University of Reading

Nonlinear process monitoring using a mixture of probabilistic PCA with clusterings

Article

Accepted Version

Creative Commons: Attribution-Noncommercial-No Derivative Works 4.0

Zhang, J., Chen, M. and Hong, X. (2021) Nonlinear process monitoring using a mixture of probabilistic PCA with clusterings. Neurocomputing, 458. pp. 319-326. ISSN 0925-2312 doi: https://doi.org/10.1016/j.neucom.2021.06.039 Available at https://centaur.reading.ac.uk/99132/

It is advisable to refer to the publisher's version if you intend to cite from the work. See Guidance on citing.

To link to this article DOI: http://dx.doi.org/10.1016/j.neucom.2021.06.039

Publisher: Elsevier

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the End User Agreement.

www.reading.ac.uk/centaur

CentAUR

University of Reading

Central Archive at the University of Reading

Reading's research outputs online

Nonlinear process monitoring using a mixture of probabilistic PCA with clusterings

Jingxin Zhang[a,*], Maoyin Chen[a] and Xia Hong[b,*]

[a]Department of Automation, Tsinghua University, Beijing 100084, China
[b]Department of Computer Science, School of Mathematical, Physical and Computational Sciences, University of Reading, RG6 6AY, U.K.

ARTICLE INFO

Keywords:
Process monitoring
SVD
probabilistic PCA
clustering

ABSTRACT

Motivated by mixture of probabilistic principal component analysis (PCA), which is time-consuming due to expectation maximization, this paper investigates a novel mixture of probabilistic PCA with clusterings for process monitoring. The significant features are extracted by singular vector decomposition (SVD) or kernel PCA, and k-means is subsequently utilized as a clustering algorithm. Then, parameters of local PCA models are determined under each clustering model. Compared with PCA clustering, SVD based clustering only utilizes the nature basis for the components of the data instead of principal components of the data. Three clustering approaches are adopted and the effectiveness of the proposed approach is demonstrated by a practical coal pulverizing system.

1. Introduction

Recently, industrial operating safety and reliability have attracted increasing attention [1, 2, 3]. Thus, numerous researchers devote themselves to the study of process monitoring and remarkable achievements have been obtained [4, 5].

Since principal component analysis (PCA) is powerful to extract critical information within abundant process data, it is widely applied for data-driven process monitoring [6]. Variants of PCA have been proposed to tackle different limitations, including nonlinearity and dynamics. Kernel PCA (KPCA) is an effective technique for nonlinear applications [7, 8], where the data are projected to high-dimensional space. KPCA is not appropriate for large data owing to high computing complexity. For dynamics, recursive PCA, adaptive PCA and dynamic PCA have been proposed [9, 10]. However, aforementioned extensions of PCA are hard to combine. Then, probabilistic PCA (PPCA) was presented within the maximum likelihood framework, which measures the 'degree of novelty' of new data points by probability density function [11, 12]. This makes it convenient to establish mixture of PCA models. Thus, mixture of PPCA has been proposed to tackle the nonlinear constraint of PPCA [13, 14]. Generally, maximum likelihood is utilized to estimate the critical parameters, followed by expectation-maximum (EM) to obtain the optimal values [15]. Zhang *et al* presented an improved mixture of PPCA (IMPPCA) for nonlinear process monitoring with missing data, where the number of local models were determined automatically and a novel monitoring statistic was designed to improve performance [16].

Based on IMPPCA, data sets are effectively partitioned into local models [16], however this is via a joint probabilistic estimation process of computationally expensive EM estimation process. Intuitively it is desirable to consider local models via data clustering algorithms, enabling fast and flexible implementation. There are additional advantages associated with this general idea since there exist various clustering approaches focusing on multiple modality of data spaces. This work investigates three types of clustering combined with PCA for comparative studies, and introduces novel clustering approach based statistics for fault detection.

Traditional clustering methods utilize the raw data directly or remove abundant information beforehand [17, 18, 19]. For instance, the PCA projected data are separated into several clusters via on-line fuzzy clustering, where PCA can handle ill-conditioned issue of covariance matrix [20]. This is not effective for large and less informative data. Besides, important feature PCA was proposed for high dimensional clustering, where a small fraction of features are extracted by Kolmogorov-Smirnov scores [21]. In our proposed clustering approaches, important features are extracted by singular vector decomposition (SVD) or KPCA, and then sparse score vectors are adopted for clustering. Besides, three clustering approaches are compared to gain more insights.

The rest of this paper is organized below. Section 2 reviews concepts and mathematical formulations of the probabilistic PCA and lays the foundation of our proposed approach. Mixture of PPCA with clusterings is proposed in Section 3, in which k-means, clustering based on SVD and KPCA clustering are described in detail, respectively. Section IV details the procedure of nonlinear process monitoring using the proposed approach and analyzes the algorithm complexity. The effectiveness of the proposed approaches is illustrated by a practical coal pulverizing system in Section 5. Concluding remarks are presented in Section 6.

