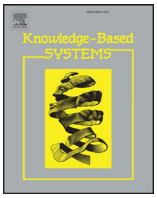
Knowledge-Based Systems 112 (2016) 54–66

Contents lists available at ScienceDirect

Knowledge-Based Systems

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



Projection support vector regression algorithms for data regression

Xinjun Peng*, Dong Xu

Department of Mathematics, Shanghai Normal University, Shanghai, 200234, China

ARTICLE INFO

Article history:
Received 6 November 2015
Revised 21 August 2016
Accepted 31 August 2016
Available online 7 September 2016

Keywords:
Support vector regression
Projection axis
Prior information
Correlation coefficient
Structural regularization

ABSTRACT

Support vector regression (SVR), which has been successfully applied to a variety of real-world problems, simultaneously minimizes the regularization error and empirical risk with a suitable penalty factor. However, it does not embed any prior information of data into the learning process. In this paper, by introducing a new term to seek a projection axis of data points, we present a novel projection SVR (PSVR) algorithm and its least squares version, i.e., least squares PSVR (LS-PSVR). The projection axis not only minimizes the variance of the projected points, but also maximizes the empirical correlation coefficient between the targets and the projected inputs. The finding of axis can be regarded as the structural information of data points, which makes the proposed algorithms be more robust than SVR. The experimental results on several datasets also confirm this conclusion. The idea in this work not only is helpful in understanding the structural information of data, but also can be extended to other regression models.

1. Introduction

In the past decade, due to the excellent generalization performance and structural risk minimization (SRM), support vector machine (SVM) [4,5,26,32,33], including support vector classification (SVC) [4,5,32,33], support vector regression (SVR) [26,32,33] and the extensions [13,19] have become the useful tools for data classification and regression, and have been successfully applied to a variety of real-world problems [10,17].

For classical SVR, it finds a function $f(\boldsymbol{x})$ that has at most ε deviation from the actually obtained targets y_i's for all training points, and, at the same time, is as flat as possible. In other words, it does not care about errors as long as they are less than ε, but will not accept any deviation larger than this. There exist many algorithms to learn SVR, such as the sequential minimal optimization (SMO) algorithm [24] and smooth algorithm [16]. Some researchers have also proposed a series of new models based on different loss functions, such as least squares SVR (LS-SVR) [28,29] and Huber loss based SVR [32,33]. Some other methods, including normal LS-SVR [21], heuristic training [35], and geometric methods [3], etc., have been discussed. Recently, we have proposed a class of novel nonparallel-planes models for data regression in the spirit of the twin SVM classifier [13], include twin SVR (TSVR) [19] and twin parametric insensitive SVR (TPISVR) models [20]. These models determine indirectly the regressor through a pair of nonparallel up- and down-bound functions solved by two smaller-sized SVM-type problems, which make they not only have the faster learning speed in theory, but also obtain the comparable generalization performance with SVR.

The generalization performance of classical SVR is obviously influenced by the parameters. Fig. 1 gives a linear example to interpret this problem. For this example, it only needs to adjust the C value in linear SVR (3) given ε. It can be found that in Fig. 1 SVR obtains a poor performance given a small C value. Factually, the weight of regularization term in SVR, i.e., $\frac{1}{2}\boldsymbol{w}_x^T\boldsymbol{w}_x = \frac{1}{2}||\boldsymbol{w}_x||^2$, will become small if C is small. Then, we have a small slope value for the regression line of this problem. To overcome this deficiency in classical SVR, one strategy is to choose a larger C value, but it may lead to the over-fitting phenomena for many real-world problems. In fact, if one can embed the prior structure information of data into the regularization factor of SVR such that a small variance value of the projected points on the normal vector direction can be obtained, this shortcoming will be easily overcome. On the other hand, the prior structure information of data is helpful in improving the performance of a regressor.

