Hybrid Approach of Selecting Hyperparameters of Support Vector Machine for Regression

Jin-Tsong Jeng

***Abstract*—To select the hyperparameters of the support vector machine for regression (SVR), a hybrid approach is proposed to determine the kernel parameter of the Gaussian kernel function and the epsilon value of Vapnik's ε-insensitive loss function. The proposed hybrid approach includes a competitive agglomeration (CA) clustering algorithm and a repeated SVR (RSVR) approach. Since the CA clustering algorithm is used to find the nearly "optimal" number of clusters and the centers of clusters in the clustering process, the CA clustering algorithm is applied to select the Gaussian kernel parameter. Additionally, an RSVR approach that relies on the standard deviation of a training error is proposed to obtain an epsilon in the loss function. Finally, two functions, one real data set (i.e., a time series of quarterly unemployment rate for West Germany) and an identification of nonlinear plant are used to verify the usefulness of the hybrid approach.**

***Index Terms*—Competitive agglomeration (CA) clustering algorithm, hyperparameters, repeated support vector machine for regression (RSVR) approach, support vector machine for regression (SVR).**

I. Introduction

The support vector machine (SVM) proposed by Vapnik and his research groups is a novel approach for solving classification problems [1]. The SVM implements the structural risk minimization principle that minimizes the upper bound of the generalization error. This induction principle is based on the fact that the generalization error is bounded by the sum of a training error and a confidence-interval term that depends on the Vapnik–Chervonenkis (VC) dimension. For the classification, the SVM tries to find the optimal hyperplane, which is expressed as a linear combination of a subset of training data (called support vectors) by solving a linearly constrained quadratic-programming (QP) problem with a maximum margin between the two classes. Additionally, with the introduction of Vapnik's ε-insensitive loss function, the SVM has been extended to solve a nonlinear-regression-estimation problem, called the SVM for regression (SVR). Recently, the SVR has been applied to various fields such as optimal control [2], time-series prediction [3], interval regression analysis [4], the determination of initial structure for the fuzzy-neural networks, and the radial basis-function networks (RBFNs) [5], [6]. In addition, some of results about the comparison between the SVR approach and other learning schemes have been proposed [7].

To obtain better performance, some parameters, called hyperparameters, must be selected carefully in the SVM or SVR [8]–[10]. The hyperparameters include the kernel parameters, the epsilon value in Vapnik's ε-insensitive loss function, and the regularization constant. Some results exist regarding the selection of hyperparameters for SVM [11], [12]. In [11], the authors reviewed some possible hyperparameter-selection methods such as cross-validation methods, bootstrapping, and Bayesian methods for the kernel parameters and the regularization constant. The validation-set approaches are still an inappropriate means of selecting the hyperparameters, since they need an iterative process involving expensive computation. Duan *et al.* [12] proposed a novel approach based on gradient-descent algorithm for automatically tuning multiple parameters in an SVM. This approach avoids holding out some data for validation, thus making full use of the training set for parameter optimization. However, an important question arose regarding how to achieve good selection for parameter tuning in an SVM.

For the SVR, some of results regarding the selected hyperparameters have been proposed [4], [8], [10], [13]–[17]. In [9] and [10], the authors had shown that the performance (results) of SVR is sensitive to the hyperparameters. It is shown in [10] that the result of SVR appears in the underfitting and overfitting situations when the hyperparameters are not chosen properly. Additionally, they also demonstrated that the result of SVR for the fixed regularization constant appears in the underfitting and overfitting situations when the kernel parameter and epsilon are improperly selected. Jeng *et al.* [4] showed that the regularization constant and epsilon have a smooth effect on the results of SVR. In [13], the authors suggest controlling another parameter v (i.e., the fraction of points outside the ε tube) instead of ε. In this approach, parameter ε has to be user defined. Chalimourda and Scholkopf [14] showed the optimal nu and its parameter settings in SVR for different noise models. Basically, selecting the epsilon value depends on a prior knowledge. Mattera and Haykin [15] proposed to choose epsilon so that the percentage of SVs in SVR was around 50% of the number of samples. However, providing some examples whose optimal performances of SVR were achieved was relatively easy when using the number of support vectors larger or smaller than 50%. Cherkarsky *et al.* [16], [17] proposed an empirical approach to select the epsilon value that considers the standard deviation of input noise and the number of samples. Based on the assumption of linear-regression concepts, the epsilon value can be estimated. Additionally, the regularization constant controls the tradeoff between the model complexity and approximation accuracy. In [15], the regularization constant was set to equal to the range of responsive values in the training data. In [16], the regularization constant is set with three times the standard deviation of the output response values. Moreover, in [8], the authors suggested that the regularization constant should be set to be sufficiently large. Due to the fact that the Gaussian kernel function was used frequently in the SVR applications, this paper put the Gaussian kernel parameter (GKp) into consideration (i.e., the spread width of the Gaussian function). In [16], the GKp is experimentally selected as $0.2 \sim 0.5$ times the input range of the training data. However, this approach does not perform a suitable range for GKp.

This paper proposes a hybrid approach to select the GKp and the epsilon value. When the kernel function is selected as the Gaussian function, the SVR approach can be viewed as a constructive learning procedure employed to implement RBFNs. Hence, traditional unsupervised learning (i.e., clustering algorithm) approaches for determining the spread width of the Gaussian function in the RBFNs and neural-fuzzy system are also suitable for determining the GKp in the SVR [18]–[20]. However, traditional clustering algorithms such as fuzzy C-mean (FCM) cannot determine the number of clusters in the clustering process. Hence, the competitive agglomeration (CA) clustering algorithm [21] is applied to overcome this problem. The clustering process obtains the nearly "optimal" number of clusters and the centers and spread width of clusters. Besides, under the results of SVR with the CA clustering algorithm, the repeated SVR (RSVR) approach is proposed to determine the epsilon value based on the standard deviation of training errors. That is, the SVR approach with $\varepsilon = 0$ is first applied to obtain the training errors. The epsilon value is obtained by the standard-deviation concept of statistical theory. Subsequently,

Manuscript received February 7, 2004; revised July 7, 2004, December 19, 2004, and June 31, 2005. This work was supported by the National Science Council under Grant NSC 91-2213-E-150-026. This paper was recommended by Associate Editor N. R. Pal.

The author is with the Department of Computer Science and Information Engineering, National Formosa University, Huwei Jen, Taiwan 632, R.O.C. (e-mail: tsong@nfu.edu.tw).

Digital Object Identifier 10.1109/TSMCB.2005.861067

the SVR approach with estimated epsilon and GKp is applied to obtain the final results of the SVR approach. In this paper, four examples were tested to verify the validity of the hybrid approach. First, the simulation results using the proposed hybrid approach demonstrate that the estimated GKp and epsilon are close to the "suboptimal" hyperparameter region, which has a lower testing rmse and fewer SVs in examples 1 and 2. Second, one real data set in example 3 (namely, the time series of quarterly unemployment rate for West Germany) is used to verify the usefulness of the proposed approach. Finally, the proposed approach was compared with the VCYM approach (i.e., practical selection of SVR parameter by Cherkassky and Ma [16]) for all examples. Based on the simulation results, the final results of SVR with the proposed approach have lower rmse and fewer SVs than the SVR with the VCYM approach. Consequently, the contribution of this paper is that a hybrid approach is proposed to select the GKp and the epsilon, and the SVR with a hybrid approach is applied to a time series of quarterly unemployment data for West Germany.

The remaining part of the paper is outlined as follows. Section II simply describes the SVR approach. In Section III, the selection of hyperparameters with a hybrid approach is proposed. In Section IV, the simulation results show the superiority of the proposed approach for the chosen hyperparameters. Concluding remarks are presented in Section V.

II. Support-Vector Regression

The SVR approach is used to approximate an unknown function with a set of noise data $\{(\vec{x}_k, y_k), k = 1, \ldots, p\}$. Assuming that a set of basis functions $\{g_s(\vec{x}), s = 1, \ldots, l\}$ is provided, a family of functions that can be expressed as a linear expansion of the basis functions exists. Restated, the problem of function approximation is transformed into that of finding the parameters of the following basis-function linear expansion:

$$f(\vec{x}, \vec{\theta}) = \sum_{s=1}^{l} \theta_s g_s(\vec{x}) + b \tag{1}$$

where $\vec{\theta} \in (\theta_1, \ldots, \theta_l)$ is a parameter vector to be identified and b is a constant. Hence, the solution for this problem is to find f that minimizes

$$R(\vec{\theta}) = \frac{1}{p} \sum_{k=1}^{p} L_\varepsilon \left(y_k - f(\vec{x}_k, \vec{\theta}) \right) \tag{2}$$

subject to the constraint

$$\|\vec{\theta}\|^2 \leq \gamma \tag{3}$$

where γ denotes the regularization constant and $L_\varepsilon(\cdot)$ represents the ε-insensitive loss function, and is defined as

$$L_\varepsilon(e) = \begin{cases} 0, & \text{for } |e| \leq \varepsilon \\ |e| - \varepsilon, & \text{otherwise.} \end{cases} \tag{4}$$

That is, if the e value is within the ε zone, the loss function is 0. Otherwise, the loss function is the magnitude of the difference between the absolute value of e and the ε zone.

