Journal of Southeast University (English Edition) Vol. 22, No. 2, pp. 228 – 233 June 2006 ISSN 1003—7985

Identification of dynamic systems using support vector regression neural networks

Li Jun Liu Junhua

(School of Electrical Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: A novel adaptive support vector regression neural network (SVR-NN) is proposed, which combines respectively merits of support vector machines and a neural network. First, a support vector regression approach is applied to determine the initial structure and initial weights of the SVR-NN so that the network architecture is easily determined and the hidden nodes can adaptively be constructed based on support vectors. Furthermore, an annealing robust learning algorithm is presented to adjust these hidden node parameters as well as the weights of the SVR-NN. To test the validity of the proposed method, it is demonstrated that the adaptive SVR-NN can be used effectively for the identification of nonlinear dynamic systems. Simulation results show that the identification schemes based on the SVR-NN give considerably better performance and show faster learning in comparison to the previous neural network method.

Key words: support vector regression; neural network; system identification; robust learning algorithm; adaptability

Multilayer feedforward neural networks (MFNN) provide good approximations when applied to highly nonlinear and complex systems[1]. However, the network is hindered by problems associated with weight optimization such as slow learning and local minimization. Furthermore, good generalization results are obtained only if the structure of the network is suitably chosen. Therefore, selecting the best structure of the neural networks is an important problem.

At present, the support vector machine (SVM) may tackle classification and regression problems and it has been applied as an alternative to conventional artificial neural networks[2-3]. To guarantee the generalization ability on the process of learning, the SVM utilized the natural learning algorithms i. e. structure risk minimization principle[4]. The solution of the SVM is unique, optimum and absent from local minimums under some limited conditions compared to traditional neural networks. But selection of the kernel parameters of the SVM is very flexible. We determine the kernel parameters by experimental methods such as cross-validation etc[5].

In order to solve the above problems, a novel adaptive support vector regression neural network (SVR-NN) is proposed. In the proposed method, the initial structure of the SVR-NN is obtained by a support vector regression (SVR). Because an SVR approach is equivalent to solving a linear constrained quadratic programming problem under a fixed structure of the SVR, the number of hidden nodes and initial weights of the SVR-NN are easily obtained. Then, an annealing robust learning process is used as the learning algorithm to adjust these hidden node parameters as well as the weights of the SVR-NN[6-7].

To encourage sparseness so as to obtain good network structure adaptively, an SVR approach with the ε-insensitive loss function can approximate the unknown functions by constrained minimization for a given precision level of the modeling error. In the constrained minimization, kernels corresponding to data points that are within the error bounds are removed. The SVR is formed by the retained kernels, and the data points associated with the retained kernels are referred to the support vectors[2,8].

Since ε-SVR[2,3] or ν-SVR[9] provides an estimated function with error bounds, the initial structure of the SVR-NN can be well obtained by the SVR approach that just provides better initialization. Furthermore, the kernels of the SVR are similar to the basis functions of the radial basis function (RBF) network with scatter partitioning. Hence, an annealing robust learning algorithm is further applied to fine tune these parameters of hidden nodes and weights of the networks.

The effectiveness of the proposed method is verified by three simulation examples.

Received 2005-10-12.

Foundation item: The National Natural Science Foundation of China (No. 60276037).

Biographies: Li Jun (1969—), male, graduate, lijun691201@yahoo.com.cn; Liu Junhua (1938—), female, professor, Junhliu@mail.xjtu.edu.cn.

1 Support Vector Regression Neural Networks

1.1 SVMs for regression

In general regression formulation and unknown function estimation, the goal is to estimate an unknown real-valued function based on a finite number set of samples $(\boldsymbol{x}_i, y_i)$ $(i = 1, 2, \ldots, l)$, where the q-dimensional input $\boldsymbol{x} \in \mathbf{R}^q$ and the output $y \in \mathbf{R}$.

In the SVR approach, the SVM provides a means of solving nonlinear modeling problems by transforming them to linear ones in some high-dimensional feature space which is related to the input data space via a prior selected nonlinear mapping and the quality of estimation is measured by the loss function. A new type of loss function called the ε-insensitive loss function[2] is an attractive choice in the implementation of the SVR since selecting the value of ε controls the numbers of support vectors, which introduces sparseness to the final model solution.

