Applied Soft Computing Journal 101 (2021) 107060

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Adaptive ranking based ensemble learning of Gaussian process regression models for quality-related variable prediction in process industries

Yiqi Liu [a], Daoping Huang [a], Bin Liu [b,*], Qiang Feng [c], Baoping Cai [d]

[a] School of Automation Science & Engineering, South China University of Technology, Wushan Road, Guang Zhou 510640, China
[b] Department of Management Science, University of Strathclyde, Glasgow, UK
[c] School of Reliability and Systems Engineering, Beihang University, Beijing, China
[d] College of Mechanical and Electronic Engineering, China University of Petroleum, Qingdao, Shan-dong, China

ARTICLE INFO

Article history:
Received 21 May 2020
Received in revised form 22 October 2020
Accepted 20 December 2020
Available online 24 December 2020

Keywords:
Gaussian process regression
Ensemble learning
Soft sensors
Wastewater treatment

ABSTRACT

The proper monitoring of quality-related but hard-to-measure variables is currently one of the bottlenecks limiting the safe and efficient operations of industrial processes. This paper proposes a novel ensemble learning algorithm by coordinating global and local Gaussian process regression (GPR) models, and this algorithm is able to capture global and local process behaviours for accurate prediction and timely process monitoring. To further address the deterioration in predictions when using the off-line training and online testing strategy, this paper proposes an adaptive ranking strategy to perform ensemble learning for the sub-GPR models. In this adaptive strategy, we use the moving-window technique to rank and select several of the best sub-model predictions and then average them together to make the final predictions. Last but not least, the least absolute shrinkage and selection operator (Lasso) works together with factor analysis (FA) in a two-step variable selection method to remove under-correlated model input variables in the first stage and to compress over-correlated model input variables in the second stage. The proposed prediction model is validated in two real wastewater treatment plants (WWTPs) with stationary and nonstationary behaviours. The results show that the proposed methodology achieves better performance than other standard methods in the context of their predictions of quality-related variables.

1. Introduction

In process industries, quality-related but hard-to-measure variables are always derived by either under-sampled off-line analyses or over-expensive online measurement devices, both of which usually lead to improper monitoring, maintenance and optimization [1,2]. Since soft-sensors can replace hardware instruments or assist in off-line analyses, they have received significant attention from both the academic and industrial communities over the past few decades. Soft-sensors always work well because they are able to properly describe the inherent relationships between the input and response variables by building a predictive model, which can be used online for prediction [3,4].

Existing approaches, such as artificial neural networks (ANNs), fuzzy algorithms, and Bayesian networks [5–7], are widely used to construct soft-sensors [8,9], particularly for approaching nonlinear processes. The Gaussian process model is a Bayesian nonparametric framework for inference that is gaining popularity in control, maintenance and reinforcement learning applications [10–12]. Gaussian process models are inherently global models that fit a distribution over data and, in turn, learn data for regressions or classifications. In global learning, GPR models are usually approximated by matrix approximation and likelihood approximation methodologies. Fully independent conditional (FIC) [13] and partially independent conditional (PIC) [14] approximations are two of most commonly used likelihood approximation algorithms for GPR models with different assumptions. Unfortunately, FIC approximations always fail to fit data with local abrupt features or nonstationary features. To describe localized features properly, PIC approximations slightly relax the conditional independence assumption among model input variables. The main disadvantage of global learning is that appropriate parameters must be chosen well to describe the observed data globally. Conversely, local learning aims to refine useful local information from the observed data, believing that closer pairs of observations exhibit stronger correlations [14,15]. Therefore, only a small

* Corresponding author.
E-mail addresses: aulyq@scut.edu.cn (Y. Liu), audhuang@scut.edu.cn (D. Huang), b.liu@strath.ac.uk (B. Liu), fengqiang@buaa.edu.cn (Q. Feng), caibaoping@upc.edu.cn (B. Cai).

https://doi.org/10.1016/j.asoc.2020.107060
1568-4946/

number of neighbouring points are used in local learning. During the local learning process, the whole domain is decomposed into several subdomains, and the selected subdomain is used for model training first. Second, the trained model makes predictions by justifying which subdomain the test point belongs to. For example, Chiwoo Park et al. proposed a DDM model that aimed at dealing with nonstationary changes adaptively with cheap computations [16]. Another method of local learning is to construct multiple local prediction models and then derive a group of local predicted values based on the local models if the new input values are given. Then, the final prediction can be achieved by weighting the average of all the local predicted values with techniques such as the Bayesian committee machine [17], local probabilistic regression [18], and a mixture of Gaussian process experts [19,20]. Although local learning outperforms global learning in some fields, building models that are purely dependent upon local information results in the loss of the global view of the data, and thus the critical data structure cannot be captured; the presence of this critical structure may yield better performances than those of models without this structure. Conversely, global learning can summarize global knowledge about the data structure. Depending on the global knowledge, new data can be reproduced accurately and fed into a model for a thorough analysis. It is envisaged that both local learning and global learning can be complementary to each other when performing model construction. Ensemble learning provides an alternative way to address this issue. The main purpose of ensemble learning is to combine a set of learned submodels to improve the prediction performance for new instances [21], even though individual inducers are comparable with each other. Additionally, by performing ensemble learning, a complex, difficult learning problem can be decomposed into easier subproblems that can be solved efficiently.

When building a proper soft-sensor either by ensemble learning or via other approaches, the priority is to select proper variables or features. It has been shown that more features will result in a more complex machine learning model and worse model fitting. Therefore, variable selection is an essential step to improve the prediction performance of a soft sensor. Additionally, performing variable selection can provide insight into the essence of a system [22]. Variable selection can be achieved by experts' knowledge or data-driven techniques. Even though expert suggestions are appealing, they are always limited to specified areas and are difficult to transfer to other fields. Data-driven techniques are able to mitigate such an inconsistency. Many diverse techniques are used for variable selection, and these techniques fall into two categories: unsupervised and supervised methods [22]. Unsupervised methods aim to create new features by mapping the original features. The new features do not have any physical meaning and are difficult to interpret. Although the number of original features is indeed reduced, the number of hardware sensors required is not reduced. The most widely used techniques are principal component analysis (PCA) and factor analysis (FA) [23]. Supervised methods, on the other hand, are able to select the relatively best subset by evaluating the input and output data. VIP-PLS (variable importance in projection-partial least squares) is one of the earliest variable selection methods that measures the contributions of input variables in describing the corresponding output variables. The contribution of each variable is quantified by a VIP value, and a value that is less than a control limit indicates an unimportant variable [4,24]. Additionally, other variable selection methods, i.e., backward variable elimination (BVE) and the genetic algorithm (GA), have been used for selecting input variables for soft-sensor modelling [25]. The GA is a preferred method for selecting variables in soft sensor modelling [26]. Through this selection process, useful and compact information can be derived to improve prediction performance. However, correlated relationships under a multivariate framework cancel the effects among input variables, thereby leading to poor selections by the GA. Additionally, regularization is another method for the selection of variables. A linear SVR with l_1-norm minimization was used by reference [27] to select input variables. Once the minimization is achieved, unimportant weights are driven to zero. The Lasso method is another a regularization method that has been used for variable selection problems with small data sets. Typically, variable selection can regularize on a weight parameter by using the Lasso algorithm. However, this method is still not able to guarantee the refinement of proper features but rather partly useful variables to represent the true variations in processes [28]. In this light, this paper proposes a two-stage variable selection strategy, termed Lasso-FA. In this strategy, FA is further used to extract simple features after implementing the Lasso algorithm. Thus, the strategy is able to solve the problem of high computational complexity and avoid model overfitting when performing sequential soft-sensor modelling. Having a very large number of models in an ensemble increases the need for performing such an efficient variable selection strategy. It is important to note that few attempts have been devoted to discussing the variable selection issues with ensemble learning-based soft-sensor modelling, which has strict computational requirements.

