Applied Soft Computing Journal 81 (2019) 105526

Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

Design teacher and supervised dual stacked auto-encoders for quality-relevant fault detection in industrial process

Shifu Yan, Xuefeng Yan *

Key Laboratory of Advanced Control and Optimization for Chemical Process of Ministry of Education, East China University of Science and Technology, Shanghai 200237, PR China

HIGHLIGHTS

- Quality variables are considered in TSSAE.
- Feature extraction and model construction are implemented separately.
- Features are compact without much redundancy based on LWPD.

ARTICLE INFO

Article history:
Received 23 January 2019
Received in revised form 14 May 2019
Accepted 21 May 2019
Available online 24 May 2019

Keywords:
Deep neural network
Fault detection
Quality-relevant
Stacked auto-encoder

ABSTRACT

Current fault detection methods based on deep neural networks only consider process information and ignore quality indicators. In order to obtain features representing both process variables and quality indicators efficiently, this paper designs teacher and supervise dual stacked auto-encoder (TSSAE) for quality-relevant fault detection in industrial process which separates the feature extraction and model construction. To separate the feature extraction and model construction, a mixing stacked auto-encoder which consists of a nonlinear encoder and a linear decoder is designed to extract features of process variables and quality indicators. Another encoder is supervised by the extracted features and further predict the process variables and quality indicators only from process variables. Then quality-relevant, quality-irrelevant and residual subspaces are constructed in a linear way and fault detection is implemented in these subspaces based on Euclidean distance and kernel density estimation. Finally, the effectiveness of TSSAE is evaluated by a numerical example and the Tennessee-Eastman process.

1. Introduction

The great advancement of automation technology enables modern industrial processes larger and more integrated. In this case, process monitoring including fault detection is crucial for personal safety and economic benefit [1,2]. As one of the most widely used methods, principal component analysis (PCA) can reduce data dimension and divide process space into two orthogonal subspaces [3]. After that, statistics like Hotelling's T^2 or square prediction error (SPE) are calculated to detect faults in industrial processes. Considering the inherent nonlinearity and dynamic property in large-scale industrial process, traditional PCA has been extended to multi-versions, such as kernel PCA (KPCA), dynamic PCA and distributed PCA [4–6].

Given the market demand for high value-added products, the quality-relevant fault detection aims at uncovering the effects on such quality indicators has received extensive attentions in recent decades [7]. Since the measurement of quality indicators is expensive and comes with a time delay, a common solution is to predict them by establishing a regression model between process variables and quality indicators [8–10]. Wang et al. presented principal component regression (PCR) and kernel PCR (KPCR) based on least squares [8]. Zhou et al. developed a total projection to latent structures (TPLS) algorithm that further decomposes the two subspaces derived by partial least squares (PLS) [9]. Yin et al. used orthogonal signal correction (OSC) to extract information that is orthogonal to the quality indicators, and thus acquire quality-relevant and quality-irrelevant parts [10]. Similarly, kernel and dynamic tricks are also combined with these approaches to handle more complex issues [11,12]. As the amount of data continues to increase, the problem with data rich while information poor has been prominent and the simple regression models may not meet the accurate prediction of quality indicators.

With the emergence of deep neural networks (DNN), feature extracting and representation learning have been greatly improved owing to the randomness of structures or weights [13]. Considering the universality of unlabeled data, unsupervised learning frameworks like stacked auto-encoder (SAE) and deep

* Correspondence to: P.O. BOX 293, East China University of Science and Technology, MeiLong Road NO. 130, Shanghai 200237, PR China.
E-mail address: xfyan@ecust.edu.cn (X. Yan).

https://doi.org/10.1016/j.asoc.2019.105526
1568-4946/

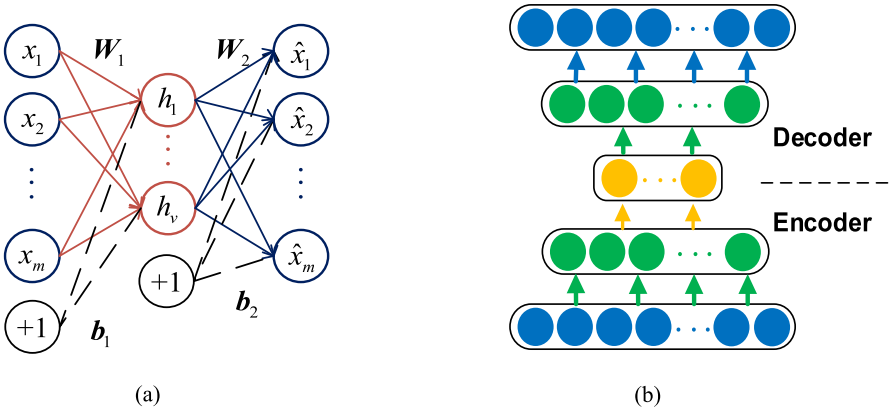

Fig. 1. Unsupervised frameworks. (a) An auto-encoder with a single hidden layer. (b) Stacked auto-encoder.

belief neural network (DBN) are more efficient in industrial processes [14]. Based on the complexity and capacity of DNN, compact and representative features of process variables can be obtained which lead to excellent performance in fault detection and diagnosis. Zhang et al. applied DBN to the complex chemical process and Yu et al. introduce the layer-by-layer enhancement strategy for better monitoring performance [15,16]. Zhang et al. and Lv et al. respectively proposed SAE-based methods for process monitoring [17,18]. In addition, Wen et al. transformed signals into two-dimensional images and convolutional neural network is trained to diagnosis the faults [19]. While the current DNN-based approaches mainly concentrate on process variables and quality indicators are not considered which limits the applications in quality-relevant fault detection.

