

Pet. Sci. (2015) 12:177–188  
 DOI 10.1007/s12182-014-0010-9

## ORIGINAL PAPER

# Improvement of the prediction performance of a soft sensor model based on support vector regression for production of ultra-low sulfur diesel

Saeid Shokri · Mohammad Taghi Sadeghi ·  
 Mahdi Ahmadi Marvast · Shankar Narasimhan

Received: 22 April 2014 / Published online: 13 January 2015  
 © The Author(s) 2015. This article is published with open access at Springerlink.com

**Abstract** A novel data-driven, soft sensor based on support vector regression (SVR) integrated with a data compression technique was developed to predict the product quality for the hydrodesulfurization (HDS) process. A wide range of experimental data was taken from a HDS setup to train and test the SVR model. Hyper-parameter tuning is one of the main challenges to improve predictive accuracy of the SVR model. Therefore, a hybrid approach using a combination of genetic algorithm (GA) and sequential quadratic programming (SQP) methods (GA–SQP) was developed. Performance of different optimization algorithms including GA–SQP, GA, pattern search (PS), and grid search (GS) indicated that the best average absolute relative error (AARE), squared correlation coefficient ( $R^2$ ), and computation time (CT) (AARE = 0.0745,  $R^2 = 0.997$  and CT = 56 s) was accomplished by the hybrid algorithm. Moreover, to reduce the CT and improve the accuracy of the SVR model, the vector quantization (VQ) technique was used. The results also showed that the VQ technique can decrease the training time and improve prediction performance of the SVR model. The proposed method can provide a robust, soft sensor in a wide range of sulfur contents with good accuracy.

**Keywords** Soft sensor · Support vector regression · Hybrid optimization method · Vector quantization · Petroleum refinery · Hydrodesulfurization process · Gas oil

## List of symbols

$w$	Weight vector
$b$	Bias term
AARE	Average absolute relative error
$R^2$	Squared correlation coefficient
$RBF$	Gaussian radial basis kernel function
$N$	Sample size
$g (1/2\sigma^2)$	Hyper-parameter
$C$	Regularization parameter (hyper-parameter)
$\hat{X}$	Code vector
$F_i$	Validation data
$T_i$	Training data
$k$	Subsets (folds)
$Exp_i$	Actual values
$Pre_i$	Predicted values
$S_{exp}$	Experimental value of sulfur content
$S_{pre}$	Predicted value of sulfur content
$Q_{H_2}$	Hydrogen flow rate
$Q_{gasoil}$	Gas–oil flow rate

## Greek symbol

$\sigma$	Width of kernel of radial basis function
$\varepsilon$	Precision parameter (hyper-parameter)
$\xi_i, \xi_i^*$	Slack variables
$\Phi(x)$	High-dimensional feature space

S. Shokri · M. T. Sadeghi (✉)  
 Department of Chemical Engineering, Iran University of Science and Technology (IUST), Tehran, Iran  
 e-mail: sadeghi@iust.ac.ir

M. A. Marvast  
 Process and Equipment Technology Development Division,  
 Research Institute of Petroleum Industry (RIPI), Tehran, Iran

S. Narasimhan  
 Department of Chemical Engineering, IIT Madras, Chennai,  
 India

Edited by Xiu-Qin Zhu

## 1 Introduction

Sulfur compounds are one of the most important impurities in crude oil and various petroleum fractions. Reduction of

sulfur content of end product to the new lower limits is one of the recent challenges in the petroleum refineries. Online determination of sulfur concentration in the end product is difficult or impossible due to the limitations in process technology and measurement techniques. This index as the key indicator of process performance is normally determined by offline sample analysis in laboratories or online hardware analyzers that are mostly expensive with high maintenance costs.

Soft sensors can be a supplement to hardware process analyzers as their measurements may often be unavailable due to instrument failure, maintenance calibration necessity, insufficient accuracy, and long dead time (Kartik and Narasimhan 2011). Moreover, soft sensors can be applied for product quality estimation for industrial processes as an alternative to laboratory testing (Bolf et al. 2010). The core of a soft sensor is the construction of a soft sensing model (Yan et al. 2004). Different classes of soft sensors are (Kadlec 2009): (1) Model-driven or white-box model, (2) Data-driven or black-box model, and (3) Hybrid model or gray-box model. The Model-driven or first principle models obtained from the fundamental process knowledge require a lot of expert process knowledge, effort, and time to develop. Data-driven models are based on the data taken from the processing plants, and thus describe the real process conditions (Kadlec et al. 2009, 2011). These data-driven models can be developed more quickly with less expense. The hybrid model is a combination of both methods.

Artificial neural networks (ANNs) have been widely used as a useful tool for nonlinear soft sensing models. However, they give no guarantees of high convergence speed or of avoiding local minima, while there are no general methods to choose the number of hidden units in the networks. Moreover, they need a large number of controlling parameters, have difficulty in obtaining stable solutions with the danger of overfitting, and thus lack generalization capability (Liu et al. 2010).

In recent years, the support vector machine (SVM) technique, based on machine learning formalism and developed by Vapnik (1995), has been gaining popularity over ANN due to its many attractive features and promising empirical performance (Pan et al. 2010).

King et al. (2000) have compared SVM with ANN and concluded that SVM can provide more reliable and better performance under the same training conditions. Li and Yuan (2006) have applied SVMs to the prediction of key state variables in bioprocesses and indicated that SVM is better than ANN.

SVM can be used for classification, regression and other tasks. Applying the SVM to solve regression problems is called the support vector regression (SVR) method (Basak et al. 2007). The SVR tries to find an optimal hyper-plane as a decision function in high-dimensional space. SVR is

different from conventional regression techniques, since it uses structural risk minimization (SRM), instead of empirical risk minimization (ERM) induction principles (Boser et al. 1992; Cristianini and Taylor 2000).

Hyper-parameter tuning is one of the main challenges in improving the predictive accuracy of an SVR model. Moreover, the generalization capability of SVR is highly dependent upon its learning parameters. The grid search method (GSM) is the most common method to determine appropriate values of hyper-parameters. Most researchers have followed a standard procedure using the GSM (Lu et al. 2009). This method's the computation time (CT) is too high, and it is unable to converge to the global optimum, and it is dependent on the parameters of boundary selection (Min and Lee 2005). There have been some research and development efforts into tuning of SVR hyper-parameters. Duan et al. (2003) have found a reasonably good hyper-parameter set for SVM using the Xi-Alpha bound. Some researchers have developed heuristic algorithms for the parameter optimization of SVR. Wu et al. (2009) have developed a kernel parameter-optimization technique using a hybrid model of GA and SVR. Huang (2012) has employed the hybrid GA–SVR methodology to solve an important stock selection for an investment problem. Chen and Wang (2007) have optimized the SVR parameters using metaheuristic algorithms.

