Tech Science Press

Adaptive Multi-Layer Selective Ensemble Least Square Support Vector Machines with Applications

Gang Yu[1,4,5], Jian Tang[2,*], Jian Zhang[3] and Zhonghui Wang[6]

[1]State Key Laboratory of Process Automation in Mining & Metallurgy, Beijing, 102600, China
[2]Faculty of Information Technology, Beijing University of Technology, Beijing, 100024, China
[3]School of Computer and Software, Nanjing University of Information Science & Technology, 210044, China
[4]Beijing Key Laboratory of Process Automation in Mining & Metallurgy, Beijing, 102600, China
[5]BGRIMM Technology Group Co., Ltd., Beijing, 102600, China
[6]Department of Computer and Information Science, The University of Mississippi, Mississippi, MS 38655, USA
[*]Corresponding Author: Jian Tang. Email: freeflytang@bjut.edu.cn

Abstract: Kernel learning based on structure risk minimum can be employed to build a soft measuring model for analyzing small samples. However, it is difficult to select learning parameters, such as kernel parameter (KP) and regularization parameter (RP). In this paper, a soft measuring method is investigated to select learning parameters, which is based on adaptive multi-layer selective ensemble (AMLSEN) and least-square support vector machine (LSSVM). First, candidate kernels and RPs with K and R numbers are preset based on prior knowledge, and candidate sub-sub-models with K*R numbers are constructed through utilizing LSSVM. Second, the candidate sub-sub-models with same KPs and different RPs are selectively fused by using the branch and bound SEN (BBSEN) to obtain K SEN-sub-models. Third, these SEN-sub-models are selectively combined through using BBSEN again to obtain SEN models with different ensemble sizes, and then a new metric index is defined to determine the final AMLSEN-LSSVM-based soft measuring model. Finally, the learning parameters and ensemble sizes of different SEN layers are obtained adaptively. Simulation results based on the UCI benchmark and practical DXN datasets are conducted to validate the effectiveness of the proposed approach.

Keywords: Multi-layer selective ensemble learning; least square support vector machine; soft measuring model; municipal solid waste incineration; dioxins emission

1 Introduction

Data-driven soft measuring techniques can be used for online estimation of offline assay process parameters and experts' estimation quantity variables [1,2]. Especially, soft measuring models are used in many fields according to their inferential estimation capability [3], and the two most common ones include artificial neural networks (ANN) and support vector machines (SVM). Although ANN is used to model DXN emission concentration [4], it has several shortcomings, such as easily falling into local

minimum, over fitting, and unstable generalization performance in terms of small samples. For SVM, it is more suitable to build a prediction model for DXN emission, however, its prediction performance heavily depends on kernel and RPs [5]. Furthermore, SVM has to suffer from the quadratic program (QP) problem, which can be solved by least square-support vector machines (LSSVMs) including a set of linear equations. Due to the data dependence of kernel learning methods [6], some optimization methods are employed to address this problem through using single- or multi-objective optimization in terms of learning parameters [7,8]. However, such methods are time-consumed and prone to obtain a sub-optimum solution [9]. In addition, as the KP determines the geometry of the feature space, it can be optimally selected through class reparability [10,11]. Thus, this parameter is also calculated by using Fisher discrimination of classification problems [12]. For regression problems, a stable generalization performance can only be found within a certain range. However, the evaluation process must be dynamic and the searching results are unstable for small samples. Therefore, a fast and definitive method for KP selection should be developed in future researches. The RP of kernel learning method has no intuitive meaning in terms of geometry and is normally determined through cross-validation or optimization search methods [13]. In Wang et al. [14], a KP selection approach is proposed for small high-dimensional mechanical frequency spectral samples, which only selects a single kernel. Moreover, multiple kernel learning algorithms exhibit high efficiency and effectiveness in both classification and regression tasks [14]. And one of them named multiple kernel ensemble learning approach has been applied into hyper-spectral remote sensing image classification [15,16]. However, the optimized selection of learning parameters is not addressed jointly. Therefore, a new adaptive method for selecting learning parameters should be developed to model small samples with complex characteristics.

Selective ensemble (SEN) modeling can selectively fuse multiple sub-models in a linear or nonlinear method and achieve better prediction performances than single modeling. However, SEN modeling still exerts several limitations. One of them is ensemble construction, which creates a set of candidate sub-models for the same training dataset. The ensemble construction method based on the resample of training samples validates the ensemble method, and many available sub-models can obtain better performances than the ensemble of all the sub-models [17]. However, the selection problem of learning parameters remains unsolved in this scenario. Consequently, based on the above ensemble construction strategy, double-layer GA-based SEN latent structure modeling is proposed for collinear and nonlinear data [18]. However, this method has disadvantages of long searching time and randomized prediction results. Another ensemble construction method named manipulation of input features is used to model multi-source multi-scale high-dimensional frequency spectral data [13,19,20], and it can construct a soft measuring model by using the interesting mechanical sub-signals and their spectral feature subsets, which focuses on the selective fusion of different multi-source feature sub-sets from the perspective of selective information fusion. As a result, this method is thus suitable for modeling frequency spectral data with multi-source and high dimension, especially for small samples.

In this paper, the SEN kernel leaning algorithm for small data-driven samples is adopted to softly measure the DXN emission. Although several SEN-LSSVMs such as fuzzy C-means cluster-based SEN-LSSVM [21] and evolutionary programming (EP)-based multi-level LSSVM [22] have been proposed, they do not address the selection problem of learning parameters and are unsuitable for small samples. As multiple KPs can clearly describe the complex characteristics of small sample data (e.g., DXN), a soft measuring model for the SEN-LSSVM-based DXN emission concentration can be built through using multiple candidate KPs and the ensemble construction strategy. Meanwhile, the RP of the kernel learning algorithm is also data-dependent. Therefore, the algorithm can also be used and selected with the same idea like KPs.

Motivated by the above problems, this paper proposes a new adaptive multi-layer SEN-LSSVM (AMLSEN-LSSVM) method for modeling small samples with complex characteristics. Compared to the

existing literatures, the distinctive contributions of this study are listed as follows. (1) A new SEN-LSSVM modeling framework is proposed to model small samples according to the candidate learning parameters at the first time; (2) The proposed method can adaptively select the kernel and RPs simultaneously. Moreover, the ensemble sizes of different SEN layers are also adaptively determined in implicit pattern; (3) A new metric index is defined for selecting the final AMLSEN-LSSVM soft measuring model to achieve tradeoff between model complexity and prediction performance.

The remainder of this paper is organized as follows. Section 2 describes the proposed modeling strategy. In Section 3, a detailed realization of the proposed approach is demonstrated. Section 4 presents the experimental results on two UCI benchmarks and practical DXN datasets for references. Finally, Section 5 concludes this paper.

2 Modeling Strategy Description

Based on the above analysis, an adaptive multi-layer SEN-LSSVM (AMLSEN-LSSVM) soft measuring strategy is proposed, which consists of candidate sub-sub-models, candidate SEN-sub-models, SEN-models, and decision selection modules. The process is shown in Fig. 1.
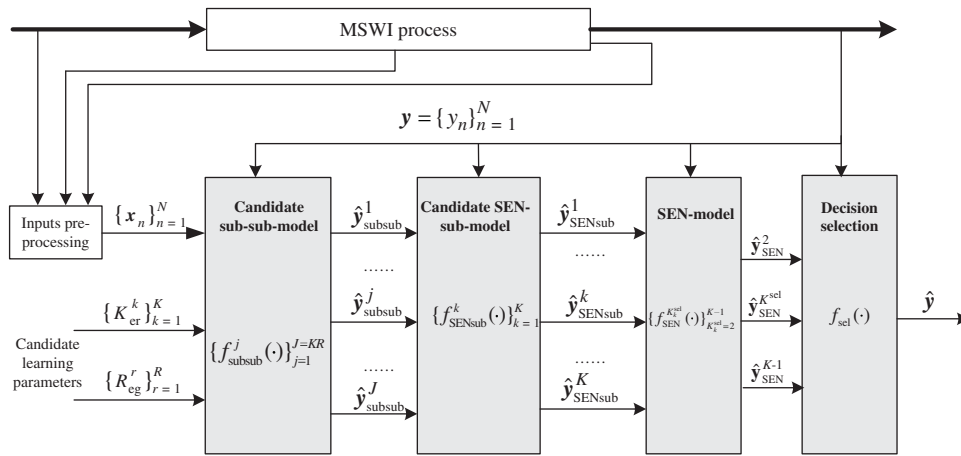

Figure 1: SEN-LSSVM-based soft measuring strategy

In Fig. 1, $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ is the input of the modeling data with features from the different phases of the MSWI process; $\mathbf{y} = \{y_n\}_{n=1}^N$ denotes the predicted data of the offline assay process; $\{K_{er}^k\}_{k=1}^K$ and $\{R_{eg}^r\}_{r=1}^R$ are the K candidate kernels and the R RPs, respectively; $\{f_{subsub}^j(\cdot)\}_{j=1}^{J=KR}$ indicates the $K \times R$ candidate sub-sub-models set; $\hat{\mathbf{y}}_{subsub}^j$ is the prediction output of the jth sub-sub-model $f_{subsub}^j(\cdot)$; $\{f_{SENsub}^k(\cdot)\}_{k=1}^K$ is the K candidate SEN-sub-model sets; $\hat{\mathbf{y}}_{SENsub}^k$ is the prediction output of the kth SEN-sub-model $f_{SENsub}^k(\cdot)$; $\{f_{SEN}^{K_k^{sel}}(\cdot)\}_{K_k^{sel}=2}^{K-1}$ is the set of candidate SEN models; $\hat{\mathbf{y}}_{SEN}^{K^{sel}}$ is the prediction output of the SEN model with an ensemble size K^{sel}; $f_{sel}(\cdot)$ and $\hat{\mathbf{y}}$ are the final SEN soft sensor model and its prediction output, respectively.

