Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090657

Improved Iterative Sparseness for Least Squares Support Vector Machines

ZHONG Lu-sheng[*], *CHEN Li-yong, GONG Jin-hong and ZHU Zhen-min*

School of Electrical and Electronic Engineering, East China Jiaotong University, Nanchang 330013, China

Received: 17 Mar. 2015, Revised: 15 May 2015, Accepted: 16 May 2015
Published online: 1 Nov. 2015

Abstract: An improved iterative sparse algorithm is proposed to accelerate the execution of sparse least squares support vector machines(LS-SVM). Firstly, the technique of iterative approximation to the L0-norm is used to sparsify the LS-SVM for regression. However, each iteration requires solving a linear system with expensive computation compared to training a single LS-SVM. In this paper, improved conjugate gradient (ICG) method is given to reduce the computational cost, which is based on transforming the constrained primal problem in LS-SVM into an unconstrained minimization problem. Then the solution to the unconstrained minimization problem is obtained by using the CG method only once at each iteration. Finally, the result of numerical experiment shows that the proposed method get sparse LS-SVM model with significant reduction in computational cost.

Keywords: least squares support vector machines, L0-norm, improved conjugate gradient

1 Introduction

Least squares support vector machines(LS-SVM) is a powerful tool for classification and function approximation [1]. The LS-SVM has excellent generalization performance by performing structural risk minimization (SRM) [2,3,4]. The LS-SVM is considered as a simplification of conventional support vector machine(SVM). Instead of using nonnegative errors in cost function and inequality constraints as in SVM, the LS-SVM uses square errors in cost function and equality constrains. As a result, the LS-SVM finds the solutions of a set of linear equations instead of a quadratic programming problem in SVM. The performance of SVM and LS-SVM are excellent in real-world problems.

Although the LS-SVM shows good performance, there are still two obvious limitations. Firstly, LS-SVM lacks sparseness which is important for accurate and fast evaluation of new data points [4,5]. To obtain sparseness in LS-SVM solution, many efficient methods have been proposed. Suykens [4] proposed a simple yet effective approach by pruning the samples that have the smallest absolute support value. Hoegaerts [5] made a comparison of six pruning methods and showed that the second pruning algorithm with weighed support values achieved excellent performance,. In [6] a fast pruning strategy is presented to delete redundant hidden nodes of MFN to impose sparseness of LS-SVM. In [7] the feature vector selection (FVS) algorithm using kernel trick to define a subspace as support vectors was given to construct a sparse LS-SVM. Similar to FVS algorithm, Carvalho [8] introduced a strategy that the support vectors of LS-SVM were selected by the reduced remaining subset (RRS) technique and the decision surface between the classes was found. Brabanter [9] proposed a fixed-size kernel model to impose sparseness of LS-SVM. Secondly, solution of LS-SVM involves inverting a square matrix whose dimension grows with the number of training data. For large training data, the inversion of the matrix leads to problems of computation and storage. In order to deal with the problems caused by large samples, Suykens [10] presented a conjugate gradient (CG) method to solve an Nth order linear system twice in each iteration with N denoting the number of training data. In [11] an iterative computation of the inverse kernel matrix was developed to reduce computational cost of the linear system. Tian [12] proposed $\varepsilon-$insensitive loss function instead of the least squares error in LS-SVM and used SMO method to resolve the dual transformation of primal problem .The work [13] applied an iterative approximation of L0-norm to sparsify LS-SVM and showed that the resulting models had a generalization ability comparable to standard

[*] Corresponding author e-mail: lszhongzju@gmail.com

LS-SVM and SVM. However, the method in [13] is computationally expensive, which is not applicable to large scale problems. In this paper, to deal with the computational problem in [13], we proposed an improved conjugate gradient(CG) algorithm, which only needs to solve the $(N-1)th$ order linear system once with N denoting the number of training data. The proposed method is based on transforming the constrained primal problem in LS-SVM into an unconstrained minimization problem. Numerical experiments demonstrate that the LS-SVM sparseness is obtained by the L0-norm term and the computational cost is reduced by improved CG method. The paper is organized as follows. Section 2 introduces LS-SVM. The formulation of iterative sparse method for LS-SVM is given in section 3. Section 4 presents an improved conjugate gradient algorithm. Numerical experiment results based on three datasets are reported in section 5. Finally, conclusions are given in section 6.

