Expert Systems with Applications 37 (2010) 4519–4525

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Reducing samples for accelerating multikernel semiparametric support vector regression

Yong-Ping Zhao *, Jian-Guo Sun, Xian-Quan Zou

Department of Energy and Power Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, PR China

ARTICLE INFO

Keywords:
Support vector regression
Multiple kernel learning
Semiparmetric technique

ABSTRACT

In this paper, the reducing samples strategy instead of classical ν-support vector regression (ν-SVR), viz. single kernel ν-SVR, is utilized to select training samples for admissible functions so as to curtail the computational complexity. The proposed multikernel learning algorithm, namely reducing samples based multikernel semiparametric support vector regression (RS-MSSVR), has an advantage over the single kernel support vector regression (classical ε-SVR) in regression accuracy. Meantime, in comparison with multikernel semiparametric support vector regression (MSSVR), the algorithm is also favorable for computational complexity with the comparable generalization performance. Finally, the efficacy and feasibility of RS-MSSVR are corroborated by experiments on the synthetic and real-world benchmark data sets.

1. Motivation

The support vector machines (SVMs) proposed by Vapnik and his group (Burges, 1998; Cristianini & Shawe-Taylor, 2000; Schölkopf & Smola, 2002; Vapnik, 1995) have a foolproof theoretical foundation, viz. structural risk minimization (SRM) principle, which minimizes the upper bound of generalization error consisting of training errors and the confidence interval. As a state-of-the-art tool, SVM was first presented to cope with binary classification problem, and then it was extended to multi-classification situation. Further, in the regression realm, as a variable, support vector regression (SVR) is employed to deal with function approximation and regression estimation (Smola & Schölkopf, 2004). Although SVM has obtained many successful applications, such as image segmentation (Chen & Wang, 2005), time series prediction (Lau & Wu, 2008), face recognition (Guo, Li, & Chan, 2001), optimal control (Suykens, Vandewalle, & De Moor, 2001), etc., there still exist some obstacles like training problem, that is, the training complexity $O(N^3)$, where N is the ensemble size of the training samples. Hence, some algorithms like Chunking (Osuna, Freund, & Girosi, 1997), SMO (Platt, 1998; Shevade, Keerthi, Bhattacharyya, & Murthy, 2000), SVM^{light} (Joachims, 1999), SVMTorch (Collobert & Bengio, 2001), LIBSVM (Chang & Lin, 2001a) were developed to alleviate the computational burden. However, these mitigations are limited. So after analyzing the avenue of selecting support vectors, Wang and Xu (2004) proposed a heuristic method to accelerate SVR training based on a measurement of similarity among training samples, i.e., at first, all the training samples are divided into several groups and then for each group, some training data will be discarded based on the measurement of similarity among samples. Subsequently, with the aid of the estimated ε-tube using multiple bootstrap samples to compute the probability for each pattern, Kim and Cho (2006) gave a pattern selection method to reduce the training time of SVR. By virtue of k-nearest neighbors (kNN) strategy, Guo and Zhang (2007) developed a method to cut the training time by deducing the number of training samples based on the observation that the support vector's target value is usually a local extremum or near extremum. Compared with the heuristic method above, Guo et al.'s method is favorable for the prediction accuracy with fewer reduced training samples. In addition, Guo et al.'s method holds obvious advantage over Kim et al's in the computational cost with the comparable generalization performance. Recently, without loss of prediction accuracy, Zhao and Sun (2009) improved Guo et al.'s method and curtailed the computational cost from $O(kMN)$ to $O(MN)$, where M is the calculation circle for the reduced procedure, through defining a measurement function instead of kNN to measure the similarity among the training samples.

Recently, multikernel learning has drawn more attentions and some achieved algorithms (Bi, Zhang, & Bennett, 2004; Lanckriet, Cristianini, Bartlett, Ghaoui, & Joran, 2004; Ong, Smola, & Williamson, 2005) have demonstrated the necessity to consider multiple kernels or the combination of kernels rather than a single fixed kernel. In many real-world systems, they own different data trends in different regions. If we only use a single kernel to learn these unknown systems, it may lead to not sufficiently learning them so as to obtain larger prediction errors for regression and higher error rate for classification. Hence, it becomes necessary to explore mult-