By using the Lagrange multiplier method, minimizing (2) yields the following dual-optimization problem; minimize

$$\psi(\alpha, \alpha^*) = \varepsilon \sum_{u=1}^{p} (\alpha_u + \alpha_u^*) - \sum_{v=1}^{p} y_v \, (\alpha_v^* - \alpha_v) + \frac{1}{2} \sum_{u,v=1}^{p} (\alpha_u^* - \alpha_u) \, (\alpha_v^* - \alpha_v) \left[\sum_{s=1}^{l} g_s(\vec{x}_u) g_s(\vec{x}_v) \right] \tag{5}$$

subject to the constraint

$$\sum_{u=1}^{p} \alpha_u^* = \sum_{u=1}^{p} \alpha_u, \qquad 0 \leq \alpha_u, \alpha_u^* \leq \gamma, \quad \text{for } u = 1, \ldots, p. \tag{6}$$

The inner product of basis functions $g_s(\vec{x})$ in (5) was shown to be replaced by the kernel function

$$K(\vec{x}_u, \vec{x}_v) = \sum_{s=1}^{l} g_s(\vec{x}_u) g_s(\vec{x}_v). \tag{7}$$

The kernel function determines the smoothness properties of the solutions, and should reflect a prior knowledge on the data. Hence, the optimization of (5) is rewritten as

$$\psi(\alpha, \alpha^*) = \varepsilon \sum_{u=1}^{p} (\alpha_u + \alpha_u^*) - \sum_{v=1}^{p} y_v \, (\alpha_v^* - \alpha_v) + \frac{1}{2} \sum_{u,v=1}^{p} (\alpha_u^* - \alpha_u) \, (\alpha_v^* - \alpha_v) \, K(\vec{x}_u, \vec{x}_v). \tag{8}$$

Vapnik [1] indicated that the solution of the SVR approach is in the form of the following linear expansion of kernel functions:

$$f(\vec{x}, \alpha, \alpha^*) = \sum_{k=1}^{p} (\alpha_k^* - \alpha_k) \, K(\vec{x}, \vec{x}_k) + b \tag{9}$$

where the constant b is written as

$$b = \text{average}_k \left\{ \varepsilon \cdot \text{sign}(\alpha_k - \alpha_k^*) + y_k - \sum_i (\alpha_i - \alpha_i^*) K(\vec{x}_i, \vec{x}_k) \right\}. \tag{10}$$

Notably, only some of $(\alpha_k^* - \alpha_k)$'s are not 0's, and the corresponding vectors $\vec{x}_k$'s are called the SVs. The Gaussian function is used as the kernel function in this paper. Hence, (9) can be rewritten as

$$f(\vec{x}, \alpha, \alpha^*) = \sum_{k=1}^{p} (\alpha_k^* - \alpha_k) \exp\left\{ \frac{-\|\vec{x} - \vec{x}_k\|^2}{2\sigma^2} \right\} + b \tag{11}$$

where σ is termed the GKp.

III. Select the Hyperparameters of SVR Using a Hybrid Approach

This section proposes a method of selecting the hyperparameters using the hybrid approach. First, the CA clustering algorithm is applied to determine the GKp. That is, the CA clustering algorithm is used to find the nearly "optimal" number of clusters and the spread width of clusters in the clustering process. The CA clustering algorithm can determine the GKp in the hybrid approach. Second, the RSVR approach based on the standard deviation of training errors is proposed

and applied to obtain the epsilon value. That is, the RSVR can determine an epsilon value based on the regression errors in the hybrid approach.

A. Obtaining the GKp With the CA Clustering Algorithm

Let $Q = \{\vec{q}_k = (\vec{x}_k, y_k) | k = 1, 2, \ldots, p\}$ denote a set of p vectors in an $(n+1)$-dimensional feature space with coordinate axis label $(x_1, x_2, \ldots, x_n, y)$. Moreover, let $B = (\beta_1, \beta_2, \ldots, \beta_C)$ represent a C-tuple of prototypes, each of which characterizes one of the C clusters. Notably, each β_i consists of a set of parameters. To detect the interaction between the input and output variables, this paper considers the product space of input and output variables (input–output space) rather than only the input space. The CA clustering algorithm minimizes the following objective function

$$J(B, U, Q) = \sum_{i=1}^{C} \sum_{j=1}^{p} u_{ij}^2 d^2(\vec{q}_j, \beta_i) - \eta \sum_{i=1}^{C} \left[\sum_{j=1}^{p} u_{ij} \right]^2 \quad (12)$$

subject to

$$\sum_{i=1}^{C} u_{ij} = 1, \qquad \text{for } 1 \le j \le p \quad (13)$$

where $d^2(\vec{q}_j, \beta_i)$ denotes the distance from feature vector $\vec{q}_j$ to the prototype β_i, u_{ij} represents the degree of membership of feature vector $\vec{q}_j$ in clusters β_i, $U = [u_{ij}]$ is a $C \times (p+1)$ matrix known as a constrained fuzzy C-partition matrix, and η is an agglomeration parameter. Proper selection of η can enable the objective function J to be used to find compact clusters of various types while partitioning the data set into a minimal number of clusters. Ideally, the agglomeration process controlled by η should be initially slow to encourage small cluster formation. The process speed should then gradually be increased to promote agglomeration. When the number of clusters approaches an appropriate value, η hopefully should again decay slowly to enable the algorithm to converge. The η can be suggested as in [21]

$$\eta(\text{itr}) = \eta_0 \, e^{\frac{-\text{itr}}{\tau}} \frac{\sum_{i=1}^{C} \sum_{j=1}^{p} \left(u_{ij}^{(\text{itr}-1)} \right)^2 d^2(\vec{q}_j, \beta_i)^{(\text{itr}-1)}}{\sum_{i=1}^{C} \left[\sum_{j=1}^{p} u_{ij}^{(\text{itr}-1)} \right]^2} \quad (14)$$

where η_0 denotes the initial value, τ represents the time constant, and itr is the iteration index. Notably, the number of clusters C is dynamically updated during the clustering process.

The Lagrange multiplier method is applied to minimize the objective function in (12) with respect to U. The Lagrange function is defined as

$$L(B, U, Q) = \sum_{i=1}^{C} \sum_{j=1}^{p} u_{ij}^2 d^2(\vec{q}_j, \beta_i) - \eta \sum_{i=1}^{C} \left[\sum_{j=1}^{p} u_{ij} \right]^2 - \sum_{j=1}^{p} \lambda_j \left(\sum_{i=1}^{C} u_{ij} - 1 \right). \quad (15)$$

Then, fix B and solve

$$\frac{\partial L}{\partial u_{st}} = 2 u_{st} d^2(\vec{q}_t, \beta_s) - 2\eta \sum_{i=1}^{p} u_{si} - \lambda_t, \qquad \text{for } 1 \le s \le C, \ 1 \le t \le p \quad (16)$$

to derive an updating equation for the membership u_{st}. To solve (16), it is assumed that the membership value u_{st} is not significantly changed within two consecutive iterations. Hence, the term $\sum_{i=1}^{p} u_{si}$ in (16) is calculated by using the membership value in the previous iteration, and (16) becomes

$$u_{st} = \frac{2\eta \times N_s + \lambda_t}{2 d^2(\vec{q}_t, \beta_s)} \quad (17)$$

where $N_s = \sum_{j=1}^{p} u_{sj}$ represents the cardinality of cluster s. The cardinality is a measure of whether the considered cluster can be merged into its adjacent cluster; namely, the agglomeration process. When the cardinality N_s is below a prespecified constant ς, cluster s is discarded in the clustering process. Additionally, λ_j can be solved by substituting (17) into (13) as

$$\lambda_t = \frac{1 - \eta \sum_{s=1}^{C} \left[\frac{N_s}{d^2(\vec{q}_t, \beta_s)} \right]}{\sum_{s=1}^{C} \left[\frac{1}{d^2(\vec{q}_t, \beta_s)} \right]}. \quad (18)$$

Next, λ_t can be eliminated by using (18), after which (17) is rewritten as

$$u_{st} = \eta \frac{N_s}{d^2(\vec{q}_t, \beta_s)} + \frac{1 - \alpha \sum_{k=1}^{C} \left[\frac{N_k}{d^2(\vec{q}_t, \beta_k)} \right]}{\sum_{k=1}^{C} \left[\frac{d^2(\vec{q}_t, \beta_s)}{d^2(\vec{q}_t, \beta_k)} \right]}. \quad (19)$$

Once the membership value u_{st} is obtained, the width of the ellipsoidal shape (i.e., σ_i) at each axis is obtained by

$$\sigma_i = \sqrt{\frac{\sum_{k=1}^{p} (u_{ik})^2 (\vec{q}_k - \delta_i)^2}{\sum_{k=1}^{p} (u_{ik})^2}} \quad (20)$$

and the center of the ellipsoidal shape (i.e., δ_i) at each axis is obtained by

$$\delta_i = \frac{\sum_{k=1}^{p} (u_{ik})^2 \vec{q}_i}{\sum_{k=1}^{p} (u_{ik})^2}. \quad (21)$$

Because of SVR with Gaussian kernel function being able to be viewed as RBFNs, clustering algorithms for determining the spread width of the Gaussian function in the RBFNs are also suitable for determining GKp. Generally, the cluster shape is supposed to have a simple geometry in the input–output space, generally with an ellipsoid shape. Hence, the spread width of the ellipsoidal shape can be represented as the parameter GKp [19], [22]. When the ellipsoidal shape for each cluster is obtained, the size of the ellipsoidal shape differs among clusters. Cluster size commonly differs for most cases. In this paper, the minimal, maximal, and average sizes of clusters are used to verify the suitable range of GKp. To catch the maximum change of the estimated function (models), the minimum cluster size is suitable for representing as estimated GKp values. In the present simulation results, the minimum cluster size was demonstrated experimentally to optimize the performance.

