

Adversarial Sample Based Semi-Supervised Learning for Industrial Soft Sensor

Liangjun Feng, Chunhui Zhao*

*State Key Laboratory of Industrial Control Technology,
College of Control Science and Engineering, Zhejiang University, Hangzhou, 310027, China
(Corresponding author, email: chhzhao@zju.edu.cn)*

Abstract: In industrial processes, soft sensor techniques are often utilized to predict the hard-to-measure quality variables. However, the labeled data which are obtained from the offline lab analysis can be quite rare. In the present work, a new divergence-based semi-supervised learning method is developed to exploit the unlabeled samples together with labeled ones for soft sensor application, namely adversarial tri-regression. First, the adversarial samples are generated based on the consideration of maximum disturbance, and through training on the combination of the adversarial samples and the original labeled samples, three regressors are initialized with divergence. Second, for each regressor, an unlabeled sample is labeled when the other two regressors agree on the labeling of this sample, which actually provides that regressor with some unknown information based on the divergence. As the three regressors label more and more samples for each other, the final regression model obtained by averaging the three base regressors presents increasingly more accurate prediction. The proposed method tackles a practical soft sensor problem for the industrial production process of cigarette.

Keywords: Adversarial samples, Divergence-based regressor, Robust modeling, Tri-training strategy, Semi-supervised learning, Soft sensor.

1. INTRODUCTION

Intelligent process monitoring (Song et al. 2019) and advanced control strategies in modern industry process highly depend on the accurate and reliable measurements. However, low reliability of measuring devices and inadequacy of measurement techniques impose massive constraints on the online monitoring of quality variables (Zhao et al. 2014). As a result, these important performance indicators are normally determined by offline laboratory analysis, which requires high-cost maintenance and suffer from low sampling rate (Shardt et al. 2015). In practice, an accurate and reliable online measurement for various quality variables is desired.

Soft sensor, a data-driven virtual sensing technique, predicts hard-to-measure quality variables based on other easy-to-measure process variables to offer economical alternatives to these expensive physical measuring sensors (Zhao et al. 2010). Qin et al. (2019) proposed a slow feature analysis based soft sensor approach to capture the process dynamics. Liu (2014) developed a sparse partial least squares method to simultaneously select the important variables and find the correlation between the variables and targets for soft sensor application. These conventional methods are conducted under the classic supervised learning paradigm, which requires massive labeled samples. However, many processes may only provide a limited number of labeled data due to the restrictions on time, cost, and other resources. In this situation, additional information or data resources may be needed to ensure the soft sensor performance (Shao et al. 2016).

Recently, several soft sensor methods which utilize the unlabeled data have been proposed under the semi-supervised paradigm. Distinguished from both supervised and unsupervised learning methods, semi-supervised learning improves the predictive performance using both the labeled and unlabeled data (Mohamed et al. 2006). Yan et al. (2016) proposed a semi-supervised Gaussian regression based on the Laplace graph regularization item, which is limited to the Gaussian process. Shang et al. (2014) analyzed the benefits of deep neural networks and conducted the semi-supervised soft sensor task using deep learning technique. But the pretraining by unlabeled data can be tedious. Besides, the classic divergence based semi-supervised method, known as Coreg (Zhou et al. 2005a), is also applied in soft sensor task, which trains two K-neighbor nearest regressors and makes them label the unlabeled samples for each other. However, the model requires that the two base regressors should be set by different distance metrics to make them different from each other, which requires tedious and time-consuming trial and error.

Actually, the initial difference between based learners (classifiers or regressors) is necessary and important to launch an effective mutually labeling process for the divergence based semi-supervised methods (Wang et al. 2007, Nigam et al. 2000). Blum et al. (1998) utilized two sufficient and redundant views to make the classifiers different for the classification of web pages. Since the two sufficient and redundant views are usually unavailable for other tasks, the diversity between learners is also achieved through the bootstrap random sampling method (Zhou et al. 2005a) or a fairly reasonable variable split (Blum et al. 1998). However,

it should be noted that the semi-supervised algorithms are usually applied for the cases where limited labeled data are offered, the data or variable sampling methods may make the available samples even less for each base learner. In some variant algorithms, different learning algorithms or initial parameters (Zhou et al. 2005b) are also utilized to make differences, which require for the tedious and time-consuming cross-validation. More effective method should be developed to build different base learners for the divergence based semi-supervised soft sensor.