2. Preliminaries

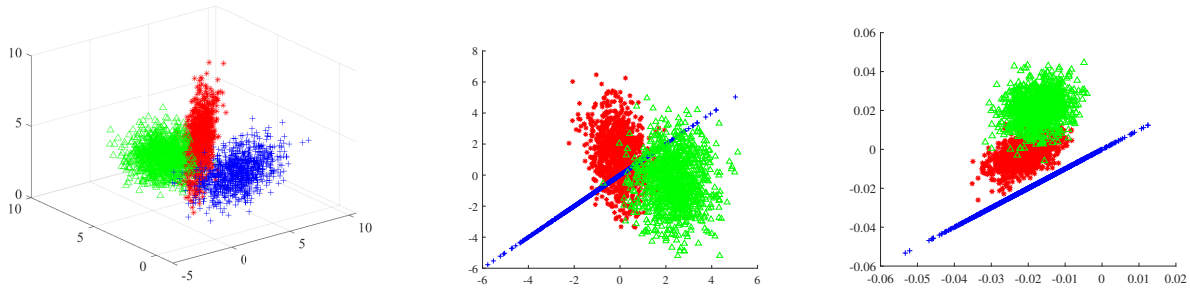
2.1. Latent variable models and PCA

Consider the following model

$$\boldsymbol{t} = f(\boldsymbol{x}; \boldsymbol{w}) + \boldsymbol{\xi} \tag{1}$$

where $\boldsymbol{t} \in R^{\mathrm{d}}$ is the observational data, $\boldsymbol{x} \in R^{\mathrm{q}}$ is the latent variable, $\boldsymbol{w}$ is the corresponding model parameter and $\boldsymbol{\xi}$ is

*Corresponding author
zjx18@mails.tsinghua.edu.cn (J. Zhang); x.hong@reading.ac.uk (X. Hong)
ORCID(s):



(a) Original data (b) PCA (c) SVD

Figure 1: PCA and SVD dimensionality reduction

the independent noise. $f(\boldsymbol{x};\boldsymbol{w})$ is the unknown function of the system. For instance, it can be represented by a linear model as follows:

$$\boldsymbol{t} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{\mu} + \boldsymbol{\xi} \tag{2}$$

Assume that process variables follow Gaussian distribution, namely, $\boldsymbol{x} \sim N(\boldsymbol{0}, \boldsymbol{I})$, $\boldsymbol{\xi} \sim N(\boldsymbol{0}, \boldsymbol{\Psi})$. $\boldsymbol{0}$ and $\boldsymbol{I}$ denote the vector of all zeros and identity matrix with appropriate dimensions respectively. $\boldsymbol{\Psi} \in R^{d\times d}$ is a diagonal matrix, $\boldsymbol{\mu} \in R^{d}$ is the mean vector, $\boldsymbol{W} \in R^{d\times q}$ is the loading matrix. Thus, we can obtain $\boldsymbol{t} \sim N(\boldsymbol{\mu}, \boldsymbol{C})$ with $\boldsymbol{C} = \boldsymbol{\Psi} + \boldsymbol{W}\boldsymbol{W}^{\mathrm{T}} \in R^{d\times d}$ based on (2).

As the detailed information of PPCA and IMPPCA has already been introduced in [15, 16], we just review the basic theory briefly in Appendix A and Section 2.2, respectively.

2.2. The mixture of probabilistic PCA

For the sake of modeling more complex data, mixture of PPCA (MPPCA) has been introduced [15] and adopted for nonlinear process monitoring [16], which utilizes an ensemble of local PCA models via defining a mixture of probabilistic densities [15]. According to probability rules, a mixture of K local PCA models is employed to describe the system (1) as below, rather than a single model of (2)

$$\begin{aligned} p(\boldsymbol{t}) &= \sum_{i=1}^{K} p(i)p(\boldsymbol{t}|i) \\ &= \sum_{i=1}^{K} \pi_i p(\boldsymbol{t}|i)) \end{aligned} \tag{3}$$

where the mixing coefficients $\pi_i \geq 0$ and $\sum_{i=1}^{K} \pi_i = 1$. $p(i)$ represents the probability of selecting the ith local model. Each of $p(\boldsymbol{t}|i)$ is the local PCA model given by

$$\boldsymbol{t} = \boldsymbol{W}_i\boldsymbol{x} + \boldsymbol{\mu}_i + \boldsymbol{\xi}_i, \quad i = 1, ..., K \tag{4}$$

which is similar to (2), and has individual projection matrix $\boldsymbol{W}_i$, mean vector $\boldsymbol{\mu}_i$, as well as $\boldsymbol{\xi}_i \sim N(\boldsymbol{0}, \sigma_i^2\boldsymbol{I})$.

For a mixture model, the log-likelihood of observation data based on (3) is described as:

$$\begin{aligned} L &= \sum_{n=1}^{N} \ln\left\{p\left(\boldsymbol{t}_n\right)\right\} \\ &= \sum_{n=1}^{N} \ln\left\{\sum_{i=1}^{K} \pi_i p\left(\boldsymbol{t}_n|i\right)\right\} \end{aligned}$$

The formulation of minimizing L has been summarized in Appendix B. Traditionally, an iterative EM algorithm was employed to jointly optimize the model parameters π_i, $\boldsymbol{\mu}_i$, $\boldsymbol{W}_i$ and σ_i^2. Detailed information can be found in [16, 22].

2.3. PCA-based and SVD-based clusterings

Given the measured data $\boldsymbol{T} = \left[\boldsymbol{t}_1, \ldots, \boldsymbol{t}_N\right]^T \in R^{N\times d}$, PCA reduces the dimensionality by SVD as follows:

$$\boldsymbol{T} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$$

where $\boldsymbol{U} \in R^{N\times d}$, $\boldsymbol{V} \in R^{d\times d}$ and $\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}_N$, $\boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{I}_d$. $\boldsymbol{\Sigma}$ is the diagonal matrix and the elements are eigenvalues of covariance matrix $\boldsymbol{T}$ with descending order. The number of principal components is q via cross validation, then the extracted data are expressed by

$$\boldsymbol{Y}_{PCA} = \boldsymbol{U}\boldsymbol{\Sigma}_q$$

where $\boldsymbol{\Sigma}_q$ is the first q columns of $\boldsymbol{\Sigma}$. Then, clustering approaches are conducted to classify the compressed data $\boldsymbol{Y}$ into several sorts subsequently.