The linear ε-insensitive loss function is defined as

$$L_\varepsilon(y, f(\boldsymbol{x}, \boldsymbol{w})) = \max(0, y - f(\boldsymbol{x}, \boldsymbol{w}) \mid -\varepsilon) \quad (1)$$

A linear model is then constructed in the feature space and the linear model $f(\boldsymbol{x}, \boldsymbol{w})$ is given by

$$f(\boldsymbol{x}, \boldsymbol{w}) = \sum_{k=1}^{m} w_k \phi_k(\boldsymbol{x}) + b \quad (2)$$

where $\phi(\cdot)$ denotes a set of nonlinear mapping, and b is the "bias" term. The inclusion of the constant term b depends on whether the bias is regularized or unregularized.

The weight vector $\boldsymbol{w}$ and threshold $\boldsymbol{b}$ for the linear ε-insensitive support vector regression are chosen to optimize the following problem:

$$\min_{w, \xi, \xi^*} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \quad (3)$$

$$\text{subject to } \begin{cases} y_i - f(\boldsymbol{x}_i, \boldsymbol{w}) \leqslant \varepsilon + \xi_i^* \\ f(\boldsymbol{x}_i, \boldsymbol{w}) - y_i \leqslant \varepsilon + \xi_i \\ \xi_i, \xi_i^* \geqslant 0 \qquad i = 1, 2, \ldots, l \end{cases}$$

The corresponding dual problem can be derived using the standard techniques

$$\max \sum_{i=1}^{l}(\hat{\alpha}_i - \alpha_i) y_i - \varepsilon \sum_{i=1}^{l}(\hat{\alpha}_i - \alpha_i) - \frac{1}{2}\sum_{i,j=1}^{l}(\hat{\alpha}_i - \alpha_i)(\hat{\alpha}_j - \alpha_j) K(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad (4)$$

$$\text{subject to } \sum_{i=1}^{l}(\alpha_i - \hat{\alpha}_i) = 0; \alpha_i, \hat{\alpha}_i \in [0, C]; \ i = 1, 2, \ldots, l$$

Considering the corresponding Karush-Kuhn-Tucker (KKT) complementarily conditions, we solve the above quadratic optimization problem, the corresponding regression function is also given by

$$f(\boldsymbol{x}) = \sum_{i=1}^{p}(\alpha_i - \hat{\alpha}_i) K(\boldsymbol{x}_i, \boldsymbol{x}) + b \quad (5)$$

where b is chosen so that $f(\boldsymbol{x}_i) - y_i = -\varepsilon$ for any i with $0 < \alpha_i - \hat{\alpha}_i < C$.

Note that p is the number of some of $(\alpha_i - \alpha_i^*)$'s which are not zeros and the corresponding vectors $\boldsymbol{x}_i$'s are called the support vectors.

To automatically tune the size ε of the tube, one of the attractive features of the ε-SVR is the ability to reformulate the problem so that the regularization parameter specifies the fraction of support vectors in the so-called ν-support vector machine[9].

1.2 SVR-NN model

The output of the SVR approach is a linear function of the weights and kernels. The weights and the structure of the SVR are obtained simultaneously by constrained minimization for a given precision level of the modeling error, avoiding inner products being performed in the feature space, since via the theory of reproducing kernel hibert space (RKHS)[10-12], they have an equivalent kernel in the input space, provided certain conditions hold.

The selection of support vectors in the SVR approach can be viewed as the definition of the hidden layer neurons of an RBF neural network, and the number of hidden neurons corresponds to the number of support vectors selected in the feature space. Hence, the resulting SVR network can be represented as a novel adaptive SVR-NN. The weights and center vector and width of the hidden node can also be optimized by learning algorithms of neural networks to overcome the biased problem of the SVR. By selecting different kernel functions and using the kernel trick, we may construct different types of the SVR-NN.

For an ordinary neural network, the feature space remains hidden and is never explored, but the main attribute of the SVR-NN gives new insight into neural networks. The architecture of the adaptive SVR-NN based on the support vectors is shown in Fig. 1. The generalization of different kinds of the adaptive SVR-NN may be obtained by choosing the appropriate kernel function which fulfills the Mercer theorem[2-3] such as Polymonial function, B-spline function etc.

