ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

Comparison of Sparse Least Squares Support Vector Regressors Trained in Primal and Dual

Shigeo Abe

Kobe University - Graduate School of Engineering
Kobe, Japan

Abstract. In our previous work, we have developed sparse least squares support vector regressors (sparse LS SVRs) trained in the primal form in the reduced empirical feature space. In this paper we develop sparse LS SVRs trained in the dual form in the empirical feature space. Namely, first the support vectors that span the reduced empirical feature space are selected by the Cholesky factorization and LS SVR is trained in the dual form by solving a set of linear equations. We compare the computational cost of the LS SVRs in the primal and dual form and clarify that if the dimension of the reduced empirical feature space is almost equal to the number of training data, the dual form is faster. But the primal form is computationally more stable and for the large margin parameter the coefficient matrix of the dual form becomes near singular. By computer experiments using some benchmark data sets we verify the above results.

1 Introduction

A least squares support vector machine (LS SVM) [1] is a variant of an SVM, in which inequality constraints in an L2 SVM is replaced by equality constraints. This leads to solving a set of linear equations instead of a quadratic programming program. But the disadvantage is that all the training data become support vectors. To solve this problem, in [1], support vectors with small absolute values of the associated dual variables are pruned and the LS SVM is retrained using the reduced training data set. This process is iterated until sufficient sparsity is realized. Because the training data are reduced during pruning, information for the deleted training data is lost for the trained LS SVM. To overcome this problem, in [2], independent data in the feature space are selected from the training data, and using the selected training data the solution is obtained by the least squares method using all the training data. In [3] based on the concept of the empirical feature space proposed in [4], LS SVMs are formulated as a primal problem and by reducing the dimension of the empirical feature space, sparse LS SVMs are realized. In [5], sparse LS SVMs are extended to function approximation. Namely, the LS SVR is trained in the primal form in the empirical feature space.

In this paper we formulate dual LS SVRs in the empirical feature space and clarify the computational complexity and stability. Since the empirical feature space is finite, we can train the dual LS SVM directly by solving a set of linear equations. To generate the mapping function to the empirical feature space, we select the maximum independent components in the kernel matrix by the

ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

Cholesky factorization. And reducing the independent components we obtain a sparse LS SVR.

Stability of computation depends on the positive definiteness of the coefficient matrix of the set of linear equations. We compare positive-definiteness of the coefficient matrices of the primal and dual forms and analyze computational stability. We also compare the computational complexity of the both methods.

In Section 2, we formulate dual LS SVRs in the empirical feature space, and analyze the computational complexity and stability. In Section 3, we show the validity of the theoretical analysis by computer experiments.

2 Training in the Empirical Feature Space

2.1 Formulation

Let the approximation function be

$$D_{\mathrm{e}}(\mathbf{x}) = \mathbf{v}^T \mathbf{h}(\mathbf{x}) + b_{\mathrm{e}}, \tag{1}$$

where $\mathbf{v}$ is the N-dimensional vector, b_{e} is the bias term, $\mathbf{h}(\mathbf{x})$ is the N-dimensional vector that maps the m-dimensional vector $\mathbf{x}$ into the empirical feature space and is given by [5]:

$$h(\mathbf{x}) = (H(\mathbf{x}_{i_1}, \mathbf{x}), \ldots, H(\mathbf{x}_{i_N}, \mathbf{x}))^T. \tag{2}$$

Here $\mathbf{x}_1, \ldots, \mathbf{x}_M$ are M training data, $N\,(\leq M)$ is the dimension of the empirical feature space, and $i_j \in \{1, \ldots, M\}, j = 1, \ldots, N$. By this formulation, $\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_N}$ become support vectors.

We use the Cholesky factorization in selecting support vectors [6]. During the Cholesky factorization, if the root of the diagonal element is smaller than the prescribed value $\eta\,(> 0)$, we delete the associated row and column and continue decomposing the matrix.

The LS SVR in the empirical feature space is trained by minimizing

$$Q(\mathbf{v}, \boldsymbol{\xi}, b_{\mathrm{e}}) = \frac{1}{2}\,\mathbf{v}^T \mathbf{v} + \frac{C}{2} \sum_{i=1}^{M} \xi_i^2 \tag{3}$$

subject to the equality constraints:

$$\mathbf{v}^T \mathbf{h}(\mathbf{x}_i) + b_{\mathrm{e}} = y_i - \xi_i \quad \text{for} \quad i = 1, \ldots, M, \tag{4}$$

where $\mathbf{v}$ is the N-dimensional vector and y_i is the output for input $\mathbf{x}_i$, ξ_i is the slack variable for $\mathbf{x}_i$, and C is the margin parameter.