In this paper, we present a novel SVR model for data regression, called the projection support vector regression (PSVR). Specifically, this PSVR is dedicated to generating a projection axis for the training points, such that the projected points have as small as possible empirical variance value. In other words, it embeds the prior information of data into the learning process by introducing a new term into its objective function. Thus, it can overcome the above shortcoming in the classical SVR. As the classical SVR, it also introduces the ε-insensitive loss function to depict the training error, and introduces the l_2-norm regularization term to avoid the

* Corresponding author.
E-mail address: xjpeng@shnu.edu.cn (X. Peng).

http://dx.doi.org/10.1016/j.knosys.2016.08.030
0950-7051/© 2016 Elsevier B.V. All rights reserved.

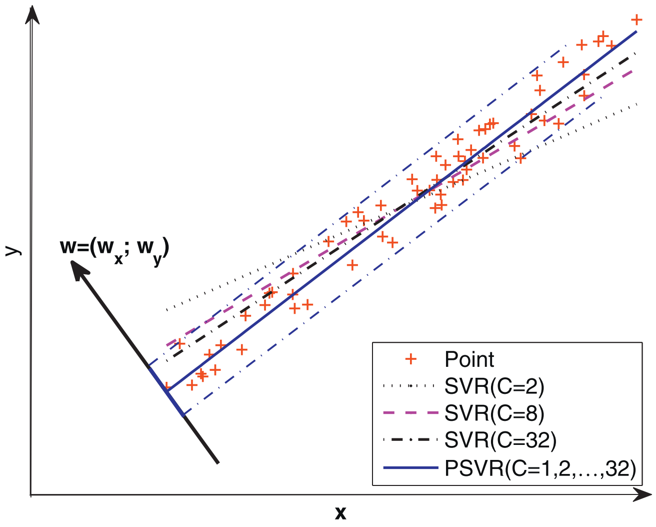


Fig. 1. Geometric interpretation for PSVR.

possible over-fitting phenomenon, i.e., to make the function be as flat as possible. The benefits of PSVR can be described as follows:

- Compared with the classical SVR, this PSVR inherits the merits of SVR by employing the ε-insensitive loss and the l_2-norm regularization, which makes the PSVR also have the sound theoretical foundation, i.e., the SRM principle. Further, the introduced projection axis in the PSVR makes the projected points have a small variance. Then, it leads the PSVR to obtain a better fitting on training points, see Fig. 1. Meanwhile, this PSVR obtains the much stable results with different C values. For example, it can be found that this PSVR in Fig. 1 obtains the same results with different C values. In addition, the PSVR has less number of support vectors (SVs) than the classical SVR. Remark that, for SVM-type tools, the SV number is one of the most important measure to evaluate the performance and reflect the robustness [27,37]. Hence, this PSVR should have a better robustness than SVR.
- The finding of projection axis in the PSVR can be regarded as the structural information of data points, see Xue et al.'s work [39]. Intuitively, this strategy leads to a good generalization performance since the structural information in data is embedded into the model. However, this structural information in our PSVR is not identical with that in Xue et al.'s work [39] since it not only embeds the structural information of input into the learning model, but also considers the structural information of target and the relationship between the input and target. That is, it maximizes the empirical correlation coefficient between the projected input points and their targets. This difference is much reasonable for regression since a regression function is to depict the relationship between the inputs and targets.

In the spirit of the LS-SVR model, we extend this PSVR to a least squares version, i.e., the least squares PSVR (LS-PSVR). In terms of generalization, the experimental results indicate that this proposed PSVR obtains the better prediction performance with less number of SVs than the classical SVR and LS-PSVR. Also, the LS-PSVR obtains the better prediction performance than the SVR and LS-SVR. In addition, the experiments show that the PSVR is much less insensitive to the penalty factor than the SVR.

There are many projection (or transformation)-based machine-learning algorithms [12], such as the partial least squares regression (PLSR) [31] and the orthogonal least squares (OLS) method [6]. It should be pointed out this PSVR is different with respect to them. For instance, instead of finding hyperplanes of minimum variance between the response and input variables, the PLSR finds a linear regression model by projecting the input variables and the response variables to a new space. The PLSR is particularly suited when the matrix of responses has more variables than inputs, and when there is multi-collinearity among input values. While the OLS method finds the regression by forward selecting variables. By contrast, standard regression will fail in these cases.

