Combining Self-organizing Feature Map with Support Vector Regression Based on Expert System[1)]

WANG Ling MU Zhi-Chun GUO Hui

(*Information Engineering School, Beijing University of Science and Technology, Beijing* 100083)
(E-mail: linda_gh@sina.com)

Abstract A new approach is proposed to model nonlinear dynamic systems by combining SOM (self-organizing feature map) with support vector regression (SVR) based on expert system. The whole system has a two-stage neural network architecture. In the first stage SOM is used as a clustering algorithm to partition the whole input space into several disjointed regions. A hierarchical architecture is adopted in the partition to avoid the problem of predetermining the number of partitioned regions. Then, in the second stage, multiple SVR, also called SVR experts, that best fit each partitioned region by the combination of different kernel function of SVR and promote the configuration and tuning of SVR. Finally, to apply this new approach to time-series prediction problems based on the Mackey-Glass differential equation and Santa Fe data, the results show that SVR experts has effective improvement in the generalization performance in comparison with the single SVR model.

Key words SOM clustering, SVR experts, single SVR, Mackey-Glass differential equation, Santa Fe data

1 Introduction

It is difficult to develop a compact mathematical model to represent the nonlinear dynamic systems. Two of the key problems are noise and non-stationarity. The noise in the data could lead to the over-fitting or under-fitting problem. The non-stationarity implies the situation changes very often over time with multiple variables and serious nonlinearity. For the last decade, there has been considerable growth in the development and application of artificial neural networks to solve these problems. Although these problems have been solved to a certain extent, there are still no effective methods to improve the generalization ability. Therefore, a new approach called support vector machines(SVMs) has been proposed. SVMs, originally developed by Vapnik, is a novel machine learning method based on the statistic learning theory. Currently, SVMs has many applications in pattern recognition, function estimation, signal procession, control and other fields.

In general, it is hard for a single model including SVMs to capture such a dynamic input-output relationship inherent in the data. Furthermore, using a single model to learn the data is somewhat mismatch as there are different noise levels in different input regions — the noise in the data could lead to over-fitting in some region or under-fitting in another region. Thus, is a potential solution is to use a mixture of experts (ME) architecture[1,2]. Milidiu *et al.*[3] generalized the ME architecture into a two-stage architecture to handle the non-stationarity in the data, As shown in Fig. 1 (a), in the first stage, the Isodata clustering algorithm[4] is used to partition the whole input space into several disjointed regions. Then, in the second stage, a mixture of experts including partial least squares, K-nearest neighbors and carbon copy are competed to solve partitioned regions. This idea of generalizing SVMs into the ME architecture was discussed in [5]. Based on the ME architecture proposed by Milodiu, this paper incorporates the ME architecture into SVMs by using a two-stage neural network architecture. As illustrated in Fig. 1 (b), in the first stage, self-organizing feature map (SOM) is used as a clustering algorithm to partition the whole input space into several disjointed regions. A hierarchical architecture is adopted in the partition to avoid the problem of predetermining the number of partitioned regions. Then, in the second stage, for each partitioned region, the output is achieved by the best combination of the weighted SVR experts for the final prediction. As the partitioned regions have more uniform distributions than the whole input space, it will become easier for SVR experts to capture such a more stationary input-output relationship. Traditional approach need to choose the most appropriate kernel function and the optimal learning parameters to enhance the robust capability of SVR, when

1) Supported by the National High Technology Research and Development Program of P. R. China (2002 AA412010) and the Technology Development Program of the Ministry of Science and Technology of P. R. China (2003EG113016)
Received September 20, 2004; in revised form January 19, 2005

the single SVR is trained to map the same input-output data. In our investigation, we disregard all parametric kernel issues and focus on the regression problem. As different partitioned regions have different characteristics, by taking this weighted averaging combination architecture, the best combination of SVR experts will be used for the final function prediction. The resulting combination is called MSE-OLC[6]. In order to assess this new approach, we compare the single SVR model with the SVR experts. The Mackey Glass Equation and the Santa Fe data are evaluated in the experiment. The simulation shows that there is great improvement in prediction performance by using SVR experts.

