IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 11, 2018, accepted November 20, 2018, date of publication November 29, 2018,
date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2883433

Robust Clustered Support Vector Machine With Applications to Modeling of Practical Processes

XIN-JIANG LU[1], (Member, IEEE), TE-TE HU[1], YI ZHANG[1], AND BIN FAN[1,2]
[1]State Key Laboratory of High Performance Complex Manufacturing, School of Mechanical and Electrical Engineering, Central South University, Hunan 410083, China
[2]College of Mechanical and Electrical Engineering, Hunan City University, Yiyang 413002, China
Corresponding authors: Xin-Jiang Lu (luxj@csu.edu.cn) and Yi Zhang (zhangyicsu@csu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51675539 and in part by the Project of State Key Laboratory of High Performance Complex Manufacturing under Grant ZZYJKT2018-17.

ABSTRACT Real datasets are often distributed nonlinearly. Although many least squares support vector machine (LS-SVM) methods have successfully modeled this kind of data using a divide-and-conquer strategy, they are often ineffective when nonlinear data are subject to noise due to a lack of robustness within each sub-model. In this paper, a robust clustered LS-SVM is proposed to model this type of data. First, the clustering method is used to divide the sample data into several sub-datasets. A local robust LS-SVM model is then developed to capture the local dynamics of the corresponding sub-dataset and to be robust to noise. Subsequently, a global regularization is constructed to intelligently coordinate all local models. These new features ensure that the global model is smooth and continuous and has a good generalization and robustness. Through the use of both artificial and real cases, the effectiveness of the proposed robust clustered LS-SVM is demonstrated.

INDEX TERMS Robust LS-SVM, robust modeling, cluster, noise, nonlinearly distributed data.

I. INTRODUCTION

Many systems cannot be characterized according to specific models, but can be characterized by a model through data analysis [1]–[4]. These systems are often strongly nonlinear and work in a large operating region, which leads to nonlinear distribution of their data in clusters due to limited clusters divided. In particular, the systems are often hybrid, composed by many nonlinear dynamic sub-processes or sub-systems that have different physical rules. This means that the data of these systems or processes incorporate many nonlinear dynamic behaviors and these nonlinear dynamic behaviors are often nonlinear in different clusters. In this way, these data are often distributed nonlinearly and contaminated by noise from different sources including sampling errors, modeling errors, measurement errors, and operational errors. For example, in the semiconductor packaging industry, the curing process requires multiple thermal zones for different temperature settings and heat transfer is very complex, including conduction, convection, and radiation. As a practical system, noise from heat loss due to the opening and closing of the oven door, etc., is inevitable [5], [6]. Modeling this kind of system has been a challenge in the machine learning community.

Many data-driven methods have been developed to model such systems in recent years. A well-known modeling method is the support vector machine (SVM) [7], which is a maximal margin classifier derived under the framework of structural risk minimization (SRM). In order to improve the modeling performance of the standard SVM, many methods are developed for various applications, including the v-SVM [8], [9], the linear programming SVM [10], the least squares SVM (LS-SVM) [11], the twin SVM [12], the fisher-regularized SVM [13], and the fuzzy SVM [14]. Among these, the LS-SVM is a particularly popular data-driven tool, with great success in applications due to its simple learning algorithm and low computation cost [15]–[17]. It takes into account both structural risk and empirical risk and is more computationally attractive as a method for solving a set of linear equations.

The standard LS-SVM method can effectively model a nonlinear system [18]. However, it may not be applicable to strongly nonlinear systems working in a large operating region. To address this issue, a "divide-and-conquer" method [19]–[21] is often adopted. One common strategy is to employ a series of different kernels to represent local information during classification [22]–[26]. As multiple kernels are selected, the simplest way to combine them is by averaging. However, in real-world applications, it may be inappropriate to assign the participant kernels with the same weight [27]. An alternative approach is the clustered SVM method [28], [29], which first divides the data into subsets using the clustering method. The sequences of local LS-SVM models are applied to capture local information, and then a final estimation is obtained by combining the outputs of all the sub-models. Although these ''divide-and-conquer'' methods can model nonlinearly distributed data, they typically neglect the relationships between local models which are prone to local minima. The clustered SVM model with global weighting [30], [31] is the exception. In each cluster, a linear support vector machine model is trained to avoid over-fitting of each local model, a global regularization is added to coordinate the sub models. However, less attention is paid at the robustness of each sub-model, thus the overall model is sensitive to noise. An effective LS-SVM method is still needed for better modeling of nonlinear data.

Recently there has been focus on improving the robustness of the LS-SVM method [32]–[50]. One approach is to directly eliminate the samples that are likely outliers [42]. First, a threshold would be set through the training error of all the samples. Then, constraints in the regression problem may guarantee the outliers are eliminated during the training procedure. For example, a simplification SVM algorithm was presented to reduce the number of support vectors [35], [51], and a hybrid robust SVM for removing outliers in training data was proposed [32]. However, it is difficult to determine such a threshold unless there is sufficient prior knowledge of the training samples. Another approach using probabilistic SVM methods isolated the noise characteristics of the underlying training data and included them in the model [33], [34], [36], [45]. Instead of estimating the specific model output, the probabilistic SVM methods aim at developing a confidence interval of the model output. For example, a Dempster-Shafer theory-based LS-SVM method was presented to improve the robustness of modeling [24], [45] and the confidence intervals of the LS-SVM for regression were derived [33]. Another common approach is the fuzzy SVM method [14], [24], [37]–[41], [43], [44], which assigns different weights to different samples. If a sample is judged as noise, it is given a small weight. In this way, the impact of noise can be reduced effectively and the robustness of the model can be improved. For example, a kernel fuzzy c-means clustering-based fuzzy SVM algorithm was developed to deal with the classification problems with outliers or noises [46], and a heuristic strategy for automatically generating fuzzy memberships of training data was also presented [37]. Although these methods can improve the robustness of the LS-SVM, they have not yet been used to analyze data with a nonlinear distribution.

In this paper, the innovation of this paper lies in a robust clustered LS-SVM method is proposed to model data with a nonlinear distribution subject to noise. Unlike existing LS-SVM methods, the proposed method considers both the nonlinear distribution of the data and the influence of noise, as well as the robustness of each sub-model, and provides a global coordination for all sub-models using global regularization. First, the clustering method [52], [58], [59] is used to divide the sample data into several sub-datasets. A local robust LS-SVM captures the local dynamics of the sub-dataset and accommodates for noise. A global regularization is then constructed to coordinate all local models, ensuring that the global model integrates all local sub-models in a way that is smooth, continuous, and robust. Finally, in order to prove the robustness of the proposed method, we use the nonzero mean Gaussian noise and uniform noise to verify the experimental results. The robustness of this method is that the model can be well predicted when the model is subjected to the noise or the outlier. This method can effectively model nonlinear data subject to noise.

II. TRADITIONAL LS-SVM

Interpreting data from practical experiments are often difficult because of both the inherent process complexity and the noise in data recording. Here, the curing process for chip manufacture [6] is used as an example to demonstrate common modeling problems. As shown in Figure 1, two heaters are embedded in the heating block and heated by the same power supply, which in turn heats the integrated circuit (IC) placed on the lead frame (LF). The working plate moves up and down to modify the thermal conditions at the IC. There is also a system at the bottom of the oven to rapidly cool the unit. The oven is filled with nitrogen to prevent oxidation. The heat transfer during the curing process is very complex since it includes conduction, convection, and radiation. In addition, some of the boundary conditions are unknown and noise is inevitable from heat loss through events such as opening and closing the oven door. This kind of system can be described as follows:

$$y = f(x, \varepsilon) \tag{1}$$

where x and y are the input and output of the system, respectively, f is an unknown nonlinear function, and ε may be either random Gaussian or non-Gaussian noise.

