Robust Modeling of Mixture Probabilistic Principal Component Analysis and Process Monitoring Application

Jinlin Zhu, Zhiqiang Ge and Zhihuan Song
Dept. of Control Science and Engineering, State Key Laboratory of Industrial Control Technology, Institute of Industrial Process Control, Zhejiang University, Hangzhou 310027, P.R. China

DOI 10.1002/aic.14419
Published online March 4, 2014 in Wiley Online Library (wileyonlinelibrary.com)

In this article, a robust modeling strategy for mixture probabilistic principal component analysis (PPCA) is proposed. Different from the traditional Gaussian distribution driven model such as PPCA, the multivariate student t-distribution is adopted for probabilistic modeling to reduce the negative effect of outliers, which is very common in the process industry. Furthermore, for handling the missing data problem, a partially updating algorithm is developed for parameter learning in the robust mixture PPCA model. Therefore, the new robust model can simultaneously deal with outliers and missing data. For process monitoring, a Bayesian soft decision fusion strategy is developed which is combined with the robust local monitoring models under different operating conditions. Two case studies demonstrate that the new robust model shows enhanced modeling and monitoring performance in both outlier and missing data cases, compared to the mixture probabilistic principal analysis model. © 2014 American Institute of Chemical Engineers *AIChE J*, 60: 2143–2157, 2014

Keywords: robust probabilistic principal component analysis, Mixture model, Maximum likelihood, Robust process modeling, Outliers and missing data

Introduction

Among data-based statistical process monitoring techniques, principal component analysis (PCA) may be one of the most popular methods in the last two decades.[1–5] The commonly used PCA is a well-defined model which conducts a linear projection from the high-dimensional observed variable space into a low-dimensional projected space where the variance is maximized. Benefited from its elegant property, PCA has been widely used in data compression and feature extraction.[6–9] Despite of its conciseness and effectiveness, one major defect of PCA is the absence of the associated probabilistic interpretation of uncertainty in projection space. To address this issue, Tipping and Bishop presented a probabilistic framework for PCA from the perspective of latent variable model, namely, PPCA.[10] PPCA has many advantages such as the ability to deal with missing data. In the past several years, PPCA has received much attention and many successful applications in process monitoring have been reported.[11–14]

Conventionally, PCA or PPCA-based statistical monitoring schemes have been designed with the assumption that the process dataset is collected in a good manner and no outliers are submerged.[15] In practice, unfortunately, this assumption can be no longer tenable. In most conditions, the outlier is regarded as an observation that appears to deviate remarkably from the normally sampling regions.[16] As a matter of fact, outliers can take place due to several reasons, for example, the process measurement can be incorrectly observed, recorded, or copied into the historical database. The presence of outliers can have serious adverse effects on the modeling, which may distort estimations of parameters and lead to model misspecification.[17,18] Specifically, PCA modeling is sensitive to outliers since the projection mechanism is regulated by the quadratic criteria of variance maximization and minimization of mean squared error. While for the PPCA case, the parameter learning process can be skewed by outliers since the probabilistic framework is represented as a multivariate Gaussian which is also sensible to deviated measurements. Therefore, the outlier issue has been attached with importance in data pretreatment and data modeling. Some data modeling methods seek to alleviate the influence from outliers by detecting and eliminating them during the data preprocessing.[19–21] Although an automatic outlier detection and elimination procedure is effective in some cases, one should be cautious since the elimination process may also wipe off some important hidden information, which may cause negative effect in the following data modeling procedure. Alternatively, the PPCA model has recently been reformulated within a robust framework based on the Student t-density.[22,23] The core idea by replacing the Gaussian density with t density is quite evident, since t-distribution can be seen as a generalization of Gaussian with a heavy tail. Informally speaking, the outliers can be tolerated in some extent by adjusting the thickness of the density tail, while for the Gaussian case, explanations on outliers are not so

Correspondence concerning this article should be addressed to Z. Ge at gezhiqiang@zju.edu.cn.

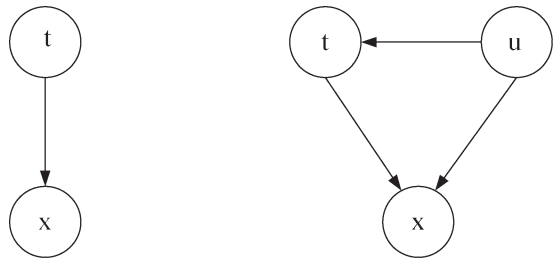
enough that the whole distribution has to move close to the outliers to contain these information, which in turns makes PPCA sensible to outliers. Owning to its fine property, the robust probabilistic principal component analysis (RPPCA) has been recently used for outlier detection by Chen et al.[24]

Although the ability of RPPCA under outliers has been explored, there still exist two important issues that have not been well settled by RPPCA for process monitoring. First of all, RPPCA assumes the single modality which can be easily violated since the process data may be collected from various operations which display multimodal characteristics. Second, parameter learning and inference should be adaptable to missing data during the modeling process since the missing values can also be mixed into cases in process database. This article intends to deal with above two issues. To capture the multimodal characteristic of process data, the single robust probabilistic model is extended to the mixture form within the expectation-maximization (EM) algorithm so that different local models can capture different modalities. To deal with the missing data problem, we reconstruct the parameter learning algorithm by learning from the rest observed entries. This can be reliable under the conditions when missing values are generated independently from different sensors, which is commonly seen in practice. Although Mixture RPPCA (MRPPCA) has the ability to cope with multimode process data, another key problem concerned with process monitoring by such mixture model is how to align the local monitoring statistics for the overall fault detection purpose. One can assign a new data sample according to each local statistic, and make the decision by choosing the status with the most supported local models that share the same decision. However, such hard assignment or classification usually makes the global discrimination without considering the possibility that each new sample may be generated from other conditions. In this work, we modify the fully Bayesian fusion method proposed by Ge et al.[25,26] Thus, the local monitoring statistics are transformed into the corresponding local likelihood probabilities through the sigmoid function, and then the posterior fault probability can be derived with the Bayesian rule. By comparing the posterior fault probability with the control limit, one can easily make the global discrimination for the current data sample.

The rest of this article is organized as follows. In robust probabilistic principal component analysis Section, a brief introduction of RPPCA is given. A detailed illustration of the EM algorithm is provided in mixture robust modeling of PPCA section, followed by the formulation of the Bayesian soft fusion scheme for process monitoring in the next section. In case studies, a simple simulation example is first given to evaluate the validity and efficiency of the proposed method, after that we compare our method with other commonly used techniques on the Tennessee Eastman challenge problem under various cases. Finally, conclusions are made.

Robust Probabilistic Principal Component Analysis

Like the PPCA method, RPPCA tries to find a linear projection from the original measurements set $\{\mathbf{x}_n|\mathbf{x}_n \in R^D\}_{n=1}^N$ to a lower dimensional latent vectors $\{\mathbf{t}_n|\mathbf{t}_n \in R^d\}_{n=1}^N$ plus the noise $\mathbf{e}$, which can be described as follows



(a) (b)

Figure 1. Left part (a) presents the Bayesian network structure for original PPCA and the right part (b) shows the Bayesian network structure for RPPCA which includes the scaling prior u.

$$\mathbf{x}_n = \mathbf{P}\mathbf{t}_n + \boldsymbol{\mu} + \mathbf{e}_n \tag{1}$$

where $\mathbf{P} \in R^{D\times d}$ is the orthogonal projection matrix, $\boldsymbol{\mu} \in R^D$ denotes the offset, $d < D$.

Unlike PPCA which assumes the multivariate Gaussian distribution for both observed space $\mathbf{x}$ and latent space $\mathbf{t}$, robust PPCA assumes both spaces followed by a multivariate student t-distribution, thus, the prior and likelihood can be defined as[22]

$$p(\mathbf{t}_n) = S(\mathbf{t}_n|\mathbf{0}, \mathbf{I}_d, v) \tag{2}$$

$$p(\mathbf{x}_n|\mathbf{t}_n) = S(\mathbf{x}_n|\mathbf{P}\mathbf{t}_n + \boldsymbol{\mu}, \boldsymbol{\Lambda}, v) \tag{3}$$

$$S(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Lambda}, v) = \frac{\Gamma\left(\frac{v+d}{2}\right)|\boldsymbol{\Lambda}|^{1/2}}{\Gamma\left(\frac{v}{2}\right)(v\pi)^{d/2}}\left(1 + \frac{(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Lambda}(\mathbf{x}-\boldsymbol{\mu})}{v}\right)^{-(v+d)/2} \tag{4}$$

where $v(>0)$ is the parameter known as "degree of freedom," $\boldsymbol{\mu}$ and $\boldsymbol{\Lambda} = \tau\mathbf{I}_D$ denote the mean and diagonal precision (inverse variance matrix), respectively, gamma function $\Gamma(\mathbf{x}) = \int_0^\infty z^{\mathbf{x}-1}\mathbf{e}^{-z}dz$. Since the student t-distribution can be represented as an infinite mixture of Gaussians with a Gamma distribution as the mixture prior (see Appendix A for details)

$$S(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Lambda}, v) = \int_0^\infty N(\mathbf{x}|\boldsymbol{\mu}, u\boldsymbol{\Lambda})Ga(u|v/2, v/2)du = \langle N(\mathbf{x}|\boldsymbol{\mu}, u\boldsymbol{\Lambda})\rangle_{u|v} \tag{5}$$

where Ga(.|.) refers to the Gamma prior distribution and N(.|.) denotes the Gaussian distribution. Therefore, the above Eqs. 2 and 3 can be rewritten as the following expectation formulation[22]

$$p(\mathbf{t}_n) = \langle N(\mathbf{t}_n|\mathbf{0}, u_n\mathbf{I})\rangle_{u_n|v} \tag{6}$$

$$p(\mathbf{x}_n|\mathbf{t}_n) = \langle N(\mathbf{x}_n|\mathbf{P}\mathbf{t}_n + \boldsymbol{\mu}, u_n\boldsymbol{\Lambda})\rangle_{u_n|v} \tag{7}$$

from which we can infer that robustness in the observed space as well as the latent space should be ensured by adjusting the scaling variable u, and the RPPCA tend to become the exact PPCA since the student t-distributions tend to be the Gaussian ones as $v \to \infty$. To make this clear, the corresponding Bayesian network structure for PPCA and RPPCA has been presented in Figure 1, moreover, the student t-distributions with $\boldsymbol{\mu}$ and $\boldsymbol{\Lambda}$ fixed for different v are depicted in Figure 2. From Figure 1, we can see that the

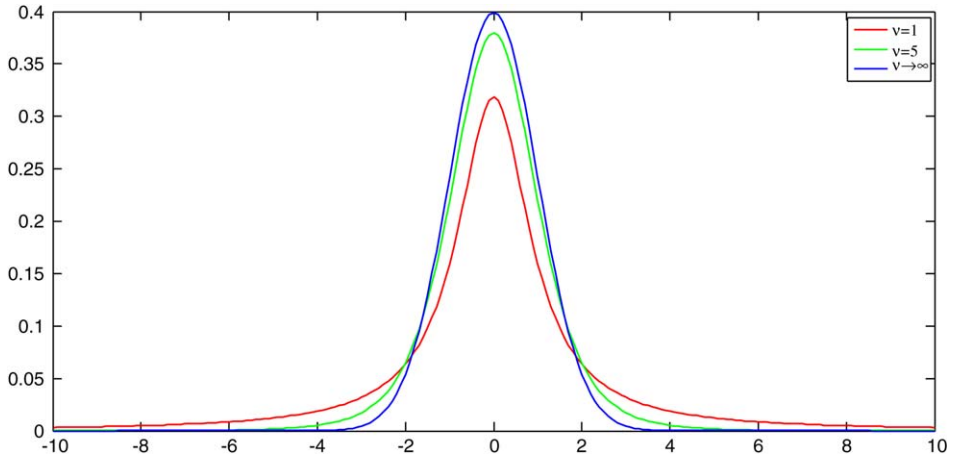


Figure 2. The student *t*-distributions with μ and Λ fixed for different v, where the distribution tends to be the Gaussian one as $v \to \infty$.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

scaling variable accounts for both the latent space and the observed space. After the model construction, the learning procedure for RPPCA, therefore, is to determine the parameter set $\theta=\{\mu, \Lambda, \mathbf{P}, v\}$, and the well-known EM algorithm can be utilized to iteratively find the good-fit maximum likelihood solution.

Mixture Robust Modeling of PPCA

For data modeling in the multimodal situation, a mixture of K local RPPCA can be incorporated and the distribution is defined as

$$\mathrm{p}(\mathbf{x})=\sum_{\mathrm{k}=1}^{\mathrm{K}} \pi_{\mathrm{k}} \mathrm{p}(\mathbf{x}|\theta_{\mathrm{k}}) \tag{8}$$

where $\{\pi_{\mathrm{k}}\}_{\mathrm{k}=1}^{\mathrm{K}}$ is component prior set that satisfies $\sum_{\mathrm{k}=1}^{\mathrm{K}} \pi_{\mathrm{k}}=1$, and $\{\theta_{\mathrm{k}}\}_{\mathrm{k}=1}^{\mathrm{K}}$ is the set of parameters within each component. If we use a binary indicator z_{nk} to declare the mixture label for each measurement $\mathbf{x}_{\mathrm{n}}$, the Bayesian network structure for MRPPCA can be shown as Figure 3.

