Knowledge-Based Systems 59 (2014) 85–96

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Training Lagrangian twin support vector regression via unconstrained convex minimization

S. Balasundaram *, Deepak Gupta

School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi 110067, India

ARTICLE INFO

Article history:
Received 3 June 2013
Received in revised form 19 January 2014
Accepted 19 January 2014
Available online 27 January 2014

Keywords:
Generalized Hessian
Gradient based iterative methods
Smooth approximation
Support vector regression
Twin support vector regression
Unconstrained convex minimization

ABSTRACT

In this paper, a new unconstrained convex minimization problem formulation is proposed as the Lagrangian dual of the 2-norm twin support vector regression (TSVR). The proposed formulation leads to two smaller sized unconstrained minimization problems having their objective functions piece-wise quadratic and differentiable. It is further proposed to apply gradient based iterative method for solving them. However, since their objective functions contain the non-smooth 'plus' function, two approaches are taken: (i) either considering their generalized Hessian or introducing a smooth function in place of the 'plus' function, and applying Newton–Armijo algorithm; (ii) obtaining their critical points by functional iterative algorithm. Computational results obtained on a number of synthetic and real-world benchmark datasets clearly illustrate the superiority of the proposed unconstrained Lagrangian twin support vector regression formulation as comparable generalization performance is achieved with much faster learning speed in accordance with the classical support vector regression and TSVR.

1. Introduction

Support Vector Machines (SVMs) introduced by Vapnik [31] are extremely powerful kernel-based machine learning tools applicable for binary classification and regression problems. SVM has been successfully applied to many real-world classification problems ranging from image classification [21], text characterization [13], biomedicine [3,10] to bankruptcy prediction [30].

Though it owns better generalization classification performance over other machine learning methods such as artificial neural networks (ANNs), one of the main challenges of SVM is its high learning cost, i.e. $O(m^3)$ where m is the number of training data points. To improve its learning speed, during the past years, multi-hyperplane SVM classifiers have been proposed in the literature [12,19,22,25] wherein non-parallel hyperplanes are constructed instead of a single hyperplane as in the classical SVM. The earlier contribution in this direction is the generalized eigenvalue proximal SVM (GEPSVM) proposed by Mangasarian and Wild [19]. In their approach, non-parallel hyperplanes are constructed in which data points of each class will be proximal to one of its two non-parallel hyperplanes. In the sprit of GEPSVM, Jayadeva et al. [12] proposed twin SVM (TWSVM) for binary classification. TWSVM seeks two non-parallel hyperplanes by solving a pair of quadratic programming problems (QPPs) of smaller size than a single large one as in the classical SVM. Since the learning speed of TWSVM is approximately four times faster than the classical SVM [12] and furthermore it owns improved generalization ability in comparison to SVM and GEPSVM, it becomes one of the most attractive methods for classification. For related works on the extension/improvement of TWSVM, see [14,15,27].

Recently, inspired by the work of TWSVM, Peng [23] proposed twin support vector regression (TSVR) in which the unknown regressor is estimated by constructing a pair of non-parallel ε-insensitive down- and up-bound functions. Similar to TWSVM, the non-parallel bound functions are obtained by solving a pair of dual QPPs of smaller size than the single large one of the classical support vector regression (SVR). Empherical results show that TSVR obtains good generalization with the added advantage of faster learning speed in comparison to SVR [23]. Formulating TSVR as a pair of strongly convex unconstrained minimization problems in primal and employing smooth technique, a new SVR called smooth twin support vector regression (STSVR) has been proposed in [4] where its solution has been obtained using Newton-Armijo algorithm [16,17]. For the study of a simple and linearly convergent Lagrangian TSVR algorithm, the interested reader is referred to [1]. Again, on the formulation of TSVR as a pair of linear programming problems, see [35] and also for other variants of TSVR, see [26,34]. Finally, on an interesting robust algorithm for classification problems with outliers or noises, we refer [33].