The key to the better monitoring performance of quality-relevant fault detection is the feature extraction of both process and quality variables. Traditional methods mainly focused on one part of them or separated these two parts which may degrade the monitoring and generalized performance. Thus the efficient information for further division in both process and quality variables needed to be extracted at the same time. Considering the combination of supervised learning and unsupervised learning, this study designs teacher and supervise dual auto-encoder (TSSAE) containing dual neural networks to extract and represent features of process variables including quality indicators. In TSSAE, feature extraction and model construction are realized separately. A mixing SAE (teacher model, T) consisting of a nonlinear encoder and a linear decoder is firstly designed to extract the efficient features. The structures of T are determined by a layer-wise performance-driven strategy (LWPD). After features extracted, another encoder (student model, S) similar to the encoder of T is supervised by the features during training and quality information of the input is absent in S. In this way, process variables and quality indicators can be predicted in a linear combination of features and thus quality-relevant, quality-irrelevant and residual subspace are formed. In each subspace, statistics are constructed based on the Euclidean distance and corresponding probability density functions (pdf) in normal status are estimated by kernel density estimation (KDE). Then, faulty information in industrial processes can be detected. Compared with other methods, TSSAE may possess the following advantages:

(1) In TSSAE, feature extraction and model construction are implemented by two neural networks, respectively. Based on the advantages of supervised and unsupervised learning, process variables and quality indicators can be better represented and the generalized performance of predicting the quality indicators can be improved.
(2) Based on the layer-wise compression, the features extracted by such SAE is compact without much redundant information and enough complex to represent the process variables and quality indicators simultaneously.
(3) The combination of linear and nonlinear structure and the LWPD strategy are convenient for us to construct the models and process information.

The rest of the paper is arranged as follows. SAE and KDE are briefly reviewed in Section 2. The detail steps of model construction and fault detection scheme are described in Section 3. The proposed scheme is tested by a numerical example and the Tennessee-Eastman process (TEP) in Section 4. Finally, conclusions are presented in Section 5.

2. Background

2.1. Stacked auto-encoder

An auto-encoder (AE) is a feedforward neural network with the same input and output shown in Fig. 1(a). Under the unsupervised-learning mechanism, the AE implements feature extraction by minimizing the mean square error (MSE) between the input and output variables. The model can be described as follows:

$$\begin{cases} \boldsymbol{h} = f_1(\boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1) \\ \hat{\boldsymbol{x}} = f_2(\boldsymbol{W}_2\boldsymbol{h} + \boldsymbol{b}_2) \end{cases}, \tag{1}$$

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_m]^T \in \boldsymbol{R}^m$, $\hat{\boldsymbol{x}} = [\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_m] \in \boldsymbol{R}^m$, $\boldsymbol{h} = [h_1, h_2, \ldots, h_v] \in \boldsymbol{R}^v$ are the input, output, hidden variables with m, m, v dimensions, respectively. $\boldsymbol{W}_1 \in \boldsymbol{R}^{v\times m}$, $\boldsymbol{b}_1 \in \boldsymbol{R}^v$ and $\boldsymbol{W}_2 \in \boldsymbol{R}^{m\times v}$, $\boldsymbol{b}_2 \in \boldsymbol{R}^m$ are weights and biases of the hidden and output layers, respectively. Activation function $f(\cdot)$ can be sigmoid, tanh, rectified linear unit.

During the training phase, backpropagation algorithm and its variants are widely used. In addition, traditional AEs can be transformed into sparse, denoising AEs by regularizing the parameters or structures and so on [14]. Considering the advantages of deep learning in feature extraction and representation learning, the SAE with multilayer encoder and decoder is stacked by several AEs shown in Fig. 1(b). Therefore, pre-trained methods are adopted due to the non-convexity of DNNs, which aid in searching for global optimums.

2.2. Kernel density estimation

As a nonparametric method, KDE is used to estimate the pdfs of unknown variables. Supposing that the pdf of variable z is

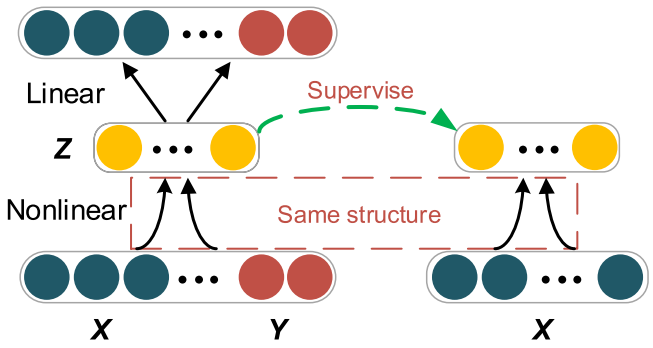

Fig. 2. Structure of TSSAE.

$p(z)$ and r samples $z_1, z_2, \ldots, z_r$ are collected. Thus, $p(z)$ can be estimated based on the Gaussian kernel function in (2) or other available kernel functions.

$$\hat{p}(z) = \frac{1}{rd}\sum_{i=1}^{r}\exp\left(-\frac{1}{2}\left(\frac{z-z_i}{d}\right)^2\right), \tag{2}$$

where d is the smoothing factor. For fault detection, the pdf estimated by KDE is usually calculated to determine the thresholds of statistics under a certain level of confidence [20].

3. Scheme of fault detection

3.1. TSSAE

Fig. 2 shows the structure of TSSAE including two neural networks. On the left side, the teacher model which is an SAE, is trained to compress redundant information and extract the efficient features of process variables and quality indicators. To represent the input data by linear combination of extracted features, the activation functions of encoder and decoder in T are nonlinear and linear, respectively.

$$\begin{cases} f_{encoder}(x) = \frac{1-\exp(-x)}{1+\exp(-x)} \\ f_{decoder}(x) = x \end{cases}, \tag{3}$$

where x is the input of corresponding hidden neurons. Before training, m process variables $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m] \in R^{n\times m}$ and p quality indicators $\boldsymbol{Y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_p] \in R^{n\times p}$ with n samples are scaled to zero mean and unit length.

Considering the issues to determine the structure of T, the LWPD strategy is presented. Since the SAE is symmetrical, we only focus on the encoder part. For the purpose of compression, the nodes in each hidden layer are all less than the previous layer. The strategy mainly includes three steps, and a single AE is used in each step. The first AE is used to compress redundant information. This AE is trained with decreasing number of hidden nodes until the MSE increases remarkably. Then, the hidden layer of this AE is used as the first hidden layer of the entire SAE. The features are calculated as the input of second AE after compression of redundant information. At the same time, the number of hidden nodes decreases. To compress information in a lossless manner and to obtain all the PCs that can completely reconstruct the original data, regression coefficient R is calculated between the input and output by Eq. (4).

$$R = \frac{\sum_{i=1}^{number}(x_i-\bar{x})\left(\hat{x}_i-\bar{\hat{x}}\right)}{\sqrt{\sum_{i=1}^{number}(x_i-\bar{x})^2\sum_{i=1}^{number}\left(\hat{x}_i-\bar{\hat{x}}\right)^2}}, \tag{4}$$

where $\bar{x}$ and $\bar{\hat{x}}$ indicate the average value of the input and output, respectively. Here, *number* stands for the number of input samples. R describes the linear correlation between the predicted and original values. When $R \approx 1.00$, we consider the features containing nearly all the required information to reconstruct the input data, and the nodes in the second layer are determined. Therefore, the features become compact after compression by the first and second AEs. To obtain the major variations in feature space, $R \approx 0.99$ which is set according to usage when the number of hidden nodes decreases in the third AE. Subsequent extraction is required when numerous hidden nodes still exist in this step. The AEs selected by the above procedures are stacked into an SAE that is fine-tuned using the original data. In this way, the most representative characteristics of process variables and quality indicators can be obtained.