2 LS-SVM regression

Given a training set $\{x_i,y_i\}_{i=1}^N$, where $x_i \in R^n$ is the ith input vector and $y_i \in R$ is the corresponding known output. Consider the regression model $y_i = f(x_i)+e_i$, $i=1,...N$ where $f: R^n \to R$ is an unknown real-valued smooth function and $e_1,...,e_N$ are uncorrelated random errors with $E[e_i]=0$, $E[e_i^2]=\sigma_e^2<\infty$ .The support vector machines(SVM) have been used for estimating the nonlinear function of the form

$$f(x)=\omega^T\cdot\varphi(x)+b \tag{1}$$

where $\varphi(x): R^n \to R^{n_H}$ denotes the feature map to the high dimensional feature space whose dimension can be infinite($n_H=\infty$), $\omega \in R_H^n, b \in R$. The LS-SVM regression problem in the primal weight space is formulated as follows

$$\begin{aligned} &\underbrace{\min}_{\omega,b,e} J(\omega,e)=\frac{1}{2}\|\omega\|^2+\frac{\gamma}{2}\sum_{i=1}^N e_i^2 \\ &s.t \quad y_i=\omega^T\cdot\varphi(x)+b+e_i, i=1,...,N \end{aligned} \tag{2}$$

where positive value γ is termed as regularization parameter to control the tradeoff between the data fitting and the smoothness of the solution. By using Lagranian multipliers, the solution of constrained optimization problem(2) can be obtained by taking Karush-Kuhn-Tucker(KKT[1,4]) conditions for optimality. The result is given by the following set of linear equations

$$\begin{bmatrix} 0 & 1_N^T \\ 1_N & \Omega+\gamma^{-1}I \end{bmatrix}\begin{bmatrix} b \\ \alpha \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{3}$$

where $y=[y_1,...,y_N]^T$, $\alpha=[\alpha_1,...,\alpha_N]^T$ denote the Lagrange multipliers, $1_N=[1,1,...,1]^T$ is a column vector of N ones,and $\Omega_{i,j}=\varphi(x_i)^T\varphi(x_j)=K(x_i,x_j)$ for $i,j=1,2,...N$ with $K(.,.)$ a positive definite kernel function. According to Mercers theorem, the resulting LS-SVM model for function estimation can be evaluated at a new point x_* as

$$\hat{f}(x_*)=\sum_{i=1}^N \alpha_i \bullet K(x_i,x_*)+b \tag{4}$$

where b,α is the solution to (3).

3 Sparse LS-SVM

The LS-SVM as a simplification of SVM has been successfully applied in many regression and classification problems[4,13,14]. Despite the good performance of LS-SVM, the. main drawback of LS-SVM is the lack of sparseness, i.e. $\alpha_i \neq 0$ for $i=1,2,...N$ .This means that nearly all patterns become support vectors. The nonsparseness of LS-SVM slows down the test speed and limits the utility of LS-SVM in large scale problems[7,14]. For this reason, a range of methods have been proposed for obtaining the sparseness of LS-SVM, see [4,5,6].Recently, the method based on an iterative approximation to the L0-norm is proposed for sparsifying SVM classifiers [16] and classical LS-SVM [13]. In this paper, we adapt the scheme in[13][16] to sparsify the LS-SVM for regression as follows.

To apply an iterative approximation to L0-norm for sparsifying LS-SVM, let us consider the following primal optimization problem

$$\begin{aligned} &\min \frac{1}{2}\sum_{i=1}^N \lambda_i\alpha_i^2+\frac{\gamma}{2}\sum_{i=1}^N e_i^2 \\ &s.t \quad \sum_{i=1}^N \alpha_j K_{ij}+b=y_i-e_i, i=1,2,...,N \end{aligned} \tag{5}$$

where λ_i are prefixed coefficients. Comparing (5) to (2), the regularization term $\|\omega\|^2$ in (2) is replaced by $\sum_{i=1}^N \lambda_i\alpha_i^2$ in (5), which is the main difference between the pruning method and relevance subset selection algorithms. The L2-norm term$\sum_{i=1}^N \lambda_i\alpha_i^2$ plays the effect to control the model complexity[16].