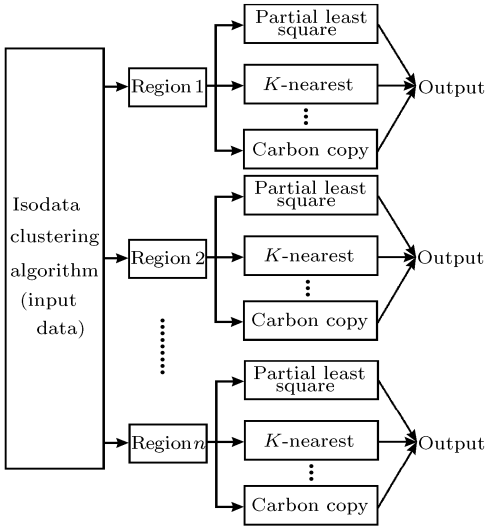

Fig. 1 (a) A generalized two-stage mixture of experts

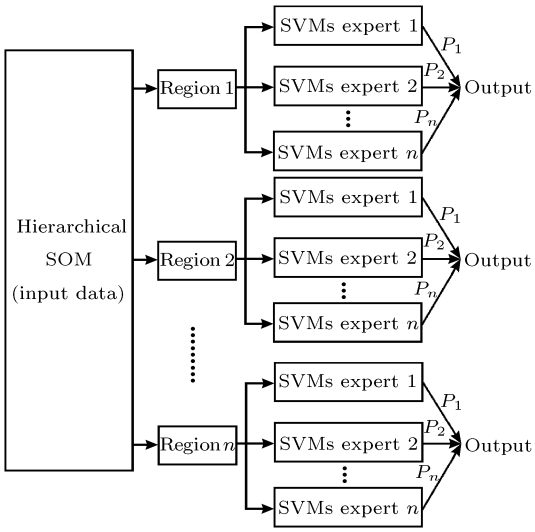

Fig. 1 (b) The generalized SVMs experts

This paper is organized as follows. An introduction of SOM algorithm is given in Section 2. In Section 3, we review the use of SVM in regression problem and present the architecture of the SVR experts. Section 4, the combination criteria are described. In Section 5, experiment results are presented and discussed. Finally, some conclusions are drawn in Section 6.

2 Clustering method using a hierarchical self-organizing feature map algorithm

The Self-organizing feature map training algorithm proposed by Kohonen[7] is summarized as follows.

Step 1. Initialization: Choose random values for the initial weights $w_j(0)$.

Step 2. Winner Finding: Find the winning neuron j^* at time k, using the minimum-distance Euclidean criterion

$$j^* = \arg\min_j \|\boldsymbol{x}(k) - w_j\|, \quad j = 1, \cdots, N^2 \tag{1}$$

where $\boldsymbol{x}(k) = [x_1(k), \cdots, x_n(k)]$ represents the kth input pattern, N^2 is the total number of neurons, and $\|\cdot\|$ indicates the Euclidean norm.

Step 3. Weights Updating: Adjust the weights of the winner and its neighbors, using the following rule:

$$w_j(k+1) = w_j(k) + \eta(k) N_{j^*}(k)(\boldsymbol{x}(k) - w_j(k)) \tag{2}$$

where $\eta(k)$ is a positive constant and $N_{j^*}(k)$ is the topological neighborhood function of the winner neuron at time k. It should be emphasized that the success of the map formation is critically dependent on how the values of the main parameters (*i.e.*, $\eta(k)$ and $N_{j^*}(k)$), initial values of weight vectors, and the number of iterations are prespecified.

Based on the above learning algorithm, we have defined a hierarchical SOM as shown in Fig. 2, and have constructed a clustering method using it to enable fast, highly reliable clustering. Our clustering method uses a two-layer SOM. The first layer SOM divides input data into rough groups, and the second-layer SOM classifies each group into detailed clusters. The method greatly reduces the SOM′s size and learning time. The clustering algorithm using the two-layer SOM is as follows.

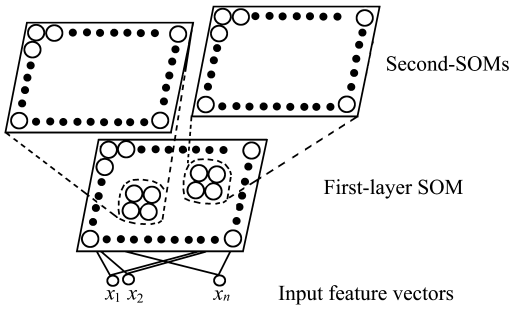

Fig. 2 Hierarchical SOM

1) Provide input feature vectors to the first-layer SOM.