Besides, another issue considered by soft sensor is the ability of resisting noises. The measurement noises and process noises commonly exist in factories (Khatibisepehr et al. 2008). Popular methods usually adopt extra filtering and smoothing algorithms as a preprocessing step before the model construction and prediction, including Kalman filtering, slow feature extraction, box plot method, linearly weighted moving average, Bayesian methods, etc. (Khatibisepehr et al. 2008, Paulsson et al. 2014) Although results show that these extra filtering algorithms do provide some anti-noise capabilities, the filtering algorithms may also remove some important information hidden in the measurements, which affects the quality prediction. It is better to build a model which can natively resist noises.

In this paper, a divergence-based semi-supervised learning method, termed adversarial tri-regression (ATR), is developed to exploit unlabeled samples together with labeled samples for soft sensor application. Specifically, adversarial tri-regression initializes three base regressors with divergence based on the combination training of the original labeled samples and the generated adversarial samples. Next, unlabeled samples are selected and labeled to augment the training set for the third regressor when the other two regressors agree on the labeling of these samples, which actually offers some unknown information for the third regressor through the divergence. The final soft sensor model is obtained by averaging the three base regressors to achieve the prediction from multiple views. Besides, since the generation of adversarial samples is supervised by labels and based on the consideration of maximum disturbance, hence, the adversarial tri-regression model presents the local smoothing characteristic to resist noises.

The contributions of this paper are summarized as below:

- (1) Adversarial samples are generated to offer an effective way to make divergence among base regressors and make the model robust to the process noises.
- (2) Unlabeled samples can be efficiently labeled and selected to augment the training set for soft sensor task by the mutually labeling process of three base regressors.

The rest of this paper is organized as follows. In the next section, the adversarial tri-regression method, and its application in soft sensors are developed. In Section II, experiments based on a real industrial process are reported. Last, we draw the conclusion.

2. METHODOLOGY

In this section, the meaning of adversarial samples and the

principle of making divergence are presented first, followed by the developing of adversarial tri-regression method for soft sensor modeling.

2.1 Make Divergence with Adversarial Samples

Adversarial tri-regression is designed as a divergence-based semi-supervised learning method to exploit the unlabeled data, in which the divergence among base learners is important and helpful for the performance improvement. Distinguished from the conventional methods (Nigam et al. 2000, Bdelgayed et al. 2018, Zhou et al. 2005b, Zhou et al. 2005a), in this article, the novel adversarial samples are generated and utilized to make divergence among base regressors without imposing other restrictions and make the model more robust to the process noises.

1) The Meaning of Adversarial Samples

Adversarial samples $X_{adv} \in \mathbb{R}^{v \times l}$ (v is the number of variables or measurements and l is the number of samples.) are generated for a regressor $h(\theta)$ based on the labeled training set $\{X_L, Y_L\}$, where $X_L \in \mathbb{R}^{v \times l}$ is the labeled data and $Y_L \in \mathbb{R}^{1 \times l}$ is the label vector. Generally, a regressor h can be trained on the labeled data $\{X_L, Y_L\}$ to obtain the parameter θ and make the output \hat{Y}_L of $h(X_L, \theta)$ as close to Y_L as possible, which can be formulated as

$$\theta = \arg \min_{\theta} D(Y_L, h(\hat{Y}_L | X_L, \theta)) \quad (1)$$

where D is a function which measures the distance between Y_L and \hat{Y}_L . For example, D can be the normal Euclidean distance

