Adaptive Soft Sensor Based on Online Support Vector Regression and Bayesian Ensemble Learning for Various States in Chemical Plants

Hiromasa Kaneko and Kimito Funatsu

Department of Chemical System Engineering, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

Corresponding author: Kimito Funatsu

Tel: +81-3-5841-7751

Fax: +81-3-5841-7771

E-mail: funatsu@chemsys.t.u-tokyo.ac.jp

Abstract

A soft sensor predicts the values of some process variable y that is difficult to measure. To maintain the predictive ability of a soft sensor model, adaptation mechanisms are applied to soft sensors. However, even these adaptive soft sensors cannot predict the y -values of various process states in chemical plants, and it is difficult to ensure the predictive ability of such models on a long-term basis. Therefore, we propose a method that combines online support vector regression (OSVR) with an ensemble learning system to adapt to nonlinear and time-varying changes in process characteristics and various process states in a plant. Several OSVR models, each of which has an adaptation mechanism and is updated with new data, predict y -values. A final predicted y -value is calculated based on those predicted y -values and Bayes' rule. We analyze a numerical dataset and two real industrial datasets, and demonstrate the superiority of the proposed method.

Keywords: Process control; Soft sensor; Degradation; Online support vector regression; Ensemble learning; Bayes' rule.

1 Introduction

Soft sensors can predict process variables that are difficult to measure online, and have been widely used in industrial modeling [1-3]. The relationship between a difficult-to-measure variable (y -variable) and easy-to-measure variables (X -variables) is constructed theoretically or statistically. In practice, data-driven soft sensors, in which models are built using statistical methods and databases, are mainly employed. By inputting values of the X -variables into the constructed model, the values of a y -variable can be predicted online, because the X -variables are measured in real time and the prediction is very fast.

However, soft sensors have some practical problems. One crucial issue is the degradation of the soft sensor models [4]. The predictive accuracy of soft sensors decreases gradually, a result of the changing state of a chemical plant due to factors such as catalyzing performance loss, and sensor and process drift.

To reduce this degradation, adaptation mechanisms can be applied to soft sensors [5]. For example, new X -variable data are measured in a chemical plant and used to reconstruct the soft sensor models and predict the y -variables. Kaneko and Funatsu categorized the degradation of soft sensor models and adaptive soft sensor models, such as moving window (MW), just-in-time (JIT), and time difference (TD) models [4]. MW models [6,7] are constructed with a recently measured dataset; JIT models [8-10] are constructed by assigning larger weights to the data that are most similar to the prediction data; and TD models [11,12] are constructed by considering the time difference of a y -variable and that of the X -variables. Ensemble learning can be applied to adaptive models [13,14]. In addition, when data distributions are multimodal, multiple modeling approaches [15,16] can be combined with adaptive models.

The characteristics of MW, JIT, and TD models have been discussed and analyzed using numerical and real industrial data [4]. These models are not entirely sufficient when rapid temporal changes occur in a process, and thus, novel techniques are required.

We previously combined an online support vector regression (OSVR) method [17], in which a nonlinear SVR [18] model is efficiently updated (see Appendix A), with a time variable [19], and discussed the optimization of the SVR hyperparameters for the OSVR model. The SVR hyperparameters can be selected using cross-validation on training data of reasonable length, and the OSVR model with this time variable can adapt to abrupt changes in process characteristics, even when the relationship between X and y is nonlinear and time varying [20].

However, one soft sensor model cannot accurately predict $\mathbf{y}$ -values for all process states in a chemical plant. An OSVR model must first set the SVR hyperparameters C , ε , and γ using a training dataset (see Appendix A), and the resulting model will be able to predict $\mathbf{y}$ -values for process states that are similar to that of the training data. However, when the process state changes significantly, the predictive ability of the OSVR model decreases.

We therefore propose a combination of more than one OSVR model, and use ensemble learning to allow the soft sensors to adapt to various process states in a chemical plant. Multiple OSVR models with different hyperparameters predict multiple $\mathbf{y}$ -values. We combine the predicted $\mathbf{y}$ -values based on the current predictive ability of each OSVR model and Bayes' rule, which has been actively applied to soft sensor analyses [16,21-23], to produce a final predicted $\mathbf{y}$ -value. The proposed method is referred to as ensemble online support vector regression (EOSVR). The predictive ability of multiple OSVR models updated with new data, and weighted appropriately, allows the accurate prediction of $\mathbf{y}$ -values in each process state. In addition, the standard deviation of the predicted $\mathbf{y}$ -values enables us to estimate the prediction error in the final predicted $\mathbf{y}$ -value for each process state.

To verify the effectiveness of the proposed method, we analyze numerical simulation data in which the relationships between $\mathbf{X}$ and $\mathbf{y}$ are strongly nonlinear and change from moment to moment. The performance of the proposed model is compared with that of other traditional adaptive models. Then, the proposed method is applied to two real industrial processes, an alkylaluminum production process and an exhaust gas denitration process.