* Corresponding author. Tel.: +91 11 26704724; fax: +91 11 26741586.
E-mail addresses: bala_jnu@hotmail.com, balajnu@gmail.com (S. Balasundaram), deepakjnu85@gmail.com (D. Gupta).

http://dx.doi.org/10.1016/j.knosys.2014.01.018
0950-7051/

Motivated by the works on Lagrangian TSVR [1] and the Newton approach for the dual SVM classification formulation of [36], a new unconstrained Lagrangian TSVR (ULTSVR) formulation has been proposed in this study. However, since its objective function contains a term having non-smooth 'plus' function, two gradient based approaches are assumed to solve the proposed minimization problem: (i) either considering its generalized Hessian [8,11] or introducing the smooth approximation function of [16] in place of the non-smooth 'plus' function, and then applying Newton-Armijo algorithm; (ii) obtaining its critical point using functional iterative method. The convergence of the Newton-Armijo algorithm and its finite termination will follow directly from the results of [16,17]. Under a sufficient condition, the linear convergence of the proposed functional iterative method is proved in this paper. Finally, the effectiveness of the proposed ULTSVR problem is demonstrated by performing experiments on a number of interesting synthetic and real-world datasets and comparing their results with SVR and TSVR.

Throughout in this work, all vectors are assumed as column vectors. The inner product of two vectors $\mathbf{x}$, $\mathbf{y}$ in the n-dimensional real space R^n is denoted by: $\mathbf{x}^t\mathbf{y}$, where $\mathbf{x}^t$ is the transpose of $\mathbf{x}$. For any vector $\mathbf{x} = (x_1,\ldots, x_n)^t \in R^n$, the plus function $\mathbf{x}_+$ is defined as: $(\mathbf{x}_+)_i = \max\{0,x_i\}$ and $i = 1, \ldots, n$. The 2-norm of a vector $\mathbf{x}$ and a matrix Q will be denoted by $\|\mathbf{x}\|$ and $\|Q\|$ respectively. We denote the vector of ones of dimension m by $\mathbf{e}$ and the identity matrix of appropriate size by I. If f is a real valued function of the variable $\mathbf{x} = (x_1, \ldots, x_n)^t \in R^n$ then its gradient vector and Hessian matrix are denoted by: $\nabla f = (\partial f/\partial x_1,\ldots, \partial f/\partial x_n)^t$ and $\nabla^2 f = (\partial^2 f/\partial x_i \partial x_j)_{i,\ j=1,\ \ldots,\ n}$ respectively.

The paper is organized as follows. In Section 2, the classical SVR and TSVR are reviewed. The proposed unconstrained TSVR problem in its dual form and three iterative methods of solving it are described in Section 3. Numerical experiments have been performed on a number of synthetic and real-world datasets and their results have been compared with that of SVR and TSVR in Section 4 and finally the conclusions and future works are drawn in Section 5.

2. Related work

In this section, the classical SVR and its variant namely twin SVR are briefly described.

Assume that a set of training samples $\{(\mathbf{x}_i, y_i)\}_{i=1,\ 2,\ \ldots,\ m}$ is given in which $y_i \in R$ is the observed value corresponding to the input data point $\mathbf{x}_i \in R^n$. Further, let us suppose that the training data points be represented by a matrix $A \in R^{m\times n}$ whose ith row is taken to be the row vector $\mathbf{x}_i^t$ and the vector of observed values be denoted by $\mathbf{y} = (y_1,\ldots, y_m)^t$.

2.1. Support vector regression (SVR) formulation

In the classical SVM for regression, the non-linear regression estimating function f: $R^n \to R$ is assumed to be of the form:

$$f(\mathbf{x}) = \mathbf{w}^t\varphi(\mathbf{x}) + b,$$

where $\varphi(.)$ is a non-linear mapping which takes the input data points into a higher dimensional feature space, $\mathbf{w}$ is a vector in the feature space and b is a scalar threshold. The ε-insensitive SVR solves the unknowns $\mathbf{w}$ and b as the solution of the constrained QPP given below [5,31]:

$$\min_{\mathbf{w},b,\xi_1,\xi_2} \frac{1}{2}\mathbf{w}^t\mathbf{w} + C(\mathbf{e}^t\xi_1 + \mathbf{e}^t\xi_2)$$

subject to

$$y_i - \mathbf{w}^t\varphi(\mathbf{x}_i) - b \leqslant \varepsilon + \xi_{1i}, \quad \mathbf{w}^t\varphi(\mathbf{x}_i) + b - y_i \leqslant \varepsilon + \xi_{2i}$$

and $\xi_{1i}, \xi_{2i} \geqslant 0$ for $i = 1, 2, \ldots, m$,

where $\xi_1 = (\xi_{11},\ldots, \xi_{1m})^t$, $\xi_2 = (\xi_{21},\ldots, \xi_{2m})^t$ are vectors of slack variables and $C > 0$, $\varepsilon > 0$ are input parameters.