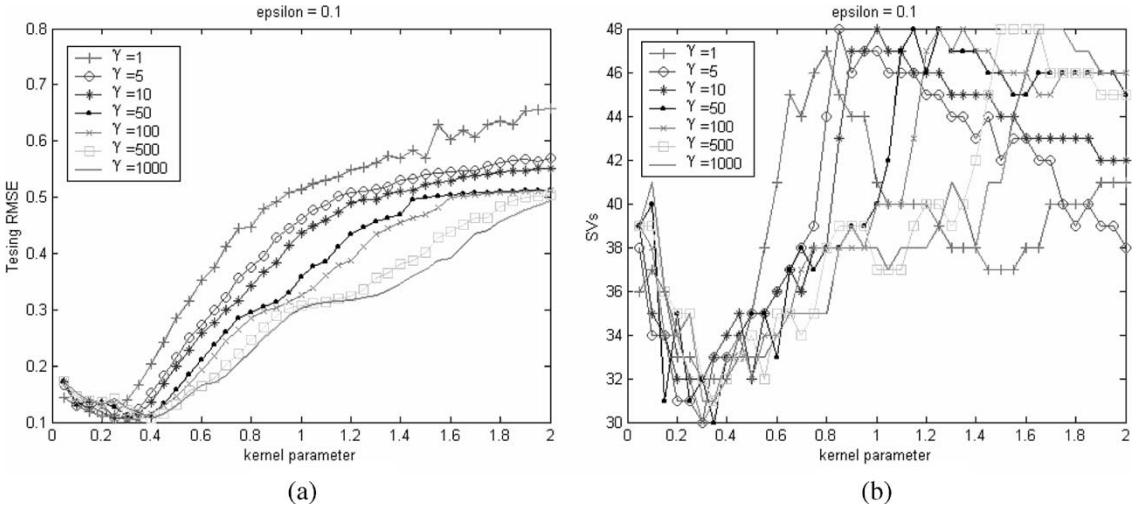


Fig. 1. (a) Testing rmse and (b) number of SVs in the SVR with different kernel parameter σ and regularization constant γ for a fixed $\varepsilon = 0.1$ are shown.

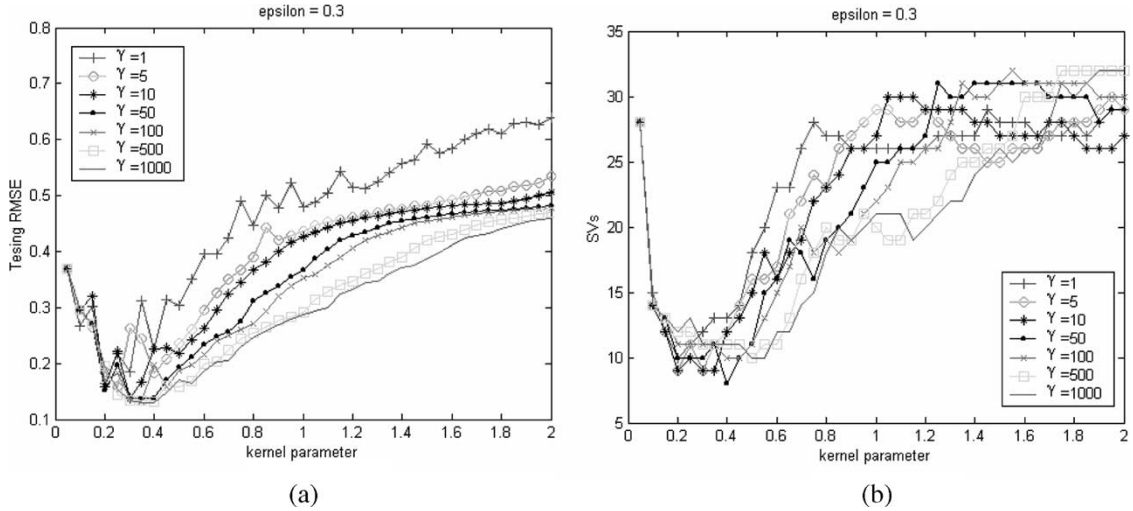


Fig. 2. (a) Testing rmse and (b) number of SVs in the SVR with different kernel parameter σ and regularization constant γ for a fixed $\varepsilon = 0.3$ are shown.

B. RSVR Approach for ε

Generally, the accuracy outputs of SVR and the number of SVs depend on the ε value in the loss function for the SVR. Additionally, the SVR with proper ε can provide a smoothing output within the $\pm\varepsilon$ constant zone based on the outputs of SVR. To select a proper ε, a method to select an ε value using SVR twice, called the RSVR approach [4], is proposed. That is, the RSVR can determine an epsilon value based on the regression errors. Because the CA clustering algorithm can obtain a good result in GKp, the RSVR uses a good initial GKp to select the epsilon value. Consequently, an epsilon is determined by the RSVR approach and stated as follows.

Step 1) The GKp is first determined using the CA clustering algorithm. The SVR with $\varepsilon = 0$ then is applied to obtain the estimated results. Consequently, the actual and expected output corresponding to the same input data were used to calculate the regression errors.

Step 2) Based on the regression errors, the appropriate ε is calculated by

$$\varepsilon = \upsilon \times \text{std}\left([\text{regression errors}]\right) \tag{22}$$

where the term std$([\text{regression errors}])$ is the standard deviation of the regression errors and υ is a constant. Generally, the constant υ was often chosen as 1.645 and 1.96, which assumed that 90% and 95% probability of regression errors fall into the $\upsilon \times$ std$([\text{regression errors}])$ regions for the Gaussian distribution [23], respectively.

Using the RSVR approach, an ε value can be obtained using (22). That is, an ε of SVR could be determined automatically based on the regression errors using the RSVR approach. In this paper, the constant υ is considered as 1.645 and 1.96. That is, 10% and 5% probability of regression errors that may be regarded as outliers are ignored when υ is chosen as 1.645 and 1.96, respectively.

IV. Simulations

This section uses four examples to evaluate the performance of the proposed method. An index, used for performance evaluation, is an rmse and is defined as

$$\text{rmse} = \sqrt{\frac{\sum_{j=1}^{N} (y_j - \hat{y}_j)^2}{N}} \tag{23}$$

where N denotes the number of the test data, y_j represents the actual output, and $\hat{y}_j$ is the output of the SVR for the jth training data.

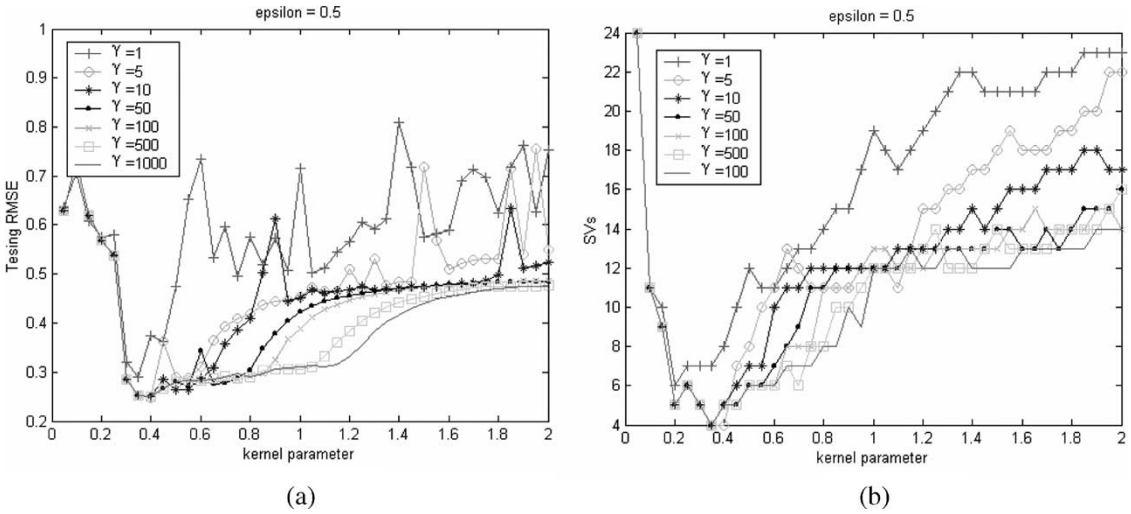


Fig. 3. (a) Testing rmse and (b) number of SVs in the SVR with different kernel parameter σ and regularization constant γ for a fixed $\varepsilon = 0.5$ are shown.