Once proper variables are selected by the Lasso-FA algorithm, the next step is to build an ensemble learning-based soft-sensor model. Bagging (bootstrap aggregating) is a well-known, simple yet effective approach for generating an ensemble of independent submodels in which each submodel is trained using a set of random samples of instances taken from the original data set [29]. Majority voting is performed on the predictions of submodels to determine the final prediction of an unseen instance. However, due to the use of random sampling with replacement, the training data set may always include the same samples while rendering other samples useless. To alleviate this issue, random forests [30], gradient boosting machines [31] and extremely randomized trees [32] are proposed accordingly. Considering the high nonlinearity of process patterns, few attempts have been devoted to combining ensemble learning and Deep ANNs for soft sensors in recent years [33]. Additionally, Minku and Yao conducted a comparable study to evaluate how diversity levels affect the prediction performance of nonlinear ensemble models, illustrating that sufficient diversity results in a reduction of the prediction error. Another work by Minku proved this by assembling two decision forests with different levels of diversity [34]. Unfortunately, data distributions and process patterns tend to change over time, thus resulting in the degradation of the predictive performance of standard ensemble learning models. Kolter J. and Maloof M. proposed a dynamic weighted majority strategy to create or remove submodels depending on their real-time predictive performances [35]. However, this adaptive strategy usually achieves highly accurate performances at the cost of intensive computations. This is mainly because these strategies must retrain the prediction model during each update step. Therefore, this paper proposes an adaptive ranking strategy to combine all the submodels. Beyond the variable selection method (called Lasso-FA), the other main contributions of this work are summarized as follows: (*i*) To increase the diversity of the submodels, three kinds of GPR models (global GPR models: PIC models; local GPR models: LPR [sparse probability Gaussian process regression] models; and DDM models) [36] are implemented as submodels. Through coordinating two seemingly different yet complementary characteristics (global and local learning) in an integrative framework, ensemble learning provides an alternative method for refining useful local information without losing the global view of the data [37]. To further increase the diversity of ensemble

learning, this paper uses bootstrapping to stimulate the training data sets and the hyperparameters of the global and local GPR models. Several training sets are reproduced by performing the bootstrap methodology and then used for GPR model training. Beyond the training data, the hyperparameters are also bootstrapped over the assigned range to diversify the GPR models. (*ii*) Different from the standard adaptive strategies, our strategy uses the moving-window technique to rank the submodel predictions, and then average them together to make the final predictions. The motive behind this strategy is based on the observation that few of the submodels with the best performances in the previous few steps will have similar prediction accuracy rates for the incoming new data points. Additionally, due to the use of a GPR model as the submodel, the proposed methodology is able to describe the uncertainties properly.

The remainder of this paper is organized as follows: Section 2 presents basic GPR models for ensemble learning. The proposed soft-sensor framework is presented in Section 3. Section 4 provides a numerical example to illustrate the proposed model. Section 5 discusses the pros and cons of the proposed soft sensor. Finally, concluding remarks and future research ideas are given in Section 6.

2. An overview of Gaussian process regression

GPR is a method for Bayesian nonlinear nonparametric regression. Given a training set $D = (x_i, y_i)|_{i=1}^{N}$ (N is the number of samples), the noisy outputs y_i can be described as a GPR model with the predictive distribution of f. In the GPR model, the noises are assumed to be additive, independent, and Gaussian for the sake of easing the computation. Thus, the relationship between x_i and y_i can be described by the following equations [38]:

$$y_i = f(x_i) + \varepsilon_i \tag{1}$$

$$\varepsilon_i \sim \mathbb{N}(0, \sigma^2) \tag{2}$$

$$f(\cdot) \sim GPR(0, k(\cdot,\cdot)) \tag{3}$$

x is the input data matrix (process variables), where $x \in \mathbb{R}^{N\times d_1}$ and $x_i \in \mathbb{R}^{1\times d_1}$. y is the response data matrix (quality variables), where $y \in \mathbb{R}^{N\times 1}$ and $y_i \in \mathbb{R}^{1\times 1}$. d_1 is the number of variables in the input data. $GPR(0, k(\cdot,\cdot))$ denotes a Gaussian process with a mean and covariance matrix of 0 and $k(\cdot,\cdot)$, respectively. Similarly, Eq. (2) represents the fact that the noises ε adhere to a Gaussian distribution with mean 0 and covariance σ^2. Additionally, $k(i, j)$ can be simplified as k_{ij}. By inference, it is easy to obtain that the outputs follow a multivariate joint Gaussian distribution:

$$y \sim \mathbb{N}(0, K_y) \tag{4}$$

where $K_y = K + \sigma^2 \mathrm{I}$, K_y is the covariance matrix of dimension $N \times N$. The corresponding element is

$$(K_y)_{ij} = cov(y_i, y_j) = k(x_i, x_j) + \sigma^2 \delta_{ij} \tag{5}$$

where δ_{ij} is the Kronecker function. The major difference between K and K_y is that K is noise-free, while K_y is the noise-induced covariance matrix. In the GPR model, the most commonly used covariance matrix is the squared-Exp (SE) kernel, which is shown as follows:

$$k_{ij} = k(x_i, x_j) = cov(f(x_i), f(x_j)) = \sigma_f^2 \exp -\frac{(x_i - x_j)^2}{2l^2} \tag{6}$$

where σ_f^2 and l are hyperparameters in need of identification. It should be noted that commonly used kernels such as the squared-exp (SE), neural network (NN) or Matérn kernels are local kernels [38] that depend only on the scaled Euclidean distance between two points.

Because $(y_1, y_2, \ldots, y_N, f(x^*))^T$ is a Gaussian distribution, the predicted mean and variance values for the new incoming sample x^* ($x^* \in \mathbb{R}^{1\times d_1}$) can be derived as follows:

$$\mu_*' = k_{*x} K_y^{-1} y \tag{7}$$

$$\sigma_*'^2 = k_{**} - k_{*x} K_y^{-1} k_{*x} \tag{8}$$

$k_{*x} = (k(x^*, x_1), \ldots, k(x^*, x_N))^T$ is used to describe the correlated relationship between x^* and x_i. $k_{**} = \mathrm{k}(\mathrm{x}^*, \mathrm{x}^*)$. More details can be found in Supplementary Information A.

2.1. Global GPR model: Sparse pseudo-input Gaussian process regression (PIC-GPR)

To preserve the sparsity and other desirable properties of the standard partially independent conditional GPR model (PIC-GPR), Snelson and Ghahramani proposed a sparse pseudo-input Gaussian process regression (PIC-GPR) [13]. This paper assumes that the likelihood follows Gaussian process distribution. The corresponding likelihood can be parameterized using a pseudo-data set $\tilde{D} = [\tilde{x}, \tilde{f}]_{m=1}^{M}$, where $\tilde{x}$ and $\tilde{f}$ represent pseudo-inputs with $\tilde{x} \in \mathbb{R}^{M\times d_1}$ and pseudo-response variables with $\tilde{f} \in \mathbb{R}^{M\times 1}$, respectively, and $M<N$. Given a new sample x^* the predictive distribution with respect to x^* is then obtained by integrating the likelihood (B.1) with the equation in section B.2 of Supplementary Information B:

$$\mathrm{p}(\mathrm{y}^*|\mathrm{x}^*, \mathrm{D}, \tilde{\mathrm{x}}) = \mathbb{N}(\mathrm{y}^*|\mu^*, \sigma_*^2) \tag{9}$$

where

$$\mu_* = \mathrm{k}_*^{\mathrm{T}} \mathrm{Q}_{\mathrm{M}}^{-1} \mathrm{K}_{\mathrm{MN}} (\Lambda + \sigma^2 \mathrm{I})^{-1} \mathrm{y} \tag{10}$$

$$\sigma_*^2 = \mathrm{k}_{**} - \mathrm{k}_*^{\mathrm{T}} \left(\mathrm{K}_{\mathrm{M}}^{-1} - \mathrm{Q}_{\mathrm{M}}^{-1}\right) \mathrm{k}_* + \sigma^2 \tag{11}$$

where $[\mathrm{K_M}]_{mm'} = \mathrm{k}(\tilde{\mathrm{x}}_m, \tilde{\mathrm{x}}_{m'})$ and $[\mathrm{k}_{x\tilde{x}}]_m = \mathrm{k}(\tilde{\mathrm{x}}_m, \mathrm{x})$. In the above equations, $\mathrm{Q_M} = \mathrm{K_M} + \mathrm{K_{MN}}(\Lambda + \sigma^2 \mathrm{I})^{-1} \mathrm{K_{NM}}$ for $m = 1, \ldots, M$ and $\mathrm{m}' = 1, \ldots, M$. Details about the definition and identification of kernel parameters can be seen in Supplementary Information B. PIC-GPR can be considered as a special case of GPR with a pseudo-input-parameterized covariance. The most attractive attribute of PIC-GPR is that it can converge to an extremely sparse solution for large data sets. It is important to note that pseudo-input points are simulated from true data points but are not limited to these true points, thus being capable of approaching some nonstationary effects.