2 Method: EOSVR

2.1 Offline Analysis

In the proposed EOSVR method, multiple combinations of the SVR hyperparameters C , ε , and γ (see Appendix A) are optimized offline for various states in a plant. The basic concept of the optimization of multiple combinations of SVR hyperparameters is shown in Figure 1. First, the window size ws is set, and then the SVR hyperparameters are optimized by moving the window. The value of ws is relatively unimportant, as OSVR models are insensitive to this parameter [20]. A certain size of ws suffices for OSVR models. The data in each window are then as follows:

$$(\mathbf{X}_1, \mathbf{y}_1), (\mathbf{X}_2, \mathbf{y}_2), \dots, (\mathbf{X}_i, \mathbf{y}_i), \dots, (\mathbf{X}_n, \mathbf{y}_n), \quad (1)$$

where $\mathbf{X}_i \in R^{ws \times v}$ and $\mathbf{y}_i \in R^{ws \times 1}$ are the i^{th} data set of the $\mathbf{X}$ -variables and that of a $\mathbf{y}$ -variable, respectively (v is the number of $\mathbf{X}$ -variables). When the window is moved by h data points, the i^{th} dataset is from the $h(i-1)+1^{\text{th}}$ data point to the hi^{th} data point. For each dataset, the SVR hyperparameters are optimized. The optimized hyperparameters are represented as follows:

$$(C_1, \varepsilon_1, \gamma_1), (C_2, \varepsilon_2, \gamma_2), \dots, (C_i, \varepsilon_i, \gamma_i), \dots, (C_n, \varepsilon_n, \gamma_n). \quad (2)$$

To optimize the SVR hyperparameters, we use an exhaustive grid search through the candidates of each hyperparameter, and the performance is measured by cross-validation on the training data. Duplicate combinations of the SVR parameters are removed, leaving the following m combinations of parameters:

$$(C_1, \varepsilon_1, \gamma_1), (C_2, \varepsilon_2, \gamma_2), \dots, (C_m, \varepsilon_m, \gamma_m). \quad (3)$$

2.2 Online Analysis

Figure 2 shows the basic concept of prediction using the proposed method. New $\mathbf{X}$ -variable data $\mathbf{x}(t)$ at time t are input into the m OSVR models; the models predict $\mathbf{y}$ -values as follows:

$$\begin{aligned} y_{p,1}(t) &= f_1(\mathbf{x}(t)) \\ y_{p,2}(t) &= f_2(\mathbf{x}(t)) \\ &\vdots \\ y_{p,m}(t) &= f_m(\mathbf{x}(t)) \end{aligned}, \quad (4)$$

where $f_1, f_2, \dots, f_m$ are the OSVR models. We then combine $y_{p,1}(t), y_{p,2}(t), \dots, y_{p,m}(t)$ to obtain a final predicted $\mathbf{y}$ -value $y_p(t)$ using Bayes' rule [22]. When S is the current (unobserved) state in a plant and M_i is the i^{th} OSVR model, the probability of M_i given S , $P(M_i|S)$, is required to combine the prediction results of the m OSVR models. Given $P(M_i|S)$, the final predicted $\mathbf{y}$ -value $y_p(t)$ is obtained as follows:

$$y_p(t) = \sum_{i=1}^m P(M_i | S) y_{p,i}(t). \quad (5)$$

Using Bayes' rule, $P(M_i|S)$ is calculated as follows:

$$P(M_i | S) = \frac{P(S | M_i) P(M_i)}{\sum_{j=1}^m P(S | M_j) P(M_j)}. \quad (6)$$

In this study, the prior probability of each M_i is assumed to be equal, that is:

$$P(M_i) = \frac{1}{m}. \quad (7)$$

In practice, by changing this prior probability, we can make the strategy of model combination more flexible. Then, we require $P(S|M_i)$, which represents the predictive ability of the i^{th} OSVR model in the current state S . In this paper, we assume that the predictive ability is inversely proportional to the root-mean-square error (RMSE). Hence, $P(S|M_i)$ is given as follows:

$$P(S | M_i) = \frac{z_i}{\sum_{j=1}^m z_j}, \quad (8)$$

where

$$z_i = \frac{1}{RMSE_i^2}. \quad (9)$$

$RMSE_i$ is the RMSE of the i^{th} OSVR model with the current dataset (for example, the latest ws data).

However, $RMSE_i$ will be too small for overfitted OSVR models, because the OSVR models are updated with the current dataset and $RMSE_i$ is calculated with the same data. Although cross-validation would normally be used to assess the prediction

accuracy of a model, we cannot cross-validate $RMSE_i$ because the OSVR models are simply updated, rather than reconstructed, and there is no concept of training data for these models. To evaluate the predictive ability of nonlinear regression models without cross-validation, the $RMSE_{\text{midknn}}$ measure was developed by Kaneko and Funatsu [24]. This is the RMSE for the midpoints between the k -nearest-neighbor data points. Further details are given in Appendix B. $RMSE_{\text{midknn},i}$ can be calculated for updated models or OSVR models. z_i can also be represented as follows:

$$z_i = \frac{1}{RMSE_{\text{midknn},i}^2}. \quad (10)$$

Then, the final predicted $\mathbf{y}$ -value $y_p(t)$ can be calculated using Eq. (5), considering the predictive ability of each OSVR model in the current state of a chemical plant.

Of course, when new data are added to the database and the window is moved (see Figure 1), a new combination of SVR parameters must be optimized. If the new parameters are not a duplication of the old parameter combinations, a new OSVR model is constructed.

2.3 Estimation of Prediction Errors

EOSVR produces both a final predicted $\mathbf{y}$ -value $y_p(t)$ and an estimated prediction error. Prediction errors are estimated using the standard deviation (STD) of the multiple predicted $\mathbf{y}$ -values represented in Eq. (4) (see Figure 2). When the predicted $\mathbf{y}$ -values are close together and the STD is small, the prediction error will also be small. Conversely, when the predicted $\mathbf{y}$ -values vary widely and the STD is large, the prediction error will also be large. The STD can be used as an index of the prediction errors.