To our best knowledge, SVD is essentially a low rank approximation technique. Therefore, we can adopt SVD to extract effective information. The extracted features $\boldsymbol{Y}_{SVD} = \boldsymbol{U}_q$ are employed for clustering. When two clustering approaches share the same dimensionality of extracted data, the input data of PCA-based clustering are actually the weighted inputs of SVD-based clustering, where the weights are the first q largest singular values of $\boldsymbol{T}$. It indicates that the clustering results of SVD-based clustering are less affected by the singular values once important features are selected.

To understand the difference intuitively, we generate numerical data and use PCA and SVD for dimensionality reduction, as shown in Figure 1. The original data are Gaussian-distributed and belong to 3 sorts. It is obviously that data by SVD processing are more dense and different classes are

separated relatively farther than PCA. It indicates that SVD is more suitable for clustering than PCA in some cases.

3. Identification of the mixture of probabilistic PCA with clustering

IMPPCA is eventually transformed into the optimal solution of (22) in Appendix B. The traditional solution is EM technique. The EM algorithm is known for its slow convergence, hence in practice it is often unavoidable to speed up by using a few iterations which hinders the performance. In this work we propose a two stage procedure: (i) each local models' parameters $\boldsymbol{\mu}_i$, $\boldsymbol{W}_i$ and σ_i^2, are identified based on a number of data clustering schemes; (ii) we estimate mixing coefficients π_i using maximum likelihood based on the resultant local models. The basic idea is that assuming that local PCA models can be found beforehand, we may reduce the burden of EM algorithm for more accurate estimation of the mixing parameters.

3.1. Identification of local models using data partition

We propose to obtain local models based on a subset of data. To this end we start with clustering D_n as M disjoint data subset $D_n^{(i)}$, $i = 1, ..., K$, with each datum $\boldsymbol{t}_n$ belongs to only one data subset. For convenience, denote the number of data samples in $D_n^{(i)}$, as $N^{(i)}$, and we have $\sum_{i=1}^{K} N^{(i)} = N$. For each local model, the PCA as described in Section 2.1 is directly applicable. We propose three clustering algorithms for data set partitioning.

3.1.1. k-means clustering

Clustering algorithms can be used to find a set of centers, which accurately reflect the distribution of the data points. From N data points $\boldsymbol{t}_j$, $j = 1, \cdots, N$, the k-means algorithm [23] seeks to partition the data points in K disjoint subset $D_n^{(i)}$, each containing $N^{(i)}$ data points, to minimize the sum-of-squares clustering function given by

$$J = \sum_{i=1}^{K} \sum_{\boldsymbol{t}_j \in D_n^{(i)}} \|\boldsymbol{t}_j - \mathbf{c}_i\|^2 \tag{5}$$

where $\in$ denotes belongs to. J is minimized when

$$\mathbf{c}_i = \frac{1}{N^{(i)}} \sum_{\boldsymbol{t}_j \in D_n^{(i)}} \boldsymbol{t}_j \tag{6}$$

The k-means clustering algorithm is applied as a baseline approach, which also forms as parts of the other two new approaches presented below. Besides, the number of clustering centers K is determined by gap statistic [24].

3.1.2. SVD clustering

The above k-means clustering algorithm is operated on d-dimensional original data space, which could be high dimensional. If the data is sparse in comparison to high input dimension, then clustering results may be not good. It is possible to perform a dimension reduction stage to map the data into most significant eigenvectors, on whose space the k-means clustering is applied. First we consider a rank-2 approximation of $\boldsymbol{T} \approx \boldsymbol{T}_{\text{SVD}} \boldsymbol{D}^{SVD} \boldsymbol{V}_{\text{SVD}}^{\text{T}}$, where $\boldsymbol{D}^{SVD} = \text{diag}\{d_1^{SVD}, d_2^{SVD}\}$. $d_1^{SVD} > d_2^{SVD}$ are two largest singular values of $\boldsymbol{T}$. $\boldsymbol{V}_{\text{SVD}}^{\text{T}}$ comprises first two right singular vectors. $\boldsymbol{T}_{\text{SVD}} = [\boldsymbol{t}_1^{\text{SVD}}, ..., \boldsymbol{t}_N^{\text{SVD}}]^{\text{T}} \in R^{N\times 2}$ comprises the first two left singular vectors of $\boldsymbol{T}$, where each row $\boldsymbol{t}_j^{\text{SVD}}$, $j = 1, ...N$, can be regarded as a result of mapping original data $\boldsymbol{t}_j$ into eigenvector space induced by SVD on data matrix $\boldsymbol{T}$. The proposed SVD clustering is simply by applying k-means clustering algorithm to $\boldsymbol{t}_j^{\text{SVD}}$, rather than the original data $\boldsymbol{t}_j$, followed by identifying the corresponding M disjoint data subset D_n^i, $i = 1, ..., K$ in the original data space.