2.2. Local GPR models: LPR-GPR and DDM-GPR

(1) Sparse probability Gaussian process regression (LPR-GPR)

LPR-GPR is inspired by locally weighted regression with nearest neighbours and aims to learn individual models from few local samples and then adapt to local regions with different behaviours [3]. In the LPR-GPR model, if compact neighbourhoods are located, a model is inferred and trained locally. Then, multimodal behaviours can be learned online once a new testing sample is received. Multiple GPR experts are consequently combined to address these multimodal behaviours. Because the number of neighbours is controlled to be at most 50, the inference process is computationally cheap. If the data are multimodal, each expert focuses on the neighbouring mode to which the test point belongs. The local models are built by the standard GPR model (Eqs. (7)–(8)). The predictive distribution of the ensemble learning is calculated as

$$\mathrm{p}(y^*|x^*, D', \beta) \approx \sum_{j=1}^{\mathrm{L}_1} \pi_{\mathrm{i}} \mathbb{N}(\mu_j^*, \sigma_{*j}^2) \tag{12}$$

where L_1 and S represent the number of experts and the maximum sample size of each local GPR, respectively. When receiving a new sample x^*, the neighbours D' can be selected. To ensure that local prediction can be integrated properly, the probability of a given expert π_i is defined to act as the weighting parameter. π_i can be calculated as the inverse predicted variance of the ith expert. μ_i^* and σ_{*i}^2 are the predictive mean and variance of the ith expert, respectively. The LPR-GPR algorithm is summarized in Supplementary Information C, and the schematic of LPR-GPR can be seen in Supplementary Information H (Fig. S1).

(2) Domain Decomposition Method-Gaussian Process Regression (DDM-GPR)

The basic idea of the DDM is to formulate the GPR as optimization problems that provide local predictions. This method decomposes the domain Ω into L_2 disjoint subdomains $\{\Omega\}_{j=1,\ldots,L_2}$. Let x_j, y_j be the subset that belongs to Ω_j. Then, a local GPR function for each subdomain is inferred. Finally, if the new sample x^* is in Ω_j, the prediction is performed by the GPR model belonging to Ω_j. The procedure of the detailed algorithm can be seen in the paper of reference [16], and a schematic of DDM-GPR can be seen in Supplementary Information H (Fig. S2). In summary, the prediction problem is formulated as a collection of local learning and inference.

3. Adaptive ranking based ensemble learning of Gaussian process models (AR-EGP)

To compress useful information into a few representative features, we propose a two-stage feature selection strategy, termed Lasso-FA (LF). The derived features are then used for sequential soft-sensor modelling. The proposed integrated framework for soft-sensor modelling is illustrated in Fig. 1.

3.1. LASSO-FA for feature selection

In ensemble learning, computational intensity, model complexity and model overfitting must be considered carefully. To address these issues, the Lasso can regularize the weight parameter by the L1 norm. By shrinking the weight estimates, the L1 regularization process is able to regularize light weights to zero and large weights to nonzero values. Thereafter, features with zero weights are cut off while nonzero weights are retained to refine useful features. In the Lasso algorithm, sparsity can be strengthened by utilizing a large regularization constant, thus leading to irrelevant features being regularized by light weights. These relevant or irrelevant features, in turn, enhance the interpretability of the Lasso. Therefore, the Lasso algorithm is able to extract simple and compact information from observed variables. Even though compact information can be derived from the Lasso, it is still necessary to refine the representative features obtained from the raw data. Factor analysis is a commonly used interdependency method for refining compact latent variables from large numbers of observed variables. It is particularly suitable for addressing the cases in which systematic interdependence exists among data variables and commonality can be represented by a few latent factors. Therefore, this paper proposes a novel feature extraction strategy by combining the Lasso algorithm with FA. The Lasso acts as a primary variable selection tool. The resulting variables are then fed to FA, which serves as a secondary variable selection mechanism. By doing so, representative and useful features can be derived as follows:

$$(x_i, y_i)|_{i=1}^N \overset{\text{Lasso}-\text{FA}}{\rightarrow} (x_i', y_i)|_{i=1}^N$$

where $(x_i, y_i)|_{i=1}^N$ represents N pairs of inputs x_i and noisy outputs y_i, whereas $(x_i', y_i)|_{i=1}^N$ represents N pairs of new mapping inputs x_i' and noisy outputs y_i after the Lasso-FA treatment is completed.

Remark 1. The reason why the two-stage strategy uses the Lasso-FA combination rather than FA-Lasso is that the features refined by the Lasso have true meanings and can be maintained; then, the meaningful features from the Lasso can further enhance the extraction of latent factors. If FA is used first in the two-stage strategy, unrealistic information (latent factors) is delivered to the Lasso, and the underlying raw data structure could be somehow destroyed. More details can be found in Supplementary Information D and E.

3.2. Adaptive ranking based ensemble of GPR models (AR-EGP)

(1) Bagging for GPR modelling

Different from fitting a single model to collected data, a large number of submodels are constructed in the proposed AR-EGP model. In ensemble learning, one of the potential ways is to execute this construction process is to increase the diversity of the training samples or submodels. Thus, to increase the diversity, the original training data are resampled by a bagging method and further used to train each submodel. Once moved to the online testing stage, the predicted values from each submodel are integrated to enhance model accuracy and robustness. The overall calculated procedure for resampling by a bagging method is shown as follows:

Given the original training data $Z_A = (x_i', y_i)|_{i=1}^{d_1}$, we randomly sample d (d<d_1) data points from Z_A with selection probabilities of $1/d_1$. For each selection round, d data points are derived and taken as a group to form a training set. The sampling procedure is repeated U times. After that, the U resampled data sets can be obtained as follows: $Z_1, \ldots, Z_U$. Finally, based on the U resampled data sets, U sub-models are developed. In the proposed model, GPR acts as the submodel, which exhibits an unstable modelling pattern and is potentially useful for the bagging method. For a GPR model, hyperparameter estimation tends to converge to local optima, and this leads to an unstable model due to its sensitivity to initialized values. However, hyperparameter sensitivity is, in turn, capable of strengthening the diversity of the learning process. Therefore, we also randomly sample the hyperparameters, 'widths' and 'kernel lengths' from the assigned ranges. In summary, beyond bagging the training data, the hyperparameters of the GPR models are also sampled using bootstrapping over the assigned ranges to ensure the diversity of the GPR models. Suppose U sets of data have been derived. More details about how to conduct ensemble learning of sub-GPR models are discussed sequentially.

(2) Ensemble learning of global–local GPR models

In general, global GPR models, such as standard GPR or PIC-GPR models, are suitable for stationary processes but are not able to handle abrupt local changes or nonstationary features. In contrast, local GPR models, such as DDM-GPR and LPR-GPR, decompose the entire domain into subdomains, train each submodel by using subdomain samples and finally make a prediction for a test point using the related trained submodel. Thus, it is envisioned that local GPR models are able to adapt to nonstationary changes while training a model with low computational intensity. However, local GPR models can achieve discontinuous predictions on the boundaries of subdomains. Given the pros and cons corresponding to global and local GPR models, this paper proposes the use of ensemble learning to coordinate global and local models and ensure that the predicted models are adaptive for both nonstationary processes. The coordination of global and local models increases the diversity of ensemble learning. Diversity is essentially a basic and necessary property for achieving acceptable accuracy rates.

Additionally, unlike standard ensemble learning using the average strategy, our method uses a moving window to select a

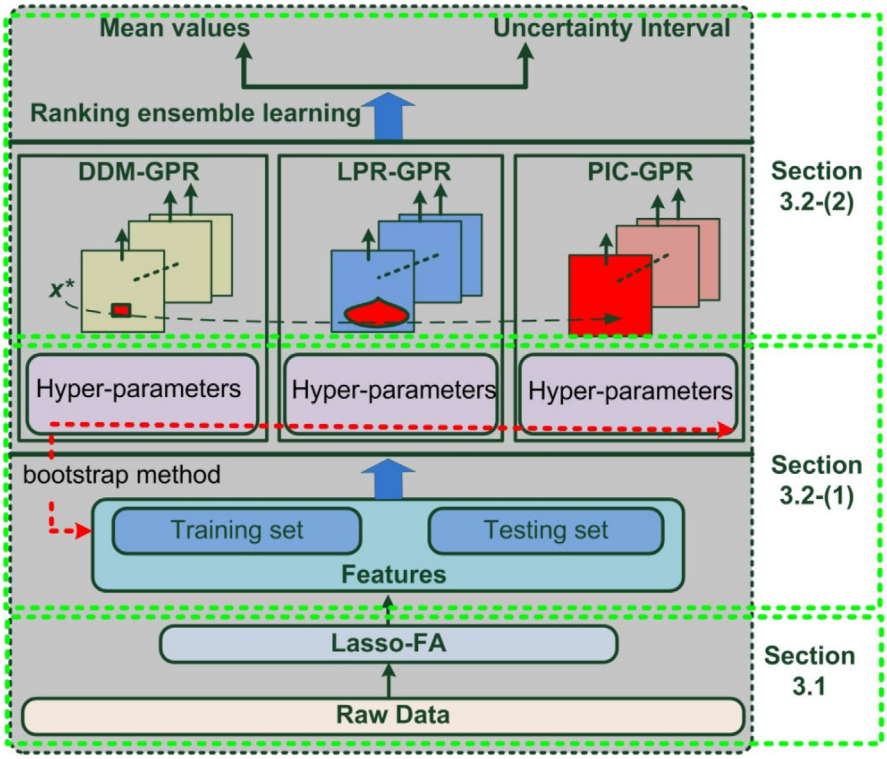

Fig. 1. Graphical illustration of AR-EGP.