2) Apply the basic SOM learning algorithm to the first-layer SOM. The neighborhood function $N_{j^*}(k)$ is defined as

$$N_{j^*}(k) = \begin{cases} 1, & \mathrm{dis}(w_j, w_{j^*}) \leqslant r(k) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where $\mathrm{dis}(\cdot)$ is defined as the distance between nodes j and j^* and $r(k) = r(0) * (1 - k/T)$. Let $r(0)$ be the initial value of $r(k)$, T be the number of learning iterations. We create the first layer of the SOM, which provides input data to the SOM′s output layer at fixed positions depending on the initial value $r(0)$.

3) For all fixed positions of the first-layer SOM, extract a unique set of data that will be input to the second-layer SOM from each fixed position.

4) Apply the learning algorithm of the basic SOM to all second-layer SOMs.

5) For each second-layer SOM, use the distance map and the clustering map[8]; then extract the cluster information by using the clustering algorithm.

Hence, the clustering number C achieved from the two-layer SOM algorithm can be used to determine the number of the partitioned regions C in the input space.

3 SVR experts

Support Vector Regression (SVR) is one of the most important application of SVMs[9]. The standard SVR is to solve the approximation problem such as

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) K(\boldsymbol{x}_i, \boldsymbol{x}) + b \tag{4}$$

where α_i^* and α_i are Lagrange multipliers. The kernel function $K(\boldsymbol{x}_i, \boldsymbol{x})$ is defined as a linear dot product of the nonlinear mapping, *i.e.*,

$$K(\boldsymbol{x}_i, \boldsymbol{x}) = \boldsymbol{\varphi}(\boldsymbol{x}_i)\boldsymbol{\varphi}(\boldsymbol{x}) \tag{5}$$

The coefficients α_i^* and α_i of (4) are obtained by minimizing the following regularized risk functional $R_{reg}[f]$, which is a combination of the model complexity and the empirical risk, for given error bound ε,

$$R_{reg}[f] = \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C \sum_{i=1}^{l} L_\varepsilon(y) \tag{6}$$

Here, $\|\boldsymbol{\omega}\|^2$ is a term which characterizes the model complexity, C is a constant determining the trade-off and the ε-insensitive loss function $L_\varepsilon(y)$ is given by

$$L_\varepsilon(y) = \begin{cases} 0, & \text{for } |f(\boldsymbol{x}) - y| < \varepsilon \\ |f(\boldsymbol{x}) - y| - \varepsilon, & \text{otherwise} \end{cases} \tag{7}$$

The minimization of regularized risk function in (3) can be converted to the following constrained optimization problem:

$$\min_{\alpha,\alpha^*} \boldsymbol{\omega}(\alpha,\alpha^*) = \min_{\alpha,\alpha^*} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i^*-\alpha_i)(\alpha_j^*-\alpha_j)K(\boldsymbol{x}_i,\boldsymbol{x}) - \sum_{i=1}^{N}(\alpha_i^*-\alpha_i)y_i + \varepsilon\sum_{i=1}^{N}(\alpha_i^*+\alpha_i) \Bigg\} \quad (8)$$
$$\text{subject to} \begin{cases} \sum_{i=1}^{N}(\alpha_i^*-\alpha_i)=0 \\ \alpha,\alpha^* \in [0,c] \end{cases} \Bigg\}$$

where the kernel function used is Gaussian and defined as

$$k(\boldsymbol{x}_i,\boldsymbol{x})\exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}\right) \quad (9)$$

where σ is a constant.

Now, we investigate the every partitioned region, which is actually the combination of M different SVR experts. In this novel approach, the weighted output in every partitioned region is given by

$$y = f(\boldsymbol{x}) = \sum_{k=1}^{M} p_k(\boldsymbol{\omega}_k\boldsymbol{\varphi}_k(\boldsymbol{x})+b_k) \quad (10)$$

where w_k, b_k are, respectively, the weights and bias of the kth expert

$$\sum_{k=1}^{M} p_k = 1, \quad 0 \leqslant p_k \leqslant 1,\ k=1,\cdots,M$$

As already mentioned, we comply with the SVR approach as follows.

