https://doi.org/10.20965/jaciii.2017.p1017

Paper:

L_1-Norm Least Squares Support Vector Regression via the Alternating Direction Method of Multipliers

Ya-Fen Ye[*,], Chao Ying[***], Yue-Xiang Jiang[*], and Chun-Na Li[**]**

[*]College of Economics, Zhejiang University
Hangzhou 310027, China
[**]Zhijiang College, Zhejiang University of Technology
Hangzhou 310024, China
[***]Rainbow City Primary School
Hangzhou 310013, China
Email: jiangyuexiang@zju.edu.cn

[Received December 30, 2016; accepted April 20, 2017]

In this study, we focus on the feature selection problem in regression, and propose a new version of L_1 support vector regression (L_1-SVR), known as L_1-norm least squares support vector regression (L_1-LSSVR). The alternating direction method of multipliers (ADMM), a method from the augmented Lagrangian family, is used to solve L_1-LSSVR. The sparse solution of L_1-LSSVR can realize feature selection effectively. Furthermore, L_1-LSSVR is decomposed into a sequence of simpler problems by the ADMM algorithm, resulting in faster training speed. The experimental results demonstrate that L_1-LSSVR is not only as effective as L_1-SVR, LSSVR, and SVR in both feature selection and regression, but also much faster than L_1-SVR and SVR.

Keywords: support vector regression, L_1-norm, least squares, feature selection, ADMM

1. Introduction

Feature selection is an important and pervasive problem in regression. The main goal of feature selection is to discard the redundant or uninformative features and retain the useful ones. Feature selection in support vector regression (SVR) [1–4] has been widely studied [5–10]. However, the solution of standard SVR method [1] lacks sparseness and may utilize all features without discrimination. Thus, it is not suitable to address the feature selection problem.

L_1-norm support vector regression (L_1-SVR) was proposed to overcome this drawback [7, 8]. Compared to the standard SVR solution, the L_1-SVR solution is much sparser. This implies that it has an inherent feature selection property [11, 12]. However, the training speed of L_1-SVR is low. We propose a least squares version of L_1-SVR, which is based on the least squares support vector regression (LSSVR) [13] known as L_1-LSSVR, to convert the inequality constraints into equality ones. We adopt an alternating direction method of multipliers (ADMM) [14, 15], which is a simple yet powerful algorithm, to solve L_1-LSSVR. The ADMM algorithm is based on a variable splitting method to obtain a constrained optimization formulation, which is then addressed with the augmented Lagrangian method. The proposed algorithm reduces the computational complexity of L_1-LSSVR significantly.

L_1-LSSVR has the following characteristics: (i) The linear L_1-LSSVR has the ability to select important features and discard the rest; (ii) When these selected features are structurally nonlinear, the nonlinear L_1-LSSVR realizes the regression problem effectively; (iii) L_1-LSSVR needs to solve only the resulting constrained optimization, leading to a higher training speed. The experimental results of both artificial and real-world data sets demonstrate the superiority of L_1-LSSVR. In particular, compared to LSSVR and SVR, the proposed L_1-LSSVR not only selects fewer features but also has good regression effectiveness. Furthermore, by using the ADMM algorithm, the training speed of L_1-LSSVR is much higher compared to that of L_1-SVR and SVR.

In Section 2 of this paper, we introduce the standard SVR, L_1-SVR, and LSSVR. In Section 3, we describe the linear and nonlinear L_1-LSSVR. Section 4 describes the artificial and UCI datasets experiments and Section 5 presents the conclusion of our study.

2. Background

Consider the following regression problem in an n-dimensional real vector space R^n. Let (A, Y), denote a training set in which A is an $l \times n$ matrix and the i-th row $A_i \in R^n$ represents the i-th training sample, where $i = 1, 2, \ldots, l$. Let $Y = (y_1, y_2, \ldots, y_l)^T$ denote the response vector of the training sample, where $y_i \in R$.

We then review the standard SVR, L_1-SVR, and LSSVR, which are closely related to the proposed L_1-LSSVR. For simplicity, we introduce only their linear versions. The optimal linear regression function is as fol-

lows:

$$f(x) = w^T x + b, \quad \ldots \quad (1)$$

where $w \in R^n$ and $b \in R$.

2.1. Support Vector Regression

The primal problem of the standard SVR [1–4], is as follows:

$$\begin{aligned} \min_{w,b,\xi,\eta} \ & \frac{1}{2}||w||^2 + C(e^T\xi + e^T\eta) \\ \text{s.t. } & Y - (Aw + eb) \le \varepsilon e + \xi, \quad \xi \ge 0, \\ & (Aw + eb) - Y \le \varepsilon e + \eta, \quad \eta \ge 0, \end{aligned} \quad \ldots \quad (2)$$

where $||.||^2$ represents the L_2-norm, $C > 0$ is a parameter determining the tradeoff between the empirical risk and the regularization term, and e is a vector of ones of appropriate dimensions.

The parameters in function (1) are determined by problem (2). The standard SVR may suffer from the presence of redundant or uninformative features because the solution w lacks sparseness, which means that the standard SVR may use all features without discrimination.

2.2. L_1-Support Vector Regression

By replacing the square of the L_2-norm in problem (2) with the L_1-norm, the linear L_1-SVR [7, 8] is expressed as follows:

$$\begin{aligned} \min_{w,b,\xi,\eta} \ & ||w||_1 + C(e^T\xi + e^T\eta) \\ \text{s.t. } & Y - (Aw + eb) \le \varepsilon e + \xi, \ \xi \ge 0, \\ & (Aw + eb) - Y \le \varepsilon e + \eta, \ \eta \ge 0, \end{aligned} \quad \ldots \quad (3)$$

where $||.||_1$ represents the L_1-norm, C is a positive parameter, and ξ and η are slack vectors.

