Sparse Least Squares Support Vector Regressors Trained in the Reduced Empirical Feature Space

Onishi, Kenta

Abe, Shigeo

Sparse Least Squares Support Vector Regressors Trained in the Reduced Empirical Feature Space

Shigeo Abe and Kenta Onishi

Graduate School of Engineering
Kobe University
Rokkodai, Nada, Kobe, Japan
abe@kobe-u.ac.jp
http://www2.eedept.kobe-u.ac.jp/~abe

Abstract. In this paper we discuss sparse least squares support vector regressors (sparse LS SVRs) defined in the reduced empirical feature space, which is a subspace of mapped training data. Namely, we define an LS SVR in the primal form in the empirical feature space, which results in solving a set of linear equations. The independent components in the empirical feature space are obtained by deleting dependent components during the Cholesky factorization of the kernel matrix. The independent components are associated with support vectors and controlling the threshold of the Cholesky factorization we obtain a sparse LS SVM. For linear kernels the number of support vectors is the number of input variables at most and if we use the input axes as support vectors, the primal and dual forms are equivalent. By computer experiments we show that we can reduce the number of support vectors without deteriorating the generalization ability.

1 Introduction

In a support vector machine (SVM), among training data only support vectors are necessary to represent a solution. However for a difficult classification problem with huge training data, many training data may become support vectors. Since this leads to slow classification, there are several approaches to reduce support vectors [1–3]. Keerthi et al. [2] proposed training L2 support vector machines in the primal form. The idea is to select basis vectors by forward selection and for the selected basis vectors train support vector machines by Newton's method. This process is iterated until some stopping condition is satisfied. Wu et al. [3] imposed, as a constraint, the weight vector that is expressed by a fixed number of kernel functions and solved the optimization problem by the steepest descent method.

A least squares support vector machine (LS SVM) [4, 5] 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 [4, 5], support vectors with small

absolute values of the associated dual variables are pruned and an LS SVM is retrained using the reduced training data set. This process is iterated until sufficient sparsity is realized. In [6], LS SVMs are reformulated using the kernel expansion of the square of Euclidian norm of the weight vector in the decision function. But the above pruning method is used to reduce support vectors. 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 [7], independent data in the feature space are selected from the training data, and using the selected training data the solution is obtained by least squares method using all the training data. In [8] based on the concept of the empirical feature space proposed in [9], least squares SVMs are formulated as a primal problem and by reducing the dimension of the empirical feature space, sparse LS SVMs are realized

In this paper we extend the sparse LS SVM discussed in [8] to function approximation. Namely, we define the LS support vector regressor (SVR) in the primal form in the empirical feature space. Since the empirical feature space is finite, we can train a primal LS SVM directly by solving a set of linear equations. To generate the mapping function to the empirical feature space, we need to calculate the eigenvalues and eigenvectors of the kernel matrix. Instead, we select the maximum independent components in the kernel matrix by the Cholesky factorization. The independent components are associated with support vectors and reducing the number of independent components we obtain a sparse LS SVM. For linear kernels the number of support vectors is the number of input variables at most and if we use the Euclidean axes as support vectors, the primal and dual forms are equivalent.

In Section 2, we clarify the characteristics of the empirical feature space, and in Section 3 we derive a set of linear equations for training LS SVMs in the empirical feature space and formulate sparse LS SVMs. In Section 4, we show the validity of the proposed method by computer experiments.

2 Empirical Feature Space

In this section, we summarize the characteristics of the empirical feature space.

Let the kernel be $H(\mathbf{x}, \mathbf{x}') = \mathbf{g}^T(\mathbf{x})\,\mathbf{g}(\mathbf{x})$, where $\mathbf{g}(\mathbf{x})$ is the mapping function that maps the m-dimensional vector $\mathbf{x}$ into the l-dimensional space. For the M m-dimensional data $\mathbf{x}_i$, the $M \times M$ kernel matrix $H = \{H_{ij}\}$ (i, $j = 1, \ldots M$), where $H_{ij} = H(\mathbf{x}_i, \mathbf{x}_j)$, is symmetric and positive semidefinite. Let the rank of H be $N\,(\leq M)$. Then H is expressed by

$$H = U\,S\,U^T, \tag{1}$$

where the column vectors of U are eigenvectors of H and $U\,U^T = U^T U = I_{M\times M}$, $I_{M\times M}$ is the $M \times M$ unit matrix, and $S = \mathrm{diag}\,(\sigma_1, \ldots, \sigma_N, 0, \ldots, 0)$. Here, $\sigma_i\,(> 0)$ are eigenvalues of H, whose eigenvectors correspond to the ith columns of U.

