Applied Soft Computing 13 (2013) 3473–3481

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Support vector regression based on optimal training subset and adaptive particle swarm optimization algorithm

JinXing Che*

Department of Science, NanChang Institute of Technology, NanChang, JiangXi 330099, China

ARTICLE INFO

Article history:
Received 13 January 2012
Received in revised form 19 January 2013
Accepted 11 April 2013
Available online 23 April 2013

Keywords:
Support vector regression
Adaptive particle swarm optimization
Optimal training subset
Parameters selection

ABSTRACT

Support vector regression (SVR) has become very promising and popular in the field of machine learning due to its attractive features and profound empirical performance for small sample, nonlinearity and high dimensional data application. However, most existing support vector regression learning algorithms are limited to the parameters selection and slow learning for large sample. This paper considers an adaptive particle swarm optimization (APSO) algorithm for the parameters selection of support vector regression model. In order to accelerate its training process while keeping high accurate forecasting in each parameters selection step of APSO iteration, an optimal training subset (OTS) method is carried out to choose the representation data points of the full training data set. Furthermore, the optimal parameters setting of SVR and the optimal size of OTS are studied preliminary. Experimental results of an UCI data set and electric load forecasting in New South Wales show that the proposed model is effective and produces better generalization performance.

1. Introduction

Recently, support vector regression (SVR), which was developed by Vapnik [20], has become very promising and popular in the field of machine learning due to its attractive features and profound empirical performance for small sample, nonlinearity and high dimensional data application [1–3]. The main technique of SVR is to use the principle of structural risk minimization (SRM) by constructing an optimal regression hyper-plane in the hidden feature space and solving the unique solution of the accordingly dual quadratic programming problem.

In the SVR, the model for forecasting is generated from the learning process with the training data set. Then, SVR has been successfully applied to solve forecasting problems in many areas, such as financial time series forecasting [4], short term wind speed prediction [5], face recognition [6], electric load prediction [7], and so on. However, these empirical results indicated that the largest problems encountered in building up the SVR are how to select the three parameters (C, ε, and δ^2) and improve the slow learning for large sample. To solve the above problems, many researchers have given some parameter setting algorithms [8–11]. Particle swarm optimization has become a popular parameters selection algorithm [12], Lin et al. utilize particle swarm optimization (PSO) for parameter determination and feature selection of support vector machines (SVM) [13], Huang and Dun present an optimization mechanism that hybridized PSO and SVM to improve the classification accuracy with an appropriate feature subset and SVM's parameters [14]. Considering that the computation complexity is $O(K \times N^2)$ (K is the number of iteration), parameter setting algorithms will lead to slow learning in large-scale training data set. This paper aims to present a model for solving the above problem.

In the large sample learning problem, the training data set contain much redundant information generally. The redundant data not only are useless for SVR learning, but also could lead to low computational efficiency and low accuracy potentially. Thus, discarding the redundant information of training data set can accelerate learning process of SVR's parameters selection. Inspired by that not all of these training data are equally important for a specific forecasting problem, only the support vectors determine the final SVR model. Better computation performance and generalization ability can be achieved by choosing the optimal training subset (OTS) containing support vectors. Therefore, the learning process can be fast and accurate by using the APSO algorithm and OTS selection method.

By combining the optimal training subset reconstruction method with APSO, here the author presents a new parameter selection algorithm for SVR, called *APSO-OTS-SVR*. Some improved techniques in the optimization framework are presented in order to simplify the APSO iteration learning algorithm and increase the learning speed. Based on *APSO-OTS-SVR*, forecasting models for an UCI data set and electric load forecasting in New South Wales are proposed. Compared with three SVR models, the experimental

* Tel.: +86 15970427685; fax: +86 15970427685.
E-mail address: jinxingche@163.com

1568-4946/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2013.04.003

results show that *APSO-OTS-SVR* provides a parameters selection and better generalization performance at higher learning speed.

The rest of the study is organized as follows. Section 2 proposes *APSO-OTS-SVR*, and the main steps of it are also given in this section. The experiment design of the forecasting model is given in Section 3. Section 4 presents the experimental results. The final conclusion is drawn in Section 5.

2. The explicit process of the new algorithm

2.1. Support vector regression

This subsection briefly introduces the brief ideas of SVMs for the case of regression. The SVR firstly considers the linear regression problem, then it can be extended to the nonlinear regression utilizing kernel functions technique. And we refer the reader to the excellent surveys for a more thorough coverage of it [15–19].