$$\begin{aligned} \text{minimize } & \frac{1}{2}\sum_k |\boldsymbol{\omega}_k|^2 + C\sum_i(\xi_i+\xi_i^*) \\ & y_i - \sum_k p_k(\boldsymbol{\omega}_k\boldsymbol{\varphi}_k(\boldsymbol{x}_i)+b_k) \leqslant \varepsilon+\xi_i^* \\ \text{subject to } & \sum_k p_k(\boldsymbol{\omega}_k\boldsymbol{\varphi}_k(\boldsymbol{x}_i) - y_i \leqslant \varepsilon+\xi_i \\ & \xi,\xi_i^* \geqslant 0, \quad \text{for } i=1,\cdots,N \end{aligned} \quad (11)$$

Then, the resulting QP problem may be written as

$$\min_{\alpha,\alpha^*} \boldsymbol{\omega}(\alpha,\alpha^*) = \min_{\alpha,\alpha^*} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{M}(\alpha_i^*-\alpha_i)(\alpha_j^*-\alpha_j)p_{ki}p_{kj}K(\boldsymbol{x}_i,\boldsymbol{x}) - \sum_{i=1}^{N}(\alpha_i^*-\alpha_i)y_i + \varepsilon\sum_{i=1}^{N}(\alpha_i^*+\alpha_i) \Bigg\} \quad (12)$$
$$\text{subject to} \begin{cases} \sum_{i=1}^{N}(\alpha_i^*-\alpha_i)=0 \\ \alpha,\alpha^* \in [0,c] \end{cases} \Bigg\}$$

In comparison with the standard SVR, if the weight p_k is obtained, the multiple experts formulation is easy to be achieved.

4 Combination

In this section, we concentrate on the selection criteria proposed by Hashem[10], which is addressed after a brief revision of the basic and generalized ensemble methods. BEM (basic ensemble method) subsumes the combination of a population of regression estimates of a function $f(\boldsymbol{x})$, defined as $f(\boldsymbol{x}) = E[y/\boldsymbol{x}]$. According to Perron and Cooper[15], we may define $f(\boldsymbol{x})_{BEM}$ as

$$f(\boldsymbol{x})_{BEM} = \frac{1}{M}\sum_{i=1}^{M} f_i(\boldsymbol{x}) \quad (13)$$

which comes to be a simple average defined over F.

Hashem has proposed four MSE-OLC variations to tackle the problem of finding the best set of weight factors. Among the proposed variations, the unconstrained MSE-OLC with a constant term has has the lowest MSE results, here we calculate the weight factor p_k using this MSE-OLC method. For the BEM estimator, as we increase the size of the expert, the assumption that all $m_i = f(\boldsymbol{x}) - f_i(\boldsymbol{x})$ (deviation from the true solution) are mutually independent does not hold any more. When this assumption fails, adding more experts to the group leads to bad performance of the whole BEM estimator.

Hence, the best choice would be to find out the optimal subset over which we could calculate the average. This process may be refined by considering the difference in MSE when we pass from a BEM estimator, with a population of K elements, to a BEM estimator with a population of $K+1$ elements. From this comparison, it is advocated that we may only include a new expert to the group if the following inequality is satisfied:

$$(2k+1)MSE[\hat{f}_N] > 2\sum_{i\neq new} E[m_{new}m_i] + E[m_{new}^2] \tag{14}$$

where $MSE[\hat{f}_N]$ is the MSE of the BEM estimator with M experts, $E[\cdot]$ refers to the mathematical expectation operator, and m_{new} is the error that should be produced by the new expert to be inserted into the group. If this criterion is not satisfied, we discard the current expert and apply the same comparative process to the next expert in the ordered sequence.

The regression quality was evaluated by comparing (*via* MSE) the output values produced by the combination of SVM experts. Below, we enumerate the various kernels adopted during the simulation experiments. A more detailed discussion on these kernels may be found elsewhere[11].

1) Linear: $K(x,y) = xy$

2) Polynomial: $K(x,y) = (xy+1)^d$

3) Gaussian radial basis function: $K(x,y) = \exp(-(x-y)^2/(2\sigma^2))$

4) Exponential radial basis function: $K(x,y) = \exp(-|x-y|/(2\sigma^2))$

5) Sigmoid: $K(x,y) = \tanh(b(xy)+c)$

6) Fourier series: $K(x,y) = \sin(N+\frac{1}{2})(x-y)/\sin(\frac{1}{2}(x-y))$

7) Linear Splines: $K(x,y) = 1 + xy + xy\min(x,y) - \frac{(x+y)}{2}(\min(x,y))^2 + \frac{1}{3}(\max(x,y))^3$

8) Bn-splines: $K(x,y) = B_{2n+1}(x-y)$

For the proposed architecture, we adopted the following algorithm.