Learning with complete data

In MRPPCA, the parameter set should be $\theta=\{\pi_{\mathrm{k}}, \mu_{\mathrm{k}}, \Lambda_{\mathrm{k}}, \mathbf{P}_{\mathrm{k}}, v_{\mathrm{k}}\}_{\mathrm{k}=1}^{\mathrm{K}}$. We can learn the parameters from data using the EM algorithm. Given the process data $\mathbf{X} \in \mathrm{R}^{\mathrm{N}\times\mathrm{D}}$, the complete log-likelihood $\mathrm{L}(\theta)$ is defined as follows

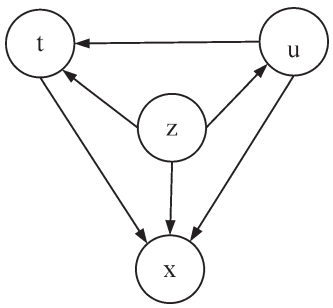


Figure 3. Bayesian network structure for MRPPCA

$$\mathrm{L}(\theta)=\sum_{k=1}^{K} \log \prod_{n=1}^{N} \mathrm{p}(z_{nk}, u_{nk}, \mathbf{t}_{nk}, \mathbf{x}_n|\theta) \tag{9}$$

By applying the independent rule, Eq. 9 can be factorized as

$$\mathrm{L}(\theta)=\sum_{k=1}^{K} \log \prod_{n=1}^{N} \mathrm{p}(z_{nk}|\theta)\mathrm{p}(u_{nk}|z_{nk}, \theta)\mathrm{p}(\mathbf{t}_{nk}|z_{nk}, u_{nk}, \theta) \mathrm{p}(\mathbf{x}_n|z_{nk}, u_{nk}, \mathbf{t}_{nk}, \theta) \tag{10}$$

where, each conditional probabilistic distribution term is defined as[23]

$$\mathrm{p}(z_{nk}|\theta)=\pi_{\mathrm{k}}{}^{z_{nk}} \tag{11}$$

$$\mathrm{p}(u_{nk}|z_{nk}, \theta)=\mathrm{Ga}\,(u_{nk}|v_{\mathrm{k}}/2, v_{\mathrm{k}}/2)^{z_{nk}} \tag{12}$$

$$\mathrm{p}(\mathbf{t}_{nk}|z_{nk}, u_{nk}, \theta)=\mathrm{N}(\mathbf{t}_{nk}|0, u_{nk}\mathbf{I}_{\mathrm{d}})^{z_{nk}} \tag{13}$$

$$\mathrm{p}(\mathbf{x}_n|z_{nk}, u_{nk}, \mathbf{t}_{nk}, \theta)=\mathrm{N}(\mathbf{x}_n|\mathbf{P}_k\mathbf{t}_{nk}+\mu_k, u_{nk}\Lambda_k)^{z_{nk}} \tag{14}$$

where $\Lambda_{\mathrm{k}}=\tau_{\mathrm{k}}\mathbf{I}_{\mathrm{D}}$ is an isotropic matrix, $\mathbf{x}_{\mathrm{nk}}=\mathbf{x}_{\mathrm{n}}$. After that, the log-likelihood can be further expanded as

$$L(\theta)=\sum_{k=1}^{K}\sum_{n=1}^{N}\left\{z_{nk}\log \pi_k+z_{nk}\left[\frac{v_k}{2}\log\frac{v_k}{2}-\log\Gamma\left(\frac{v_k}{2}\right)+\left(\frac{v_k}{2}-1\right)\log u_{nk}-\frac{v_k}{2}u_{nk}\right]-\frac{z_{nk}}{2}\log|u_{nk}^{-1}\Lambda_k^{-1}|-\frac{z_{nk}u_{nk}}{2}\|\mathbf{x}_n-(\mathbf{P}_k\mathbf{t}_{nk}+\mu_k)\|^2_{\Lambda_k^{-1}}\right\} \tag{15}$$

where $\|\mathbf{x}_{\mathrm{n}}-(\mathbf{P}_{\mathrm{k}}\mathbf{t}_{nk}+\mu_{\mathrm{k}})\|^2_{\Lambda_{\mathrm{k}}^{-1}}$ represents the Mahalanobis distance, the detailed expression is

$$\|\mathbf{x}_{\mathrm{n}}-(\mathbf{P}_{\mathrm{k}}\mathbf{t}_{nk}+\mu_{\mathrm{k}})\|^2_{\Lambda_{\mathrm{k}}^{-1}}=\mathrm{tr}\left(\Lambda_{\mathrm{k}}(\mathbf{x}_{\mathrm{n}}-\mu_{\mathrm{k}})\left(\mathbf{x}_{\mathrm{n}}-\mu_{\mathrm{k}}\right)^{\mathrm{T}}\right) -2\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}\Lambda_{\mathrm{k}}(\mathbf{x}_{\mathrm{n}}-\mu_{\mathrm{k}})+\mathrm{tr}\left(\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}\Lambda_{\mathrm{k}}\mathbf{P}_{\mathrm{k}}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\right) \tag{16}$$

Notice that in Eq. 15, those terms do not contain θ can be considered as unrelated items that have been omitted for simplicity. Now we can perform the EM algorithm. In the E-step, the expectation of log-likelihood $\mathrm{L}(\theta)$ is computed with respect to the conditional joint distribution with respect to the latent variables, which can be written as

$$\mathrm{Q}(\theta|\theta_{old}) \equiv \mathrm{E}(\mathrm{L}(\theta)|\mathbf{X}, \theta_{old}) \tag{17}$$

While in the M-step, we update parameters so that Eq. 17 is maximized.

In the E-step, first of all, we have to obtain the posterior distribution for each variable so as to compute the expectation term. By applying the Bayes rule, the posterior distribution for indicator z_{nk} as well as its expectation (scalar) can be obtained as

$$\mathrm{p}(z_{nk}=1|\mathbf{x}_\mathrm{n},\boldsymbol{\theta}_{old})=\frac{\pi_\mathrm{k}\mathrm{S}(\mathbf{x}_\mathrm{n}|\boldsymbol{\mu}_\mathrm{k},\mathbf{A}_\mathrm{k},\nu_\mathrm{k})}{\sum_{\mathrm{n}=1}^{\mathrm{N}}\pi_\mathrm{k}\mathrm{S}(\mathbf{x}_\mathrm{n}|\boldsymbol{\mu}_\mathrm{k},\mathbf{A}_\mathrm{k},\nu_\mathrm{k})}=\langle z_{nk}\rangle \tag{18}$$

Then, the posterior distribution for u_{nk} under current z_{nk} can also be induced with Bayes rule as

$$\mathrm{p}(u_{nk}|\mathbf{x}_\mathrm{n},z_{nk}=1,\boldsymbol{\theta}_{old})=\frac{\mathrm{p}(\mathbf{x}_\mathrm{n}|z_{nk}=1,u_{nk},\boldsymbol{\theta}_{old})\mathrm{p}(u_{nk}|z_{nk}=1,\boldsymbol{\theta}_{\mathrm{old}})}{\mathrm{p}(\mathbf{x}_\mathrm{n}|z_{nk}=1,\boldsymbol{\theta}_{old})} \tag{19}$$

where, the left numerator is

$$\mathrm{p}(\mathbf{x}_\mathrm{n}|z_{nk}=1,u_{nk},\boldsymbol{\theta}_{old})=\mathrm{N}(\boldsymbol{\mu}_\mathrm{k},u_{nk}\mathbf{A}_\mathrm{k}) \tag{20}$$

where $\mathbf{A}_\mathrm{k}^{-1}=\mathbf{P}_\mathrm{k}^{\mathrm{T}}\mathbf{P}_\mathrm{k}+\boldsymbol{\Lambda}_\mathrm{k}^{-1}$, combining Eqs. 12 and 20, we have

$$\mathrm{p}(\mathrm{u}_{nk}|\mathbf{x}_\mathrm{n},z_{nk}=1,\boldsymbol{\theta}_{old})=\mathrm{Ga}\left(\mathrm{u}_{nk}|\frac{\mathrm{D}+\mathrm{v}_\mathrm{k}}{2},\frac{\|\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k}\|^2_{\mathbf{A}_\mathrm{k}^{-1}}+\mathrm{v}_\mathrm{k}}{2}\right) \tag{21}$$

according to Eq. 21 and the property that if $\tau\sim\mathrm{Ga}(\alpha,\beta)$ then $\langle\tau\rangle=\alpha/\beta$, the expectation is

$$\langle\mathrm{u}_{nk}\rangle=\frac{\mathrm{D}+\mathrm{v}_\mathrm{k}}{\|\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k}\|^2_{\mathbf{A}_\mathrm{k}^{-1}}+\mathrm{v}_\mathrm{k}} \tag{22}$$

Similarly, the posterior for $\mathbf{t}_{nk}$ can be inferred as

$$\mathrm{p}(\mathbf{t}_{nk}|\mathbf{x}_n,z_{nk}=1,u_{nk},\boldsymbol{\theta}_{old})=\frac{\mathrm{p}(\mathbf{x}_n|z_{nk},u_{nk},\mathbf{t}_{nk},\boldsymbol{\theta}_{old})\mathrm{p}(\mathbf{t}_{nk}|z_{nk},u_{nk},\boldsymbol{\theta}_{old})\Big)}{\mathrm{p}(\mathbf{x}_n|\mathrm{z}_{nk},u_{nk},\boldsymbol{\theta}_{old})} \tag{23}$$

by combining Eqs. 13 and 14, then we have (for details, see Appendix B)

$$\mathrm{p}(\mathbf{t}_{nk}|\mathbf{x}_\mathrm{n},z_{nk}=1,u_{nk},\boldsymbol{\theta}_{old})=\mathrm{N}(\mathbf{t}_{nk}|\mathbf{B}_\mathrm{k}^{-1}\mathbf{P}_\mathrm{k}^{\mathrm{T}}\boldsymbol{\Lambda}_\mathrm{k}(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k}),u_{nk}\mathrm{B}_\mathrm{k}) \tag{24}$$

therefore, we obtain the expectation for latent variable as

$$\langle\mathbf{t}_{nk}\rangle=\mathbf{B}_\mathrm{k}^{-1}\mathbf{P}_\mathrm{k}^{\mathrm{T}}\boldsymbol{\Lambda}_\mathrm{k}(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k}) \tag{25}$$

where matrix $\mathbf{B}_k=\mathbf{P}_k^T\boldsymbol{\Lambda}_k\mathbf{P}_k+\mathbf{I}_d$.

Following the E-step, we now come to the M-step, and each parameter is updated in its turn while the others are fixed. First, consider updating the component weight set $\{\pi_\mathrm{k}\}_{\mathrm{k}=1}^{\mathrm{K}}$. Since the weights satisfy the constraints $\sum_{\mathrm{k}=1}^{\mathrm{K}}\pi_\mathrm{k}=1$, by adding a Lagrange multiplier λ into $\mathrm{Q}(\boldsymbol{\theta}|\boldsymbol{\theta}_{old})$, we have

$$\mathrm{Q}'(\boldsymbol{\theta}|\boldsymbol{\theta}_{old})=\mathrm{Q}(\boldsymbol{\theta}|\boldsymbol{\theta}_{old})+\lambda\left(\sum_{\mathrm{k}=1}^{\mathrm{K}}\pi_\mathrm{k}-1\right) \tag{26}$$

Take the derivation of Eq. 26 with respect to π_k and set the derivation to zero, we get

$$\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle+\lambda\pi_\mathrm{k}=0 \tag{27}$$

Summing both sides over π_k, and after some manipulations we have

$$\lambda=-\sum_{k=1}^{K}\sum_{n=1}^{N}\langle\mathrm{Z}_{nk}\rangle \tag{28}$$

therefore, the update formulation for weight should be

$$\pi_\mathrm{k}=-\frac{1}{\lambda}\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle=\frac{\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle}{\sum_{\mathrm{k}=1}^{\mathrm{K}}\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle}=\frac{\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle}{\mathrm{N}} \tag{29}$$

Now turn to the updating of mean for each component. Take the derivation of Eq. 17 with respect to μ_k to zero and we obtain

$$\sum_{n=1}^{N}\langle\mathrm{z}_{nk}u_{nk}\boldsymbol{\Lambda}_k\left(\mathbf{x}_n-(\mathbf{P}_k\mathbf{t}_{nk}+\boldsymbol{\mu}_k)\right)\rangle=0 \tag{30}$$

We can easily get

$$\boldsymbol{\mu}_\mathrm{k}=\frac{\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle\langle\mathrm{u}_{nk}\rangle\mathbf{x}_\mathrm{n}-\mathbf{P}_\mathrm{k}\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle\langle\mathrm{u}_{nk}\mathbf{t}_{nk}\rangle}{\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle\mathrm{z}_{nk}\rangle\langle\mathrm{u}_{nk}\rangle} \tag{31}$$