Generally, quality indicators are absent in the real-time situation. Thus, their features must be learned and represented only by process variables. Suppose features $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_l] \in R^{n\times l}$ with l dimensions are obtained, then the student model which owns the same hidden layer structure of the encoder in T is supervised by $\boldsymbol{Z}$. TSSAE is completed with T and S models.

3.2. Fault detection method

In this section, fault detection method is described. After features $\boldsymbol{z}$ are calculated based on the above dual models, the original process variables and quality indicators can be reconstructed in a linear manner as follows.

$$\begin{cases} \hat{\boldsymbol{y}} = \boldsymbol{W}_y\boldsymbol{z} + \boldsymbol{b}_y \\ \hat{\boldsymbol{x}} = \boldsymbol{W}_x\boldsymbol{z} + \boldsymbol{b}_x \end{cases}. \tag{5}$$

Eq. (5) is implemented by two simple feedforward neural networks in this paper. $\boldsymbol{W}_y, \boldsymbol{W}_x, \boldsymbol{b}_y, \boldsymbol{b}_x$ are the weights and biases. $\boldsymbol{z} \in \Re^l$ is the output of hidden neurons. $\hat{\boldsymbol{x}} \in \Re^m$ and $\hat{\boldsymbol{y}} \in R^p$ are the predicted values of a sample for process variables and quality indicators, respectively.

To construct the statistics for fault detection, traditional PCA is performed on the quality-relevant information to obtain $\hat{\boldsymbol{Y}} = \boldsymbol{T}_y\boldsymbol{Q}_y^T$ with the number of PCs A determined by cumulative percent variance, and thus the quality-relevant scores can be calculated as follows [9]:

$$\boldsymbol{T}_y = \left(\boldsymbol{Z}\cdot\boldsymbol{W}_y^T + 1_n\boldsymbol{b}_y^T\right)\boldsymbol{Q}_y, \tag{6}$$

where $\boldsymbol{Q}_y \in R^{p\times A}$ is the loading matrix. 1_n is a column vector with n dimensions and the values are all ones. Thus, quality-relevant features $\boldsymbol{Z}_y = \boldsymbol{T}_y\boldsymbol{P}_y^T$ and quality-irrelevant features $\boldsymbol{Z}_o = \boldsymbol{Z} - \boldsymbol{Z}_y$ can be obtained where $\boldsymbol{P}_y^T = \left(\boldsymbol{T}_y^T\boldsymbol{T}_y\right)^{-1}\boldsymbol{T}_y^T\boldsymbol{Z}$. Then, quality-irrelevant scores $\boldsymbol{T}_o = \boldsymbol{Z}_o$ are constructed. Notably, these two subspaces $\boldsymbol{Z}_y$ and $\boldsymbol{Z}_o$ are orthogonal to each other, which is proved by (7).

$$\boldsymbol{T}_y^T\boldsymbol{T}_o = \boldsymbol{T}_y^T\left(\boldsymbol{I} - \boldsymbol{T}_y\left(\boldsymbol{T}_y^T\boldsymbol{T}_y\right)^{-1}\boldsymbol{T}_y^T\right)\boldsymbol{Z} = 0. \tag{7}$$

Therefore, quality-relevant, quality-irrelevant and residual subspaces are designed. In each subspace, statistic is constructed based on the Euclidean distance between training and testing samples. When a test sample $\boldsymbol{x}_{new}$ is collected, feature $\boldsymbol{z}_{new}$ is obtained by S model and three statistics are calculated as follows.

Firstly, statistic D_x in the residual subspace is based on the reconstruction error of process variables and calculated as

$$D_x = \left\|\boldsymbol{x}_{new} - \hat{\boldsymbol{x}}_{new}\right\|^2 = \left\|\boldsymbol{x}_{new} - \boldsymbol{W}_x\cdot\boldsymbol{z}_{new} - \boldsymbol{b}_x\right\|^2. \tag{8}$$

And D_x calculated by training samples are used to determine the threshold $J_{x,th}$ by KDE.

Secondly, mean values $\overline{\boldsymbol{T}}_y$, $\overline{\boldsymbol{T}}_o$ of $\boldsymbol{T}_y$, $\boldsymbol{T}_o$ in feature space are calculated as baselines. For training samples, variations in the neighborhood of the baselines are considered to be fault-free and are used to determine the thresholds $J_{y,th}$ and $J_{o,th}$. For a new

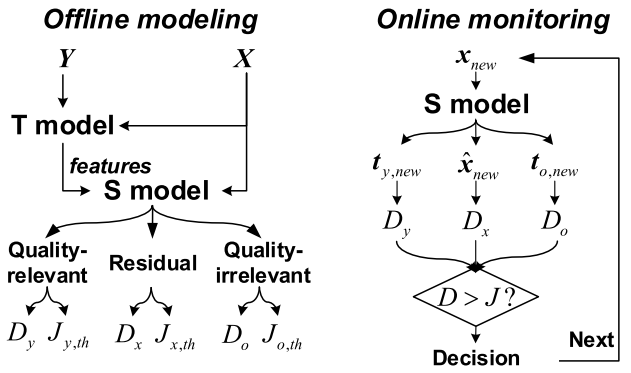

Fig. 3. Flowchart of TSSAE for quality-relevant fault detection.

sample $\boldsymbol{x}_{new}$, quality-relevant and quality-irrelevant scores and corresponding statistics are

$$\begin{cases} \boldsymbol{t}_{y,new}^T = \left(\boldsymbol{W}_y \cdot \boldsymbol{z}_{new} + \boldsymbol{b}_y\right)^T \boldsymbol{Q}_y & \rightarrow D_y = \left\| \boldsymbol{t}_{y,new}^T - \overline{\boldsymbol{T}}_y \right\|^2 \\ \boldsymbol{t}_{o,new}^T = \boldsymbol{z}_{new}^T - \boldsymbol{t}_{y,new}^T \boldsymbol{P}_y^T & \rightarrow D_o = \left\| \boldsymbol{t}_{o,new}^T - \overline{\boldsymbol{T}}_o \right\|^2 \end{cases}. \quad (9)$$

After calculating such statistics, if D_y exceeds $J_{y,th}$, then a quality-relevant fault is considered to be occurred. If D_y does not exceed but D_o or D_x exceed corresponding thresholds, then a quality-irrelevant fault happened.