By using the L_1-norm, a small enough C will drive some coefficients of w_i toward zero [11, 12]. This means that w is more sparse than that of the standard SVR. Thus, L_1-SVR has an inherent feature selection property.

2.3. Least Squares Support Vector Regression

The standard SVR [1–4] is time-consuming because it involves solving a quadratic programming problem (QPP) with linear inequality constraints. To improve the training speed, [13] used an equality constraint to introduce LSSVR, which is expressed as follows:

$$\begin{aligned} \min_{w,b,\xi} \ & \frac{1}{2}||w||^2 + \frac{C}{2}\xi^T\xi \\ \text{s.t. } & Y - (Aw + eb) = \xi. \end{aligned} \quad \ldots \quad (4)$$

In comparison, LSSVR solves only a system of linear equations and improves the training speed significantly. However, LSSVR tends to lose sparseness [16] because it is formulated based on the L_2-norm.

3. L_1-Norm Least Squares Support Vector Regression

Combining the idea of L_1-SVR and LSSVR, we propose a new feature selection algorithm called L_1-LSSVR. In the following sections, we will present a linear L_1-LSSVR version, and then extend the linear L_1-LSSVR to a nonlinear version.

3.1. Linear L_1-Norm Least Squares Support Vector Regression

L_1-LSSVR searches for an optimal linear regression function:

$$f(x) = w^T x + b, \quad \ldots \quad (5)$$

where $w \in R^n$ and $b \in R$. By introducing a regularization term $\|w\|_1 + |b|$ and a slack variable ξ, the primal problem of the proposed L_1-LSSVR can be expressed as follows:

$$\begin{aligned} \min_{w,b,\xi} \ & \|w\|_1 + |b| + \frac{C}{2}\xi^\top\xi \\ \text{s.t. } & Y - (Aw + eb) = \xi, \end{aligned} \quad \ldots \quad (6)$$

where $C > 0$ is a parameter determining the trade-off between the empirical risk and regularization term. The regularization term, $\|w\|_1 + |b|$ in problem (6), is similar to that in [17–19].

To solve L_1-LSSVR, we adopt the ADMM algorithm. We define $z = [w; b]$ and $G = [A, e]$. Then, problem (6) can be expressed as follows:

$$\begin{aligned} \min_{z,\xi} \ & \|z\|_1 + \frac{C}{2}\xi^\top\xi \\ \text{s.t. } & Y - Gz = \xi. \end{aligned} \quad \ldots \quad (7)$$

According to the constraint in problem (7), we can obtain the following problem:

$$\min_z \ \|z\|_1 + \frac{C}{2}\|Y - Gz\|_2^2 \quad \ldots \quad (8)$$

Using the following translation form,

$$\begin{aligned} \min_{z,u} \ & \|u\|_1 + \frac{C}{2}\|Y - Gz\|_2^2 \\ \text{s.t. } & u - z = 0, \end{aligned} \quad \ldots \quad (9)$$

the ADMM iterating procedures become

$$z^{k+1} = \arg\min_z \frac{C}{2}\|Y - Gz\|_2^2 + \frac{\mu}{2}\|z - u^k - d^k\|_2^2, \quad (10)$$

$$u^{k+1} = \arg\min_u \|u\|_1 + \frac{\mu}{2}\|z^{k+1} - u - d^k\|_2^2, \quad \ldots \quad (11)$$

$$d^{k+1} = d^k - (z^{k+1} - u^{k+1}), \quad \ldots \quad (12)$$

where $\{z^k \in R^{n+1},\ k = 0, 1, \ldots\}$, $\{u^k \in R^{n+1},\ k = 0, 1, \ldots\}$, and $\{d^k \in R^{n+1},\ k = 0, 1, \ldots\}$ are three sequences.

Problem (10) requires solving a quadratic problem, the solution of which is

$$z^{k+1} \leftarrow B^{-1}w, \quad \ldots \quad (13)$$

where $B \equiv CG^TG + \mu I$ and $w \equiv CG^TY + \mu(u^k + d^k)$. B is always invertible, as $\mu > 0$.

Algorithm 1 ADMM for problem (7)

Input: Training data matrix $G = [A, e]$; Parameters C, and μ.
Output: Solution w^* and b^*.
Process:
1 Set $k = 1$, choose u^0, and d^0;
2 **repeat**
3 $w \leftarrow CG^T Y + \mu(u^k + d^k)$
4 $z^{k+1} \leftarrow B^{-1}w$
5 $v^{k+1} \leftarrow z^{k+1} - d^k$
6 $u^{k+1} \leftarrow soft(v^k, 1/\mu)$
7 $d^{k+1} \leftarrow d^k - (z^{k+1} - u^{k+1})$
8 $k \leftarrow k+1$
9 **until** stopping criterion is satisfied;
10 Obtain solution $(w^{*T}, b^*) = z^*$.
end

Algorithm 2 Linear L_1-LSSVR with feature selection

Input: Training data matrix $G = [A, e]$; Parameters C and μ.
Output: Selected feature index set F'; Components of $\widetilde{x}$; Approximate solution $\widetilde{w}^*$ and b^*.
Process:
1 Apply **Algorithm 1** to problem (7) to get the solution $(w^*; b^*)$;
2 Select the feature index set: $F' = \{j | |[w^*]_j| > 0, j = 1, \ldots, n\}$;
3 Set $\widetilde{w}^* = ([w^*]_{s_1}, \ldots, [w^*]_{s_k})$ and $\widetilde{x} = ([x]_{s_1}, \ldots, [x]_{s_k})$, where $s_j \in F'$;
4 Construct regression function $f(\widetilde{x}) = (\widetilde{w}^*)^T \widetilde{x} + b^*$.
end

The solution of problem (11) would be the well-known threshold [14, 15]:

$$u^{k+1} \leftarrow soft\left(v^k, \frac{1}{\mu}\right), \quad \ldots\ldots\ldots (14)$$

where $v^k \equiv z^{k+1} - d^k$. The ADMM algorithm for problem (7) is detailed in **Algorithm 1**.