1) A hierarchical architecture of self-organizing feature map (SOM) is used as a clustering algorithm to partition the whole input space into several disjointed regions.

2) Generate and train SVM experts in every partitioned region, where each kernel type is assigned to a different SVM expert and the combination weights are all equal.

3) Calculate the output of every partitioned region for the selection data set.

4) Select the best combination for every partitioned region *via* Perrone and Cooper′s criterion (14).

5) Calculate the weight factors using the MSE-OLC method.

6) Cluster the validation set into the partitioned regions by the above SOM algorithm.

7) Obtain the outputs of the SVM experts for the validation set.

For an unknown data point in testing, it is clustered into one of the partitioned regions by SOM. Then its output is produced by the corresponding SVR experts.

5 Experimental result

Experiment 1

Our application is a high dimensional chaotic system generated by the Mackey-Glass delay differential equation:

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -0.1x + \frac{0.2x(t-t_d)}{1+x^{10}(t-t_d)}, \quad t_d > 17 \tag{15}$$

In our simulation, we set the parameter $t_d = 30$. (15) was originally introduced as a model of blood cell regulation[12] and became quite common as artificial forecasting benchmark. The goal of this

task is to use known values of the time series up to the point $x = t$ to predict the value at some point in the future $x = t + \tau$. The standard method for this type of prediction is to create a mapping from $m = 9$ points of the time series and a step size $\tau = 1(\boldsymbol{x}_t = (x(t-(m-1)\tau), \cdots, x(t-\tau), x(t)$ to a predicted future value $x(t+\tau)$. The values $m = 9$ and $\tau = 1$ were used, *i.e.*, nine point values in the series were used to predict the value of the next time point. Fig. 3 shows 1000 points of this chaotic series used to test our model.

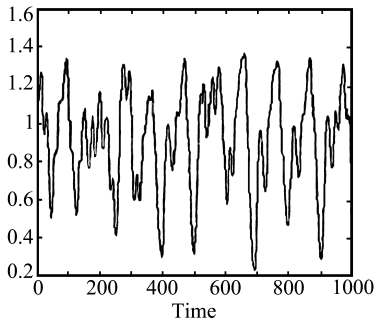

Fig. 3 Mackey-glass chaotic time series

For (15), we compare the MSE between the single SVR and SVR experts with test data. The achieved results are shown in Table 1. Here, MSE=$\sqrt{\frac{1}{2}\sum_{i=1}^{k}(x_i - \hat{x}_i)^2}$, x_i is the true output, $\hat{x}_i$ is the estimate output, k is the number of the test data. In the whole experiment, the sequential minimal optimization algorithm solving the regression problem[13] is implemented for training SVR and the program is developed by using VC++.net language. We apply 10-fold cross-validation to select insensitive values $\varepsilon = 0.02$ and $C = 10$ to produce the smallest prediction errors. In Table 1, there are five clusters and each cluster has different kernel types. The investigated kernel functions are restricted into five categories: Gaussian kernel, exponential kernel, Fourier kernel, linear splines kernel and Bn-splines kernel. For example, in the second cluster, the Gaussian kernel achieves better performance than the Linear splines kernel and the Bn-splines kernel. But the SVR experts could achieve a smaller MSE than the best single SVR by using the Gaussian kernel. The aim here is to show that, for a small training set, the SVR experts produce a good generalization capability because a single SVR overfits the testing data. Figs. 4(a) and (b) depicts the overall prediction performance from single SVR without SOM and SVR experts combined with SOM, respectively. Obviously, the experts forecast more closely to the actual values than the single SVR model by using the Gaussian kernel in most of the testing time period. And there are corresponding smaller MSE prediction errors in the SVR experts than the best single SVR model, as illustrated in Figs. 4(a) and (b).