The functions of the different modules are illustrated as follows:

(1) Candidate sub-sub-model module: Construct the $K \times R$ LSSVM sub-sub-models by using different candidate learning parameters.

(2) Candidate SEN-sub-model module: Construct the K SEN-sub-models by using the prediction outputs of R sub-sub-models based on the same KP and R different RPs.

(3) SEN-model module: Construct the SEN-models with an ensemble size from 2 to $(K-1)$ by using the prediction outputs of K SEN-sub-models based on different KPs.

(4) Decision selection module: Select the final soft measuring model from SEN-models with the different ensemble sizes by defining a new metric index and making a trade-off between prediction accuracy and model complexity.

3 Modeling Realization

3.1 Candidate Sub-Sub-Model Module

As the candidate kernels and RPs of the LSSVM model are respectively denoted as $\{K_{er}^k\}_{k=1}^K$ and $\{R_{eg}^r\}_{r=1}^R$, the candidate learning parameters can be used to obtain the following matrix:

$$M_{\text{para}} = \begin{bmatrix} [K_{er}^1, R_{eg}^1] & \cdots & [K_{er}^1, R_{eg}^r] & \cdots & [K_{er}^1, R_{eg}^R] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [K_{er}^k, R_{eg}^1] & \cdots & [K_{er}^k, R_{eg}^r] & \cdots & [K_{er}^k, R_{eg}^R] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ [K_{er}^K, R_{eg}^1] & \cdots & [K_{er}^K, R_{eg}^r] & \cdots & [K_{er}^K, R_{eg}^R] \end{bmatrix}_J \tag{1}$$

where $[K_{er}^k, R_{eg}^r]$ is the jth element of learning parameter matrix M_{para}, in which $M_{\text{para}}^j = [K_{er}^k, R_{eg}^r]$ with $j = 1, \cdots, J$. Especially, $J = K \times R$ is the element number of matrix M_{para}, that is, the number of LSSVM sub-sub-models.

Taken the jth pair $[K_{er}^k, R_{eg}^r]$ of learning parameters for an example, the process of the sub-sub-model is built as follows. First, nonlinear mapping function $\varphi(\mathbf{x}_n)$ is used to map $\{\mathbf{x}_n\}_{n=1}^N$ into a higher dimensional feature space. Then, the optimization problem is solved by the LSSVM algorithm:

$$\begin{cases} \min\limits_{\mathbf{w}^j, b^j} & O^j_{\text{LS-SVM}} = \frac{1}{2}\mathbf{w}^{jT}\mathbf{w}^j + \frac{1}{2}R_{eg}\sum\limits_{n=1}^N \zeta^j_n{}^2 \\ s.t: & \hat{y}^j_n = \mathbf{w}^{jT}\varphi(\mathbf{x}_n^j) + b^j + \zeta^j_n \end{cases} \tag{2}$$

where $\mathbf{w}^j$ is the weighted coefficient, b^j indicates the bias, and ζ^j_n denotes the prediction error of the nth sample. Thus, the Lagrange method can be employed to solve the optimization problem (2),

$$L^j(\mathbf{w}^j, b^j, \boldsymbol{\zeta}^j, \boldsymbol{\beta}^j) = \frac{1}{2}\mathbf{w}^{jT}\mathbf{w}^j + \frac{1}{2}\sum\limits_{n=1}^N (\zeta^j_n)^2 - \sum\limits_{n=1}^N \beta^j_n[\mathbf{w}^{jT}\varphi(\mathbf{x}^j_n) + b^j + \zeta^j_n - \hat{y}^j_n] \tag{3}$$

where $\boldsymbol{\beta}^j = [\beta^j_1, \cdots, \beta^j_n, \cdots, \beta^j_N]$ is the Lagrange operator vector, and $\boldsymbol{\zeta}^j = [\zeta^j_1, \cdots, \zeta^j_n, \cdots, \zeta^j_N]$ is the prediction error vector. Taken the derivation of the parameters in Eq. (3), the optimum solution can be gotten by

$$\frac{\partial L^j}{\partial \mathbf{w}^j} = 0, \frac{\partial L^j}{\partial b^j} = 0, \frac{\partial L^j}{\partial \xi^j} = 0, \frac{\partial L^j}{\partial \beta^j} = 0, \tag{4}$$

$$\Omega^j_{\text{ker}}(\mathbf{x}, \mathbf{x}_n) = \ <\varphi(\mathbf{x}) \cdot \varphi(\mathbf{x}_n)> \ = f^j_{\text{ker}}(\mathbf{x}, \mathbf{x}_n, K_{\text{ker}}), \tag{5}$$

where $\Omega^k_{\text{ker}}(\cdot)$ is the kernel function with parameter K_{er}^k.

Accordingly, the above problem becomes a linear equation system:

$$
\begin{bmatrix}
0 & 1 & \cdots & 1 \\
1 & f_{\mathrm{ker}}^{j}(\mathbf{x}_1, \mathbf{x}_1, K_{\mathrm{ker}}), + \dfrac{1}{R_{\mathrm{eg}}^{r}} & \cdots & f_{\mathrm{ker}}^{j}(\mathbf{x}_1, \mathbf{x}_{N1}, K_{\mathrm{ker}}) \\
\vdots & \vdots & \vdots & \vdots \\
1 & f_{\mathrm{ker}}^{j}(\mathbf{x}_N, \mathbf{x}_1, K_{\mathrm{ker}}) & \cdots & f_{\mathrm{ker}}^{j}(\mathbf{x}_N, \mathbf{x}_N, K_{\mathrm{ker}}) + \dfrac{1}{R_{\mathrm{eg}}^{r}}
\end{bmatrix}
\cdot
\begin{bmatrix}
b^{j} \\
\beta^{j}{}_{1} \\
\vdots \\
\beta^{j}{}_{N}
\end{bmatrix}
=
\begin{bmatrix}
1 \\
y_1 \\
\vdots \\
y_N
\end{bmatrix}
\tag{6}
$$

By solving the above system, β^{j} and b^{j} are obtained. Then, the LSSVM sub-sub-model can be denoted as

$$
\hat{\mathbf{y}}_{\mathrm{subsub}}^{j} = \sum_{n=1}^{N} \beta^{j}{}_{n} \cdot \Omega_{\mathrm{ker}}^{j}(\mathbf{x}, \mathbf{x}_n) + b^{j}
\tag{7}
$$

For a concise expression, (7) can also be expressed as

$$
\hat{\mathbf{y}}_{\mathrm{subsub}}^{j} = f_{\mathrm{subsub}}^{j}(\mathbf{x}; M_{\mathrm{para}}^{j}) = f_{\mathrm{subsub}}^{j}(\mathbf{x}^{j}; K_{\mathrm{er}}^{k}, R_{\mathrm{eg}}^{r}) = f_{\mathrm{subsub}}^{k,r}(\cdot)
\tag{8}
$$

Therefore, all these candidate sub-sub-models are denoted as $\{f_{\mathrm{subsub}}^{j}(\cdot)\}_{j=1}^{J}$, and the prediction outputs are $\{\hat{\mathbf{y}}_{\mathrm{subsub}}^{j}\}_{j=1}^{J}$.

3.2 Candidate SEN-Sub-Model Module

The prediction outputs of these candidate sub-sub-models can be rewritten as

$$
\{\hat{\mathbf{y}}_{\mathrm{subsub}}^{j}\}_{j=1}^{J} =
\left\{
\begin{array}{l}
y_{\mathrm{subsub}}^{1,1}, \cdots, y_{\mathrm{subsub}}^{1,r}, \cdots, y_{\mathrm{subsub}}^{1,R}, \\
\vdots \\
y_{\mathrm{subsub}}^{k,1}, \cdots, y_{\mathrm{subsub}}^{k,r}, \cdots, y_{\mathrm{subsub}}^{k,R}, \\
\vdots \\
y_{\mathrm{subsub}}^{K,1}, \cdots, y_{\mathrm{subsub}}^{K,r}, \cdots, y_{\mathrm{subsub}}^{K,R}
\end{array}
\right\}
\tag{9}
$$

where $\hat{\mathbf{y}}_{\mathrm{subsub}}^{1} = f_{\mathrm{subsub}}^{1,1}(\cdot)$, $\hat{\mathbf{y}}_{\mathrm{subsub}}^{j} = f_{\mathrm{subsub}}^{k,r}(\cdot)$, $\hat{\mathbf{y}}_{\mathrm{subsub}}^{J} = f_{\mathrm{subsub}}^{K,R}(\cdot)$.