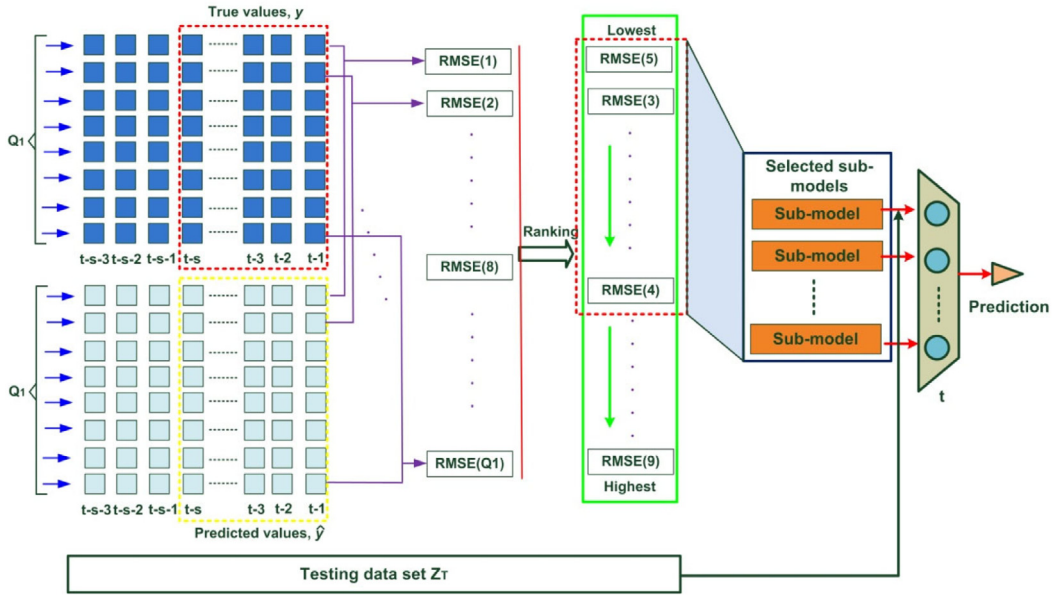

Fig. 2. Graphical illustration of the submodel selection process for online prediction.

few submodels with the best performances (mainly dependent on their RMSE values) to serve as a criterion for justifying the selection of the models used for sequential prediction. The procedure of adaptive ranking with ensemble learning is briefly shown as follows (Fig. 2). Given the original testing samples $Z_T = (x'_i, y_i)|_{i=1}^{d_2}$, the moving window length is s, and the number of submodel prediction values at each predicted step is Q_1. At the current time t, the moving window envelops the data set $(x'_i, y_i)|_{i=t-s}^{t-1}$. Since the predicted values for $x'_{t-s,\ldots,t-1}$ are already known, i.e., the predicted values for Q_1 submodels $(\hat{y}^j_{t-s,\ldots,t-1})|_{j=1}^{Q_1}$ have been derived, the RMSE value of each submodel in the moving window can be obtained by using the following equation:

$$RMSE = \sqrt{\frac{1}{s}\sum_{j=1}^{s}(y_{t-j} - \hat{y}_{t-j})^2} \tag{13}$$

It is reasonable to assume that the prediction performance of the model with respect to time t is similar to the most recent prediction performances, such as the predictions at times $t-s, \ldots, t-1$. Therefore, a few submodels with the best performances at times $t-s, \ldots, t-1$ can be used for prediction at time t once they receive the current input values $x^{*\prime}$. Q_2 submodels with the best

RMSE values in the moving window are selected as the base models for the sequential prediction for a given $x^{*\prime}$.

The predictive mean and variance of the proposed AR-EGP model can be obtained as follows:

$$E(y^*) = \frac{1}{Q_2}\sum_{i=1}^{Q_2}\hat{y}^*(i) \tag{14}$$

$$\text{var}\left(y^*\right) = \frac{1}{Q_2}\sum_{i=1}^{Q_2}\sigma_*^2(i) + \frac{1}{Q_2}\sum_{i=1}^{Q_2}(\hat{y}^*(i) - E(y^*))^2 \tag{15}$$

where $\hat{y}^*(i)$ represents the prediction of the ith model. The motive behind the averaging rule is to alleviate the negative effects of prediction disturbances and to improve the prediction accuracy and robustness of the AR-EGP model. It is important to note that the standard variance can be achieved simultaneously along with the mean values in the GPR model. var (y^*), 2*var (y^*) and 3*var (y^*) represent confidence rates of 68.3%, 95.5% and 99.7%, respectively, meaning that the percentage of erroneous predictions in entire predicted data should be at most 31.7%, 4.5% and 0.3%. Depending on the confidence measurement, soft sensors can indicate how confident each prediction is and can provide interval prediction values that are able to check how reliable the given predictive regions are.

Remark 2. The reason why both global and local GPR models are used to serve as submodels is twofold: (i) the use of global and local GPR models can increase the diversity of ensemble learning, which potentially improves the prediction performance; (ii) predictions of local models are sometimes too aggressive and deviate far from the true values, whereas predictions of global models are too smooth and are likely to converge to the mean of the true values. It is envisioned that the ensemble of the global and local predictions can make the aggregated prediction close to the true value.

4. Case studies

Two case studies are used for validation purposes. The data for both cases were collected from the field. First, a prediction performance comparison is made between the prediction model with and without feature selection. Then, the predictive performance is assessed by comparing the AR-EGP -based soft sensor with other models (PIC: PIC-GPR model; DDM: DDM-GPR model; LPR: LPR-GPR model; BGP: Bagging ensemble of GPR; AGP: Ensemble of GPR with averaged prediction outputs; R-EGP: Ensemble of GPR with ranked prediction outputs)

The root mean square error (*RMSE*) and correlation coefficient (r) are used to assess the predictive performance of the inferential model. The root mean square error (RMSE) criterion is defined as follows for quality comparisons between different methods:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)} \tag{16}$$

where and are the measured and predicted values, respectively.

$$r(Y, \hat{Y}) = \frac{\text{COV}(Y, \hat{Y})}{\sqrt{\text{VAR(Y)VAR}(\hat{Y})}} \tag{17}$$

where Y is the true value vector and $\hat{Y}$ is the prediction value vector. COV represents the covariance, and VAR represents the variance of the model.

4.1. A full-scale WWTP

(1) Background

Activated sludge-based WWTPs (wastewater treatment plants) are widely and efficiently used for organic matter and nutrient removal, accounting for almost 90% of all WWTPs in China. Thus, an activated sludge-based WWTP is studied in the first case study. The microorganism population (both in terms of quality and the number of species) is dynamically varied over time. The dynamic scenario is most severe when suffering from extreme weather conditions. These fluctuations often result in the degradation or failure of online analysers. In this case study, data are sampled at one time every day due to the fact that there are few online sensors. Among the data, 314 points are used for model training, and the remaining 209 samples are used for testing. All the collected variables are tabulated in Table S2 in Supporting Information F.

(2) Performance of feature selection

To verify the proposed feature selection algorithm, the Lasso and FA are separately used with the base learners (PIC, DDM and LPR). To assess the variable selection performances for the base learners, the *RMSE* and correlation coefficient, r, are presented as indicators. As profiled in Table 1, the Lasso-FA variable selection strategy for all three base models (PIC, DDM and LPR) achieved the best performance in predicting biological oxygen demand (BOD) when compared with the other two scenarios (Lasso or FA) based on the *RMSE* and r values obtained. This is mainly because the Lasso-FA algorithm can allow for true feature extractions by using the primary step, Lasso, and then useful dimension reduction by using the secondary step, FA. Among the three variable selection strategies, the prediction performance of the Lasso strategy is better than that of FA. This is mainly because factors derived from FA may lose some useful information by using data mapping to achieve dimension reduction. In contrast, the Lasso is able to select the true variables by using regularization on a weight parameter with little information loss. With the Lasso-FA algorithm, the PIC base learner achieves the most acceptable results in terms of *RMSE* and r. In this case study, the processed data exhibit stationary features, which are most suitable for the global model (PIC), enabling the PIC model to achieve the best performance.