Therefore, potentials of the hybrid strategies for optimization of these parameters need to be further investigated. This study proposes a novel hybrid metaheuristic approach for the SVR models to increase their performance both in accuracy and CT by hybridization of GA and SQP.

Moreover, despite having large datasets in process industries, the important issue is the need for high-speed and extensive memory capacities to process the data. The data compression phenomena provided by the VQ technique can be employed to overcome the problem (Somasundaram and Vimala 2010). Using VQ, the training time for choosing optimal parameters is greatly reduced. The most impactful gain here is the robustness of such systems.

The objectives of the present study are (1) Designing a robust and reliable data-driven soft sensor using an SVR model for prediction of sulfur content of treated gasoil. (2) Applying the VQ technique for data compression in the SVR model. This technique can simplify and compress the training set and speed up the computing time and also simultaneously improve the accuracy of the SVR model. (3) Optimizing the hyper-parameter of SVR model. An integrated hybrid GA–SQP algorithm was employed for optimizing the SVR hyper parameters using a fivefold cross-validation technique. To validate the prediction accuracy of the proposed hybrid model, the prediction performance of the proposed hybrid model was compared to those of GS–SVR, PS–SVR and GA–SVR.

## 2 Methodology

### 2.1 Support vector regression (SVR)

The basic concept of SVR is to map nonlinearly the original data  $x$  into a higher-dimensional feature space and solve a linear regression problem in this feature space (Gunn 1998). A number of loss functions such as the Laplacian, Huber's, Gaussian, and  $\varepsilon$ -insensitive can be used in the SVR formulation. Among these, the robust  $\varepsilon$ -insensitive loss function ( $L_\varepsilon$ ) is more common (Vapnik et al. 1996; Si et al. 2009):

$$L_\varepsilon(f(x) - y) = \begin{cases} |f(x) - y| - \varepsilon & \text{for } |f(x) - y| \geq \varepsilon \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where  $\varepsilon$  is a precision parameter representing the radius of the tube located around the regression function,  $f(x)$  (see Fig. 1). The goal of using the  $\varepsilon$ -insensitive loss function is to find a function that can fit current training data with a deviation less than or equal to  $\varepsilon$ . The optimization problem can be reformulated as

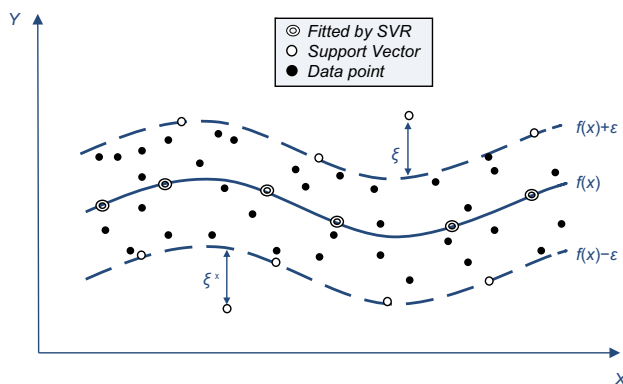
$$\text{Min } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i^- + \xi_i^+) \quad (2)$$

subject to the following constraints:

$$y = \begin{cases} y_i - (\langle w, x_i \rangle + b) \leq \varepsilon + \xi_i \\ (\langle w, x_i \rangle + b) - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

The positive slack variables  $\xi_i$  and  $\xi_i^*$  represent the distance from actual values and the corresponding boundary values of the  $\varepsilon$ -tube, respectively. The constant  $C > 0$  is a parameter determining the trade-off between the empirical risk and the model flatness.

The basic idea in SVR is to map the dataset  $x_i$  into a high-dimensional feature space via nonlinear mapping. Kernel functions perform nonlinear mapping between the input space and a feature space. Different kernel trick functions were used (Table 1) (Yeh et al. 2011).



**Fig. 1** A schematic diagram of SVR using an  $\varepsilon$ -sensitive loss function

**Table 1** Different kernel functions

No.	Kernel type	Equation
1	Linear kernel	$K(x, y) = x_i^T \times y_i$
2	Polynomial kernel	$K(x, y) = (x_i^T y_i + t)^d$
3	Sigmoid kernel	$K(x, y) = \tanh(x_i^T \times y_i)$
4	Gaussian (radial basis function, RBF) kernel	$K(x, y) = \exp\left(-\frac{\ x_i - y_j\ ^2}{2\sigma^2}\right)$

### 2.2 Vector quantization (VQ)

VQ is a data compression method based on the principle of block coding. VQ is applied to reduce a large dataset replacing examples by prototypes. Using VQ, the training time for choosing optimal parameters is greatly reduced. The most impactful gains here are the robustness of such systems.

The prediction speed is very important in soft sensor design. Therefore, in order to speed up the training time and reliability prediction of SVR model, the VQ technique is applied for data compression. The main goal of this method is to simplify the training set and increase the prediction accuracy. In the VQ technique, the data are quantized in the form of contiguous blocks called vectors rather than individual samples. VQ maps a  $K$ -dimensional vector  $x$  in the vector space  $R^k$  to another  $K$ -dimensional vector  $y$  that belongs to a finite set  $C$  (code book) of output vectors (code words).

In this method,  $K$ -dimensional input vectors are derived from input data  $\{X\} = \{x_i; i = 1, 2, \dots, N\}$ . Data vectors are quantized into a finite set of code words  $\{Y\} = \{y_j; j = 1, 2, \dots, K\}$ . Each vector  $y_j$  is called a code vector or a code word, and the set of all the code words is called a code book where the overall distortion of the system should be minimized. The purpose of the generated code book is to provide a set of vectors which generate minimal distortion between the original vector and the quantized vector.

The generation of the code book is the most important process that determines the performance of VQ. The aim of code book generation is to find code vectors (code book) for a given set of training vectors by minimizing the average pairwise distance between the training vectors and their corresponding code words (Hornig 2012).

Each vector is compared with a collection of representative code vectors,  $\hat{X}_i (i = 1, 2, \dots, N_c)$ , taken from a previously generated code book. The best-matching code vector is chosen using a minimum distortion rule (Gersho and Gray 1992). To minimize the distortion, the following formula is used to determine the distance between two code words:

$$d(X, \hat{X}) = \frac{1}{N} \sum_{i=1}^n (x_i - \hat{x}_i)^2 \quad (3)$$

where  $d(X, \hat{X})$  denotes the distortion incurred in replacing the original vector  $X$  with the code vector  $\hat{X}$ .

Therefore, VQ comprises three stages: (1) Code book generation, (2) Vector encoding, and (3) Vector decoding. It works by encoding values from a multidimensional vector space into a finite set of values from a discrete subspace of lower dimension.