Defining the first N vectors of U as the $M \times N$ matrix P and

$$\Lambda = \mathrm{diag}\,(\sigma_1, \ldots, \sigma_N), \tag{2}$$

we can rewrite (1) as follows:

$$H = P\,\Lambda\,P^T, \tag{3}$$

where $P^T P = I_{N\times N}$ but $P\,P^T \neq I_{M\times M}$.

We must notice that if $N < M$, the determinant of H vanishes. Thus, from $H(\mathbf{x}, \mathbf{x}') = \mathbf{g}^T(\mathbf{x})\,\mathbf{g}(\mathbf{x})$, the following equation holds:

$$\sum_{i=1}^{M} a_i\,\mathbf{g}^T(\mathbf{x}_i) = 0, \tag{4}$$

where $a_i (i = 1, \ldots, M)$ are constant and some of them are nonzero. Namely, if $N < M$, the mapped training data $\mathbf{g}(\mathbf{x}_i)$ are linearly dependent. And if $N = M$, they are linearly independent and there are no non-zero a_i that satisfy (4).

Now we define the mapping function that maps the m-dimensional vector $\mathbf{x}$ into the N-dimensional space called *empirical feature space* [9]:

$$\mathbf{h}(\mathbf{x}) = \Lambda^{-1/2}\,P^T\,(H(\mathbf{x}_1, \mathbf{x}), \ldots, H(\mathbf{x}_M, \mathbf{x}))^T. \tag{5}$$

We define the kernel associated with the empirical feature space by

$$H_{\mathrm{e}}(\mathbf{x}, \mathbf{x}') = \mathbf{h}^T(\mathbf{x})\,\mathbf{h}(\mathbf{x}'). \tag{6}$$

Clearly, the dimension of the empirical feature space is N. Namely, the empirical feature space is spanned by the linearly independent mapped training data.

We can prove that the kernel for the empirical feature space is equivalent to the kernel for the feature space if they are evaluated using the training data. Namely [9, 8],

$$H_{\mathrm{e}}(\mathbf{x}_i, \mathbf{x}_j) = H(\mathbf{x}_i, \mathbf{x}_j) \qquad \text{for} \quad i, j = 1, \ldots, M. \tag{7}$$

The relation given by (7) is very important because a problem expressed using kernels can be interpreted, without introducing any approximation, as the problem defined in the associated empirical feature space. The dimension of the feature space is sometimes very high or infinite. But the dimension of the empirical feature space is the number of training data at most. Thus, instead of analyzing the feature space, we only need to analyze the associated empirical feature space.

3 Least Squares Support Vector Regressors

3.1 Training in the Empirical Feature Space

The LS SVR in the feature space is trained by minimizing

$$\frac{1}{2}\,\mathbf{w}^T\mathbf{w} + \frac{C}{2}\sum_{i=1}^{M}\xi_i^2 \tag{8}$$

subject to the equality constraints:

$$\mathbf{w}^T\mathbf{g}(\mathbf{x}_i) + b = y_i - \xi_i \quad \text{for} \quad i = 1, \ldots, M, \tag{9}$$

where y_i is the output for input $\mathbf{x}_i$, $\mathbf{w}$ is the l-dimensional vector, b is the bias term, $\mathbf{g}(\mathbf{x})$ is the l-dimensional vector that maps the m-dimensional vector $\mathbf{x}$ into the feature space, ξ_i is the slack variable for $\mathbf{x}_i$, and C is the margin parameter.