(3) Performance of the AR-EGP algorithm

To evaluate the prediction performance and demonstrate the advantage of the AR-EGP algorithm fairly, the Lasso-FA algorithm is used for variable selection of all predicted models. In this section, the traditional ensemble learning method, BGP, is used as the comparison scenario. Additionally, to display the advantage of the ranking strategy, the standard averaging strategy serves as a comparison scenario as well. As profiled in Table 1, the AR-EGP algorithm performs best in terms of the BOD prediction, with RMSE and r values of 0.31 and 0.96, respectively. This is mainly because of the fact that the AR-EGP model is not only able to capture the diversity of data by using diverse submodels but also to make full use of the best few submodels to adaptively converge to the best prediction. It is important to note that even though both the BGP and AGP models are in fact using ensemble learning with averaging, the use of diverse submodel structures can indeed improve the prediction performances of these two models compared with those of the PIC models (Table 1). Fitted predictions can be seen in Fig. 3.

(4) Uncertainty description

To assess the uncertainty level of a model, the indicator used is the negative log predictive density (NLPD):

$$NLPD = \frac{1}{Q_2}\sum_{i=1}^{Q_2}\left[\frac{(\hat{y}^*(i) - \mu_*(i))}{2\sigma_*^2(i)} + \frac{1}{2}\log(2\pi\sigma_*^2(i))\right] \tag{18}$$

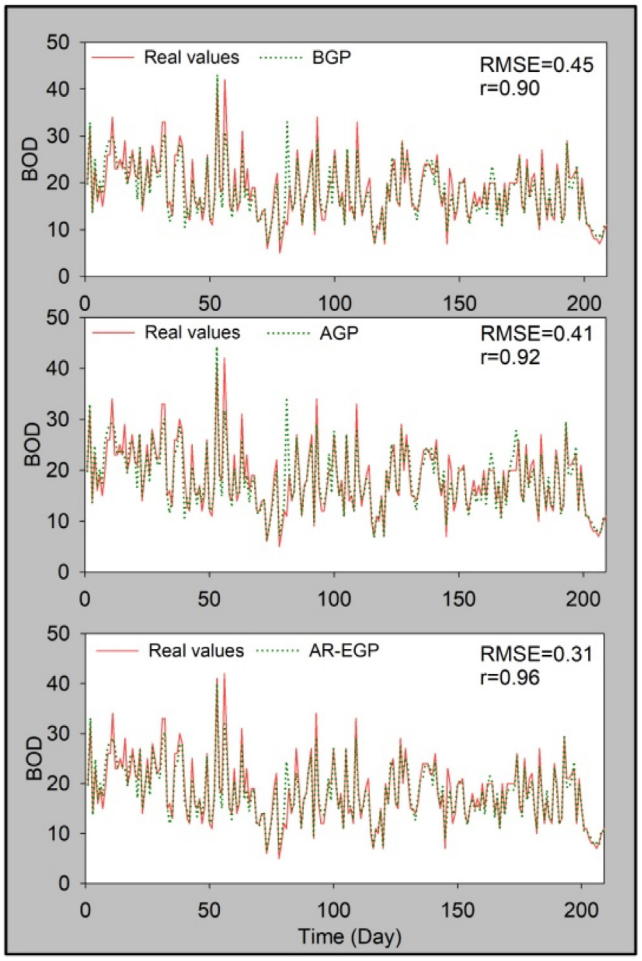

Fig. 3. The BOD prediction performances of the BGP, AGP and AR-EGP models in case study 1.

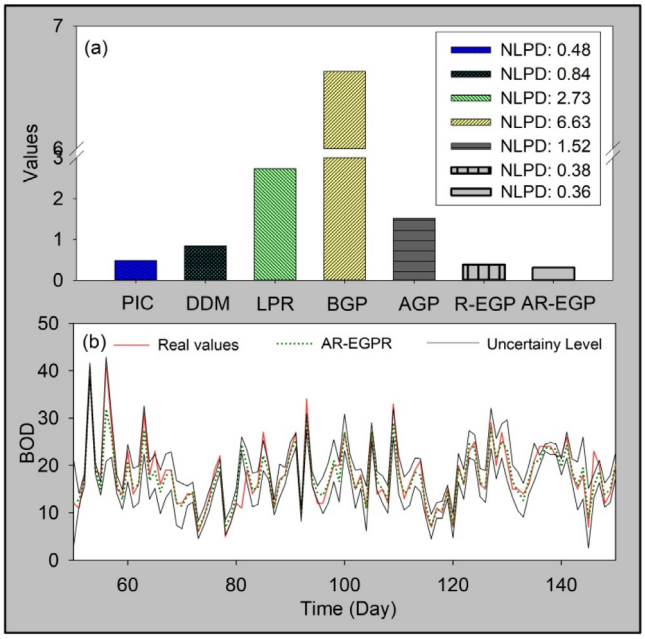

Fig. 4. (a) A comparison of the NPLD values of all GPR models. (b) The fitted predictions of the AR-EGP model and the true values for the testing samples from day 50 to day 150.

Table 1
Prediction performance for the case study 1.

	RMSE - Prediction with and without feature selection methods			*r* - Prediction with and without feature selection methods		
Models	Lasso	FA	Lasso-FA	Lasso	FA	Lasso-FA
PIC	0.71	0.88	0.42	0.73	0.59	0.91
DDM	0.67	0.79	0.55	0.79	0.57	0.83
LPR	0.73	0.85	0.45	0.58	0.53	0.91
BGP	/	/	0.45	/	/	0.9
AGP	/	/	0.41	/	/	0.92
AR-EGP	/	/	0.31	/	/	0.96

Table 2
Empirical Reliability of AR-EGP for the case study 1.

Comments	Empirical Reliability (%)		
Empirical confidence	90%	95%	99%
Predicted confidence	90.43%	96.21%	99.56%

It is important to note that both the predictive variance σ_*^2 and mean μ_* are taken into account in this equation. As profiled in Fig. 4, the NLPD of the AR-EGP model is the smallest value than, implying that the AR-EGP model fits the data with the least uncertainties. In contrast, the PIC, DDM, LPR, BGP, AGP and R-EGP models have higher NLPD values than that of the AR-EGP model. As seen in the definition of the NLPD, a larger RMSE and a smaller predictive variance lead to higher NLPD values. The differences in the NLPD values mainly come from the differences in RMSE and σ_*^2. Due to the close RMSEs of the AR-EGP and AGP models, we can conclude for this case study that the NLPD differences between the AR-EGP and AGP models mainly result from the small predictive variances of the AGP model. The reason why the AGP model has small predictive variances is because the AGP model underestimates them. It is also important to note that the EGP model has the highest RMSE and NLPD values compared with those of the other models, demonstrating that the EGP model may be uncompetitive for stationary data sets (for example, this case study). In other words, the AR-EGP model achieves the best performance compared with those of all other methods when using stationary data sets.

To assess the reliability of the model predictions, predictive intervals that do not include the true values are considered incorrect predictions. Therefore, we use the percentage of samples that are not able to offer a proper predictive region including the true values. Fig. 4(b) shows that some real samples are out of the predictive regions. Table 2 displays the empirical reliability performance.

It is important to highlight that the predictive regions are widened if the values of the process variable deviate from the original state values. The tightness of the corresponding predictive regions can, in turn, be used to indicate a specific significance level and further measure how efficient our algorithm is for prediction.

4.2. A real wastewater plant

(1) Background

The data for this case study are collected from a real WWTP (Beijing, China). This plant aims to collect and treat household wastewater covering 480,000 populations by using a modified activated sludge process, termed the oxidation ditch (OD) process. More details about the background of this plant can be found in Supplementary Information L.

Table 3
Prediction performance for the case study 2.

	RMSE - Prediction with and without feature selection methods			*r* - Prediction with and without feature selection methods		
Models	Lasso	FA	Lasso-FA	Lasso	FA	Lasso-FA
PIC	.54	.59	.37	.83	.83	.91
DDM	.53	.65	.47	.83	.76	.88
LPR	.52	.68	.41	.85	.77	.91
BGP	/	/	0.39	/	/	0.9
AGP	/	/	0.37	/	/	0.91
AR-EGP	/	/	0.29	/	/	0.95

In this case study, a filamentous sludge bulking fault occurred and lasted over almost half a year. Filamentous sludge bulking often results in poor operational performances and untreated wastewater entering rivers directly. The sludge volume index (SVI) is an empirical measurement used in the field to show how severe a case of filamentous sludge bulking is. In this case study, the SVI is set to 200 mL/g and used to act as a control limit to indicate if filamentous sludge bulking is occurring. However, the SVI is always difficult to measure, so we must resort to a lab analysis. To build a model, the selected training variables are tabulated as Table S3 in Supplementary Information G. In this case study, 212 samples are collected, among which the first 127 days of data are for training, whereas the others are for testing.