In this paper, we select the Gaussian kernel function in the SVR-NN model, then Eq. (5) can be rewritten as

$$f(\boldsymbol{x}) = \sum_{j=1}^{p}(\alpha_j - \hat{\alpha}_j)\exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{v}_j\|^2}{2\gamma^2}\right) + b = \sum_{j=1}^{p} w_j' \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{v}_j\|^2}{2\gamma^2}\right) + w_{p+1}' \quad (6)$$

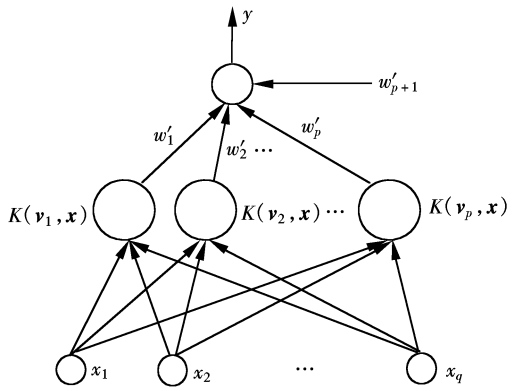

Fig. 1 The architecture of the proposed SVR-NN based on support vectors

From Fig. 1, it is shown that the above SVR-NN given by Eq. (5) can be regarded as a two-layer neural network linear in its weights. It is intuitive to reformulate it as an RBF network using normalized basis functions[8, 12].

After the support vectors of the SVR-NN are obtained by ε-SVR or ν-SVR, some learning algorithms such as the least-squares method or the gradient decent learning algorithm may be applied to adjust the parameters of the Gaussian function and the synaptic weights for an improved approximated performance. However, most of the traditional neural network approaches may be easily affected by outliers. An advanced annealing robust learning approach is proposed here to overcome these problems of the traditional neural network approaches for function approximation.

2 Training of Adaptive SVM-NN

To train the adaptive SVR-NN with Gaussian basis functions, the annealing robust learning algorithm is proposed to solve the overfitting problems in neural network learning. A cost function for the SVR-NN is defined here[6]:

$$E = \sum_{i=1}^{l} \sigma(e_i(t); \mu(t)) = \sum_{i=1}^{l} \frac{\mu(t)}{2} \ln\left[1 + \frac{e_i^2(t)}{\mu(t)}\right] \tag{7}$$

where t is the epoch number; $e_i(t) = y_i - f(\boldsymbol{x}_i)$ is the error between the i-th desired output and the i-th output of the SVR-NN at epoch t; $\mu(t)$ is a deterministic annealing schedule acting like the cut-off points; and $\sigma(\cdot)$ is a logistic loss function.

The initial j-th hidden node center vector of the SVR-NN is the j-th support vector. Refer to Fig. 1. We have $\boldsymbol{v}_j = \{v_{j,1}, v_{j,2}, \ldots, v_{j,q}\}^{\mathrm{T}}$, and the initial width vector $\{\gamma_1, \gamma_2, \ldots, \gamma_p\}^{\mathrm{T}}$ of the hidden node is taken as the width γ of the Gaussian kernel function.

Based on the gradient-decent kind of learning algorithms, the weights w'_j, the centers $\boldsymbol{v}_j$ and width γ_j of the hidden layer node are updated as

$$\Delta w'_j = -\eta \frac{\partial E_j}{\partial w'_j} = \eta \rho(e_j; \mu) \exp\left(-\frac{\| \boldsymbol{x} - \boldsymbol{v}_j \|^2}{2\gamma_j^2}\right) \tag{8}$$

$$\Delta v_{n,j} = -\eta \frac{\partial E_j}{\partial v_j} = \eta \rho(e_j; \mu) w'_j \frac{x_n - v_{n,j}}{\gamma_j^2} \cdot \exp\left(-\frac{\| \boldsymbol{x} - \boldsymbol{v}_j \|^2}{2\gamma_j^2}\right) \tag{9}$$

$$\Delta \gamma_j = -\eta \frac{\partial E_j}{\partial \gamma_j} = \mu \rho(e_j; \mu) w'_j \frac{\| \boldsymbol{x} - \boldsymbol{v}_j \|^2}{\gamma_j^3} \cdot \exp\left(-\frac{\| \boldsymbol{x} - \boldsymbol{v}_j \|^2}{2\gamma_j^2}\right) \tag{10}$$

where $\rho(e_j; \mu) = \dfrac{\partial \sigma(e_j(t); \mu(t))}{\partial e_j(t)} = \dfrac{e_j(t)}{1 + e_j^2(t)/\mu(t)}$, η is a learning constant, and $n = 1, 2, \ldots, q$.