Introducing the Lagrange multipliers $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_M)^T$ into (8) and (9), we obtain the dual form as follows:

$$\Omega\boldsymbol{\alpha} + \mathbf{1}b = \mathbf{y}, \tag{10}$$

$$\mathbf{1}^T\boldsymbol{\alpha} = 0, \tag{11}$$

where $\mathbf{1}$ is the M-dimensional vector: $\mathbf{1} = (1, \ldots, 1)^T$ and

$$\Omega_{ij} = \mathbf{g}^T(\mathbf{x}_i)\,\mathbf{g}(\mathbf{x}_j) + \frac{\delta_{ij}}{C}, \quad \delta_{ij} = \begin{cases} 1 & \text{for} \quad i = j, \\ 0 & \text{for} \quad i \neq j, \end{cases} \quad \mathbf{y} = (y_1, \ldots, y_M)^T. \tag{12}$$

Setting $H(\mathbf{x}, \mathbf{x}') = \mathbf{g}^T(\mathbf{x})\,\mathbf{g}(\mathbf{x}')$, we can avoid the explicit treatment of variables in the feature space.

The original minimization problem is solved by solving (10) and (11) for $\boldsymbol{\alpha}$ and b as follows. Because of $1/C\,(> 0)$ in the diagonal elements, Ω is positive definite. Therefore,

$$\boldsymbol{\alpha} = \Omega^{-1}(\mathbf{y} - \mathbf{1}\,b). \tag{13}$$

Substituting (13) into (11), we obtain

$$b = (\mathbf{1}^T\Omega^{-1}\mathbf{1})^{-1}\mathbf{1}^T\Omega^{-1}\mathbf{y}. \tag{14}$$

Thus, substituting (14) into (13), we obtain $\boldsymbol{\alpha}$. We call the LS SVR obtained by solving (13) and (14) *dual LS SVR*.

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

$$Q(\mathbf{v}, \boldsymbol{\xi}, b) = \frac{1}{2}\,\mathbf{v}^T\mathbf{v} + \frac{C}{2}\sum_{i=1}^{M}\xi_i^2 \tag{15}$$

subject to the equality constraints:

$$\mathbf{v}^T\mathbf{h}(\mathbf{x}_i) + b = y_i - \xi_i \quad \text{for} \quad i = 1, \ldots, M, \tag{16}$$

where $\mathbf{v}$ is the N-dimensional vector and $\mathbf{h}(\mathbf{x})$ is the N-dimensional vector that maps the m-dimensional vector $\mathbf{x}$ into the empirical feature space.

Substituting (16) into (15), we obtain

$$Q(\mathbf{v}, \boldsymbol{\xi}, b) = \frac{1}{2}\,\mathbf{v}^T\mathbf{v} + \frac{C}{2}\sum_{i=1}^{M}(y_i - \mathbf{v}^T\mathbf{h}(\mathbf{x}_i) - b)^2. \tag{17}$$

Equation (17) is minimized when the following equations are satisfied:

$$\frac{\partial Q(\mathbf{v}, \boldsymbol{\xi}, b)}{\partial \mathbf{v}} = \mathbf{v} - C \sum_{i=1}^{M} (y_i - \mathbf{v}^T \mathbf{h}(\mathbf{x}_i) - b)\, \mathbf{h}(\mathbf{x}_i) = \mathbf{0} \tag{18}$$

$$\frac{\partial Q(\mathbf{v}, \boldsymbol{\xi}, b)}{\partial b} = -C \sum_{i=1}^{M} (y_i - \mathbf{v}^T \mathbf{h}(\mathbf{x}_i) - b) = 0. \tag{19}$$

From (19)

$$b = \frac{1}{M} \sum_{i=1}^{M} (y_i - \mathbf{v}^T \mathbf{h}(\mathbf{x}_i)). \tag{20}$$

Substituting (20) into (18), we obtain

$$\left(\frac{1}{C} + \sum_{i=1}^{M} \mathbf{h}(\mathbf{x}_i)\, \mathbf{h}^T(\mathbf{x}_i) - \frac{1}{M} \sum_{i,j=1}^{M} \mathbf{h}(\mathbf{x}_i)\, \mathbf{h}^T(\mathbf{x}_j) \right) \mathbf{v} = \sum_{i=1}^{M} y_i\, \mathbf{h}(\mathbf{x}_i) - \frac{1}{M} \sum_{i,j=1}^{M} y_i\, \mathbf{h}(\mathbf{x}_j). \tag{21}$$

Therefore, from (21) and (20) we obtain $\mathbf{v}$ and b. We call the LS SVR obtained by solving (21) and (20) *primal LS SVR*.