In order to solve the constrained optimization problem(5), a Lagrangian [4,13,16] for problem (5) is

$$\begin{aligned} L(\alpha,b,e,\beta)&=\frac{1}{2}\sum_{i=1}^N \lambda_i\alpha_i^2+\frac{\gamma}{2}\sum_{i=1}^N e_i^2 \\ &-\sum_{i=1}^N \beta_i(\sum_{j=1}^N \alpha_j^2 K_{ij}+b+e_i-y_i) \end{aligned} \tag{6}$$

where $\beta=[\beta_1,\beta_2,...,\beta_N]^T$ is the new Lagrange multipliers. The Karush-Kuhn-Tucker(KKT) conditions

[4,5,13] for optimality are

$$\begin{cases} \frac{\partial L}{\partial \alpha_i}=0 \Rightarrow \alpha_i=\sum_{j=1}^{N}\beta_j K_{ij}/\lambda_i \\ \frac{\partial L}{\partial b}=0 \Rightarrow \sum_{i=1}^{N}\beta_i=0 \\ \frac{\partial L}{\partial e_i}=0 \Rightarrow \beta_i=\gamma e_i \\ \frac{\partial L}{\partial \beta_i}=0 \Rightarrow \sum_{j=1}^{N}\alpha_j K_{ij}+b=y_i-e_i \\ i,j=1,2,...N \end{cases} \tag{7}$$

Eliminating variables α_i and e_i yields the following linear equations in Lagrange multipliers [13,16]

$$\begin{bmatrix} 0 & 1_N^T \\ 1_N & \Pi \end{bmatrix}\begin{bmatrix} b \\ \beta \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{8}$$

where $\Pi = Kdiag(\lambda)^{-1}K+I/\gamma$,I is identity matrix. To get sparse Lagrange multipliers α is sparse, an iterative sparse LS-SVM (IS-LS-SVM) algorithm is proposed by iteratively approximating to the L0-norm [13,16]. The IS-LS-SVM algorithm is summarized as follows

Step1: Get α^t and b by solving linear system (3),let $t=1$.

Step2: Set $\lambda_i^t=\alpha_i^t.i=1,2,...N$.

Step3: Solve linear system (8) to get β^t and b^t , then update $\alpha_i^{t+1}=\sum_{i=1}^{N}\beta_i^t K_{ij}/\lambda_i^t$ according to eqs(7).

Step4: Compute $\lambda_i^{t+1}=1/\alpha_i^{(t+1)^2}$, set $t=t+1$.

Step5: if $t\leq 50$ or$\| \alpha_i^{t+1}-\alpha_i^t \| /N \geq 10^{-4}$, return to step3.

Step6: Get the final α and b.

As discussed in [13,16], the multipliers α converges to a stationary point α^* as $t\rightarrow\infty$, and the L2-norm regularization term $\sum_{i=1}^{N}\lambda_i\alpha_i^2$ in (5) converges to L0-norm $\sum_{i=1}^{N}\alpha_i^*,(\alpha_i^*\neq 0)$. This makes the IS-LS-SVM model to be sparse. This scheme of iterative approximation to L0-norm is different from the pruning method[4,5,6] selecting the support vectors by pruning the samples with the smallest absolute support value. The sparseness of IS-LS-SVM model can reduce computational load for accurate and fast evaluation of new data points. However, the above algorithm to sparsifying LS-SVM is time-consuming since each iteration involves solving the linear system (8) with complexity $O(N^3)$. The computational cost is reduced by ICG algorithm in the next section.

4 Improved conjugate gradient for IS-LS-SVM

In this section, we adopt the improved conjugate gradient (ICG) algorithm [14] to accelerate the execution of linear regression system(8). The ICG algorithm gives the solutions of β and b in (8) with an unconstrained minimization dual problem, which involves solving an $(N-1)th$ order linear system by conjugate gradient (CG) algorithm. The principle of ICG method is described as follows.