The expectations for $\langle\mathrm{z}_{nk}\rangle$ and $\langle\mathrm{u}_{nk}\rangle$ are given, respectively, in Eqs. 18 and 22, we just need to compute the expectation for $\langle\mathrm{u}_{nk}\mathbf{t}_{nk}\rangle$. To solve this, we have to make use of the conditional dependent relationship between these two variables in Figure 2, and we have

$$\begin{aligned}\langle u_{nk}\mathbf{t}_{nk}\rangle&=\mathrm{E}(u_{nk}\mathbf{t}_{nk}|\mathbf{x}_\mathrm{n},\boldsymbol{\theta}_{\mathrm{old}})\\&=\mathrm{E}\Big(u_{nk}\mathrm{E}(\mathbf{t}_{nk}|\mathbf{x}_\mathrm{n},u_{nk},\boldsymbol{\theta}_{\mathrm{old}})|\mathbf{x}_\mathrm{n},\boldsymbol{\theta}_{\mathrm{old}}\Big)\\&=\mathrm{E}(u_{nk}\langle\mathbf{t}_{nk}\rangle|\mathbf{x}_\mathrm{n},\boldsymbol{\theta}_{\mathrm{old}})\\&=\langle u_{nk}\rangle\langle\mathbf{t}_{nk}\rangle\end{aligned} \tag{32}$$

Similar to the mean updating, we now consider the updating for covariance matrix. Take the derivation of Eq. 17 with respect to $\boldsymbol{\Lambda}_\mathrm{k}^{-1}$ to zero and we obtain

$$\begin{aligned}\sum_{\mathrm{n}=1}^{\mathrm{N}}[&-\langle z_{nk}\rangle\boldsymbol{\Lambda}_\mathrm{k}+\langle z_{nk}\rangle\langle u_{nk}\rangle\boldsymbol{\Lambda}_\mathrm{k}(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k})(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k})^{\mathrm{T}}\\&\times\boldsymbol{\Lambda}_\mathrm{k}-2\boldsymbol{\Lambda}_\mathrm{k}\mathbf{P}_\mathrm{k}\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\rangle(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k})^{\mathrm{T}}\boldsymbol{\Lambda}_\mathrm{k}\\&+\boldsymbol{\Lambda}_\mathrm{k}\mathbf{P}_\mathrm{k}\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle\mathbf{P}_\mathrm{K}^{\mathrm{T}}\boldsymbol{\Lambda}_\mathrm{k}]=0\end{aligned} \tag{33}$$

Rearrange Eq. 33 and solve for $\boldsymbol{\Lambda}_\mathrm{k}^{-1}$, and the updating for original covariance matrix $\overline{\boldsymbol{\Lambda}}_\mathrm{k}^{-1}$ can be obtained as

$$\begin{aligned}\overline{\boldsymbol{\Lambda}}_\mathrm{k}^{-1}=\frac{1}{\mathrm{N}\pi_\mathrm{k}}\sum_{\mathrm{n}=1}^{\mathrm{N}}\Big[&\langle z_{nk}\rangle\langle u_{nk}\rangle(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k})(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k})^{\mathrm{T}}\\&-2\mathbf{P}_\mathrm{k}\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\rangle(\mathbf{x}_\mathrm{n}-\boldsymbol{\mu}_\mathrm{k})^{\mathrm{T}}+\mathbf{P}_\mathrm{k}\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle\mathbf{P}_\mathrm{k}^{\mathrm{T}}\Big]\end{aligned}$$

After some manipulations (add matrix operator trace to both sides), one can further obtain

$$\begin{aligned}\tau_k^{-1}=\frac{1}{\mathrm{ND}\pi_k}\sum_{\mathrm{n}=1}^{\mathrm{N}}tr[&\langle z_{nk}\rangle\langle u_{nk}\rangle(\mathbf{x}_n-\boldsymbol{\mu}_k)(\mathbf{x}_n-\boldsymbol{\mu}_k)^{\mathrm{T}}\\&-2\mathbf{P}_k\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\rangle(\mathbf{x}_n-\boldsymbol{\mu}_k)^{\mathrm{T}}+\mathbf{P}_k\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{nk}^{\mathrm{T}}\rangle\mathbf{P}_k^{\mathrm{T}}]\end{aligned} \tag{34}$$

Where tr(.) computes the trace for matrix, the expectation $\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle$ can be computed as

$$\begin{aligned}\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle &= \mathrm{E}(u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}|\mathbf{x}_{\mathrm{n}},\boldsymbol{\theta}_{\mathrm{old}})\\ &= \mathrm{E}\Big(u_{nk}\mathrm{E}(\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}|\mathbf{x}_{\mathrm{n}},u_{nk},\boldsymbol{\theta}_{\mathrm{old}})|\mathbf{x}_{\mathrm{n}},\boldsymbol{\theta}_{\mathrm{old}}\Big)\\ &= \mathrm{E}(u_{nk}\langle\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle|\mathbf{x}_{\mathrm{n}},\boldsymbol{\theta}_{\mathrm{old}})\\ &= \langle u_{nk}\rangle\langle\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle\end{aligned} \tag{35}$$

According to the property $(u_{nk})^{-1}\mathbf{B}_{\mathrm{k}}^{-1}=\langle\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle-\langle\mathbf{t}_{nk}\rangle\langle\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle^{\mathrm{T}}$, we can finally obtain $\langle\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle=(u_{nk})^{-1}\mathbf{B}_{\mathrm{k}}^{-1}+\langle\mathbf{t}_{nk}\rangle\langle\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle^{\mathrm{T}}$.

Now consider the updating for the projection matrix. The same as above, we take the derivation with respect to the projection matrix and set it to zero, with some computations and we have

$$\sum_{\mathrm{n}=1}^{\mathrm{N}}[\boldsymbol{\Lambda}_k(\mathbf{x}_n-\boldsymbol{\mu}_k)\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\rangle^{T}-\boldsymbol{\Lambda}_k\mathbf{P}_k\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{nk}^{T}\rangle]=0 \tag{36}$$

which leads to

$$\mathbf{P}_{\mathrm{k}}=\left[\sum_{\mathrm{n}=1}^{\mathrm{N}}(\mathbf{x}_{\mathrm{n}}-\boldsymbol{\mu}_{\mathrm{k}})\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\rangle^{\mathrm{T}}\right]\left[\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle\right]^{-1} \tag{37}$$

Finally, we turn to the update for v_{k}. By taking the gradient with respect to v_{k}, one should obtain the nonlinear equation as

$$\frac{\mathrm{N}\pi_{\mathrm{k}}}{2}\left(\log\frac{v_{\mathrm{k}}}{2}+1\right)-\frac{\mathrm{N}\pi_{\mathrm{k}}}{2}\psi\left(\frac{v_{\mathrm{k}}}{2}\right)+\frac{1}{2}\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle(\langle\log u_{nk}\rangle-\langle u_{nk}\rangle)=0 \tag{38}$$

Where $\psi(\mathbf{x})=\frac{\mathrm{d}\ln\Gamma(\mathbf{x})}{\mathrm{d}\mathbf{x}}$ is known as the digamma function, and the log expectation can be obtained through

$$\langle\log u_{nk}\rangle=\psi\left(\frac{\mathrm{D}+v_{\mathrm{k}}}{2}\right)-\log\left(\frac{\|\mathbf{x}_{\mathrm{n}}-\boldsymbol{\mu}_{\mathrm{k}}\|^2_{\mathbf{A}_{\mathrm{k}}^{-1}}+v_{\mathrm{k}}}{2}\right) \tag{39}$$

according to the Gamma distribution property that if $\tau\sim\mathrm{Ga}(\alpha,\beta)$, then $\langle\log\tau\rangle=\psi(\alpha)-\log(\beta)$. However, Eq. 38 is nonlinear and has no closed form solution, thus, we have to resort to other numerical tricks like nonlinear maximization method, see for example Ref. [27].

So far, we have completed the expectation maximization algorithm for MRPPCA parameter learning under complete data case, and the obtained algorithm has to be repeated until convergence criteria is satisfied. Specifically, we compute the expectations with Eqs. 18, 22, 25, 32, and 35 in the E-step, and update parameters, respectively, with Eqs. 29, 31, 34, 37, and 38 in the M-step, these two steps are carried out iteratively until the parameters change little in proportion (say 10^{-6}). In practice, however, data may be lost in some time points during the course of sampling and transmission, due to noise, sensor failure, multiple rate sampling scheme or network packet loss and so forth.[28–30] Therefore, it is quite important that we should realize the data modeling algorithm under the missing data case. In the next subsection, the parameter learning algorithm is modified to adapt the missing data situation.

Learning with missing data

In this section, a partially updating scheme based on the latent variable model is presented to deal with the missing values. This partially updating process uses all the information of the incomplete sample and updates the parameters of which the corresponding measured part in sample collection can be observed. This procedure makes no iterative computing mechanisms which can be effective and efficient in both modeling and monitoring with missing values. Notice that the missing data mechanism in this work is assumed to be completely missing at random.[31]

Given the incomplete data sample $\mathbf{x}'_{\mathrm{n}}=(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}},\mathbf{x}_{\mathrm{n}}^{\mathrm{m}})$, where $\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}$ stands for the observed part and $\mathbf{x}_{\mathrm{n}}^{\mathrm{m}}$ for the missing part, the E-step is obtained as follows

$$\langle z_{nk}\rangle'=\frac{\pi_{\mathrm{k}}\mathrm{S}(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}|\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}},\mathrm{A}_{\mathrm{k}}^{\mathrm{o}},\mathrm{v}_{\mathrm{k}})}{\sum_{\mathrm{n}=1}^{\mathrm{N}}\pi_{\mathrm{k}}\mathrm{S}(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}|\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}},\mathrm{A}_{\mathrm{k}}^{\mathrm{o}},\mathrm{v}_{\mathrm{k}})} \tag{40}$$

$$\langle u_{nk}\rangle'=\frac{\mathrm{D}^{\mathrm{o}}+\mathrm{v}_{\mathrm{k}}}{\|\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}}\|^2_{(\mathrm{A}_{\mathrm{k}}^{\mathrm{o}})^{-1}}+\mathrm{v}_{\mathrm{k}}} \tag{41}$$

$$\langle\mathbf{t}_{nk}\rangle'=\mathrm{B}_{\mathrm{k}}^{-1}(\mathbf{P}_{\mathrm{k}}^{\mathrm{o}})^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathrm{o}}(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}}) \tag{42}$$

$$\langle\log u_{nk}\rangle=\psi\left(\frac{\mathrm{D}^{\mathrm{o}}+\mathrm{v}_{\mathrm{k}}}{2}\right)-\log\left(\frac{\|\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}}\|^2_{(\mathrm{A}_{\mathrm{k}}^{\mathrm{o}})^{-1}}+\mathrm{v}_{\mathrm{k}}}{2}\right) \tag{43}$$

$$\langle u_{nk}\mathbf{t}_{nk}\rangle'=\langle u_{nk}\rangle'\langle\mathbf{t}_{nk}\rangle' \tag{44}$$

$$\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle'=\langle u_{nk}\rangle'\langle\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle'=\mathrm{B}_{\mathrm{k}}^{-1}+\langle u_{nk}\rangle'\langle\mathbf{t}_{nk}\rangle'(\langle\mathbf{t}_{nk}\rangle')^{\mathrm{T}} \tag{45}$$

Where $\mathrm{D}^{\mathrm{o}}=\dim(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}})$ denotes the dimensionality of observed part for $\mathbf{x}'_{\mathrm{n}}$, and $\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}},\mathbf{A}_{\mathrm{k}}^{\mathrm{o}},\mathbf{P}_{\mathrm{k}}^{\mathrm{o}}$ denotes the corresponding submatrix with entries set to zeros for missing part. The M-step follows the E-step as

$$\pi_{\mathrm{k}}^{\mathrm{o}}=\frac{\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle'}{\mathrm{N}} \tag{46}$$

$$\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{o}}=\frac{\sum_{\mathrm{n}=1}^{\mathrm{N}}\mathrm{N}\langle z_{nk}\rangle'\langle u_{nk}\rangle'\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\mathbf{P}_{\mathrm{k}}\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle'\langle u_{nk}\mathbf{t}_{nk}\rangle'}{\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle'\langle u_{nk}\rangle'} \tag{47}$$

$$\begin{aligned}(\tau_{\mathrm{k}}^{\mathrm{o}})^{-1}=\frac{1}{\mathrm{N}\pi_{\mathrm{k}}}\sum_{\mathrm{n}=1}^{\mathrm{N}}\mathrm{tr}[&\langle z_{nk}\rangle'\langle u_{nk}\rangle'(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{0})(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{0})^{\mathrm{T}}\\ &-2\mathbf{P}_{\mathrm{k}}^{\mathrm{o}}\langle z_{nk}\rangle'\langle u_{nk}\mathbf{t}_{nk}\rangle'(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{0})^{\mathrm{T}}\\ &+\mathbf{P}_{\mathrm{k}}^{\mathrm{o}}\langle z_{nk}\rangle'\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle'(\mathbf{P}_{\mathrm{k}}^{0})^{\mathrm{T}}]\end{aligned} \tag{48}$$

$$\begin{aligned}\mathbf{P}_{\mathrm{k}}^{\mathrm{o}}=&\left[\sum_{\mathrm{n}=1}^{\mathrm{N}}(\mathbf{x}_{\mathrm{n}}^{\mathrm{o}}-\boldsymbol{\mu}_{\mathrm{k}}^{0})\langle z_{nk}\rangle'(\langle u_{nk}\mathbf{t}_{nk}\rangle')^{\mathrm{T}}\right]\\ &\times\left[\sum_{\mathrm{n}=1}^{\mathrm{N}}\langle z_{nk}\rangle'\langle u_{nk}\mathbf{t}_{nk}\mathbf{t}_{\mathrm{nk}}^{\mathrm{T}}\rangle'\right]^{-1}\end{aligned} \tag{49}$$

Notice that the re-estimate for $\mathbf{A}_{\mathrm{k}}$, $\mathbf{B}_{\mathrm{k}}$ and v_{k} is similar to the data complete case which is omitted here for simplicity. Also, to obtain the optimal parameters of the model, the E-step and M-step alternates until convergence.