3.3. Flowchart of process monitoring

The flowchart of fault detection scheme using TSSAE is shown in Fig. 3. The implementation contains two parts: offline modeling and online monitoring. The detail steps are presented as follows.

Offline modeling:

(1) Normalize the process variables $\boldsymbol{X}$ and quality indicators $\boldsymbol{Y}$ to zero mean and unit length;
(2) Use $\boldsymbol{X}$ and $\boldsymbol{Y}$ as the input data of the teacher model (a stacked auto-encoder) decided by LWPD strategy and extract the representative features $\boldsymbol{Z}$;
(3) Supervise the student model to represent $\boldsymbol{Z}$ only from $\boldsymbol{X}$ and predict both process variables and quality indicators;
(4) Form quality-relevant, quality-irrelevant and residual subspaces;
(5) Calculate three thresholds of the statistics in corresponding subspaces.

Online monitoring:

(1) Normalize a new sample $\boldsymbol{x}_{new}$ and get features $\boldsymbol{z}_{new}$;
(2) Calculate the statistics D_y, D_o and D_x in each subspace;
(3) Generate the faulty information.

Based on the ability of DNNs in representation learning, both process variables and quality indicators can be predicted by the efficient features. Thus TSSAE can provide comprehensive and reliable information for operators about specific processes.

4. CaSe study

To evaluate the performance of TSSAE, false alarm rate (FAR) and fault detection rate (FDR) are used and defined as

$$\begin{cases} \text{FAR} = \dfrac{\text{Number of false alarms}}{\text{Number of fault-free samples}} \\ \text{FDR} = \dfrac{\text{Number of alarms}}{\text{Number of faulty samples}} \end{cases}. \quad (10)$$

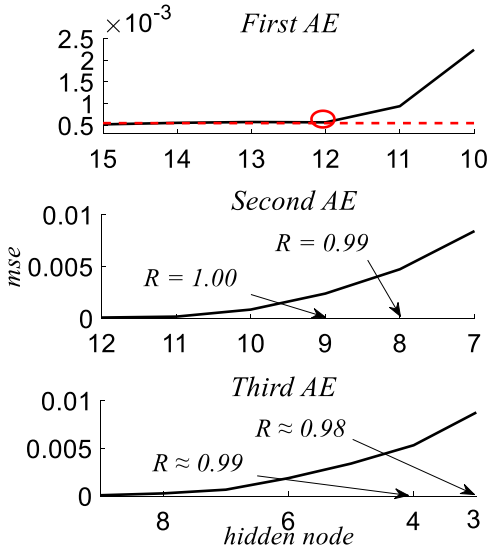

Fig. 4. MSE of three AEs by the LWPD strategy for numerical example.

4.1. Numerical example

To illustrate the performance of the proposed method, a numerical example is designed as follows. Variables of this example are formed by basic components and redundant information or nonlinearities are introduced in this way. The quality variable is related to some of them to test the ability in feature extractions.

$$\begin{array}{lll} x_1 = s_1 + e_1 & x_2 = s_1 + e_2 & x_3 = 2s_1 + e_3 \\ x_4 = s_1 - s_2 + e_4 & x_5 = s_1 + s_2 + e_5 & x_6 = s_3^2 + e_6 \\ x_7 = s_3 \times s_4 + e_7 & x_8 = s_4 + e_8 & x_9 = s_4 + s_5^2 + e_9 \\ x_{10} = s_6^2 + s_7^2 + e_{10} & x_{11} = s_3 \times s_8 + e_{11} & x_{12} = s_7 - s_8 + e_{12} \\ x_{13} = s_7 - s_8 + e_{13} & x_{14} = s_7^3 - s_8^3 + e_{14} & x_{15} = s_7^3 - s_8^3 + e_{15} \\ y = x_1 \times x_8 + x_9 + e_{16} & & \end{array}, \quad (11)$$

where s_i $(i = 1 \text{ to } 8)$ follow the uniform distribution from 0 to 1 consisting of eight principal components that form the example. e_j $(j = 1, 2, \ldots, 16)$ indicates the Gaussian noise with zero mean and 0.01 standard deviation to be more realistic. Besides, 15 process variables and a quality indicator are included. Then, 1000 samples under normal condition are used to construct models. In this case, we generate another 2 fault data sets, i.e. fault 1 and fault 2, by adding a step-fault of 1.5 to interfere x_5, x_{10} and x_8, x_{13}, respectively.

To determine the structure of T, three AEs are trained as shown in Fig. 4. In the first one, the MSE does not change significantly when the number of nodes in hidden layer is 15, 14, 13 and 12. Thus the nodes of first hidden layer is set to 12. Then features are extracted for the next AE. In the second AE, we get $R \approx 0.99$ until the number of nodes drops to 8 and 9 is chosen as the size of second hidden layer. Similarly, 4 is set for the third hidden layer. Finally, 16-12-9-4-9-12-16 and 15-12-9-4 are decided for T and S in this numerical example.

In order to avoid the adverse effects of data fluctuations and noise, statistics like D_y, D_o and D_x are processed using the moving average involving 4 samples simultaneously in this paper. The FARs and FDRs (%) by TSSAE and PCR are calculated in Table 1. In PCR, T_y^2 and T_o^2 stand for quality-relevant and quality-irrelevant statistics. Obviously, two faults are detected successfully, while TSSAE exhibits a better performance in quality-relevant statistic. To be more specific, fault 2 is quality-relevant and the monitoring results of fault 2 are shown in Figs. 5 and 6. The more efficient

Table 1
FARs / FDRs (%) for numerical example by TSSAE and PCR.