Update the weights and hidden node parameters of the SVR-NN according to

$$w'^{(k+1)}_j = w'^{(k)}_j + \eta \delta w'_j + \alpha(w'^{(k)}_j - w'^{(k-1)}_j) \tag{11}$$

$$\boldsymbol{v}_j^{(k+1)} = \boldsymbol{v}_j^{(k)} + \eta \delta v_j + \alpha(\boldsymbol{v}_j^{(k)} - \boldsymbol{v}_j^{(k-1)}) \tag{12}$$

$$\gamma_j^{(k+1)} = \gamma_j^{(k)} + \eta \delta \gamma_j + \alpha(\gamma_j^{(k)} - \gamma_j^{(k-1)}) \tag{13}$$

Because of the existence of the logistic cost function, $\rho(e_j; \mu)$ is usually called the influence function. They have a great impact on the approximation results when outliers exist. When the decay is too quick, the approximation of the majority data may not have enough time to converge and the training data may mostly be degraded. If the decay is too slow, the robust learning algorithm may not be timely enough to discriminate against those outliers before overfitting occurs.

The suitable annealing schedule $\mu(t) = A/t$ is founded experimentally to achieve the best performance in the annealing algorithm, where A is a constant. To determine the values of annealing schedule $\mu(t) = A/t$ for any t epoch in our experiments, A is set as $2\max\{|e_i|_{\text{initial}}\}$ according to Ref. [6].

3 Application to Identification of Nonlinear Dynamic Systems

To illustrate the performance of the presented adaptive SVRNN approach, three numerical instances are provided. The accuracy of the model is assessed using root mean square error (RMSE). The learning rate η is chosen as 0.04 and the momentum coefficient is 0.6 in these instances.

Example 1 The example is taken from Narendra and Parthasarathy[13] in which the plant to be identified is given by the second-order nonlinear difference equation,

$$y_{k+1} = \frac{y_k y_{k-1}(y_k + 2.5)}{1 + y_k^2 + y_{k-1}^2} + u_k \qquad (14)$$

From the plant model (14), 500 simulated data points are generated. The first 400 data points are obtained by assuming a random input signal uniformly distributed in the interval $[-1, 1]$, and the last 100 data points are obtained by using a sinusoid input signal, i. e., $u_k = \sin(2\pi k/25)$. We use the first 400 data points to build an identifier model based on the SVR-NN and the performance of the SVR-NN model is verified using the remaining 100 data points.

y_k, y_{k-1} and u_k are selected as the input variables and the parameters in ε-SVR are set as $\varepsilon = 0.15$, $C = 3$, the Gaussian kernel function with $\gamma = 1$. First, an initial structure of the adaptive SVR-NN with the hidden nodes (i. e. the number of support vectors) is obtained as 36 by an ε-SVR approach. Hence, the SVR-NN model consists of 36 radial basis functions with three-dimensional premises and the resulting structure of the SVR-NN model is $NN_{3\text{-}36\text{-}1}$.

These parameters of the SVR-NN are then adjusted by the robust learning algorithm. The maximum number of the training epochs needed is set as 60 and the error convergence curve is also shown in Fig. 2. The RMSE is 0.028 9 in the training region and the RMSE is 0.062 5 in the test region, respectively. The final result of the SVR-NN in the test set is shown in Fig. 3.