3.1.3. Kernel PCA Clustering

KPCA [25] is one of the kernel methods that is based on the so-called "kernel trick". Briefly speaking, it is initially assumed that there exists a unknown nonlinear feature mapping $\boldsymbol{\phi}(\boldsymbol{t})$, whose dimension is generally unknown and could even be indefinite. In KPCA, only the inner product is specified, so that

$$k(\boldsymbol{t}_i, \boldsymbol{t}_j) = \boldsymbol{\phi}(\boldsymbol{t}_i)^{\text{T}} \boldsymbol{\phi}(\boldsymbol{t}_j) \tag{7}$$

KPCA implicitly finds the leading eigenvectors and eigenvalues of the covariance of the data in feature space $\boldsymbol{\phi}(\boldsymbol{t}_n)$. Given D_n and a kernel function $k(\boldsymbol{t}_i, \boldsymbol{t}_j)$, e.g. $k(\boldsymbol{t}_i, \boldsymbol{t}_j) = \exp(-\frac{(\boldsymbol{t}_i - \boldsymbol{t}_j)^2}{2\tau^2})$, τ is the bandwidth. The centered kernel matrix is given by

$$\tilde{\boldsymbol{K}} = \boldsymbol{K} - \frac{1}{N}\mathbf{1}_{N\times N}\boldsymbol{K} - \frac{1}{N}\boldsymbol{K}\mathbf{1}_{N\times N} + \frac{1}{N^2}\mathbf{1}_{N\times N}\boldsymbol{K}\mathbf{1}_{N\times N} \tag{8}$$

where $\boldsymbol{K} = \{k_{i,j}\} = \{k(\boldsymbol{t}_i, \boldsymbol{t}_j)\}$ is the uncentered kernel matrix. $\mathbf{1}_{N\times N} \in R^{N\times N}$ is the matrix of all ones.

We propose a new KPCA clustering scheme based on rank-2 approximation of $\tilde{\boldsymbol{K}} \approx \boldsymbol{T}_{\text{KPCA}} \boldsymbol{D}_{\text{KPCA}} \boldsymbol{T}_{\text{KPCA}}^{\text{T}}$, where $\boldsymbol{D}_{\text{KPCA}} = \text{diag}\{d_1^{\text{KPCA}}, d_2^{\text{KPCA}}\}$. $d_1^{\text{KPCA}} > d_2^{\text{KPCA}}$ are two largest eigenvalues $\tilde{\boldsymbol{K}}$. $\boldsymbol{T}_{\text{KPCA}} = [\boldsymbol{t}_1^{\text{KPCA}}, ..., \boldsymbol{t}_N^{\text{KPCA}}]^{\text{T}} \in R^{N\times 2}$ comprises the two largest eigenvectors of $\tilde{\boldsymbol{K}}$, where each row $\boldsymbol{t}_j^{\text{KPCA}}$, $j = 1, ...N$, can be regarded as a result of mapping original data $\boldsymbol{t}_j$ into a nonlinear feature's eigenvector space. The proposed KPCA clustering is simply by applying k-means algorithm to $\boldsymbol{t}_j^{\text{KPCA}}$, rather than the original data $\boldsymbol{t}_j$, followed by identifying the corresponding M disjoint data subset D_n^i, $i = 1, ..., K$ in the original data space.

3.2. Specific parameter estimation of local models

With any of the clustering algorithms, PCA of Section 2.1 can be applied to obtain $\boldsymbol{\mu}_i$, $\boldsymbol{W}_i$ and σ_i^2 of each local model as summarized below.

Clearly the mean vector $\boldsymbol{\mu}_i = \frac{1}{N^{(i)}} \sum_{n=1}^{N^{(i)}} \boldsymbol{t}_n^{(i)}$. Let the sample covariance matrices of each local models be denoted by $\boldsymbol{S}^{(i)} = \frac{1}{N^{(i)}} \left(\boldsymbol{T}^{(i)} - \boldsymbol{u}_i \mathbf{1}^{\text{T}}\right) \left(\boldsymbol{T}^{(i)} - \boldsymbol{u}_i \mathbf{1}^{\text{T}}\right)^{\text{T}}$. We obtain the eigenvalue decomposition of $\boldsymbol{S}^{(i)} = \tilde{\boldsymbol{W}}^{(i)} \boldsymbol{\Lambda}^{(i)} \tilde{\boldsymbol{W}}^{(i)^{\text{T}}}$, where $\boldsymbol{\Lambda}^{(i)} =$

$\text{diag}\{\lambda_1^{(i)}, ..., \lambda_q^{(i)}, \lambda_{q+1}^{(i)}, ..., \lambda_d^{(i)}\}$, with $\boldsymbol{W}^{(i)}$ being the first q columns of $\tilde{\boldsymbol{W}}^{(i)}$. We can also obtain σ_i^2 as

$$\sigma_i^2 = \frac{1}{d-q} \sum_{j=q+1}^{d} \lambda_j^{(i)} \tag{9}$$

Then, we need to estimate the mixing coefficients π_i and posterior probability R_{ni}. Suppose that $R_{ni} = p\left(i|\boldsymbol{t}_n\right)$ is the posterior probability of the ith local model for generating data $\boldsymbol{t}_n$, it can be estimated by the following formula:

$$R_{ni} = \frac{p\left(\boldsymbol{t}_n|i\right)\pi_i}{p\left(\boldsymbol{t}_n\right)} \tag{10}$$