Fault No.	TSSAE			PCR	
	D_y	D_o	D_x	T_y^2	T_o^2
1	1.0 / 1.1	1.0 / 91.6	0.5 / 100	0.0 / 0.5	2.0 / 100
2	1.0 / 87.0	0.5 / 99.8	0.5 / 100	0.0 / 26.5	0.0 / 100

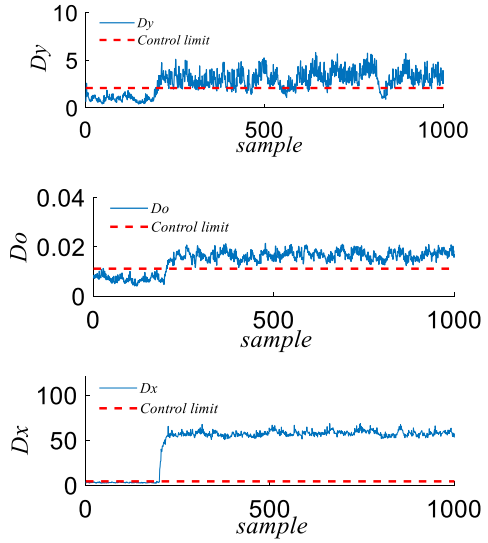

Fig. 5. Fault detection results of fault 2 in numerical example by TSSAE.

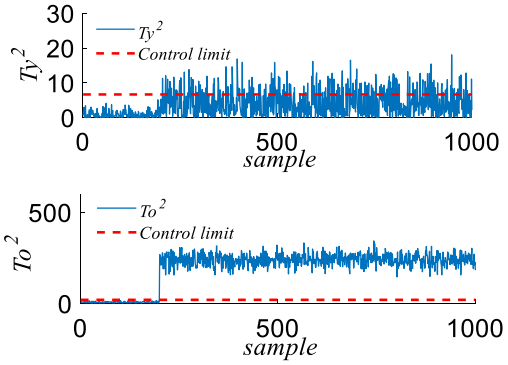

Fig. 6. Fault detection results of fault 2 in numerical example by PCR.

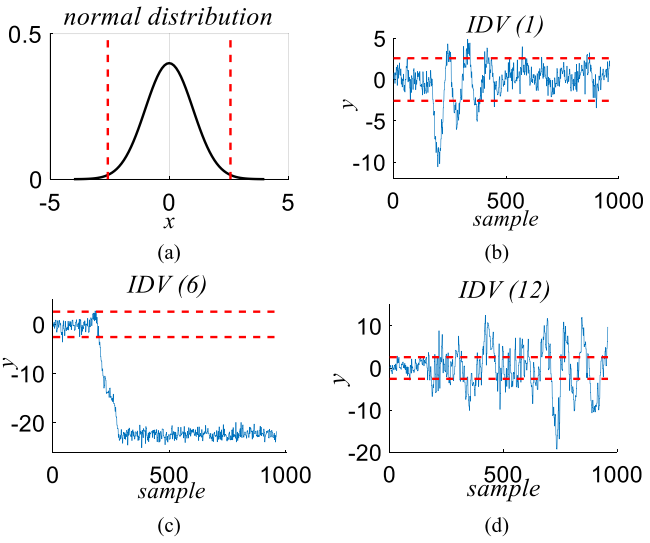

Fig. 7. (a) Probability density curve of normal distribution. (b-d) Real values of quality indicator in IDV 1, IDV 6 and IDV 12.

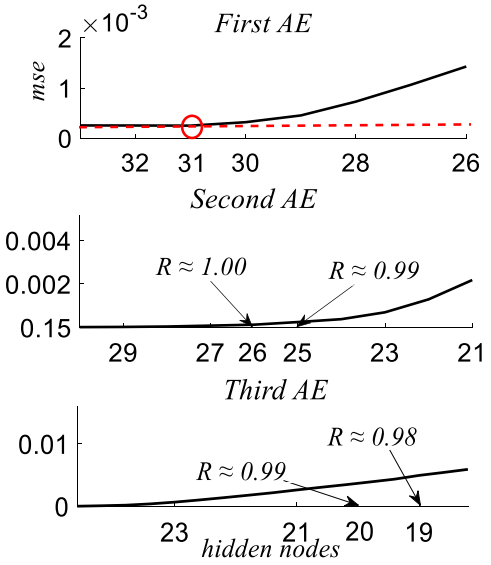

Fig. 8. MSE of three AEs by the LWPD strategy for TEP.

quality-relevant information has been obtained by the statistic D_y since it provides the higher FDR than T_y^2 of PCR.

4.2. Tennessee-Eastman Process

The TEP described by Downs and Vogel is highly nonlinear and coupled, which is most widely used as a benchmark in process monitoring [21]. Four reactants (A, C, D and E) and two products (G and H) are involved together with 41 measured variables and 12 manipulated variables. In this paper, 22 measured variables XMEAS (1–22) and 11 manipulated variables XMV (1–11) are selected. For ease of description, process variables and quality indicator are indicated as $x_1, x_2, \ldots, x_{33}, y$. Notably, the sampling time of the above 33 variables are 3 min while the quality indicator is sampled every 6 min because of time delay. In order to be continuous, the unmeasured value of y is estimated by the average of previous and next samples. The data used in this paper include 22 sets. Among them, the training one is sampled under normal condition. The test data sets stand for 21 faults i.e., IDV 1–21 in TEP and each fault consists of 160 fault-free and 800 faulty samples.

As a reference, y is assuming to be normal distributed after being scaled, and thresholds are set under confidence level $\alpha = 0.99$ in Fig. 7(a). Thus 21 faults can be divided into quality-relevant and quality-irrelevant types. Furthermore, quality-relevant ones present three states. First one like IDV 1, 5, 7 recovered normal within a certain time due to the closed-loop control technology. The second one includes IDV 2, 6, 18, 21 indicating those unable to eliminate the negative effects. The third one presented a loose oscillation involving IDV 8, 12, 13. For more intuitively, IDV 1, 4, 12 are illustrated in Fig. 7(b–d). According to the LWPD strategy, the results for TEP are shown in Fig. 8. Therefore, the structures of T and S models are 34-31-26-20-26-31-34 and 33-31-26-20, respectively.

The FDRs of statistic D_y in quality-relevant and quality-irrelevant faults shown in Table 2. Here we compare TSSAE with

Table 2
FDRs (%) of quality-relevant statistics of TSSAE, TPLS, PCR, and KPCR for TEP.