In example 1, a simple function is considered, and is defined as

$$y = x + 2 \cdot \exp(-16x^2) + \text{Noise}, \qquad x \in [-1, 1]. \tag{24}$$

One hundred one training data are generated from (24) and added Gaussian noise with $[\text{mean}, \text{variance}] = [0, 0.2]$. Furthermore, 402 data pairs are used to assess the performance of the SVR. The relation between the GKp (σ) and the regularization constant (γ) is analyzed for the fixed ε. In this case, the $\varepsilon = \{0.1, 0.3, 0.5\}$, and the testing rmse of the SVR were fixed with the different σ and γ, as shown in Figs. 1(a), 2(a), and 3(a), respectively. Moreover, the number of SVs with the different σ and γ is shown in Figs. 1(b), 2(b), and 3(b), respectively. In Figs. 1(a), 2(a), and 3(a), the testing rmse of SVR shows the same trend for different γ, and the testing rmse of SVR with larger γ is lower, among others. It matches previous results that the regularization constant should be sufficiently large. In this paper, the regularization constant γ is chosen as 1000. Figs. 1(a), 2(a), and 3(a) reveal that the lower testing rmse appears on the range of $\sigma = [0.2, 0.4]$, $\sigma = [0.25, 0.45]$, and $\sigma = [0.32, 0.55]$ for $\varepsilon = 0.1$, $\varepsilon = 0.3$ and, $\varepsilon = 0.5$, respectively. These experimental results demonstrate that the reasonable range of GKp (σ) appears to draft by increasing ε. Based on those results, the GKp and epsilon are two key factors in the SVR.

A hybrid approach for selecting GKp and epsilon is proposed. First, the GKp (σ) is determined using the CA clustering algorithm. For example 1, the required parameters are set as the initial clusters $C = 20$, $\varsigma = 3$, $\eta_0 = 3$, and $\tau = 6$ in the CA clustering algorithm. Generally, some clusters are obtained using the CA clustering algorithm. However, the spread width of the ellipsoidal shape differs among clusters. The minimal, average, and maximal width of the ellipsoidal shape for all clusters are used to represent the minimum, average, and maximum GKp (i.e., $\sigma_{\min}$, σ_{avg}, and $\sigma_{\max}$), respectively. Based on these GKp values, the epsilon is determined using the RSVR approach. That is, in step 1 of the RSVR approach, the required parameters are set to $\gamma = 1000$ and $\sigma = \{\sigma_{\min}, \sigma_{\text{avg}}, \sigma_{\max}\}$. An epsilon is then obtained via step 2 of the RSVR approach with the constant υ chosen as 1.645 and 1.96. When the parameters ε and σ are determined using the hybrid approach, the testing rmse and the number of SVs, for example 1, are tabulated in Table I. Besides, Fig. 4 reveals the contour of the testing rmse and the number of SVs. These contours are constructed using the results of SVR with equal space of GKp and epsilon value. This grid-search-like experiment procedure is used to check the results of the hybrid approach. In Fig. 4, the symbols (Xa, Xb), (Oa, Ob), and (@a, @b) are represented as the results of SVR with υ of 1.645 and 1.96 and fixed $\{\sigma_{\min}, \sigma_{\text{avg}}, \sigma_{\max}\}$, respectively. The regions of optimal hyperparameters $\{\varepsilon^*, \sigma^*\}$ of the SVR that have lower testing rmse and fewer SVs are represented as shadows in Fig. 4. From the above results, this paper proposes that the hybrid approach for selected hyperparameters of the SVR with noise are close to the region of optimal hyperparameters $\{\varepsilon^*, \sigma^*\}$.

TABLE I
TESTING RMSE AND THE NUMBER OF SVS IN THE SVR FOR EXAMPLE 1 BASED ON THE GKP AND THE EPSILON VALUE THAT ARE OBTAINED BY THE PROPOSED HYBRID APPROACH

Gaussian kernel parameter (σ)	Epsilon (ε) 90%	The number of SVs	RMSE
$\sigma_{\min} = 0.4538$	0.1715	15	0.0648
$\sigma_{avg} = 0.4958$	0.1752	15	0.0895
$\sigma_{\max} = 0.5427$	0.1894	20	0.1075
Note. The regularization constant is experimentally chosen as 1000. Time for Gkp is 7.2030 seconds/ 25 epochs. Time for the epsilon is 0.8157 second.			

Gaussian kernel parameter (σ)	Epsilon (ε) 95%	The number of SVs	RMSE
$\sigma_{\min} = 0.4538$	0.2044	11	0.0761
$\sigma_{avg} = 0.4958$	0.2088	11	0.0972
$\sigma_{\max} = 0.5427$	0.2256	12	0.1244
Note. The regularization constant is experimentally chosen as 1000. Time for Gkp is 7.2030 seconds/ 25 epochs. Time for the epsilon is 0.8158 second.			

In example 2, a sinc function is considered as

$$y = \frac{\sin(x)}{x} + \text{Noise}, \qquad x \in [-10, 10]. \tag{25}$$

One hundred one training data sets are generated from (25), and Gaussian noise is added with [0, 0.1]. A further 402 data pairs are used to evaluate the performance of the SVR. When the GKp sets

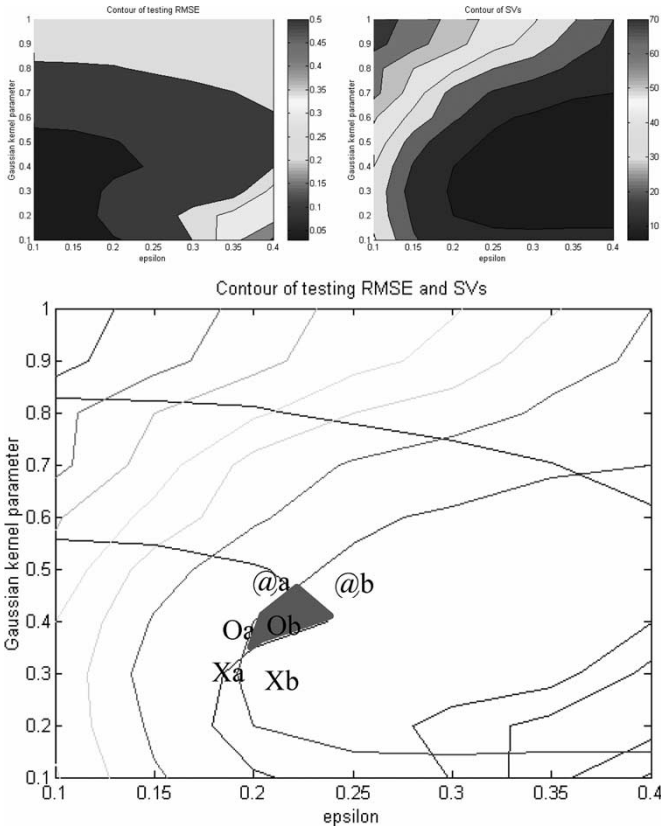


Fig. 4. Contour of the testing rmse and the number of SVs are shown for example 1. In this figure, the symbols (Xa, Xb), (Oa, Ob), and (@a, @b) are represented as the results of the SVR with υ equal to 1.645 and 1.96 for fixed $\{\sigma_{\min}, \sigma_{\text{avg}}, \sigma_{\max}\}$, respectively. The regions of "optimum" parameters $\{\varepsilon^*, \sigma^*\}$ of SVR that have lower testing rmse and less number of SVs are represented as shadows.