Discussions

Generally speaking, outliers can be treated as obvious ones and nonobvious ones according to whether or not the values are beyond the meaningful ranges which should be quite similar to the cases of fault in some conditions.[32] However, outliers can be different from faults in several aspects. First, the outliers can be mainly introduced by reasons such as random variations, incorrect recorded observations and so forth, and hence the major characteristic for outliers should be defined as the randomness. The generated outliers can be

Table 1. Estimated Component Priors by MRPPCA and GMM

	MRPPCA			GMM		
Mode	1	2	3	1	2	3
Estimated	0.3428	0.3238	0.3334	0.3448	0.3220	0.3332
Real	0.3333	0.3333	0.3333	0.3333	0.3333	0.3333

Table 2. Estimated Component Priors by MRPPCA and GMM Under 1% Outliers

	MRPPCA			GMM		
Mode	1	2	3	1	2	3
Estimated	0.3345	0.3300	0.3355	0.0341	0.3195	0.6464
Real	0.3333	0.3333	0.3333	0.3333	0.3333	0.3333

unrelated (independent) with each other both in temporal and intraslice dimension. On the contrary, the presence of fault is usually caused by instrument failures. Besides, the faults can also be propagated from one operating level to the next which makes the corresponding sampling variables dependent with each other in both temporal and intratime-slice dimensions. Second, the outliers are usually introduced by the external causes which do not reflect the potential internal operating conditions. Moreover, the outliers do no harms to the instruments in most cases. Therefore, the outliers can be technically alleviated in the modeling case. Base on these factors, the MRPPCA is constructed with outlier-contaminated conditions since the student *t*-distribution can "ignore" the adverse affects by tolerating them with the heavy tail. On the other side, the faults introduced by internal malfunctioning unit are usually harmful and should be detected and discharged by proper actions. Although we have not considered outliers in the testing dataset in this study, this should be an interesting issue that we may give a highlight in further studies.

Bayesian Monitoring Scheme Based on MRPPCA

By far, we have obtained the MRPPCA model which can be used for robust process modeling under outliers and missing data. In this section, we intend to extend the Bayesian monitoring strategy which was proposed by Ge et al. for mixture factor analysis (MFA) model into the robust models so that the proposed model can be further used for process monitoring.[26] Different from the MFA situation, the monitoring statistics in robust models with *t*-distributions satisfy the noncentral/central F distribution.[33] Specifically, suppose $\mathbf{x} \sim \mathbf{S}(\boldsymbol{\mu}, \boldsymbol{\Lambda}, \nu)$, $\mathbf{x} \in \mathrm{R}^{\mathrm{D}}$, let $\mathbf{Q}$ denotes the corresponding correlation matrix for $\boldsymbol{\Lambda}$, then $\mathbf{x}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{x}/\mathrm{D} \sim \mathrm{ncF}(\mathrm{D}, \nu, \boldsymbol{\mu}^{\mathrm{T}}\mathbf{Q}^{-1}\boldsymbol{\mu}/\mathrm{D})$, where ncF(.) denotes the noncentral F distribution. Furthermore, if $\boldsymbol{\mu}=0$, the distribution is central F which further satisfies $\mathbf{x}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{x}/(\mathrm{D}+\mathbf{x}^{\mathrm{T}}\mathbf{Q}^{-1}\mathbf{x}) \sim \mathrm{Beta}(\mathrm{D}/2, \nu/2)$. However, such upper control limits are often beyond the ideal limits and can be insensitive for abnormal samples.[24] In this article, to relieve the insensitivity from *t*-distribution, we adopt the same scheme as Chen and regard the robust probabilistic model as a way to robustly obtain the projections. In this work, we design the T^2 and SPE statistics as

$$\mathrm{T}^2_{\mathrm{nk}} = \langle \mathbf{t}_{nk} \rangle \langle \mathbf{t}_{nk} \rangle^{\mathrm{T}} \tag{50}$$

$$\mathrm{SPE}_{\mathrm{nk}} = \hat{\mathbf{e}}^{\mathrm{T}}_{\mathrm{nk}} \tau_{\mathrm{k}}^{-1} \hat{\mathbf{e}}_{\mathrm{nk}} \tag{51}$$

Notice that, the SPE statistic is experimentally designed as Eq. 51 instead of $\hat{\mathbf{e}}^{\mathrm{T}}_{\mathrm{nk}} \tau_{\mathrm{k}} \hat{\mathbf{e}}_{\mathrm{nk}}$ so as to get rid of the insensitivity

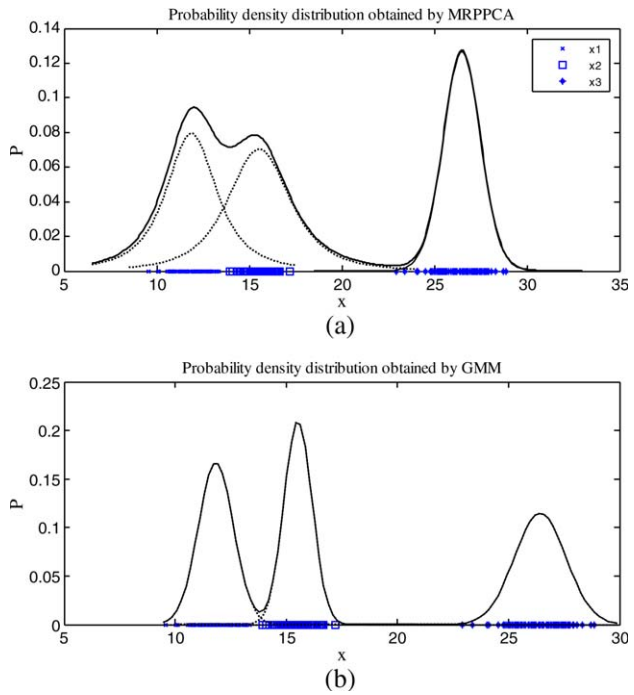


Figure 4. Detailed comparisons of probability density distributions obtained by (a) MRPPCA and (b) GMM.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

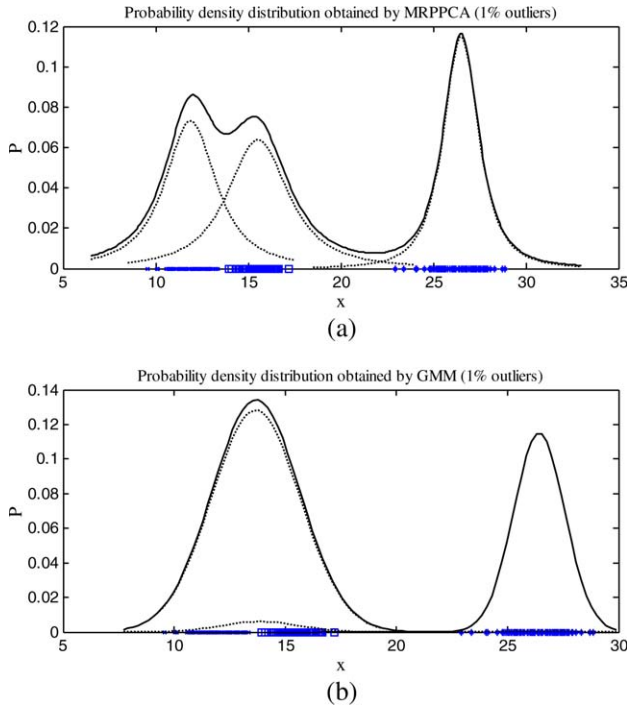


Figure 5. Detailed comparisons of obtained probability density distributions under noise contaminated conditions by (a) MRPPCA and (b) GMM.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

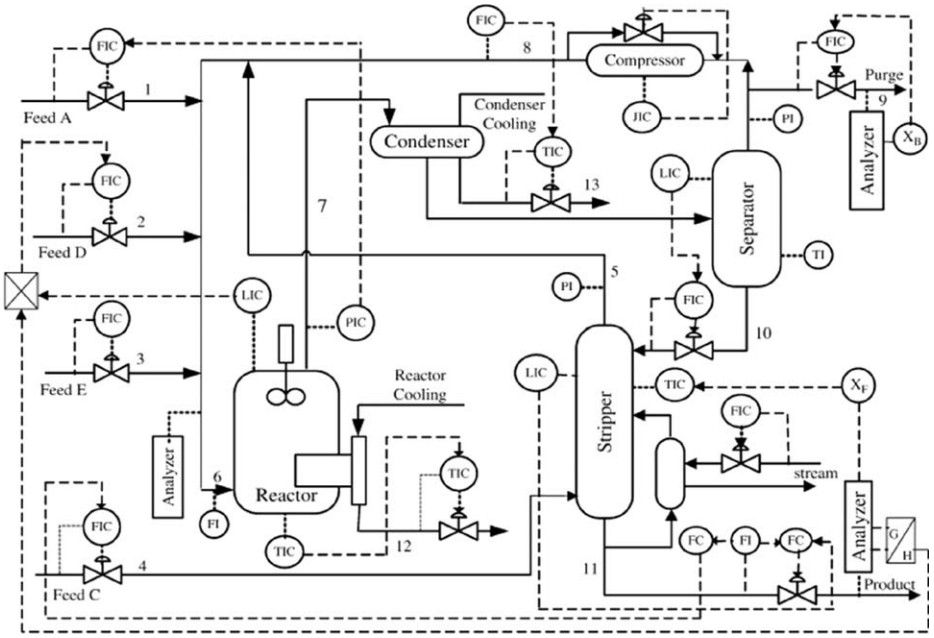


Figure 6. Flow chart of TE process

in some extent as well as obtain better monitoring performance for residual space, where the residual vector $\hat{\mathbf{e}}_{\mathrm{nk}}$ is given as

$$\hat{\mathbf{e}}_{\mathrm{nk}} = \mathbf{x}_{\mathrm{n}} - \boldsymbol{\mu}_{\mathrm{k}} - \mathbf{P}_{\mathrm{k}} \langle \mathbf{t}_{nk} \rangle \tag{52}$$

We retain the control limits for $\mathrm{T}^2_{\mathrm{nk}}$ and $\mathrm{SPE}_{\mathrm{nk}}$ as in PPCA which are given as

$$\mathrm{T}^2_{\mathrm{nk}} \le \mathrm{T}^2_{\mathrm{lim}} = \chi^2_{\alpha}(\mathrm{d}) \tag{53}$$

$$\mathrm{SPE}_{\mathrm{nk}} \le \mathrm{SPE}_{\mathrm{lim}} = \chi^2_{\alpha}(\mathrm{D}) \tag{54}$$

where d is the dimensionality of the latent space, D is the number of process variables, and α is the significance level. Based on the mixture model, $\{\mathrm{T}^2_{\mathrm{nk}}\}_{\mathrm{k}=1}^{\mathrm{K}}$ and $\{\mathrm{SPE}_{\mathrm{nk}}\}_{\mathrm{k}=1}^{\mathrm{K}}$ can be calculated through Eqs. 50 and 51 for each data sample. Then the Bayesian method is applied to softly combine these two index sets into two comprehensive indexes which we name as comprehensive T^2 (CT^2) and the comprehensive squared prediction error (SPE) (CSPE).