Fault No.	TSSAE	TPLS	PCR	MKLS	KPCR	Fault No.	TSSAE	TPLS	PCR	MKLS	KPCR
	D_y	T_y^2	T_y^2	T_y^2	T_y^2		D_y	T_y^2	T_y^2	T_y^2	T_y^2
3	2.1	**0.2**	1.3	5.77	1.2	1	25.8	29.2	29.3	**53.77**	52.5
4	**0.4**	3.9	0.8	6.02	0.8	5	17.4	15.8	13.9	**30.36**	17.0
9	4.1	1.0	**0.6**	5.02	0.8	7	58.1	**61.9**	26.4	39.49	29.9
10	33.0	26.3	**18.4**	48.86	20.7	2	75.5	88.4	**95.6**	88.37	62.9
11	3.4	4.2	3.3	9.66	**2.5**	6	**98.6**	96.7	97.3	47.82	97.8
14	**0.0**	0.4	2.0	3.39	**0.0**	18	86.9	86.8	86.6	34.47	**87.9**
15	11.1	3.9	**2.6**	12.67	4.3	21	**55.4**	50.2	23.3	30.61	19.2
16	28.4	13.2	**10.1**	20.08	10.9	8	85.1	**85.9**	74.9	77.04	82.2
17	**16.0**	40.5	31.4	13.3	17.3	12	68.9	76.8	70.3	76.29	**79.5**
19	0.6	**0.0**	0.3	1.03	0.0	13	**88.1**	86.0	73.8	79.42	83.0
20	11.8	**9.9**	14.3	28.48	25.7						

Table 3
FDRs (%) of quality-irrelevant statistics of TSSAE, PCA, ICA, KPCA, AE, and DBN for TEP.

Fault No.	TSSAE	PCA	ICA	KPCA	AE	NCA	DBN
	D_o/D_x	T^2/SPE	I^2/SPE	T^2/SPE	T^2/SPE	T^2/SPE	SPE
1	99.5 / 99.8	99.2 / 99.7	99.7 / 99.7	99.8 / 0.01	99.3 / 99.9	99.5 / 99.3	**100**
2	98.6 / 98.5	98.0 / 98.6	**98.9** 98.6	98.6 / 2.9	97.9 / 98.1	98.3 / 98.5	98.0
3	1.9 / 11.8	0.2 / 0.9	2.7 / 7.7	11.9/ **15.0**	0.0 / 5.6	1.6 / 2.1	3.0
4	99.8 /**100**	54.4 / 96.2	77.5 / 88.7	99.9 / 94.0	97.8/ **100**	36.7 / 94.3	98.0
5	31.4 /**100**	22.5 / 25.4	**100** / **100**	34.0 / 23.5	38.9 / **100**	25.6 / 28.9	99.0
6	**100** / **100**	99.2 /**100**	**100** / **100**	**100** / 0.5	98.8 / **100**	99.5 / 99.1	**100**
7	**100** / **100**	**100** / **100**	**100** / **100**	**100** / 79.1	**100** / **100**	**100** / **100**	**100**
8	96.0 / 98.3	97.5 / 97.5	96.9 / 98.7	**99.4** / 36.4	97.2/ 98.3	97.3 / 97.5	98.0
9	4.5 /**13.8**	0.6 / 1.9	1.1 / 3.1	10.3 / 11.5	1.5 / 5.3	2.1 / 6.0	3.0
10	63.5 / **94.4**	33.4 / 34.1	74.6 / 75.5	52.8 / 67.1	43.8 / 86.5	33.7 / 27.2	55.0
11	93.3 / **99.4**	20.6 / 64.4	69.7 / 55.4	74.8 / 72.6	82.9 / 98.9	48.0 / 69.9	65.0
12	99.3 / **99.8**	97.1 / 97.5	99.7 / 99.7	99.5 / 40.1	99.5 / **99.8**	99.0 / 94.5	99.0
13	94.3 / **95.8**	94.0 / 95.5	95.4 / 95.5	95.1 / 14.6	94.0 / 96.1	94.9 / 95.1	94.0
14	**100** / **100**	**100** / 98.9	**100** / 99.9	**100** / 45.4	99.8/ 99.9	99.6 /**100**	**100**
15	0.9 / 15.2	1.2 / 2.7	4.0 / 15.2	**19.8** / 18.5	1.2 / 13.0	2.5 / 9.1	4.0
16	33.5 / **97.1**	15.7 / 24.5	80.4 / 74.9	39.6 / 15.8	22.6 / 90.8	14.4 / 23.4	6.0
17	95.8 / **97.8**	74.1 / 89.2	87.9 / 90.4	95.1 / 38.3	94.0 / **97.8**	84.6 / 46.7	94.0
18	89.0 / **91.4**	88.7 / 89.9	90.7 / 89.6	90.8 / 3.4	88.5 / 90.6	90.1 / 87.9	90.0
19	29.3 / 99.5	13.9 / 28.0	64.8 / 37.4	11.8 / 24.3	11.3 / **99.8**	0.0 / 0.6	45.0
20	75.3 / **91.1**	31.5 / 60.2	77.9 / 74.2	64.0 / 56.0	58.8 / 90.6	30.9 / 48.0	56.0
21	43.8 / **59.1**	26.4 / 43.0	42.5 / 49.0	43.4 / 48.0	40.1 / 56.6	35.2 / 28.0	48.0

Table 4
FARs (%) of quality-irrelevant statistics of TSSAE, PCA, ICA, KPCA, AE, and DBN for TEP.

Algorithm	TSSAE	PCA	ICA	AE	DBN
Statistics	D_o/D_x	T^2/SPE	I^2/SPE	T^2/SPE	SPE
Average FARs	1.1 / 4.3	1.3 / 8.2	2.5 / 2.4	0.2 / 2.3	2.2

TPLS, PCR, KPCR and modified kernel least squares (MKLS) [22,23]. For quality-irrelevant faults in the left part, we consider a lower FDR of statistics performs better. While in quality-relevant faults arranged by faulty types in the right side, a higher FDR of statistics is preferred.

Other than D_y, the results of D_o and D_x by TSSAE are compared with other methods, such as PCA, ICA, KPCA and neural component analysis (NCA) in Table 3 [24–26]. Moreover, we also list the FDRs of a simple AE with 20 nodes in hidden layer and DBN. For most cases in the TEP, TSSAE exhibits a better performance in the FDRs than other methods especially for IDV (9)-(21). Obviously, TSSAE is more efficient in the extraction of fault-relevant information. Not only that, the average FARs of some algorithms are summarized in Table 4 and FARs of TSSAE are acceptable. Since TSSAE models in an interpretable way, operators or engineers can make a more reliable decision based on the results.