$\sigma = \{\sigma_{\min}, \sigma_{\text{avg}}, \sigma_{\max}\}$ are obtained using the CA clustering algorithm and the epsilon is obtained using the RSVR approach, the testing rmse and the number of SVs of SVR using the proposed hybrid approach for example 2 are shown in Table II. The contour of testing rmse and the number of SVs, for example 2, are shown in Fig. 5. In Fig. 5, the symbols (Xa, Xb), (Oa, Ob), and (@a, @b) represent the results of SVR with υ of 1.645 and 1.96 for fixed $\{\sigma_{\min}, \sigma_{\text{avg}}, \sigma_{\max}\}$, respectively. Fig. 5 displays two "suboptimal" regions (regions I and II). Region I has a lower number of SVs while the testing rmse is not lower, among others. Moreover, region II has lower testing rmse, while the SVs are not lower, among others. In those examples, the hyperparameters of SVR are determined via using the proposed hybrid approach that is still close to the "suboptimal" regions (regions I and II). Based on the results of examples 1 and 2, the SVR with $\{\sigma_{\min}\}$ (i.e., minima size of cluster) was found experimentally to optimize the performance. Moreover, the effects of the regularization constant for SVR are considered again. For examples 1 and 2, the results of the SVR with the proposed hybrid approach (i.e., GKp and epsilon) and different regularization constants are also shown in Figs. 6 and 7. Clearly, the performance of the SVR appears to have the same trend for the regularization constant is chosen to have as large a value as possible. In fact, the SVR with regularization constant $\gamma \geq 1000$ was also found experimentally to achieve the best performance.

To verify the validation of the proposed hybrid approach, another approach, which is called the VCYM approach [16], is considered for comparison. In this approach, the GKp is suggested to be $(0.2 - 0.5) * \text{range}(x)$, where $\text{range}(x)$ represents the input range

TABLE II
TESTING RMSE AND THE NUMBER OF SVS IN THE SVR FOR EXAMPLE 2 BASED ON THE GKP AND THE EPSILON VALUE THAT ARE OBTAINED BY THE PROPOSED HYBRID APPROACH

Gaussian kernel parameter (σ)	Epsilon (ε) 90%	The number of SVs	RMSE
$\sigma_{\min} = 4.0554$	0.1487	13	0.0297
$\sigma_{avg} = 4.3604$	0.1482	13	0.0286
$\sigma_{\max} = 4.7509$	0.1485	13	0.0327
Note. The regularization constant is experimentally chosen as 1000. Time for Gkp is 14.9220 seconds/ 25 epochs. Time for the epsilon is 0.9370 second.			

Gaussian kernel parameter (σ)	Epsilon (ε) 95%	The number of SVs	RMSE
$\sigma_{\min} = 4.0554$	0.1771	9	0.0468
$\sigma_{avg} = 4.3604$	0.1765	10	0.0450
$\sigma_{\max} = 4.7509$	0.1770	11	0.0473
Note. The regularization constant is experimentally chosen as 1000. Time for Gkp is 14.9220 seconds/ 25 epochs. Time for the epsilon is 0.9372 second.			

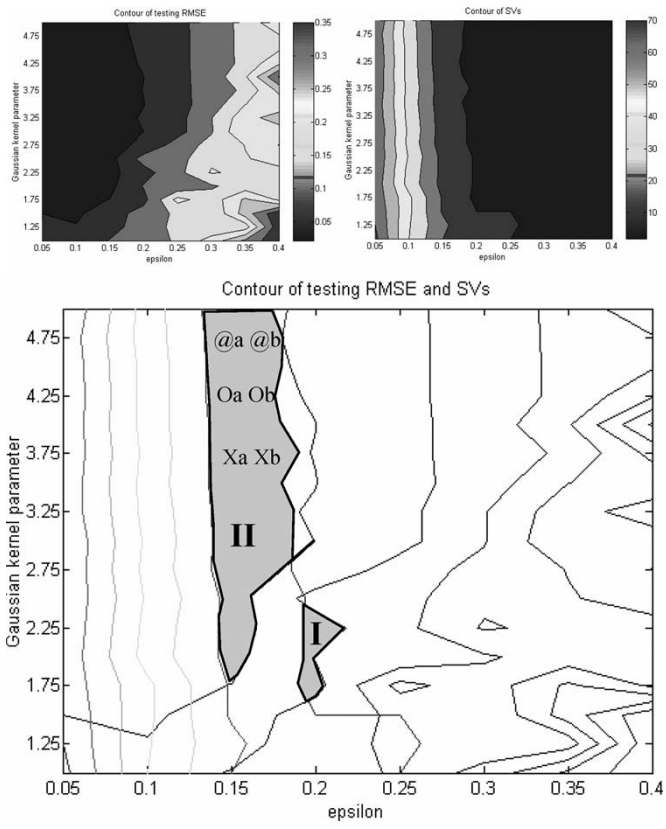
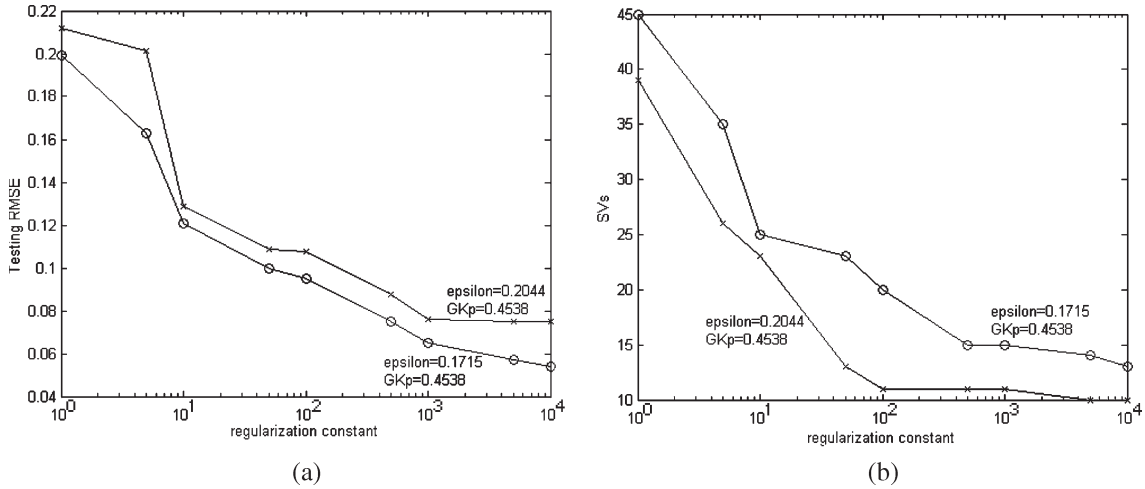


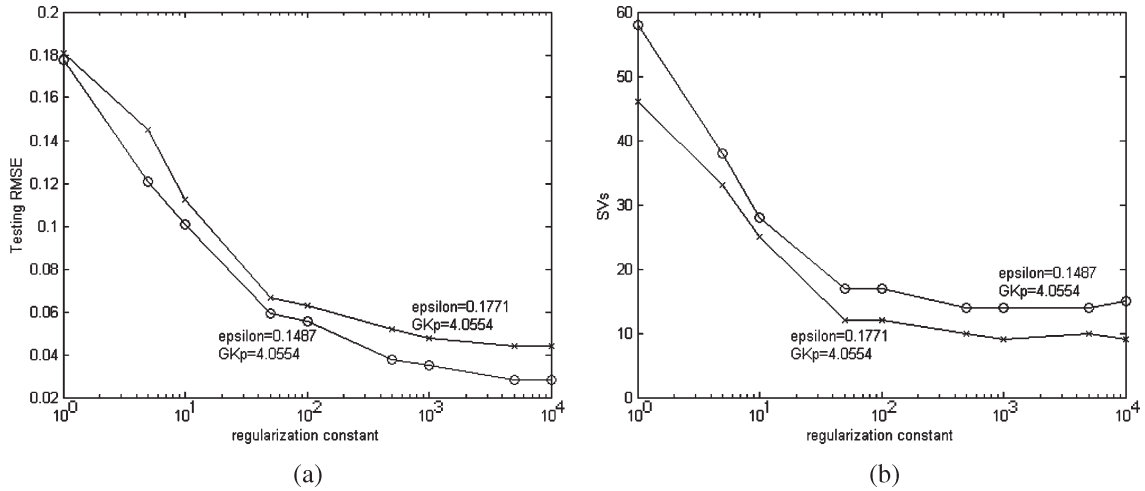
Fig. 5. Contour of the testing rmse and the number of SVs are shown for example 2. The region I has lower SVs while the testing rmse is not lower, among others. The region II has lower testing rmse while the number of SVs is not lower, among others.

of the training/test data. Moreover, the epsilon is suggested to be $3(N_{\text{noise}})\sqrt{\ln n/n}$, where N_{noise} denotes the standard deviation of noise and n represents the number of training data. Based on this



(a) (b)

Fig. 6. (a) Testing rmse and (b) number of SVs of SVR with the proposed hybrid approach and different regularization constants are shown for example 1.



(a) (b)

Fig. 7. (a) Testing rmse and (b) number of SVs of SVR with the proposed hybrid approach and different regularization constants are shown for example 2.