By obtaining the solution of problem (7) w^* using **Algorithm 1**, we have either $|[w^*]_j| \neq 0$ or $|[w^*]_j| = 0$, where $j = 1, 2, \ldots, n$. When $|[w^*]_j| \neq 0$, the corresponding features are selected. The remaining features are considered redundant and thus discarded. Therefore, the linear L_1-LSSVR can realize feature selection effectively by using **Algorithm 2**.

3.2. Nonlinear L_1-Norm Least Squares Support Vector Regression

To extend the linear L_1-LSSVR to a nonlinear one, we express the regression function in kernel space as follows:

$$f(x) = K(x^T, A^T)w + b, \quad \ldots\ldots\ldots (15)$$

where K is a Gaussian kernel. Using the same idea as the linear L_1-LSSVR, the primal problem of the nonlinear

Algorithm 3 ADMM for problem (16)

Input: Training data matrix $H = [K(A, A^T), e]$; Parameters C, and μ.
Output: w^* and b^*.
Process:
1 Set $k = 1$, choose u^0, and d^0;
2 **repeat**
3 $w \leftarrow CH^T Y + \mu(u^k + d^k)$
4 $z^{k+1} \leftarrow B^{-1}w$
5 $v^{k+1} \leftarrow z^{k+1} - d^k$
6 $u^{k+1} \leftarrow soft(v^k, 1/\mu)$
7 $d^{k+1} \leftarrow d^k - (z^{k+1} - u^{k+1})$
8 $k \leftarrow k+1$
9 **until** stopping criterion is satisfied;
10 Obtain solution $(w^{*T}, b^*) = z^*$.
end

L_1-LSSVR is formulated as follows:

$$\min_{w,b,\xi} \ \frac{1}{2}\|w\|_1 + \frac{C}{2}\xi^\top \xi \quad \text{s.t. } Y - (K(A, A^T)w + eb) = \xi. \quad \ldots\ldots (16)$$

Problem (16) can be also be rewritten as

$$\min_{z,\xi} \ \|z\|_1 + \frac{C}{2}\xi^\top \xi \quad \text{s.t. } Y - Hz = \xi, \quad \ldots\ldots\ldots (17)$$

where $z = [w; b]$ and $H = [K(A, A^T), e]$. We now apply the ADMM algorithm using the following translation form:

$$\min_{z,\xi} \ \|z\|_1 + \frac{C}{2}\|Y - Hz\|_2^2 \quad \text{s.t. } u - z = 0. \quad \ldots\ldots\ldots (18)$$

The ADMM iterating procedures are

$$z^{k+1} = \arg\min_z \frac{C}{2}\|Y - Hz\|_2^2 + \frac{\mu}{2}\|z - u^k - d^k\|_2^2, \quad (19)$$

$$u^{k+1} = \arg\min_u \|u\|_1 + \frac{\mu}{2}\|z^{k+1} - u - d^k\|_2^2, \quad \ldots (20)$$

$$d^{k+1} = d^k - (z^{k+1} - u^{k+1}), \quad \ldots\ldots\ldots (21)$$

where $\{z^k \in R^{l+1},\ k = 0, 1, \ldots\}$, $\{u^k \in R^{l+1},\ k = 0, 1, \ldots\}$, and $\{d^k \in R^{l+1},\ k = 0, 1, \ldots\}$ are three sequences.

The z-update, which involves solving a quadratic problem, can be written explicitly as follows:

$$z^{k+1} \leftarrow B^{-1}w, \quad \ldots\ldots\ldots (22)$$

where $B \equiv CH^T H + \mu I$, and $w \equiv CH^T Y + \mu(u^k + d^k)$.

The solution of problem (20) would be the threshold [14, 15]:

$$u^{k+1} \leftarrow soft\left(v^k, \frac{1}{\mu}\right), \quad \ldots\ldots\ldots (23)$$

where $v^k \equiv z^{k+1} - d^k$. The solution of problem (16) is obtained by using **Algorithm 3**.

Algorithm 4 Nonlinear L_1-LSSVR with feature selection

Input: Training data matrix $G = [A, e]$; Parameters C, and μ.
Output: w^* and b^*.
Process:
1 Use **Algorithm 2** to get the selected feature index set F' and $\widetilde{x}$;
2 Set $H = [K(\widetilde{A}, \widetilde{A}^T)\ \widetilde{e}]$, where $\widetilde{A}$ is the new input data, as the definition of matrix A; choose parameters C and μ.
3 Use **Algorithm 3** to get solution $(w^{*T}, b^*) = z^*$ and construct regression function $f(\widetilde{x}) = K(\widetilde{x}^T, \widetilde{A}^T)w^* + b^*$.
end

When a data set is structurally nonlinear, we combine the superiority of both linear and nonlinear L_1-LSSVR to realize feature selection and nonlinear regression in two steps. We generate reduced input space features via the linear L_1-LSSVR as in [20]. Based on this reduced input space, the regression procedure is then realized via the nonlinear L_1-LSSVR. The nonlinear L_1-LSSVR with the feature selection property is described in **Algorithm 4**.

4. Numerical Test

In this section, we conduct experiments to demonstrate the regression effectiveness of the proposed L_1-LSSVR, and then analyze its feature selection ability. We then test the training speed of the proposed L_1-LSSVR. All experiments are conducted in a MATLAB R2011b environment on a PC running on a Windows XP OS with 64 bit, 3.10 GHz Intel(R) Xeon(R) processor equipped with 6 GB of RAM. In our experiments, the parameters of these algorithms, including the Gaussian kernel parameter δ, are obtained by searching in the range of 2^{-8} to 2^8.