2.2 Primal Form

Substituting (4) into (3), we obtain

$$Q(\mathbf{v}, \boldsymbol{\xi}, b_{\mathrm{e}}) = \frac{1}{2}\,\mathbf{v}^T \mathbf{v} + \frac{C}{2} \sum_{i=1}^{M} (y_i - \mathbf{v}^T \mathbf{h}(\mathbf{x}_i) - b_{\mathrm{e}})^2. \tag{5}$$

Taking the partial derivatives of (5) with respect to $\mathbf{v}$ and b_{e} and equating them to zero, we obtain

$$b_{\mathrm{e}} = \frac{1}{M}\sum_{i=1}^{M}(y_i - \mathbf{v}^T\mathbf{h}(\mathbf{x}_i)), \tag{6}$$

$$\left(\frac{1}{C} + \sum_{i=1}^{M}\mathbf{h}(\mathbf{x}_i)\,\mathbf{h}^T(\mathbf{x}_i) - \frac{1}{M}\sum_{i,j=1}^{M}\mathbf{h}(\mathbf{x}_i)\,\mathbf{h}^T(\mathbf{x}_j)\right)\mathbf{v} = \sum_{i=1}^{M} y_i\,\mathbf{h}(\mathbf{x}_i) - \frac{1}{M}\sum_{i,j=1}^{M} y_i\,\mathbf{h}(\mathbf{x}_j). \tag{7}$$

Therefore, from (7) and (6) we obtain $\mathbf{v}$ and b_{e}. We call the LS SVR obtained by solving (7) and (6) *primal LS SVR (PrLS SVR)*.

2.3 Dual Form

Introducing the Lagrange multipliers $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_M)^T$ into (3) and (4), we obtain the unconstrained objective function as follows:

$$Q(\mathbf{v}, \boldsymbol{\xi}, b_{\mathrm{e}}, \boldsymbol{\beta}) = \frac{1}{2}\,\mathbf{v}^T\mathbf{v} + \frac{C}{2}\sum_{i=1}^{M}\xi_i^2 - \sum_{i=1}^{M}\beta_i(\mathbf{v}^T\mathbf{h}(\mathbf{x}_i) + b_{\mathrm{e}} - y_i + \xi_i). \tag{8}$$

Taking the partial derivative of (8) with respect to $\mathbf{v}$, b_{e} and ξ_i, in addition to (4), we obtain

$$\mathbf{v} = \sum_{i=1}^{M}\beta_i\,\mathbf{h}(\mathbf{x}_i), \quad \sum_{i=1}^{M}\beta_i = 0, \quad \boldsymbol{\beta} = C\boldsymbol{\xi}. \tag{9}$$

Therefore, from (4) and (9)

$$\boldsymbol{\beta} = \Omega_{\mathrm{e}}^{-1}(\mathbf{y} - \mathbf{1}\,b_{\mathrm{e}}), \qquad b_{\mathrm{e}} = (\mathbf{1}^T\Omega_{\mathrm{e}}^{-1}\mathbf{1})^{-1}\mathbf{1}^T\Omega_{\mathrm{e}}^{-1}\mathbf{y}, \tag{10}$$

where

$$\Omega_{\mathrm{e}ij} = \mathbf{h}^T(\mathbf{x}_i)\,\mathbf{h}(\mathbf{x}_j) + \frac{\delta_{ij}}{C}, \quad \mathbf{y} = (y_1, \ldots, y_M)^T, \quad \mathbf{1} = (1, \ldots, 1)^T. \tag{11}$$

We call the LS SVR obtained by solving (10) *dual LS SVR (DuLS SVR)*. And we denote LS SVRs trained in the empirical feature space in the primal or dual form as ELS SVRs.

To increase sparsity of LS SVRs, we increase the value of η. The optimal value is determined by cross-validation. We call thus trained LS SVRs *sparse LS SVRs*. To show that sparse LS SVRs are trained in the primal or dual form, we denote sparse PrLS SVRs or sparse DuLS SVRs.

2.4 Comparison

We compare the calculation time and stability of calculations by primal and dual forms for the empirical feature space. In comparing the calculation time, we evaluate the time for setting the coefficient matrix and solving the set of linear equations by the Cholesky factorization since they occupy the greater part of computation time. In setting the coefficient matrix Ω_{e} in (10) M^2 dot products with length N (M^2N multiplications) are required. We can reduce them almost by half because Ω_{e} is symmetric. Although the number of multiplications in setting the coefficient matrix in (7) is almost the same but since $\mathbf{h}(\mathbf{x}_i)\,\mathbf{h}^T(\mathbf{x}_j)$ is not a dot product, setting of the coefficient matrix in (7) is not efficient as that of Ω_{e}. The Cholesky factorization of Ω_{e} includes $o(M^3)$ multiplications and M square roots. Since the coefficient matrix in (7) is $N \times N$ and $N \leq M$, the Cholesky factorization of the coefficient matrix in (7) is faster. Thus there is a cross point in training in the dual and primal forms in the empirical feature space; if $N \approx M$, the dual form is faster and as N is smaller than the value at the cross point, the primal form is faster.