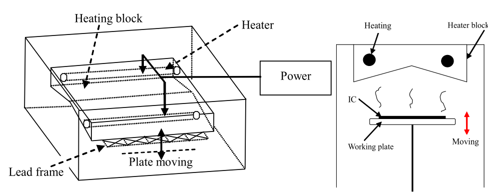


FIGURE 1. Snap curing oven system.

This unknown system is often represented by the following LS-SVM model:

$$y = \omega^T \phi(x) + b \tag{2}$$

where ϕ is an unknown projection function that maps the inputs to a higher dimensional feature space, and ω and b are the weight vector and bias, respectively.

This model can be derived by solving the following optimization problem from the given training set $\{x_l, y_l\}_{l=1}^N$

$$\min_{\omega,b,e_l} J(\omega, b, e_l) = \frac{1}{2}\|\omega\|^2 + \frac{\gamma}{2}\sum_{l=1}^{N} e_l^2$$
$$s.t.\ y_l = \omega^T\phi(x_l) + b + e_l, \quad l = 1, \ldots, N \tag{3}$$

where e is the modeling error and γ is the regularization parameter used to control the trade-off between the approximation accuracy and the model complexity. In order to solve Eq. (3), the following Lagrangian function is constructed:

$$L(\omega, b, e_l; \alpha_l) = J(\omega, b, e_l) - \sum_{l=1}^{N}\alpha_l(\omega^T\phi(x_l) + b + e_l - y_l) \tag{4}$$

where α_l is the Lagrange multipliers. The conditions for optimization are given by:

$$\begin{cases} \dfrac{\partial L}{\partial \omega} = 0 \rightarrow \omega = \sum_{l=1}^{N}\alpha_l\phi(x_l) \\ \dfrac{\partial L}{\partial b} = 0 \rightarrow \sum_{l=1}^{N}\alpha_l = 0 \\ \dfrac{\partial L}{\partial e_l} = 0 \rightarrow \alpha_l = \gamma e_l & l = 1, \ldots, N \\ \dfrac{\partial L}{\partial \alpha_l} = 0 \rightarrow \omega^T\phi(x_l) + b + e_l - y_l = 0, & l = 1, \ldots, N \end{cases} \tag{5}$$

By solving Eq. (5), α_l and b can be obtained. As a result, the LS-SVM model becomes:

$$y = \sum_{l=1}^{N}\alpha_l K(x, x_l) + b \tag{6}$$

With

$$K(x, x_l) = \phi(x)^T\phi(x_l)$$

Although the standard LS-SVM method presented here has been successfully applied, it is less robust to noise and outlier since it assumes that all data have the same importance. Moreover, it does not consider the possible nonlinear distribution of data. Thus, the development of an effective LS-SVM for nonlinear data with noise is needed.

III. ROBUST CLUSTERED LS-SVM METHOD

Here a robust clustered modeling methodology, as indicated in Figure 2, is proposed for modeling data with a nonlinear distribution that is also subject to noise [60]. The data are first divided into many clusters. The sub-models are then trained to capture the main feature of each cluster and are robust to noise by assigning samples with different weights using data distribution information. The continuity and smoothness of the global model are also ensured by coordinating these sub-models. This new method improves the traditional clustered LS-SVM methods, which pay no attention to the robustness of the local models. As a result, a more robust model with better global performance can be developed, even for data with a nonlinear distribution and subject to noise.

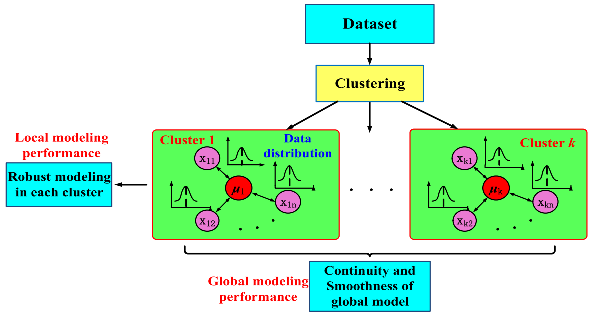


FIGURE 2. Robust clustered modeling methodology.

Based on the robust clustered modeling methodology, a robust clustered LS-SVM method (Figure 3) is proposed for modeling of nonlinear data with noise. This method integrates the advantages of clustering, local robust LS-SVM modeling, and global regularization. It improves the model robustness within each cluster and guarantees continuity and smoothness of the global model between clusters. First, the clustering method divides the sample data. Local robust LS-SVM modeling is then developed to capture the local dynamics of each cluster. Model robustness is also improved by using data distribution information. Finally, a global regularization is constructed to ensure that the integration of all local models is smooth and continuous and accurately represents the data. The details of this proposed robust clustered LS-SVM method are discussed in the following sections.

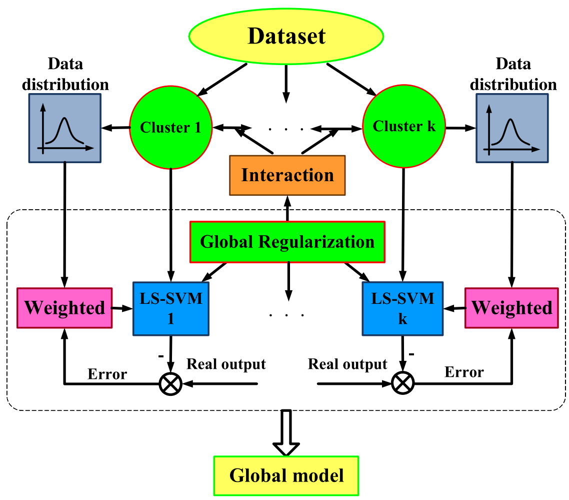


FIGURE 3. Robust clustered LS-SVM method.

A. DATA CLUSTERING

In many applications, many systems are often strongly nonlinear and work in a large operating region. In particular, the systems are often hybrid, composed by many nonlinear dynamic sub-processes or sub-systems that have different physical rules in different operating region. For example, the dynamic response during the forging process varies over time as the pistons of the driving cylinders move with or without the deformation load of the forging. Even when moving with the deformation load, the dynamic behavior of the hydraulic press machine (HPM) depends on the position and velocity of the work plate. In addition, each working region has nonlinear dynamics due to nonlinear deformation load, nonlinear hydraulic driving force, and other factors. In this sense, many processes have different models as working at different positions and velocities. Thus, models in this kind of processes may be differentiated according to positions or velocities. This means that the k-means might be able to separate a dataset mixed by several nonlinear models as taking position or velocity into account.

Given a dataset, the dataset is divided into k $(k < N)$ groups, $S = \{S_1, S_2, \ldots, S_k\}$. Here, the widely used k-means Clustering algorithm [52] is employed. And the cluster number k is decided according to the prior system knowledge or experts' knowledge from the actual industrial processes. It obtains the clusters by minimizing the following objective function:

$$J = \sum_{l=1}^{N} \sum_{i=1}^{k} r_{il} \left\| z_l - \mu_i \right\|^2$$

With

$$z_l = (x_l, y_l) \tag{7}$$

In Eq. (7), when a data point z_l is assigned to the ith category, r_{il} is 1, otherwise r_{il} is 0; μ_i represents the center of the ith category, and $\|z_l - \mu_i\|^2$ represents the distance between the data point z_l and the cluster center μ_i.