### 2.3 $K$ -fold cross-validation (CV)

The quality of the soft sensor model identified from data can be assessed using CV. In  $k$ -fold cross-validation, the original sample is randomly partitioned into  $k$  subsets (folds) of approximately equal size. Of the  $k$  subsets, a single subset is retained as the validation data for testing the model, and the remaining  $k-1$  subsets are used as training data (An et al. 2007). Therefore, the training dataset  $X$  is randomly divided into  $k$  mutually exclusive folds of approximately equal size parts  $Z_i$  ( $i = 1, 2, \dots, k$ ). By training the model  $k$  times and leaving out one of the  $k$  subsets each time,  $k$  pairs are obtained as follows:

$$\begin{aligned} F_1 &= Z_1, T_1 = Z_2 \cup Z_3 \cup \dots \cup Z_k \\ F_2 &= Z_2, T_1 = Z_1 \cup Z_3 \cup \dots \cup Z_k \\ &\vdots \\ F_k &= Z_k, T_k = Z_1 \cup Z_2 \cup \dots \cup Z_{k-1} \end{aligned} \quad (4)$$

where  $F_i$  represents the validation dataset, and  $T_i$  represents the training dataset. The  $k$  results from the folds can then be averaged to produce a single estimation. The advantage of this method over repeated random sub sampling is that all the observations are used for both training and validation, while each observation is used for validation exactly once. As  $k$  increases, the percentage of training samples increases, and a more robust estimator can be obtained; however, the validation sets become smaller.

## 3 Experimental set-up

As one of the vital catalytic units in oil refineries, the HDS process is very effective in sulfur removal from petroleum fractions where the molecules containing sulfur lose their sulfur atoms via hydrogenation reactions (Zahedi et al. 2011). HDS of gas–oil fractions is commonly accomplished in a trickle-bed reactor where there are three phases, namely gas (hydrogen), liquid (gasoil), and solid (catalyst particles) (Froment 2004; Korsten and Hoffmann 1996).

A pilot plant facility for HDS processing of petroleum streams has been set up at the Research Institute of

Petroleum Industry of Iran (RIPI). A schematic diagram of the experimental set-up used in this work is shown in Fig. 2. The major parameters of the set-up are shown in Table 2. Gas–oil containing 7,200 ppm (by weight) of sulfur is fed into the reactor. Feedstock selected for HDS set-up is gasoil with the characteristics listed in Table 3.

Gasoil is first pumped into the unit, preheated and mixed with hydrogen. The mixture is then passed through the trickle-bed reactor. Output of the reactor is directed to the condenser in which the treated gas–oil and  $H_2S$  are separated. Co-Mo HDS catalyst on alumina support (DC-130) procured from CRITERION Company is used in the experiments.

The content of sulfur in the product depends on (1) reactor temperature, (2) reactor pressure, and (3)  $H_2$ /Oil ratio. Therefore, in order to train and test the SVR model, a set of experiments were carried out using the setup. The inlet temperature varied from 320 to 370 °C, while the reactor pressure changed from 50 to 70 bars and  $H_2$ /oil ratio from 85 to 170  $Nm^3/m^3$ . Values of the parameters are shown in Table 4.

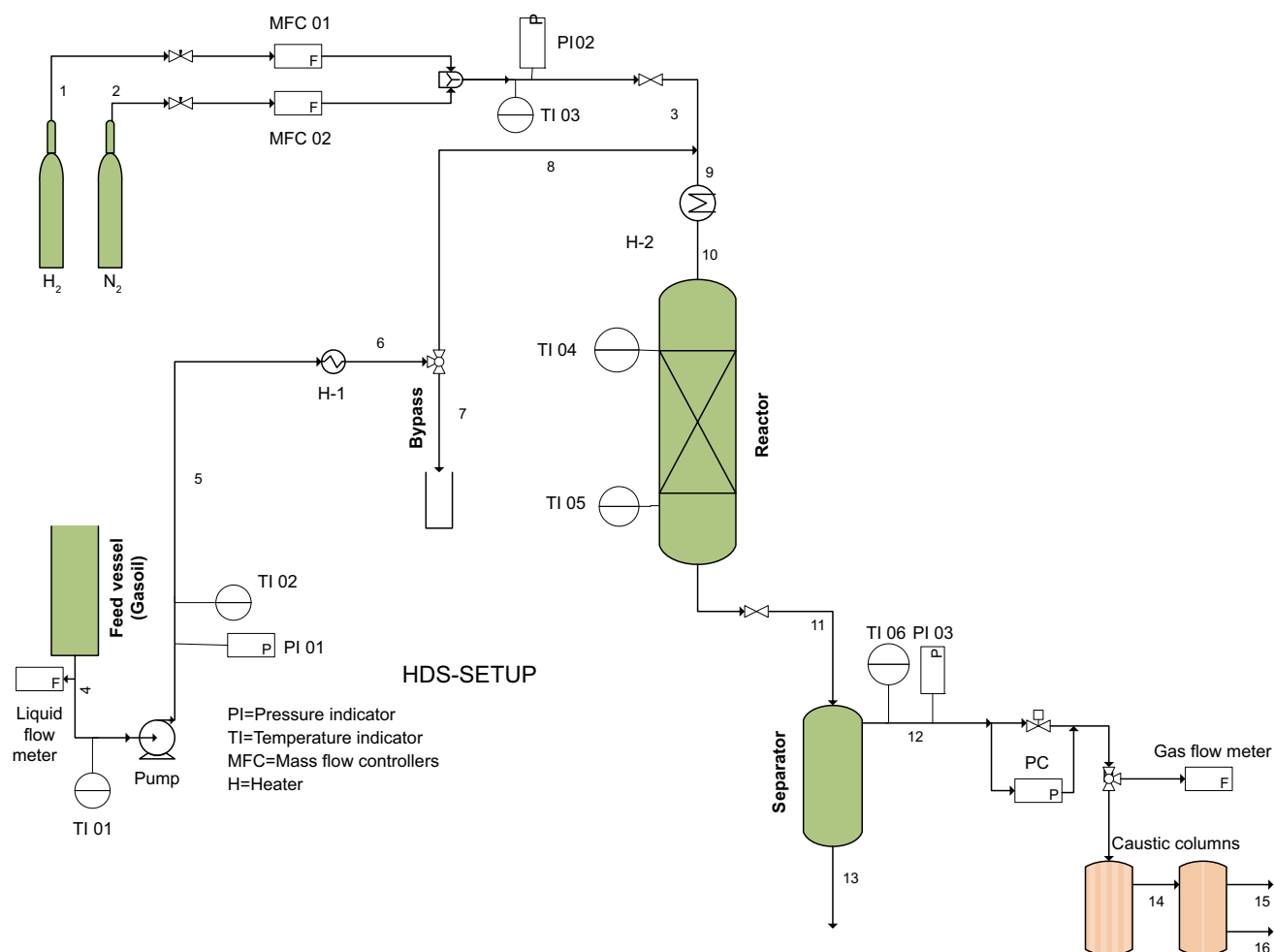
Only one factor was allowed to change in every test evaluating the parameters. Over 300 experiments were performed in the laboratory to find the values. Minimum and maximum contents in the products were 10 and 4,900 ppm wt, respectively. A single model capable of predicting the product sulfur concentration over the wide range is sought. The samples are collected based on 4 h of operation under nearly steady-state conditions. A time interval of 2 h between every experiment was required to reach the next steady-state condition. Treated gasoil sulfur content is collected from each experiment as the output values.