In (9), the $k$throw contains all the sub-sub-models with the same K_{er}^{k} and different $\{R_{\mathrm{eg}}^{r}\}_{r=1}^{R}$. Thus, $\{\hat{\mathbf{y}}_{\mathrm{subsub}}^{j}\}_{j=1}^{J}$ can be rewritten as

$$
\{\hat{\mathbf{y}}_{\mathrm{subsub}}^{j}\}_{j=1}^{J} =
\left\{
\begin{array}{l}
\{f_{\mathrm{subsub}}^{1,r}(\cdot)\}_{r=1}^{R}, \\
\vdots \\
\{f_{\mathrm{subsub}}^{k,r}(\cdot)\}_{r=1}^{R}, \\
\vdots \\
\{f_{\mathrm{subsub}}^{K,r}(\cdot)\}_{r=1}^{R}
\end{array}
\right\}
\tag{10}
$$

The candidate SEN-sub-models are built for each row of (10) by selecting and combining the candidate sub-sub-models based on different $\{R_{\mathrm{eg}}^{r}\}_{r=1}^{R}$. For example, the kth row $\{f_{\mathrm{subsub}}^{k,r}(\cdot)\}_{r=1}^{R}$ can construct the kth SEN-sub-model $f_{\mathrm{SENsub}}^{k}(\cdot)$ through the same K_{er}^{k} and $\{R_{\mathrm{eg}}^{r}\}_{r=1}^{R}$. So the prediction output $(\hat{y}_{\mathrm{SENsub}}^{k})^{n}$ of the nth modeling sample can be calculated by

$$(\hat{y}_{\text{SENsub}}^{k})^{n} = f_{\text{SENsub}}^{k}(\cdot) = \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} \cdot (\hat{y}_{r_k^{\text{sel}}}^{n})^{\text{sel}} = \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} \cdot f_{\text{subsub}}^{k,r_k^{\text{sel}}}(\mathbf{x}^n, K_{\text{er}}^k, R_{\text{eg}}^{r_k^{\text{sel}}}) \tag{11}$$

where $f_{\text{subsub}}^{k,r_k^{\text{sel}}}(\cdot)$, $R_{\text{eg}}^{r_k^{\text{sel}}}$, and $(\hat{y}_{r_k^{\text{sel}}}^{l})^{\text{sel}}$ are the selected r_k^{sel}th sub-sub-model, its RP, and the prediction output of the nth modeling sample, respectively; R_k^{sel} indicates the ensemble size, and $w_{r_k^{\text{sel}}}^{\text{sel}}$ is the weighted coefficient of the selected sub-sub-models, which is calculated by

$$w_{r_k^{\text{sel}}}^{\text{sel}} = \frac{1}{\sigma_{r_k^{\text{sel}}}^2 \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} \frac{1}{\sigma_{r_k^{\text{sel}}}^2}} \tag{12}$$

where $\sigma_{r_k^{\text{sel}}}$ is the standard variance of the prediction outputs $\{(\hat{y}_{r_k^{\text{sel}}}^{n})^{\text{sel}}\}_{n=1}^N$ form the selected r_k^{sel}th sub-sub-model. As the ensemble size R_k^{sel} and the weighted algorithm are pre-set and determined, the ensemble sub-sub-models for constructing the kth SEN-sub-model $f_{\text{SENsub}}^{k}(\cdot)$ can be obtained by solving the following optimization problem:

$$(\hat{y}_{\text{SENsub}}^{k})^{n} = \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} \cdot (\hat{y}_{r_k^{\text{sel}}}^{n})^{\text{sel}} = \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} \cdot f_{\text{subsub}}^{k,r_k^{\text{sel}}}(\mathbf{z}^n, K_{\text{er}}^k, R_{\text{eg}}^{r_k^{\text{sel}}})$$

$$s.t. \begin{cases} \min\left\{ \left\{ RMSE\left((\hat{y}_{\text{SENsub}}^{k,})_{R_k^{\text{sel}}} \right) \right\}_{R_k^{\text{sel}}=2}^{R} \right\} \\[2mm] RMSE\left((\hat{y}_{\text{SENsub}}^{k,})_{R_k^{\text{sel}}} \right) = \sqrt{\frac{1}{N} \sum_{n=1}^{N} \left(y^n - \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} \cdot f_{\text{subsub}}^{k,r_k^{\text{sel}}}(\mathbf{z}^n, K_{\text{er}}^k, R_{\text{eg}}^{r_k^{\text{sel}}}) \right)^2} \\[2mm] 2 \le R_k^{\text{sel}} \le R, \\[2mm] \sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} = 1, 0 < w_{r_k^{\text{sel}}}^{\text{sel}} < 1 \\[2mm] \{f_{\text{subsub}}^{k,r_k^{\text{sel}}}\}_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} \in \{f_{\text{subsub}}^{k,r}(\cdot)\}_{r=1}^{R} \\[2mm] \{R_{\text{eg}}^{r_k^{\text{sel}}}\}_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} \in \{R_{\text{eg}}^{r}\}_{r=1}^{R} \end{cases} \tag{13}$$

where the root mean square error (RMSE) is used to evaluate the generalization performance of the SEN-sub-model with ensemble size R_k^{sel}.

In order to solve the above problem, the optimized ensemble sub-sub-models and their weighted coefficients are obtained through repeating the BBSEN optimization algorithm $R - 2$ times [11], and this process is expressed by

$$\left\{ \{f_{\text{subsub}}^{k,r}(\mathbf{x}; K_{\text{er}}^k, R_{\text{eg}}^{r})\}_{r=1}^{R}, \{\mathbf{y}_n\}_{n=1}^{N} \right\} = f_{\text{BBSEN}}\left(\{f_{\text{subsub}}^{k,r_k^{\text{sel}}}(\mathbf{x}, K_{\text{er}}^k, R_{\text{eg}}^{r_k^{\text{sel}}})\}_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}}, \{w_{r_k^{\text{sel}}}^{\text{sel}}\}_{r=1}^{R_k^{\text{sel}}} \right) \tag{14}$$

where $\{f_{\text{subsub}}^{k,r}(\cdot)\}_{k=1}^{K_k^{\text{sel}}}$ and $\{w_{\text{subsub}}^{k,r}\}_{k=1}^{K_k^{\text{sel}}}$ are the selected ensemble sub-sub-models and their weighted coefficients, respectively; and $R_k^{\text{sel}}$is the number of the selected sub-sub-models.

For simplification, all the RPs of the kth SEN-sub-model are denoted as $(R_{\text{eg}}^k)_{\text{SENsub}} = \{R_{\text{eg}}^{r_k^{\text{sel}}}\}_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}}$. After repeating the above modeling procedure K times, all the SEN-sub-models with KPs $\{K_{\text{er}}^k\}_{k=1}^{K}$ are obtained.

And then these candidate SEN-sub-models and their prediction outputs can be denoted as $\{f_{\text{SENsub}}^k(\cdot)\}_{k=1}^K$ and $\{\hat{\mathbf{y}}_{\text{SENsub}}^k\}_{k=1}^K$, respectively. Therefore, different RPs are adaptively selected in the construction process of these SEN-sub-models.

3.3 SEN-Model Module

Based on the above sub-sections, the SEN-sub-models with the same KPs and different RPs are obtained. Then, Equation (10) can be transformed into the following form.

$$
\{\hat{\mathbf{y}}_{\text{subsub}}^j\}_{j=1}^J =
\left\{
\begin{array}{l}
\{f_{\text{subsub}}^{1,r}(\cdot)\}_{r=1}^R, \\
\vdots \\
\{f_{\text{subsub}}^{k,r}(\cdot)\}_{r=1}^R, \\
\vdots \\
\{f_{\text{subsub}}^{K,r}(\cdot)\}_{r=1}^R
\end{array}
\right\}
\Rightarrow\Rightarrow
\left\{
\begin{array}{l}
\{f_{\text{subsub}}^{1,r_1^{\text{sel}}}(\cdot)\}_{r_1^{\text{sel}}=1}^{R_1^{\text{sel}}}, \\
\vdots \\
\{f_{\text{subsub}}^{k,r_k^{\text{sel}}}(\cdot)\}_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}}, \\
\vdots \\
\{f_{\text{subsub}}^{K,r_K^{\text{sel}}}(\cdot)\}_{r_K^{\text{sel}}=1}^{R_K^{\text{sel}}}
\end{array}
\right\}
=
\left\{
\begin{array}{l}
f_{\text{SENsub}}^1(\cdot), \\
\vdots \\
f_{\text{SENsub}}^k(\cdot), \\
\vdots \\
f_{\text{SENsub}}^K(\cdot),
\end{array}
\right\}
\tag{15}
$$

$$
=\{f_{\text{SENsub}}^k(\cdot)\}_{k=1}^K = \{\hat{\mathbf{y}}_{\text{SENsub}}^k\}_{k=1}^K
$$