Before applying the Bayes rule, the prior probability for the normal event (represented with N) and fault case (represented with F) are first defined as

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{N}) = \mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathrm{N}) = 1 - \alpha = \alpha' \tag{55}$$

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{F}) = \mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathrm{F}) = \alpha \tag{56}$$

Then, according to the Bayes rule, we have

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{F}|\mathbf{x}_{\mathrm{n}}) = \frac{\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathbf{x}_{\mathrm{n}}|\mathrm{F})\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{F})}{\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathbf{x}_{\mathrm{n}}|\mathrm{F})\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{F}) + \mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathbf{x}_{\mathrm{n}}|\mathrm{N})\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{N})} \tag{57}$$

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathrm{F}|\mathbf{x}_{\mathrm{n}}) = \frac{\mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathbf{x}_{\mathrm{n}}|\mathrm{F})\mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathrm{F})}{\mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathbf{x}_{\mathrm{n}}|\mathrm{F})\mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathrm{F}) + \mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathbf{x}_{\mathrm{n}}|\mathrm{N})\mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathrm{N})} \tag{58}$$

Since the degree of fault depends on the distance between $\mathrm{T}^2_{\mathrm{nk}}$ and $\mathrm{T}^2_{\mathrm{lim}}$, $\mathrm{SPE}_{\mathrm{nk}}$, and $\mathrm{SPE}_{\mathrm{lim}}$, and there should be two factors: (1) if the sample is far below the control limit, then the likelihood of fault should be relatively small; (2) if the sample tends to touch or exceed the control limit, then the likelihood should rise steeply to fault. These two factors make the decision looks like a step function which is converted from a (normal) to b (fault) around the switching point c (control limit). In this work, we use the ratio between statistics and control limit as the relative "distance," and the sigmoid function as the "switch function." Therefore, the likelihood can be defined as

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathbf{x}_{\mathrm{n}}|\mathrm{F}) = \exp\left\{-\eta \frac{\mathrm{T}^2_{\mathrm{lim}}}{\mathrm{T}^2_{\mathrm{nk}}}\right\}, \mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathbf{x}_{\mathrm{n}}|\mathrm{F}) = \exp\left\{-\eta \frac{\mathrm{SPE}_{\mathrm{lim}}}{\mathrm{SPE}_{\mathrm{nk}}}\right\} \tag{59}$$

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathbf{x}_{\mathrm{n}}|\mathrm{N}) = \exp\left\{-\eta \frac{\mathrm{T}^2_{nk}}{\mathrm{T}^2_{\mathrm{lim}}}\right\}, \mathrm{p}^{\mathrm{k}}_{\mathrm{SPE}}(\mathbf{x}_{\mathrm{n}}|\mathrm{N}) = \exp\left\{-\eta \frac{\mathrm{SPE}_{nk}}{\mathrm{SPE}_{\mathrm{lim}}}\right\} \tag{60}$$

where η controls the degree of steep and big η makes the sigmoid function close to a step function (see Appendix C for details). Then, the fault posterior within each local model k can be further written as

$$\mathrm{p}^{\mathrm{k}}_{\mathrm{T}^2}(\mathrm{F}|\mathbf{x}_{\mathrm{n}}) = \frac{\alpha \exp\left\{-\eta \frac{\mathrm{T}^2_{\mathrm{lim}}}{\mathrm{T}^2_{nk}}\right\}}{\alpha \exp\left\{-\eta \frac{\mathrm{T}^2_{\mathrm{lim}}}{\mathrm{T}^2_{nk}}\right\} + \alpha' \exp\left\{-\eta \frac{\mathrm{T}^2_{nk}}{\mathrm{T}^2_{\mathrm{lim}}}\right\}} = \frac{1}{1 + \frac{\alpha'}{\alpha} \exp\left\{-\eta \left[\frac{\mathrm{T}^2_{nk}}{\mathrm{T}^2_{\mathrm{lim}}} - \frac{\mathrm{T}^2_{\mathrm{lim}}}{\mathrm{T}^2_{nk}}\right]\right\}} \begin{Bmatrix} <\alpha, \mathrm{T}^2_{nk} < \mathrm{T}^2_{\mathrm{lim}} \\ =\alpha, \mathrm{T}^2_{nk} = \mathrm{T}^2_{\mathrm{lim}} \\ >\alpha, \mathrm{T}^2_{nk} > \mathrm{T}^2_{\mathrm{lim}} \end{Bmatrix} \tag{61}$$

which acts exactly as a sigmoid function. One can further infer that the significance level for each local posterior is also α. Notice that we also call the local posterior as transformed probability since the sigmoid function has

transformed the T_{nk}^2 from a chi-square value into the probability scope. Similarly, the SPE case is

$$p_{SPE}^{k}(F|\mathbf{x}_n)=\frac{\alpha\exp\left\{-\eta\frac{SPE_{lim}}{SPE_{nk}}\right\}}{\alpha\exp\left\{-\eta\frac{SPE_{lim}}{SPE_{nk}}\right\}+\alpha'\exp\left\{-\eta\frac{SPE_{nk}}{SPE_{lim}}\right\}}=\frac{1}{1+\frac{\alpha'}{\alpha}\exp\left\{-\eta\left[\frac{SPE_{nk}}{SPE_{lim}}-\frac{SPE_{lim}}{SPE_{nk}}\right]\right\}}\begin{Bmatrix}<\alpha, SPE_{nk}<SPE_{lim}\\ =\alpha, SPE_{nk}=SPE_{lim}\\ >\alpha, SPE_{nk}>SPE_{lim}\end{Bmatrix} \tag{62}$$

after the conditional posteriors for T^2 and SPE in each local component have been obtained, we can softly synthesize them into two comprehensive indexes by multiplying corresponding local component priors which acts as posterior weights, the final indices for measurement $\mathbf{x}_n$ can be define as

$$CT_n^2=\sum_{k=1}^{K}p(k|\mathbf{x}_n)p_{T^2}^{k}(F|\mathbf{x}_n) \tag{63}$$

$$CSPE_n^2=\sum_{k=1}^{K}p(k|\mathbf{x}_n)p_{SPE}^{k}(F|\mathbf{x}_n) \tag{64}$$

Table 3. Selected Variables for Monitoring

No.	Variable	No.	Variable
1	A feed	9	Product separator temperature
2	D feed	10	Product separator pressure
3	E feed	11	Product separator underflow
4	A and C feed	12	Stripper pressure
5	Recycle flow	13	Stripper temperature
6	Reactor feed rate	14	Stripper steam flow
7	Reactor temperature	15	Reactor cooling water outlet temperature
8	Purge rate	16	Separator cooling water outlet temperature

Here, the control limit for CT_n^2 and $CSPE_n^2$ is defined as the significance level α, therefore, we can conduct the fault detection by monitoring these two comprehensive indexes.

Case Studies

To demonstrate the effectiveness of MRPPCA in process modeling and monitoring, two case studies are given in this section. First, a simple numerical simulation is given to illustrate the advantage of mixture robust model for modeling multimode process with noisy data. In the next part, the case study on Tennessee Eastman process is given to clarify the superiority of MRPPCA on modeling and monitoring the noisy and contaminated industrial process data.

A numerical example

In this study, the objective is to model a three-mode simple system, and the detailed state-space is given as follows[34]

$$\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}=\begin{bmatrix}0.3723 & 0.6815\\0.4890 & 0.2954\\0.9842 & 0.1793\end{bmatrix}\begin{bmatrix}t_1\\t_2\end{bmatrix}+\begin{bmatrix}e_1\\e_2\\e_3\end{bmatrix} \tag{65}$$

where the latent space $[t_1 \quad t_2]^T$ obeys the Gaussian distribution, respectively. And the added noise is assumed as white

Table 4. Monitoring Results for Different Methods Under Noisy Modeling Conditions

	MRPPCA		MPPCA		MRPPCA		MPPCA	
	2% Outliers				5% Outliers			
Fault	CT^2	CSPE	CT^2	CSPE	CT^2	CSPE	CT^2	CSPE
0	0.004	0.049	**0.002**	0.091	**0.000**	0.180	0.001	0.106
1	0.009	**0.007**	0.645	0.029	0.111	**0.031**	0.796	0.040
2	0.028	**0.018**	0.367	0.042	0.039	**0.021**	0.296	0.046
3	0.832	0.773	0.828	**0.722**	0.833	**0.661**	0.830	0.711
4	0.833	0.820	0.833	**0.737**	0.833	**0.686**	0.833	0.719
5	0.679	**0.594**	0.817	0.629	0.692	**0.468**	0.830	0.620
6	**0.000**	**0.000**	0.043	0.025	0.015	**0.003**	0.045	0.030
7	0.574	**0.487**	0.731	0.597	0.609	**0.407**	0.774	0.591
8	0.084	**0.037**	0.679	0.251	0.144	**0.061**	0.750	0.298
9	0.828	0.784	0.826	**0.742**	0.833	**0.646**	0.830	0.731
10	0.706	**0.466**	0.813	0.664	0.755	**0.314**	0.823	0.655
11	0.773	**0.534**	0.829	0.667	0.818	**0.351**	0.831	0.659
12	0.050	**0.030**	0.635	0.237	0.085	**0.051**	0.722	0.264
13	0.066	**0.046**	0.447	0.101	0.082	**0.048**	0.580	0.110
14	0.105	**0.003**	0.821	0.034	0.256	**0.017**	0.833	0.031
15	0.827	0.773	0.829	**0.743**	0.833	**0.640**	0.831	0.728
16	0.781	**0.625**	0.802	0.675	0.798	**0.456**	0.811	0.663
17	0.195	**0.072**	0.556	0.222	0.266	**0.047**	0.644	0.233
18	0.092	**0.080**	0.134	0.096	0.096	**0.087**	0.150	0.095
19	0.833	0.760	0.833	**0.595**	0.833	**0.508**	0.833	0.590
20	0.698	**0.503**	0.826	0.616	0.745	**0.332**	0.830	0.625
21	0.633	**0.403**	0.827	0.668	0.620	**0.315**	0.829	0.708

Table 5. Typical Estimation of Degree of Freedom (DoF)

	DoF	
Conditions	v_1	v_2
No outliers	40.603	77.680
2% outliers	2.746	2.861
5% outliers	1.902	2.120

noise with covariance $0.01I$. The system is assumed to work under three modes which are defined as

$$\text{Mode 1}: \quad t_1 \sim N(10, 0.8) t_2 \sim N(12, 1.3)$$

$$\text{Mode 2}: \quad t_1 \sim N(5, 0.6) t_2 \sim N(20, 0.7)$$

$$\text{Mode 3}: \quad t_1 \sim N(16, 1.5) t_2 \sim N(30, 2.5)$$

To test the method, the system is assumed to work under the above three modes with equal priors. The training set is generated by 100 samples from each mode, then the MRPPCA and the Gaussian mixture model (GMM) are used for data modeling. The component priors for MRPPCA and GMM after training are provided in Table 1. As an illustration, we use the first dimension to demonstrate the modeling performance and the detailed comparisons of the probability density distributions obtained by these two methods are shown in Figure 4a, b, respectively.

From Table 1, we can infer that MRPPCA and GMM have almost the same modeling ability within normal environment where there are no outliers. One can also see from Figure 4 that both of the two mixture models can well handle the multimode process data and the robust mixture model has heavier tails than GMM. To test the modeling ability for both methods under contaminated environment, we randomly add 1% outliers to the training samples, and both models are trained by these contaminated data, the results are given in Table 2. The detailed comparisons of the probability density distributions obtained by these two methods under outlier contaminated conditions are shown in Figure 5. As can be seen from Table 2 and Figure 5, the estimated parameters for MRPPCA are almost unaffected by outliers and are much better that the GMM method. Actually, the degree of freedom for MRPPCA in the contaminated environment in one of the typical simulations is $[1.6135 \quad 2.6119 \quad 1.8215]^T$, which shows a heavily tailed student t-distribution, therefore, the obtained MRPPCA is more robust than the GMM method with light tails.

The Tennessee Eastman benchmark process

The Tennessee Eastman process is a well-known benchmark that has been widely applied to evaluate and compare the

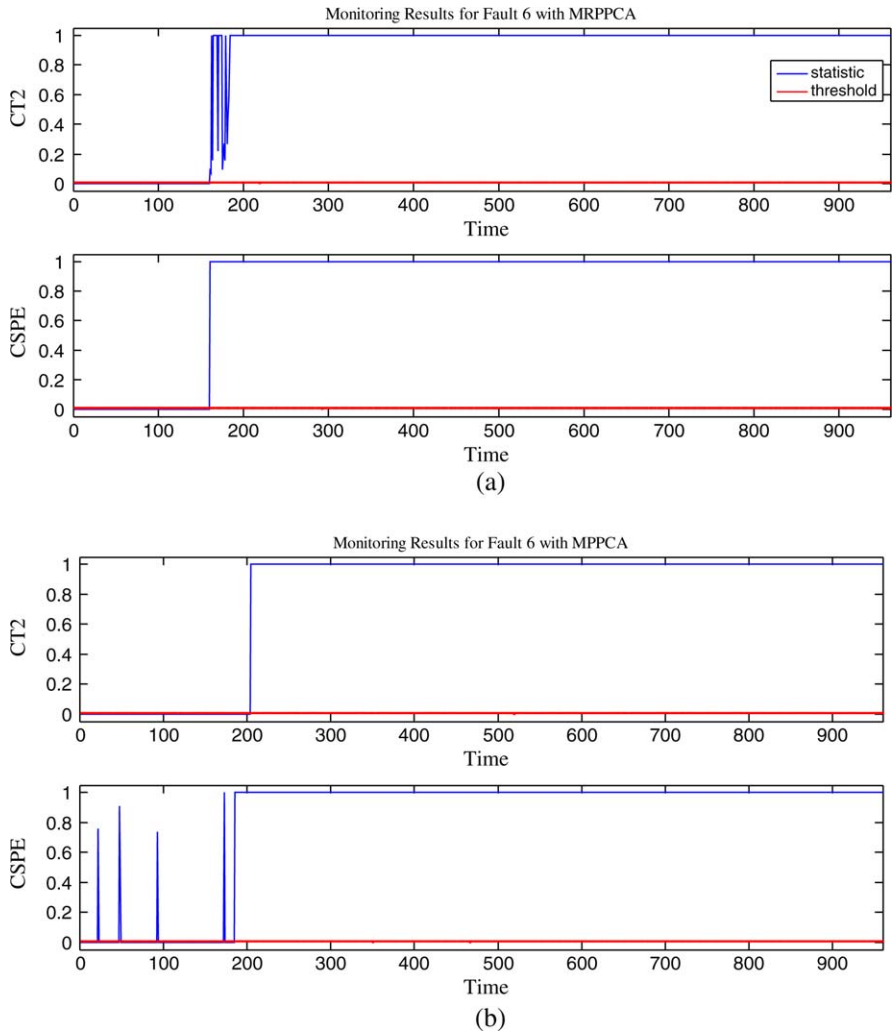


Figure 7. Monitoring results for Fault 6, (a) MRPPCA; (b) MPPCA.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