The following iterative approach is generally used to solve this optimization problem:

Step 1: In the initial dataset, k data points are randomly selected to represent the initial group centers, namely, $\{\mu_1, \mu_2, \ldots, \mu_k\}$.

Step 2: Remaining data points are assigned to the group with the shortest distance from the group center.

Step 3: After all the data points are assigned, new centers of the k groups are calculated.

Step 4: Step 2 and Step 3 are repeated until the centers do not move.

In this way, a dataset can be divided into k sub-clusters and each sub-cluster includes part of the dataset.

B. LOCAL LS-SVM MODEL

Since the data of many strongly nonlinear systems or processes are nonlinearly distributed in clusters due to limited clusters divided, a nonlinear LS-SVM is used to represent the feature of each sub-cluster. For the ith $(i = 1, 2, \ldots, k)$ sub-cluster, the following LS-SVM model is employed:

$$y_i = \omega_i^T \varphi_i(x) + b_i \tag{8}$$

where x_i and y_i are the input and output of the ith local sub-cluster, ω_i and b_i are the local weight vector and local bias term, respectively, and the nonlinear function φ_i maps the input space to a higher dimensional feature space.

In this way, the features of each cluster can be represented accurately by a LS-SVM model.

C. ROBUST GLOBAL LS-SVM MODEL

A robust global LS-SVM modeling method is then proposed, which considers the robustness of the local LS-SVM model at each cluster along with the global modeling performance for all clusters. Given the training dataset $\{x_{ij}, y_{ij}\}_{i=1,\ldots,k; j=1,\ldots,n}$, where x_{ij} and y_{ij} are the jth input and output in the ith cluster, respectively, the optimization problem becomes:

$$\begin{aligned} \min_{\omega,\omega_i,e_{ij}} \; & \frac{\delta}{2}\sum_{i=1}^{k} ||\omega_i||^2 + \frac{\lambda}{2}\sum_{i=1}^{k} ||\omega_i - \omega||^2 \\ & + \frac{\gamma}{2}\sum_{i=1}^{k}\sum_{j=1}^{n} \theta_{ij} e_{ij}^2 \\ s.t.\; & y_{ij} = \omega_i^T \varphi_i(x_{ij}) + b_i + e_{ij}, \\ & i = 1, \ldots, k; \quad j = 1, \ldots, n \end{aligned} \tag{9}$$

where ω denotes the global weight vector that connects all the local models from all clusters, e_{ij} is the modeling error, θ_{ij} is the error weight, and δ, λ, and γ are the regularization parameters. Here, the error weight θ_{ij} is defined as follows:

$$\theta_{ij} = e^{-\frac{d_{ij}^2}{\sigma^2}} \tag{10}$$

where d_{ij} is the Euclidean distance between the data point and its cluster center, and σ is the width of a Gaussian function. For a data point with a very short Euclidean distance to the cluster center, the error weight θ_{ij} is close to 1. As the point moves further from the cluster center, the error weight θ_{ij} moves toward 0. When the data point is placed on the cluster center, its error weight θ_{ij} is equal to 1.

In Eq. (9), the global weight vector ω establishes a bridge among local models of different clusters, thereby different local models can affect with each other, and the continuity and smoothness between the local models can be achieved by coordinating these sub-models using the global weight vector ω. In this way, the connection between local and global modeling is constructed. Moreover, the robustness is achieved by assigning samples with different weights using data distribution information. As a result, a more robust model can be developed with better global performance, even for data with a nonlinear distribution and subject to noise. Specifically, this new objective function has the following features:

- ❑ $\frac{\delta}{2}\sum_{i=1}^{k} \|\omega_i\|^2$ is the local regularization term, which avoids over-fitting of the local LS-SVM models.
- ❑ $\frac{\lambda}{2}\sum_{i=1}^{k} \|\omega_i - \omega\|^2$ is the global regularization term, where the weight vector of each local LS-SVM model aligns with the global weight vector. This global weight vector is used to coordinate all local LS-SVM models in order to achieve a better global modeling performance. Through this process, the interaction, continuity,

and smoothness between the clusters can be considered together. The global weight vector may improve the global generalization of the model.

- $\frac{\gamma}{2}\sum_{i=1}^{k}\sum_{j=1}^{n}\theta_{ij}e_{ij}^2$ is the global error term that is used to improve the regression accuracy and robustness. In this term, the modeling error of each training point aligns with a corresponding error weight. It is well-known that noise samples found in training data are typically separated from the main data. This feature is often seen in data distributions where noisy samples are farther away from the center of each cluster. Samples that are close to the cluster center should be less contaminated by noise and are considered important. Data points that are far from the cluster center are considered noise. The error weight is thus derived from the data distribution that is constructed using the Euclidean distance between each sample and its cluster center. The further the distance is, the smaller the error weight is and the less importance the sample is assigned, otherwise the error weight is larger and the sample is considered more important. Therefore, the error weight inherently contains noise information, which can improve the robustness of the model.

This new objective function improves modeling robustness by giving smaller error weights to noise data. Then, through the combination of local models, the global model may be more robust due to the local robust models. Moreover, the connection of global weight vector to local models could improve the continuity and smoothness of global model. Thus, the proposed method has a better global modeling performance and is robust to noise.

In this proposed method, the local models are not trained only using data of the local cluster; they are achieved using all the data by solving Eq.(9). The details of the solving process of Eq.(9) are presented below.

Let $v_i = \omega_i - \omega$, then the objective function Eq. (9) may be rewritten as:

$$\begin{aligned} \min_{\omega, v_i, e_{ij}} \ & \frac{\delta}{2}\sum_{i=1}^{k}\|v_i+\omega\|^2 + \frac{\lambda}{2}\sum_{i=1}^{k}\|v_i\|^2 + \frac{\gamma}{2}\sum_{i=1}^{k}\sum_{j=1}^{n}\theta_{ij}e_{ij}^2 \\ s.t.\ & y_{ij} = (v_i+\omega)^T\varphi_i(x_{ij}) + b_i + e_{ij}, \\ & i = 1,\ldots,k;\ j = 1,\ldots,n \end{aligned} \tag{11}$$

Then, define $\tilde{\omega} = \left[\sqrt{\delta}(v_1+\omega)^T, \ldots, \sqrt{\delta}(v_k+\omega)^T, \sqrt{\lambda}\ v_1^T, \ldots, \sqrt{\lambda}v_k^T\right]^T$, Eq. (11) can be further expressed as follows:

$$\begin{aligned} \min_{\tilde{\omega}, e_{ij}} \ & \frac{1}{2}\|\tilde{\omega}\|^2 + \frac{\gamma}{2}\sum_{i=1}^{k}\sum_{j=1}^{n}\theta_{ij}e_{ij}^2 \\ s.t.\ & y_{ij} = \tilde{\omega}^T\tilde{\varphi}(x_{ij}) + b_i + e_{ij}, \quad i = 1,\ldots,k;\ j = 1,\ldots,n \end{aligned} \tag{12}$$

where $\tilde{\varphi}$ represents a new mapping function used to guarantee $\tilde{\omega}^T\tilde{\varphi}(x_{ij}) = \omega_i^T\varphi_i(x_{ij})$. Eq.(12) has the same form as that of the original LS-SVM and, thus, it has the same solving process as that of the original LS-SVM. To solve the optimization problem with constraints, a Lagrangian function is constructed:

$$\begin{aligned} \Gamma(\tilde{\omega}, a_{ij}, b_i, e_{ij}) = & \frac{1}{2}\|\tilde{\omega}\|^2 + \frac{\gamma}{2}\sum_{i=1}^{k}\sum_{j=1}^{n}\theta_{ij}e_{ij}^2 \\ & - \sum_{i=1}^{k}\sum_{j=1}^{n}a_{ij}\left\{\tilde{\omega}^T\tilde{\varphi}(x_{ij}) + b_i + e_{ij} - y_{ij}\right\} \end{aligned} \tag{13}$$

Here, a_{ij} is the Lagrange multipliers.