## 4 Development of model

The input and output variables of the SVR model were selected as shown in Table 5. In order to consider the effect of reactor (catalyst) size, the reactor outlet temperature was selected as one of the input variables of the SVR model. In this way, the trained model could be applied to industrial scale reactors independent of the (catalyst) size. A five-dimensional input vector  $X = [x_1, x_2, \dots, x_5]^T$  and the corresponding row of  $Y$  matrix denoting the one-dimensional desired (target) output vector  $Y = [y_1]^T$  were employed in training the SVR model.

The SVR model is developed using the LIBSVM package (Chang and Lin 2001). Implementation of the model was carried out using MATLAB 7.10 simulation software. The experimental results were obtained using a personal computer equipped with Intel (R) Core (TM) 2 CPU (3.0 GHz) and 3.25 GB of RAM.

To build an SVR model efficiently, the SVR parameters must be specified carefully. These parameters include (1)



**Fig. 2** Schematic of HDS set-up

**Table 2** Setup specification

<i>Reactor</i>	
Reactor diameter, m	0.0127
Reactor length, m	0.63
Catalyst bed length, m	0.11
<i>Catalyst</i>	
Chemical composition, wt% dry basis	
Cobalt	3.4
Molybdenum	13.6
Physical properties	
Surface area, m <sup>2</sup> /g	235
Pore volume, cc/g (H <sub>2</sub> O)	0.53
Flat plate crush strength, N/cm (lb/mm)	200
Attrition index	99
Compacted bulk density, g/c	0.72

Kernel function, (2) bandwidth of the kernel function ( $\sigma^2$ ), (3) regularization parameter  $C$ , and (4) the tube size of  $\varepsilon$ -insensitive loss function ( $\varepsilon$ ). Furthermore, in order to

**Table 3** Characteristics of selected gasoil

	Temperature °C	Fraction vol %
Distillation curve	171.4	0
	204.4	5
	216.2	10
	240.8	20
	263.7	30
	281.0	40
	295.3	50
	308.7	60
	322.8	70
	338.5	80
	357.5	90
Specific gravity, g/cm <sup>3</sup>	370.6	95
	371.7	100
		0.865
Total sulfur in feed, ppm		7,200

**Table 4** The parameter levels

Parameter	Level 1	Level 2	Level 3	Level 4	Level 5
Inlet temperature, °C	320	337	353	370	
Reactor pressure, bar	50	60	70		
H <sub>2</sub> /Oil ratio, nm <sup>3</sup> /m <sup>3</sup>	85	100	120	140	170
Liquid flow rate, cc/min	0.20	0.23	0.26	0.29	0.32

**Table 5** Input and output parameters for SVR model

Input variables (X)	Output variable (Y)
Hydrogen flow rate ( $Q_{H_2}$ )	Product sulfur content
Gasoil flow rate ( $Q_{gasoil}$ )	
Reactor pressure	
Inlet temperature of reactor	
Outlet temperature of reactor	

simplify the training set and to reduce training time, the VQ technique was applied. In this study, different algorithms including GS, GA, PS, and GA–SQP were applied for optimizing the SVR hyper-parameters. The  $C$  and  $g(\frac{1}{2\sigma^2})$  hyper-parameters were selected as the optimization parameters. Figure 3 represents the structure of the proposed method and details of the parameter-optimization procedure. About 70 experiments were selected randomly for testing data, and the 230 data were used as training data. According to Fig. 3, the main steps of model development were as follows:

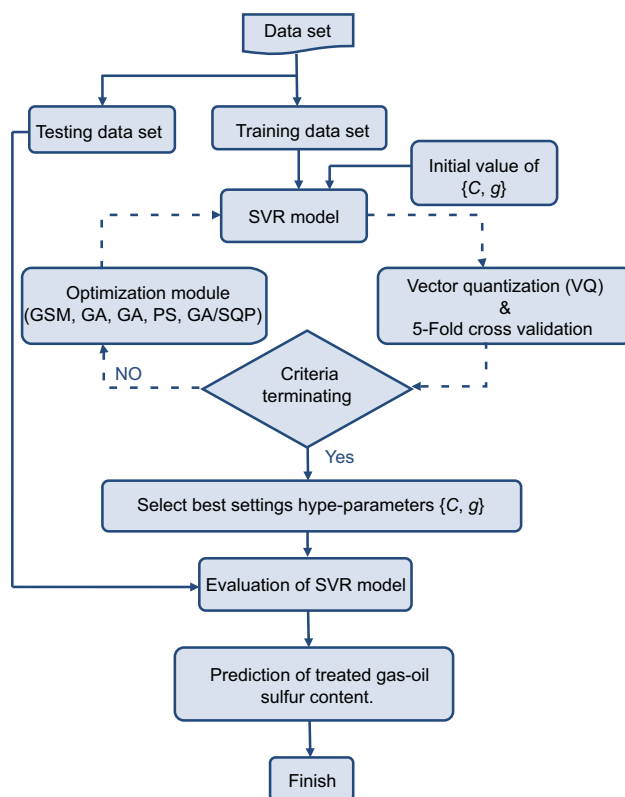
**Step 1:** Data compression: Extracting a collection of raw data and generate training and testing sets and reduce CT of the SVR model by applying the VQ technique. Thus, the SVR model was trained with low-dimensional dense datasets, which can lead to speeding up of computation with a reasonable accuracy.

**Step 2:** Selecting the SVM (the  $\varepsilon$ -SVR model was used); applying cross-validation technique (the fivefold cross-validation technique was used); and selecting the type of core kernel.