The STD is an index of monitoring prediction reliability (IMPR). This was previously known as the “distance to model” (DM) [25,26] as a concept of the applicability domain [27,28]. However, the DM is a slightly confusing term, and so we use IMPR, or more simply, prediction reliability (PR). The PR values can be used to give the estimated prediction errors [28].

3 Results and Discussion

To verify the effectiveness of the proposed method, we analyzed a numerical simulation dataset in which the relationship between $\mathbf{X}$ and $\mathbf{y}$ is strongly nonlinear, and

two real industrial process datasets. The models used for comparison are as follows:

- JITPLS (just-in-time partial least squares): PLS (partial least squares) model constructed with data whose Euclidian distances to new data are smaller than those of other data
- LWPLS (locally weighted PLS): PLS model constructed by weighting data with a similarity matrix [8,29]
- MWPLS (moving window PLS): PLS model constructed with most-recently measured data
- OSVR: Updated SVR model [20]
- EOSVR: Proposed model

The optimal number of components in PLS modeling is selected using 5-fold cross-validation, and the maximum size of the database for JITPLS and LWPLS is 5000 data points. The localization parameter for LWPLS is determined by 5-fold cross-validation as one of 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, and 1.5. The window sizes for JITPLS, MWPLS, OSVR, and EOSVR are 50 data points. The SVR hyperparameters C , ε , and γ (see Appendix A) are optimized using 5-fold cross-validation on the training set for the OSVR. C is varied from 2^0 to 2^{11} by squaring; ε is varied from 2^{-10} to 2^0 by squaring; and γ is increased from 2^{-15} to 2^{11} by squaring. These candidate parameter values are the same as those of EOSVR. For EOSVR, ws is set to 50, and h is set to 20 in the numerical simulation data analysis, and to 50 in the real industrial data analyses. z_i is calculated from the RMSE (Eq. (9)) or $RMSE_{midknn}$ (Eq. (10)), and these EOSVR models are described as EOSVR(RMSE) and EOSVR($RMSE_{midknn}$), respectively.

In the case studies, we do not compare any TD models. TD models can adapt to shifts in both $\mathbf{y}$ -values and $\mathbf{X}$ -values, but cannot adapt to a change of slope between $\mathbf{X}$ and $\mathbf{y}$ [4], which is the focus of this paper.

3.1 Data with a Nonlinear Relationship between $\mathbf{X}$ and $\mathbf{y}$

We verify the performance of the proposed method when the relationship between $\mathbf{X}$ and $\mathbf{y}$ is nonlinear. The relationship between $\mathbf{X}$ and $\mathbf{y}$ in the data is as follows:

$$\begin{aligned} \exp(\mathbf{y}) = & (1 + (\mathbf{x}_1 + \mathbf{x}_2 + 1)^2(19 - 14\mathbf{x}_1 + 3\mathbf{x}_1^2 - 14\mathbf{x}_2 + 6\mathbf{x}_1\mathbf{x}_2 + 3\mathbf{x}_2^2)) \\ & \times (30 + (2\mathbf{x}_1 - 3\mathbf{x}_2)^2(18 - 32\mathbf{x}_1 + 12\mathbf{x}_1^2 + 48\mathbf{x}_2 - 36\mathbf{x}_1\mathbf{x}_2 + 27\mathbf{x}_2^2)) \end{aligned} \quad (11)$$

Equation (11) has been used by Li et al. [30] as a test problem. The $\mathbf{X}$ -variable data were generated as a random walk of within ± 2 . Figure 3 shows the time plots of $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{y}$. Random numbers from the normal distribution with a mean of 0 and standard deviation of 0.01 were added to the $\mathbf{y}$ -variable. A total of 2000 data points were generated; the first 1000 were used for training, and the next 1000 were used as test data.

Table 1 shows the prediction results. r_p^2 is the determination coefficient for the test data, and represents the prediction accuracy of each model. The higher the r_p^2 -value of a model, the more accurate the model prediction. $RMSE_p$ is the $RMSE$ for the test data. Lower $RMSE_p$ -values imply higher prediction accuracy. The prediction results given by OSVR were better than those of JITPLS, LWPLS, and MWPLS. This is because the OSVR model can, to some extent, adapt to nonlinear time-varying changes in the relationship between $\mathbf{X}$ and $\mathbf{y}$. Furthermore, the prediction accuracy was greatly improved by the proposed EOSVR($RMSE$) and EOSVR($RMSE_{midknn}$) models with 24 OSVR models, i.e., $m = 24$ in Eqs. (3)–(8). The $RMSE_p$ -values for EOSVR are almost half of those for OSVR. The nonlinearity in Eq. (11) depends on the values of $\mathbf{x}_1$ and $\mathbf{x}_2$, and the EOSVR models achieved high predictive ability for the changing nonlinearity. The $RMSE_p$ -value of EOSVR($RMSE$) is slightly greater than that of EOSVR($RMSE_{midknn}$), but the difference is small compared with that of the $RMSE_p$ -values of the other models.

The relationships between observed and predicted $\mathbf{y}$ -values of test data for the JITPLS, OSVR, and EOSVR models are shown in Figure 4. From the plots of JITPLS and OSVR, there appear to be several data points whose predicted errors are large when the observed $\mathbf{y}$ -values are less than 7. The JITPLS model and the OSVR model could not adapt to this process state, i.e., the nonlinearity between $\mathbf{X}$ and $\mathbf{y}$. In contrast, the proposed EOSVR models predicted all $\mathbf{y}$ -values accurately. Even when the observed $\mathbf{y}$ -values were low, the proposed model was able to adapt to the relationship between $\mathbf{X}$ and $\mathbf{y}$. In this data analysis, there is little difference between the performance of EOSVR($RMSE$) and that of EOSVR($RMSE_{midknn}$). This is because the prediction data were similar to the dataset used to construct the OSVR models, and model overfitting did not overly affect the prediction result.