The conditions for optimization are given by:

$$\begin{cases} \frac{\partial\Gamma}{\partial\tilde{\omega}} = 0 \rightarrow \tilde{\omega} - \sum_{i=1}^{k}\sum_{j=1}^{n}a_{ij}\tilde{\varphi}(x_{ij}) = 0 \\ \frac{\partial\Gamma}{\partial e_{ij}} = 0 \rightarrow \gamma\theta_{ij}e_{ij} - a_{ij} = 0 \\ \frac{\partial\Gamma}{\partial a_{ij}} = 0 \rightarrow \tilde{\omega}^T\tilde{\varphi}(x_{ij}) + b_i + e_{ij} - y_{ij} = 0 \\ \frac{\partial\Gamma}{\partial b_i} = 0 \rightarrow \sum_{j=1}^{n}a_{ij} = 0, \quad i = 1,\ldots,k;\ j = 1,\ldots,n \end{cases} \tag{14}$$

From Eq. (14), we have:

$$\begin{cases} \sum_{j=1}^{n}a_{ij} = 0, \quad i = 1,\ldots,k;\ j = 1,\ldots,n \\ \sum_{p=1}^{k}\sum_{q=1}^{n}a_{pq}\tilde{\varphi}(x_{pq})^T\tilde{\varphi}(x_{ij}) + b_i + \frac{1}{\gamma\theta_{ij}}a_{ij} = y_{ij} \end{cases} \tag{15}$$

Then, a kernel function $\boldsymbol{K}$ is defined as:

$$K(x_{ij}, x_{pq}) = \tilde{\varphi}(x_{ij})^T\tilde{\varphi}(x_{pq}), \quad i, p = 1,\ldots,k;\ j, q = 1,\ldots,n \tag{16}$$

The kernel function $\boldsymbol{K}$ is positive and needs to satisfy the Mercer condition. The typical options include linear kernel, radial basis function (RBF) kernel, polynomial kernel, and multilayer perceptron. Here, the RBF kernel functions will be employed, since it has well ability to approximate nonlinear behavior and is easy to be trained due to fewer parameters required to optimization.

Finally, the Lagrange multipliers a_{ij} and bias term b_i can be easily solved from Eq. (15). The resulting LS-SVM model is as follows:

$$\hat{y} = \sum_{i=1}^{k}\sum_{j=1}^{n}a_{ij}K(x, x_{ij}) + b_i \tag{17}$$

D. COMPLEXITY ANALYSIS

According to [53], the computational complexity of an m-by-n-order matrix is about $m \cdot n$. The computational complexity of multiplying an m_1-by-n-order matrix and an n-by-m_2-order matrix is about $m_1 \cdot m_2 \cdot (n-1)$. Following this, the computational complexity of the generalized inverse of an m-by-n-order matrix would be $4n^3 + 4m^2n + 4mn^2 - 4n^2 - m^2 - mn$, which is equal to $12n^3 - 6n^2$ (if $m = n$). Given a dataset $\{x_i, y_i\}_{i=1}^{n}$, $x_i \in R^m$, the computational complexity of a polynomial kernel function on this dataset is approximately $n^2(m-1) + 2n^2(n-1) + n^2$.

Thus, for a dataset $\{x_i, y_i\}_{i=1}^{n}$, $x_i \in R^m$ and $y_i \in R^d$, the computational complexity of the traditional LS-SVM method may be calculated:

O(Traditional LS-SVM)

$$\begin{aligned} &= n^2(m-1) + 2n^2(n-1) + n^2 + n^2 + 12(n+1)^3 \\ &\quad - 6(n+1)^2 + (n+1)\cdot d\cdot(n+1-1) \\ &\approx 14n^3 + (29+m+d)n^2 + (24+d)n + 6 \end{aligned} \tag{18}$$

In comparison, the computational complexity of the robust clustered LS-SVM method applied to the same dataset divided into k clusters is estimated as follows:

O(Robust Clustered LS-SVM)

$$\begin{aligned} &= k^2\left((\frac{n}{k})^2(m-1) + 2(\frac{n}{k})^2(\frac{n}{k}-1) + (\frac{n}{k})^2\right) + n^2 \\ &\quad + 12(n+k)^3 - 6(n+k)^2 + (n+k)\cdot d\cdot(n+k-1) \\ &\approx (12+\frac{2}{k})n^3 + (36k-7+m+d)n^2 \\ &\quad + (36k^2-12k+2dk-d)n + 12k^3 - 6k^2 + dk^2 - dk \end{aligned} \tag{19}$$

Analyzing Eqs. (18) and (19) shows the computational complexities of the traditional LS-SVM and the robust clustered LS-SVM can be approximated to $O(14n^3)$ and $O((12+\frac{2}{k})n^3)$, respectively. Besides, this robust clustered LS-SVM should include the complexity of the K-Means clustering. The time complexity of the K-Means clustering is about $O(nkt)$, where t is the iteration number. Thus, the total complexity of the robust clustered LS-SVM is about $O((12+\frac{2}{k})n^3 + nkt)$. Since both k and t are much smaller than n, and k is always greater than 1, the computational complexity of the proposed method is smaller than the traditional LS-SVM. Also, the larger the sample number n, the smaller the computational complexity of the proposed method as compared to the traditional LS-SVM.

E. SUMMARY

The proposed method includes the following key steps:

Step 1: Collect the training data;
Step 2: Cluster the samples and derive the error weights;
Step 3: Construct local models for all clusters;
Step 4: Solve the optimization problem (9) and Eq. (15) to obtain the Lagrange multipliers a_{ij} and bias term b_i, upon which the robust LS-SVM model is derived as Eq. (17).

IV. CASE STUDIES

In this section, the modeling performance of the proposed robust clustered LS-SVM is evaluated and compared with those of the traditional LS-SVM [11], the fuzzy LS-SVM [14], the clustered LS-SVM [28], the local linear method [30], and the weighted LS-SVM [33] by using one mathematical case and three real cases. The kernel function parameters and the regularization parameters δ, λ, and γ are determined according to the cross-validation algorithm [57]. The root mean square error (RMSE) is defined as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{l=1}^{N}(y_l - \hat{y}_l)^2} \tag{20}$$

where N denotes the number of data samples, and y_l and $\hat{y}_l$ are the actual output and model output, respectively.

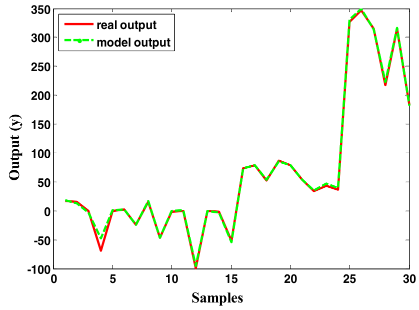


FIGURE 4. Model on testing samples.

A. EXAMPLE 1

Data are generated from the following function:

$$y = ((15x_1 - 1)\cdot x_2)\, exp(x_2/5) + \varepsilon \tag{21}$$

where ε is the Gaussian noise with variance $\mathrm{Var}(\varepsilon) = 0.1$, and x_1 and x_2 are randomly taken within the interval [0 5]. The concrete steps of the proposed method are described as follows. First, 90 samples were generated from the above function. 60 samples were randomly selected as the training samples and the remaining samples were used as test samples. According to the Majority-rule consensus method [54], a method for choosing the size of the subsets, the training samples were divided into three clusters and based on the within-class distance, the error weight θ was determined. Three sub-models were constructed based on these clusters and the global model was built by solving the optimization problem (11) based on these sub-models. From that, the model was achieved as Eq. (17). Finally, the remaining 30 samples were used to verify the ability to generalize the built model. From Figure 4, it is clear that the model output approximates the test data well. Thus, the robust clustered LS-SVM method can be used to quantify this system.