**Step 3:** Hyper-parameter optimization: Optimizing model parameters ( $C$  and  $g(\frac{1}{2\sigma^2})$ ) using GSM, GA, PS, and GA–SQP algorithms;

**Step 4:** Validating the model and predicting the sulfur content.

Hyper-parameter optimization is one of the vital challenges in SVR models. In addition to the commonly used GS, other techniques were also employed in SVR (or SVM) to correct appropriate values of hyper-parameters. Huang and Wang (2006) presented a GA-based feature selection and parameters' optimization for SVM. Also, Momma and Bennett (2002) developed a fully automated pattern search (PS) methodology for model selection of SVR.

**Fig. 3** The procedure of parameter tuning in SVR

#### 4.1 Parameter tuning of SVR with GSM

The GSM is the most common method used to determine the appropriate values of hyper-parameters. This method suffers from the main drawbacks of being very time consuming, lacks guarantee of convergence to a global optimal solution, and involves dependency on the parameters' boundary selection.

In this study, two typical ranges were selected for hyper-parameters' boundary of GSM. First,  $\log \frac{C}{2}$  and  $\log \frac{g}{2}$  varied between  $[-3 \ 3]$  and  $[-5 \ 4]$ , respectively. Then,  $\log \frac{C}{2}$  and  $\log \frac{g}{2}$  varied between  $[-2 \ 2]$  and  $[-3 \ 2]$ , respectively. Since  $\varepsilon$  has little effect on ARRE, it was assumed to be 0.01. Typical results by this method are shown in the Table 7.

Selecting a wide range for this method can increase the accuracy but the CT would become very long. Since the accuracy of the SVR model depends on a proper setting of



SVR hyper-parameters, some optimization algorithms have been developed.

#### 4.2 Optimizing the SVR parameters based on GA

The concept of GA was developed by Holland (1975). GA is a heuristic search method that mimics the process of natural evolution. Furthermore, it is a stochastic search technique that can be used in finding the global optimum solution in a complex multidimensional search space. It can search large and complicated spaces using ideas from natural genetics and the evolutionary principle (Goldberg 1989). In this work, the procedure for hyper-parameter optimization with GA method is summarized in the following steps:

- (1) *Start* Initialize the parameters for GA and choose a randomly generated population, population size, the number of subpopulations and individuals per subpopulation, the type of kernel function, and the range of the SVR parameters. The SVR hyper-parameters  $\{C, g, \text{ and } \varepsilon\}$  are directly coded to generate the chromosome randomly.
- (2) *Calculating the fitness* The fitness function is defined as the AARE cross-validation on the training dataset as follows:

$$\begin{aligned} \text{Min}(\text{Fitness}) &= \text{Min}(\text{AARE of CV}) \\ &= \text{Min} \frac{1}{k} \sum_{i=1}^k \left( \frac{k}{m} \sum_{i=1}^{m/k} \left| \frac{(\text{Exp}_i - \text{Pre}_i)}{\text{Exp}_i} \right| \right) \end{aligned} \quad (5)$$

where  $\text{Exp}_i$  and  $\text{Pre}_i$  are the actual values and the predicted values, respectively. In this research, a fivefold cross-validation method was being used ( $k = 5$ ).  $m$  denotes the total number of training sets ( $m = 230$ ).

- (3) *Creating the offspring by genetic operators* To select the subpopulation individuals for the mating pool. The integration of discrete recombination and line recombination is applied to randomly paired chromosomes, which determines whether a chromosome should be mutated in the next generation.
- (4) *Elitist strategy* Elitist reinsertion is used to prevent losing good information and is a recommended method.
- (5) *Migration* The migration model is used to divide the population into multiple subpopulations.
- (6) *Check the termination condition* If the executed generation number equals the special generation number, the algorithm ends; otherwise, it goes back to step 2. The GA creates generations by selecting and reproducing parents until termination criteria are met.

#### 4.3 Parameters tuning of SVR based on the PS algorithm

The PS method is a class of direct search methods to solve nonlinear optimization problems. The PS algorithm can calculate the function values of a pattern and tries to find the minimum value. For the hyper-parameter optimization with the PS algorithm, the procedures are summarized in the following steps:

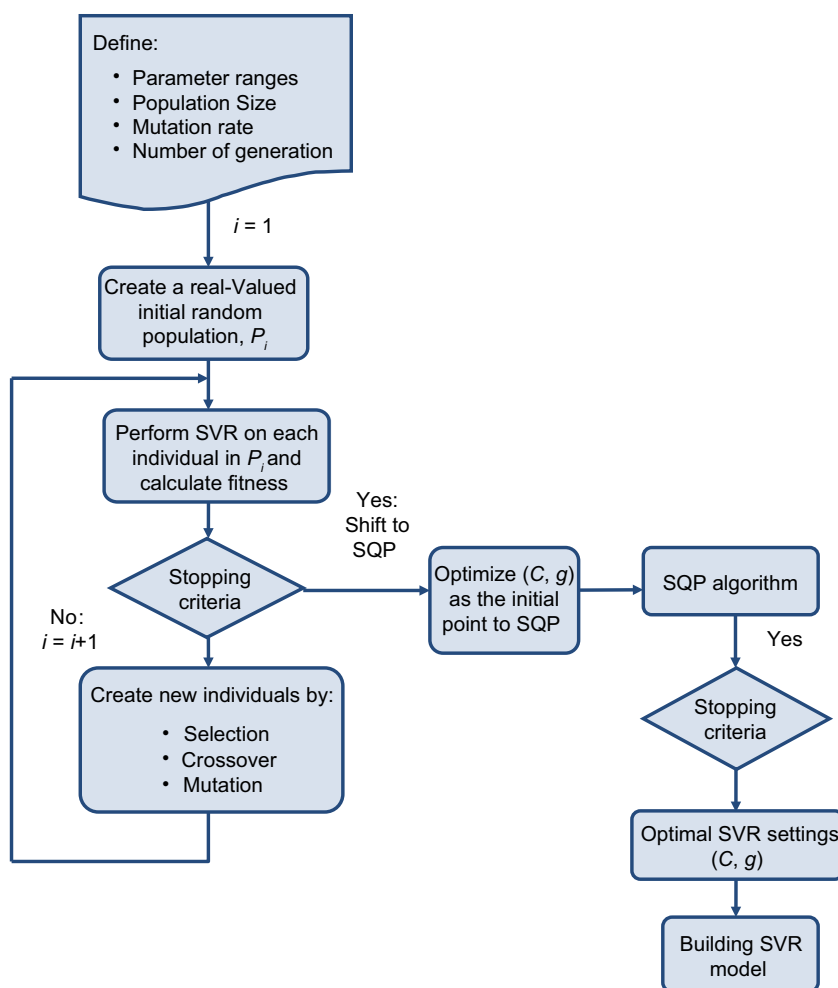
- (1) Parameters setting, set iteration  $i = 0$
- (2) Set iteration  $i = i + 1$
- (3) Model training: Hyper-parameter optimization, five-fold CV
- (4) Fitness definition and evaluation
- (5) Termination: The evolutionary process proceeds until a stopping criterion is met (maximum iterations predefined or the error accuracy of the fitness function). Otherwise, we go back to step (2).