In this context, BBSEN is further employed to obtain the prediction output $\hat{y}_{\text{SEN}}^{k^{\text{sel}}}$ of SEN-model $f_{\text{SENsub}}^{k^{\text{sel}}}(\cdot)$ by optimally selecting and combining SEN-sub-models based on KPs $\{K_{\text{er}}^k\}_{k=1}^K$. Hence, the optimization description is presented as

$$
\hat{y}_{\text{SEN}}^n = \sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} w_{k^{\text{sel}}}^{\text{SENsel}} \cdot (\hat{y}_{\text{SEN}}^{k^{\text{sel}}})^n = w_{k^{\text{sel}}}^{\text{SENsel}} \cdot f_{\text{SENsub}}^{k^{\text{sel}}}(\cdot)
$$

$$
.t.
\begin{cases}
RMSE\left(\hat{\mathbf{y}}_{\text{SEN}}^{K^{\text{sel}}}\right) = \sqrt{\dfrac{1}{N}\sum_{n=1}^N \left(y^n - \sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} w_{k^{\text{sel}}}^{\text{SENsel}} \cdot (\hat{y}_{\text{SEN}}^{k^{\text{sel}}})^n\right)^2} \\
\qquad = \sqrt{\dfrac{1}{N}\sum_{n=1}^N \left(y^n - \sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} w_{k^{\text{sel}}}^{\text{SENsel}} \cdot f_{\text{SENsub}}^{k^{\text{sel}}}\left(\mathbf{z}^n, K_{\text{er}}^{k^{\text{sel}}}, (R_{\text{eg}}^{k^{\text{sel}}})_{\text{SENsub}}\right)\right)^2} \\
2 \le K^{\text{sel}} \le K, \\
\sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} w_{k^{\text{sel}}}^{\text{SENsel}} = 1, \quad 0 < w_{k^{\text{sel}}}^{\text{SENsel}} < 1 \\
\{f_{\text{SENsub}}^{k^{\text{sel}}}\}_{k^{\text{sel}}=1}^{K^{\text{sel}}} \in \{f_{\text{SENsub}}^k(\cdot)\}_{k=1}^K \\
\{K_{\text{eg}}^{k^{\text{sel}}}\}_{k^{\text{sel}}=1}^{K^{\text{sel}}} \in \{K_{\text{eg}}^r\}_{r=1}^K
\end{cases}
\tag{16}
$$

where $(R_{\text{eg}}^{k^{\text{sel}}})_{\text{SENsub}}$ is the RP set for constructing the k^{sel}th SEN-sub model with size $R_{\text{SENsub}}^{k^{\text{sel}}}$; K^{sel} indicates the number of selected ensemble SEN-sub-models (ensemble size) for building the SEN-model; $f_{\text{SENsub}}^{k^{\text{sel}}}(\cdot)$ and $K_{\text{eg}}^{k^{\text{sel}}}$ represent the k^{sel}thensemble SEN-sub-model and its KP, respectively; $\hat{\mathbf{y}}_{\text{SEN}}^{K^{\text{sel}}}$ denotes the prediction outputs of the nth samples of the SEN-model with ensemble size K^{sel}; and $w_{k^{\text{sel}}}^{\text{SENsel}}$ is the weighted coefficient of the k^{sel}th ensemble SEN-sub-model, which is calculated by

$$w_{k^{\text{sel}}}^{\text{SENsel}} = \frac{1}{\sigma_{k^{\text{sel}}}^2 \sum\limits_{k^{\text{sel}}=1}^{K^{\text{sel}}} \frac{1}{\sigma_{k^{\text{sel}}}^2}} \tag{17}$$

where $\sigma_{k^{\text{sel}}}$ is the standard variance of the prediction outputs $\{(\hat{y}_{\text{SEN}}^{k^{\text{sel}}})^n\}_{n=1}^N$ based on the selected k^{sel}th ensemble SEN-sub model. Consequentially, the above modeling process can be represented by

$$\left.\begin{array}{c} \{f_{\text{SENsub}}^k(\mathbf{x}, K_{\text{er}}^k, (R_{\text{eg}}^k)_{\text{SENsub}})\}_{k=1}^K \\ \{\mathbf{y}_n\}_{n=1}^N \end{array}\right\} \xrightarrow{\text{BBSEN}} \left\{\begin{array}{c} \{f_{\text{SENsub}}^{k^{\text{sel}}}(\mathbf{x}, K_{\text{er}}^{k^{\text{sel}}}, (R_{\text{eg}}^{k^{\text{sel}}})_{\text{SENsub}})\}_{k^{\text{sel}}=1}^{K^{\text{sel}}} \\ \{w_{k^{\text{sel}}}^{\text{SENsel}}\}_{k^{\text{sel}}=1}^{K^{\text{sel}}} \end{array}\right. \tag{18}$$

where $\{f_{\text{SENsub}}^{k^{\text{sel}}}(\cdot)\}_{k^{\text{sel}}=1}^{K^{\text{sel}}}$ and $\{w_{\text{SENsub}}^{k^{\text{sel}}}\}_{k^{\text{sel}}=1}^{K_k^{\text{sel}}}$ are the selected ensemble SEN-sub-models and their weighted coefficients, respectively; and K_k^{sel} is the number of selected SEN-sub-models.

Accordingly, the SEN model $f_{\text{SEN}}^{K_k^{\text{sel}}}(\cdot)$ with ensemble size K_k^{sel} can be denoted as

$$\hat{\mathbf{y}}_{\text{SEN}}^{K^{\text{sel}}} = \sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} w_{k^{\text{sel}}}^{\text{SENsel}} \cdot f_{\text{SENsub}}^{k^{\text{sel}}}(\cdot) = \sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} w_{k^{\text{sel}}}^{\text{SENsel}} \cdot \left(\sum_{r_k^{\text{sel}}=1}^{R_k^{\text{sel}}} w_{r_k^{\text{sel}}}^{\text{sel}} \cdot f_{\text{subsub}}^{k, r_k^{\text{sel}}}(\mathbf{z}, K_{\text{er}}^{k^{\text{sel}}}, R_{\text{eg}}^{r_k^{\text{sel}}})\right)_{k^{\text{sel}}} \tag{19}$$

In Eq. (19), the SEN-model $f_{\text{SEN}}^{K^{\text{sel}}}(\cdot)$ selects both the kernel and RPs adaptively from the candidates. Moreover, the ensemble size of SEN-sub-model is also decided adaptively in the SEN modeling process.

3.4 Decision Selection Module

Supposed that $K_k^{\text{sel}} \in [2, K)$ is hold, and then $(K - 2)$ SEN models can be constructed. However, the complexity of the SEN model significantly increases with the ensemble size. In order to solve this problem, the model complexity in this paper is measured by the number of single LSSVM models. Thus, the complexity of the best sub-sub-model, the kth SEN-sub-model $f_{\text{SENsub}}^k(\cdot)$, and the SEN model $f_{\text{SEN}}^{K^{\text{sel}}}(\cdot)$ are 1, R_k^{sel}, and $\sum_{k^{\text{sel}}=1}^{K^{\text{sel}}} R_{\text{SENsub}}^{k^{\text{sel}}}$, respectively. To make a tradeoff between prediction accuracy and model complexity, the metric index for selecting the suitable soft sensor model is defined by

$$\zeta_{\text{SEN}}^{K^{\text{sel}}} = \lambda \frac{RMSE(f_{\text{SEN}}^{K^{\text{sel}}}(\cdot))}{\sum\limits_{K^{\text{sel}}=2}^{K-1} RMSE(f_{\text{SEN}}^{K^{\text{sel}}}(\cdot))} + (1-\lambda) \frac{\sum\limits_{k^{\text{sel}}=1}^{K^{\text{sel}}} R_{\text{SENsub}}^{k^{\text{sel}}}}{\sum\limits_{K^{\text{sel}}=2}^{K-1} \sum\limits_{k^{\text{sel}}=1}^{K^{\text{sel}}} R_{\text{SENsub}}^{k^{\text{sel}}}} \tag{20}$$

where λ is the coefficient between 0 and 1.

According to the above metric index, the SEN-model with the lowest RMSE is selected as the final DXN soft measuring model. Thus, the final prediction output is obtained by

$$\hat{y} = f_{\text{sel}}(\hat{\mathbf{y}}_{\text{SEN}}^2, \cdots, \hat{\mathbf{y}}_{\text{SEN}}^{K^{\text{sel}}}, \cdots, \hat{\mathbf{y}}_{\text{SEN}}^{K-1})$$
$$s.t. \quad \min\{\zeta_{\text{SEN}}^2, \cdots, \zeta_{\text{SEN}}^{K^{\text{sel}}}, \cdots, \zeta_{\text{SEN}}^{K-1}\} \tag{21}$$

where $f_{\text{sel}}(\cdot)$ represents the final AMLSEN-LSSVM model. In this way, the learning parameters and ensemble sizes of different SEN layers are selected adaptively.

4 Application Study

In this section, two UCI benchmarks and a practical DXN datasets are exploited to validate the proposed methods. Primarily, each dataset is divided into two parts: training data and testing data. Then radius basis function (RBF) is used for the kernel type of the LSSVM. Finally, the candidate kernel and RP datasets are selected as Can_1{0.1, 1, 100, 1000, 2000, 4000, 6000, 8000, 10000, 20000, 40000, 60000, 80000, 160000} and Can_2{0.1, 1, 6, 12, 25, 50, 100, 200, 400, 800, 1600, 3200, 6400, 12800, 25600, 51200, 102400}, respectively. Seen from the samples, Can_1 and Can_2 illustrate that the candidate learning parameters have a wide range. The order number and their values are shown in Tab. 1.