Table 1 lists the evaluation criteria [19, 21, 22]. Let m be the number of testing samples, $\hat{y}_i$ be the prediction value of y_i , and $\bar{y} = 1/m \cdot \sum_{i=1}^{m} y_i$ be the average value of $y_1, \ldots, y_m$. The sum of squared error (SSE) is used to evaluate the predicted ability of an estimator. The total sum of squares (SST) reflects the under lying variance of the testing samples. The sum of squares for regression (SSR) reflects the explanation ability of the regressor. NMSE is normalized mean squared error. R^2 is the coefficient of determination. The definitions of these evaluation criteria show that the statistical information obtained from the testing samples increases R^2 increases and NMSE decreases.

4.1. Regression Analysis

In this section, we test the regression effectiveness of the proposed L_1-LSSVR. We consider function $y = x^{2/3}$. The training samples of this data set are corrupted by Gaussian noise with a mean of 0 and standard deviation of 0.2. In practice, the training samples (x_i, y_i) are as follows:

$$\begin{aligned} &y_i = x_i^{\frac{2}{3}} + \zeta_i, \\ &x_i \sim U[-2,2], \quad \zeta_i \sim N(0, 0.2^2). \end{aligned} \qquad \ldots\ldots (24)$$

To avoid a biased comparison, we use a MATLAB toolbox to generate 10 independent groups of noisy samples which consist of 200 training and 200 noise test samples. **Fig. 1** shows the estimated results obtained by L_1-LSSVR, L_1-SVR, LSSVR, and SVR. **Table 2** lists their corresponding performances. This experimental results show that SSE and NMSE decrease and R^2 increases using L_1-LSSVR. This indicates that the regression effectiveness of the proposed L_1-LSSVR is as good as that of other algorithms. In addition, L_1-LSSVR requires less CPU time than L_1-SVR and SVR for the linear equality constraints and ADMM algorithm.

To further test the effect of parameter selection on the effectiveness of the proposed L_1-LSSVR regression, we investigate the influence of parameters C, μ, and δ on the NMSE and CPU time for the artificial data set. We fix parameter δ as the optimal value, and investigate the influence of C and μ on the NMSE and CPU time. In **Figs. 2(a)** and **(b)**, we see that the training speed of L_1-LSSVR changes significantly as the value of parameter μ increases. Thus, C has a strong influence on the regression results, while μ affects the training speed significantly.

Table 1. Performance metrics and their calculation.

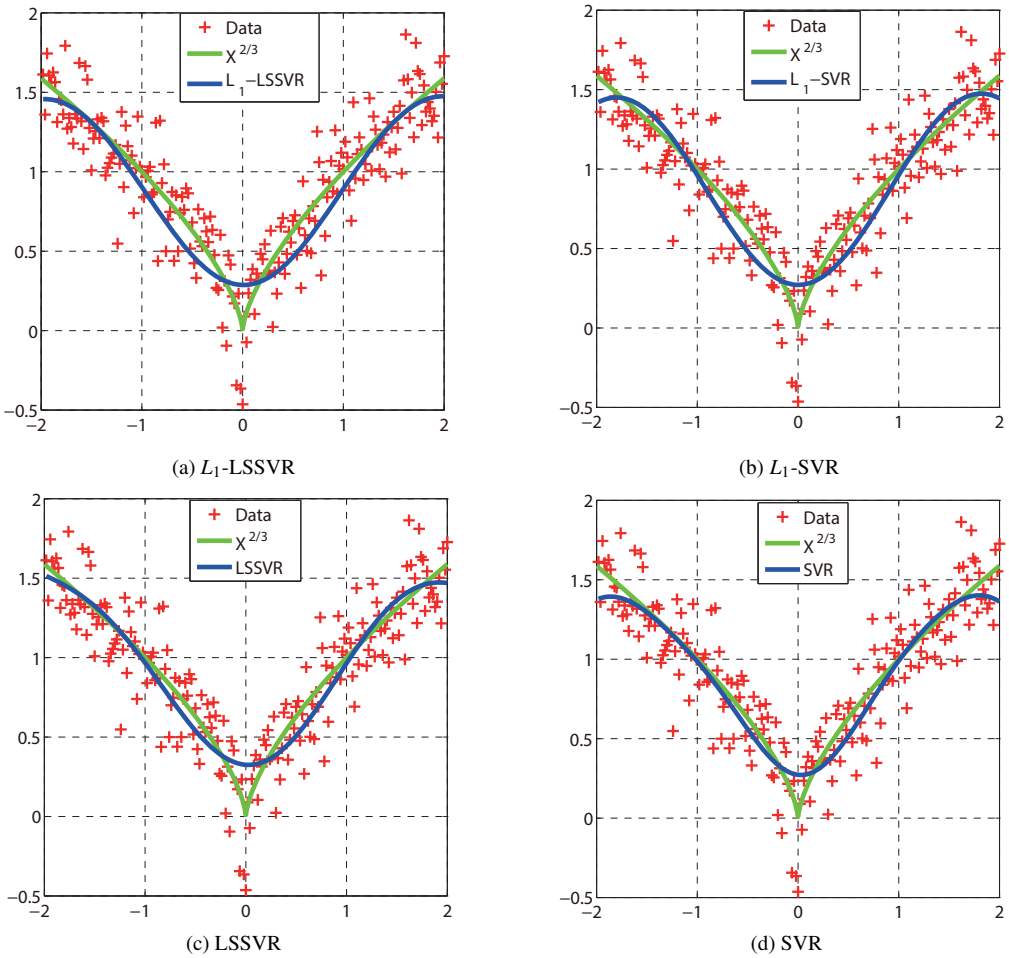
SSE	$SSE = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$
SST	$SST = \sum_{i=1}^{m} (y_i - \bar{y})^2$
SSR	$SSR = \sum_{i=1}^{m} (\hat{y}_i - \bar{y})^2$
NMSE	$NMSE = \frac{SSE}{SST} = \frac{\sum_{i=1}^{m} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{m} (y_i - \bar{y})^2}$
R^2	$R^2 = \frac{SSR}{SST} = \frac{\sum_{i=1}^{m} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{m} (y_i - \bar{y})^2}$