#### 4.4 Parameter tuning of SVR based on GA–SQP hybrid algorithm

This method relies on both local search and global search techniques. The SQP method is a deterministic method, while the GA is a stochastic method. The SQP method is one of the most effective gradient-based algorithms for constrained nonlinear optimization problems. The method is sensitive to initial point selection. It can guarantee local optima as it follows a gradient search direction from the starting point toward the optimum point. GA is efficient for global optimization by finding the most promising regions of search space. Hybridization of GA and SQP can complement the qualities of GA by focusing on accuracy and solution time. The GA is first applied to produce the proper estimation point for SQP. In other words, GA and SQP were used in series.

The algorithm starts with the GA, since the SQP is sensitive to the initial point. Therefore, GA is the main optimizer, and the SQP is used to fine tune for improvement of the every solution of the GA. GA has shown to be efficient on global optimization by finding the most promising regions of search space; however, it suffers from excessive solution time and low accuracy. On the other hand, the SQP can complement the qualities of GA by focusing on accuracy and solution time. GA can be applied first in order to refine the initial point, and then the SQP will be able to reach the solution fast. In other words, the calculation continues with the GA for a specific number of generations or a user-specified number for stall generation during which the approximate solution becomes closer to the real solution. The algorithm then shifts to the SQP which is a faster method. Details of the procedure are illustrated in the flowchart shown in Fig. 4.

**Fig. 4** Flow diagram of the combined GA–SQP and SVR for parameter optimization



According to Fig. 4, the procedure of hyper-parameter optimization with GA–SQP method is summarized as follows:

- (1) *Start* To define the parameters for GA and choose a randomly generated population, population size, the number of subpopulations and individuals per subpopulation, the mutation rate, type of kernel function, and the range of the SVR parameters.
- (2) *Calculating the fitness* The fitness function is defined as the AARE cross-validation on the training dataset as per Eq. 5.
- (3) *Creating the offspring by genetic operators* The GA uses selection, crossover, and mutation operators to generate the offspring of the existing population. Offspring replaces the old population and forms a new population in the next generation. The evolutionary process proceeds until a stopping criterion is satisfied.
- (4) *Shift to the SQP* The GA creates generations by selecting and reproducing parents until a stopping

criterion is met. One of the stopping criteria is a specified maximum number of generations. Another stopping strategy involves population convergence criteria. After satisfying a stopping criterion, the algorithm shifts to the SQP method. The search will continue until a stopping criterion is satisfied.

## 5 Results and discussion

In this research, over 300 experiments were conducted on a pilot scale hydro-desulfurization set up. Gas–oil flow rate,  $H_2$  flow rate, reactor pressure, and inlet temperature were chosen as the different operating parameters in the experiments. The gas–oil sulfur contents used in experiments varied from 10 to 4,900 ppm. Besides the mentioned parameters, the reactor outlet temperature was also selected as an input parameter of the SVR model. This facilitates the application of the developed SVR model to simulate the behavior of industrial reactors. Note that when catalyst



deactivation occurs during the time, the outlet temperature would change for the same input conditions.

One of the important factors in forecasting performance of SVR is the kernel function. In this work, different kernels namely linear kernel, polynomial kernel, sigmoid, and radial basis function (RBF) kernel were used, and the effects of these kernel functions on SVR model based on GS-optimization method are summarized in Table 6. The results show that SVR model with Gaussian (RBF) kernel provides a lower AARE. Furthermore, in order to obtain better accuracy of SVR model and data compression, the VQ technique was employed. The impacts of VQ on CT and prediction accuracy of SVR model are shown in Table 6. The VQ technique can reduce the CT and simultaneously improve the accuracy of the SVR model.

The most important factor influencing the efficiency and robustness of the SVR algorithm is hyper-parameter tuning. Hence, the optimization method is the most critical factor to determine the convergence speed of the SVR model and the ability to search for the global optimal solution.

The effects of different optimization methods on the SVR model are shown in Table 7. The performance of

these methods was evaluated by the statistical criteria (AARE and  $R^2$ ).

It is seen that the GS results depend completely on the boundary value of  $C$  and  $g$ . PS gives a better result of AARE and  $R^2$  than GA; however, the GA–SQP algorithm gives the best AARE,  $R^2$ , and CT.

From the results, it can be concluded that the performances of the PS, GA, and GA–SQP integrated with SVR are relatively superior to GSM integrated with SVR. On the other hand, integrating these methods (PS, GA, and GA–SQP) with SVR presented attractive advantages compared with GSM and SVR, as follows:

- (1) Optimization of the SVR parameters without drawbacks of GSM.
- (2) Reduction of computational time.

Some of the results from the hybrid GA–SQP algorithm are shown in Table 8. As seen in this table, integration of GA–SQP with SVR model has good accuracy for prediction of sulfur content of the treated gas–oil in a wide range.

The parity plots for different optimization algorithms integrated with SVR model are shown in Fig. 5. It shows that the

**Table 6** The impact of kernel function and VQ on prediction by SVR model

	Kernel type	AARE (with VQ)	AARE (without VQ)	CT (with VQ)	CT (without VQ)
1	Linear	1.295	4.274	228	327
2	Polynomial	0.227	2.894	264	368
3	Sigmoid	0.246	2.928	243	339
4	Gaussian	0.083	0.546	312	595