Figure 5 shows the relationship between the STD and the absolute prediction errors. When the STD is small, the absolute prediction errors are also small and concentrated close to 0. As the STD becomes large, the variance of the absolute prediction errors increases. The size of the prediction errors can be estimated using the STD as IMPR.

3.2 Application to an Alkylaluminum Production Process

We applied the proposed method to industrial data obtained from an alkylaluminum production process. One of the raw materials is powdered aluminum, and alkylaluminum is produced through a series of reactions.

The density of aluminum slurry is controlled online by changing the amount of powdered aluminum, and a soft sensor can be used to predict the density. Therefore, the y -variable is the density and the $\mathbf{X}$ -variables are 48 process variables such as temperature, pressure, flow rate, and liquid level in various reactors, tanks, and pipes. We prepared 20,000 data (20,000 min.) from March 2013; the first 10,000 formed the training dataset, and the remaining 10,000 were the test data.

The prediction results for the density are shown in Table 2. In this process, MW models such as MWPLS and OSVR had higher predictive ability than JIT models such as JITPLS and LWPLS. The change in the relationship between $\mathbf{X}$ and y occurs because JIT models cannot adapt to this type of change, whereas MW models can to some extent [4]. The r_p^2 -values of the proposed EOSVR models with 22 OSVR models ($m = 22$) exceeded 0.9, and the proposed method exhibited the best performance. The $RMSE_p$ -values are almost half those of the JIT models. The amount of powder aluminum changes, and accordingly the process characteristics change. Therefore, several process states exist in the alkylaluminum production process. The proposed EOSVR models could adapt to these changes, and could predict the density accurately across multiple process states. The performance of EOSVR(RMSE) is the same as that of EOSVR(RMSE_{midknn}). The prediction data may be similar to those used to update the OSVR models, and the overfitting had little effect on the results, as for the data analyzed in Section 3.1.

Figure 6 shows time plots of the measured and predicted density. The left-hand side (Figure 6-a,c,e) gives the OSVR results, which are superior to those of the traditional methods, and the right-hand side (Figure 6-b,d,f) shows those of EOSVR(RMSE). The prediction errors of EOSVR(RMSE) are smaller than those of OSVR for each point in time. In addition, the EOSVR model adapted appropriately to each process state, corresponding to different amounts of powdered aluminum, and had the highest predictive ability. For example, after about 16000 min. (see Figure 6-e,f), a notable process change occurs, and the proposed model was able to predict the y -values accurately following this change.

The relationship between STD and absolute prediction errors is shown in Figure 7. As in Section 3.1, the prediction errors are small when the STD is small, and

the distribution of prediction errors becomes wide as the STD increases. We can consider the STD as an index to estimate the prediction errors of the soft sensor model. The above results confirm that the proposed models are able to deal with sudden and time-varying changes in the process states, and estimate the prediction errors for each data point.

3.3 Application to an Exhaust Gas Denitration Process

To verify the prediction ability, we applied the proposed method to an exhaust gas denitration process at Mitsui Chemicals, Inc. NH_3 is injected into a denitration reactor where exhaust gas passes through a catalytic layer, and NOx is decomposed into harmless N_2 and water vapor. The reactions are as follows:

Soft sensors are used to control the remaining NH_3 concentration at the outlet of the denitration reactor (denitration outlet NH_3), NOx concentration at the outlet of the denitration reactor (denitration outlet NOx), and NOx concentration at the outlet of the outlet gas duct (outlet gas duct NOx). Therefore, the $\mathbf{y}$ -variables are the denitration outlet NH_3 , denitration outlet NOx , and outlet gas duct NOx , and the $\mathbf{X}$ -variables are 23 process variables such as temperature, pressure, and flow rate at the gas mixer and the denitration reactor. We prepared 20000 data (20,000 min.) from November 2012; the first 10,000 were used to train the models, and the remaining 10,000 were used as the test data.

Table 3 shows the prediction results for denitration outlet NH_3 , denitration outlet NOx , and outlet gas duct NOx . Considering the r_p^2 -values and the $RMSEP$ -values, the predictive accuracy for the denitration outlet NOx and outlet gas duct NOx were improved using the proposed method. For denitration outlet NH_3 , the r_p^2 -values increased more than 0.1, and the predictive accuracy significantly improved with the proposed EOSVR models. The $RMSEP$ -values of the proposed models decreased by about 15–50% compared with the traditional models. We confirmed the highly predictive ability of the proposed EOSVR models. For EOSVR, 26, 22, and 23 OSVR models were constructed for the denitration outlet NH_3 , denitration outlet NOx , and outlet gas duct NOx , respectively.

Examples of the time plots of measured and predicted denitration outlet NH_3 concentration are shown in Figure 8. The left-hand side (Figure 8-a,c,e) shows the OSVR results, which exhibit the best performance of the traditional methods, and the right-hand side (Figure 8-b,d,f) shows the results given by EOSVR($\text{RMSE}_{\text{midknn}}$). Clearly, the predicted $\mathbf{y}$ -values of EOSVR($\text{RMSE}_{\text{midknn}}$) are closer to the measured $\mathbf{y}$ -values than are those of OSVR. Even when the measured $\mathbf{y}$ -values varied because of changes in fuel oil pressure, the amount of fuel, and the ignition and extinction of burners, the EOSVR model could adapt to the changes accurately, and could handle the different process states in the plant, reflecting the high predictive ability of the proposed model.