The proposed method was also compared with the other five approaches. The comparison of results is shown in Figure 5, where absolute errors on the 30 test samples are shown, except that of the local linear method because its mean is larger than 70. The RMSEs of the different methods are listed in Table 1. From these results, it can be seen that the proposed method is more general and robust than the other methods investigated.

Moreover, 50 repetitive experiments were conducted, and the RMSE and its statistic information were estimated.

TABLE 1. **Comparison of RMSEs between different methods on the exponential function.**

Method	Proposed method	Fuzzy LS-SVM	Clustered LS-SVM	Traditional LS-SVM	Local linear method	Weighted LS-SVM
RMSE	4.361	5.331	6.580	10.505	72.168	4.656
Mean ± variance	4.459 ± 0.013	5.436 ± 0.017	6.395 ± 0.157	10.182 ± 0.088	71.332 ± 1.434	4.584 ± 0.016
Training time(s)	0.0313	0.0401	0.0211	0.0383	0.0118	0.0323

TABLE 2. **Comparison of RMSEs between different methods under different variances on the exponential function.**

Method \ Var(ε)		0.2	0.3	0.4	0.5	0.6
Proposed method	RMSE	4.431	4.137	4.382	4.720	4.486
	Mean±variance	4.453±0.028	4.481±0.012	4.528±0.033	4.396±0.074	4.497±0.066
Fuzzy LS-SVM	RMSE	5.507	5.424	5.233	5.476	5.347
	Mean±variance	5.420±0.029	5.432±0.016	5.401±0.029	5.344±0.066	5.541±0.067
Clustered LS-SVM	RMSE	5.745	6.704	6.427	5.567	6.598
	Mean±variance	6.084±0.189	6.608±0.186	6.272±0.236	6.260±0.223	6.126±0.142
Traditional LS-SVM	RMSE	9.876	10.512	10.426	10.708	10.402
	Mean±variance	10.382±0.078	10.381±0.112	10.482±0.058	10.460±0.043	10.411±0.113
Local linear method	RMSE	72.883	73.915	74.056	76.484	75.198
	Mean±variance	71.684±1.563	71.713±1.365	73.473±1.654	74.854±1.270	74.953±1.344
Weighted LS-SVM	RMSE	4.589	4.319	4.511	4.716	4.619
	Mean±variance	4.547±0.028	4.483±0.013	4.691±0.026	4.853±0.069	4.706±0.070

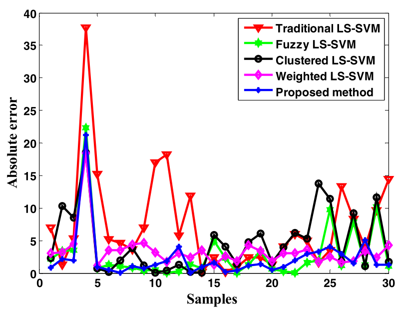


FIGURE 5. **Comparison of absolute errors between different methods.**

These results are shown in Table 1. Also, the comparison of complexity is analyzed by the training time of model, where the analysis of the complexity includes clustering analysis. The complexity of the six approaches was compared and is shown in Table 1, where the mean of the training time was calculated. Table 1 also shows that the proposed method reduces the complexity as compared to the traditional LS-SVM, the weighted LS-SVM and the fuzzy LS-SVM methods. Although the local linear method has the shortest training time, it has an inferior modeling accuracy because it does not consider the relationship between local models. Taking into consideration both modeling accuracy and complexity, the proposed robust clustered LS-SVM method is more effective than the other methods examined.

Finally, to examine the model response to different levels of noise, we selected Var(ε) from the range [0.2, 0.3, 0.4, 0.5, 0.6]. For each Var(ε), 30 test samples were taken, and the RMSEs and their statistic information from different modeling methods were calculated (Table 2). In each case, the proposed method showed smaller RMSE and statistical values than the other five approaches in comparison. Thus, even when the sample is subject to noise, the proposed robust clustered LS-SVM is more general and robust than the other models examined.

B. EXAMPLE 2

This dataset from the UCI Machine Learning Repository [55], [56] was collected from a Combined Cycle Power Plant while the plant was working under a full load. It contains the hourly average ambient variables Temperature (T), Ambient Pressure (AP), Relative Humidity (RH), and Exhaust Vacuum (EV), which are used to predict the net hourly electrical energy output (EP) of the plant.

In this test, 5400 observations were selected randomly from the dataset. Noise randomly taken from [0 10] was mixed into 5400 observations. Of these, 3600 observations were used as training samples to construct the model, and the remaining observations were used as test data to verify the model. In order to improve the quality of Figure 6, only 120 randomly selected modeling results are displayed. Figure 6 shows that the proposed model captures the experimental system well.

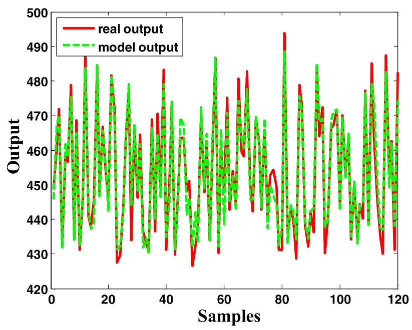


FIGURE 6. Model on testing samples.

Finally, the comparison of the complexities of the six approaches is shown in Table 3. For each model, 50 experiments were conducted and the mean of the training time, RMSEs and their mean and variance were calculated. As shown, the proposed method reduces the complexity compared to the traditional LS-SVM, the weighted LS-SVM and the fuzzy LS-SVM methods. Although the local linear method has the shortest training time, it is not accurate because it does not consider the relationship between local models. A comprehensive assessment of both modeling accuracy and complexity shows that the proposed method is faster and more efficient than the other five methods.

C. EXAMPLE 3

The robust clustered LS-SVM model was examined using a physical experiment regarding the thermal curing of chips in a snap oven [5-6]. The snap oven, as shown in Figure 7, uses four heaters to heat the chips. A thermocouple is used to measure the temperature. As previously described, the curing thermal process is complex.

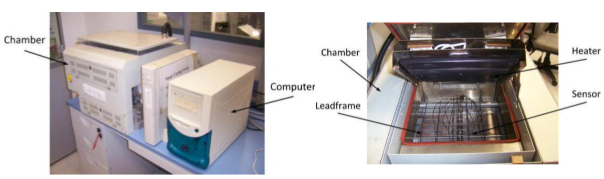


FIGURE 7. Snap curing oven system.

In this experiment, 2800 temperature data points were collected with a sampling interval Δt=10s, and four random input signals were used to drive the heaters. Noise randomly taken from [0 10] was mixed into real temperature data. For example, the input signal of one of the heaters is shown in Figure 8. 1867 data points were used for training and the remaining 933 points data were used for testing and validation. In order to improve the quality of Figure 9, only 100 randomly selected data points are displayed. As shown, the proposed modeling method follows both the training and test samples closely. Thus, the proposed method successfully models the curing thermal process.

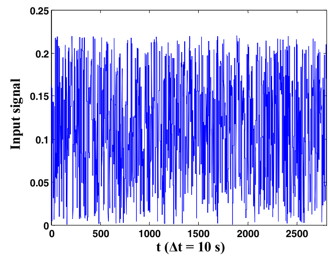


FIGURE 8. Input signals of heater h2.