$$D(Y_L, \hat{Y}_L) = \|Y_L - \hat{Y}_L\|^2 \quad (2)$$

For the well-trained regressor $h(\theta)$, the adversarial samples X_{adv} are generated to cause maximum error by adding limited perturbations $R_{adv} \in \mathbb{R}^{v \times l}$ to the original samples X_L under the supervised style. The adversarial samples X_{adv} proposed by Goodfellow et al. (2015) and Szegedy et al. (2014) are designed as below

$$X_{adv} = X_L + R_{adv} \quad (3)$$

and

$$R_{adv} = \arg \max_R D(Y_L, h(\hat{Y}_L | X_L + R, \theta)) \quad (4)$$

$$\text{const. } \|R\|_{\ell} < \varepsilon \quad (5)$$

where $\|R\|_{\ell}$ is the ℓ norm of R , and ε is a given parameter to control the intensity of the adversarial perturbation R_{adv} .

The equation (3) shows that the adversarial samples X_{adv} are linearly produced by adding the generated adversarial perturbations R_{adv} to X_L . In equation (4), the labels Y_L of X_L are utilized to supervise the generation process and requires that the adversarial perturbations R_{adv} should cause

maximum error between the output \hat{Y}_L of $h(X_{adv}, \theta)$ and the true label Y_L under the limited noise intensity. While the well-learned parameter θ minimize the loss of h on X_L , the generated X_{adv} aims to maximize the error with only a limited disturbance. Hence, X_{adv} can be aggressive to the well trained regressor $h(\theta)$.

As for the solution, the adversarial samples X_{adv} is designed as a linear sum of the original samples X_L and the adversarial perturbations R_{adv} . Intuitively, it is difficult to obtain a closed-form solution for the exact R_{adv} from (4) and (5). However, based on the linear characteristic, an approximating for R_{adv} is always available. When the ℓ in (5) is adopted as infinite-norm, a very simple approximation for the adversarial perturbation can be obtained

$$R_{adv} \approx \text{sign}(G) \quad (6)$$

where sign is used to judge the symbol of every item of G , and G is the derivative of X_L and can be denoted by

$$G = \nabla_{X_L} D(Y_L, h(\hat{Y}_L | X_L, \theta)) \quad (7)$$

From the solution in (6) and (7), it is observed that the adversarial samples X_{adv} are actually generated by searching for the maximum perturbation direction under the supervision of the label Y_L .

2) How Adversarial Samples Make Effects

As mentioned above, adversarial samples X_{adv} make $h(\theta)$ have the largest loss. However, if we retrain the regressor h with the combination of the original training samples X_L and the adversarial samples X_{adv} to minimize the distance between the output $[\hat{Y}_L, \hat{Y}_{adv}]$ of $h_1([X_L, X_{adv}])$ and $[Y_L, Y_L]$, a very different parameter θ_{adv} and regressor $h_1(\theta_{adv})$ can be obtained, which is formulated as

$$\theta_{adv} = \arg \min_{\theta} D([Y_L, Y_L], h_1([\hat{Y}_L, \hat{Y}_{adv}] | [X_L, X_{adv}], \theta)) \quad (8)$$

Comparing the updated regressor $h_1(\theta_{adv})$ and the initial regressor $h(\theta)$, two main issues can be observed. First, there is significant divergence between θ_{adv} and θ , since the supervised training in (4) makes $h(\theta)$ have a weaker prediction ability for X_{adv} , while the combination training in (8) makes $h_1(\theta_{adv})$ learn the sample pairs (X_{adv}, Y_L) well. But it is worth mentioning that both of $h(\theta)$ and $h_1(\theta_{adv})$ learn from the original data and have a reasonable prediction ability for X_L . Second, the combination training in (8) makes X_{adv} and X_L share the same label Y_L . Since X_{adv} are generated by modifying X_L with limited disturbances R_{adv} , the learning in (8) actually makes the trained regressor $h_1(\theta_{adv})$ smoother at each sample point surrounding X_L , which means $h_1(\theta_{adv})$ output the similar predictions around

each point. Considering that the adversarial samples are designed as the most aggressive noises using the supervised paradigm, the model smoothed by the combination training in (8) can be robust to the common noises and perturbation.