3.2 Sparse Least Squares Support Vector Regressors

In training LS SVRs in the empirical feature space we need to transform input variables into the variables in the empirical feature space by (5). But this is time consuming. Thus, instead of using (5), we select independent training data that span the empirical feature space. Let the first $M' (\leq M)$ training data be independent in the empirical feature space. Then, instead of (5), we use the following equation:

$$h(\mathbf{x}) = (H(\mathbf{x}_1, \mathbf{x}), \ldots, H(\mathbf{x}_{M'}, \mathbf{x}))^T \tag{22}$$

By this formulation, $\mathbf{x}_1, \ldots, \mathbf{x}_{M'}$ becomes support vectors. Thus, support vectors do not change even if the margin parameter changes. And the number of support vectors is the number of selected independent training data that span the empirical feature space. Thus for linear kernels, the number of support vectors is the number of input variables at most. The selected training data span the empirical feature space but the coordinates are different from those of the empirical feature space, namely those given by (5). Thus, the solution is different from that using (5) because SVRs are not invariant for the linear transformation of input variables [10]. As the computer experiments in the following section show, this is not a problem if we select kernels and the margin parameter properly.

We use the Cholesky factorization in selecting independent vectors [10]. Let H be positive definite. Then H is decomposed by the Cholesky factorization into

$$H = L\,L^T, \tag{23}$$

where L is the regular lower triangular matrix and each element L_{ij} is given by

$$L_{op} = \frac{H_{op} - \sum_{n=1}^{p-1} L_{pn}\,L_{on}}{L_{pp}} \qquad \text{for} \quad o = 1, \ldots, M, \quad p = 1, \ldots, o-1, \tag{24}$$

$$L_{aa} = \sqrt{H_{aa} - \sum_{n=1}^{a-1} L_{an}^2} \qquad \text{for} \quad a = 1, 2, \ldots, M. \tag{25}$$

Here, $H_{ij} = H(\mathbf{x}_i, \mathbf{x}_j)$.

Then during the Cholesky factorization, if the diagonal element is smaller than the prescribed value $\eta\,(> 0)$:

$$H_{aa} - \sum_{n=1}^{a-1} L_{an}^2 \leq \eta, \tag{26}$$

we delete the associated row and column and continue decomposing the matrix. The training data that are not deleted in the Cholesky factorization are independent. If no training data are deleted, the training data are all independent in the feature space.

The above Cholesky factorization can be done incrementally [10, 11]. Namely, instead of calculating the full kernel matrix in advance, if (26) is not satisfied, we overwrite the ath column and row with those newly calculated using the previously selected data and $\mathbf{x}_{a+1}$. Thus the dimension of L is the number of selected training data, not the number of training data.

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*.

If we use linear kernels we do not need to select independent variables. Instead of (22), we use

$$h(\mathbf{x}) = \mathbf{x}. \tag{27}$$

This is equivalent to using $\mathbf{e}_i$ $(i = 1, \ldots, m)$, where $\mathbf{e}_i$ are the basis vectors in the input space, in which the ith element is 1 and other elements 0. We call the primal LS SVR using (27) *primal LS SVR with orthogonal support vectors (OSV)*, and the primal LS SVR with selected independent training data *LS SVR with non-orthogonal support vectors (NOSV)*.

4 Performance Evaluation

We compared the generalization ability of primal, sparse, and dual LS SVRs using the Mackey-Glass [12], water purification [13], orange juice[1], and Boston[2] problems listed in Table 1. For the first three problems, one set of training data set and test data set are given. But the Boston problem is not divided into training and test data sets. As discussed in [14], we used the fifth and the 14th variables as the outputs and call the problems the Boston 5 and 14 problems, respectively. For each problem, we randomly divided the data set into two with almost equal sizes and generated 100 sets of training and test data sets.

For primal LS SVRs we set $\eta = 10^{-9}$ and for sparse LS SVRs, we selected the value of η from $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 0.05\}$ by fivefold cross-validation.