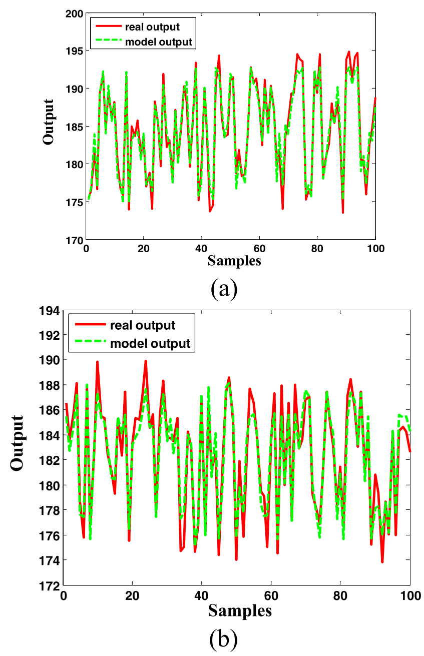


FIGURE 9. (a) Model verification on training samples; (b) Model verification on testing samples.

Lastly, the proposed method was compared to the other five approaches to model the snap oven. The comparison of results is shown in Table 4, where the test error of the proposed method is smaller than those of the other five methods in comparison. The complexity of each of the six approaches was calculated as the mean training time over 50 repetitive experiments. As shown in Table 4, the proposed method reduces the complexity as compared to the traditional LS-SVM, the weighted LS-SVM and the fuzzy LS-SVM methods. Again, the local linear method has the shortest

TABLE 3. Comparison of RMSEs between different methods on the combined cycle power plant.

Method	Proposed method	Fuzzy LS-SVM	Clustered LS-SVM	Traditional LS-SVM	Local linear method	Weighted LS-SVM
RMSE	5.216	5.318	5.464	5.442	5.575	5.223
Mean ± variance	5.116 ± 0.006	5.512 ± 0.005	5.438 ± 0.029	5.331 ± 0.013	5.471 ± 0.022	5.184 ± 0.008
Training time(s)	127.542	237.016	61.351	237.436	30.365	236.755

TABLE 4. Comparison of RMSEs between different methods on the curing thermal process.

Method	Proposed method	Fuzzy LS-SVM	Clustered LS-SVM	Traditional LS-SVM	Local linear method	Weighted LS-SVM
RMSE	3.013	3.557	3.393	7.441	3.706	3.168
Mean ± variance	2.980 ± 0.197	3.500 ± 0.217	3.328 ± 0.175	7.576 ± 1.464	3.681 ± 0.186	3.094 ± 0.266
Training time(s)	25.938	47.104	10.827	45.400	5.561	44.629

training time; however, it is less accurate than the robust clustered LS-SVM method. The proposed method improves the accuracy of the other five methods without increasing complexity.

D. EXAMPLE 4

The practical hydraulic actuator, as shown in Figure 10, was used to validate the proposed method. The hydraulic actuator consists of two cylinders: a driving cylinder and a load cylinder. The load cylinder simulates nonlinear deformation force from an arbitrary work piece by adjusting the pressure within the cylinder. The flow in the cylinders and the velocity response of the piston are regarded as input and output, respectively. Pressure data were collected using the pressure sensors installed at the inlet of the cylinders. The displacement sensor was installed at the transverse column.

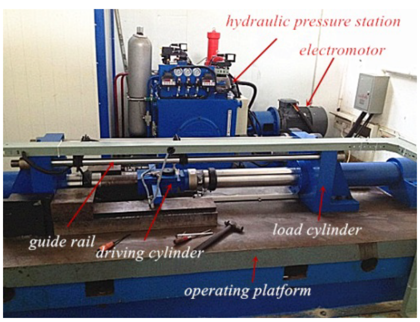


FIGURE 10. Practical hydraulic actuator.

In this experiment, noise randomly taken from [0 0.5] was mixed into the real output data, and the first 809 data points were used to train the model, while the remaining 404 data points were used to examine the model generalization. The flows of the two cylinders are described in Figure 11. Training and testing results are shown in Figure 12.

It is clear that the proposed approach successfully approximates the output of the practical system. The robust

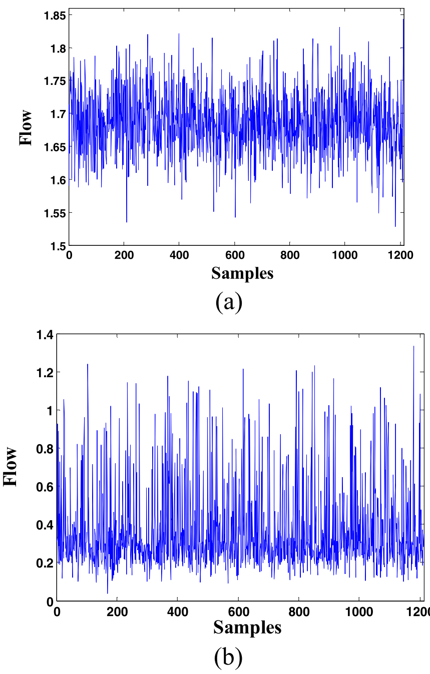


FIGURE 11. (a) Flow of driving cylinder; (b) Flow of load cylinder.

clustered LS-SVM method was compared with the other five approaches using the same dataset. The comparison of the results shown in Table 5 indicates that the error of the proposed method is much smaller than those of the other five methods. Thus, the robust clustered LS-SVM method is more general and robust.

Finally, the complexity of each of the six approaches was calculated as the mean training time over 50 repetitive experiments. As shown in Table 5, the proposed method reduces the complexity as compared to the traditional LS-SVM, the weighted LS-SVM and the fuzzy LS-SVM methods. Although the local linear method has the shortest training time, it has a lower accuracy than the robust clustered

TABLE 5. Comparison of RMSEs between different methods on the hydraulic driving process.

Method	Proposed method	Fuzzy LS-SVM	Clustered LS-SVM	Traditional LS-SVM	Local linear method	Weighted LS-SVM
RMSE	0.083	0.089	0.096	0.101	0.090	0.089
Mean ± variance	0.081± 2.944×10^{-5}	0.090± 2.918×10^{-5}	0.090± 3.064×10^{-5}	0.116± 3.168×10^{-5}	0.094± 3.058×10^{-5}	0.087± 2.918×10^{-5}
Training time(s)	2.776	4.364	1.218	4.339	0.741	4.358

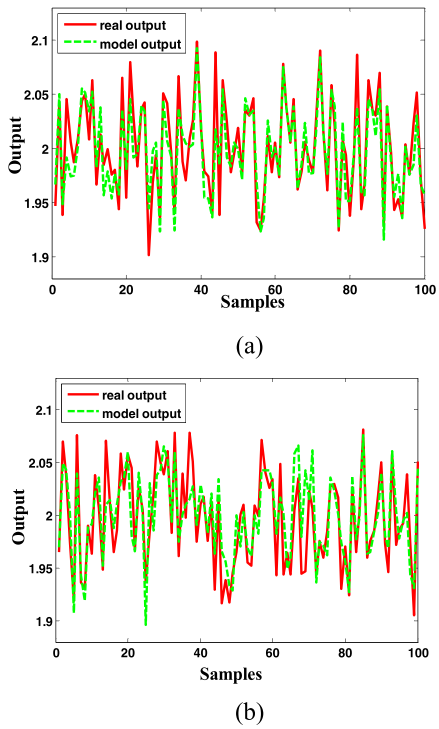


FIGURE 12. (a) Model verification on training samples; (b) Model verification on testing samples.