* Corresponding author.
E-mail addresses: y.p.zhao@nuaa.edu.cn, y.p.zhao@tom.com (Y.-P. Zhao), jgspe@nuaa.edu.cn (J.-G. Sun), xqzou2006@yahoo.cn (X.-Q. Zou).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.12.058

ikernel learning algorithms suitable for these complicated systems. After combining with semiparametric technique (Smola, Frie, & Schölkopf, 1998), Nguyen and Tay (2008) proposed multikernel semiparametric support vector regression (MSSVR) to boost learning effects of the systems owning different data trends in different regions. This multikernel learning algorithm needs two stages to complete it. At first stage, we utilize the ν-support vector regression (Chang & Lin, 2001b, 2002; Schölkopf, Smola, Williamson, & Bartlett, 2000) to select a small subset of the training samples, and then the nonlinear mappings, which center the preselected training samples using their corresponding kernel functions, are utilized to construct admissible functions for the next stage. Finally, considering multikernel admissible functions, a multikernel semiparametic model is obtained. Although admissible functions are not regularized in the cost function, the generalization performance is not affected seriously due to the sufficiently small number of admissible functions. The MSSVR can achieve good results for some systems, but its training cost is more expensive than that of classical SVR. To this end, through analyzing MSSVR in substantially more detail, we find that it takes a lot of time to select samples to construct admissible functions for MSSVR using classical ν-SVR. Therefore, if we are able to seek an algorithm with less computational complexity instead of the classical ν-SVR, the training burden of MSSVR is naturally reduced even comparable with that of the classical SVR. Hence, in this paper, we utilize the reducing samples strategy as a surrogate of ν-SVR to select training samples for admissible functions of MSSVR so that the computational complexity is obviously curtailed without loss of the prediction accuracy. Meantime, the effectiveness and feasibility of the proposed algorithm are confirmed by experiments on the synthetic and real-world benchmark data sets.

The remainder of the paper is organized as follows: In Section 2, in a nutshell, the multikernel learning algorithms, including multikernel semiparametric ε-SVR and multikernel semiparametric ν-SVR, are introduced. In the following section, to curtail the training time of selecting admissible functions, the reducing samples based MSSVR (RS-MSSVR) is detailedly depicted. Subsequently, the effectiveness and feasibility of the proposed RS-MSSVR are confirmed with the experimental results on the synthetic and real-world benchmark data sets. Finally, conclusions follow.

2. Multikernel semiparametric support vector regression

2.1. Multikernel semiparametric ε-SVR

Considering the training set $\{(\boldsymbol{x}_i, d_i)\}_{i=1}^N$, where $\boldsymbol{x}_i \in \Re^n$ is the input variable and $d_i \in \Re$ is the corresponding output variable, with the ε-insensitive loss function, we can get the following model

$$\min_{\boldsymbol{w},\boldsymbol{b}} \quad \left\{\frac{1}{2}\boldsymbol{w}_s^T\boldsymbol{w}_s + C\sum_{i=1}^N(\xi_i^* + \xi_i)\right\} \tag{1}$$
$$\text{s.t.} \quad d_i - \boldsymbol{w}_s\cdot\varphi_s(\boldsymbol{x}_i) - \sum_{p=1}^B b_p\phi_p(\boldsymbol{x}_i) \leqslant \varepsilon + \xi_i^*$$
$$\boldsymbol{w}_s\cdot\varphi_s(\boldsymbol{x}_i) + \sum_{p=1}^B b_p\phi_p(\boldsymbol{x}_i) - d_i \leqslant \varepsilon + \xi_i$$
$$\xi_i^*, \quad \xi_i \geqslant 0, \quad i = 1,\ldots,N$$

where $\varepsilon > 0$ is the width of the tolerance band, $C > 0$ is the user-selected regularization parameter, $\boldsymbol{w}_s$ represents the model complexity, $\varphi_s(\cdot)$ is usually a nonlinear mapping which is induced from the kernel function $k_s(\cdot,\cdot)$, $\phi_p(\cdot)$ is the admissible function, B is the total number of the admissible functions. Eq. (1) is the semiparametric ε-SVR in the primal. However, if $\phi_p(\cdot)$ is induced from another kernel function or different parameter kernel function from $\varphi_s(\cdot)$, then we obtain multikernel semiparametric ε-SVR. In order to solve (1), through constructing the Lagrange function, we can find the dual of (1) as