4.2. Feature Selection Analysis

To test the feature selection and regression effectiveness, we consider one artificial data set function [19] as follows:

$$\begin{aligned} &y_i = \frac{\sin(x_{1i} + x_{2i})}{(x_{1i} + x_{2i})} + \xi_i, \\ &x_{1i} \sim U[-4\pi, 4\pi], \\ &x_{2i} = -4\pi + 8\pi\varepsilon_i, \\ &\xi_i \sim N(0, 0.1^2), \\ &\varepsilon_i \sim U[0,1], \end{aligned} \qquad \ldots\ldots\ldots (25)$$

where $U[a,b]$ represents the uniformly random variables in $[a,b]$, and $N(c,d^2)$ represents the Gaussian random



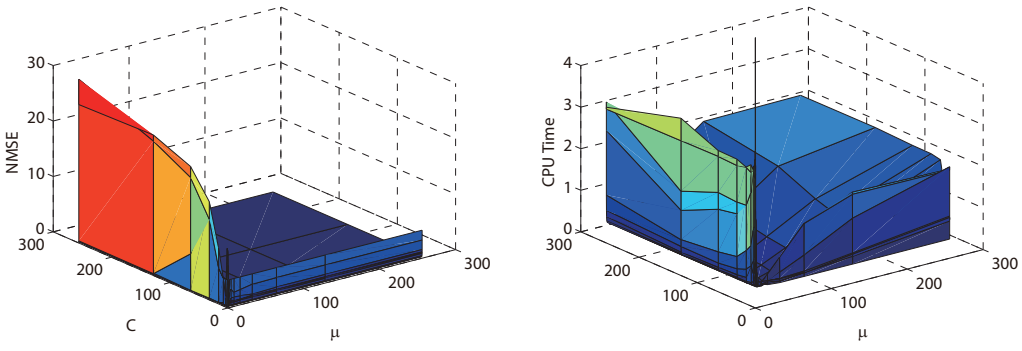
(a) L_1-LSSVR (b) L_1-SVR

(c) LSSVR (d) SVR

Fig. 1. Prediction of L_1-LSSVR, L_1-SVR, LSSVR, and SVR of function $y = x^{\frac{2}{3}}$.

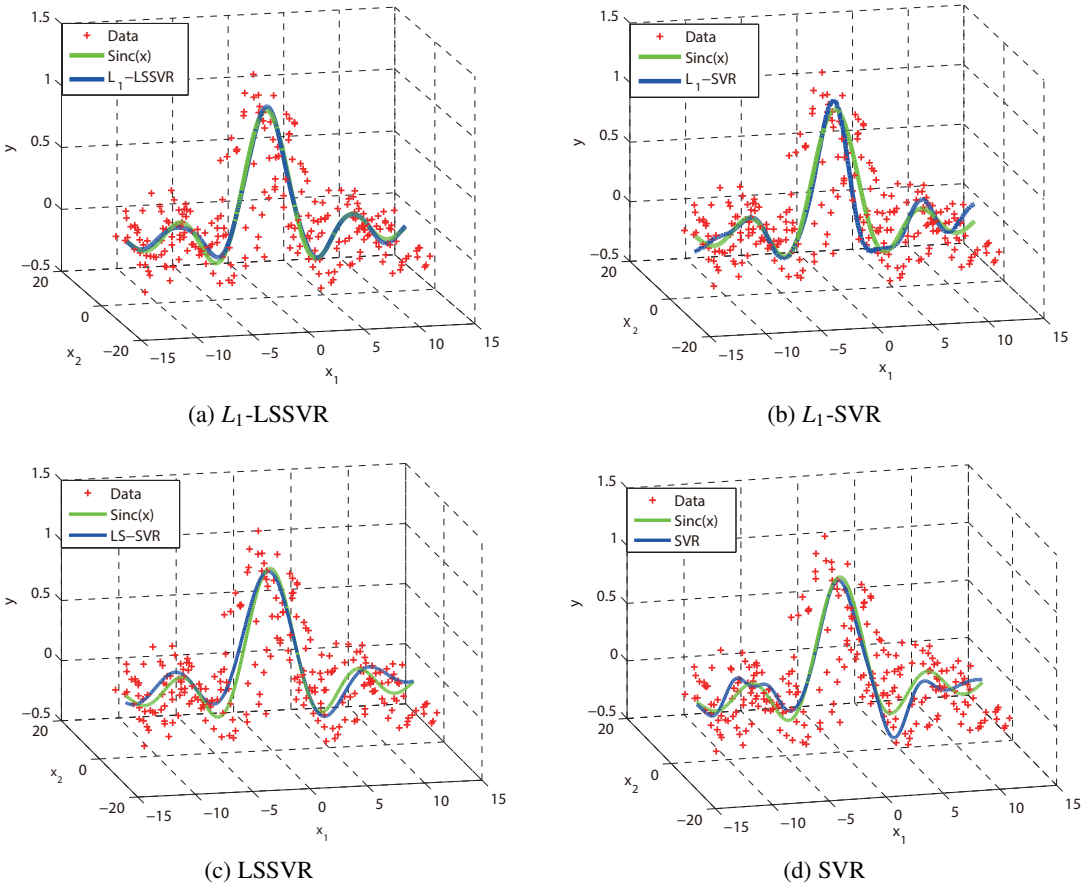
Table 2. Comparison results of L_1-LSSVR, L_1-SVR, LSSVR, and SVR in artificial data sets.