**Table 7** Optimal SVR hyper-parameters obtained by different algorithms ( $\epsilon = 0.1$ )

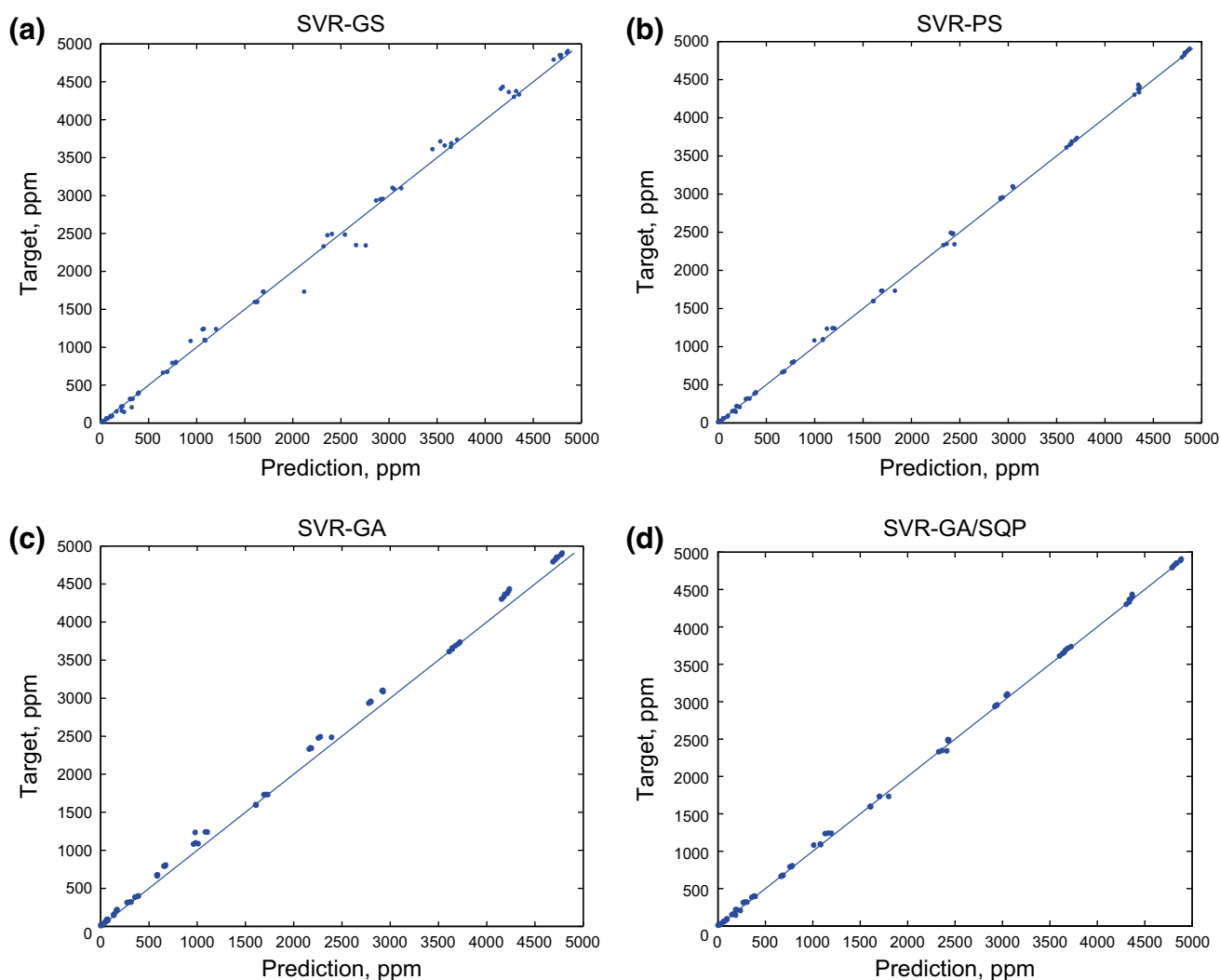
No.	Method	Boundary	$C$	$g$	Type of kernel	AARE (training)	AARE (test)	$R^2$ (training)	$R^2$ (test)	CT(S)
1	GS–SVR	$C:2^{[-2\ 2]}, g:2^{[-3\ 2]}$	4.0	0.125	RBF	0.0978	0.1063	0.985	0.983	291
2	GS–SVR	$C:2^{[-3\ 3]}, g:2^{[-5\ 4]}$	8.0	0.125	RBF	0.0885	0.0893	0.988	0.986	323
3	PS–SVR	$C:[0.01\ 1e4], g:[0.01\ 1e4]$	52.0	0.100	RBF	0.0723	0.0828	0.996	0.995	107
4	GA–SVR	$C:[0.01\ 1e4], g:[0.01\ 1e4]$	48.9	0.030	RBF	0.0734	0.0844	0.995	0.994	112
5	GA–SQP–SVR	$C:[0.01\ 1e4], g:[0.01\ 1e4]$	50.5	0.099	RBF	0.0652	0.0745	0.998	0.997	56

**Table 8** Typical input and output data for the SVR testing with GA–SQP method

Test no.	$Q_{\text{gasoil}}$ , kg/s	$Q_{\text{H}_2}$ , m <sup>3</sup> /h	$P$ , kPa	$T_{\text{in}}$ , °C	$T_{\text{out}}$ , °C	$S_{\text{exp}}$ , ppm	$S_{\text{pre}}$ , ppm
1	2.85E–06	3.54E–05	5,000	320	322.27	3,660	3,652
2	2.85E–06	3.54E–05	5,000	337	339.24	1,598	1,586
3	2.85E–06	3.54E–05	5,000	353	355.76	313	273
4	2.85E–06	6.90E–05	5,000	370	373.28	15	20
5	3.71E–06	6.90E–05	5,000	320	320.66	4,333	4,354
6	3.71E–06	6.90E–05	5,000	337	338.47	2,346	2,354
7	3.71E–06	6.90E–05	5,000	353	355.99	675	670

**Table 8** continued

Test no.	$Q_{\text{gasoil}}$ , kg/s	$Q_{\text{H}_2}$ , m <sup>3</sup> /h	$P$ , kPa	$T_{\text{in}}$ , °C	$T_{\text{out}}$ , °C	$S_{\text{exp}}$ , ppm	$S_{\text{pre}}$ , ppm
8	4.56E−06	6.90E−05	5,000	337	338.92	2,950	2,921
9	2.85E−06	6.90E−05	5,000	320	321.27	3,644	3,634
10	2.85E−06	6.90E−05	5,000	353	356.44	320	297
11	2.85E−06	3.54E−05	7,000	320	321.18	3,737	3,705
12	2.85E−06	3.54E−05	7,000	337	338.59	1,733	1,819
13	2.85E−06	3.54E−05	7,000	353	356.43	384	369
14	2.85E−06	3.54E−05	7,000	370	373.56	18	22
15	4.56E−06	6.90E−05	7,000	320	320.60	4,885	4,859
16	4.56E−06	6.90E−05	7,000	337	338.44	3,097	3,054
17	3.71E−06	1.13E−04	7,000	320	320.60	4,377	4,343
18	3.71E−06	1.13E−04	7,000	337	338.87	2,477	2,429
19	3.71E−06	1.13E−04	7,000	353	356.33	805	7,83
20	2.85E−06	1.13E−04	7,000	337	339.33	1,731	1,691

**Fig. 5** The parity plot for different algorithms. **a** Hyper-parameter optimization using GS. **b** Hyper-parameter optimization using PS. **c** Hyper-parameter optimization using GA. **d** Hyper-parameter optimization using GA–SQP

SVR model is a robust and reliable model to predict the treated gas–oil sulfur content, no matter what algorithm is being selected for hyper-parameter optimization. Consequently, the model can be applied with good confidence to predict sulfur content in the industrial plants with any characteristics.