LS-SVM method. Thus, the proposed method improves the accuracy of the other five methods without increasing complexity.

V. CONCLUSION

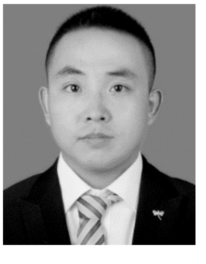
A novel robust clustered LS-SVM method is proposed for modeling nonlinearly distributed data with noise. Unlike the traditional clustered LS-SVM method which is sensitive to noise, the proposed method is robust and considers both local and global generalizations. The proposed solution generates a robust model of each cluster while maintaining a smooth and continuous global model, thus it generates an accurate, robust, and global generalization. Through the use of both simulated and real datasets, the proposed LS-SVM method demonstrated superior performance in comparison to the original LS-SVM, the fuzzy LS-SVM, the clustered LS-SVM, the local linear modeling, and the weighted LS-SVM methods.

REFERENCES

[1] G. F. Qin and X. J. Lu, "Integration of weighted LS-SVM and manifold learning for fuzzy modeling," *Neurocomputing*, vol. 282, pp. 184–191, Mar. 2018.
[2] D. Kaur, G. S. Aujla, N. Kumar, A. Y. Zomaya, C. Perera, and R. Ranjan, "Tensor-based big data management scheme for dimensionality reduction problem in smart grid systems: SDN Perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 10, pp. 1985–1998, Oct. 2018, doi: 10.1109/TKDE.2018.2809747.
[3] X. Lu, C. Zhou, M. Huang, and W. Lv, "Regularized online sequential extreme learning machine with adaptive regulation factor for time-varying nonlinear system," *Neurocomputing*, vol. 174, pp. 617–626, Jan. 2016.
[4] Y. Gao, S. Song, X. Zhu, J. Wang, X. Lian, and L. Zou, "Matching heterogeneous event data," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 11, pp. 2157–2170, Nov. 2018, doi: 10.1109/TKDE.2018.2815695.
[5] H.-X. Li and C. Qi, *Spatio-Temporal Modeling of Nonlinear Distributed Parameter Systems: A Time/Space Separation Based Approach*. New York, NY, USA: Springer, 2011.
[6] X. Lu, H.-X. Li, and M.-H. Huang, "Data-driven robust design for a curing oven," *IEEE Trans. Compon. Packag. Manuf. Technol.*, vol. 4, no. 8, pp. 1366–1373, Aug. 2014.
[7] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[8] R. Khemchandani, K. Goyal, and S. Chandra, "TWSVR: Regression via twin support vector machine," *Neural Netw.*, vol. 74, pp. 14–21, Feb. 2016.
[9] B. Schölkopf and A. Smola, "Learning with kernels: Support vector machines, regularization, optimization, and beyond," *J. Amer. Stat. Assoc.*, vol. 98, no. 1, p. 489, Feb. 2003, doi: 10.1198/jasa.2003.s269.
[10] N. Deng, Y. Tian, and C. Zhang, *Support Vector Machines: Optimization Based Theory, Algorithms, and Extensions*. Boca Raton, FL, USA: CRC Press, 2012.
[11] J. A. K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. Singapore: World Scientific, 2002.
[12] Z. Qi, Y. Tian, and Y. Shi, "Twin support vector machine with Universum data," *Neural Netw.*, vol. 36, pp. 112–119, Dec. 2012.
[13] L. Zhang and W.-D. Zhou, "Fisher-regularized support vector machine," *Inf. Sci.*, vols. 343–344, pp. 79–93, May 2016.
[14] C.-F. Lin and S.-D. Wang, "Fuzzy support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 464–471, Mar. 2002.
[15] T.-T. Chen and S.-J. Lee, "A weighted LS-SVM based learning system for time series forecasting," *Inf. Sci.*, vol. 299, pp. 99–116, Apr. 2015.
[16] C. L. Liu, W. H. Hsaio, C. H. Lee, and H. C. Chi, "An HMM-based algorithm for content ranking and coherence-feature extraction," *IEEE Trans. Syst., Man, Cybernetics, Syst.*, vol. 43, no. 2, pp. 440–450, Mar. 2013.
[17] Y.-P. Zha, K.-K. Wang, and F. Li, "A pruning method of refining recursive reduced least squares support vector regression," *Inf. Sci.*, vol. 296, pp. 160–174, Mar. 2015.
[18] J. A. K. Suykens, "Nonlinear modelling and support vector machines," in *Proc. IEEE 18th Instrum. Meas. Technol. Conf.*, vol. 1, May 2001, pp. 287–294.
[19] D. Gómez, E. Zarrazola, J. Yáñez, and J. Montero, "A Divide-and-Link algorithm for hierarchical clustering in networks," *Inf. Sci.*, vol. 316, pp. 308–328, 2015.
[20] T. E. Mallouk, "Water electrolysis: Divide and conquer," *Nature Chem.*, vol. 5, pp. 362–363, Apr. 2013.
[21] C. Zhong, M. Malinen, D. Miao, and P. Fränti, "A fast minimum spanning tree algorithm based on *K*-means," *Inf. Sci.*, vol. 295, pp. 1–17, Feb. 2015.