Figure 9 shows the relationships between STD and absolute prediction errors for the denitration outlet NH_3 , denitration outlet NO_x , and outlet gas duct NO_x . For all $\mathbf{y}$ -variables, an adequate correlation between STD and absolute prediction errors was verified, which means that the distribution of absolute prediction errors increased as the STD increased. Using STD as an index of IMPR, we can estimate the prediction error for each new input data point.

We confirmed the high predictive ability and accurate estimation of prediction errors of the proposed model.

4 Conclusions

We proposed an ensemble learning soft sensor method based on multiple OSVR models with an adaptation mechanism. Several OSVR models with different hyperparameters are constructed and updated with new data. For prediction, Bayes' rule is employed to combine the multiple $\mathbf{y}$ -values predicted by the OSVR models, giving a final predicted $\mathbf{y}$ -value. In addition, the error in the predicted $\mathbf{y}$ -value can be estimated using the STD of the $\mathbf{y}$ -values predicted by the OSVR models. The STD ensures the reliability of the predicted $\mathbf{y}$ -value, and can be used to detect and diagnose $\mathbf{y}$ -analyzer faults and process faults.

The effectiveness of the proposed method and its superiority over traditional adaptive soft sensors were demonstrated by analyzing a numerical simulation dataset in which the relationship between $\mathbf{X}$ and $\mathbf{y}$ is nonlinear and two real industrial datasets. Even when the process had several states, the proposed EOSVR model was able to adapt to nonlinear time-varying changes of process characteristics, and exhibited high predictive ability.

In this paper, abnormal data were eliminated in advance. However, abnormal

data will typically be included in the database from which the EOSVR model is constructed, and this will decrease the predictive ability of the model. Hence, in the future, we will combine anomaly detection and database management [31] with the proposed method.

The application of our proposed method to process control should enable the effective and stable operation of chemical plants. This approach can also achieve highly accurate predictions and adapt to process characteristics and multiple process states.

Acknowledgements

The authors acknowledge the support of the Core Research for Evolutional Science and Technology (CREST) project “Development of a knowledge-generating platform driven by big data in drug discovery through production processes” of the Japan Science and Technology Agency (JST). H. Kaneko is grateful for the financial support of the Japan Society for the Promotion of Science (JSPS) through a Grant-in-Aid for Young Scientists (B) (No. 24760629). The authors acknowledge the support of Mitsui Chemicals, Inc.

References

- [1] M. Kano, Y. Nakagawa, Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry, *Comput. Chem. Eng.* 32 (2008) 12–24.
- [2] P. Kadlec, B. Gabrys, S. Strandt, Data-driven soft sensors in the process industry, *Comput. Chem. Eng.* 33 (2009) 795–814.
- [3] M. Kano, K. Fujiwara, Virtual sensing technology in process industries: Trends and challenges revealed by recent industrial applications, *J. Chem. Eng. Jpn.* 46 (2013) 1–17.
- [4] H. Kaneko, K. Funatsu, Classification of the degradation of soft sensor models and discussion on adaptive models, *AIChE J.* 59 (2013) 2339–2347.
- [5] P. Kadlec, R. Grbic, B. Gabrys, Review of adaptation mechanisms for data-driven soft sensors, *Comput. Chem. Eng.* 35 (2011) 1–24.
- [6] S.J. Qin, Recursive PLS algorithms for adaptive data modeling, *Comput. Chem. Eng.* 22 (1998) 503–514.
- [7] J.L. Liu, D.S. Chen, J.F. Shen, Development of self-validating soft sensors using fast moving window partial least squares, *Ind. Eng. Chem. Res.* 49 (2010) 11530–11546.
- [8] S. Schaal, C.G. Atkeson, S. Vijayakumar, Scalable techniques from onparametric

- statistics for real time robot learning, *Appl. Intell.* 17 (2002) 49–60.
- [9] C. Cheng, M.S. Chiu, A new data-based methodology for nonlinear process Modeling, *Chem. Eng. Sci.* 59 (2004) 2801–2810.
- [10] K. Fujiwara, M. Kano, S. Hasebe, A. Takinami, Soft-sensor development using correlation-based just-in-time modeling, *AIChE J.* 55 (2009) 1754–1765.
- [11] H. Kaneko, K. Funatsu, Maintenance-free soft sensor models with time difference of process variables, *Chemom. Intell. Lab. Syst.* 107 (2011) 312–317.
- [12] H. Kaneko, K. Funatsu, A soft sensor method based on values predicted from multiple intervals of time difference for improvement and estimation of prediction accuracy. *Chemom. Intell. Lab. Syst.* 109 (2011) 197–206.
- [13] P. Kadlec, B. Gabrys, Local learning-based adaptive soft sensor for catalyst activation prediction. *AIChE J.* 57 (2010) 1288–1301.
- [14] R. Grbića, D. Slišković, P. Kadlec, Adaptive soft sensor for online prediction and process monitoring based on a mixture of Gaussian process models, *Comput Chem Eng.* 58 (2013) 84–97.
- [15] J. Yu, Multiway gaussian mixture model based adaptive kernel partial least squares regression method for soft sensor estimation and reliable quality prediction of nonlinear multiphase batch processes, *Ind. Eng. Chem. Res.* 51 (2012) 13227–13237.
- [16] J. Yu, A Bayesian inference based two-stage support vector regression framework for soft sensor development in batch bioprocesses, *Comput Chem Eng.* 41 (2012) 134–144.
- [17] J. Ma, J. Theliler, S. Perkins, Accurate on-line support vector regression, *Neural Comput.* 15 (2003) 2683–2703
- [18] C.M. Bishop, *Pattern recognition and machine learning*, Springer, New York, 2006.
- [19] H. Kaneko, K. Funatsu, Application of online support vector regression for soft sensors, *AIChE J.* 60 (2014) 600–612.
- [20] H. Kaneko, K. Funatsu, Adaptive soft sensor model using online support vector regression with the time variable and discussion on appropriate hyperparameters and window size, *Comput Chem Eng.* 58 (2013) 288–297.
- [21] W. Yan, H. Shao, X. Wang, Soft sensing modeling based on support vector machine and Bayesian model selection, *Comput Chem Eng.* 28 (2004) 1489–1498.
- [22] S. Khatibisepehr, B.A. Huang, S. Khare, Design of inferential sensors in the process industry: A review of Bayesian methods, *J. Process Control*, 23 (2013) 1575–1596.
- [23] Z.Q. Ge, B.A. Huang, Z.H. Song, Mixture semisupervised principal component