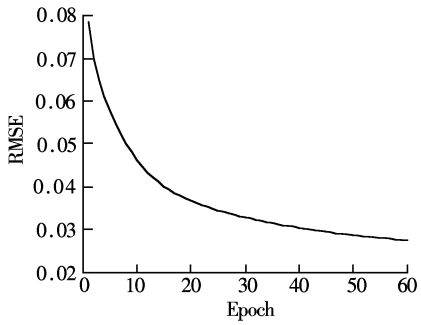

Fig. 2 Error convergence curve of adaptive SVR-NN

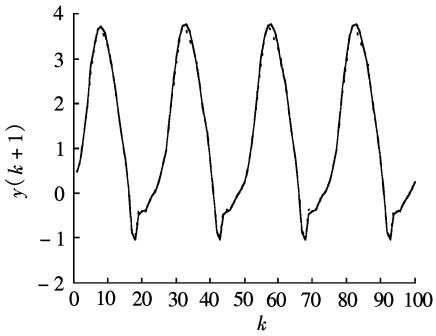

Fig. 3 Output of plant (solid line) and identification model based on SVR-NN (dashed line)

Obviously, it can be seen that the output from the identified model attains a fairly good match with that of the actual model in the test set. Hence, the adaptive SVR-NN shows good approximation accuracy for the nonlinear system model.

In the example, only 400 data points have been used to identify the model in the training data region; while in Ref. [13], 10^5 data points have been used to identify a corresponding multilayer feedforward neural network model, which has two hidden layers. If the number of data points used to build the model is increased, the performance of the identified model based on the SVR-NN can be further improved.

Example 2 Consider the gas furnace data of Box-Jenkins[14]. The data is well known and frequently used as a benchmark instance to illustrate identification algorithms. The data set consists of 296 pairs of input-output measurements taken from a laboratory furnace with a sampling time of 9 s. The process input u_k is the methane flow rate and the output y_{k+1} is the percentage of CO_2 in the off gas, respectively. We use a total of 296 data points: the first 250 for training and the remaining 46 for checking. A number of researchers concluded that a structure of the dynamic model for this system is

$$y_{k+1} = f(y_k, u_{k-3}) \qquad (15)$$

We apply SVR-NN to the above dynamic model and the parameters in ε-SVR are set as $\varepsilon = 0.15$, $C = 1$, the Gaussian kernel function with $\gamma = 1$. First, an initial structure of the adaptive SVR-NN with the hidden nodes (i. e. the number of support vectors) is obtained as nine by an ε-SVR approach. Therefore, the resulting structure of the SVR-NN model is $NN_{2\text{-}9\text{-}1}$.

All the parameters of the SVR-NN are then adjusted by the robust learning algorithm. When the number of training epochs needed is 160, the RMSE is 0.044 in the training region and the RMSE is 0.118 in the test region, respectively. The corresponding training error convergence curve is also shown in Fig. 4 and the final result of the SVR-NN in the whole data set is shown in Fig. 5.

It can be seen from Fig. 5 that the performance is also very good. The approximation power of the model can be appreciated if we compare the achieved modeling performance (MSE) with other results[15].

Example 3 In the example, the input is seen to occur nonlinearly in the difference equation describing the plant. The plant has the form[13]

$$y_{k+1} = \frac{y_k}{1 + y_k^2} + u_k^3 \qquad (16)$$

A series-parallel identification model based on

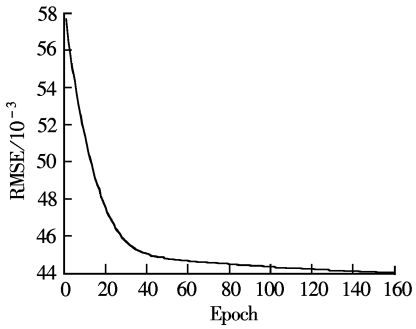

Fig. 4 Error convergence curve of adaptive SVR-NN

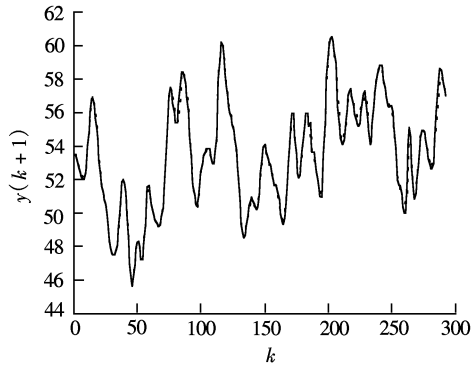

Fig. 5 Output of the actual gas furnace model (solid line) and the identified model (dashed line)

the SVR-NN can be applied, which is described as

$$\hat{y}_{k+1} = \mathrm{NN}[y_k, y_{k-1}, y_{k-2}, u_k, u_{k-1}] \qquad (17)$$

Hence, the adaptive SVR-NN model has five input variables and one output. From the plant model (16), 1 600 simulated data points are generated. The first 800 data points are obtained by assuming a random input signal uniformly distributed in the interval $[-1, 1]$, and the last 800 data points are obtained by using a sinusoid input signal, i. e. ,

$$u_k = \begin{cases} \sin\dfrac{2\pi k}{250} & k \leqslant 500 \\ 0.8\sin\dfrac{2\pi k}{250} + 0.2\sin\dfrac{2\pi k}{25} & k > 500 \end{cases} \qquad (18)$$