Table 1: Order number and learning parameters value

Order	1	2	3	4	5	6
K_{er}^k	0.1	1	100	1000	2000	4000
R_{eg}^r	0.1	1	6	12	25	50
Order	**7**	**8**	**9**	**10**	**11**	**12**
K_{er}^k	6000	8000	10000	20000	40000	60000
R_{eg}^r	100	200	400	800	1600	3200
Order	**13**	**14**	**15**	**16**	**17**	**–**
K_{er}^k	80000	160000	–	–	–	–
R_{eg}^r	6400	12800	25600	51200	102400	–

4.1 Data Description

In order to present preliminary results, two UCI benchmark datasets, Boston housing and concrete compressive strength, are used to validate the proposed method, which are listed as follows.

For Boston housing data, the inputs include: (1) the *per capita* crime rate of the town (CRIM);(2) the proportion of residential land zoned over 25,000 sq. ft. (ZN); (3) the proportion of non-retail business acres per town (INDUS); (4) the Charles River dummy variables (CHAS); (5) the nitric oxide concentrations (NOX); (6) the average number of rooms per dwelling (RM); (7) the proportion of owner-occupied units built before 1940 (AGE); (8) the weighted distances to five employment centers of Boston (DIS); (9) the index of radial highway accessibility (RAD); (10) the full property tax rate per $10,000 (TAX); (11) the pupil–teacher ratio of the town (B); (12) the lower status of the population (LSTAT); and (13) the median value of owner-occupied homes per $1000 (MEDV). The output is the housing values of the suburbs of Boston with data size 506.

For concrete compressive strength data, the inputs include: (1) cement; (2) blast furnace slag; (3) fly ash; (4) water; (5) superplasticizer; (6) coarse aggregate; (7) fine aggregate of the various ingredients of concrete placement per cubic meter; and (8) conserved days. The output is concrete compressive strength with data size 1030.

4.2 Modeling Results

According to the candidate learning parameters, $14 \times 17 = 238$ sub-sub-models based on the LSSVM are constructed. Correspondingly, the BBSEN method is employed to build 14 SEN-sub-models. The statistical results are shown in Tab. 2.

Table 2: Statistical results of different SEN-sub-models for benchmark datasets

Order number	Kernel parameter (KP)	Models	Boston housing data			Concrete compressive strength data		
			RMSE (Testing)	Regularization parameter (RP, Order number)	Ensemble size	RMSE (Testing)	Regularization parameter (RP, Order number)	Ensemble size
1	0.1	EnAll-sub-sub	8.9735	all	17	16.1059	all	17
		SEN-sub	8.5902	16–17	2	15.0883	15–17	2
		Best-sub-sub	8.5902	17	1	15.0883	17	1
2	1	EnAll-sub-sub	6.9940	all	17	12.8921	all	17
		SEN-sub	5.1031	16–17	2	12.4234	2–17	6
		Best-sub-sub	5.1031	17	1	11.8064	6	1
3	100	EnAll-sub-sub	**3.4288**	all	17	**9.8368**	all	17
		SEN-sub	**3.1146**	4–17	14	**7.3687**	4–17	4
		Best-sub-sub	**3.1953**	8	1	**7.7284**	10	1
4	1000	EnAll-sub-sub	6.5492	all	17	15.9703	all	17
		SEN-sub	3.2483	10–17	8	8.1098	11–17	7
		Best-sub-sub	3.2637	13	1	8.1362	13	1
5	2000	EnAll-sub-sub	7.8296	all	17	16.4346	all	17
		SEN-sub	3.2653	13–17	5	8.1475	14–17	4
		Best-sub-sub	3.2681	15	1	8.1459	15	1
6	4000	EnAll-sub-sub	8.5535	all	17	16.6256	all	17
		SEN-sub	3.2763	16–17	2	8.1642	16–17	2
		Best-sub-sub	3.2705	17	1	8.1511	17	1
7	6000	EnAll-sub-sub	8.7715	all	17	16.6813	all	17
		SEN-sub	3.3318	16–17	2	8.3123	16–17	2
		Best-sub-sub	3.3022	17	1	8.2182	17	1
8	8000	EnAll-sub-sub	8.8700	all	17	16.7074	all	17
		SEN-sub	3.4023	16–17	2	8.5411	16–17	2
		Best-sub-sub	3.3586	17	1	8.3811	17	1
9	10000	EnAll-sub-sub	8.9245	all	17	16.7225	all	17
		SEN-sub	3.4733	16–17	2	8.7862	16–17	2
		Best-sub-sub	3.4187	17	1	8.5831	17	1
10	20000	EnAll-sub-sub	9.0216	all	17	16.7514	all	17
		SEN-sub	3.7978	16–17	2	9.7130	16–17	2
		Best-sub-sub	3.7030	17	1	9.4900	17	1
11	40000	EnAll-sub-sub	9.0632	all	17	16.7652	all	17
		SEN-sub	4.2073	16–17	2	10.4129	16–17	2
		Best-sub-sub	4.1183	17	1	10.2943	17	1
12	60000	EnAll-sub-sub	9.0759	all	17	16.7697	all	17
		SEN-sub	4.3858	16–17	2	10.6267	16–17	2
		Best-sub-sub	4.3268	17	1	10.5610	17	1

Table 2 (continued).

Order number	Kernel parameter (KP)	Models	Boston housing data			Concrete compressive strength data		
			RMSE (Testing)	Regularization parameter (RP, Order number)	Ensemble size	RMSE (Testing)	Regularization parameter (RP, Order number)	Ensemble size
13	80000	EnAll-sub-sub	9.0821	all	17	16.7719	all	17
		SEN-sub	4.4689	16–17	2	10.7145	16–17	2
		Best-sub-sub	4.4297	17	1	10.6731	17	1
14	16000	EnAll-sub-sub	9.0910	all	17	16.7752	all	17
		SEN-sub	4.5524	16–17	2	10.8124	16–17	2
		Best-sub-sub	4.5645	17	1	10.7968	17	1

Tab. 2 illustrates that: (1) the best SEN-sub-model of Boston housing dataset has KP 100 and RPs {25, 50, 100, 200, 400, 800, 1600, 3200, 6400, 12800, 25600, 51200, 102400} with RMSE 3.1146 and an ensemble size 14. The best sub-sub-model picks up KP 100 and RP 200 with RMSE 3.1953, and the best EnAll-sub-sub-model possesses the same KP with RMSE 3.4288;(2) the best SEN-sub-model of concrete compressive strength data has the KP 100 and RPs {25, 50, 100, 200, 400, 800, 1600, 3200, 6400, 12800, 25600, 51200, 102400} with RMSE 7.368 and an ensemble size 14. The best sub-sub-model picks up KP 100 and RP 800 with RMSE 7.7284, and the best EnAll-sub-sub-model possesses the same KP with RMSE 9.8368;(3) different KPs have different effects on the prediction performance of the SEN-sub-models, so that parts of the best sub-sub-models have lower RMSEs than those of the SEN-sub-models. Thus, setting a suitable RP is very important to demonstrate the relationship among RMSEs, ensemble sizes, and different types of models with the single KP, which are shown in Figs. 2 and 3.
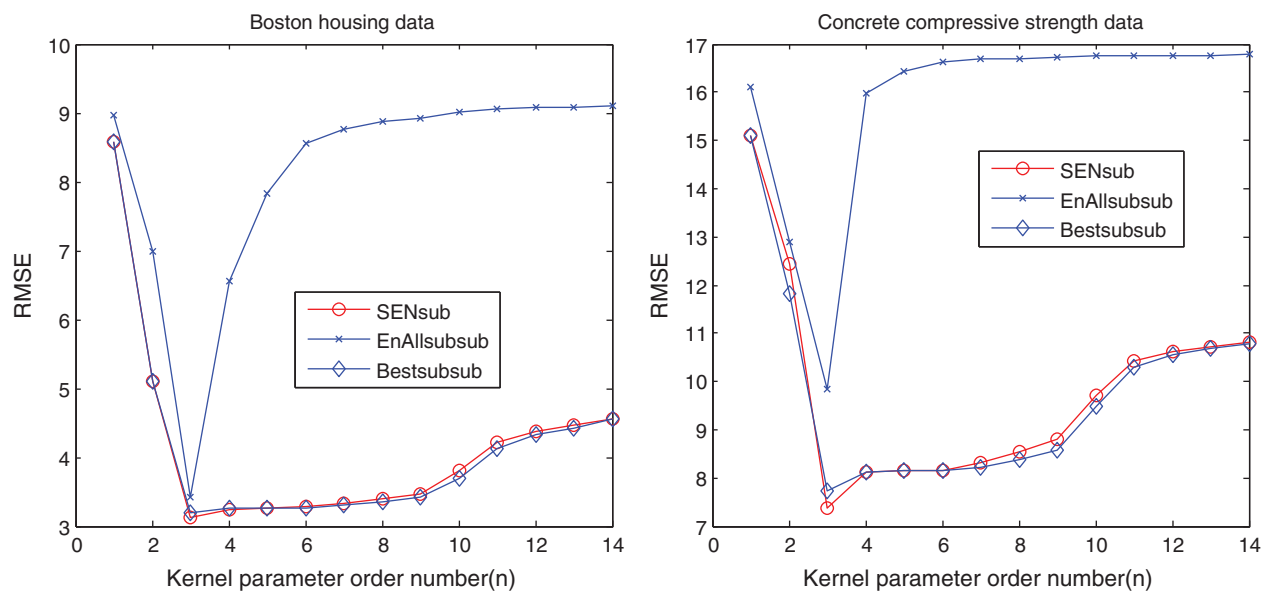