- regression model and soft sensor application, *AIChE J.* 60 (2014) 533–545.
- [24] H. Kaneko, K. Funatsu, Criterion for evaluating the predictive ability of nonlinear regression models without cross-validation, *J. Chem. Inf. Model.* 53 (2013) 2341–2348.
- [25] I.V. Tetko, I. Sushko, A.K. Pandey, H. Zhu, A. Tropsha, E. Papa, T. Öberg, R. Todeschini, D. Fourches, A. Varnek, Critical assessment of QSAR models of environmental toxicity against *Tetrahymena pyriformis*: Focusing on applicability domain and overfitting by variable selection. *J. Chem. Inf. Model.* 48 (2008) 1733–1746.
- [26] I. Sushko, S. Novotarskyi, R. Korner, A.K. Pandey, A. Cherkasov, J.X. Lo, P. Gramatica, K. Hansen, T. Schroeter, K.R. Muller, L.L. Xi, H.X. Liu, X.J. Yao, T. Oberg, F. Hormozdiari, P.H. Dao, C. Sahinalp, R. Todeschini, P. Polishchuk, A. Artemenko, V. Kuz'min, T.M. Martin, D.M. Young, D. Fourches, E. Muratov, A. Tropsha, I. Baskin, D. Horvath, G. Marcou, C. Muller, A. Varnek, V. Prokopenko, I. V. Tetko, Applicability domains for classification problems: Benchmarking of distance to models for Ames mutagenicity set. *J. Chem. Inf. Model.* 50 (2010) 2094–2111.
- [27] D. Horvath, G. Marcou, A. Varnek, Predicting the predictability: A unified approach to the applicability domain problem of QSAR models. *J. Chem. Inf. Model.* 49 (2009) 1762–1776.
- [28] H. Kaneko, M. Arakawa, K. Funatsu, Applicability domains and accuracy of prediction of soft sensor models, *AIChE J.* 57 (2011) 1506–1513.
- [29] S. Kim, M. Kano, H. Nakagawa, S. Hasebe, Estimation of active pharmaceutical ingredients content using locally weighted partial least squares and statistical wavelength selection, *Int. J. Pharm.* 421 (2011) 269–274.
- [30] G. Li, V. Aute, S. Azarm, An accumulative error based adaptive design of experiments for offline metamodeling, *Struct. Multidiscip. O.* 40 (2010) 137–155.
- [31] H. Kaneko, K. Funatsu, Database monitoring index for adaptive soft sensors and the application to industrial process, *AIChE J.* 60 (2014) 160–169.

Appendix A: SVR and OSVR [20]

The SVR method applies a support vector machine (SVM) to regression analysis, and can be used to construct non-linear models by applying a kernel trick along with the SVM. The OSVR method efficiently updates an SVR model to meet the Karush–Kuhn–Tucker (KKT) conditions that must be fulfilled when training data are added or deleted.

The primal form of the SVR can be given as the following optimization problem.

Minimize

$$\frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^N |y_i - f(\mathbf{x}_i)|_{\varepsilon}, \quad (\text{A.1})$$

where y_i and $\mathbf{x}_i$ are training data, f is the SVR model, $\mathbf{w}$ is a weight vector, ε is a threshold, N is the number of training data, and C is a penalty factor that controls the trade-off between model complexity and training errors. The second term of Eq. (A.1) is the ε -insensitive loss function, which is given as follows:

$$|y_i - f(\mathbf{x}_i)|_{\varepsilon} = \max(0, |y_i - f(\mathbf{x}_i)| - \varepsilon). \quad (\text{A.2})$$

By minimizing Eq. (A.1), we can construct a regression model with a good balance between its generalization capabilities and its ability to adapt to the training data. A $\mathbf{y}$ -value predicted by inputting data $\mathbf{x}$ is represented as follows:

$$f(\mathbf{x}) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(\mathbf{x}_i, \mathbf{x}) + b, \quad (\text{A.3})$$

where b is a constant and K is a kernel function. The kernel function in our application is a radial basis function:

$$K(\mathbf{x}_i, \mathbf{x}) = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}\|^2\right), \quad (\text{A.4})$$

where γ is a tuning parameter that controls the width of the kernel function. From Eqs. (A.1) and (A.2), α_i and α_i^* in Eq. (A.3) can be obtained by minimizing the following equation:

$$\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N K_{ij} (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) - \sum_{i=1}^N y_i (\alpha_i - \alpha_i^*) + \varepsilon \sum_{i=1}^N (\alpha_i + \alpha_i^*) \quad (\text{A.5})$$