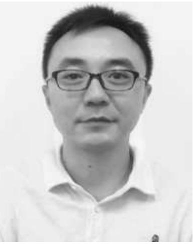
[22] L. Jian, Z. Xia, X. Liang, and C. Gao, "Design of a multiple kernel learning algorithm for LS-SVM by convex programming," *Neural Netw.*, vol. 24, no. 5, pp. 476–483, 2011.
[23] S. Qiu and T. Lane, "A framework for multiple kernel support vector regression and its applications to siRNA efficacy prediction," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 6, no. 2, pp. 190–199, Apr./Jun. 2009.
[24] L. V. Utkin, A. I. Chekh, and Y. A. Zhuk, "Binary classification SVM-based algorithms with interval-valued training data using triangular and Epanechnikov kernels," *Neural Netw.*, vol. 80, pp. 53–66, Aug. 2016.
[25] L. Xie, H. Zheng, and L. Zhang, "Electricity price forecasting by clustering-LSSVM," in *Proc. Int. Power Eng. Conf. (IPEC)*, Dec. 2007, pp. 697–702.
[26] R. Langone and J. A. K. Suykens, "Efficient multiple scale kernel classifiers," in *Proc. Big Data*, Dec. 2016, pp. 128–133.
[27] Q. Fan, D. Gao, and Z. Wang, "Multiple empirical kernel learning with locality preserving constraint," *Knowl.-Based Syst.*, vol. 105, pp. 107–118, Aug. 2016.
[28] T. N. Do, *Non-Linear Classification of Massive Datasets With a Parallel Algorithm of Local Support Vector Machines*. Cham, Switzerland: Springer, 2015.
[29] Y. Tian, X. Ju, and Y. Shi, "A divide-and-combine method for large scale nonparallel support vector machines," *Neural Netw.*, vol. 75, pp. 12–21, Mar. 2016.
[30] Q. Gu and J. Han, "Clustered support vector machines," in *Proc. AISTATS*, 2013, pp. 307–315.
[31] X. Lu, B. Fan, and M. Huang, "A novel LS-SVM modeling method for a hydraulic press forging process with multiple localized solutions," *IEEE Trans. Ind. Inform.*, vol. 11, no. 3, pp. 663–670, Jun. 2015.
[32] C. C. Chuang and Z.-J. Lee, "Hybrid robust support vector machines for regression with outliers," *Appl. Soft Comput.*, vol. 11, no. 1, pp. 64–72, Jan. 2011.
[33] K. De Brabanter, J. De Brabanter, J. A. K. Suykens, and B. De Moor, "Approximate confidence and prediction intervals for least squares support vector regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 22, no. 1, pp. 110–120, Jan. 2011.
[34] B. Fan, X. Lu, and H.-X. Li, "Probabilistic inference-based least squares support vector machine for modeling under noisy environment," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 46, no. 12, pp. 1703–1710, Dec. 2016.
[35] S. F. Tian and H. K. Huang, "A simplification algorithm to support vector machines for regression," *J. Softw.*, vol. 13, no. 6, pp. 1169–1172, 2002.
[36] H.-X. Li, J.-L. Yang, G. Zhang, and B. Fan, "Probabilistic support vector machines for classification of noise affected data," *Inf. Sci.*, vol. 221, pp. 60–71, Feb. 2013.
[37] C.-F. Lin and S.-D. Wang, "Training algorithms for fuzzy support vector machines with noisy data," *Pattern Recognit. Lett.*, vol. 25, no. 14, pp. 1647–1656, 2004.
[38] Z. Liu, S. Xu, C. L. P. Chen, Y. Zhang, X. Chen, and Y. Wang, "A three-domain fuzzy support vector regression for image denoising and experimental studies," *IEEE Trans. Cybern.*, vol. 44, no. 4, pp. 516–525, Apr. 2014.
[39] T. Reitmaier and B. Sick, "The responsibility weighted Mahalanobis kernel for semi-supervised training of support vector machines for classification," *Inf. Sci.*, vol. 323, pp. 179–198, Dec. 2015.
[40] Y. Ren, X. Liu, and J. Cao, "A parsimony fuzzy rule-based classifier using axiomatic fuzzy set theory and support vector machines," *Inf. Sci.*, vol. 181, no. 23, pp. 5180–5193, 2011.
[41] R. Saltos and R. Weber, "A Rough–Fuzzy approach for support vector clustering," *Inf. Sci.*, vol. 339, pp. 353–368, Apr. 2016.
[42] W. Wen, Z. Hao, and X. Yang, "Robust least squares support vector machine based on recursive outlier elimination," *Soft Comput.*, vol. 14, no. 11, pp. 1241–1251, Sep. 2010.
[43] X. Yang, G. Zhang, J. Lu, and J. Ma, "A kernel fuzzy c-means clustering-based fuzzy support vector machine algorithm for classification problems with outliers or noises," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 1, pp. 105–115, Feb. 2011.
[44] Y. Zhang, S. Xu, K. Chen, Z. Liu, and C. L. P. Chen, "Fuzzy density weight-based support vector regression for image denoising," *Inf. Sci.*, vol. 339, pp. 175–188, Apr. 2016.
[45] C. Zhou, X. Lu, and M. Huang, "Dempster–Shafer theory-based robust least squares support vector machine for stochastic modelling," *Neurocomputing*, vol. 182, pp. 145–153, Mar. 2016.
[46] X. Yang, L. Tan, and L. He, "A robust least squares support vector machine for regression and classification with noise," *Neurocomputing*, vol. 140, pp. 41–52, Sep. 2014.
[47] B. Fan, X.-J. Lu, and M.-H. Huang, "A novel LS-SVM control for unknown nonlinear systems with application to complex forging process," *J. Central South Univ.*, vol. 24, no. 11, pp. 2524–2531, 2017.
[48] L. You, L. Jizhen, and Q. Yaxin, "A new robust least squares support vector machine for regression with outliers," *Procedia Eng.*, vol. 15, pp. 1355–1360, Dec. 2011.
[49] R. Khemchandani and S. Sharma, "Robust least squares twin support vector machine for human activity recognition," *Appl. Soft Comput.*, vol. 47, pp. 33–46, Oct. 2016.
[50] C. Chen, C. Yan, and Y. Li, "A robust weighted least squares support vector regression based on least trimmed squares," *Neurocomputing*, vol. 168, pp. 941–946, Nov. 2015.
[51] C. Huang, F.-L. Chung, and S. Wang, "Multi-view L2-SVM and its multi-view core vector machine," *Neural Netw.*, vol. 75, pp. 110–125, Mar. 2015.
[52] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm. Journal of the royal statistical society," *Ser. C (Appl. Statist.)*, vol. 28, no. 1, pp. 100–108, 1979.
[53] R. W. Farebrother, "Linear least squares computations," *Technometrics*, vol. 33, no. 3, pp. 368–369, 1991.
[54] K. L. Nylund, T. Asparouhov, and B. O. Muthén, "Deciding on the number of classes in latent class analysis and growth mixture modeling: A Monte Carlo simulation study," *Structural Equation Model., Multidisciplinary J.*, vol. 14, no. 4, pp. 535–569, 2007.
[55] A. Asuncion and D. Newman, "UCI machine learning repository," Irvine Univ. California, Irvine, CA, USA, 2007. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant
[56] C. Blake and C. J. Merz, "UCI machine learning repository," Univ. California, Oakland, CA, USA, 1998. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Combined+Cycle+Power+Plant
[57] S. An, W. Liu, and S. Venkatesh, "Fast cross-validation algorithms for least squares support vector machine and kernel ridge regression," *Pattern Recognit.*, vol. 40, no. 8, pp. 2154–2162, 2007.
[58] A. Bryant and K. Cios, "RNN-DBSCAN: A density-based clustering algorithm using reverse nearest neighbor density estimates," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 6, pp. 1109–1121, Jun. 2018, doi: 10.1109/TKDE.2017.2787640.
[59] S. Lin, B. Azarnoush, and G. C. Runger, "CRAFTER: A tree-ensemble clustering algorithm for static datasets with mixed attributes and high dimensionality," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 9, pp. 1686–1696, Sep. 2018, doi: 10.1109/TKDE.2018.2807444.
[60] X. Miao, Y. Gao, L. Zhou, W. Wang, and Q. Li, "Optimizing quality for probabilistic skyline computation and probabilistic similarity search," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 9, pp. 1741–1755, Sep. 2018, doi: 10.1109/TKDE.2018.2805824.

XIN-JIANG LU (M'12) received the B.E. and M.E. degrees from the School of Mechanical and Electrical Engineering, Central South University, China, and the Ph.D. degree from the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong. He is currently a Professor with the School of Mechanical and Electrical Engineering, Central South University. His research interests include machine learning, process modeling and control, integration of design and control. He was awarded the Excellent Thesis Award for Master's Degree of Hunan Province in 2007, the Hiwin Doctoral Dissertation Award in 2011, and the New Century Excellent Talents award by the Chinese Ministry of Education in 2013. He served for editorial board membership of three international journals.



TE-TE HU received the M.E. degree from the School of Mechanical and Electrical Engineering, China University of Geosciences, China. He is currently pursuing the Ph.D. degree with the Department of Mechanical and Electrical Engineering, Central South University, China. His current research interests include machine learning, system modeling and control.

YI ZHANG received the Ph.D. degree from the Department of Mechanical and Electrical Engineering, Central South University, China. He is currently a Lecturer with the School of Mechanical and Electrical Engineering, Changsha University, China. His current research interests include machine design, system modeling and control.



BIN FAN received the master's degree in engineering from the Jiangsu University of Science and Technology in 2005. He is currently pursuing the Ph.D. degree with the School of Mechanical and Electrical Engineering, Central South University, Changsha, China. He is the Director of the Department of Mechanical Design, Manufacturing and Automation, Hunan City University. He is mainly engaged in teaching and scientific research on the control and automation of electro-hydraulic integration system.