2.2 Adversarial Tri-Regression for Soft Sensor

Basically, adversarial tri-regression builds three regressors with divergence and labels the unlabeled samples for each other to utilize the information of unlabeled data. The implementation of adversarial tri-regression consists of two main steps. First, three different regressors are built based on the combination training of adversarial samples and the original labeled samples. Second, the three initialized regressors iteratively label the unlabeled samples for each other and use the expanded training datasets to update themselves. The final soft sensor model is obtained by averaging the three updated regressors.

1) Build Three Regressors with Divergence

The discussion in the first subsection points out that through the combination training on the original samples and the generated adversarial samples, a robust and different regressor can be obtained. To get three different regressors $\{h_1, h_2, h_3\}$, the generation of adversarial samples and combination training are iteratively conducted three times. The training set of $\{h_1, h_2, h_3\}$ is denoted as $\Psi_i = \{[X_L, X_{adv}^i], [Y_L, Y_L]\}$ ($i=1,2,3$). The adversarial samples X_{adv}^i are obtained by

$$X_{adv}^i = X_L + R_{adv}^i, \quad (i=1,2,3) \quad (10)$$

where

$$R_{adv}^i = \arg \max_R D(Y_L, h_{i-1}(\hat{Y}_L | X_L + R, \theta_{adv}^{i-1})) \quad (11)$$

$$\text{const. } \|R\|_L < \varepsilon \quad (12)$$

and

$$\theta_{adv}^i = \arg \min_{\theta} D([Y_L, Y_L], h_i([\hat{Y}_L, \hat{Y}_{adv}^i] | [X_L, X_{adv}^i], \theta)) \quad (13)$$

Here, $h_0(\theta_{adv}^0) = h(\theta)$ is obtained by (3). After generating the adversarial samples X_{adv}^i , $\{h_1, h_2, h_3\}$ can be easily trained based on Ψ_i ($i=1,2,3$), which makes $h_1(\theta_{adv}^1)$, $h_2(\theta_{adv}^2)$, and $h_3(\theta_{adv}^3)$ different from each other and robust to the common noise.

2) Exploit Unlabeled Samples with Three Regressors

In practical industrial process, there are usually much more unlabeled samples $X_U \in \mathbb{R}^{v \times u}$ saved in the distributed control system. v is the number of variables and u is the number of unlabeled samples. Making good use of these unlabeled samples is of great importance to improve the performance of soft sensor. In the classic Coreg, an unlabeled sample with its pseudo label is added into the training set if the sample pair makes the updated regressor consistent with the original labeled data. However, retraining the model for each unlabeled sample can be extremely time-consuming. Hence,

the proposed adversarial tri-regression utilizes three regressors $\{h_1, h_2, h_3\}$ to simplify the mutually labeling process.

Specifically, adversarial tri-regression utilizes two regressors, h_j and h_k collaboratively label the unlabeled samples for the third regressor h_i ($i, j, k=1,2,3$ and $i \neq j \neq k$) instead of the tedious confidence evaluation of Coreg. For the unlabeled samples X_U, P ($P \ll U$) most confident predictions \hat{Y}_P can be selected from $\hat{Y}_U=1/2(h_j(X_U)+h_k(X_U))$ if the distances between the predictions $h_j(X_U)$ and $h_k(X_U)$ are minimum. The samples which correspond to \hat{Y}_P are denoted as X_P . Add the selected sample pairs (X_P, \hat{Y}_P) into the initial training set Ψ_i , and update the base regressor h_i based on the expanded Ψ_i to learn more information from h_j and h_k . Repeatedly, multiple rounds of labeling process can be conducted for the updated three regressors until reaching the given training rounds T . The final regressor of adversarial tri-regression is determined by averaging the three regressors.

Compared with the evaluation of Coreg, the collaborative labeling by three regressors allows to find a certain number of confident predictions at one time instead of retraining the regressors for each unlabeled sample, which is more efficient. The application of the proposed adversarial tri-regression on soft sensor modeling is summarized in Algorithm 1.