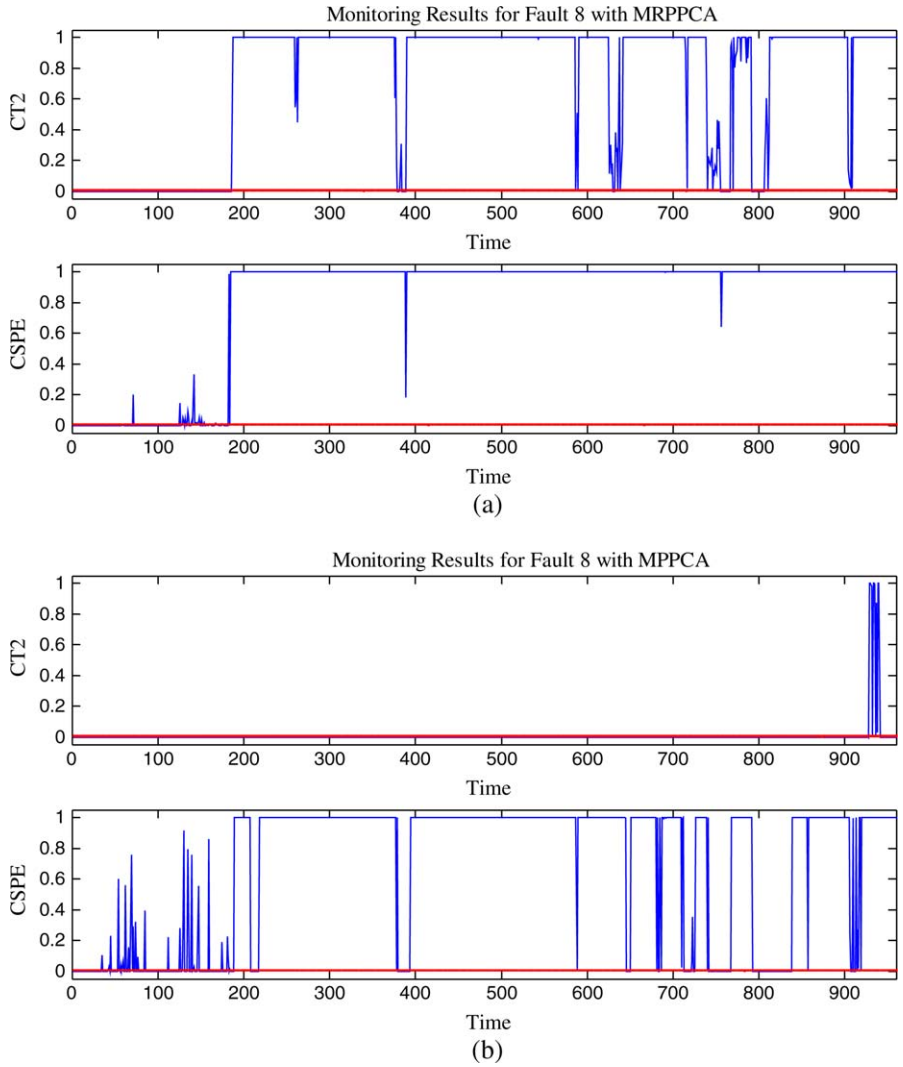


Figure 8. Monitoring results for Fault 8, (a) MRPPCA; (b) MPPCA.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

effectiveness of different process monitoring methods. The detailed information of this process can be found in various works.[35–38] The process consists of 41 measured variables and 12 manipulated variables, the flowchart is depicted in Figure 6. In this work, a set of 16 continuous variables are selected for process monitoring study, which are given in Table 3.

In this work, the sampling interval for collecting the training and testing data is set as 3 min and the normal data is

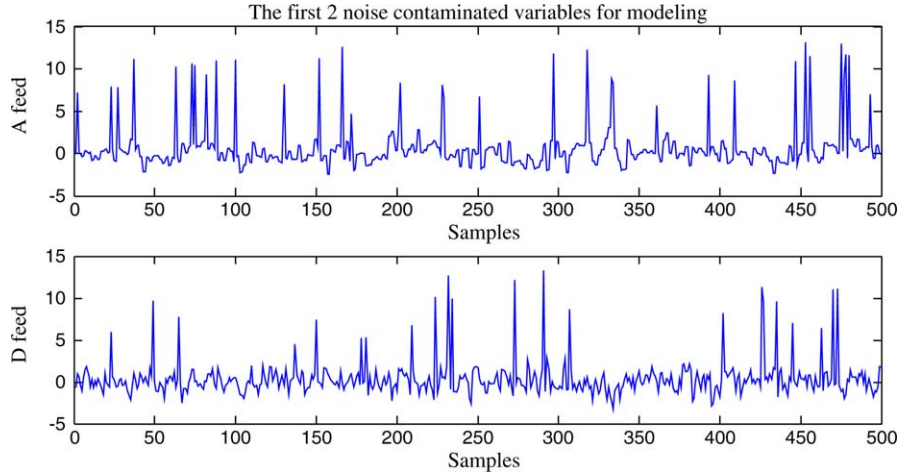


Figure 9. Illustration of the 2 variables in the training data (contaminated by 5% outliers).

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

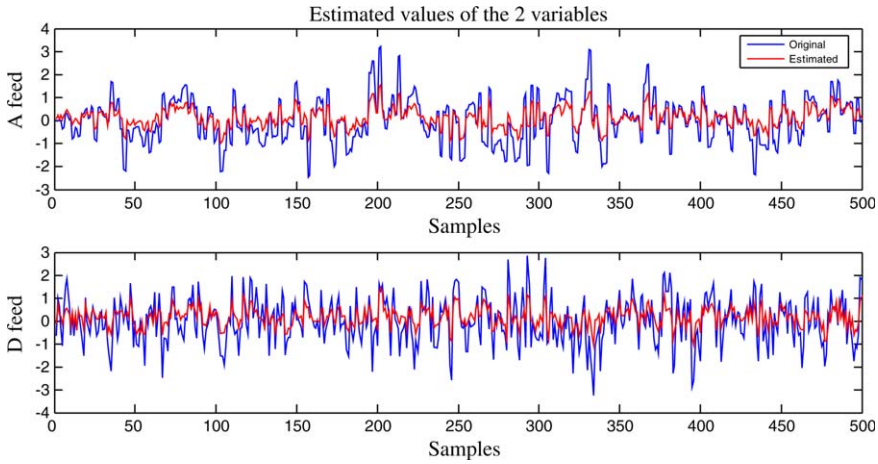


Figure 10. Estimated values with MRPPCA for A feed and D feed.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

collected for training the model as well as building the comprehensive control limits. A total of 21 faults are simulated so as to test the monitoring behavior. The training set is a collection of 500 normal samples, while the test set for each fault is a collection of 960 samples and the fault is introduced by the 161th sampling time. All the data we use in this work can be downloaded from Braatz's research group web link: http://web.mit.edu/braatzgroup/links.html. To demonstrate the robust superiority of the proposed method, we make a comparison with mixture of probabilistic PCA (MPPCA). Both of the two mixture models contain two components and the latent dimensions are experimentally set to five. To verify the robust modeling ability of the proposed method, 2% outliers and 5% outliers are randomly added to the training data, respectively, afterward, the original test data is used to check out the monitoring ability in outlier case. For the sake of simplicity, we first normalize the collected samples by means and variances, and then the outliers are randomly induced. The monitoring results for both methods are presented in Table 4, and the estimation for degree of freedoms for the mixture robust model is given in Table 5. During the simulations, the significance level α is set to 0.01, and the steep parameter for sigmoid function is set as $\eta = 100$. The overall monitoring results with respect to error rates for all methods are given in Table 4 where the normal situation is named as Fault 0. Notice that the error means both the alarm missing and false alarm conditions. Each statistic is obtained by averaging five Monte Carlo simulations and the statistics with the best performance are highlighted with bolded fonts.

While comparable monitoring results have been observed through the two methods, one can easily judge from Table 4 that the robust model is more effective and shows much more outstanding performances than the nonrobust method. Furthermore, as shown in Table 5, since the degree of freedoms for student t-distributions become smaller in the outlier contaminated conditions, the constructed model becomes much more robust to outliers. On the contrary, typical latent variable models do not have the scheme to deal with the outliers, which in terms lead to the deterioration of monitoring performance. As an illustration, we show the monitoring

Table 6. Monitoring Results of MRPPCA Under Different Missing Data Modeling Conditions

	2% m		5% m		10% m		15% m	
Fault	CT^2	CSPE	CT^2	CSPE	CT^2	CSPE	CT^2	CSPE
0	0.016	0.006	0.016	0.001	0.015	0.001	0.005	0.002
1	0.011	0.007	0.012	0.007	0.012	0.007	0.011	0.007
2	0.025	0.019	0.024	0.019	0.025	0.019	0.026	0.018
3	0.819	0.832	0.822	0.829	0.821	0.828	0.828	0.821
4	0.828	0.832	0.829	0.832	0.830	0.832	0.832	0.830
5	0.650	0.674	0.654	0.667	0.651	0.655	0.663	0.640
6	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
7	0.535	0.585	0.533	0.576	0.529	0.562	0.550	0.533
8	0.071	0.037	0.066	0.035	0.072	0.030	0.077	0.028
9	0.802	0.831	0.798	0.831	0.805	0.828	0.818	0.818
10	0.611	0.679	0.614	0.664	0.608	0.636	0.650	0.578
11	0.738	0.675	0.737	0.659	0.741	0.637	0.764	0.599
12	0.035	0.048	0.035	0.042	0.039	0.034	0.038	0.027
13	0.051	0.049	0.052	0.049	0.054	0.045	0.063	0.045
14	0.134	0.001	0.146	0.001	0.149	0.002	0.178	0.002
15	0.804	0.831	0.805	0.827	0.798	0.822	0.816	0.802
16	0.707	0.753	0.708	0.754	0.699	0.741	0.733	0.704
17	0.195	0.120	0.203	0.115	0.203	0.105	0.208	0.096
18	0.091	0.088	0.091	0.087	0.092	0.087	0.093	0.085
19	0.830	0.829	0.829	0.826	0.831	0.824	0.833	0.817
20	0.617	0.657	0.628	0.647	0.635	0.635	0.686	0.586
21	0.578	0.532	0.581	0.521	0.590	0.505	0.664	0.458

Table 7. Monitoring Results of MRPPCA Under Different Corrupted Modeling Conditions

	2%m+2%o		5%m+2%o		5%m+5%o	
Fault	CT^2	CSPE	CT^2	CSPE	CT^2	CSPE
0	0.005	0.040	0.003	0.054	0.001	0.251
1	0.009	0.007	0.010	0.008	0.010	0.038
2	0.029	0.018	0.029	0.017	0.035	0.025
3	0.831	0.794	0.831	0.771	0.833	0.612
4	0.833	0.820	0.833	0.813	0.833	0.616
5	0.672	0.606	0.678	0.583	0.696	0.427
6	0.000	0.000	0.001	0.000	0.008	0.006
7	0.552	0.488	0.558	0.480	0.617	0.366
8	0.071	0.035	0.077	0.042	0.133	0.072
9	0.825	0.788	0.828	0.769	0.833	0.599
10	0.677	0.477	0.688	0.451	0.769	0.261
11	0.775	0.527	0.782	0.506	0.806	0.318
12	0.045	0.034	0.050	0.039	0.106	0.067
13	0.059	0.046	0.061	0.045	0.089	0.046
14	0.127	0.003	0.126	0.003	0.199	0.026
15	0.823	0.773	0.826	0.760	0.833	0.591
16	0.755	0.625	0.768	0.600	0.815	0.379
17	0.197	0.072	0.203	0.067	0.263	0.056
18	0.093	0.081	0.093	0.078	0.098	0.090
19	0.833	0.768	0.833	0.747	0.833	0.456
20	0.677	0.510	0.689	0.489	0.755	0.297
21	0.582	0.425	0.597	0.393	0.700	0.296
mean	0.430	0.361	0.435	0.351	0.467	0.268

("m" represents "missing," "o" represents "outlier")

results for fault 6 and fault 8, to make a comparison between MRPPCA and MPPCA, which are shown in Figures 7 and 8, respectively.