For given $\boldsymbol{\mu}_i$, $\boldsymbol{W}_i$ and σ_i^2, the solution of mixing coefficients π_i can be significantly simplified as the optimization problem

$$\begin{cases} \max \quad \langle L_c \rangle + \lambda \left(\sum_{i=1}^{K} \pi_i - 1 \right) \\ s.t. \quad \sum \pi_i = 1 \end{cases} \tag{11}$$

so we have

$$\sum_{n=1}^{N} \frac{R_{ni}}{\pi_i} + \lambda = 0 \tag{12}$$

Since $\sum_{i=1}^{K} \frac{R_{ni}}{\pi_i} = 1$, we have $\lambda = -N$, and

$$\pi_i = \frac{1}{N} \sum_{n=1}^{N} R_{ni} \tag{13}$$

Initialize $\pi_i = 1/K, \forall i$, R_{ni} can simply be iteratively calculated by (10), in which $p\left(\boldsymbol{t}_n|i\right)$ is evaluated only once without iteration via

$$p(\boldsymbol{t}_n|i) = (2\pi)^{-d/2} \left|\boldsymbol{C}_i\right|^{-1/2} \exp\left\{-\frac{1}{2}(\boldsymbol{t}_n - \boldsymbol{\mu}_i)^{\mathrm{T}} \boldsymbol{C}_i^{-1} (\boldsymbol{t}_n - \boldsymbol{\mu}_i)\right\} \tag{14}$$

where $\boldsymbol{C}_i = \sigma_i^2 \boldsymbol{I} + \boldsymbol{W}_i \boldsymbol{W}_i^{\mathrm{T}}$.

4. Summary and discussion

The procedure of a mixture of PPCA with clusterings is summarized as follows. First, the model framework and corresponding parameters can be acquired by three steps: 1) identification of local models is obtained using data partition, where k-means, SVD clustering, KPCA clustering are adopted, respectively; 2) critical parameters of each local model can be computed just by PCA, which is fast and simple; 3) the weights of samples belong to each local model can be calculated by maximum likelihood, where the convergence of iterative process is acceptable and fast. Then, with regard to each local model, two monitoring statistics, namely, T-squared (T^2) and squared prediction error (SPE), are established. Next, global monitoring statistics are constructed based on the statistics of local models and weights. To this end, the procedure of on-line process monitoring is developed for nonlinear processes.

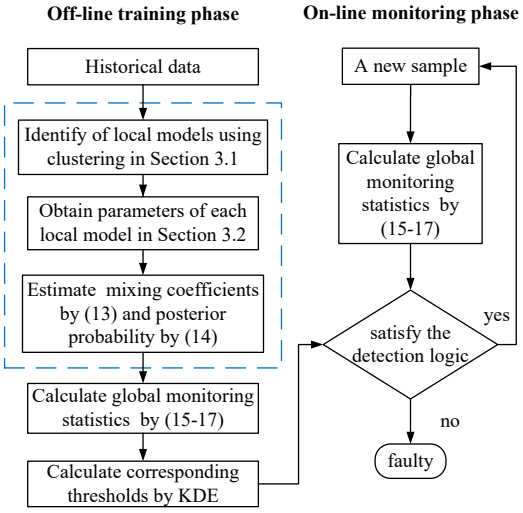


Figure 2: Flowchart of a mixture of PPCA with clusterings for nonlinear process monitoring

4.1. Establishing monitoring statistics

The number of local models is determined by the clustering approaches mentioned in Section 3.1. With respect to ith local model, let T_i^2 and SPE_i be the monitoring statistics for principal component subspace and residual component subspace, respectively. The monitoring statistics of the nth sample are computed below

$$T_i^2 = \left\| \boldsymbol{M}_i \boldsymbol{W}_i^{\mathrm{T}} \boldsymbol{t}_n \right\|^2 \tag{15}$$

$$\text{SPE}_i = \left\| \sigma_i^{-1} (\boldsymbol{I} - \boldsymbol{W}_i \boldsymbol{M}_i \boldsymbol{W}_i^{\mathrm{T}}) \boldsymbol{t}_n \right\|^2 \tag{16}$$

As for the proposed approach, if the posterior probability of the sample belonging to kth local model is the largest among these local models, then the global outcome is the statistic indice of the kth model. Briefly speaking, the global monitoring statistics can be computed by:

$$\begin{cases} T^2 = T_i^2 \\ SPE = SPE_i \end{cases}, \quad i = \arg\max_{j=1,...,K} R_{nj} \tag{17}$$

For process monitoring task, the relationship between monitoring statistics and the corresponding thresholds is the reference standard of operating condition, faulty or normal. The thresholds are calculated by kernel density estimation (KDE), labeled as J_{th,T^2} and $J_{th,\text{SPE}}$, respectively. Thus, the corresponding detection logic satisfies

$T^2 \le J_{th,T^2}$ and $SPE \le J_{th,\text{SPE}} \Rightarrow$ fault free, otherwise faulty.

The procedure of a mixture of PPCA with clusterings for nonlinear process monitoring is summarized in Figure 2.

4.2. Algorithm complexity analysis

In this paper, time complexity is discussed specifically and space complexity is briefly compared. We use the term

Table 1
Computational complexity of three clustering approaches

Algorithm	kernel matrix	SVD	k-means	PCA
PCA based on k-means	-	-	$O(NKdt)$	$\frac{1}{2}Nd^2+\frac{9}{2}Kd^3$
PCA based on SVD clustering	-	$O(N^2d+Nd^2+d^3)$	$O(NKqt)$	$\frac{1}{2}Nd^2+\frac{9}{2}Kd^3$
PCA based on KPCA clustering	$O(N^2d)$	$O(N^3)$	$O(NKqt)$	$\frac{1}{2}Nd^2+\frac{9}{2}Kd^3$

flam to count, a compound operation consisting of one addition and one multiplication [26].

For k-means, the computational complexity mainly focuses on the iteration of clustering centers. The time computational complexity is $O(NKdt)$, where t is the number of iterations. For SVD, assume that $\boldsymbol{X}=\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$. Specially, calculating the $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ needs $\frac{1}{2}Nd^2$ *flam*. Then, calculating the eigenvectors of $\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}$ requires $\frac{9}{2}d^3$ *flam*. Thus, recovering $\boldsymbol{U}$ from $\boldsymbol{V}$ needs N^2d *flam* under the assumption that $\boldsymbol{X}$ is of full rank. As calculating the eigenvectors of $\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}}$ directly requires $\frac{9}{2}N^3$ *flam*, this indirect mode of calculation can achieve a significant saving especially for large data. For SVD clustering, only the first two largest singular values and corresponding singular vectors are required, thus the computational complexity is $O(N^2d+Nd^2+d^3)$.