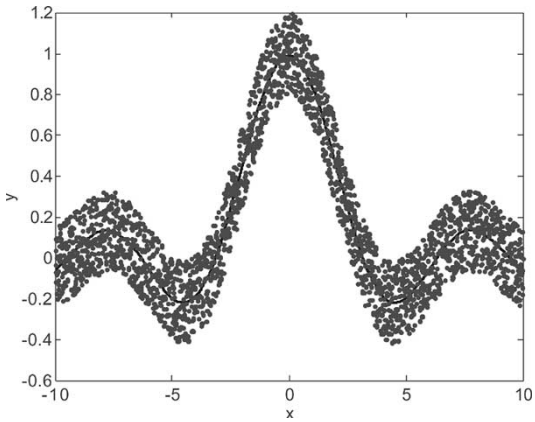


Fig. 8. Final results of SVR with the proposed approach under data points $= 2500$.

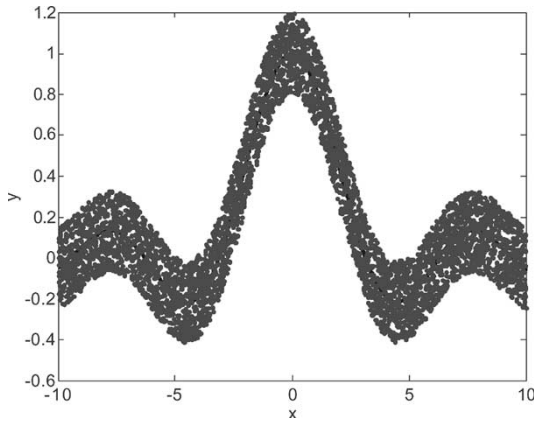


Fig. 9. Final results of the SVR with the proposed approach under data points $= 5000$, under data points $= 5000$, and epsilon value $= 0.1948$ (90%).

approach, the GKp are calculated as the range of [0.4, 1], [4, 10], and [2−5] for examples 1 and 2, respectively. However, the range of GKp exceeds that with the proposed hybrid approach. Based on known N_{noise}, the epsilon values are calculated as 0.0962 for examples 1 and 2, respectively. Notably, the same epsilon value is obtained for the same standard deviation of noise and n. The regularization constant is also suggested to be three times the standard deviation of the training response values (output). According to this suggestion, the regularization constants are calculated as 2.6427 and 1.2323 for examples 1 and 2, respectively. Based on the above approach, the testing rmse of SVR with $\{\varepsilon, \sigma, \gamma\} = \{0.0962, 0.4, 2.6427\}$ and

TABLE III
COMPARISON RESULTS ON TIME AND EPOCHS

Method	Time (Second)/Epochs for the GKp and the epsilon	Final Results
Sinc with 2500 data points via proposed method	Time for GKp= 9583/ 25 epochs. Time for epsilon= 726.	Epsilon=0.1971, C=1000, Gkp=4.1, Testing RMSE =0.0059, SVs = 40.
Sinc with 2500 data points via the grid search	Epsilon = {0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4}, Gkp = { 1.25 1.75 2.25 2.75 3.25 3.75 4.25 4.75}, Time for the Gkp and the epsilon = 16576	Epsilon = 0.2, C=1000, Gkp = 4.25, Testing RMSE = 0.0071, SVs = 21.
Sinc with 2500 data points via the grid search	Epsilon = {0.05 0.15 0.25 0.35} Gkp = { 1.25 2 2.75 3.5 4.25 5} Time for the Gkp and the epsilon = 8649.7	Epsilon = 0.25, C=1000, Gkp = 4.25, Testing RMSE = 0.0426, SVs = 15.
Sinc with 5000 data points via proposed method ($\upsilon = 1.645$)	Time for GKp = 34350/ 25 epochs. Time for epsilon = 3265.2.	Epsilon=0.1948, C=1000, Gkp=4.15, Testing RMSE =0.0033, SVs =50.
Sinc with 5000 data points via proposed method ($\upsilon = 1.96$)	Time for GKp = 34350/ 25 epochs. Time for epsilon = 3265.2.	Epsilon=0.2309, C=1000, Gkp=4.15, Testing RMSE =0.031, SVs =19.
Sinc with 5000 data points via the grid search	Epsilon = {0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4}, Gkp = {1.25 1.75 2.25 2.75 3.25 3.75 4.25 4.75}, Time for the Gkp and the epsilon = 54081	Epsilon = 0.2, C=1000, Gkp = 4.25, Testing RMSE = 0.0056, SVs = 31.
Sinc with 5000 data points via the grid search	Epsilon = {0.05 0.15 0.25 0.35} Gkp = {1.25 2 2.75 3.5 4.25 5} Time for the Gkp and the epsilon = 27751	Epsilon = 0.25, C=1000, Gkp = 4.25, Testing RMSE = 0.0426, SVs = 15.
Example 3 with the proposed method ($\upsilon = 1.645$)	Time for GKp= 89.4841/ 25 epochs. Time for epsilon = 2.1044.	Epsilon=0.9335, C=1000, Gkp ($\sigma_{\min}$) = 4.6813, RMSE = 0.7059,
Example 3 with the proposed method ($\upsilon = 1.96$)	Time for GKp= 89.4841/ 25 epochs. Time for epsilon = 2.1044.	Epsilon=1.1122, C=1000, Gkp ($\sigma_{\min}$) = 4.6813, RMSE = 1.5412,
Example 3 with the VCYM method	Time for GKp,epsilon and regularization constant < 1 second	Epsilon=0.2571 C=9.7005, Gkp = 4.8 RMSE = 1.9541,
Example 3 with the grid search	Epsilon = {0.1 0.3 0.5 0.7 0.9 1.1 1.3 1.5} Gkp = {2 2.5 3 3.5 4 4.5 5 5.5} Time for the Gkp and the epsilon = 126.7190	Epsilon = 1.1, C=1000, Gkp = 4.5, RMSE = 1.4676
Example 4 with the proposed method ($\upsilon = 1.645$)	Time for GKp= 85.2531/ 25 epochs. Time for epsilon = 1.0346.	Epsilon=0.3808, C=1000, Gkp ($\sigma_{\min}$) = 2.4481, Testing RMSE =0.8180, SVs =41.
Example 4 with the proposed method ($\upsilon = 1.96$)	Time for GKp= 85.2531/ 25 epochs. Time for epsilon = 1.0346.	Epsilon=0.4537, C=1000, Gkp ($\sigma_{\min}$) = 2.4481, Testing RMSE =0.9210, SVs =52.
Example 4 with the VCYM method	Time for GKp,epsilon and regularization constant < 1 seconds	Epsilon=0.2591 C=5.1787, Gkp = 1.8302, Testing RMSE =0.8791, SVs =91.
Example 4 with the grid search	Epsilon = {1.5 1.7 1.9 2.1 2.3 2.5 2.7 2.9}, Gkp = {0.2 0.25 0.3 0.35 0.4 0.45 0.5 0.55} Time for the Gkp and the epsilon = 424.3230	Epsilon = 0.4, C=1000, Gkp = 2.5, Testing RMSE = 0.8835, SVs = 45.

$\{\varepsilon, \sigma, \gamma\} = \{0.0962, 4, 1.2323\}$ for examples 1 and 2 are obtained as 0.1664 and 0.0305, respectively. Similarly, the SVs of examples 1 and 2 are obtained as 30 and 56, respectively. These results clearly indicate that the hyperparameter set $\{\varepsilon, \sigma\}$ identified using the VCYM approach is far from the "optimal" hyperparameter set $\{\varepsilon^*, \sigma^*\}$ and the proposed hybrid approach. Furthermore, the final results of SVR with the VCYM approach are worse than for the proposed hybrid approach. Besides, comparing the proposed approach with the extreme learning machine (ELM) proposed by G.-B. Huang is one case. This case is to approximate the "SinC" function (this example and the results of the ELM come from [24]). The MATLAB software was used to simulate the CA clustering algorithm for GKp. In the RSVR, the SVMTorch software is used to estimate the epsilon values. The computer is P4 2.4G. In this simulation, there are 2500 data points in case 1 and 5000 data points in case 2. The final results of the SVR with the proposed approach for case 1 and case 2-1 are shown in Figs. 8, and 9, respectively. Moreover, the hyperparameters of the SVR, the number of support vectors, and rmse are listed in Table III. When considering the SVR in specific feedforward networks, the computational time of the proposed approach is larger than the ELM, from the results of Fig. 9. However, the rmse of the proposed approach (0.0033) is less than that of the ELM (0.0097) [24]. How to reduce computational time is a further topic in the SVR. It notes that, based on the determined parameters of the proposed approach, the computational time for the final results of the SVR is smaller than the computational time for epsilon with the RSVR— that is, the main computational time the proposed approach spends in the determination of hyperparameters. Besides, this work uses the CA clustering algorithm once and the SVR once to obtain the hyperparameters. Final results can be obtained by using the SVR once with the determined hyperparameters. On the other hand, a grid-search approach is also applied in two cases. Additionally, the parameters of epsilon and GKp are considered in this study. However, Lin's grid search [25] is mainly focused on the parameter selection of regularization constant and GKp. It is worthy to note that my proposed methods are different from the simple grid-search method. That is, the proposed method can overcome the drawback of the validation-set approaches that need a grid process involving expensive computation. At the same time, the proposed