subject to

$$0 \leq \alpha_i, \alpha_i^* \leq C \quad i = 1, 2, \dots, N \quad (\text{A.6})$$

$$\sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0 \quad (\text{A.7})$$

K_{ij} in Eq. (A.5) is represented as follows:

$$K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) \quad (\text{A.8})$$

Next, we define θ_i as follows:

$$\theta_i = \alpha_i - \alpha_i^* \quad (\text{A.9})$$

From Eqs. (A.3), (A.4), and (A.8), the predicted $\mathbf{y}$ -value of data point $\mathbf{x}_i$ is given as:

$$f(\mathbf{x}_i) = \sum_{j=1}^N K_{ij} \theta_j + b \quad (\text{A.10})$$

where θ_i satisfies the following equation:

$$\sum_{i=1}^N \theta_i = 0 \quad (A.11)$$

C , ε , and γ are the hyperparameters of an SVR model, and we must first set these parameters.

The OSVR method can rapidly update α_i and b . The error function h is defined as:

$$\begin{aligned} h(\mathbf{x}_i) &\equiv f(\mathbf{x}_i) - y_i \\ &= \sum_{j=1}^N K_{ij} \theta_j + b - y_i \end{aligned} \quad (A.12)$$

The KKT conditions can then be summarized as follows:

$$h(\mathbf{x}_i) > \varepsilon, \quad \theta_i = -C \quad (A.13)$$

$$h(\mathbf{x}_i) = \varepsilon, \quad -C < \theta_i < 0 \quad (A.14)$$

$$|h(\mathbf{x}_i)| < \varepsilon, \quad \theta_i = 0 \quad (A.15)$$

$$h(\mathbf{x}_i) = -\varepsilon, \quad 0 < \theta_i < C \quad (A.16)$$

$$h(\mathbf{x}_i) < -\varepsilon, \quad \theta_i = C \quad (A.17)$$

Each training datum must satisfy one of Eqs. (A.13)–(A.17).

When new data $\mathbf{x}_c$ and y_c are added, there is no need to update the SVR model θ_i , b if $\mathbf{x}_c$ and y_c satisfy Eq. (A.15). Conversely, if $\mathbf{x}_c$ and y_c satisfy Eq. (A.13) or Eq. (A.17), or the data satisfy Eq. (A.14) or Eq. (A.16), the initial value of θ_c (θ_i corresponding to $\mathbf{x}_c$ and y_c) is set to 0, and θ_c , θ_i , and b are gradually changed to meet the KKT conditions. One training data point will move to another region because of the slight changes. The changes and movement are repeated until all the training data satisfy the KKT conditions¹⁸. When one datum is deleted from the training data, the same iterative calculation is performed until all the data satisfy the KKT conditions.

In this study, previously developed OSVR programs (<http://onlinesvr.altervista.org/>)

were used as the machine learning software.

Appendix B: $RMSE_{\text{midknn}}$ [24]

$RMSE_{\text{midknn}}$ is the $RMSE$ calculated with only the midpoints between the k -nearest-neighbor data points (MIDKNNs) in the training data. In the calculation of the MIDKNNs, the k data points that are most similar to a given data point are selected. In this study, the Euclidean distance was used as an index of the similarity. The k midpoints with the lowest Euclidean distance values were obtained for each data point. When the number of data points is m , the number of MIDKNNs is $m \times k$, and the overlapping midpoints are deleted from the MIDKNNs. Then, $RMSE_{\text{midknn}}$ is defined as follows:

$$RMSE_{\text{midknn}} = \sqrt{\frac{\sum_{\text{midknn}=1}^{N_{\text{midknn}}} (y^{\text{midknn}} - y_{\text{pred}}^{\text{midknn}})^2}{N_{\text{midknn}}}} \quad (\text{B.1})$$

where N_{midknn} is the number of MIDKNNs, y^{midknn} is the y -value of the MIDKNN, and $y_{\text{pred}}^{\text{midknn}}$ is the predicted y -value of the MIDKNN. The corrected $RMSE_{\text{midknn}}$ resulting from the reduction of the variance in the calculation of the midpoints is given as follows:

$$RMSE_{\text{midknn}} = \sqrt{\frac{2(m+1) \sum_{\text{midknn}=1}^{N_{\text{midknn}}} (y^{\text{midknn}} - y_{\text{pred}}^{\text{midknn}})^2}{m(N_{\text{midknn}} - 1)}} \quad (\text{B.2})$$

The derivation of Eq. (B.2) was given by Kaneko and Funatsu²⁵. The predictive performance can be estimated without cross-validation from Eq. (B.2).

Through case studies on simulation data with a nonlinear relationship between $\mathbf{X}$ and $\mathbf{y}$, and two real datasets, it was confirmed that $RMSE_{\text{midknn}}$ could evaluate the predictive ability of the nonlinear regression models, and select the hyperparameters with which to construct the predictive model. The case studies demonstrated that $k = 10$ is suitable for $RMSE_{\text{midknn}}$, regardless of differences in the number of training data and $\mathbf{X}$ -variables.

List of Figure Captions

Figure 1. Basic concept of optimizing the multiple combinations of SVR hyperparameters. Datasets $(\mathbf{X}_1, \mathbf{y}_1)$, $(\mathbf{X}_2, \mathbf{y}_2)$, ..., $(\mathbf{X}_i, \mathbf{y}_i)$, ..., and $(\mathbf{X}_n, \mathbf{y}_n)$ are used to optimize the SVR hyperparameters $(C_1, \varepsilon_1, \gamma_1)$, $(C_2, \varepsilon_2, \gamma_2)$, ..., $(C_i, \varepsilon_i, \gamma_i)$, ..., and $(C_n, \varepsilon_n, \gamma_n)$, respectively.