In all studies, we normalized the input ranges into $[0, 1]$ and used linear and RBF kernels. We determined the parameters C, γ for RBF kernels, and η by fivefold cross-validation; the value of C was selected from among $\{1, 10, 50, 100, 500, 1000, 2000, 3000, 5000, 8000, 10000, 50000, 10^5, \ldots 10^{14}\}$, the value of γ from among $\{0.01, 0.1, 1, 5, 10, 15, 20, 30\}$.

Table 1. Benchmark data sets.

Data	Inputs	Train.	Test
Mackey-Glass	4	500	500
Water Purification	10	241	237
Orange Juice	700	150	68
Boston	13	506	—

We determined the margin parameters and kernel parameters by fivefold cross-validation. For RBF kernels we determined the optimal values of γ and C for primal and dual LS SVRs. Then setting the optimal values of γ determined by cross-validation for primal LS SVRs, we determined the optimal values of η and C for sparse LS SVRs.

For the Boston 5 and 14 problems we performed cross-validation using the first five training data sets. For RBF kernels we performed cross-validation for each training data set changing the values of γ and C and selected the value of γ whose associated average of the absolute approximation errors (AAAE) is the smallest. Then we took the median among five γ values as the best value of γ. Then, again we took the median among the best values of C for the best γ associated with the five training data sets. Then, for the best values of γ and C, we trained the SVR for the 100 training data sets and calculated the AAAEs

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

and the standard deviation of the approximation errors for the test data sets. For linear kernels we took the median of the best values of C associated with the first five training data sets.

For sparse LS SVRs, for each value of η we determined the best value of C. We selected the largest value of η whose associated AAAE for the validation data set is comparable with that for $\eta = 10^{-9}$.

Table 2 lists the parameters obtained according to the preceding procedure. The margin parameters for the OSV are the same with the dual LS SVR except for the Mackey-Glass problem. Except for those of OSV, the values of C for the primal LS SVRs are sometimes larger than those for the dual LS SVRs. And the values of γ for water purification, orange juice, and Boston 5 problems are different.

Table 2. Parameter setting for linear and RBF kernels. The parameters were determined by fivefold cross-validation.

Data	Linear			RBF					
	OSV	NOSV	Dual	Primal		Sparse		Dual	
	C	C	C	γ	C	η	C	γ	C
Mackey-Glass	10^{13}	10^5	10^7	30	10^9	10^{-6}	10^9	30	10^9
Water Purification	10	10^6	10	20	50	0.05	50	30	10
Orange Juice	100	10^5	100	10	10^{10}	10^{-4}	10^8	0.01	10^7
Boston 5	1	1	1	10	500	10^{-2}	500	15	50
Boston 14	1	10^4	1	10	3000	10^{-2}	3000	10	50

Table 3 shows AAAEs for the test data sets. For Boston 5 and 14 problems the standard deviations are also shown. For Boston 5 and 14, we statistically analyzed the average and standard deviations with the significance level of 0.05. Numerals in italic show that they are statistically inferior among linear or RBF kernels.

For linear kernels, as theory tells us OSV and dual LS SVR show the same results. The AAAE for the orange juice problem by NOSV is smaller than that of the dual LS SVR (OSV) but the AAAE for the Boston 14 problem by NOSV is statistically inferior. The reason for the Boston 14 problem is that for the dual LS SVR (OSV) the best values of C are the same for the five training data sets but for NOSV the best values ranged from 50 to 10^8. Thus, the median of the best values was not best even for the first five files. For RBF kernels the AAAEs for the Mackey-Glass, water purification, and Boston 5 problems by the primal LS SVR and the sparse LS SVR are worse than by the dual LS SVR but there is not much difference.

Table 3. Comparison of the averages of the absolute approximation errors and the standard deviations of the errors for the linear and RBF kernels.