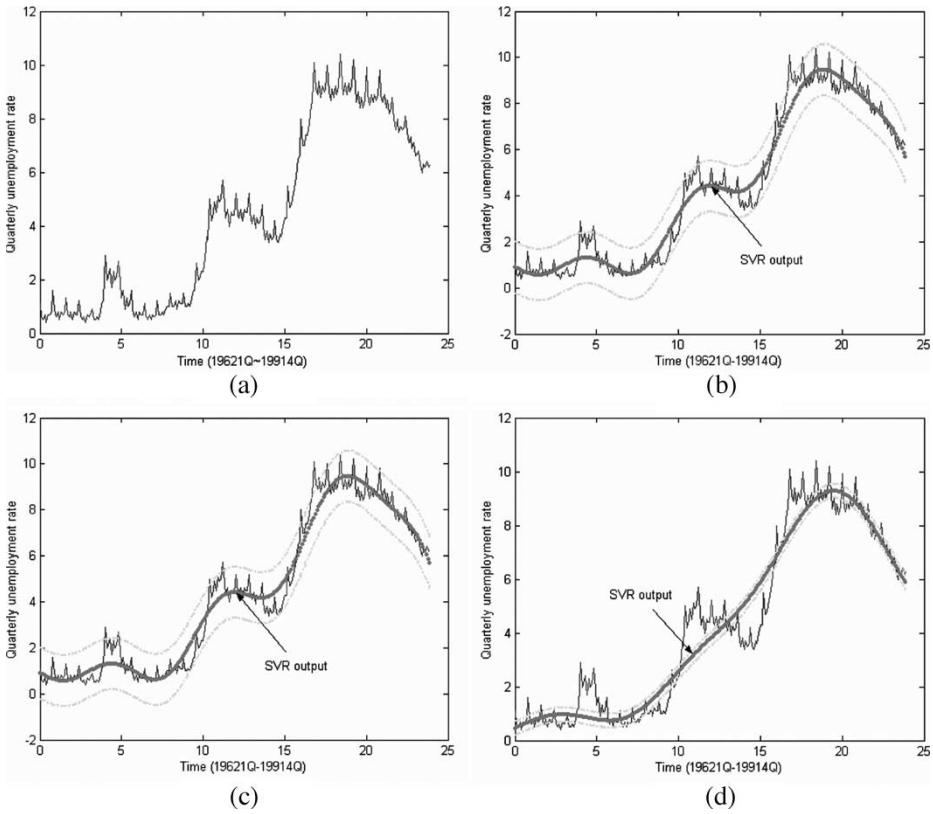


Fig. 10. (a) Time series of quarterly unemployment rate for West Germany (from the first quarter of 1962 to the fourth quarter of 1991) is shown, (b) final results of the SVR with $\{\varepsilon, \sigma_{\min}, \gamma\} = \{1.1122, 4.6813, 1000\}$ for constant $\upsilon = 1.96$, which are obtained by the proposed hybrid approach, (c) final results of the SVR with $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.9335, 4.6813, 1000\}$ for constant $\upsilon = 1.645$, which are obtained by the proposed hybrid approach, and (d) final results of the SVR with $\{\varepsilon, \sigma, \gamma\} = \{0.2571, 4.8, 9.7005\}$, which are obtained by the VCYM approach.

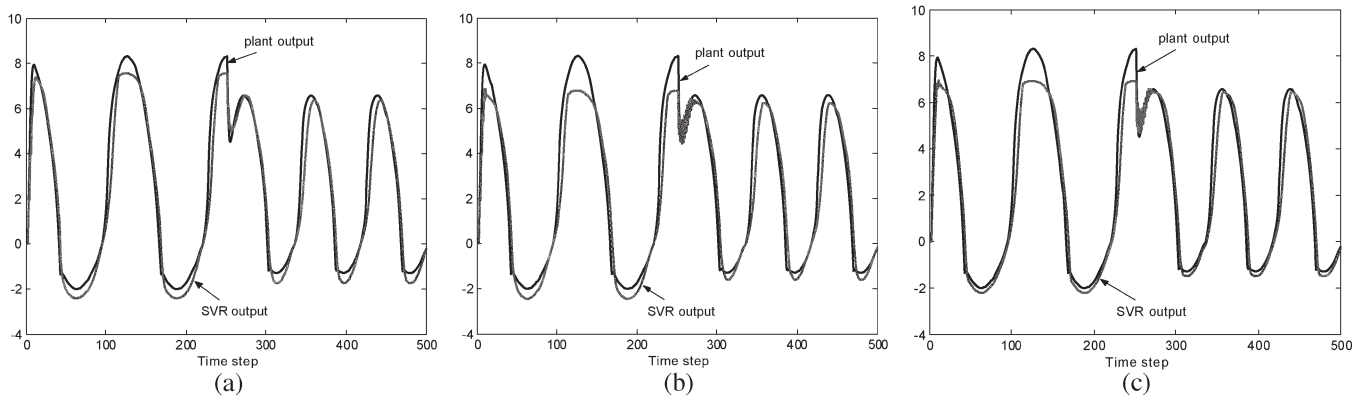


Fig. 11. (a) Prediction result of the SVR with $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.3808, 2.4481, 1000\}$ for $\upsilon = 1.645$, (b) prediction result of the SVR with $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.4537, 2.4484, 1000\}$, for $\upsilon = 1.96$, (c) prediction result of the SVR with $\{\varepsilon, \sigma, \gamma\} = 0.2591, 1.8302, 5.1787$, which are obtained by the VCYM approach.

method can also overcome the drawback of trial and error. These results are all shown in Table III.

In example 3, a real data set (namely, the time series of quarterly unemployment rate for West Germany from the first quarter of 1962 to the fourth quarter of 1991) is considered [26]. There are 240 data points in example 3. The original data set can be downloaded from http://www-personal.buseco.monash.edu.au/~hyndman/TSDL/, and is shown in Fig. 10(a). Based on the proposed hybrid approach, the hyperparameter sets are obtained as $\{\varepsilon, \sigma_{\min}, \gamma\} = \{1.1122, 4.6813, 1000\}$ and $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.9335, 4.6813, 1000\}$ for constant $\upsilon = 1.96$ and $\upsilon = 1.645$, respectively. Additionally, the hyperparameter sets using VCYM approach are obtained as $\{\varepsilon, \sigma, \gamma\} = \{0.2571, 4.8, 9.7005\}$. Table III lists the hyperparameter sets using the proposed hybrid approach and the VCYM approach. Figs. 10(b) and 11(d) show the final results of SVR based on those hyperparameter sets with the proposed approach and the VCYM approach. In those figures, the SVR using the proposed hybrid approach can be performed using a smoothing output and its epsilon zone. It is obvious that most of the training data sets are included in the zone of the SVR output plus and minus epsilon. That is, the errors between the SVR output and training data points are less than and equal to epsilon. In this case, the SVR output can catch the trend of the training data set (time series) within the range of epsilon. Additionally, the local property of the training data set is missed when the SVR with the VCYM approach is used. In

the meantime, the numbers of SVs are larger than the SVR using the proposed hybrid approach. Consequently, the SVR using the proposed hybrid approach is superior to the VCYM approach. It notes that in examples 1, 2, and 4, the true functions are known. Then, another testing data set is generated to evaluate the performance (i.e., testing rmse) of the SVR. On the other hand, because the true functions are unknown in example 3, cross validation is needed in this example. The cross-validation method with five folds is made for evaluating the performance of the proposed approach and other approaches in example 3. That is, 240 data points are randomly separated into five data subsets (48 points). These data subsets include four training subsets and one testing subset. Hence, there are five combinations. The training subsets are applied to train the SVR. The testing rmse is obtained using the testing subset. Hence, the cross validation with five folds can generate five testing rmses. Finally, the rmse is obtained by averaging the five testing rmses. These results are also shown in Table III.