Data set	Regressor	SSE	NMSE	R^2	CPU Sec.
(24)	L_1-LSSVR	9.9200	0.2297	0.8247	0.5205
	L_1-SVR	9.7069	0.2248	**0.8810**	1.7899
	LSSVR	**9.1991**	**0.2130**	0.7804	**0.0183**
	SVR	8.9982	0.2083	0.7291	1.0352



(a) NMSE-$X^{\frac{2}{3}}$ (b) CPU time-$X^{\frac{2}{3}}$

Fig. 2. Influence of parameters C, and μ on NMSE (a), and CPU time (b). Parameter δ is fixed as 2.



(a) L_1-LSSVR (b) L_1-SVR

(c) LSSVR (d) SVR

Fig. 3. Prediction of L_1-LSSVR, L_1-SVR, LSSVR, and SVR.

Table 3. Comparison results of the proposed L_1-LSSVR, L_1-SVR, LS-SVR, and SVR for one artificial data set.

Data set	Regressor	No. of selected features	SSE	NMSE	R^2	CPU sec.
(25)	L_1-LSSVR	1	**0.3019**	**0.0056**	**0.9939**	0.2050
	L_1-SVR	1	6.2849	0.1160	0.9429	2.0285
	LSSVR	2	3.4692	0.0641	0.9703	**0.0107**
	SVR	2	4.0604	0.0750	0.9835	1.6128

variable with mean c and variance d^2. The training samples are corrupted by Gaussian noise, with a mean of 0 and a standard deviation of 0.1. Our data set consists of 252 training samples and 503 test samples.

Figures 3(a)-(d) illustrate the estimated functions obtained by using L_1-LSSVR, L_1-SVR, LSSVR, and SVR, respectively. Because the solutions of SVR and LSSVR lack sparseness, both the algorithms select two features. However, L_1-LSSVR and L_1-SVR select only the first feature x_1 and discard the second feature x_2.

Table 3 lists the regression results for different criteria. We observe that the proposed L_1-LSSVR derives the smallest NMSE and largest R^2 when compared to L_1-SVR, LSSVR, and SVR. This implies that the statistical information in the training data set is well captured by the proposed L_1-LSSVR. Moreover, the training speed of the proposed L_1-LSSVR is much higher than that of L_1-SVR for the linear equality constraints.

Furthermore, feature selection and regression tests are conducted on five real-world data sets, namely Orange Juice[1], Wine[1], Bankruptcy Prediction [23], Auto Price[2], and Boston Housing[3]. **Table 4** lists the specifications of these data sets.

Table 5 lists the feature selection and regression results of the proposed L_1-LSSVR, L_1-SVR, LSSVR, and SVR for these five data sets. We consider three comparison results, including the number of selected features, NMSE, and R^2. **Table 5** shows that the proposed L_1-LSSVR selects fewer features and results in a small NMSE and

1. http://mlg.info.ucl.ac.be/index.php?page=DataBases [accessed August 19, 2015]
2. http://www.ics.uci.edu/mlearn/ MLRepository.html [accessed July 29, 2015]
3. http://www.ics.uci.edu/mlearn/MLRepository.html [accessed July 30, 2015]

Table 4. Specification of real-world regression cases.

Data set	Training samples	Testing samples	No. of features
Orange Juice	150	68	700
Wine	94	30	256
Bankruptcy Prediction	200	300	41
Auto price	80	79	15
Boston Housing	300	206	13

Table 5. Comparison of L_1-LSSVR, L_1-SVR, LSSVR and SVR for five benchmark data sets.

Data set	Regressor	No. of selected features	SSE	NMSE	R^2	CPU sec.
Orange Juice	L_1-LSSVR	39	23.7176	0.4835	0.8583	0.2624
	L_1-SVR	**6**	53.1003	1.0825	**0.9352**	0.9096
	LSSVR	700	23.3334	0.4757	0.9197	**0.0108**
	SVR	700	**22.1670**	**0.4519**	0.8496	5.1472
Wine	L_1-LSSVR	8	0.7040	0.0517	0.8937	0.5839
	L_1-SVR	**4**	0.4893	0.0360	0.9037	0.2333
	LSSVR	256	**0.2905**	**0.0213**	**0.9320**	**0.0083**
	SVR	256	0.6265	0.0460	0.9196	0.5713
Bankruptcy	L_1-LSSVR	**11**	**29.7447**	**0.3987**	0.9421	0.1785
Prediction	L_1-SVR	35	89.3111	1.1973	0.9777	0.7558
	LSSVR	41	183.6322	0.6116	**0.9941**	**0.0094**
	SVR	41	449.1753	1.5084	0.9900	8.3231
Auto Price	L_1-LSSVR	6	16.2974	0.3360	0.7473	0.1401
	L_1-SVR	**5**	19.9546	0.4114	**0.8672**	0.1697
	LSSVR	15	10.8150	0.2229	0.8058	**0.0078**
	SVR	15	**9.9902**	**0.2059**	0.7854	0.0282
Boston Housing	L_1-LSSVR	**11**	273.0308	1.8217	0.9481	0.2944
	L_1-SVR	12	255.7137	1.7061	0.8873	5.0612
	LSSVR	13	285.0096	1.9016	**0.9770**	**0.0141**
	SVR	13	**227.6036**	**1.5186**	0.9612	0.6189

Table 6. Best parameters of L_1-LSSVR and L_1-SVR for five real-world data sets.