(2) Performance of feature selection

The comparative scenarios are defined similarly to those in case study 1. As displayed in Table 3, the Lasso-FA variable selection strategy with all three base models (PIC, DDM and LPR) achieved the best performances for SVI prediction when compared with the performances of the other two scenarios (Lasso or FA) in terms of *RMSE* and *r*. This is mainly because the Lasso-FA algorithm allows for true feature extractions by using the primary step (Lasso) and then an efficient dimension reduction by using the secondary step (FA). Obviously, the prediction performance of the Lasso strategy is better than that of FA based on the *RMSE* and *r* values in Table 3. This is mainly because factors derived from FA may lose some useful information by using data mapping to achieve dimension reduction. In contrast, the Lasso is able to select the true variables by using regularization on a weight parameter with little information loss. With the Lasso-FA algorithm, the PIC base learner achieves the most acceptable results based on the *RMSE* and *r* values in Table 3. However, the performance of PIC in case study 1 is slightly better than that in case study 2 in terms of *r* (Table 3). This can be explained by the slightly nonstationary features in case study 2. Overall, coordination of the Lasso and FA approaches can indeed improve the prediction accuracy of soft sensors.

(3) Performance of the AR-EGP algorithm

The Lasso-FA algorithm is used for variable selection of all predicted models in this case study. In this section, the traditional ensemble learning model, BGP, is used as a comparison scenario. Additionally, to display the advantage of the adaptive ranking strategy, the AGP model acts as a comparison scenario as well. As profiled in Table 3, the AR-EGP model achieves the best performance for SVI prediction, with RMSE and *r* values of 0.29 and 0.95, respectively. This can be explained by the ability of the AR-EGP model to adaptively deal with the processed data. It is also obvious that the deviation occurs mainly in the stages with significant variation. Even though the performances of the BGP model and AGP are poorer than that of the AR-EGP model in terms of RMSE and *r*, averaging ensemble learning with diverse submodel structures can indeed improve the prediction performance over those of the PIC models (Table 3). The main reason why averaging ensemble learning is able to improve the prediction performance is that the generalization error of averaging ensemble learning is proven to be smaller than the expected error of each submodel individually. The prediction performance of the AGP model can be further validated by the differences between the predicted and true values, as shown in Fig. 5(b).

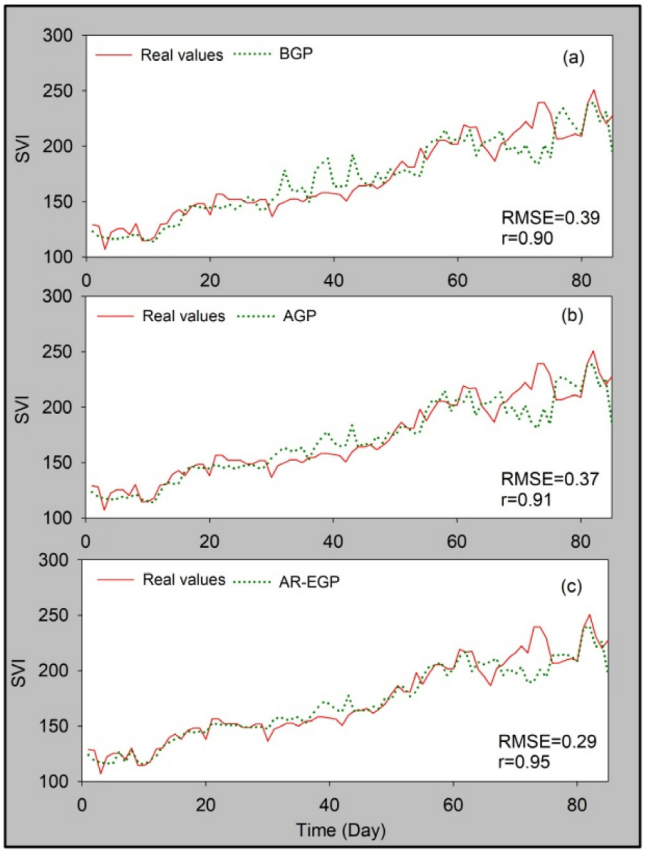

Fig. 5. The SVI prediction performance of the BGP, AGP and AR-EGP models in case study 2.

(4) Uncertainty description

As shown in Fig. 6, the AR-EGP model achieves the smallest NLPD, suggesting that the AR-EGP model fits the data best. For this case study, the PIC, DDM, LPR, BGP, AGP, and R-EGP models have larger NLPD values than that of the AR-EGP model. As seen in the definition of the NLPD, a larger RMSE and a smaller predictive variance lead to higher NLPD values. The differences in the NLPD values mainly come from the differences in RMSE and σ_*^2. Due to the close RMSEs of the AR-EGP and AGP models, we can conclude for this case study that the NLPD differences between the AR-EGP and AGP models mainly result from the small predictive variances of the AGP model. The reason why the AGP model has small predictive variances is because the AGP model underestimates them. For the slightly nonstationary data set, the performances of all predicted models with respect to their NLPD values are very similar (Figs. 4(a) and 6(a)). The BGP model has the highest RMSE and NLPD values, demonstrating that the BGP model may not be suitable for either stationary or nonstationary data sets. By comparing the R-EGP and AR-EGP models in Fig. 4(a), it is also important to note that they achieve very similar performances with NLPD values of 0.38 and 0.36, respectively. In contrast, the NLPD of the AR-EGP model is 19.4% better than that of the R-EGP model for the nonstationary scenario. This can be explained by the fact that the moving-window-based

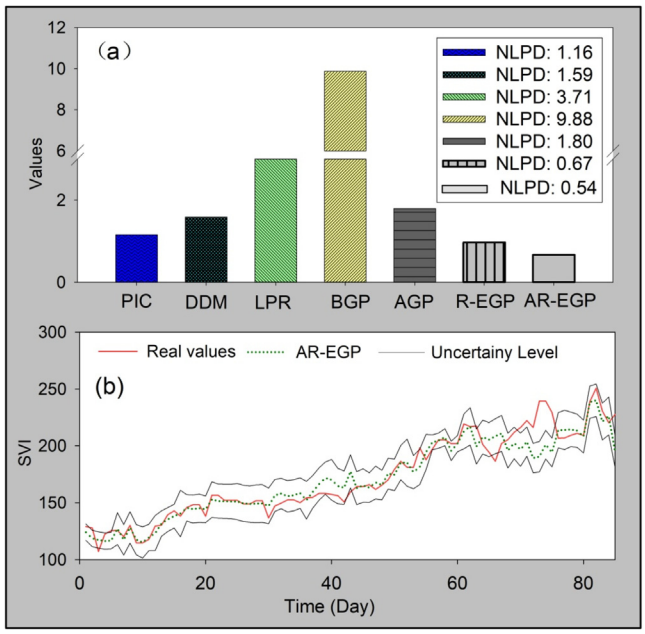

Fig. 6. (a) Comparison of all GPR models in terms of their NPLD values. (b) The differences between the values the predicted by the AR-EGP model and the true values for the testing samples.

Table 4
Empirical Reliability of AR-EGP for the case study 1.

Comments	Empirical Reliability (%)		
Empirical confidence	90%	95%	99%
Predicted confidence	82%	95.4%	99.7%

adaptive ranking strategy is capable of enhancing the prediction performance of a model. Overall, even for a nonstationary pattern, the AR-EGP model is able to achieve the best performance.

To further check the reliability of the prediction performance, the samples that are not included in prediction intervals are considered incorrect predictions, as in case study 1. To quantify the predictive reliability of a model, the successful prediction rate should be greater than the desired accuracy. Table 4 suggests that the predicted results at the 90% confidence level are not acceptable since 18% of the predicted values are outside of the control intervals. In contrast, the predicted results at the 95% and 99% confidence levels perform well. As profiled in Fig. 6(b), the differences between the predicted values and true values are acceptable, even though this is true only with an 82% confidence level. The deviations between predicted and true values mainly occur in places with abrupt changes.

Fig. 6(b) shows that the confidence could be widened due to the deviations of steady state values. This enables one to further check the tightness of predictive regions over a specific significance level.