According to eqs.(6)(7), the dual problem (5) can be equivalently be expressed as

$$\begin{aligned} &\min W(\beta)=\frac{1}{2}\beta^T\Pi\beta-\beta^T y \\ &s.t \quad \sum_{i=1}^{N}\beta_i=0 \end{aligned} \tag{9}$$

where $\beta\in R^N$,$\Pi\in R^{N\times N}$ are respectively defined in (6) and (8) , i.e.$\beta=[\beta_1,\beta_2,...,\beta_N]^T$, $\Pi=Kdiag(\lambda)^{-1}K+I/\gamma$. The matrix Π is denoted as

$$\Pi=\begin{pmatrix} \Pi^{(N-1)} & h \\ h^T & \Pi_{NN} \end{pmatrix} \tag{10}$$

where $\Pi^{(N-1)}$ is the $(N-1)th$ order principal submatrix of Π ,Π_{NN} is the last element of the Nth row vector of Π, and h is the last column of Π by removing Π_{NN}.In terms of the constraint (9), β_N can be computed as

$$\beta_N=-1_{N-1}^T\beta^{(N-1)} \tag{11}$$

where $\beta^{(N-1)}=[\beta_1,\beta_2,...,\beta_{N-1}]^T$,$1_{N-1}=[1,1,...1]^T$ is a column vector of $(N-1)$ ones

Substituting (11) into the objective function of (9) leads to the following unconstrained minimization problem

$$\min\frac{1}{2}\beta^{(N-1)^T}\tilde{\Pi}\beta^{(N-1)}-(y^{(N-1)}-y_N 1_{N-1})^T\beta^{(N-1)} \tag{12}$$

where
$\tilde{\Pi}=\Pi^{(N-1)}-1_{(N-1)}h^T-h1_{N-1}^T+\Pi_{NN}1_{(N-1)}1_{(N-1)}^T$, $y^{(N-1)}=[y_1,y_2,...,y_{N-1}]^T$ and y_N is the last element of vector y.

Comparing (12) to (9), the parameter vector to be optimized is $\beta^{N-1}\in R^{N-1}$ instead of $\beta\in R^N$ in (9), which means that the number of optimization parameters is reduced by one in (12).

The solution β^{N-1} to the unconstrained optimiazation (12) is described as follows.

Firstly, we denote an invertible matrix

$$\rho=\begin{pmatrix} I^{(N-1)} & -1_{N-1} \\ 0 & 1 \end{pmatrix} \tag{13}$$

where $I^{(N-1)}$ is $(N-1)\times(N-1)$ identity matrix.

In terms of eqs.(8), we can get a linear system

$$\rho(b1_N+\Pi\beta)=\rho y \tag{14}$$

The left side of (14) can be formulated as

$$b\rho 1_N + \rho\Pi\rho^T(\rho^T)^{-1}\beta = \begin{pmatrix} 0_{N-1} \\ b \end{pmatrix} + \begin{pmatrix} \widetilde{\Pi} & h - \Pi_{NN}1_{N-1} \\ h^T - \Pi_{NN}1_{N-1}^T & \Pi_{NN} \end{pmatrix} \times \begin{pmatrix} \beta^{(N-1)} \\ 0 \end{pmatrix} \quad (15)$$
$$= \begin{pmatrix} 0_{N-1} \\ b \end{pmatrix} + \begin{pmatrix} \widetilde{\Pi}\beta^{(N-1)} \\ (\Pi\beta)_N \end{pmatrix}$$

where $(\Pi\beta)_N = h^T\beta^{(N-1)} - \Pi_{NN}1_{N-1}^T\beta^{(N-1)}$

The right side of (14) is

$$\begin{pmatrix} I^{(N-1)} & -1_{N-1} \\ 0 & 1 \end{pmatrix} y = \begin{pmatrix} y^{(N-1)} - y_N 1_{N-1} \\ y_N \end{pmatrix} \quad (16)$$

By combining (15)(16), the linear system (14) can be equivalently expressed as

$$\begin{pmatrix} 0_{N-1} \\ b \end{pmatrix} + \begin{pmatrix} \widetilde{\Pi}\beta^{(N-1)} \\ \Pi\beta_N \end{pmatrix} = \begin{pmatrix} y^{(n-1)} - y_N 0_{N-1} \\ y_N \end{pmatrix} \quad (17)$$