The rest of this paper is organized as follows: Section 2 briefly introduces the classical SVR and LS-SVR. Section 3 presents the proposed PSVR and LS-PSVR. Furthermore, it gives some discussion on our methods and some other related methods. Experimental results on benchmark datasets are given in Section 4. Some conclusions and remarks are discussed in Section 5.

2. Background

In this section, we first introduce briefly the classical SVR [26,32,33] and LS-SVR [28,29], and then review some related work for this study.

2.1. SVR & LS-SVR

Without loss of generality, the training samples are denoted by a set $\mathcal{D}=\{\boldsymbol{z}_i=(\boldsymbol{x}_i;y_i), i=1\ldots,n\}$, where the inputs $\boldsymbol{x}_i\in\mathcal{X}\subset\mathcal{R}^m$, the targets (or responses) $y_i\in\mathcal{R}$, $i=1,\ldots,n$, and $\mathcal{X}$ denotes the space of the input patterns.

As a state-of-the-art of machine-learning algorithm, the classical SVR [26,32,33] finds a function $f(\boldsymbol{x})$ that has at most ε deviation from the actually obtained targets y_i's for all the training data, and, at the same time, is as flat as possible. In other words, it does not care about errors as long as they are less than ε, but will not accept any deviation larger than this. Specifically, the SVR finds a linear regression function f, taking the form

$$f(\boldsymbol{x})=\boldsymbol{w}_x^T\boldsymbol{x}+b, \tag{1}$$

tolerating a small error in fitting this given data set, where $\boldsymbol{w}_x\in\mathcal{X}$ and $b\in\mathcal{R}$. Flatness in the case of (1) means that one seeks a small $\boldsymbol{w}_x$. One way to ensure this is to minimize its l_2-norm, i.e., $||\boldsymbol{w}_x||^2=\boldsymbol{w}_x^T\boldsymbol{w}_x$. We can write this problem as a convex optimization problem:

$$\begin{aligned}\min\quad & \frac{1}{2}\boldsymbol{w}_x^T\boldsymbol{w}_x\\ \text{s.t.}\quad & y_i-\left(\boldsymbol{w}_x^T\boldsymbol{x}_i+b\right)\le\varepsilon,\\ & \left(\boldsymbol{w}_x^T\boldsymbol{x}_i+b\right)-y_i\le\varepsilon,\ \forall i,\end{aligned} \tag{2}$$

The tacit assumption in (2) is that such a function f actually exists that approximates all pairs $(\boldsymbol{x}_i;\ y_i)$ with ε precision, or in other words, that the convex optimization problem is feasible. Sometimes, however, this may not be the case, or we also may want to allow for some errors. For this aim, one can introduce slack variables ξ_i, ξ_i^* to deal with the infeasible constraints of the optimization problem (2). That is, we introduce the ε-insensitive loss function

$$l_\varepsilon(\boldsymbol{x},y,f)=\begin{cases}0, & \text{if } |y-f(\boldsymbol{x})|\le\varepsilon,\\ |y-f(\boldsymbol{x})|-\varepsilon, & \text{otherwise},\end{cases}$$

for the training points. Hence we arrive at the formulation shown in the follows:

$$\begin{aligned}\min\quad & \frac{1}{2}\boldsymbol{w}_x^T\boldsymbol{w}_x+\frac{C}{n}\sum_{i=1}^{n}\left(\xi_i+\xi_i^*\right)\\ \text{s.t.}\quad & y_i-\left(\boldsymbol{w}_x^T\boldsymbol{x}_i+b\right)\le\varepsilon+\xi_i,\ \xi_i\ge0,\\ & \left(\boldsymbol{w}_x^T\boldsymbol{x}_i+b\right)-y_i\le\varepsilon+\xi_i^*,\ \xi_i^*\ge0,\ \forall i.\end{aligned} \tag{3}$$