Algorithm 1 Adversarial tri-regression for soft sensor application	
Preprocessing stage	
1	Collect process data and laboratory analysis data from factory
2	Select key variables using theoretical analysis and experience
3	Remove outliers and normalize data, obtain $\Theta=\{X_L, X_U, Y_L\}$.
Model training stage	
4	Select the regression algorithm h
5	Train three regressors $\{h_1, h_2, h_3\}$ using $\{\Psi_1, \Psi_2, \Psi_3\}$
6	For $j=1:T$ do
7	For $i=1:3$ do
8	select p unlabeled samples with pseudo labels to expand Ψ_i ,
9	update h_i using Ψ_i ,
10	remove newly labeled samples from X_U ,
11	End
12	End
13	The final model H is obtained by averaging $\{h_1, h_2, h_3\}$
Model test stage	
14	Preprocess the test data as step 1 to step 3
15	Apply the built soft sensor model H to the test data

3. CASE STUDY

In this section, a soft sensor application is presented for the chemical cigarette manufacturing process of a tobacco corporation based on the proposed adversarial tri-regression method.

3.1 Data Description and Experiment Setting

Generally, the cigarette manufacturing process consists of

three operation stages including leaf processing, silk processing, and blending and spicing. The silk processing which transfers the leaves into leaf-silk is the critical stage.

Table 1.
The structure of designed neural network

Layer	Symbol	Operator	Dimension
0	Input	Input data	20
1	Layer1	Linear mapping	20x512
1	Kernel1	ReLU	512
2	Layer2	Linear mapping	512x256
2	Kernel2	ReLU	256
3	Layer3	Linear mapping	256x128
3	Kernel3	ReLU	128
4	Layer4	Linear mapping	128x1
4	Kernel4	Sigmoid	1

During the transformation, the SIROX warming and dampening machine, and KLD drier machine, are instrumented to reduce the moisture content of leaf-silk from 20% to 18% and from 18% to 12% using the heated barrel with saturated steam. Since the moisture content directly determines the flavor and style characteristic of cigarette, it is an important quality index in the cigarette production.

Twenty process variables, including the flow rate, vapor pressure, and etc., are online measured as input variables to predict the three moisture content related quality variables. As a very limited number of labeled samples are available, only 100 labeled samples and 360 unlabeled samples are used to build the soft sensor model. 150 samples are set as the test data.

The proposed adversarial tri-regression is compared with the classic semi-supervised Coreg method (Zhou et al. 2005a) and the semi-supervised deep believe network (SSDBN) (Shang et al. 2014). For the Coreg method, the distance metric parameters of the two k-neighbor regressors of Coreg are searched from 2 to 5, the number of neighbors is searched from 3 to 9, the maximum iteration is set as 120, and the pool

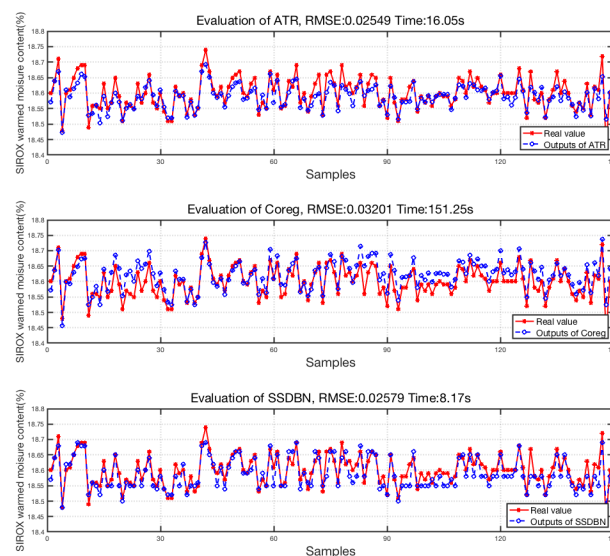


Fig. 1. The comparison of prediction results for the SIROX warmed moisture content in cigarette manufacturing process.

size is set as the number of the unlabeled samples. SSDBN is proposed by Shang et al. and consists of two training stages.