In example 4, the nonlinear plant is considered and expressed as

$$y_p(k+1) = \frac{y_p(k)\left(y_p(k-1)+2\right)\left(y_p(k)+2.5\right)}{8.5+\left(y_p(k)\right)^2+\left(y_p(k-1)\right)^2} + u(k) \quad (26)$$

where $y_p(k)$ is the output of the plant at the kth time step and $u(k)$ is the plant input, which is a uniformly bounded function of time. The plant is stable at $u(k) \in [-2, 2]$. That is, 301 points of the training data are generated by uniform distribution with $[-2, 2]$. Let the identification model be in the form of

$$y_{pi}(k+1) = \text{SVR}\left(y_p(k), y_p(k-1)\right) + u(k) \quad (27)$$

where $\text{SVR}(y_p(k), y_p(k-1))$ represents the SVR with two inputs $y_p(k)$ and $y_p(k-1)$ and output $y_{pi}(k+1)$ for system identification. After the SVR is trained, its prediction power is tested for the input $u(k) = 2\cos(2\pi k * 0.008)$, $k \leq 250$, and $u(k) = 1.2\sin(2\pi k * 0.012)$, $250 < k \leq 500$. Based on the proposed approach, the hyperparameter sets are obtained as $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.3808, 2.4481, 1000\}$ and $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.4537, 2.4481, 1000\}$ for $\upsilon = 1.645$ and $\upsilon = 1.96$, respectively. According the above hyperparameter sets, the prediction results of SVR are show in Fig. 11(a) and (b) . Additionally, the performance (prediction) and the number of SVs for the SVR are tabulated in Table III. Based on the VCYM approach, the hyperparameter sets are obtained as $\{\varepsilon, \sigma, \gamma\} = \{0.2591, 1.8302, 5.1787\}$. The prediction result of the SVR with the VCYM approach is also shown in Fig. 11(c). Additionally, the performance and the number of SVs for the SVR are also tabulated in Table III. According to the above results, it is obvious that the performance of SVR with the proposed approach (i.e., $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.3808, 2.4481, 1000\}$ for $\upsilon = 1.645$) is better than the SVR with the VCYM approach. Moreover, the number of SVs using the SVR with the proposed approach is smaller than the SVR with the VCYM approach. Although the performance of SVR with the proposed approach (i.e., $\{\varepsilon, \sigma_{\min}, \gamma\} = \{0.4537, 2.4481, 1000\}$ for $\upsilon = 1.96$) is worse than the SVR with the VCYM approach, the number of SVs using the proposed approach is smaller than those using the VCYM approach.

Learning time/epoch is added in this study. The MATLAB software is used to simulate the CA clustering algorithm for GKp. In the RSVR, the SVMTorch software is used to estimate the epsilon values. The SVR of the grid search is also to use SVMTorch. The computer is P4 2.4G. First, the computational time for the GKp with the CA clustering algorithm and the computational time for the epsilon with the RSVR in examples 1 and 2 are shown in Tables I and II, respectively. Second, Table III shows the comparison results for case 1, case 2, and examples $3 \sim 4$ with the proposed method, the grid search, and the VCYM. Table III shows the main computational time of the proposed approach and the time the grid search spends in the determination of hyperparameters. From Table III, if there is no prior knowledge, the hybrid approach has less computational time for the hyperparameters than the grid search. On the contrary, the grid search maybe faster than the hybrid approach. However, when there is a lack prior knowledge on the range of parameters, the time spent on finding the appropriate parameters is long with the grid search. Hence, this study proposes the hybrid approach to use the CA clustering algorithm once and the SVR once to obtain the hyperparameters. Final results can be obtained by using the SVR once with the determined hyperparameters. Hence, the proposed method can overcome the drawback of the validation-set approaches that need a grid process involving expensive computation. That is, when there is no prior knowledge on the range of parameters, the grid search needs some trial-and-error procedure. Hence, the proposed hybrid approach can also overcome the drawback of trial and error. Besides, the VCYM approach has a fast computational time for the hyperparameters than the hybrid approach. However, from Table III, it is obvious that the performance of the SVR with the proposed approach under $\upsilon = 1.645$ is better than the SVR with the VCYM approach. Moreover, the number of SVs using the SVR with proposed approach is smaller than the SVR with the VCYM approach. Although the performance of the SVR with the proposed approach under $\upsilon = 1.96$ is worse than the SVR with the VCYM approach, the number of SVs using the proposed approach is smaller than those using the VCYM approach. Hence, the proposed method can overcome the drawback of the validation-set approaches that need an iterative process involving expensive computation. At the same time, the proposed method can also overcome the drawback of trial and error or prior experience.

V. Conclusion

To select the hyperparameters of SVR, this paper presents a novel hybrid approach to determine the values of GKp and epsilon. First, the GKp is selected using the CA clustering algorithm. Second, an RSVR approach that relies on the standard deviation of the training error is proposed to obtain an epsilon in the loss function. Based on the efficient hybrid approach for selecting the hyperparameters, the results of the proposed method can directly obtain a near "suboptimal" hyperparameter region. Additionally, a time series of quarterly unemployment rates for West Germany from the first quarter of 1962 to the fourth quarter of 1991 is considered. Based on the simulation results, the final results of the SVR using the proposed approach have lower rmse and fewer SVs than those of the SVR using the VCYM approach.

Acknowledgment

The author would like to thank Dr. C. C. Chuang for the discussions.

References

[1] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[2] J. A. K. Suykens, J. Vandewalle, and B. D. Moor, "Optimal control by least squares support vector machines," *Neural Netw.*, vol. 14, no. 1, pp. 23–25, Jan. 2001.

[3] S. Mukherjee, E. Osuna, and F. Girosi, "Nonlinear prediction of chaotic time series using a support vector machine," in *Neural Networks Signal Processing (NNSP)*, Amelia Island, FL, 1997, pp. 24–26.

[4] J. T. Jeng, C. C. Chuang, and S. F. Su, "Support vector interval regression networks for interval regression analysis," *Fuzzy Set Syst.*, vol. 138, no. 2, pp. 283–300, Sep. 2003.

[5] J. T. Jeng, C. C. Chuang, and C. C. Hsiao, "Support vector machine for annealing robust fuzzy neural network with outlier," in *Proc. 9th Nat. Conf. Fuzzy Theory and Applications*, Taoyuan, Taiwan, 2001, pp. 553–558.

[6] C. C. Chuang, J. T. Jeng, and P. T. Lin, "Annealing robust radial basis function networks for function approximation with outliers," *Neurocomputing*, vol. 56, no. 1, pp. 123–139, Jan. 2004.

[7] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, Budapest, Hungary, Jul. 25–29, 2004, pp. 985–990.

[8] F. E. H. Tay and L. Cao, "Application of support vector machines in financial time series forecasting," *Omega—Int. J. Manag. Sci.*, vol. 29, no. 4, pp. 309–317, Aug. 2001.

[9] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ: Prentice-Hall, 1999.

[10] C. C. Chuang, S. F. Su, J. T. Jeng, and C. C. Hsiao, "Robust support vector regression networks for function approximation with outliers," *IEEE Trans. Neural Netw.*, vol. 13, no. 6, pp. 1322–1330, Nov. 2002.

[11] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Mach. Learn.*, vol. 46, no. 1–3, pp. 131–159, 2000.

[12] K. Duan, S. S. Keerthi, and A. N. Poo, "Evaluation of simple performance measures for tuning SVM hyperparameters," Nat. Univ. Singapore, Singapore, Tech. Rep. CD-01-11, 2001.

[13] B. Schölkopf, P. Bartlett, A. Smola, and R. Williamson, "Support vector regression with automatic accuracy control," in *Proc. Int. Conf. Artificial Neural Networks (ICANN), Perspectives Neural Computing*, Skövde, Sweden, 1998, pp. 111–116.

[14] A. Chalimourda, B. Scholkopf *et al.*, "Experimentally optimal ν in SVR for different noise models and parameter settings," *Neural Netw.*, vol. 17, no. 1, pp. 127–141, Jan. 2004.

[15] D. Mattera and S. Haykin, "Support vector machines for dynamic reconstruction of a chaotic system," in *Advances in Kernel Methods: Support Vector Machine*, B. Schölkopf, J. Burges, and A. Smola, Eds. Cambridge, MA: MIT Press, 1999.

[16] V. Cherkarsky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural Netw.*, vol. 17, no. 1, pp. 113–126, Jan. 2004.

[17] V. Cherkarsky *et al.*, "Selection of meta-parameters for support vector regression," in *Proc. Int. Conf. Artificial Neural Networks*, Madrid, Spain, Aug. 28–30, 2002, pp. 687–693.

[18] C. W. Wu, J. L. Chen, and J. W. Wang, "Self-organizing mountain method for clustering," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, Tucson, AZ, 2001, vol. 4, pp. 2434–2438.

[19] C. T. Lin and C. S. G. Lee, *Neural Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[20] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[21] H. Frigui and R. Krishnapuram, "A robust competitive clustering algorithm with applications in computer vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 450–465, May 1999.

[22] J. A. Dickerson and B. Kosko, "Fuzzy function approximation with ellipsoidal rules," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 24, no. 4, pp. 542–560, Aug. 1996.

[23] M. Evans, N. Hastings, and B. Peacock, *Statistical Distributions*. New York: Wiley, 1993.

[24] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine," School Elect. Electron. Eng., Nanyang Technol. Univ., Singapore, Tech. Rep. ICIS/03/2004, Jan. 2004.

[25] J.-H. Lee and C.-J. Lin. (2000, Nov.). Automatic model selection for support vector machines. Dept. Comput. Sci. Inf. Eng., Nat. Taiwan Univ., Taipei. [Online]. Available: http://www.csie.ntu.edu.tw/cjlin/papers/modelselect.ps.gz

[26] J. R. Poo, *Computer-Aided: Introduction to Econometrics*. Berlin, Germany: Springer-Verlag, 2003.