Figure 2: Relationship between RMSEs and different types of models based on the single KP for the UCI benchmark datasets
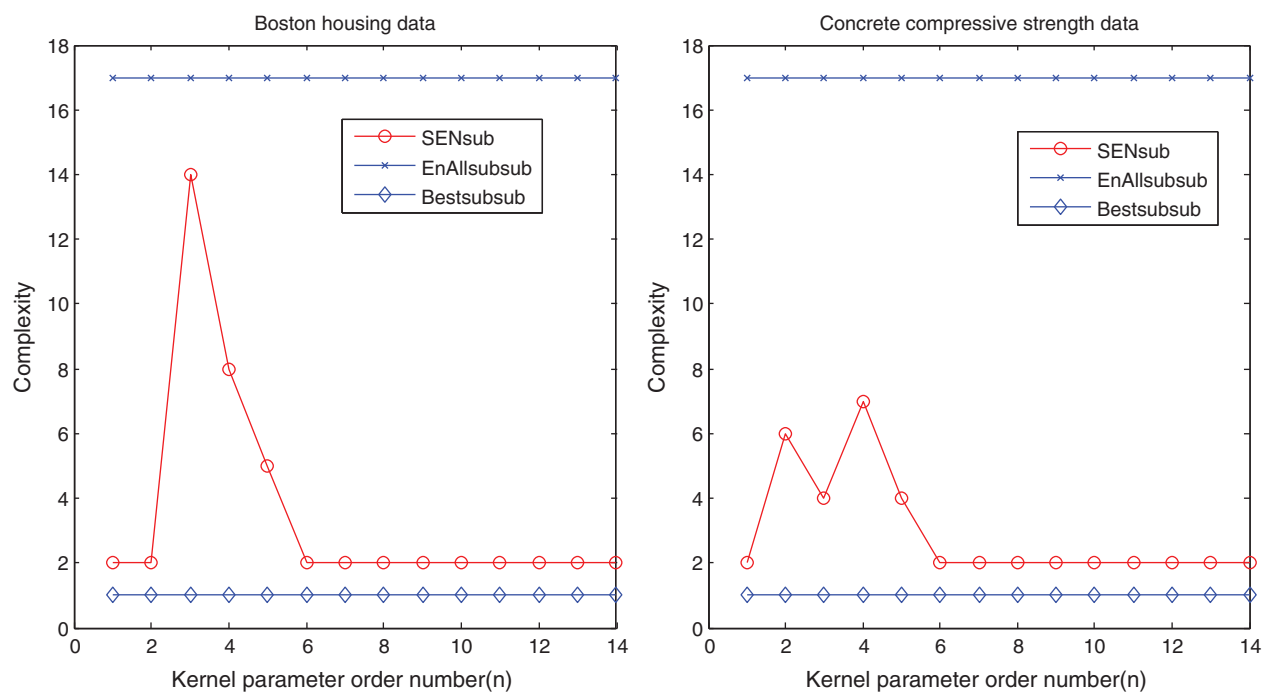
Figure 3: Relationships between ensemble sizes and different models based on the single KP for the UCI benchmark datasets

Figs. 2 and 3 show that suitable learning parameters are crucial. From this perspective, the ensemble size with the best prediction performance of the Housing and Concrete data are 14 and 6 in terms of the same KP 100, respectively. However, the ensemble size of the SEN-sub-model does not increase with the KPs after 4000.

For the 14 SEN-sub-models, the BBSEN method is used again to obtain the SEN-model with a different ensemble size, and the detailed statistical results of the SEN-models are illustrated in Tab. 3.

Table 3a: Statistical results of all SEN-models for the Boston housing dataset

Ensemble Size	SEN-sub-model (KP) number	RMSE	Ensemble size	Metric index $\lambda = 0.5$
2	2 1	6.5189	4	0.07497
3	3 2 1	4.7546	18	0.07358
4	4 3 2 1	4.0554	26	0.07610
5	5 4 3 2 1	3.7270	31	0.07884
6	6 5 4 3 2 1	3.5499	33	0.07944
7	7 6 5 4 3 2 1	3.4546	35	0.08093
8	8 7 6 5 4 3 2 1	3.4036	37	0.08289
9	**9 8 7 6 5 4 3 2 1**	**3.3781**	**39**	**0.08512**
10	10 9 8 7 6 5 4 3 2 1	3.3827	41	0.08767
11	11 10 9 8 7 6 5 4 3 2 1	3.4121	43	0.09050
12	12 11 10 9 8 7 6 5 4 3 2 1	3.4538	45	0.09345
13	13 12 11 10 9 8 7 6 5 4 3 2 1	3.5011	47	0.09646

Table 3b: Statistical results of all SEN-models for the concrete compressive strength dataset

Ensemble Size	SEN-sub-model (KP) number	RMSE	Ensemble size	Metric index $\lambda = 0.5$
2	2 1	12.8102	8	0.07228
3	3 2 1	9.9116	12	0.06506
4	4 3 2 1	8.9488	19	0.07158
5	5 4 3 2 1	8.5528	23	0.07602
6	6 5 4 3 2 1	8.3599	25	0.07827
7	7 6 5 4 3 2 1	8.2641	27	0.08097
8	8 7 6 5 4 3 2 1	8.2247	29	0.08393
9	**9 8 7 6 5 4 3 2 1**	**8.2221**	**31**	**0.08706**
10	10 9 8 7 6 5 4 3 2 1	8.2942	33	0.09054
11	11 10 9 8 7 6 5 4 3 2 1	8.4228	35	0.09429
12	12 11 10 9 8 7 6 5 4 3 2 1	8.5617	37	0.09808
13	13 12 11 10 9 8 7 6 5 4 3 2 1	8.6963	39	0.10185

Tab. 3 shows that:(1) in the Boston housing dataset, the SEN-model with an ensemble size 9, KPs {0.1, 1, 100, 1000, 2000, 4000, 6000, 8000, 10000}, has the best prediction performance (RMSE **3.3781**) among all the SEN-models, which is higher than that of the best SEN-sub-mode, but lower than that of the EnAll-SEN-sub-model;(2) in the concrete compressive strength dataset, the SEN-model with an ensemble size 9, KPs {0.1, 1, 100, 1000, 2000, 4000, 6000, 8000, 10000}, has the best prediction performance (RMSE **8.2221**) among all the SEN-models. However, it is larger than that of the best SEN-sub-mode, but lower than that of the EnAll-SEN-sub-model; (3) the best SEN-sub-models in the above datasets has the best prediction performance. Moreover, the prediction performance of different SEN-models is not further improved with the increase of the ensemble size. The relationship between RMSEs and the ensemble sizes of SEN models is shown in Figs. 4 and 5.
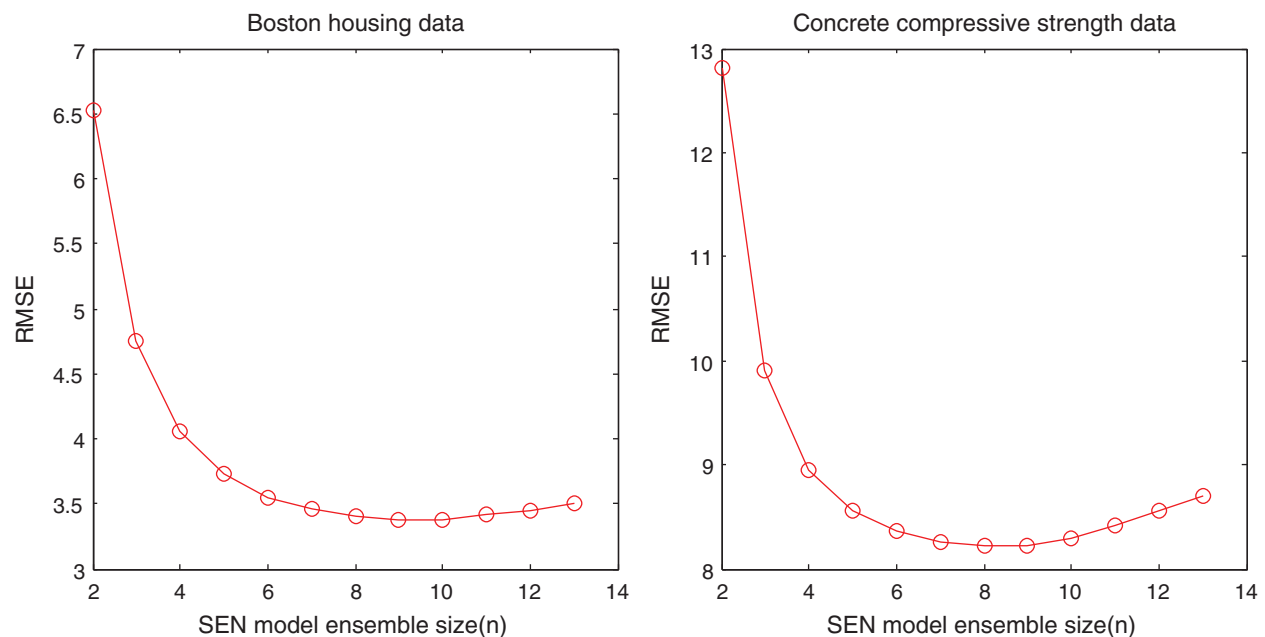
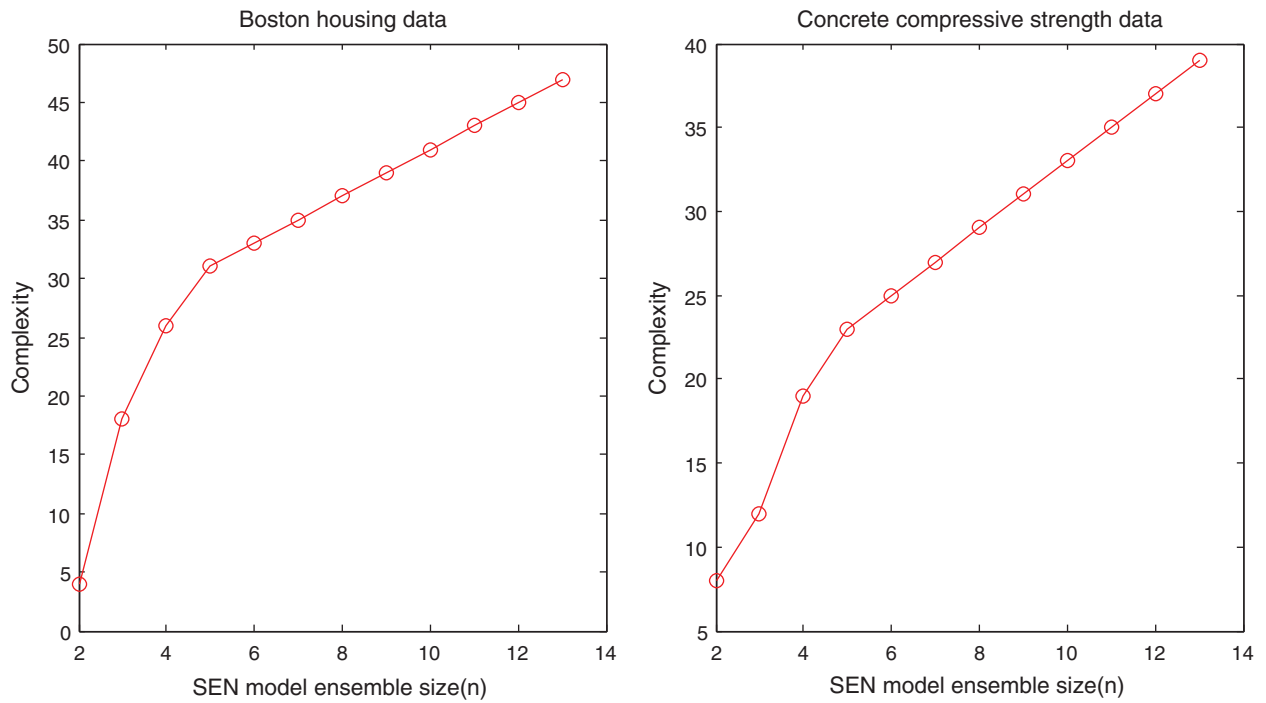**Figure 4:** RMSE of SEN model with different ensemble sizes for the UCI benchmark datasets