Suppose the training data are $(x_1, y_1), \ldots, (x_n, y_n) \subset \mathcal{W} \times \mathbb{R}$, where $\mathcal{W}$ denotes the space of the input patterns x_i (e.g. $\mathcal{W} = \mathbb{R}^n$), and y_i is the associated output values of x_i. For linear regression problem, ε-SVR [20] produce a regression function $f(x) = \omega^T x + b$ by solving the following formulation.

$$\min_{\omega,b,\xi,\xi^*} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{1}$$

$$s.t. \begin{cases} y_i - (\omega^T x_i + b) \le \varepsilon + \xi_i \\ (\omega^T x_i + b) - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \tag{2}$$

The parameter $C > 0$ decides the trade-off of the above two terms in Eq. (1), ξ_i denotes the training error above ε, whereas ξ_i^* denotes the training error below $-\varepsilon$, and n represents the number of samples.

The first term of Eq. (1) controls the generalization ability of the regression function. The second term penalizes the learning error of $f(x_i)$ and y_i by using ε-insensitive tube $|y_i - (\langle\omega, x_i\rangle + b)| \le \varepsilon$. This ε-insensitive loss function $|\xi_\varepsilon|$ can be described as the following.

$$|\xi|_\varepsilon := \begin{cases} 0, \text{if } |\xi| < \varepsilon \\ |\xi| - \varepsilon, \text{otherwise} \end{cases} \tag{3}$$

The primal form of Lagrange multiplier method is as following:

$$L_p = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) - \sum_{i=1}^{n}\alpha_i(\varepsilon + \xi_i - (y_i - (\omega^T x_i + b))) - \sum_{i=1}^{n} r_i\xi_i - \sum_{i=1}^{n}\alpha_i^*(\varepsilon + \xi_i^* - ((\omega^T x_i + b) - y_i)) - \sum_{i=1}^{n} r_i^*\xi_i^* \tag{4}$$

where $\alpha_i, \alpha_i^* \ge 0$ and $r_i, r_i^* \ge 0$ are the well-known Lagrange multipliers, they are obtained by maximizing the dual function of Eq. (4). And according to the partial derivatives of ξ_i and ξ_i^* in Eq. (4), one gets:

$$\begin{aligned} &\omega + \sum_{i=1}^{n} x_i(\alpha_i - \alpha_i^*) = 0 \\ &\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \\ &C - \alpha_i - r_i = 0 \\ &C - \alpha_i^* - r_i^* = 0 \end{aligned} \tag{5}$$

This will lead to the following dual problem by replacing ω of Eq. (4).

$$\max_{\alpha_i,\alpha_i^*} \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) - \varepsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)x_i^T x_j \tag{6}$$

subject to

$$\begin{cases} \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \\ 0 \le \alpha_i \le C \\ 0 \le \alpha_i^* \le C \\ i = 1, 2, \ldots, n \end{cases} \tag{7}$$

According to the Karush–Kuhn–Tucker's (KKT) conditions of solving quadratic programming problem, only some of $(\alpha_i - \alpha_i^*)$ in Eq. (6) are non-zero. These data points on non-zero coefficient are referred to as the *support vector*. Usually, the ε value determines the number of support vectors, which represents the sparser level of the solution.

By using ω of Eq. (5), the regression function can be expressed as follows.

$$f(x) = \sum_{i=1}^{n}(\alpha_i^* - \alpha_i)x_i^T x + b \tag{8}$$

From Eq. (8), the support vectors, whose approximation errors will equal to or larger than ε, determine the final regression function. Intuitively, since errors lower than ε are accepted, training data points lying inside the so called "ε-tube" have no impact on the problem solution. In a sense, these data points are redundant, the discarding of them can reduce the problem complexity and accelerate the learning process.

For nonlinear regression problems, "kernel trick" is the key to extending the above linear regression of support vector machine [21]. This is done by mapping the input patterns into a higher-dimensional space $\mathcal{F}$ by a function $\phi : \mathcal{W} \to \mathcal{F}$. Therefore, the linear regression on the higher-dimensional space will be equivalent to the nonlinear regression on the input patterns. Kernel function that satisfies Mercer's condition can be used in this part [22], and the dot product $\langle x_i, x_j\rangle$ in Eq. (8), therefore, becomes a kernel function $\langle\phi(x_i), \phi(x_j)\rangle = K(x_i, x_j)$ for nonlinear cases. In this study, the author discusses the use of Gaussian kernel function

$$K(x, x\prime) = \exp\left(\frac{-(x - x\prime)^2}{2 \times \delta^2}\right) \tag{9}$$

The Gaussian kernel parameter δ^2 is determined by the user.

It should be notice that the forecasting ability of a SVR model depends on a good setting of the three parameters (C, ε, and δ^2). Thus, the selection of all three parameters is an important issue. For the above reason, the APSO algorithm is used in the proposed SVR model to optimize the parameter selection.

2.2. Adaptive particle swarm optimization for SVR prediction model

It is difficult to decide the best parameters of the SVR. APSO algorithm is employed here to find the optimum parameters (C, ε, and δ^2).

2.2.1. Principle of particle swarm optimization

Inspired by swarm behaviors such as bird socking and fishes schooling, Kennedy and Eberhart firstly presented a new swarm