To compare the stability of calculations, we first show that the coefficient matrix in (7) is positive definite. Let $\mathbf{z}$ be an N-dimensional vector. Then the coefficient matrix is positive definite if

$$\mathbf{z}^T \left(\frac{1}{C} + \sum_{i=1}^{M} \mathbf{h}(\mathbf{x}_i)\,\mathbf{h}^T(\mathbf{x}_i) - \frac{1}{M} \sum_{i,j=1}^{M} \mathbf{h}(\mathbf{x}_i)\,\mathbf{h}^T(\mathbf{x}_j) \right) \mathbf{z} > 0 \tag{12}$$

for any $\mathbf{z} \neq \mathbf{0}$. The left hand side of the inequality is rewritten as follows:

$$\frac{1}{C}\mathbf{z}^T\mathbf{z} + \sum_{i=1}^{M} \left(\mathbf{z}^T\mathbf{h}(\mathbf{x}_i)\right)^2 - \frac{1}{M} \left(\sum_{i=1}^{M} \mathbf{z}^T\mathbf{h}(\mathbf{x}_i) \right)^2 . \tag{13}$$

Let $a_i = \mathbf{z}^T\mathbf{h}(\mathbf{x}_i)$. Then (13) becomes

$$\frac{1}{C}\mathbf{z}^T\mathbf{z} + \sum_{i=1}^{M} (a_i)^2 - \frac{1}{M} \left(\sum_{i=1}^{M} a_i \right)^2 . \tag{14}$$

Now for the last two terms in (14), the special case of the Cauchy-Schwarz inequality $\sum_{i=1}^{M}(a_i)^2 - \frac{1}{M}\left(\sum_{i=1}^{M} a_i\right)^2 \geq 0$ holds, where the strict equality holds when $a_i = 0\,(i = 1, \ldots, M)$ or $a_i = c$ (constant) $(i = 1, \ldots, M)$.

Since $\mathbf{h}(\mathbf{x}_i)$ $(i = 1, \ldots, N)$ are linearly independent, $a_i = 0\,(i = 1, \ldots, N)$ for $\mathbf{z} = \mathbf{0}$. And for $a_i = c$ (constant) $(i = 1, \ldots, N)$, $\mathbf{z}$ is uniquely determined. Since $\mathbf{h}(\mathbf{x}_i)$ $(i \in N+1, \ldots, M)$ is a linear combination of $\mathbf{h}(\mathbf{x}_i)$ $(i = 1, \ldots, N)$, $a_i = c$ $(i = N+1, \ldots, M)$ only when any of $\mathbf{h}(\mathbf{x}_i)$ $(i = N+1, \ldots, M)$ is equal to one of $\mathbf{h}(\mathbf{x}_i)$ $(i = 1, \ldots, N)$. As a special case, the last two terms in (14) is zero when $N = M$. Therefore, for $M > N$, the strict inequality is considered to hold. In addition, because of the first term in (13), the coefficient matrix in (7) is positive definite.

ESANN'2008 proceedings, European Symposium on
Artificial Neural Networks - Advances in Computational Intelligence and Learning
Bruges (Belgium), 23-25 April 2008, d-side publi., ISBN 2-930307-08-0.

From (11), the rank of $M \times M$ matrix Ω_e is N for $C = \infty$. Although for finite C, Ω_e is positive definite, for large C Ω_e approaches to positive semidefinite. Thus comparing Ω_e and the coefficient matrix in (7) the latter is more stable.

3 Performance Evaluation

We compared the computational stability and time of PrLS SVRs and DuLS SVRs using the Mackey-Glass [7] (4 inputs, 500 training and 500 test data), water purification [8] (10, 241, 237), orange juice[1] (700, 150, 68), and Boston[2] problems (13, 506). For the Boston problem we used the fifth and the 14th variables as the outputs [9] and call them the Boston 5 and 14 problems, respectively. For each problem, we randomly divided the data set into two and generated 100 sets of training and test data sets. In all cases, we used RBF kernels.

For the ELS SVR we determined the parameters C and γ by fivefold cross-validation [5]. For the sparse PrLS SVR, we determined the values of C and η by fivefold cross-validation. For γ we used the values determined for the PrLS SVR. For the DuLS SVR and sparse DuLS SVR, we used the parameter values determined for the PrLS SVR and sparse PrLS SVR, respectively.