According to Eqs.(11)(17), the solution of unconstrained optimization (12) is obtained

$$\begin{cases} \widetilde{\Pi}\beta^{(N-1)} = y^{N-1} - y_N 1_{N-1} \\ b = y_N - (\Pi\beta)_N \\ \beta_N = -1_{N-1}^T\beta^{N-1} \end{cases} \quad (18)$$

It can be seen from (18) that we only need to solve an $(N-1)th$ order linear system by CG method once at each iteration with complexity $O((3m+2)N^2)$, where m is the number of iterations for solving the $(N-1)th$ order linear system $\widetilde{\Pi}\beta^{(N-1)} = y^{N-1} - y_N 1_{N-1}$. As a comparison, solving Eqs.(8) directly involves inverting an $(N+1)th$ order square matrix which is computationally expensive with large N, The method proposed by Suykens[4] involve solving an Nth order linear system twice at each iteration.

Notice that there are some Lagrange multipliers $\alpha_i < 0, i = 1,2,...N$ in Step1 of IS-LS-SVM algorithm in section 3, which yields that the matrix $\widetilde{\Pi}$ defined in(12) is not positive definite due to relation $\lambda_i^1 = \alpha_i^1, i = 1,2,...N$. In order to make $\widetilde{\Pi}$ positive definite which is important to solve the linear system $\widetilde{\Pi}\beta^{(N-1)} = y^{N-1} - y_N\overrightarrow{1}$ by CG method, the initial values of coefficient λ_i^1 are settled as $\lambda_i^1 = 1/(\alpha_i^1 - \alpha_{min} + \eta), \eta > 0$, where α_{min} is the minimum value of $\alpha_i^1, i = 1,...,N$.

In addition, the execution of linear system (8) is accelerated by using improved CG algorithm. The detailed analysis of ICG method is given in Ref [14].

5 Experiments

To illustrate the performance of IS-LS-SVM computed by ICG method, we run experiments on three datasets: simulation dataset of *sinc* function, motorcycle dataset and diabetes dataset. Detailed information about the three datasets is presented in the following subsection. We compare the IS-LS-SVM against LS-SVM and SVM over the three datasets. We also compare the running time to solve linear system(8) by the proposed ICG method, CG method in [4] and INV method with INV denoting solution to (8) by inverting an $(N+1)th$ square matrix. In all experiment, the radial basis function is used as kernel function ,i.e.

$$K(x,x_i) = exp(-\|x - x_i\|/2\sigma^2) \quad (19)$$

where the hyper-parameters (kernel bandwidth σ and regularization parameter γ) is optimized by coupled simulated annealing (CSA) algorithm[17] , which performs 10-fold cross validation (CV) to minimize the mean squared error (MSE) of prediction value

$$MSE = \sum_{i=1}^{n}(y(x_i) - y_i)^2/n \quad (20)$$

where $y(x_i)$ is the predicted value of models(i.e. IS-LS-SVM, LS-SVM and SVM), and y_i is output of the sample, n is the number of samples.

5.1 The simulation dataset

The following *sinc* function model is considered to illustrate the sparseness and the running time of solving linear system (8)

$$y = sin(x) + \varepsilon^* \sqrt{0.05x^2 + 0.01} \quad (21)$$

where y is the output, x is the input equally spaced between -5 and 5. And ε is random disturbance subject to normal distribution. 300 datapoints are generated by *sinc* model(21), which are randomly divided into 200 training data and 100 testing data. The regression estimates of IS-LS-SVM and LS-SVM model based on training data is shown in Fig 1.

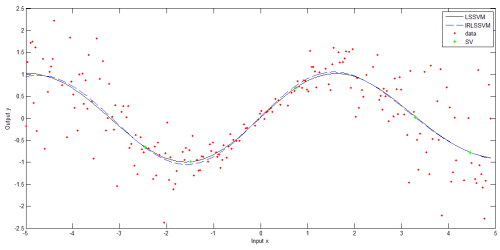

Fig. 1: The regression curve for simulation data

The red and green points respectively represent training data and support vectors (SV). The LS-SVM

regression curve is marked by black line, the blue dot-dash line denotes the IS-LS-SVM curve. Fig.1 illustrates that the IS-LS-SVM model only needs 5 support vectors while all the 200 datapoints are support vectors in LS-SVM model. In addition, The IS-LS-SVM model is almost functionally identical to the LS-SVM model. Table1 shows the number of SV and mean squared error(MSE) of the IS-LS-SVM, LS-SVM and SVM.