For fault 6, both methods can successfully detect faults; however, the MRPPCA method performs much better since both statistics can detect faults exactly at the 161th time as can be seen in Figure 7. In contrast, MPPCA has a significant detection delay and also shows a number of false alarms. Precisely, the CT^2 statistic cannot detect faults until the 205th sampling time, and the CSPE statistic can only detect faults at 173th sampling time. For the case of fault 8, the MRPPCA method reveals its ability to successfully detect more faults and has fewer false alarms. While for MPPCA, conversely, both statistics are fairly ineffective since the CT^2 can only detect faults at nearly the ending of the process. Although the CSPE for MPPCA shows better results, it gives more false and missing alarms. During the training phase, the MPPCA assumes no robust mechanisms and hence can be susceptible to outliers. Therefore, the obtained MPPCA should be skewed from normal samples regions and is inclined to accept more abnormal cases as normal ones which result in more significant detection delays. The SPE case is totally on the opposite side, the skewed latent space projection for MPPCA is so conservative that the corresponding residual has been amplified which directly leads to more false alarms. As a further investigation, the robust model constructed under 5% outliers is also used to "re-estimate" the training data. The original uncontaminated training data is used as the test data for MRPPCA, the re-estimated values are compared with the original data. Notice that in the mixture probabilistic model, each reconstructed values within a single mixture model is weighted with the corresponding mixture weight and the entire weighted form is used as the final estimation. For illustrations, the first two outlier-contaminated training variables (A feed and D feed in normalized form) in the training set are plotted in Figure 9, the reconstructed values and the initial uncontaminated values are both shown in Figure 10. From the two figures, one can easily infer that the estimated values are quite similar to those of the data samples in the training dataset.

Furthermore, to investigate the MRPPCA modeling performance under missing data cases, we randomly generate 2,

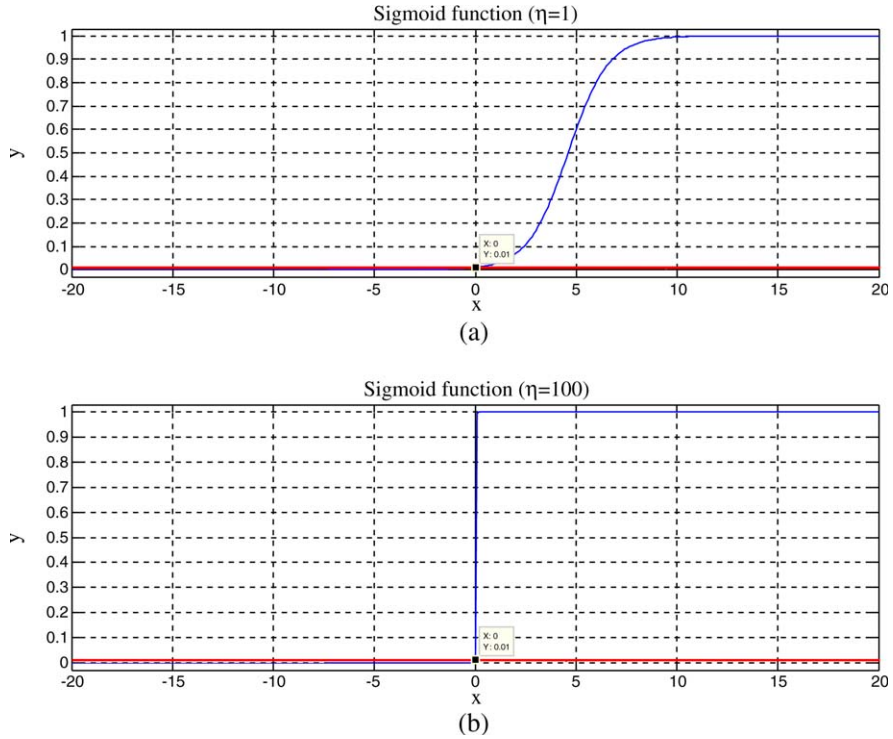


Figure 11. Sigmoid function for (a) $\eta = 1$, and (b) $\eta = 100$.

[Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

5, 10, and 15% missing data in the training data. Then the test set is used to evaluate the monitoring performance of the constructed model. Detailed monitoring results for 2–15% missing data modeling conditions are given in Table 6. Same as above, all results are averaged under five random runs. One can easily find that the MRPPCA method is also effective under the missing data modeling environment, as the missing rate increases from 2 to 15%, the monitoring performance does not change significantly, which clearly verifies the robustness for MRPPCA.

Finally, we add 2% (or 5%) outliers and 2% (or 5%) missing data simultaneously into the training set to verify the robust method. Monitoring results for different contaminated modeling conditions are given in Table 7. As can be easily inferred from Table 7, the performance of MRPPCA does not change much under various data contaminated conditions. Specifically, the mean performance of CT^2 deteriorates slightly while the mean of CSPE turns to a slightly better performance, this further indicates that a contaminated training data can skew the obtained principal subspaces; however, due to the robustness of the model, the negative effects might be relieved in some extent in the residual subspaces. By comparing Tables 4, 6, and 7, one can easily conclude that simply missing data along shows insignificant impacts on robust modeling phase which clearly demonstrate the effectiveness of the proposed partially updating mechanism. Furthermore, the comparison also indicates that the tolerance ability for outliers can be effective within a small portion of outliers since the adaptive ability of the heavy tail which is regulated by the degree of freedom should be limited (see also Table 5). Anyhow, the results have shown that MRPPCA is more reliable in outlier/missing data contaminated modeling conditions and hence can enhance the performance in the monitoring phases.

Conclusions

In this work, a mixture form of robust probabilistic PCA has been developed. Compared to the traditional mixture models like GMM and MPPCA, MRPPCA can conduct robust modeling in noisy, outlier contaminated, and missing data conditions, which are common cases in practice. To perform process monitoring, the Bayesian soft fusion strategy has been introduced, which is constructed by converting the traditional monitoring statistics into probabilities by the sigmoid formulation so that Bayesian inference can be performed. The modeling and monitoring ability of the proposed method is evaluated through two case studies. Simulation results of two examples indicate the effectiveness and robustness of the developed MRPPCA model.

For outlooks, the proposed method can be further extended for fault diagnosis or fault classification. To be specific, in this work, the sigmoid function is used to describe the probabilities of two classes (normal or fault). With slight modifications, the multiclass formulation of sigmoid function can be induced so that the model can deal with the fault classification problem. Another potential issue is to deal with the dynamic property of process data. One may consider to carry out dynamic robust modeling so that the obtained model can deal with the outlier contaminated process data under a dynamic framework, which could be more applicable to industrial processes. Besides, the component number of the developed model has been experimentally determined. As a further modification, an adaptive modification strategy should be derived in the following work to try to automatically select the proper component number according to criteria such as the Akaike information criterion (AIC), the Bayesian information criterion (BIC), and the minimum description length (MDL).

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (NSFC) (61370029), Project National 973 (2012CB720500), and the Fundamental Research Funds for the Central Universities (2013QNA5016).

Literature Cited

1. Ge ZQ, Song ZH. *Multivariate Statistical Process Control: Process Monitoring Methods and Applications*. London: Springer, 2013.
2. Kruger U, Xie L. *Statistical Monitoring of Complex Multivariate Processes*. West Sussex: Wiley, 2012.
3. Qin SJ. Survey on data-driven industrial process monitoring and diagnosis. *Annu Rev Control*. 2012;36:220–234.
4. Ge ZQ, Song ZH, Gao FR. Review of recent research on data-based process monitoring. *Ind Eng Chem Res*. 2013;52:3543–3562.
5. Jiang QC, Yan XF, Zhao W. Fault detection and diagnosis in chemical processes using sensitive principal component analysis. *Ind Eng Chem Res*. 2013;52:1635–1644.
6. Kruger U, Antory D, Hahn J, Irwin GW, McCullough G. Introduction of a nonlinearity measure for principal component models. *Comput Chem Eng*. 2005;29:2355–2362.
7. Venkatasubramanian V, Rengaswamy R, Kavuri SN, Yin K. A review of process fault detection and diagnosis Part III: Process history based methods. *Comput Chem Eng*. 2003;27:327–346.
8. Kano M, Nakagawa Y. Data-based process monitoring, process control, and quality improvement: recent developments and applications in steel industry. *Comput Chem Eng*. 2008;32:12–24.
9. Jiang QC, Yan XF. Chemical processes monitoring based on weighted principal component analysis and its application. *Chemometr Intell Lab Syst*. 2012;119:11–20.
10. Tipping ME, Bishop CM. Probabilistic principal component analysis. *J R Stat Soc Series B Stat Methodol*. 1999;61:611–622.
11. Ge ZQ, Song ZH. Kernel. Generalization of PPCA for nonlinear probabilistic monitoring. *Ind Eng Chem Res*. 2010;49:11832–11836.
12. Chen T, Sun Y. Probabilistic contribution analysis for statistical process monitoring: a missing variable approach. *Control Eng Pract*. 2009;17:469–477.
13. Kim D, Lee IB. Process monitoring based on probabilistic PCA. *Chemometr Intell Lab Syst*. 2003;67:109–123.
14. Ge ZQ, Song ZH. Mixture Bayesian regularization method of PPCA for multimode process monitoring. *AIChE J*. 2010;56:2838–2849.
15. De La Torre F, Black MJ. A framework for robust subspace learning. *Int J Comput Vis*. 2003;54:117–142.
16. Cousineau D, Chartier S. Outliers detection and treatment: a review. *Int J Psychol Res*. 2010;3:58–67.
17. Yuan KH, Bentler PM. Effect of outliers on estimators and tests in covariance structure analysis. *Br J Math Stat Psychol*. 2001;54:161–175.
18. Hsu CC, Chen MC, Chen LS. A novel process monitoring approach with dynamic independent component analysis. *Control Eng Pract*. 2010;18:242–253.
19. Liu H, Shah S, Jiang W. On-line outlier detection and data cleaning. *Comput Chem Eng*. 2004;28:1635–1647.
20. Hodge VJ, Austin J. A survey of outlier detection methodologies. *Artif Intell Rev*. 2004;22:85–126.
21. Chiang LH, Pell RJ, Seasholtz MB. Exploring process data with the use of robust outlier detection algorithms. *J Process Control*. 2003;13:437–449.
22. Archambeau C, Delannay N, Verleysen M. Robust probabilistic projections. *Proceedings of the 23rd International Conference on Machine Learning*. New York: ACM, 2006;33–40.
23. Archambeau C, Delannay N, Verleysen M. Mixtures of robust probabilistic principal component analyzers. *Neurocomputing*. 2008;71:1274–1282.
24. Chen T, Martin E, Montague G. Robust probabilistic PCA with missing data and contribution analysis for outlier detection. *Comput Stat Data Anal*. 2009;53:3706–3716.
25. Ge ZQ, Song ZH. Multimode process monitoring based on Bayesian method. *J Chemometr*. 2009;23:636–650.
26. Ge ZQ, Song ZH. Maximum-likelihood mixture factor analysis model and its application for process monitoring. *Chemometr Intell Lab Syst*. 2010;102:53–61.

27. Svensén M, Bishop CM. Robust Bayesian mixture modelling. *Neurocomputing*. 2005;64:235–252.
28. Imtiaz S, Shah S. Treatment of missing values in process data analysis. *Can J Chem Eng*. 2008;86:838–858.
29. Zhang ZD, Zhu JL, Pan F. Fault detection and diagnosis for data incomplete industrial systems with new Bayesian network approach. *J Syst Eng Electron*. 2013;24:500–511.
30. Khatibisepehr S, Huang B. Dealing with irregular data in soft sensors: Bayesian method and comparative study. *Ind Eng Chem Res*. 2008;47:8713–8723.
31. Schafer JL, Graham JW. Missing data: our view of the state of the art. *Psychol Methods*. 2002;7:147–177.
32. Kadlec P, Gabrys B, Strandt S. Data-driven soft sensors in the process industry, *Comput Chem Eng*. 2009;33:795–814.
33. Kotz S, Nadarajah S. *Multivariate T-Distributions and their Applications*. Cambridge: Cambridge University Press, 2004.
34. Yu J, Qin SJ. Multimode process monitoring with Bayesian inference-based finite Gaussian mixture models. *AIChE J*. 2008;54:1811–1829.
35. Ge ZQ, Song ZH. Improved kernel PCA-based monitoring approach for nonlinear processes. *Chem Eng Sci*. 2009;64:2245–2255.
36. Ge ZQ, Gao FR, Song ZH. Mixture probabilistic PCR model for soft sensing of multimode processes. *Chem Intell Lab Syst*. 2011;105:91–105.
37. Yin S, Ding SX, Haghani A, Hao HY, Zhang P. A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process. *J Process Control*. 2012;22:1567–1581.
38. Yu J. A particle filter driven dynamic Gaussian mixture model approach for complex process monitoring and fault diagnosis. *J Process Control*. 2012;22:778–788.