Table 1 The used kernel types and the MSE in a partitioned region

Clustering order	Single SVR		SVR experts	
	Kernel type	MSE	Kernel type	Weighted average MSE
1	4	0.0034	4	0.026
	7	0.0036	7	
	8	0.0045	8	
2	3	0.0016	3	0.0013
	7	0.0027	7	
	8	0.0019	8	
3	3	0.0067	4	0.0045
	6	0.0058	6	
	8	0.0064	8	
4	3	0.0027	7	0.0022
	7	0.0031	3	
	8	0.0035	8	
5	8	0.0041	4	0.0039
	4	0.0049	8	
	7	0.0043	7	

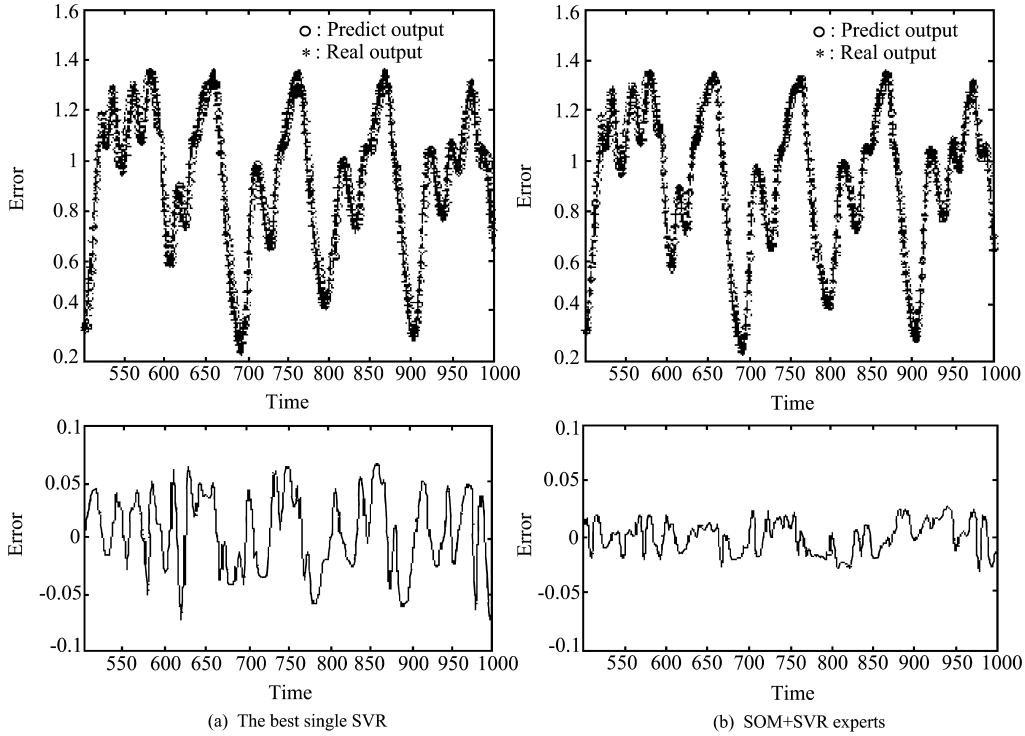

Fig. 4 Simulation output and error of time series from $x(501) \sim x(1000)$

Experiment 2

In this experiment, the data set D in Santa Fe was adopted. We can obtain these data from http://www.physionet.org/physionbank/database/sa-ntafe/. For the data set D, the mean square error (MSE) is also used to evaluate the performance of SVMs as this criterion is used in the previous studies[14].

The result of the SVR experts and the single SVR are given in Table 2. The parameters are $\varepsilon = 0.06$ and $C = 15$, which are selected by 10-fold cross-validation to produce the smallest prediction errors. The investigated kernel functions are restricted into four categories: the poly kernel, the Gaussian kernel, the sigmoid kernel, the Fourier kernel. As given in Table 2, in the first cluster, the sigmoid

Table 2 The results in Santa Fe time series

Clustering order	Single SVR		SVR experts	
	Kernel type	MSE	Kernel type	Weighted average MSE
1	2	0.0224	2	0.0188
	3	0.0256	3	
	5	0.0212	5	
2	3	0.0254	3	0.0199
	5	0.0215	5	
	6	0.0208	6	
3	2	0.0235	2	0.0207
	3	0.0250	3	
	6	0.0217	6	
4	2	0.0227	2	0.0226
	5	0.0231	5	
	6	0.0235	6	

kernel performs best among the single SVR models, while the SVR experts achieve the smaller MSE than the Sigmoid kernel. As illustrated in Fig. 5, the SVR experts achieve the smaller prediction errors than the best single SVR model by using the Gaussian kernel (without SOM) in most of the testing time period. Obviously, the SVR experts prediction accuracy is better than that of the best single SVR

model.