Table 1 Number of SV and MSE for simulation dataset

Method	SV	Training MSE	Testing MSE
SVM	192	0.3965	0.4479
LS-SVM	200	0.3579	0.4414
IS-LS-SVM	5	0.3549	0.4465

Table 1 shows that IS-LS-SVM obtain better sparseness with 5 SV comparing to 192 SV for SVM model. Regarding sparsity, IS-LS-SVM is sparser than SVM and LS-SVM. From the MSE results we can see that the training MSE of IS-LS-SVM is smallest and the generalization ability of IS-LS-SVM is comparable to that of SVM and LS-SVM.

Moreover, we compared the running time of three methods: the proposed ICG method, the CG method in [4] and INV method with INV denoting solution to linear system (8) by inverting an $(N+1)th$ square matrix. The training time of these three methods is tabulated in Table 2.

Table 2 Running time for simulation datase

	INV	CG	ICG
Running time	0.0452	0.0008377	0.0005584

Table 2 shows that the ICG method is faster than CG method and INV method is slower than CG method. The reason is that ICG method solves the $(N-1)th$ order linear system only once instead of solving an Nth linear system twice by CG method in [4]. On the other hand, the testing time for LS-SVM model and IS-LS-SVM model are respectively $1.37\times10^{-4}s$ and $7.69\times10^{-5}s$. So the prediction speed of IS-LS-SVM model is faster than LS-SVM due to the sparseness of IS-LS-SVM.

5.2 The motorcycle dataset

The motorcycle data consists of accelerometer readings through time following a simulated motor-cycle crash to determine the efficacy of crash-helmets [7]. The motorcycle dataset can be downloaded at http://www.stat.cmu.edu/~larry/all-of-statistics/=data/motor.dat.We take the time of motorcycle data as input and the accelerometer readings as output. The original motorcycle data is preprocessed by eliminating samples with the same time, which results in 94 datapoints instead of the original 133 samples. The 94 datapoints are used to establish IS-LS-VM model and LS-SVM model. Fig 2 shows the performance of the IS-LS-SVM and LS-SVM model to the 94 training data. Table 3 gives the number of SV and training MSE. From the results of Fig 2 and Table 3, we can see that the IS-LS-SVM model with 14 SV is sparser than LS-SVM model with 94 SV. And the training MSE of LS-SVM is larger than that of IS-LSSVM.

In addition, the running time to solve linear system (8) by ICG, CG and INV is tabulated in Table 2. We can see that the INV method with 0.0425 second is slower than CG method with 8.37710^{-4} second, and ICG method is faster than INV method with 0.0425 second, but CG method is slower than ICG method with 5.8410^{-4} second.

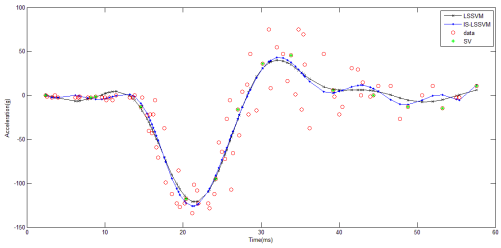

Fig. 2: The regression curve for Motorcycle data

Table 3 Number of SV and MSE for motorcycle dataset

Method	SV	Training MSE
LS-SVM	94	429.47
IS-LS-SVM	14	414.698

Table 4 Running time for Motorcycle dataset

	INV	CG	ICG
Running time	0.0112	0.0028	0.0018

5.3 The diabetes dataset

The diabetes dataset came from measurements of 442 diabetes patients[19]. The diabetes dataset describes the relation between ten baseline variables and a quantitative response variable of disease progression one year after baseline. The ten baseline variables are age, sex, body mass index, average blood pressure and six blood serum measurements. Since the diabetes dataset is related to the health problems of human being, it is widely used in the fields such as information science, statistical learning and pattern recognition etc. The diabetes dataset is downloaded at http://www.stanford.edu/~hastie/Papers/LARS/. The diabetes dataset is first standardized with

each element having zero mean and unit variance. In this experiment, the ten baseline variables and the response variable are respectively used as input vector and output for IS-LS-SVM, LS-SVM and SVM models. The total 442 samples are randomly divided into 300 training data and 112 testing data. The training MSE, testing MSE and support vectors(SV) of LS-SVM, SVM and IS-LS-SVM models are tabulated in Table 5. Fig 3 shows the predicted values for the 142 testing data using the LS-SVM, SVM and IS-LS-SVM models.