## 6 Conclusion

The aim of this study was to improve the prediction performance and the CT of a data-driven, soft sensor used in the production of ultra-low sulfur diesel. A novel, soft sensor model integrating VQ technique with SVR model was proposed. Selection of optimal parameters of the model is a vital challenge directly affecting prediction accuracy. An integrated GA and SQP (GA–SQP) optimization procedure that is a relatively a fast alternative to the time-consuming GS approach was employed.

The other important factor in the predictive performance of SVR model is kernel function. Four different kernels, namely, linear, polynomial, sigmoid, and Gaussian kernels were evaluated. Results show that the SVR model with a Gaussian (RBF) kernel gives a lower AARE. The model was validated against a wide range of experimental data taken from the gas–oil HDS set-up. The results revealed that the proposed VQ–SVR model coupled with hybrid GA–SQP optimization algorithm is superior to other methods and gives the best prediction for the sulfur content with the highest accuracy (AARE = 0.0745,  $R^2$  = 0.997) and the lowest computation time (CT = 56 s).

The proposed approach can pave the way for design of reliable data-driven soft sensors in petroleum industries.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

## References

- An S, Liu W, Venkatesh S. Fast cross-validation algorithms for least squares support vector machine and kernel ridge regression. *Pattern Recogn.* 2007;40(8):2154–62.
- Basak D, Pal S, Patranabis DC. Support vector regression. *Neural Inf Process.* 2007;11(10):203–25.
- Boser B, Guyon I, Vapnik V. A training algorithm for optimal margin classifiers. In: *Proceedings of 5th Annual Workshop on Computer Learning Theory*, Pittsburgh: ACM. 1992;144–152.
- Bolf N, Galinec G, Baksa GT. Development of soft sensor for diesel fuel quality estimation. *Chem Eng Technol.* 2010;33(3):405–13.
- Chang CC, Lin CJ. LIBSVM: a library for support vector machines; software. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>. 2001.
- Chen K, Wang C. Support vector regression with genetic algorithms in forecasting tourism demand. *Tour Manag.* 2007;28(1):215–26.
- Cristianini N, Taylor JS. *An introduction to support vector machines*. London: Cambridge University Press; 2000.
- Duan K, Keerthi S, Poo A. Evaluation of simple performance measures for tuning SVM hyperparameters. *Neuro Comput.* 2003;51:41–59.
- Froment GF. Modeling in the development of hydrotreatment processes. *Catal Today.* 2004;98(1–2):43–54.
- Gunn SR. *Support vector machines for classification and regression*, ISIS Technical Report. Southampton: University of Southampton; 1998.
- Gersho A, Gray RM. *Vector quantization and signal compression*. Dordrecht: Kluwer Academic Publishers; 1992.
- Goldberg DE. *Genetic algorithms in search, optimization and machine learning*. Boston: Addison-Wesley Longman Publishing Co, Inc.; 1989.
- Holland JH. *Adaptation in Natural and Artificial Systems*. Michigan: University of Michigan Press; 1975.
- Horng MH. Vector quantization using the Firefly algorithm for image compression. *Expert Syst Appl.* 2012;39(1):1078–91.
- Huang CL, Wang CJ. A GA-based feature selection and parameters optimization for support vector machines. *Expert Syst Appl.* 2006;31(2):231–40.
- Huang CF. A hybrid stock selection model using genetic algorithms and support vector regression. *Appl Soft Comput.* 2012;12(2):807–18.
- Kartik CKN, Narasimhan S. A theoretically rigorous approach to soft sensor development using principle components analysis. In: *21st European Symposium on Aided Process Engineering-ESCAPE2*. 2011;793–798.
- Kadlec P. On robust and adaptive soft sensors. PhD Thesis, Bournemouth: Bournemouth University; 2009.
- Kadlec P, Gabrys B, Strandt S. Data-driven soft sensors in the process industry. *Comput Chem Eng.* 2009;33(4):795–814.
- Kadlec P, Grbic R, Gabrys B. Review of adaptation mechanisms for data-driven soft sensors. *Comput Chem Eng.* 2011;35(1):1–24.
- Korsten H, Hoffmann U. Three-phase reactor model for hydrotreating in pilot trickle-bed reactors. *AIChE J.* 1996;42(5):1350–60.
- King SL, Bennett KP, List S. Modeling noncatastrophic individual tree mortality using logistic regression, neural networks, and support vector methods. *Comput Electron Agric.* 2000;27(1–3):401–6.
- Liu G, Zhou D, Xu H, et al. Model optimization of SVM for a fermentation soft sensor. *Expert Syst Appl.* 2010;37(4):2708–13.
- Li Y, Yuan J. Prediction of key state variables using support vector machines in bioprocesses. *Chem Eng Technol.* 2006;29(3):313–9.
- Lu CJ, Lee TS, Chiu CC. Financial time series forecasting using independent component analysis and support vector regression. *Decis Support Syst.* 2009;47(2):115–25.
- Min JH, Lee YC. Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. *Expert Syst Appl.* 2005;28(4):603–14.
- Momma M, Bennett KP. A pattern search method for model selection of support vector regression. In: *Second SIAM International Conference on Data Mining*. 2002.
- Pan M, Zeng D, Xu G. Temperature prediction of hydrogen producing reactor using SVM regression with PSO. *J Comput.* 2010;5(3):388–93.
- Si F, Romero CE, Ya Z, et al. Inferential sensor for on-line monitoring of ammonium bisulfate formation temperature in coal-fired power plants. *Fuel Process Technol.* 2009;90(1):56–66.
- Somasundaram K, Vimala S. Fast encoding algorithm for vector quantization. *Int J Eng Sci Technol.* 2010;2(9):4876–9.
- Vapnik V, Golowich S, Smola A. Support vector method for function approximation, regression estimation and signal processing. In: *Mozer M, Jordan M, Petsche T, editors. Advances in neural information processing systems*. 9th ed. Cambridge: MIT Press; 1996. p. 281–7.
- Vapnik VN. *The nature of statistical learning theory*. New York: Springer; 1995.

- Wu CH, Tzeng GH, Lin RH. A novel hybrid genetic algorithm for kernel function and parameter optimization in support vector regression. *Expert Syst Appl.* 2009;36(3):4725–35.
- Yan W, Shao H, Wang X. Soft sensing modeling based on support vector machine and bayesian model selection. *Comput Chem Eng.* 2004;28(8):1489–98.
- Yeh CY, Huang CW, Lee SJ. A multiple-kernel support vector regression approach for stock market price forecasting. *Expert Syst Appl.* 2011;38:2177–86.
- Zahedi S, Shokri S, Baloochi B, et al. Analysis of sulfur removal in gasoil hydrosulfurization process by application of response surface methodology. *Korean J Chem Eng.* 2011;28(1):93–8.