For KPCA, computing kernel matrix requires $O(N^2d)$. Eig-decomposition of kernel matrix needs $\frac{9}{2}N^3$ *flam*. Computational complexity of three clustering approaches is concluded in Table 1. For KPCA-based fault detection method, all singular values are required and utilized to calculate q. Then, the corresponding q eigenvectors are calculated thereafter. For the proposed KPCA-based clustering, only the first two largest singular values and the corresponding left and right singular vectors are required. For large-scale dataset, $q \gg 2$. Thus, KPCA-based fault detection method is more computational complicated than KPCA-based clustering. The computational cost of IMPPCA [16] mainly focuses on the solution of model parameters iteratively, and the time complexity is at least $O(KNtd^3)$. IMPPCA is a joint optimization issue and may converge after considerable iterations. Besides, matrix inversion should be calculated many times and may be ill-conditioned.

As for space complexity, among the approaches aforementioned, kernel methods cost the most memory due to the construction of kernel matrix. Besides, KPCA clustering algorithm needs less storage space than KPCA as only two largest singular values and corresponding singular vectors are preserved. IMPPCA occupies the most storage memory because the parameters during iteration process need to be stored. Besides, PCA based on k-means may cost the least memory among these methods.

5. Case study

The 1000-MW ultra-supercritical thermal power plant is increasingly popular and highly complex. In this paper, one important unit of boiler is investigated, namely, the coal pulverizing system. It is expected to provide the proper pulverized coal, where the coal fineness and the temperature should range within the prescribed limits. The coal pulverizing system includes coal feeder, coal mill, rotary separator, raw coal hopper and stone coal scuttle, as shown in Figure 3. Raw coal is ground by a coal mill and fed into a rotary separator. Then, the pulverized coal with desired fineness is sent to the boiler for combustion through the rotary separator.

We select two typical cases to demonstrate the effectiveness of the proposed approach, namely abnormality from outlet temperature and rotary separator. In this case study, the sampling interval is 1 minute. Data information is summarized in Table 2. The numbers of training and testing samples are denoted as NoTrS and NoTeS, respectively. To reduce false alarms, the critical variables are different for various types of faults. In this paper, the variables are selected based on prior knowledge from experts and basic theory. For the coal pulverizing system, the instantaneous coal supply varies with plant's load and is not invariably stationary. Besides, the other parameters of the coal pulverizing system changes with the instantaneous coal supply and the type of coal. Thus, the most continuous variables are always weakly non-stationary, as illustrated in Figure 4.

The monitoring results of 6 faults are summarized in Table 3, including FDRs (%) and FARs (%). PCA fails to detect the faults accurately and timely. Although the FAR of Fault 1 is acceptable, the FDR is lower than other clustering based algorithms. The FDR of Fault 2 is 100%, but the FAR is 10%. PCA can detect Faults 1 and 3, but the FDRs of other faults are lower than 30%. KPCA fails to distinguish the nor-

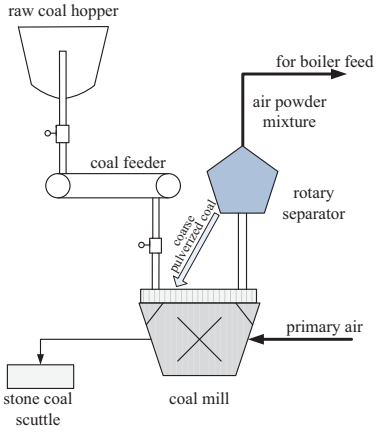


Figure 3: Schematic of the coal pulverizing system

Table 2
Data of the practical coal pulverizing system

Fault type	Key variables	Fault number	NoTrS/NoTeS	Fault location	Fault cause
Outlet temperature abnormality	21 variables: outlet temperature, pressure of air powder mixture, primary air temperature, hot/cold primary air main pressure, etc.	Fault 1	2160/2880	909	Internal deflagration owing to high outlet temperature
		Fault 2	1080/1080	533	Hot primary air electric damper failure
		Fault 3	1440/1440	626	Air leakage at cold and hot primary air interface
Rotary separator	12 variables: rotary separator speed and current, bearing temperature, instantaneous coal feeding capacity of coal feeder, etc.	Fault 4	2880/1080	806	Frequency conversion cabinet output short circuit alarm
		Fault 5	720/720	352	High temperature of rotary separator bearing
		Fault 6	2880/2160	134	Large vibration

Table 3
Evaluation indexes of the case study

Fault type	Indexes	PCA		KPCA		PPCA		k-means		SVD clustering		KSVD clustering	
		T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE	T^2	SPE
Fault 1	FDRs	0	89.15	99.65	83.27	99.80	99.95	99.75	99.75	99.75	99.75	99.75	99.75
	FARs	0	8.81	23.35	0.44	0	13.88	0	0	0	0	0	0
Fault 2	FDRs	0	21.17	99.45	93.43	99.09	100	98.36	98.36	97.81	97.99	98.36	98.36
	FARs	0	14.66	41.92	0.19	0	5.83	0	0	0	0	0	0
Fault 3	FDRs	1.23	100	99.75	99.75	100	100	100	100	100	100	100	100
	FARs	0.16	10	9.92	0	28.00	100	1.60	0	0.16	1.44	4.80	5.76
Fault 4	FDRs	0	0	100	95.27	100	100	100	100	100	100	100	100
	FARs	0	0	4.6	0	0	0	0	0	0.12	0.12	0.12	0.12
Fault 5	FDRs	0	0	100	100	2.44	100	99.73	99.73	99.73	99.73	99.73	99.73
	FARs	0	0	7.41	3.70	0	4.27	3.42	3.42	3.42	3.13	3.42	3.13
Fault 6	FDRs	0	27.04	85.30	85.99	29.60	92.60	96.50	96.50	95.56	96.30	95.95	96.69
	FARs	0	0	0	1.50	0	0	1.50	1.50	1.50	1.50	1.50	1.50