Table 5 Number of SV and mse for diabetes dataset

Method	SV	Training MSE	Testing MSE
SVM	297	0.4811	0.4627
LS-SVM	300	0.4593	0.4607
IS-LS-SVM	6	0.4739	0.4658

Table 5 illustrates that IS-LS-SVM with 6 SV is sparser than SVM with 297 SV and LS-SVM with 300 SV, which is similar to the result of Table 1. From the MSE results we can see that the training MSE of LS-SVM is smaller than that of SVM and IS-LS-SVM, and the testing MSE of IS-LS-SVM is comparable to that of SVM and LS-SVM. Fig 3 shows that the predicted value of the three models can fit well the 142 testing data.

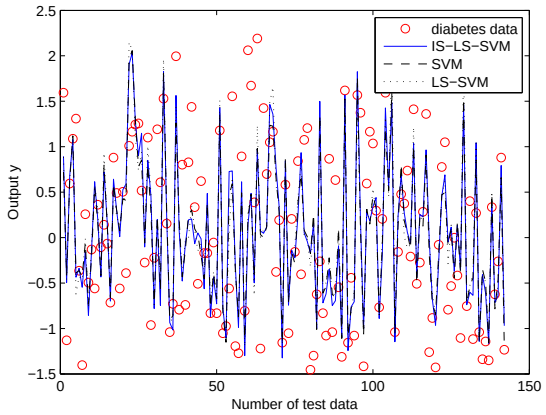

Fig. 3: Predicted performance of the three models

In addition, the running time to solve linear system (8) by ICG, CG and INV is tabulated in Table 6. We can see that the INV method with 0.1482 second is slower than CG method with 0.0653 second. The proposed ICG method with 0.017 second is faster than CG method.

Table 6 Running Time for diabetes dataset

	INV	CG	ICG
Running time	0.1482	0.0653	0.0170

Remark: Note that the number of training data for simulation dataset, motorcycle dataset and diabetes dataset are respectively 300, 94, 200. By comparing the running time of each method(i.e. INV, CG , ICG) in Table 6, Table 4 and Table2, we can see that the running time of INV method increased rapidly with more training data. The proposed ICG method is more efficient than INV and CG method for small and medium scale training datasets. Moreover, from the sparseness results of Table 1, Table 3 and Table 5, it is shown that the proposed IS-LS-SVM model is sparser than the SVM and LS-SVM model for the three different datasets. In addition, the generation ability of IS-LS-SVM is comparable to that of SVM and LS-SVM for the three different datasets.

6 Conclusion

In this paper, we apply the technique of iterative approximation to the L0-norm to sparsify the LS-SVM model. In order to reduce the computational cost of solving a $(N+1)th$ order linear system with N denoting the number of training data, improved conjugate gradient (ICG) method is given by transforming the constrained primal problem in LS-SVM into an unconstrained minimization problem. Then CG method is used to get solutions to the unconstrained minimization problem which involves solving a $(N-1)th$ order linear system only once at each iteration. Numerical experiments on several regression datasets show that the proposed method get sparse LS-SVM model as well as significant reduction in computational cost.

Acknowledgement

The first author acknowledges the financial support by National Natural Science Foundation (60904049, 61263010), The natural science foundation of Jiangxi Province (20114BAB211014, 20122BAB216026),Project of Education Department of Jiangxi Province (GJJ14399) The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments.

References

[1] Likun Hou, Qingxin Yang, Jinlong An, An improved LS-SVM regression algorithm, International Conference on Computational Intelligence and Natural Computing, pp.138-140, 2009.

[2] Mathias M. Adankon, Mohamed Cheriet, Model selection for the LS-SVM. Application to handwriting recognition, Pattern Recognition, Vol. **42**, pp.3264-3270, 2009.