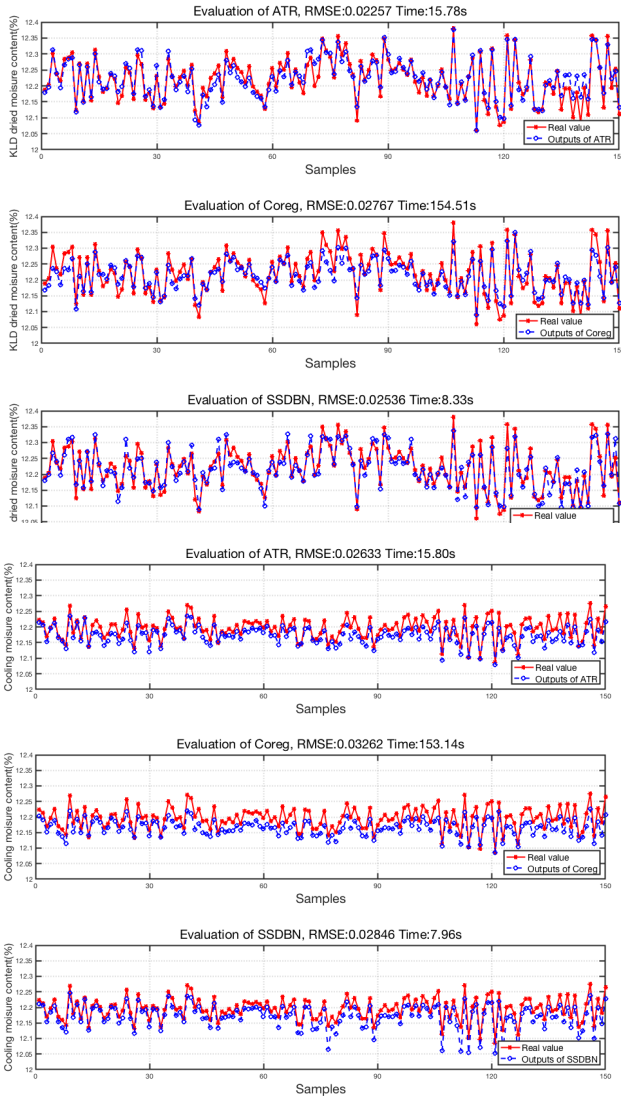


Fig. 3. The comparison of prediction results for the cooling moisture content in cigarette manufacturing process.

The structure of the SSDBN is determined by trial and error, which is shown in Table 1. For the proposed adversarial tri-regression, the rounds T is set as 5, the number of labeling samples for each regressor at each round p is set as 20, and the intensity of adversarial samples ϵ is set as 0.2. The parameters are usually selected by cross validation and will be studied shortly. Here, the neural network shown in Table 1 is utilized as the base learner of the proposed model and all of the base regressors have the same structures. The neural network is trained with the Adam optimizer, 20 rounds training are conducted, and the learning rate is set as 0.001.

3.2 Results and Parameter Study

The RMSE and training time are utilized to evaluate and compare the modeling performance. The results of SSDBN, Coreg, and adversarial tri-regression (ATR) on the three quality variables are comparatively presented in Figure 1 to Figure 3. It is observed that adversarial tri-regression usually has less RMSE than the Coreg method and SSDBN. Although the SSDBN takes less training time than the divergence-based semi-supervised methods, the training time

of adversarial tri-regression has been reduced significantly than that of Coreg, since it avoids evaluating the labeling confidence one by one.