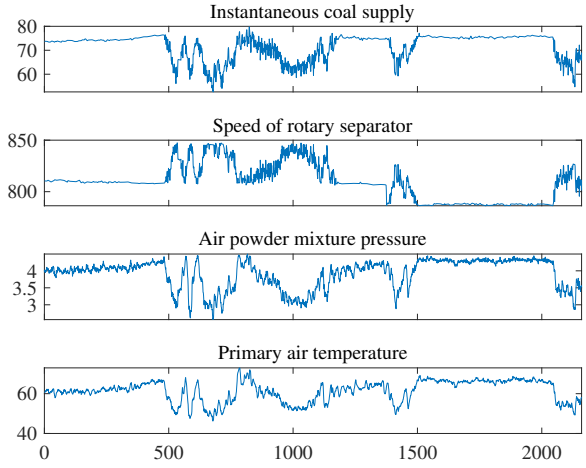


Figure 4: Partial data of the coal pulverizing system

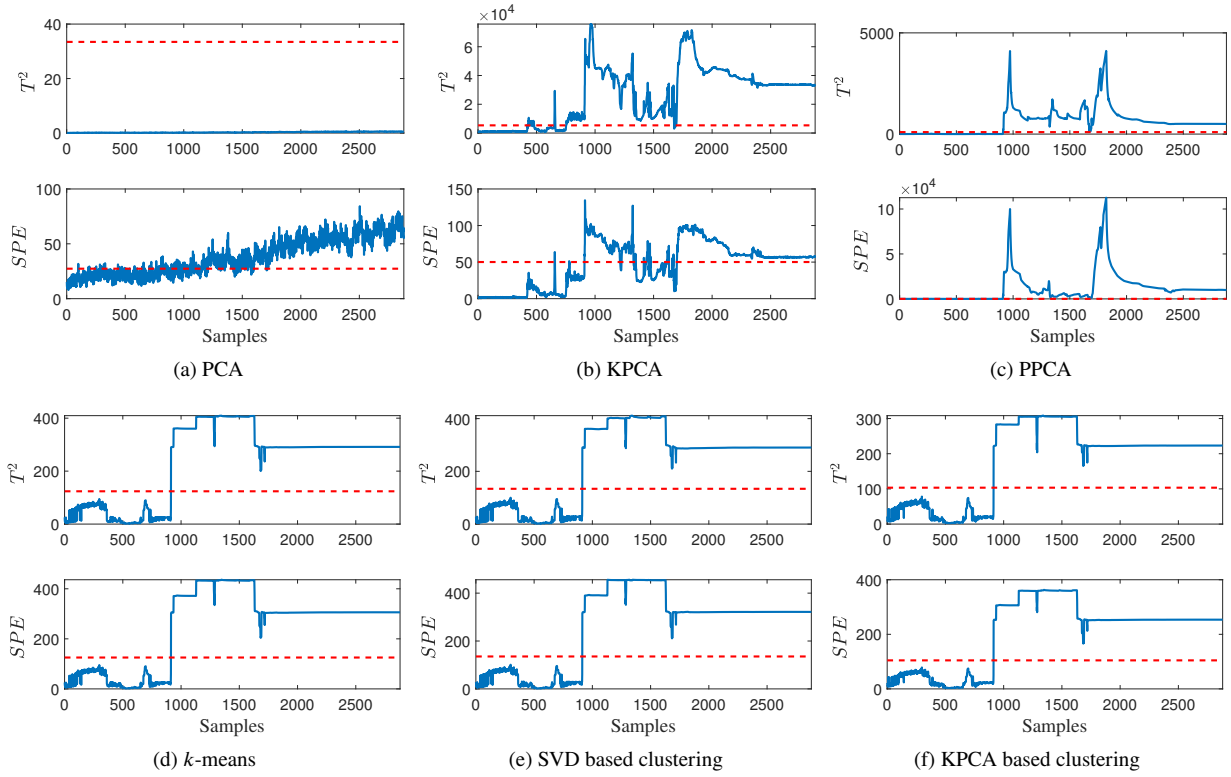
mal variations from real faults for Faults 1 and 2, because the FARs are more than 20%. For other faults, KPCA provides the similar FDRs but the FARs are a little higher than clustering based methods. For PPCA, the simulation results of Faults 2, 4, 5 and 6 are pretty excellent and the FARs are acceptable. The FARs of Faults 1 and 3 are relatively high, especially the Fault 3. In this case study, three clustering based algorithms provide similar monitoring performance, where k-means, SVD based clustering and KPCA based clustering algorithms are adopted and probabilistic PCA parameters are estimated thereafter. The FDRs approach to 100% and the FARs are lower than 6%. Moreover, only the monitoring charts of Fault 1 are listed in Figure 5 owing to space limitations.

Table 4 lists the practical training and testing time. It is evident that KPCA costs the most expensive computational resources and PCA needs the lowest resource. Similarly, PPCA also has lower computational complexity. For three clustering algorithms, the computational complexity is similar and the KSVD clustering needs considerable more resources than the others, which is consistent with Table 1.