5. Discussions

5.1. Parameter discussions

We present a variable selection methodology using the Lasso together with FA as a template. Regularization methodologies, such as the Lasso algorithm, can work together with dimension reduction methods, such as principal component analysis (PCA) and partial least squares (PLS). These regularization methodologies are purposely used to remove under-correlated model input variables, and the dimension reduction methods aim to compress over-correlated model input variables. This variable selection method can reduce the computational intensity and model complexity sequentially.

To clarify the uses of all the hyperparameters, we have summarized them in Fig. 7. In stage 1, the number of cross validation (CV) steps for Lasso learning should be selected. In case study 1, 314 samples with 19 variables are used for Lasso-based variable selection learning, whereas 127 samples with 13 variables are used in case study 2. To ensure that the Lasso algorithm can be used for validation, we must guarantee that $\frac{314}{CV} > 19$ in case 1 and $\frac{123}{CV} > 13$ in case 2. This means that the Lasso regression matrix based on the validated data set should be non-singular. By considering this point, the CV steps of cases 1 and 2 are set to 30 and 20, respectively. At around these iteration numbers, the Lasso values can converge to a stable state. Since we have already implemented the Lasso algorithm for the primary variable selection process, an 85% control limit for accumulated contributions is used for both cases 1 and 2. As tabulated in Table 5 (Line 3), three factors are sufficient to represent all the necessary information for sequential modelling. Thus, it is envisioned that the Lasso-FA algorithm is able to lower the complexity of the model and improve its predictive performance.

It is important to note that to increase the diversity of ensemble learning, the "number of pseudo-inputs" in PIC-GPR, the "number of training samples allocated to each local expert" in LPR-GPR and the "mesh size" in DDM-GPR are all bootstrapped by specifying a range, and these ranges are shown in Table 5. The minimum value of the specified range should be greater than 19 (the variable number in case 1) and 13 (the variable number in case 2) to ensure that there are sufficient samples to train a local model, whereas the maximal value should ensure that a sufficient number of submodels can be derived and not cost too much in terms of computational intensity. For the "number of random sites chosen for local hyperparameter learning" in LPR-GPR and the "number of locations to check for the continuity of predictions over the boundary" in DDM-GPR, these two hyperparameters are used to decide the number of locations used to calculate the local hyperparameters. The "covariance function" is one of the important hyperparameters for determining the shape of a GPR-based prediction curve. By having deep insight into the evolution of the training data set, we use 'covSEard'+'covNoise' in case study 1. As profiled in Fig. 3, the processed data exhibit a relatively stable pattern but with high variations around a mean value. In contrast, the processed data in case 2 exhibit a decreasing trend with many variations. Therefore, 'covSEard'+'covLINard' is selected as the covariance function of all GPR models.

To optimize the "average ranking with ensemble learning" model, two hyperparameters (the window size and the number of selected submodels) are required for a proper setup. As profiled in Fig. 2, the samples in the window are mainly used to calculate RMSE values to justify the submodels selected for sequential ensemble learning. In case 1, since the variations of the processed data are relatively stable, the window size is selected as 6. However, to adapt to the decreasing trend in case 2, the window size must be shortened to 4 to ensure that proper submodels can be selected by capturing the most recent variations. Additionally, in this manuscript, 1/3 of the submodels with the best RMSE values are selected for sequential ensemble learning. In fact, an alternative method is to set up a control limit for the RMSE and to select the submodels with RMSE values greater than the control limit.

5.2. Future works

In this paper, the proposed AR-EGP model is validated by simulated and field data. The results demonstrate that the AR-EGP model performs well for both case studies. Even though the

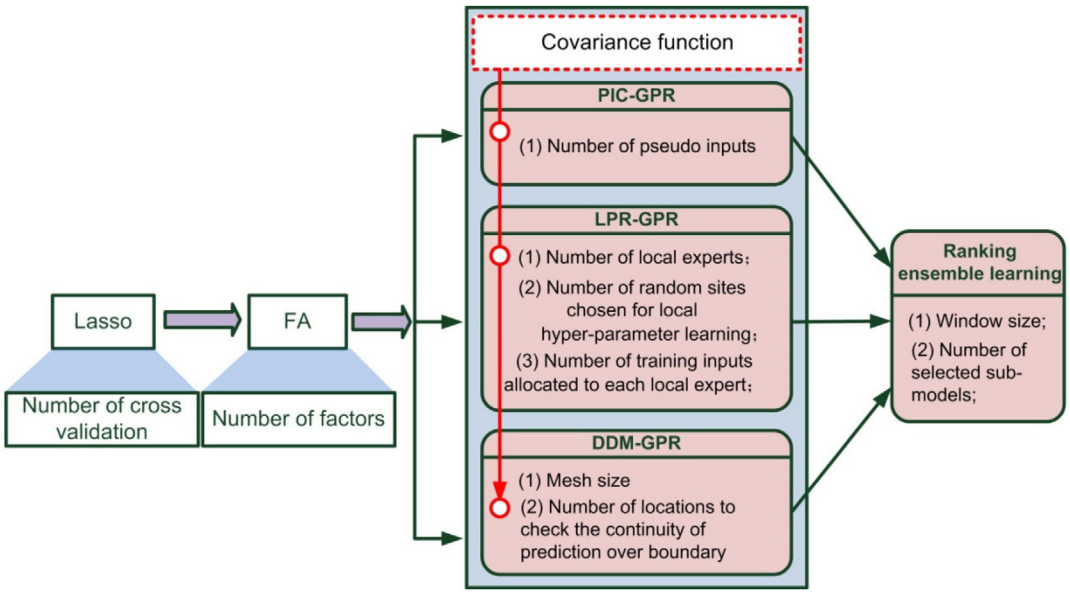

Fig. 7. Hyperparameter Analysis.

Table 5
Hyperparameter analysis for both case studies.

Algorithms	Sub-models	Hyper-parameters	Case study 1	Case study 2
Lasso	(/)	Number of cross validation	30	20
FA	(/)	Number of factors	3	3
Sub-model	PIC-GPR	Number of pseudo inputs	Bootstrapping [19, 40]	Bootstrapping [13, 30]
	LPR-GPR	Number of local experts	18	20
		Number of random sites chosen for local hyper-parameter learning	20	10
		Number of training samples allocated to each local sub-model	Bootstrapping [19, 49]	Bootstrapping [13, 43]
	DDM-GPR	Mesh size	Bootstrapping [19, 58]-> [30,58]	Bootstrapping [13, 58]-> [40, 58]
		Number of locations to check the continuity of prediction over boundary	10	5
	(/)	Covariance Function	'covSEard'+'covNoise'	'covSEard'+'covLINard'
Ranking ensemble learning	(/)	Window size	6	4
	(/)	Number of selected sub-models	20	10

AR-EGP model works well and can represent the true scenarios, the proposed method still requires the implementation in a programmable logic controller (PLC) rather than the limit in MATLAB. Additionally, it is necessary to take extreme conditions into account, for example, other fault patterns. Many disasters come from unpredictable, abnormal events. Fortunately, the predictive intervals of the AR-EGP model can alleviate this issue by indicating how reliable the predicted values are. In this way, the predicted intervals sound alarms before the occurrence of the failure. Although ensemble learning and GPR models have previously been demonstrated to be able to predict hard-to-measure variables in extreme conditions, further investigation is required to gain insight into how to predict a variable reliably and robustly. In our study, global and local GPR models are considered submodels for prediction. Additionally, GPR models can provide sufficient information for descriptions of uncertainty. It is envisaged that fuzzy logic models offer similar performances in terms of their abilities to capture uncertainties. Future research will focus on the integration of soft sensors with fault diagnosis and process management.

eXtreme Gradient Boosting (XGBoost), CatBoost and LightGBM are three of boosting ensemble learning algorithms and are all based on Gradient Boosting Decision Tree (GBDT) algorithm [39]. XGBoost proposed a pre-sorted algorithm to enhance GBDT, aiming improve the prediction efficiency of GBDT. However, XGBoost is mainly based on the depth-wise or level-wise search optimization, which will implicates intense computational costs. LightGBM and CatBoost can lower the computational requirement while maintain prediction performance. Future researches will devote to learning these boosting algorithms to improve the proposed AR-EGP to be more accurate with less computational costs.