Figure 5: Ensemble size of SEN model with different ensemble sizes for the UCI benchmark datasets

The above results show that it is essential to make a tradeoff between prediction performance and model complexity. The proposed indices of different SEN-models with $\lambda = 0.5$ are shown in Tab. 3, and the metric index curves of the SEN-model with different λ values are shown in Fig. 6.
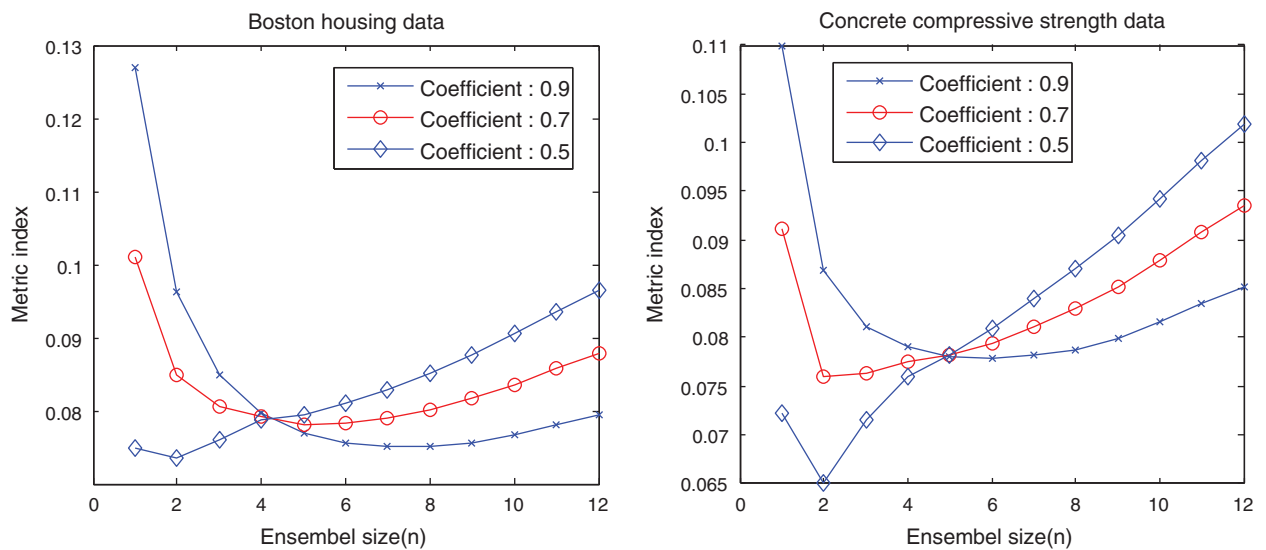

Figure 6: Metric index curves of the SEN-models with different λ values

Seen form Fig. 6, the optimal value of the metric index shifts to the left with increasing coefficient (λ) value. Thus, it can be determined by the practical requirement. The prediction curves of the best SEN-sub-model (SENsub) and the best SEN-model (SenSENsub) on the minimum prediction errors as well as the ensemble all SEN-sub-model (EnSENsub) are shown in Figs. 7 and 8, respectively.
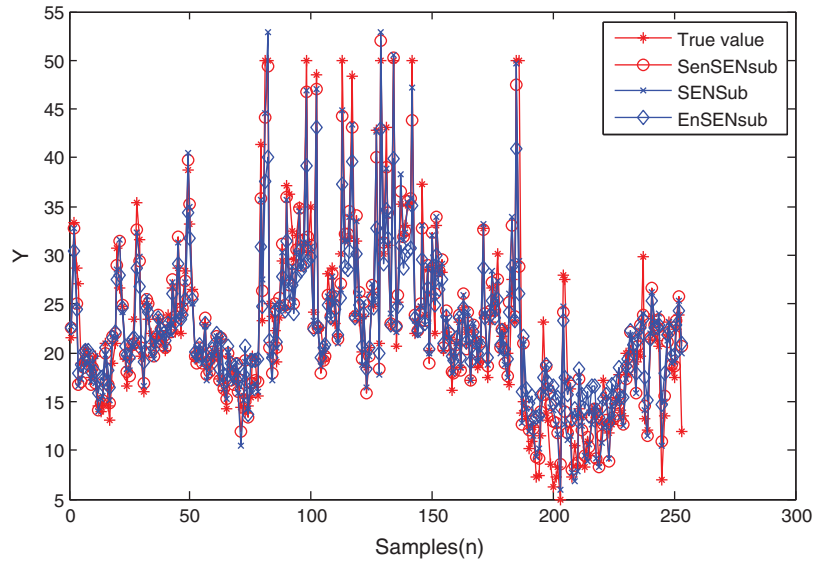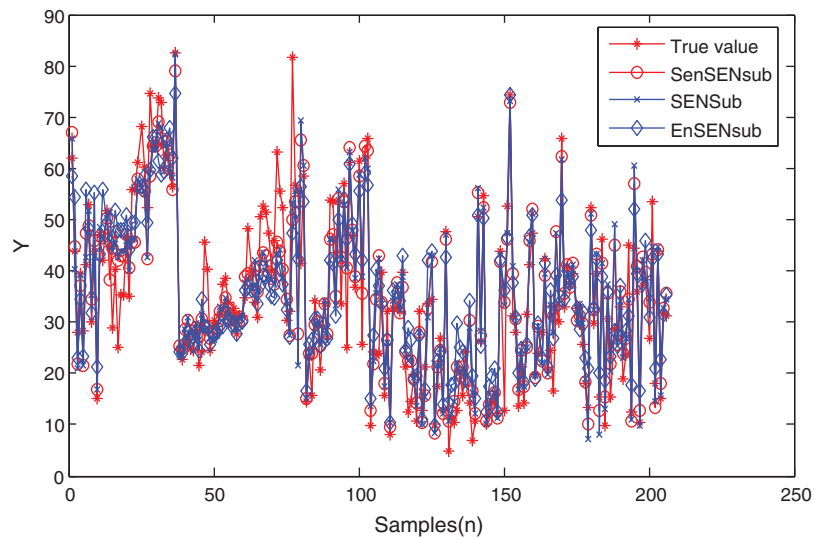

Figure 7: Prediction curves of the Boston housing data

Figure 8: Prediction curves of the concrete compressive strength data

The above results show that the proposed method can model the two UCI benchmark datasets effectively, which can make an adaptive selection of the learning parameters.