Figure 2. Basic concept of prediction using EOSVR. Std. dev. (standard deviation, STD) of $y_{p,1}(t)$, $y_{p,2}(t)$, ..., and $y_{p,m}(t)$ is used to assess the prediction error of $y_p(t)$.

Figure 3. Time plots of $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{y}$ for the numerical simulation data.

Figure 4. Relationships between observed and predicted $\mathbf{y}$ -values of the test data when the numerical simulation data were generated with Eq. (11).

Figure 5. Relationship between STD and absolute prediction errors in the numerical simulation data.

Figure 6. Time plots of measured and predicted density. Left side (a, c, e) shows results of OSVR and right side (b, d, f) shows results of EOSVR(RMSE).

Figure 7. Relationship between STD and absolute prediction errors in the alkylaluminum production process.

Figure 8. Time plots of measured and predicted denitration outlet NH_3 concentrations. Left side (a, c, e) shows results of OSVR and right side (b, d, f) shows results of EOSVR(RMSE_{midknn}).

Figure 9. Relationships between STD and absolute prediction errors in the exhaust gas denitration process.

Table 1. Prediction results using simulation data where the relationship between $\mathbf{X}$ and $\mathbf{y}$ is nonlinear.

	r_p^2	$RMSE_p$
JITPLS	0.963	0.50
LWPLS	0.950	0.58
MWPLS	0.962	0.49
OSVR	0.989	0.27
EOSVR (RMSE)	0.997	0.12
EOSVR ($RMSE_{midknn}$)	0.997	0.14

Table 2. Prediction results for density in the alkylaluminum production process.

	r_p^2	$RMSE_P$
JITPLS	0.743	18.0
LWPLS	0.684	19.9
MWPLS	0.855	13.5
OSVR	0.895	11.5
EOSVR	0.928	9.5
(RMSE)		
EOSVR	0.928	9.5
($RMSE_{midknn}$)		

Table 3. Prediction results in the exhaust gas denitration process.

	Denitration outlet NH ₃		Denitration outlet NO _x		Outlet gas duct NO _x	
Model	r_p^2	$RMSEP$	r_p^2	$RMSEP$	r_p^2	$RMSEP$
JITPLS	0.667	0.136	0.936	1.92	0.941	1.80
LWPLS	0.440	0.176	0.934	1.94	0.939	1.83
MWPLS	0.735	0.121	0.960	1.51	0.963	1.44
OSVR	0.742	0.119	0.960	1.51	0.965	1.39
EOSVR (RMSE)	0.863	0.087	0.975	1.21	0.975	1.18
EOSVR (RMSE _{midknn})	0.875	0.083	0.975	1.19	0.976	1.16

Figure 1. Basic concept of optimizing the multiple combinations of SVR hyperparameters. Datasets $(\mathbf{X}_1, \mathbf{y}_1)$, $(\mathbf{X}_2, \mathbf{y}_2)$, ..., $(\mathbf{X}_i, \mathbf{y}_i)$, ..., and $(\mathbf{X}_n, \mathbf{y}_n)$ are used to optimize the SVR hyperparameters $(C_1, \varepsilon_1, \gamma_1)$, $(C_2, \varepsilon_2, \gamma_2)$, ..., $(C_i, \varepsilon_i, \gamma_i)$, ..., and $(C_n, \varepsilon_n, \gamma_n)$, respectively.

Figure 2. Basic concept of prediction using EOSVR. Std. dev. (standard deviation, STD) of $y_{p,1}(t)$, $y_{p,2}(t)$, ..., and $y_{p,m}(t)$ is used to assess the prediction error of $y_p(t)$.

Figure 3. Time plots of $\mathbf{x}_1$, $\mathbf{x}_2$, and $\mathbf{y}$ for the numerical simulation data.

(a) JITPLS

(b) OSVR

(c) EOSVR (RMSE)

(d) EOSVR (RMSE_{midknn})

Figure 4. Relationships between observed and predicted y -values of the test data when the numerical simulation data were generated with Eq. (11).

Figure 5. Relationship between STD and absolute prediction errors in the numerical simulation data.

(a) OSVR
From 11300 to 11510

(b) EOSVR (RMSE)
From 11300 to 11510

(c) OSVR
From 14700 to 14850

(d) EOSVR (RMSE)
From 14700 to 14850

(e) OSVR
From 15950 to 16050

(f) EOSVR (RMSE)
From 15950 to 16050

Figure 6. Time plots of measured and predicted density. Left side (a, c, e) shows results of OSVR and right side (b, d, f) shows results of EOSVR(RMSE).

Figure 7. Relationship between STD and absolute prediction errors in the alkylaluminum production process.

(a) OSVR
From 13850 to 14200

(b) EOSVR (RMSE_{midknn})
From 13850 to 14200

(c) OSVR
From 15600 to 16100

(d) EOSVR (RMSE_{midknn})
From 15600 to 16100

(e) OSVR
From 16600 to 17200

(f) EOSVR (RMSE_{midknn})
From 16600 to 17200

Figure 8. Time plots of measured and predicted denitration outlet NH_3 concentrations. Left side (a, c, e) shows results of OSVR and right side (b, d, f) shows results of EOSVR(RMSE_{midknn}).

(a) Denitration outlet NH_3

(b) Denitration outlet NO_x

(c) Outlet gas duct NO_x

Figure 9. Relationships between STD and absolute prediction errors in the exhaust gas denitration process.