6. Conclusions

To monitor quality-related but hard-to-measure variables in industrial processes, this paper proposes an adaptive ensemble learning framework for Gaussian process models. The framework is able to coordinate local and global GPR models to capture

process behaviours properly and to ensemble the sub-GPR models adaptively to obtain robust and accurate predictions. Additionally, the proposed methodology can describe uncertainties and show how reliable the predicted values are. The proposed prediction model, AR-EGP, is validated in a simulated WWTP with stationary behaviours and a true WWTP with a drifting fault. Quality-related variables can be predicted effectively by the AR-EGP model, with RMSE values of 0.31 and 0.96 in the first case study and values of 25.6% and 21.6% in the second case study, which are better than those of the bagging GPR model and the average ensemble GPR model. Ensemble learning achieves improved accuracy at the cost of increasing the computational cost required. Future works will focus on how to build an ensemble learning model with low computational intensity and how to optimize the structure of the ensemble learning model.

CRediT authorship contribution statement

Yiqi Liu: Material preparation, Data collection, Analysis, Performed the experiments, Wrote the paper, Funding. **Daoping Huang:** Material preparation, Data collection, Analysis, Funding. **Bin Liu:** Material preparation, Data collection, Analysis, Reviewed and revised the paper. **Qiang Feng:** Material preparation, Data collection, Analysis. **Baoping Cai:** Material preparation, Data collection, Analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (61873096, 62073145), Guangdong Basic and Applied Basic Research Foundation (2020A1515011057), Guangdong Technology International Cooperation Project Application (2020A0505100024), Fundamental Research Funds for the Central Universities , SCUT, China (D2201200).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.asoc.2020.107060.

References

[1] L. Wang, H. Jin, X. Chen, J. Dai, K. Yang, D. Zhang, Soft sensor development based on the hierarchical ensemble of Gaussian process regression models for nonlinear and non-Gaussian chemical processes, Ind. Eng. Chem. Res. 55 (2016) 7704–7719.
[2] P. Zhou, S.W. Lu, T. Chai, Data-driven soft-sensor modeling for product quality estimation using case-based reasoning and fuzzy-similarity rough sets, IEEE Trans. Autom. Sci. Eng. 11 (2014) 992–1003.
[3] Y. Liu, C. Yang, M. Zhang, Y. Dai, Y. Yao, Development of adversarial transfer learning soft sensor for multigrade processes, Ind. Eng. Chem. Res. 59 (2020) 16330–16345.
[4] Y. Liu, M. Xie, Rebooting data-driven soft-sensors in process industries: A review of kernel methods, J. Process Control 89 (2020) 58–73.
[5] B. Cai, L. Huang, M. Xie, Bayesian networks in fault diagnosis, IEEE Trans. Ind. Inf. 13 (2017) 2227–2240.
[6] B. Cai, X. Shao, Y. Liu, X. Kong, H. Wang, H. Xu, W. Ge, Remaining useful life estimation of structure systems under the influence of multiple causes: Subsea pipelines as a case study, IEEE Trans. Ind. Electron. 67 (2020) 5737–5747.
[7] Q. Feng, X. Zhao, D. Fan, B. Cai, Y. Liu, Y. Ren, Resilience design method based on meta-structure: A case study of offshore wind farm, Reliab. Eng. Syst. Saf. 186 (2019) 232–244.
[8] L. Xie, J. Zeng, C. Gao, Novel just-in-time learning-based soft sensor utilizing non-Gaussian information, IEEE Trans. Control Syst. Technol. 22 (2014) 360–368.
[9] C.-C. Yang, M.-D. Shieh, A support vector regression based prediction model of affective responses for product form design, Comput. Ind. Eng. 59 (2010) 682–689.
[10] R. Boloix-Tortosa, J.J. Murillo-Fuentes, F.J. Payan-Somet, F. Perez-Cruz, Complex Gaussian processes for regression, IEEE Trans. Neural Netw. Learn. Syst. 29 (2018) 5499–5511.
[11] B. Sun, Y. Li, Z. Wang, Y. Ren, Q. Feng, D. Yang, An improved inverse Gaussian process with random effects and measurement errors for RUL prediction of hydraulic piston pump, measurement, 2020, 108604.
[12] B. Sun, Y. Li, Z. Wang, L. Zhifeng, Q. Xia, Y. Ren, Q. Feng, D. Yang, C. Qian, Physics-of-failure and computer-aided simulation fusion approach with a software system for electronics reliability analysis, Eksploat. Niezawodn. Maint. Reliab. 22 (2020) 340–351.
[13] E. Snelson, Z. Ghahramani, Sparse Gaussian process using pseudo-inputs, Adv. Neural Inf. Process. Syst. 18 (2006) 1257–1264.
[14] E. Snelson, Z. Ghahramani, Local and global sparse Gaussian process approximations, J. Mach. Learn. Res. Proc. Track 2 (2007) 524–531.
[15] X. Yuan, Z. Ge, B. Huang, Z. Song, Y. Wang, Semisupervised JITL framework for nonlinear industrial soft sensing based on locally semisupervised weighted PCR, IEEE Trans. Ind. Inf. 13 (2017) 532–541.
[16] C. Park, J. Huang, Y. Ding, Domain decomposition approach for fast Gaussian process regression of large spatial data sets, J. Mach. Learn. Res. 12 (2011) 1697–1728.
[17] A. Schwaighofer, V. Tresp, Transductive and inductive methods for approximate Gaussian process regression, 2003.
[18] R. Urtasun, T. Darrell, Sparse probabilistic regression for activity-independent human pose inference, 2008.
[19] C. Rasmussen, Z. Ghahramani, Infinite mixtures of Gaussian process experts, Adv. Neural Inf. Process. Syst. 2 (2002).
[20] Y. Liu, B. Liu, X. Zhao, M. Xie, A mixture of variational canonical correlation analysis for nonlinear and quality-relevant process monitoring, IEEE Trans. Ind. Electron. 65 (2018) 6478–6486.
[21] B. Krawczyk, L.L. Minku, J. Gama, J. Stefanowski, M. Woźniak, Ensemble learning for data stream analysis: A survey, Inf. Fusion 37 (2017) 132–156.
[22] F. Curreri, S. Graziani, M.G. Xibilia, Input selection methods for data-driven soft sensors design: Application to an industrial process, Inform. Sci. 537 (2020) 1–17.
[23] T. Mehmood, K.H. Liland, L. Snipen, S. Sæbø, A review of variable selection methods in partial least squares regression, Chemom. Intell. Lab. Syst. 118 (2012) 62–69.
[24] S. Kim, M. Kano, H. Nakagawa, S. Hasebe, Input variable scaling for statistical modeling, Comput. Chem. Eng. 74 (2015) 59–65.
[25] Y. Liu, B. Liu, X. Zhao, M. Xie, Development of RVM-based multiple-output soft sensors with serial and parallel stacking strategies, IEEE Trans. Control Syst. Technol. 27 (2019) 2727–2734.
[26] Y. Liu, Y. Pan, D. Huang, Development of a novel adaptive soft-sensor using variational Bayesian PLS with accounting for online identification of key variables, Ind. Eng. Chem. Res. 54 (2015) 338–350.
[27] A. Beinrucker, Ü. Dogan, G. Blanchard, Extensions of stability selection using subsamples of observations and covariates, Statist. Comput. 26 (2016) 1059–1077.
[28] K. Hirose, S. Konishi, Variable selection via the weighted group lasso for factor analysis models, Canad. J. Statist. 40 (2012) 345–361.
[29] L. Breiman, Bagging predictors, Mach. Learn. 24 (1996) 123–140.
[30] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.
[31] J. Friedman, Greedy function approximation: A gradient boosting machine, Ann. Statist. 29 (2000).
[32] A. Paul, A. Furmanchuk, W.-k. Liao, A. Choudhary, A. Agrawal, Property prediction of organic donor molecules for photovoltaic applications using extremely randomized trees, Mol. Inform. 38 (2019) 1900038.
[33] Z.-H. Zhou, J. Feng, Deep forest: Towards an alternative to deep neural networks, 2017.
[34] L. Minku, X. Yao, DDD: A new ensemble approach for dealing with concept drift, IEEE Trans. Knowl. Data Eng. 24 (2012) 619–633.
[35] J. Kolter, M. Maloof, Dynamic weighted majority: A new ensemble method for tracking concept drift, 2003.
[36] Y.S. Liu, Yarong, Jurg Keller, Philip Bond, Guangming Jiang, Prediction of concrete corrosion in sewers with hybrid Gaussian processes regression model, RSC Adv. 7 (2017) 30894–30903.
[37] O. Sagi, L. Rokach, Ensemble learning: A survey, Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 8 (2018) e1249.
[38] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, MIT Press, London, 2006.
[39] C. Bentéjac, A. Csörgő, G. Martínez-Muñoz, A comparative analysis of gradient boosting algorithms, Artif. Intell. Rev. (2020).