To be more specific, IDV 1, 4, 12 are discussed in detail. IDV 1 is caused by a step fault in A/C ratio and B composition. In Fig. 9, quality-relevant statistic D_y returned to normal station

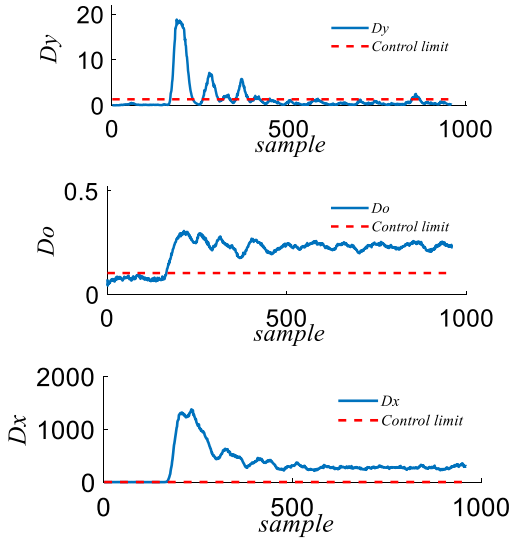

Fig. 9. Fault detection results of IDV 1 in TEP by TSSAE.

which is consistent with the real value in Fig. 7(b) and the fault is successfully detected by D_o or D_x. IDV 4 is a quality-irrelevant fault which resulted in the abnormality of the reactor cooling water inlet temperature. In Fig. 10, D_y only alarms at 3

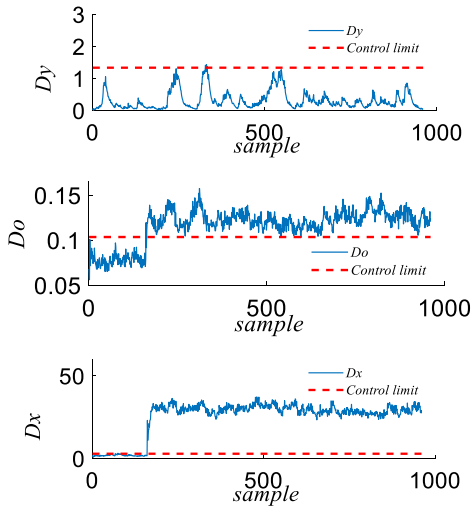

Fig. 10. Fault detection results of IDV 4 in TEP by TSSAE.

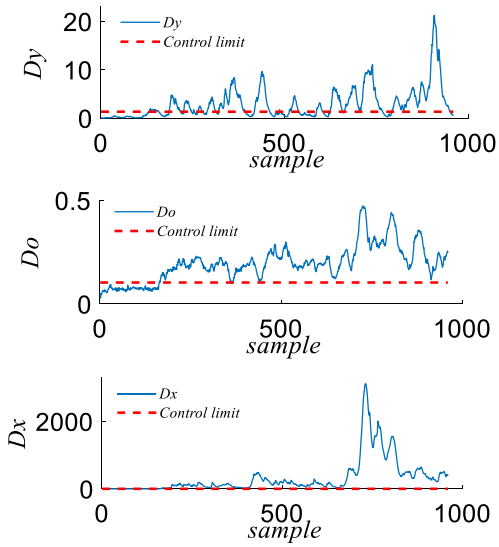

Fig. 11. Fault detection results of IDV 12 in TEP by TSSAE.

samples and D_o, D_x find out all the faulty samples in process variables. Unlike IDV 4, IDV 12 happened in the condenser cooling water inlet temperature. Thus D_y exhibits a divergent oscillation state in Fig. 11. At the same time, process variables were also unstable. These examples reveal the effectiveness of both the quality-relevant or quality-irrelevant features extracted by TSSAE and the overall advantages of simultaneously considering process and quality variables in modeling can be reflected.

4.3. Visualization

In order to show the efficiency of features learned and represented by TSSAE, PCA is run in the features extracted and the first three principal components (PCs) are selected. Taking fault 2 in numerical example and IDV 6, 8 in TEP as examples, the PCs are plotted in Fig. 12.

Notably, fault-free samples are marked in black and faulty ones are marked in red. It can be clearly seen that fault-free samples are concentrated in an area while the faulty ones deviate from this area. Because IDV 6 obviously deviated from normal station, it can be seen directly. Samples detected with a small deviation like fault 2 or IDV 8 are distributed closer to the normal area. In this case, it can be seen that TSSAE has learned the efficient characteristics of process and performs effectively in fault detection.

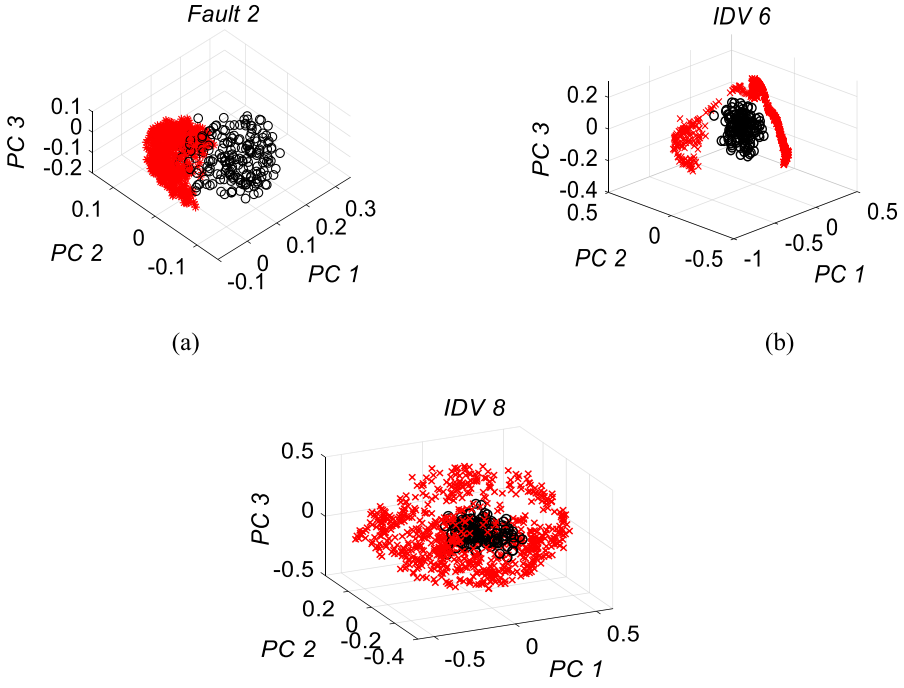

Fig. 12. Three PCs of features extracted by T model designed by TSSAE. (a) Fault 2 in numerical example. (b) IDV 6 in TEP. (c) IDV 8 in TEP.