Rather than solving the primal problem considered above, by introducing Lagrange multipliers $\mathbf{u}_1 = (u_{11},\ldots, u_{1m})^t$ and $\mathbf{u}_2 = (u_{21},\ldots, u_{2m})^t$ in R^m and applying the kernel trick [5,31], i.e. taking:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i)^t\varphi(\mathbf{x}_j)$$

where $k(.,.)$ is a kernel function, its dual problem of the following form is solved:

$$\min_{\mathbf{u}_1,\mathbf{u}_2} \frac{1}{2}\sum_{i,j=1}^{m}(u_{1i}-u_{2i})k(\mathbf{x}_i,\mathbf{x}_j)(u_{1j}-u_{2j}) + \varepsilon\sum_{i=1}^{m}(u_{1i}+u_{2i}) - \sum_{i=1}^{m}y_i(u_{1i}-u_{2i})$$

subject to

$$\sum_{i=1}^{m}(u_{1i} - u_{2i}) = 0 \text{ and } 0 \leqslant \mathbf{u}_1, \mathbf{u}_2 \leqslant C\mathbf{e}.$$

In this case, the decision function $f(.)$ will become [5,31]: For any input data $\mathbf{x} \in R^n$,

$$f(\mathbf{x}) = \sum_{i=1}^{m}(u_{1i} - u_{2i})k(\mathbf{x}, \mathbf{x}_i) + b.$$

2.2. Twin support vector regression (TSVR) formulation

Motivated by the work of TWSVM [12] for binary classification problems, twin support vector regression (TSVR) was proposed in [23] wherein two non-parallel functions are constructed as estimators for the ε–insensitive down- and up-bounds of the unknown regression function. Unlike solving a single QPP having $2m$ number of constraints where m being the number of input data points, a pair of QPPs each having m number of linear inequality constraints is solved in TSVR. This strategy makes the training of TSVR faster than the classical SVR [23].

Suppose, let the down- and up-bound regressors for the linear TSVR in the input space be expressed as: For any $\mathbf{x} \in R^n$,

$$f_1(\mathbf{x}) = \mathbf{w}_1^t\mathbf{x} + b_1 \text{ and } f_2(\mathbf{x}) = \mathbf{w}_2^t\mathbf{x} + b_2 \tag{1}$$

respectively, where $\mathbf{w}_1, \mathbf{w}_2 \in R^n$ and $b_1, b_2 \in R$ are unknowns. Then, the linear TSVR algorithm determines the down- and up-bound regressors (1) as the solutions of the following pair of QPPs defined by [23]:

$$\min_{(\mathbf{w}_1,b_1,\xi_1)\in R^{n+1+m}} \frac{1}{2}\|\mathbf{y} - \varepsilon_1\mathbf{e} - (A\mathbf{w}_1 + b_1\mathbf{e})\|^2 + C_1\mathbf{e}^t\xi_1$$

subject to

$$\mathbf{y} - (A\mathbf{w}_1 + b_1\mathbf{e}) \geqslant \varepsilon_1\mathbf{e} - \xi_1, \quad \xi_1 \geqslant 0 \tag{2}$$

and

$$\min_{(\mathbf{w}_2,b_2,\xi_2)\in R^{n+1+m}} \frac{1}{2}\|\mathbf{y} + \varepsilon_2\mathbf{e} - (A\mathbf{w}_2 + b_2\mathbf{e})\|^2 + C_2\mathbf{e}^t\xi_2$$

subject to

$$(A\mathbf{w}_2 + b_2\mathbf{e}) - \mathbf{y} \geqslant \varepsilon_2\mathbf{e} - \xi_2, \quad \xi_2 \geqslant 0 \tag{3}$$

respectively, where $C_1, C_2 > 0$; $\varepsilon_1, \varepsilon_2 > 0$ are input parameters and ξ_1, ξ_2 are vectors of slack variables in R^m.

Remark 1. The optimization problem (2) can be equivalently written as a minimization problem of the form

$$\min_{(\mathbf{w}_1,b_1)\in R^{n+1}} \frac{1}{2}\sum_{i=1}^{m}(y_i - f_1(\mathbf{x}_i) - \varepsilon_1)^2 + C_1\sum_{i=1}^{m}\max\{0, -(y_i - f_1(\mathbf{x}_i) - \varepsilon_1)\},$$

whose objective function represents the training error generated by the whole training samples in approximating the ε-insensitive