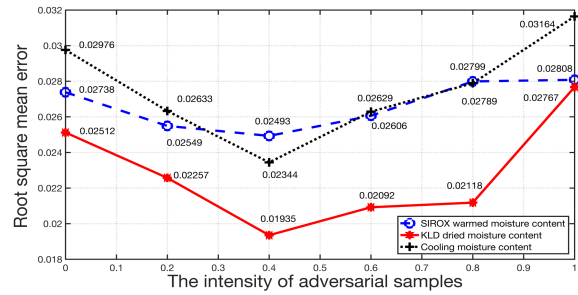


Fig. 4. The effects of the intensity of adversarial samples

Next, the parameter studies are conducted for the adversarial tri-regression method. The effects of the intensity of adversarial samples ϵ and the training rounds T are explored on the three quality variables. For the intensity of adversarial samples ϵ , we adjust it from 0 to 1. Other parameters remain the same as the settings mentioned above. The variation of RMSE with the increasing of ϵ is drawn in Figure 4. Usually, setting ϵ as 0.2 to 0.6 is helpful to obtain a satisfactory result. As for the effects of the training rounds T , it actually decides the number of unlabeled samples which are desired to be labeled. 10 rounds of training are conducted, and 10 unlabeled samples are labeled for each regressors at each round. The variation of RMSE is presented in Figure 5. In the first four rounds, the RMSE is reduced, and later, the performance is gradually steady. Generally, the number of proper training rounds can be 4 to 6 which are helpful to obtain a compromise between the performance and training time.

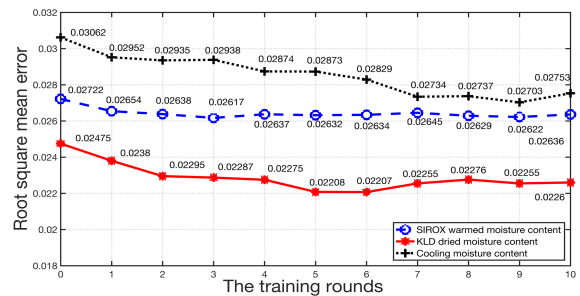


Fig. 5. The effects of training rounds for labeling unlabeled data

3.3 Test for the De-Nosing Ability

Last, the capability of resisting noises of adversarial tri-regression is presented. Adversarial tri-regression is compared with SSDBN and Coreg. In this experiment, the input measurements are added with noises. The noises are generated based on Gaussian distribution. Both of the mean value of noises and variance are 0.2. For a better comparison, the moving average (MA) filter which can smoothen the input measurements is applied as the filtering algorithm before the training of SSDBN and Coreg. The moving window of the moving average filter is set as 3. The parameter settings of SSDBN, Coreg, and adversarial tri-regression are the same as mentioned before. The results are comparatively presented in Table 2. As the table shows, the regular SSDBN and Coreg have little abilities to resist noises,

Table 2.
The prediction results on test data with noises of three quality variables

Performance(RMSE)		SSDBN	MA+SSDBN	Coreg	MA+Coreg	ATR
SIROX warmed moisture content	Clear	0.02579	0.04692	0.033201	0.04692	0.02549
	Noise	0.02989	0.04810	0.03415	0.04819	0.02574
	Error	0.00410	0.00118	0.00214	0.00127	0.00025
KLD dried moisture content	Clear	0.02536	0.05552	0.02767	0.06054	0.02257
	Noise	0.03297	0.05642	0.03003	0.06332	0.02613
	Error	0.00761	0.00090	0.00263	0.00278	0.00356
Cooling moisture content	Clear	0.02846	0.04081	0.03262	0.04102	0.02633
	Noise	0.03637	0.04021	0.03886	0.04601	0.02898
	Error	0.00791	0.00060	0.00664	0.00499	0.00265

“Clear” denotes the input measurements are original clear data, “Noise” denotes the input measurements are combined with noises, “Error” denotes the error caused by the added noises.

and the added Gaussian noises usually make the RMSE larger (about 0.004 to 0.007). The extra moving average filter indeed helps the soft sensor to resist noise, the increase of RMSE is only around from 0.001 to 0.002. However, the utilization of extra filter affects the performance of the soft sensor itself, which means that the filter not only removes the noise but also removes some important information from the measurements.