	L_1-LSSVR			L_1-SVR	
Data set	μ	C	δ.	C	δ.
Orange Juice	2^{-2}	2^6	2^0	2^6	2^0
Wine	2^0	2^6	2^1	2^4	2^0
Bankruptcy Prediction	2^0	2^6	2^2	2^0	2^1
Auto Price	2^2	2^5	2^1	2^4	2^1
Boston Housing	2^0	2^7	2^1	2^4	2^1

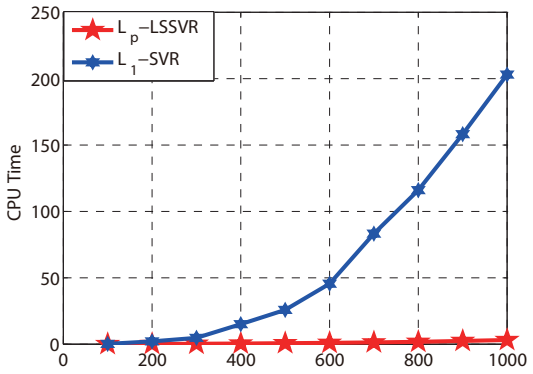
large R^2, which indicates that a few selected features capture useful information. Furthermore, **Table 5** also shows the corresponding training CPU time of these data sets, implying that the training speed of the proposed L_1-LSSVR is higher than that of L_1-SVR in the Orange Juice, Bankruptcy Prediction, Auto price, and Boston Housing data sets. Therefore, the proposed L_1-LSSVR has a higher training speed than L_1-SVR. **Table 6** lists the best parameters selected by the L_1-LSSVR and L_1-SVR algorithms.

4.3. Time Analysis

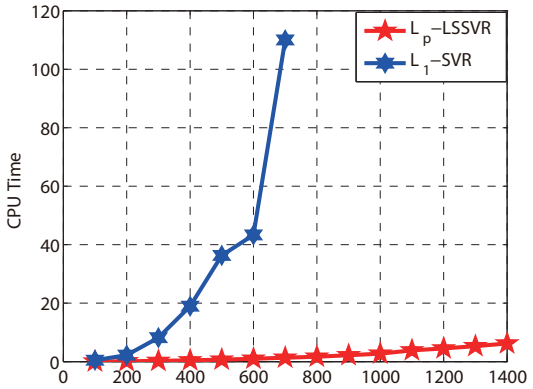
In this section, we test the influence of the sample number on the training time of the proposed L_1-LSSVR and L_1-SVR. The data sets are Anthrokids[4] and Delve-Census[5]. In our experiments, we apply the proposed L_1-LSSVR and L_1-SVR for feature selection in both the data sets. Then, both the new data sets are randomly split into training samples and testing samples. We then test the influence of the data set size on the training time.

Figures 4(a) and **(b)** show the comparison results of the computation time of both the algorithms. We observe the following: (1) The proposed L_1-LSSVR is much faster than L_1-SVR. (2) The training time of L_1-SVR exhibits a sharp increase, particularly when the data sets contain more than 400 training samples. However, the training time of the proposed L_1-LSSVR remains steady. (3) When the training samples reach 700 for the Delve-Census data set, the L_1-SVR algorithm runs out of memory, while the L_1-LSSVR algorithm still has memory. L_1-LSSVR uses an equality constraint instead of an inequality one in L_1-SVR. L_1-LSSVR applies the ADMM algorithm to solve its optimal problem.

4. http://research.cs.aalto.fi/aml/index.shtml [accessed August 22, 2015]
5. http://www.cs.toronto.edu/ delve/data/census-house/desc.html [accessed August 23, 2015]



(a) CPU time-Anthrokids



(b) CPU time-Delve-Census

Fig. 4. Training time along with the training set size on Anthrokids (a) and Delve-Census (b).

5. Conclusion

We proposed a new feature selection method in regression called L_1-LSSVR. L_1-LSSVR uses the L_1-norm that gives it an inherent features selection property. Computational comparisons between the proposed L_1-LSSVR, and L_1-SVR, LSSVR, and SVR on several data sets indicate that the proposed L_1-LSSVR can select less features with good regression performance. In addition, the proposed L_1-LSSVR operates much faster than L_1-SVR due to the employment of the ADMM algorithm, which decomposes a difficult problem into a sequence of simpler ones. Furthermore, the feature selection ability and speed of the proposed L_1-LSSVR are superior compared to those of L_1-SVR, LSSVR, and SVR.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 11201426, No. 11071252, No. 11161045, No. 11426200, and No. 61603338), the Zhejiang Provincial Natural Science Foundation of China (No. LY18G010018, No. LQ13F030010, No. LQ14G010004, No. LY15F030013, No. Y16A010057, and No. LY16A010020), Ministry of Education, Humanities and Social Sciences Research Project (No. 13YJC910011), China Postdoctoral Science Foundation (No. 2015M571848), and the 2016 Statistical Research Project of Zhejiang Province.

References:

[1] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support Vector Regression Machines," Advances in Neural Information Processing Systems 9 (NIPS 1996), 1997.

[2] C. Burges, "A tutorial on support vector machines for pattern recognition," Data Min. Knowl. Discov. Vol.2, pp. 121-167, 1998.

[3] J. Bi and K. P. Bennett, "A geometric approach to support vector regression," Neurocomputing, Vol.55, No.1-2, pp. 79-108, 2003.

[4] A. Smola and B. Schölkopf, "A tutorial on support vector regression," Statistic Computing, Vol.14, No.3, pp. 199-222, 2004.

[5] C. L. Huang and C. Y. Tsai, "A hybrid SOFM-SVR with a filter-based feature selection for stock market forecasting," Expert Systems with Application, Vol.36, pp. 1529-1539, 2009.

