as a special case.