Appendix A

In this part, we try to make clear the mechanism behind the fact that the student *t*-distribution should be an infinite integral for Gaussian distribution with Gamma as the prior.

First, we give and prove a useful equation that will be used later. The following equation stands as

$$\int_0^\infty t^a e^{-(\alpha t)^\beta} dt = \frac{\Gamma\left(\frac{a+1}{\beta}\right)}{\beta \alpha^{a+1}} \tag{A1}$$

where $\Gamma(.)$ is the Gamma function.

Proof: Let $x=(\alpha t)^\beta$, then we have

$$dx = \beta \alpha^\beta t^{\beta-1} dt$$
$$t = \alpha^{-1} x^{1/\beta}$$

therefore, Eq. A1 can be further written as

$$\int_0^\infty e^{-x} \beta^{-1} \alpha^{-(a+1)} x^{(a+1)\beta^{-1}-1} dx$$
$$= \frac{1}{\beta \alpha^{a+1}} \int_0^\infty e^{-x} x^{(a+1)\beta^{-1}-1} dx$$
$$= \frac{\Gamma\left(\frac{a+1}{\beta}\right)}{\beta \alpha^{a+1}}$$

Now we can further extend the Eq. 5 as

$$\begin{aligned} S(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Lambda}, \nu) &= \int_0^\infty N(\mathbf{x}|\boldsymbol{\mu}, u\boldsymbol{\Lambda}) Ga(u|\nu/2, \nu/2) du \\ &= \int_0^\infty (2\pi)^{-D/2} |u\boldsymbol{\Lambda}|^{1/2} e^{-0.5(\mathbf{x}-\boldsymbol{\mu})^T u\boldsymbol{\Lambda}(\mathbf{x}-\boldsymbol{\mu})} \frac{(\nu/2)^{\nu/2} (u)^{\nu/2-1}}{\Gamma(\nu/2)} e^{-\nu u/2} du \\ &= (2\pi)^{-D/2} \frac{(\nu/2)^{\nu/2}}{\Gamma(\nu/2)} |\boldsymbol{\Lambda}|^{1/2} \int_0^\infty u^{(\nu+D)/2-1} e^{-0.5(\mathbf{x}-\boldsymbol{\mu})^T u\boldsymbol{\Lambda}(\mathbf{x}-\boldsymbol{\mu})-\nu u/2} du \\ &= (2\pi)^{-D/2} \frac{(\nu/2)^{\nu/2}}{\Gamma(\nu/2)} |\boldsymbol{\Lambda}|^{1/2} \int_0^\infty u^{(\nu+D)/2-1} e^{-u(m+\nu)/2} du \end{aligned} \tag{A2}$$

where $m=(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Lambda}(\mathbf{x}-\boldsymbol{\mu})$, notice that $|u\boldsymbol{\Lambda}|^{1/2}=u^{D/2}|\boldsymbol{\Lambda}|^{1/2}$. Using Eq. A1, the above equation can be simplified as

$$\begin{aligned} \text{Eq .A2} &= (2\pi)^{-D/2} \frac{(\nu/2)^{\nu/2}}{\Gamma(\nu/2)} |\boldsymbol{\Lambda}|^{1/2} \frac{\Gamma((\nu+D)/2)}{((m+\nu)/2)^{(\nu+D)/2}} \\ &= (2\pi)^{-D/2} \frac{(\nu/2)^{\nu/2}}{\Gamma(\nu/2)} |\boldsymbol{\Lambda}|^{1/2} \Gamma((\nu+D)/2) \left(\frac{\nu}{2}\right)^{-\frac{\nu+D}{2}} \left(1+\frac{m}{\nu}\right)^{-\frac{\nu+D}{2}} \\ &= \frac{\Gamma((\nu+D)/2)|\boldsymbol{\Lambda}|^{1/2}}{\Gamma\left(\frac{\nu}{2}\right)(\pi\nu)^{D/2}} \left(1+\frac{m}{\nu}\right)^{-\frac{\nu+D}{2}} \end{aligned} \tag{A3}$$

which is exactly the student *t*-distribution.

Appendix B

To obtain the Bayes formulas for Gaussian variables, the direct computation of Eq. 23 seems to be complex and unnecessary. Since the conditional probability and marginal probability for Gaussians are still Gaussians, therefore, in this part we resort to the method of undetermined coefficients that the mean and precision for Gaussian distribution can be determined by the

exponent terms.[35] For instance, suppose $\mathbf{x} \sim \mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Lambda})$, then the exponent terms of $\mathbf{x}$ can be written as

$$\begin{aligned} &-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Lambda}(\mathbf{x}-\boldsymbol{\mu}) \\ &=-\frac{1}{2}\mathbf{x}^{\mathrm{T}}\boldsymbol{\Lambda}\mathbf{x}+\mathbf{x}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{\mu}+\mathrm{Const} \end{aligned} \tag{B1}$$

where the term "Const" denotes the constant. Therefore, the precision can be determined by the second-order term, while the mean can be inferred from the linear first-order term.

First of all, Eqs. 13 and 14 are abbreviated as

$$\mathbf{P}(\mathbf{t}_{nk}|u_{nk},\boldsymbol{\theta}) = \mathrm{N}(\mathbf{t}_{nk}|\mathbf{0}, u_{nk}\mathbf{I}_{\mathrm{k}}) \tag{B2}$$

$$\mathbf{P}(\mathbf{x}_{\mathrm{nk}}|u_{nk},\mathbf{t}_{nk},\boldsymbol{\theta}) = \mathrm{N}(\mathbf{x}_{\mathrm{nk}}|\mathbf{P}_{\mathrm{k}}\mathbf{t}_{nk}+\boldsymbol{\mu}_{\mathrm{k}}, u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}) \tag{B3}$$

in which we omit the indicator z_{nk} for simplicity.

Suppose the combined random variable $\mathbf{g}_{\mathrm{nk}} = \begin{pmatrix} \mathbf{t}_{nk} \\ \mathbf{x}_{\mathrm{nk}} \end{pmatrix}$, the corresponding mean and precision should be $\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{g}} = \begin{pmatrix} \boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}} \\ \boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}} \end{pmatrix}$, $\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathrm{g}} = \begin{pmatrix} \boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}} & \boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{tx}} \\ \boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{xt}} & \boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{x}} \end{pmatrix}$. Then, the exponent terms for combined probability can be written as

$$\begin{aligned} &-\frac{1}{2}\left(\mathbf{g}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{g}}\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathrm{g}}\left(\mathbf{g}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{g}}\right) \\ &=-\frac{1}{2}[\left(\mathbf{t}_{nk}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}}\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}}\left(\mathbf{t}_{nk}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}}\right)+\left(\mathbf{t}_{nk}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}}\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{tx}}\left(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}}\right) \\ &+\left(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}}\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{xt}}\left(\mathbf{t}_{nk}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}}\right)+\left(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}}\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{x}}\left(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}}\right)] \end{aligned} \tag{B4}$$

Since the conditional probability for $\mathbf{t}$ which is still Gaussian can be viewed as the probability function that fix $\mathbf{x}$, therefore, the coefficient of second-order term and of the first-order term with respect to $\mathbf{t}_{nk}$ should satisfy[35]

$$\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}|\mathbf{x}} = (\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}|\mathbf{x}})^{-1}[\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}}\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}}-\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{xt}}(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}})] \tag{B5}$$

$$\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}|\mathbf{x}} = \boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}} \tag{B6}$$

To compute the terms in Eqs. B5 and B6, using Eqs. B2 and B3, we have the logarithm of likelihood as

$$\begin{aligned} &\ln[\mathbf{P}(\mathbf{t}_{nk}|\mathbf{x}_{\mathrm{nk}},u_{nk},\boldsymbol{\theta})] = \ln[\mathbf{P}(\mathbf{x}_{\mathrm{nk}}|u_{nk},\mathbf{t}_{nk},\boldsymbol{\theta})]+\ln[\mathbf{P}(\mathbf{t}_{nk}|u_{nk},\boldsymbol{\theta})] \\ &=-\frac{1}{2}(\mathbf{t}_{nk})^{\mathrm{T}}u_{nk}\mathbf{I}_{\mathrm{k}}(\mathbf{t}_{nk})-\frac{1}{2}[\mathbf{x}_{\mathrm{nk}}-(\mathbf{P}_{\mathrm{k}}\mathbf{t}_{nk}+\boldsymbol{\mu}_{\mathrm{k}})]^{\mathrm{T}}u_{nk} \\ &\quad\times\boldsymbol{\Lambda}_{\mathrm{k}}[\mathbf{x}_{\mathrm{nk}}-(\mathbf{P}_{\mathrm{k}}\mathbf{t}_{nk}+\boldsymbol{\mu}_{\mathrm{k}})]+\mathrm{const}. \end{aligned} \tag{B7}$$

the whole second-order terms can be obtained from Eq. 76 as

$$\begin{aligned} &-\frac{1}{2}(\mathbf{t}_{nk})^{\mathrm{T}}\left(u_{nk}\mathbf{I}_{\mathrm{k}}+\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}}\right)\mathbf{t}_{nk}-\frac{1}{2}\mathbf{x}_{\mathrm{nk}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{x}_{\mathrm{nk}} \\ &+\frac{1}{2}\mathbf{x}_{\mathrm{nk}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}}\mathbf{x}_{\mathrm{nk}}+\frac{1}{2}(\mathbf{t}_{nk})^{\mathrm{T}}\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{x}_{\mathrm{nk}} \\ &=-\frac{1}{2}\begin{pmatrix}\mathbf{t}_{nk}\\ \mathbf{x}_{\mathrm{nk}}\end{pmatrix}^{\mathrm{T}}\begin{pmatrix}u_{nk}\mathbf{I}_{\mathrm{k}}+\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}} & -\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\\ -u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}} & u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\end{pmatrix}\begin{pmatrix}\mathbf{t}_{nk}\\ \mathbf{x}_{\mathrm{nk}}\end{pmatrix} \end{aligned} \tag{B8}$$

Thus we obtain precision as $\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathrm{g}} = \begin{pmatrix} u_{nk}\mathbf{I}_{\mathrm{k}}+\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}} & -\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}} \\ -u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}} & u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}} \end{pmatrix}$. Using the factor in Eq. B1 that the coefficient of the first-order term equals to $\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathrm{g}}\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{g}}$, therefore, we have the mean as

$$\boldsymbol{\mu}_{\mathrm{k}}^{\mathrm{g}} = (\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathrm{g}})^{-1}\begin{pmatrix}-\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\boldsymbol{\mu}_{\mathrm{k}}\\ u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\boldsymbol{\mu}_{\mathrm{k}}\end{pmatrix} = \begin{pmatrix}\mathbf{0}\\ \boldsymbol{\mu}_{\mathrm{k}}\end{pmatrix} = \begin{pmatrix}\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}}\\ \boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}}\end{pmatrix} \tag{B9}$$

According to Eqs. B5, B6, and B9, we obtain

$$\boldsymbol{\Lambda}_{\mathrm{k}}^{\mathbf{t}|\mathbf{x}} = u_{nk}\mathbf{I}_{\mathrm{k}}+\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}} = u_{nk}\mathrm{B}_{\mathrm{k}} \tag{B10}$$

$$\begin{aligned} \boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{t}|\mathbf{x}} &= (u_{nk}\mathbf{I}_{\mathrm{k}}+\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{P}_{\mathrm{k}})^{-1}\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}u_{nk}\boldsymbol{\Lambda}_{\mathrm{k}}\left(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}^{\mathbf{x}}\right) \\ &= \mathrm{B}_{\mathrm{k}}^{-1}\mathbf{P}_{\mathrm{k}}^{\mathrm{T}}\boldsymbol{\Lambda}_{\mathrm{k}}(\mathbf{x}_{\mathrm{nk}}-\boldsymbol{\mu}_{\mathrm{k}}) \end{aligned} \tag{B11}$$

Appendix C

The normally designed sigmoid function in this work can be given as

$$\mathrm{y} = \frac{1}{1+\frac{\alpha'}{\alpha}\exp(-\eta\mathbf{x})} \tag{C1}$$

where $\alpha = 0.01$. To see how η controls the degree of steep, we give the shapes of sigmoid function when $\eta = 1$ and $\eta = 100$, see Figure 11a, b, the red line is the threshold equals to 0.01. We can easily judge that

$$\mathrm{y}\begin{cases} \in [0, 0.01), \mathrm{x} < 0 (\mathrm{T}_{\mathrm{nk}}^2 < \mathrm{T}_{\mathrm{lim}}^2) \\ = 0.01, \mathrm{x} = 0 (\mathrm{T}_{\mathrm{nk}}^2 = \mathrm{T}_{\mathrm{lim}}^2) \\ \in (0.01, 1], \mathrm{x} > 0 (\mathrm{T}_{\mathrm{nk}}^2 > \mathrm{T}_{\mathrm{lim}}^2) \end{cases}$$

therefore, a bigger η makes the sigmoid function more like a step function, when the statistics exceeds the control limit, the transformed probability exceeds the threshold.

Manuscript received Nov. 30, 2013, and revision received Jan. 15, 2014.