4. CONCLUSION

This paper presents a new semi-supervised learning algorithm, i.e., adversarial tri-regression, for the industrial soft sensor development. Distinguished from the conventional divergence based semi-supervised learning algorithms, the proposed model generates the novel adversarial samples to make three base regressors different from each other and exploit the unlabeled samples to offer more process information. By generating the adversarial samples, all of the labeled samples can be available for the training of each base regressors instead of the subsets, and the trained regressors are robust to the process noises, which are important for the practical application. The accurate and robust performance of adversarial tri-regression is validated on a real industrial process.

ACKNOWLEDGMENT

This work is supported by NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization (No. U1709211), and Zhejiang Key Research and Development Project (2019C03100 and 2019C01048)

REFERENCES

Song B., Zhou X.G., Liu H.B., and Tao Y., et al., (2019). Performance indicator oriented concurrent subspace process monitoring method. *IEEE Transactions on Industrial Electronics*, 66, 5535-5545.

Zhao C.H., (2014). A quality-relevant sequential phase partition approach for regression modeling and quality prediction analysis in manufacturing processes. *IEEE Transactions on Automation Science and Engineering*, 11, 983-991.

Shardt Y.A.W., Hao H., and Ding S.X., (2015). A new soft-sensor-based process monitoring scheme incorporating infrequent KPI measurements. *IEEE Transactions on Industrial Electronics*, 62,3843-3851.

Zhao C.H., Wang F., and Mao Z., et al., (2010). Quality prediction based on phase-specific average trajectory for batch processes. *AIChE Journal*, 54, 693-705.

Qin Y., Zhao C.H., and Huang B., (2019). A new soft-sensor

algorithm with concurrent consideration of slowness and quality Interpretation for dynamic process. *Chemical Engineering Science*, 199, 28-39.

Liu J.L., (2014). Developing a soft sensor based on sparse partial least squares with variable selection. *Journal of Process Control*, 24, 1046-1056.

Shao W., Tian X., (2016). Semi-supervised selective ensemble learning based on distance to model for nonlinear soft sensor development. *Neurocomputing*, 222, 91-104.

Mohamed F.A.H., and Friedhelm S., (2006). Semi-supervised learning. *Journal of Royal Statistical Society*, 172, 530-530.

Yan W., Guo P., and Yu T., et al., (2016) A framework and modeling method of data-driven soft sensors based on semi-supervised Gaussian regression. *Industrial & Engineering Chemistry Research*, 55.

Shang C., Yang F., and Huang B., et al., (2014), Data-driven soft sensor development based on deep learning technique, *Journal of Process Control*, 24, 223-233.

Zhou Z.H., and Li M., (2005a). Semi-supervised regression with co-training. in *Proceeding 19th International Joint Conference on Artificial Intelligence*, 908-913.

Wang W., and Zhou Z.H., (2007). Analyzing co-training style algorithms. in *Proceeding European Conference Machine Learning*, 454-465.

Nigam K. and Ghani R., (2000). Analyzing the effectiveness and applicability of co-training. in *Proceeding International Conference on Information Knowledge Management*, 86-93.

Blum A., and Mitchell T., (1998). Combining labeled and unlabeled data with co-training. in *Proceeding Conference Computational Learning Theory*, 92-100.

Zhou Z.H., and Li M., (2005b). Tri-training: exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering*, 17, 1529-1541.

Khatibisepehr S., and Huang B., (2008). Dealing with irregular data in soft sensors: Bayesian method and comparative study, *Industrial & Engineering Chemistry Research*, 47, 8713-8723.

Paulsson D., Gustavsson R., and Mandenius C.F., (2014). A soft sensor for bioprocess control based on sequential filtering of metabolic heat signals. *Sensors*, 14, 17864-17882.

Goodfellow I.J., Shlens J., and Szegedy C., (2015). Explaining and harnessing adversarial examples. in *Proceeding International Conference Learning Representation (ICLR)*.

Szegedy C., and Goodfellow I.J., et al., (2014). Intriguing properties of neural networks, in *Proceeding International Conference Learning Representation*.

Bdelgayed T.S., Morsi W.G., and Sidhu T.S., (2018). Fault detection and classification based on co-training of semi-supervised machine learning. *IEEE Transactions on Industrial Electronics*, 65, 1595-1605.