$$\min_{\boldsymbol{\alpha}^*,\boldsymbol{\alpha}} \quad \left\{\frac{1}{2}\sum_{i,j=1}^N(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)k_s(\boldsymbol{x}_i,\boldsymbol{x}_j) + \varepsilon\sum_{i=1}^N(\alpha_i^* + \alpha_i) - \sum_{i=1}^N d_i(\alpha_i^* - \alpha_i)\right\} \tag{2}$$
$$\text{s.t.} \quad \sum_{i=1}^N(\alpha_i^* - \alpha_i)\phi_p(\boldsymbol{x}_i) = 0$$
$$0 \leqslant \alpha_i^*, \quad \alpha_i \leqslant C, \quad i = 1,\ldots,N, \ p = 1,\ldots,B$$

where $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*, \ldots, \alpha_N^*]^T$ and $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T$ are Lagrange multipliers, $k_s(\boldsymbol{x}_i,\boldsymbol{x}_j) = \varphi_s(\boldsymbol{x}_i)\cdot\varphi_s(\boldsymbol{x}_j)$ is the kernel function. Usually the Gaussian is our choice. After solving (2), the following predictor is obtained

$$f(\boldsymbol{x}) = \sum_{i=1}^N(\alpha_i^* - \alpha_i)k_s(\boldsymbol{x}_i,\boldsymbol{x}) + \sum_{p=1}^B b_p\phi_p(\boldsymbol{x}) \tag{3}$$

2.2. Multikernel semiparametric ν-support vector regression

Instead of the fixed ε, the ν-SVR introduces another parameter ν which can automatically tune the width of the ε-insensitive band so as to control the sparseness of SVR. To be more meaningful, the parameter $\nu \in [0, 1]$ is an upper bound on the fraction of margin errors and lower bound of the fraction of support vectors. Similarly, the semiparametric technique can be also extended to ν-SVR

$$\min_{\boldsymbol{w},\boldsymbol{b},\varepsilon} \quad \left\{\frac{1}{2}\boldsymbol{w}_s^T\boldsymbol{w}_s + C\left(N\nu\varepsilon + \sum_{i=1}^N(\xi_i^* + \xi_i)\right)\right\} \tag{4}$$
$$\text{s.t.} \quad d_i - \boldsymbol{w}_s\cdot\varphi_s(\boldsymbol{x}_i) - \sum_{p=1}^B b_p\phi_p(\boldsymbol{x}_i) \leqslant \varepsilon + \xi_i^*$$
$$\boldsymbol{w}_s\cdot\varphi_s(\boldsymbol{x}_i) + \sum_{p=1}^B b_p\phi_p(\boldsymbol{x}_i) - d_i \leqslant \varepsilon + \xi_i$$
$$\varepsilon, \xi_i^*, \quad \xi_i \geqslant 0, \quad i = 1,\ldots,N.$$

The dual form of (4) is

$$\min_{\boldsymbol{\alpha}^*,\boldsymbol{\alpha}} \quad \left\{\frac{1}{2}\sum_{i,j=1}^N(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)k_s(\boldsymbol{x}_i,\boldsymbol{x}_j) + \varepsilon\sum_{i=1}^N(\alpha_i^* + \alpha_i) - \sum_{i=1}^N d_i(\alpha_i^* - \alpha_i)\right\} \tag{5}$$
$$\text{s.t.} \quad \sum_{i=1}^N(\alpha_i^* - \alpha_i)\phi_p(\boldsymbol{x}_i) = 0$$
$$\sum_{i=1}^N(\alpha_i^* + \alpha_i) \leqslant CN\nu$$
$$0 \leqslant \alpha_i^*, \quad \alpha_i \leqslant C, \quad i = 1,\ldots, \ N, \ p = 1,\ldots,B.$$

Through solving (5), we can get the similar result to (3). However, there is a problem not to posit, i.e., how to decide $\phi_p(\cdot)$. Hence, Nguyen and Tay (2008) proposed a two-stage strategy to deal with it. At the first stage, since the classical ν-SVR is easy to control the sparseness of the solution, it is utilized to choose some so-called support vectors. Subsequently, during the second stage, the nonlinear mappings induced from the above-selected support vectors and their corresponding kernel functions are regarded as admissible functions to construct multikernel semiparametric ε-SVR, viz. (2). Thus, the multikernel semiparametric support vector regression (MSSVR) is obtained, for the admissible function $\phi(\cdot)$ and $\varphi_s(\cdot)$