The first 800 data points are used to build an identifier model based on the SVR-NN. The performance of the SVR-NN model is verified using the remaining 800 data points. The parameters in ε-SVR are set as $\varepsilon = 0.15$, $C = 1$, the Gaussian kernel function with $\gamma = 3$. First, an initial structure of the adaptive SVR-NN with the hidden nodes (i. e. the number of support vectors) is obtained as 14 by an ε-SVR approach. The initial result with ε-SVR of SVR-NN in the test set is shown in Fig. 6 (dotted line). Therefore, the resulting structure of SVR-NN model is $\mathrm{NN}_{5\text{-}14\text{-}1}$.

These parameters of the SVR-NN are then adjusted by the robust learning algorithm. When the number of the training epochs needed is up to 600, the corresponding RMSE is 7.6×10^{-3} in the training region and the RMSE is 2.18×10^{-2} in the test region, respectively. The error convergence curve is also shown in Fig. 7 and the final result of the SVR-NN in the test set is shown in Fig. 8. Besides, it can be seen that the output from the identified model attains a fairly good match with that of the actual model in the test set. Hence, the adaptive SVR-NN shows good approximation accuracy for the nonlinear system model.

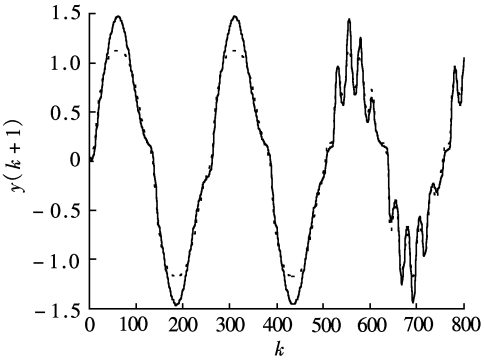

Fig. 6 Initial identification result with ε-SVR (dotted line) and output of actual model (solid line)

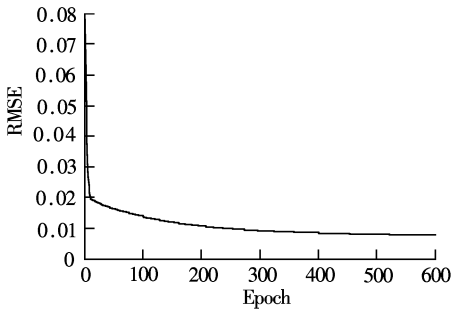

Fig. 7 Error convergence curve of adaptive SVR-NN

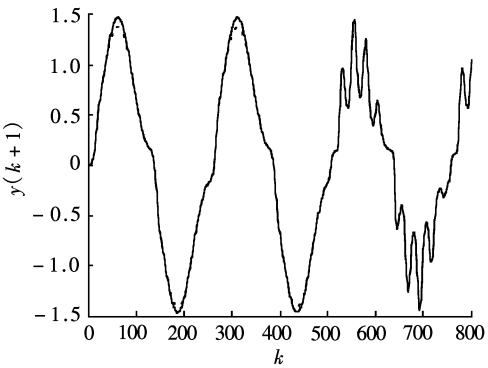

Fig. 8 Output of adaptive SVR-NN (dotted line) and actual model (solid line)

4 Conclusion

A novel adaptive SVR-NN is derived from support vector regression. In the proposed approach, an ε-SVR approach is used to determine the initial structure of the adaptive SVR-NN and an annealing robust learning algorithm is then applied to adjust the weight

as well as the parameters of hidden nodes of the SVR-NN. The main advantage of the SVR-NN is that its structure and initial parameters may be adaptively selected based on the support vectors. As these support vectors are usually scattered over the input space, good generalization results can be obtained using fewer hidden nodes. The SVR-NN with the robust learning algorithm not only has fast and stable convergence performance and precision, but it can also suppress the overfitting phenomena when the training data include outliers.