[3] Xigao Shao, KunWu, and Bifeng Liao, Single directional SMO algorithm for least squares support vector machines,Computational Intelligence and Neuroscience, pp.1-7, 2013.

[4] J. A. K. Suykens, L. Lukas, and J.Vandewalle, Sparse approximation using least squares support vector machines, in Proc. IEEE Int. Symp. Circuits and System (ISCAS2000), pp.757-760, 2000.
[5] L.Hoegaerts, J. A. K. Suykens, J. Vandewalle, and B. De Moor, A comparison of pruning algorithms for sparse least squares support vector machines, In 11th ICONIP. **3316**, Calcutta, India, pp. 1247-1253.
[6] Shaohui Tao, Dezhao Chen, and Weixiang Zhao, Fast pruning algorithm for multi-output LS-SVM and its application in chemical pattern classification, Chemometrics and Intelligent Laboratory Systems, pp. 63-69, 2009.
[7] Gavin C. Cawley, Nicola L.C. Talbot, Fast exact Leave-one-out cross-validation of sparse least-squares support vector machines, IEEE Trans. Neural Netw. **17**, pp.1467-1475, 2004.
[8] Carvalho, B.P.R., Lacerda, W.S., Braga, A.P., RRS+ LS-SVM: a new strategy for a priori sample selection, Neural Computing and Applications. Springer, London, 2007.
[9] K. De Brabanter, J. De Brabanter, J.A.K. Suykens, B. De Moor, Optimized fixed-size kernel models for large data sets, Computational Statistics and Analysis,**54**, pp.1484-1504, 2010.
[10] 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, in proc, Eur. Conf. Circuit Theory and Design (ECCTD99), pp.839-842,1999.
[11] Licheng Jiao, Liefeng Bo, and Ling Wang, Fast sparse approximation for least squares support vector machine, IEEE Trans. Neural Netw. **18**, pp.685-699, 2007.
[12] Yingjie Tian, Xuchan Ju, Zhiquan Qi, and Yong Shi, Efficient sparse least squares support vector machines for pattern classification, 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp.697-701, 2012.
[13] J.Lopez, K. De Brabanter, J. R. Dorronsoro1 and J.A.K. Suykens, Sparse LS-SVMs with L0-norm minimization, in proc, of the 19th European Symposium on Artificial Neural Networks (ESANN), 2011.
[14] Bing Li, Shiji Song, Kang Li, Improved conjugate gradient implementation for least squares support vector machines, Pattern Recognition Letters, **33**, pp.121-125, 2012.
[15] Junjie Zou, Zhengtao Yu, Huangyun Zong, Xing Zhao, Active learning for sparse Least squares support vector machines, Artificial Intelligence and Computational Intelligence, **7003**, pp.672-679, 2011.
[16] Kaizhu Huang, Danian Zheng, Jun Sun, Yoshinobu Hotta, Katsuhito Fujimoto, and Satoshi Naoi, Sparse learning for support vector classification, Pattern Recognition Letters, **31**, pp.1944-1951, 2010.
[17] P. Karsmakers, K. Pelckmans, K. De Brabanter, H. Van Hamme, and J.A.K. Suykens. Sparse conjugate directions pursuit with application to fixed-size kernel models, Machine Learning, **85**, pp.109łl48, 2010.
[18] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani. Least Angle Regression. Annals of Statistics, 32(2), 407-499, 2004.

ZHONG Lu-sheng received his Ph.D. degree from Zhejiang University in Computer Engineering in 2007. He is associate professor at East China Jiaotong University. His main research interests are in the areas of statistical learning, pattern recognition, optimization, system identification.

CHEN Li-yong received his M.E degree Computer Engineering in 2014. His main research interests are in the areas of pattern recognition, optimization, system identification.

GONG Jin-hong received her M.E degree Computer Engineering in 2008. Her main research interests are in the areas of predictive control, pattern recognition, optimization, system identification.

ZHU Zhen-min received his Ph.D. degree from Tianjin University in Computer Engineering in 2011. He is associate professor at East China Jiaotong University. His main research interests are in the areas of computer vision, pattern recognition, predictive control.