[6] J. B. Yang and C. J. Ong, "Feature selection using probabilistic prediction of support vector regression," IEEE Trans. on Neural Networks, Vol.22, pp. 954-962, 2011.

[7] Y. F. Ye, H. Cao, L. Bai, Z. Wang, and Y. H. Shao, "Exploring determinants of inflation in China based on L_1-ε-twin support vector regression," Procedia Computer Science, Vol.17, pp. 514-522,2013.

[8] X. Peng and D. Xu, "A local information-based feature-selection algorithm for data regression," Pattern Recognition, Vol.46, pp. 2519-2530, 2013.

[9] Y. F. Ye, Y. X. Jiang, Y. H. Shao, and C. N. Li, "Financial conditions index construction through weighted lp-norm support vector regression," J. Adv. Comput. Intell. Intell. Inform., Vol.19, pp. 397-406, 2015.

[10] Y. F. Ye, Y. H. Shao, and C. N. Li, "Wavelet lp-norm support vector regression with feature selection," J. Adv. Comput. Intell. Intell. Inform., Vol.19, pp. 407-416, 2015.

[11] R. Tibshirani, "Regression shrinkage and selection via the lasso," J. of the Royal Statistical Society, Series B (Methodological), Vol.58, No.1, pp. 207-288, 1996.

[12] L. Meier, S. van de Geer, and P. Bühlmann, "The group lasso for logistic regression," J. of the Royal Statistical Society, Series B (Statistical Methodology), Vol.70, No.1, pp. 53-71, 2008.

[13] J. A. K. Suykens, L. Lukas, P. Van Dooren, B. De Moor, and J. Vandewalle, "Least squares support vector machine classifiers: a large scale algorithm," Proc. of European Conf. of Circuit Theory Design, 1999.

[14] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," Foundations and Trends in Machine Learning, Vol.3, pp. 1-122, 2010.

[15] J. M. Bioucas-Dias and M. A. T. Figueiredo, "Alternating direction algorithm for constrained sparse regression: application to hyperspectral unmixing," 2010 2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, pp. 1-4, 2010.

[16] J. A. K. Suykens, J. D. Brabanter, L. Lukas, and J. Vandewalle, "Weighted least squares support vector machines: robustness and sparse approximation," Neurocomputing, Vol.48, pp. 85-105, 2002.

[17] Y. F. Ye, Y. X. Jiang, Y. H. Shao, and C. N. Li, "Financial conditions index construction through weighted lp-norm support vector regression," J. Adv. Comput. Intell. Intell. Inform., Vol.19, pp. 397-406, 2015.

[18] Y. F. Ye, Y. H. Shao, and C. N. Li, "Wavelet lp-norm support vector regression with feature selection," J. Adv. Comput. Intell. Intell. Inform., Vol.19, pp. 407-416, 2015.

[19] Y. H. Shao, C. H. Zhang, Z. M. Yang, L. Jing, and N. Y. Deng, "An ε-twin support vector machine for regression," Neural Computing and Applications, Vol.23, pp. 175-185, 2013.

[20] O. L. Mangasarian and E. W. Wild, "Feature selection for nonlinear kernel support vector machines," IEEE 7th Int. Conf. on data mining, 2007.

[21] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," Int. J. of Forecasting, Vol.22, pp. 679-688, 2006.

[22] Y. F. Ye, Y. H. Shao, and W. J. Chen, "Comparing inflation forecasts using an ε-wavelet twin support vector regression," J. of Information and Computational Science, Vol.10, pp. 2041-2049, 2013.

[23] Q. Yu, Y. Miche, E. Séverin, and A. Lendasse, "Bankruptcy prediction using extreme learning machine and financial expertise," Neurocomputing, Vol.128, pp. 296-302, 2014.

Name:
Ya-Fen Ye

Affiliation:
College of Science, Zhejiang University of Technology

Address:
288 Liuhe Road, Hangzhou 310023, China

Brief Biographical History:
2008 Received Master's degree in Quantitative Economics from Zhejiang Gongshang University
2011 Received Ph.D. degree in Statistics in College of Statistics and Mathematics from Zhejiang Gongshang University
2014- Associate Professor at the Zhijiang College, Zhejiang University of Technology

Main Works:
- Quantitative economics
- Machine learning and data mining

Membership in Academic Societies:
- OPTIMAL Group (http://www.optimal-group.org/)

Name:
Chao Ying

Affiliation:
Rainbow City Primary School

Address:
501 Weiye Road, Hangzhou 310013, China

Brief Biographical History:
2009 Received her Bachelor's degree in applied mathematics from Shaoxing University
2016- First-Grade Teacher at Rainbow City Primary School

Main Works:
- Machine learning and data mining

Name:
Yue-Xiang Jiang

Affiliation:
College of Economics, Zhejiang University

Address:
Hangzhou 310024, China

Brief Biographical History:
1996 Received his Ph.D. degree of Statistics from University of Bern
1999 Received his Ph.D. degree of Management from Zhejiang University
2005- Professor and Doctoral Supervisor, College of Economics, Zhejiang University

Main Works:
- Measure economics
- Random financial theory and its application
- Macro economic theory and policy

Membership in Academic Societies:
- Zhejiang Social Insurance Association, Director
- Swiss Statistics Association Research Areas, Member

Name:
Chun-Na Li

Affiliation:
Zhijiang College, Zhejiang University of Technology

Address:
182 Zhijiang Road, Hangzhou 310024, China

Brief Biographical History:
2009 Received her Master's degree from Harbin Institute of Technology
2012 Received her Ph.D. degree in Department of Mathematics from Harbin Institute of Technology
2012- Lecturer at the Zhijiang College, Zhejiang University of Technology

Main Works:
- Optimization methods
- Machine learning and data mining

Membership in Academic Societies:
- OPTIMAL Group (http://www.optimal-group.org/)