Table 1 shows the average of the absolute approximation errors (AAAE) for the test data sets, the numbers of support vectors, and training time for ELS SVRs, sparse ELS SVRs, and LS SVRs. In theory the PrLS SVR and DuLS SVR give the same results. But because of the numerical instability of the DuLS SVR, the AAAEs for the Mackey-Glass data set were different. They are shown in parentheses and '—' denotes that the solution was not obtained because the argument of the root in Cholesky factorization became negative. Except for the Mackey-Glass data set, this did not happen.

For the number of support vectors the numerals in the parentheses show the percentage of the support vectors for the sparse LS SVR against those for the LS SVR. By the sparse LS SVR the number of support vectors reduces to 43% to 77% of that of the LS SVR.

As for the training time for the ELS SVR, sparse LS SVR, and LS SVR, we measured the time for training an SVR for given parameter values and testing for the training and test data sets using a workstation (3.6GHz, 2GB memory, Linux operating system). The numeral in the parentheses shows the time for the DuLS SVR. According to our theoretical analysis, if $N \approx M$, the dual form is faster and as N decreases the tendency is reversed. From the table, with $N \approx M$ the DuLS SVR is faster than the PrLS SVR and for the sparse ELS SVRs with $N < M$, the PrLS SVR is equal to or faster.

4 Conclusions

In this paper we formulated the dual LS SVR (DuLS SVR) in the empirical feature space and analytically compared the computation cost and numerical

[1] ftp://ftp.ics.uci.edu/pub/machine-learning-databases/
[2] http://www.cs.toronto.edu/~delve/data/datasets.html

Table 1: Comparison of the averages of the absolute approximation errors and the standard deviations of the errors, support vectors, and training time.

Data	Errors			Support Vectors			Training Time		
	ELS	Sparse	LS	ELS	Sparse	LS	ELS	Sparse	LS
M.-G.	0.000300	0.000316	0.000374	495	384 (77)	500	1.74	1.11	0.42
	(—)	(0.000336)					(1.20)	(1.16)	
W. P.	0.982	0.980	0.945	241	103 (43)	241	0.19	0.07	0.07
							(0.15)	(0.12)	
O. J.	6.20	5.88	4.42	150	112 (75)	150	0.44	0.38	0.41
							(0.43)	(0.38)	
B. 5	0.0290	0.0292	0.0276	255	134 (53)	255	0.22	0.10	0.08
	±0.00156	±0.00160	±0.00181	±12	±5	±12	(0.18)	(0.15)	
B. 14	*2.36*	*2.38*	2.27	255	132 (52)	255	0.22	0.10	0.08
	±0.164	±0.153	±0.145	±12	±5	±12	(0.18)	(0.15)	

stability of the primal LS SVR (PrLS SVR) and DuLS SVR. According to the analysis, if the dimension of the empirical feature space is roughly equal to the number of the training data, DuLS SVR is faster in training, and for the smaller dimension, the PrLS SVR is faster. According to the analysis of the positive-definiteness of the coefficient matrix, the PrLS SVR is more numerically stable than the DuLS SVR for the small margin parameter value.

The computer experiment for some benchmark data sets confirmed the above results and for the small margin parameter, training of DuLS SVR became unstable and in an extreme case, the solution was not obtained.

References

[1] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*, World Scientific Publishing, 2002.

[2] J. Valyon and G. Horváth, A sparse least squares support vector machine classifier, *Proc. IJCNN 2004*, vol. 1, pp. 543–548, 2004.

[3] S. Abe, Sparse least squares support vector training in the reduced empirical feature space, *Pattern Analysis and Applications*, vol. 10, no. 3, pp. 203–214, 2007.

[4] H. Xiong, M. N. S. Swamy, and M. O. Ahmad, Optimizing the kernel in the empirical feature space, *IEEE Trans. Neural Networks*, vol. 16, no. 2, pp. 460–474, 2005.

[5] S. Abe and K. Onishi, Sparse least squares support vector regressors trained in the reduced empirical feature space, *Proc. ICANN 2007, Part II*, pp. 527–536, 2007.

[6] S. Abe, *Support Vector Machines for Pattern Classification*, Springer, 2005.

[7] R. S. Crowder, Predicting the Mackey-Glass time series with cascade-correlation learning, *Proc. 1990 Connectionist Models Summer School*, pp. 117–123, 1990.

[8] K. Baba, I. Enbutu, and M. Yoda, Explicit representation of knowledge acquired from plant historical data using neural network, *Proc. IJCNN '90*, vol. 3, pp. 155–160, 1990.

[9] D. Harrison and D. L. Rubinfeld, Hedonic prices and the demand for clean air, *J. Environmental Economics and Management*, vol. 5, pp. 81–102, 1978.