4.3 Comparative Results

The proposed method is compared with the PLS, KPLS, GASEN-BPNN, and GASEN-LSSVM approaches. Concretely, the number of latent variables (LVs) and kernel LVs (KLVs) are decided by the leave-one-out cross-validation, and the number of hidden nodes of the GASEN-BPNN is set to two times plus one of the original inputs' number. Thus, the structure of BPNN is 13-27-1 for the housing data, and 8-16-1 for concrete data. Meanwhile, the learning parameters of GASEN-LSSVM are used as ones of the best sub-sub-model in the proposed method. For GASEN-BPNN and GASEN-LSSVM, the modelling process is repeated 10 times to overcome their randomness.

Tab. 4 shows that the proposed method has the best prediction performance without any disturbance.

Table 4: Statistical results of comparative methods

	Boston housing data			Concrete compressive strength data			Denote
	RNSEs			RMSEs			
	Max	Mean	Min	Max	Mean	Min	
PLS	–	4.681	–	–	10.92	–	LVs = 4/ LVs = 7
KPLS	–	3.195	–	–	8.179	–	KLVs = 8/ KLVs = 8
GASEN-BPNN	11.2478	8.9307	5.0509	14.8580	12.0756	10.3971	(13-27-1)/(8-17-1)
GASEN-LSSVM	3.3480	3.2899	3.2393	10.1777	10.0149	9.7490	(100,200)/(100,800)
This paper		3.027			7.163		

Although the GASEN-based prediction results are disturbed within a certain range due to the random initialization of the input weights and the bias of BPNN and population GA, the PLS/KPLS methods can activate the extracted latent variables to construct a linear or nonlinear model with a single KP for stabilization.

4.4 Discussions

4.4.1 Kernel Parameter

KP determines the geometry of the feature space, which can be calculated by using the Fisher discrimination for classification problem. Actually, due to the deficiency of the range obtained for the regression problem, the candidate KP set in our proposed method serves as the key factor to obtain an effective soft sensor model. In addition, a wide range of KPs must be used for very small sample data with complex characteristics, such as DXN.

4.4.2 Regularization Parameter

RP is always decided by the cross validation or other optimization methods. In terms of results, a small number of modeling samples require more RPs, that is, the ensemble size increases with decrease of the number of samples. Moreover, a large KP requires a small number of RPs. As a result, the reasonable range should also be optimized in the future studies.

4.4.3 Ensemble Size

Selecting a suitable number of ensemble sub-models from candidates in the SEN-modeled process means that more ensemble sub-models leads to the more complex method, which implies sub-sub-models can measure the complexity of the proposed AMLSEN method. In practice, the ensemble size of the SEN-model in the proposed method is selected to make a trade-off between prediction accuracy and model complexity. Additionally, the ensemble size of the SEN-sub-model is implicitly determined by

using the BBSEN algorithm in terms of prediction performance. Thus, the proposed method has a flexible structure to demonstrate the conclusion that the ensemble size increases with the decrease of the number of training samples.

5 Conclusions

A new soft measuring method is proposed based on the AMLSEN-LSSVM algorithm, and many sub-sub-models based on the different candidate learning parameters are constructed by using the LSSVM. According to the same KP, these candidate sub-sub-models are selectively fused to obtain SEN-sub-models by using the BBSEN, which are selectively combined by using the BBSEN again for building SEN models with different ensemble sizes. Ultimately, the final soft measuring model is determined based on a newly defined metric index. The simulation results based on the benchmark datasets show the effectiveness of the proposed method, which demonstrate that not only the proposed method can be used for softly measuring other difficult-to-measure process parameters in different industrial processes, but also a more common adaptive multi-layer SEN modeling framework can be explored.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] W. Wang, T. Y. Chai and W. Yu, "Modelling component concentrations of sodium aluminate solution via hammerstein recurrent neural networks," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 4, pp. 971–982, 2012.

[2] J. Tang, T. Y. Chai, W. Yu and L. J. Zhao, "Modeling load parameters of ball mill in grinding process based on selective ensemble multisensor information," *IEEE Transactions on Automation Science and Engineering*, vol. 10, no. 3, pp. 726–740, 2013.

[3] M. Kano and K. Fujiwara, "Virtual sensing technology in process industries: trends & challenges revealed by recent industrial applications," *Journal of Chemical Engineering of Japan*, vol. 46, pp. 1–17, 2013.

[4] S. Bunsan, W. Y. Chen, H. W. Chen, Y. H. Chuang and N. Grisdanurak, "Modeling the dioxin emission of a municipal solid waste incinerator using neural networks," *Chemosphere*, vol. 92, no. 3, pp. 258–264, 2013.

[5] T. A. F. Gomes, R. B. C. Prudêncio, C. Soares, A. L. D. Rossi and A. Carvalho, "Combining meta-learning and search techniques to select parameters for support vector machines," *Neurocomputing*, vol. 75, no. 1, pp. 3–13, 2012.

[6] J. Tang, Z. Liu, J. Zhang, Z. W. Wu, T. Y. Chai *et al.,* "Kernel latent features adaptive extraction and selection method for multi-component non-stationary signal of industrial mechanical device," *Neurocomputing*, vol. 216, no. Suppl. 1–4, pp. 296–309, 2016.

[7] A. Bemani, A. Baghban, S. Shamshirband, A. Mosavi, P. Csiba *et al.,* "Applying ANN, ANFIS, and LSSVM models for estimation of acid solvent solubility in supercritical co2," *Computers, Materials & Continua*, vol. 63, no. 3, pp. 1175–1204, 2020.

[8] C. Soares, "A hybrid meta-learning architecture for multi-objective optimization of SVM parameters," *Neurocomputing*, vol. 143, no. 1–3, pp. 27–43, 2014.

[9] S. Yin and J. Yin, "Tuning kernel parameters for SVM based on expected square distance ratio," *Information Sciences*, vol. 370–371, no. 12, pp. 92–102, 2016.

[10] J. Sun, C. Zheng, X. Li and Y. Zhou, "Analysis of the distance between two classes for tuning svm hyperparameters," *Neural Netw. IEEE Trans*, vol. 21, no. 2, pp. 305–318, 2010.

[11] K. P. Wu and S. D. Wang, "Choosing the kernel parameters for support vector machines by the inter-cluster distance in the feature space," *Pattern Recognition*, vol. 42, no. 5, pp. 710–717, 2009.

[12] W. Wang, Z. Xu, W. Lu and X. Zhang, "Determination of the spread parameter in the Gaussian kernel for classification and regression," *Neurocomputing*, vol. 55, no. 3-4, pp. 643–663, 2003.

[13] J. Tang, T. Y. Chai, Q. M. Cong, B. C. Yuan, L. J. Zhao *et al.,* "Soft sensor approach for modeling mill load parameters based on EMD and selective ensemble learning algorithm," *Acta Automatica Sinica*, vol. 40, no. 9, pp. 1853–1866, 2014.

[14] Y. Wang, X. Liu, Y. Dou, Q. Lv and Y. Lu, "Multiple kernel learning with hybrid kernel alignment maximization," *Pattern Recognition*, vol. 70, no. 1, pp. 104–111, 2017.

[15] C. Qi, Z. Zhou, Y. Sun, H. Song, L. Hu *et al.,* "Feature selection and multiple kernel boosting framework based on PSO with mutation mechanism for hyperspectral classification," *Neurocomputing*, vol. 220, no. 10, pp. 181–190, 2017.

[16] U. Suleymanov, B. K. Kalejahi, E. Amrahov and R. Badirkhanli, "Text classification for azerbaijani language using machine learning," *Computer Systems Science and Engineering*, vol. 35, no. 6, pp. 467–475, 2020.

[17] Z. H. Zhou, J. Wu and W. Tang, "Ensembling neural networks: Many could be better than all," *Artificial Intelligence*, vol. 137, no. 1–2, pp. 239–263, 2002.

[18] J. Tang, J. Zhang, Z. Wu, Z. Liu, T. Chai *et al.,* "Modeling collinear data using double-layer GA-based selective ensemble kernel partial least squares algorithm," *Neurocomputing*, vol. 219, no. 4, pp. 248–262, 2017.

[19] J. Tang, T. Y. Chai, Q. M. Cong, L. J. Zhao, Z. Liu *et al.,* "Modeling mill load parameters based on selective fusion of multi-scale shell vibration frequency spectrum," *Control Theory & Application*, vol. 32, no. 12, pp. 1582, 2015.

[20] J. Tang, W. Yu, T. Y. Chai, Z. Liu and X. J. Zhou, "Selective ensemble modeling load parameters of ball mill based on multi-scale frequency spectral features and sphere criterion," *Mechanical Systems and Signal Processing*, vol. 66–67, pp. 485–504, 2016.

[21] Y. Lv, J. Liu and T. Yang, "A novel least squares support vector machine ensemble model for NOx, emission prediction of a coal-fired boiler," *Energy*, vol. 55, no. 1, pp. 319–329, 2013.

[22] L. Yu, "An evolutionary programming based asymmetric weighted least squares support vector machine ensemble learning methodology for software repository mining," *Information Sciences*, vol. 191, no. 8, pp. 31–46, 2012.