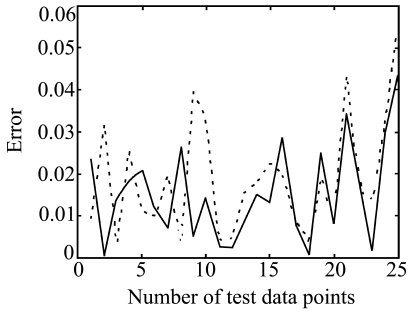

Fig. 5 Prediction error in the SVR experts (real line) and the best single SVR (dot line)

6 Conclusions

In this paper, we show that the employment of SVR experts may significantly improve the regression accuracy when compared with the conventional, single SVR method. The simulation results show that the SVR experts model is more effective and efficient in forecasting noisy and non-stationary practical problem than the single SVR model. As future work, we will continue to investigate other possibilities of automatically combining different kernel functions into the same neural structure, as well as to compare SVR experts with other approach.

References

1 Jacobs R A, Jordan M A, Nowlan S J, Hinton G E. Adaptive mixtures of local experts. *Neural Computation*, 1991, **3**(1): 79∼87
2 Jordan M I, Jacobs R A. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 1994, **6**(2): 181∼214
3 Milidiu R L, Machado R J, Rentera R P. Time-series forecasting through wavelets transformation and a mixture of expert models. *Neurocomputing*, 1999, 28: 145∼146
4 Venkateswarlu N B, Raju P S V S. Fast ISODATA clustering algorithms. *Pattern Recognition*, 1992, **25**(3): 335∼343
5 Kwok T J. Support vector mixture for classification and regression problems, In: Proceedings of the 14th International Conference on Pattern Recognition, Brisbane, Australia: IEEE Computer Society, 1998. 255∼258
6 Hashem S, Schmeiser B. Improving model accuracy using optimal linear combinations of trained neural networks. *IEEE Transactions on Neural Networks*, 1995, **6**(3): 792∼794
7 Kohonen T. Self-organizing Maps. Berlin Heidelberg: Springer-verlag, 1995
8 Endo M, Ueno M, Tanabe T, Yamamoto M. Clustering method using self-organizing map. In: Proceedings of the 2000 IEEE Signal Processing Society Workshop, Piscateway, NJ, USA: IEEE, 2000, (1): 261∼270
9 Vapnik V. The Nature of Statistical Learning Theory. New York: Springer Verlag, 1995
10 Hashem S. Optimal linear combinations of neural networks. *Neural Network*, 1997, **10**(4): 599∼614
11 Gunn S. Support vector machine for classification and regression. Image Speech & Intelligent Systems Group, Technical Report ISIS-1-98, University of Southampton, 1998
12 Mackey M C, Glass L. Oscillation and chaos in physiological control system. *Science*, 1997, **197**(4300): 287∼289
13 Smola A J, Scholkpf B. A tutorial on support vector regression. NeuroCOLT Technical Report NC-TR-98-030, Royal Holloway College, University of London, UK, 1998
14 Muller K R, Smola A J, Ratsch G, Scholkopf B, Kohlmorgen J. Using support vector machines for time series prediction. In: B. Scholkopf, C.J.C. Burges, A.J. Smola (Eds.), Advances in Kernel Methods–Support Vector Learning, Cambridge, MA: MIT Press, 1999. 243∼254
15 Perrone M P, Cooper L N. When network disagree: Ensemble method for neural networks. In: Mammone, R. J.editor, Neural Networks for Speech and Image Processing, 1993, 126∼142

WANG Ling Ph. D. candidate in the Institute of Information Engineering at Beiing University of Science and Technology. Her research interests include artificial intelligent, machine learning, and data mining.

MU Zhi-Chun Professor at Beijing University of Science and Technology. His research interests include the artificial intelligent control and machine learning.

GUO Hui Ph. D. candidate in the Institute of Information Engineering at Beijing the University of Science and Technology. His research interests include machine learning and data mining.