Experimental results confirm that the identification schemes based on the SVR-NN give considerably better performance and show faster learning in comparison to previous methods. Furthermore, the SVR-NN can also be used as adaptive controllers for nonlinear systems. In conclusion, the adaptive SVR-NN exhibits powerful function approximation capabilities, reasonable identification and prediction power for system modeling.

References

[1] Lin C T, Lee C S G. *Neural fuzzy system* [M]. New Jersey: Prentice-Hall, 1996.

[2] Vapnik V N. *The nature of statistical learning theory*. 2rd ed[M]. New York: Springer-Verlag, 1999.

[3] Burges C J C. A tutorial on support vector machines for pattern recognition [J]. *Data Mining Knowledge Discovery*, 1998, **2**(2): 1 - 43.

[4] Poggio T, Rifkin R, Mukherjee S, et al. General conditions for predictivity in learning theory[J]. *Nature*, 2004, **428**: 419 - 422.

[5] Keerthi S S. Efficient tuning of SVM hyperparameters using radius/margin bound and iterative algorithms[J]. *IEEE Trans on Neural Networks*, 2002, **13**(5): 1225 - 1229.

[6] Chuang C C, Su S F, Hsiao C C. The annealing robust backprogagation(ARBP) learning algorithms[J]. *IEEE Trans on Neural Networks*, 2000, **11**(5): 1067 - 1077.

[7] Chuang C C, Su S F. Robust support vector regression networks for function approximation with outliers[J]. *IEEE Trans on Neural Networks*, 2002, **13**(6): 1322 - 1330.

[8] Chan W C, Chan C W, Cheung K C, et al. On the modeling of nonlinear dynamic systems using support vector neural networks[J]. *Engineering and Applications of AI*, 2001, **14**(2): 105 - 114.

[9] Schölkopf B, Smola A J, Williamson R C, et al. New support vector algorithms[J]. *Neural Computation*, 2000, **12**(5): 1207 - 1245.

[10] Smola A J, Schölkopf B, Müller K R. The connection between regularization operators and support vector kernels [J]. *Neural Networks*, 1998, **11**(4): 637 - 649.

[11] Vapnik V N, Chapelle O. Bounds on error expectation for support vector machines [J]. *Neural Computation*, 2000, **12**(9): 2013 - 2036.

[12] Schölkopf B, Sung K K, Burges C J. Comparing support vector machines with Gaussian kernels to radial basis function classifiers[J]. *IEEE Trans on Signal Processing*, 1997, **45**(11): 2758 - 2765.

[13] Narendra K S, Parthasarathy K. Identification and control of dynamical systems using neural networks[J]. *IEEE Trans on Neural Networks*, 1990, **1**(1): 4 - 27.

[14] Box G E P, Jenkins G M. *Time series analysis: forcasting and control*. 2rd ed[M]. San Francisco: Holden-Day, 1976.

[15] Liang W, Langari R. Complex system modeling via fuzzy logic[J]. *IEEE Trans Syst, Man and Cybern, Part B*, 1996, **26**(1): 100 - 106.

基于支持向量回归神经网络的动态系统辨识

李 军 刘君华

(西安交通大学电气工程学院,西安 710049)

摘要:结合支持向量机和神经网络各自的优点,提出了一种新颖的自适应支持向量回归神经网络(SVR-NN). 首先,利用支持向量回归方法确定 SVR-NN 的初始结构和初始化权值,基于支持向量自适应地构造 SVR-NN 神经网络的隐层节点;然后,使用退火过程的鲁棒学习算法更新网络节点参数和权值. 为了验证所提出方法的有效性,给出了自适应 SVR-NN 应用于非线性动态系统辨识的实例. 仿真结果表明,与以前的神经网络方法相比,基于 SVR-NN 网络的辨识方案能获得相当好的性能,它具有很快的收敛速度. 因此,自适应的 SVR-NN 为非线性系统辨识提供了极有吸引力的新途径.

关键词:支持向量回归;神经网络;系统辨识;鲁棒学习算法;自适应性

中图分类号:TP183