Data	Linear			RBF		
	OSV	NOSV	Dual	Primal	Sparse	Dual
Mackey-G.	0.0804	0.0804	0.0804	0.000508	0.000531	0.000323
Water P.	1.20	1.20	1.20	0.982	0.980	0.962
Orange J.	4.31	4.09	4.31	3.94	4.02	3.99
Boston 5	0.0425	0.0429	0.0425	*0.0290*	*0.0292*	0.0276
	±0.00160	±0.00169	±0.00160	±0.00156	±0.00160	±0.00181
Boston 14	3.41	*3.47*	3.41	2.36	2.38	2.27
	±0.146	±0.148	±0.146	±0.164	±0.158	±0.145

Table 4 lists the number of support vectors for linear and RBF kernels. The numerals in the parentheses show the percentage of the support vectors for the sparse LS SVR against those for the dual LS SVR. For OSV we used all the input variables. Thus, the number of support vectors is the number of input variables. But for NOSV, we selected independent data. For the orange juice problem the support vectors were reduced from 700 to 120. By setting $\eta = 10^{-3}$ and $C = 10^5$ we could still reduce the number to 41 with AAAE of 4.16. Thus even if the number of input variables is larger than that of the training data, we can considerably reduce the number of support vectors by NOSV. If the number of input variables is much smaller than that of the training data, we can reduce support vectors considerably using OSV or NOSV.

For RBF kernels, the number of support vectors for primal solutions is the number of training data at most. By sparse LS SVR the reduction ratio was 42% to 77%.

Table 4. Comparison of support vectors for the linear and RBF kernels.

Data	Linear			RBF		
	OSV	NOSV	Dual	Primal	Sparse	Dual
Mackey-G.	4	5 (1)	500	498	384 (77)	500
Water P.	10	10 (4)	241	241	103 (43)	241
Orange J.	700	120 (80)	150	150	63 (42)	150
Boston 5	13±0	13±0.2 (5)	255±12	255±12	134±5 (53)	255±12
Boston 14	13±0	13±0.1 (5)	255±12	255±12	132±5 (52)	255±12

5 Conclusions

In this paper we formulated the primal LS SVR in the empirical feature space and derived the set of linear equations to train the primal LS SVRs. Then we proposed the sparse LS SVR restricting the dimension of the empirical feature space controlling the threshold of the Cholesky factorization. According to the computer experiments, for all the data sets tested, the sparse solutions could realize sparsity while realizing generalization ability comparable with that of primal and dual solutions.

References

1. C. J. C. Burges. Simplified support vector decision rules. In L. Saitta, editor, *Machine Learning, Proceedings of the Thirteenth International Conference (ICML '96)*, pages 71–77. Morgan Kaufmann, San Francisco, 1996.
2. S. S. Keerthi, O. Chapelle, and D. DeCoste. Building support vector machines with reduced classifier complexity. *Journal of Machine Learning Research*, 7:1493–1515, 2006.
3. M. Wu, B. Schölkopf, and G Bakir. A direct method for building sparse kernel learning algorithms. *Journal of Machine Learning Research*, 7:603–624, 2006.
4. J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.
5. J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least Squares Support Vector Machines*. World Scientific Publishing, Singapore, 2002.
6. G. C. Cawley and N. L. C. Talbot. Improved sparse least-squares support vector machines. *Neurocomputing*, 48:1025–1031, 2002.
7. J. Valyon and G. Horvath. A sparse least squares support vector machine classifier. In *Proceedings of International Joint Conference on Neural Networks (IJCNN 2004)*, volume 1, pages 543–548, Budapest, Hungary, 2004.
8. S. Abe. Sparse least squares support vector training in the reduced empirical feature space. *Pattern Analysis and Applications (accepted)*.
9. H. Xiong, M. N. S. Swamy, and M.O. Ahmad. Optimizing the kernel in the empirical feature space. *IEEE Transactions on Neural Networks*, 16(2):460–474, 2005.
10. S. Abe. *Support Vector Machines for Pattern Classification*. Springer-Verlag, New York, 2005.
11. K. Kaieda and S. Abe. KPCA-based training of a kernel fuzzy classifier with ellipsoidal regions. *International Journal of Approximate Reasoning*, 37(3):145–253, 2004.
12. R. S. Crowder. Predicting the Mackey-Glass time series with cascade-correlation learning. In *Proceedings of 1990 Connectionist Models Summer School*, pages 117–123, Carnegie Mellon University, 1990.
13. K. Baba, I. Enbutu, and M. Yoda. Explicit representation of knowledge acquired from plant historical data using neural network. In *Proceedings of 1990 IJCNN International Joint Conference on Neural Networks*, volume 3, pages 155–160, San Diego, 1990.
14. D. Harrison and D. L. Rubinfeld. Hedonic prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5:81–102, 1978.