5. Conclusions

This study presented and demonstrated a quality-relevant fault detection scheme using the designed TSSAE. In TSSAE, dual neural networks are used to extract and represent the features. In T, quality indicators are treated similar to process variables. On the basis of the approximation capability of DNNs, process variables and quality indicators can be simultaneously predicted in S. If process variables or quality indicators are modeled separately, several valuable information benefitting the prediction of the other parts may be lost. For instance, if a model is trained to predict quality indicators, valuable features for quality prediction are extracted, and the remaining information that are beneficial for predicting process variables are lost. In addition, the weights or biases of the student model can be supervised to a better status using the features extracted by the trained teacher model. Therefore, TSSAE exhibits good results on quality-relevant or quality-irrelevant fault detection. As it only models from fault-free data, TSSAE is suitable for other similar issues and achieves persuasive performance.

In general, deep learning methods are data-driven, and no prior knowledge is involved. For quality-relevant fault detection, the way our proposed method combines the quality-relevant information has played a certain role in the integration of prior information. In order to obtain more reliable monitoring performance, prior knowledge aids in improving the predictive accuracy of quality indicators and eases the interpretability of models. At the same time, accurate description of process information and extraction of efficient features are the keys to detect the faults sensitively in the field of data-driven methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (21878081) and the Fundamental Research Funds for the Central Universities under Grant of China (222201917006).

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2019.105526.

References

[1] S. Yin, S.X. Ding, X. Xie, H. Luo, A review on basic data-driven approaches for industrial process monitoring, IEEE Trans. Ind. Electron. 61 (11) (2014) 6418–6428.
[2] S.J. Qin, Survey on data-driven industrial process monitoring and diagnosis, Annual Rev. Contr. 26 (2) (2012) 220–234.
[3] Q. Jiang, X. Yan, B. Huang, Performance-driven distributed PCA process monitoring based on fault-relevant variable selection and Bayesian inference, IEEE Trans. Ind. Electron. 63 (1) (2016) 377–386.
[4] Q. Jiang, X. Yan, Parallel PCA-KPCA for nonlinear process monitoring, Control Eng. Pract. 80 (2018) 17–25.
[5] Y. Yao, F. Gao, Batch process monitoring in score space of two-dimensional dynamic principal component analysis (PCA), Ind. Eng. Chem. Res. 46 (24) (2007) 8033–8043.
[6] J. Zhu, Z. Ge, Z. Song, Distributed parallel PCA for modeling and monitoring of large-scale plant-wide process with big data, IEEE Trans. Ind. Inform. 13 (4) (2007) 1877–1885.
[7] K. Zhang, H. Hao, Z. Chen, S.X. Ding, K. Peng, A comparison and evaluation of key performance indicator-based multivariate statistics process monitoring approaches, J. Process Control 33 (2015) 112–126.
[8] G. Wang, H. Luo, K. Peng, Quality-related fault detection using linear and nonlinear principal component regression, J. Franklin Inst. 353 (10) (2016) 2159–2177.
[9] D. Zhou, G. Li, S.J. Qin, Total projection to latent structures for process monitoring, AIChE J. 56 (1) (2010) 168–178.
[10] S. Yin, G. Wang, H. Gao, Data-driven process monitoring based on modified orthogonal projections to latent structures, IEEE Trans. Control Syst. Technol. 24 (4) (2016) 1480–1487.
[11] S. Yan, J. Huang, X. Yan, Monitoring of quality-relevant and quality-irrelevant blocks with characteristic-similar variables based on self-organizing map and kernel approaches, J. Process Control 73 (2019) 103–112.
[12] G. Li, B. Liu, S.J. Qin, D. Zhou, Quality relevant data-driven modeling and monitoring of multivariate dynamic process: the dynamic T-PLS approach, IEEE Trans. Neural Netw. 22 (12) (2011) 2262–2271.
[13] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444.
[14] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.
[15] Z. Zhang, J. Zhao, A deep belief network based fault diagnosis model for complex chemical processes, Comput. Chem. Eng. 107 (2017) 395–407.
[16] J. Yu, X. Yan, Layer-by-layer enhancement strategy of favorable features of the deep belief network forindustrial process monitoring, Ind. Eng. Chem. Res. 57 (2018) 15479–15490.
[17] F. Lv, C. Wen, M. Liu, Z. Bao, Weighted time series fault diagnosis based on a stacked sparse autoencoder, J. Chemometr. 31 (9) (2017).
[18] Z. Zhang, T. Jiang, S. Li, Y. Yang, Automated feature learning for nonlinear process monitoring - An approach using stacked denoising autoencoder and k-nearest neighbor rule, J. Process Control 64 (2018) 49–61.
[19] L. Wen, X. Li, L. Gao, Y. Zhang, A new convolutional neural network-based data-driven fault diagnosis method, IEEE Trans. Ind. Electron. 65 (7) (2018) 5990–5998.
[20] Q. Jiang, X. Yan, Just-in-time recognized PCA integrated with SVDD for chemical process monitoring, AIChE J. 60 (3) (2014) 949–965.
[21] J.J. Downs, E.F. Vogel, A plant-wide industrial process control problem, Comput. Chem. Eng. 17 (3) (1993) 245–255.
[22] J. Huang, X. Yan, Quality relevant and independent two block monitoring based on mutual information and KPCA, IEEE Trans. Ind. Electron. 64 (8) (2017) 6518–6527.
[23] G. Wang, J. Jiao, A kernel least squares based approach for nonlinear quality-related fault detection, IEEE Trans. Ind. Electron. 64 (4) (2017) 3195–3204.
[24] J. Huang, X. Yan, GaussIan and non-Gaussian double subspace statistical process monitoring based on principal component analysis and independent component analysis, Ind. Eng. Chem. Res. 54 (3) (2015) 1015–1027.
[25] J. Huang, X. Yan, Related and independent variable fault detection based on KPCA and SVDD, J. Process Control 39 (2016) 88–99.
[26] H. Zhao, Neural component analysis for fault detection, Chemometr. Intell. Lab. Syst. 176 (2018) 11–21.