In conclusion, the proposed probabilistic PCA with clusterings provides excellent performance for nonlinear processes. Besides, it is less computational complicated than KPCA. Although the computational complexity is a little higher than that of PPCA, the accuracy is more satisfactory. Here, we

Table 4
Simulation time (s) of the case study

Fault type	PCA		KPCA		PPCA		k-means		SVD clustering		KSVD clustering	
	training	testing	training	testing	training	testing	training	testing	training	testing	training	testing
Fault 1	0.1493	0.0563	319.9	421.07	0.3588	0.0201	1.4062	1.0645	1.5268	1.0314	4.9589	1.0715
Fault 2	0.0477	0.0211	32.82	34.69	0.2176	0.0062	0.7370	0.4074	0.7486	0.4219	1.1281	0.4323
Fault 3	0.0732	0.0291	130.85	121.667	0.2535	0.0081	1.0416	0.5434	1.0386	0.5750	1.7466	0.6070
Fault 4	0.1769	0.0184	1194.15	452.25	0.3378	0.0061	1.7183	0.2511	2.0596	0.2613	13.8107	0.2749
Fault 5	0.0290	0.0116	6.731	6.916	0.1377	0.0046	0.3678	0.1738	0.3584	0.177	0.4634	0.17
Fault 6	0.1785	0.0359	506.66	357.8033	0.4766	0.0138	0.9096	0.4124	0.9833	0.4114	11.3460	0.4103



(a) PCA (b) KPCA (c) PPCA

(d) k-means (e) SVD based clustering (f) KPCA based clustering

Figure 5: Monitoring charts of Fault 1

explain the reason why IMPPCA in [16] is not utilized as a comparative approach. For IMPPCA, the local model parameters are estimated by jointly optimizing the likelihood and obtained when the objective converges. In the iteration process, as the variables may be nonstationary in Figure 4, the posterior probability is ill-conditioned and the algorithm is impossible to acquire the parameters. That is to say, IMPPCA in [16] has more application limitations and is not suitable for this nonlinear process.

6. Conclusion

This paper presents a nonlinear process monitoring algorithm based on probabilistic PCA with clusterings. The innovation lies in the SVD instead of traditional PCA dimensionality reduction. In some cases, data after SVD processing would be more dense and different classes are separated more farther than PCA. Based on this property, to tackle the algorithm complexity and potential ill-condition of the IMPPCA, we determine local PCA model beforehand by three clustering approaches, and then parameters of local models are calculated simply by once iteration. Furthermore, we analyze the algorithm complexity and the proposed algorithm is less computationally complicated compared with KPCA and IMPPCA. Eventually, compared with PCA, KPCA and PPCA, the effectiveness and superiorities of the proposed approach are demonstrated by a practical coal pulverizing system.

In future, spectral clustering would be investigated further and more extensions may be explored for nonstationary process monitoring.

A. The probabilistic PCA

Providing that noise $\boldsymbol{\xi} \sim N(\mathbf{0}, \sigma^2 \boldsymbol{I})$, for a given $\boldsymbol{x}$, the conditional probability over $\boldsymbol{t}$ is defined by

$$p(\boldsymbol{t}|\boldsymbol{x}) = (2\pi\sigma^2)^{-d/2} \exp\left\{-\frac{1}{2\sigma^2}\|\boldsymbol{t} - \boldsymbol{W}\boldsymbol{x} - \boldsymbol{\mu}\|^2\right\} \quad (18)$$

The prior probability over $\boldsymbol{x}$ can be calculated as

$$p(\boldsymbol{x}) = (2\pi)^{-q/2} \exp\left\{-\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}\right\} \quad (19)$$

Then, the marginal distribution of $\boldsymbol{t}$ is obtained by

$$\begin{aligned} p(\boldsymbol{t}) =& \int p(\boldsymbol{t}|\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x} \\ =&(2\pi)^{-d/2}|\boldsymbol{C}|^{-1/2} \exp\left\{-\frac{1}{2}(\boldsymbol{t}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{C}^{-1}(\boldsymbol{t}-\boldsymbol{\mu})\right\} \end{aligned} \quad (20)$$

where $|\cdot|$ represents matrix determinant. The model covariance is computed by $\boldsymbol{C} = \sigma^2\boldsymbol{I} + \boldsymbol{W}\boldsymbol{W}^{\mathrm{T}}$.

Based on Bayesian theory, the associated posterior distribution of $\boldsymbol{x}$ for a specific $\boldsymbol{t}$ is acquired by:

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{t}) =& \exp\left\{-\frac{1}{2}\left\{\boldsymbol{x} - \boldsymbol{M}^{-1}\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{t}-\boldsymbol{\mu})\right\}^{\mathrm{T}}(\sigma^{-2}\boldsymbol{M})\right. \\ & \left.\left\{\boldsymbol{x} - \boldsymbol{M}^{-1}\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{t}-\boldsymbol{\mu})\right\}\right\} \times (2\pi)^{-q/2}\left|\sigma^{-2}\boldsymbol{M}\right|^{1/2} \end{aligned}$$

where $\boldsymbol{M} = \boldsymbol{I} + \boldsymbol{W}^{\mathrm{T}}\boldsymbol{W} \in R^{q\times q}$ is the posterior covariance.

The log-likelihood of the observation data is

$$\begin{aligned} L =& \sum_{n=1}^{N} \ln\left\{p\left(\boldsymbol{t}_n\right)\right\} \\ =& -\frac{N}{2}\left\{d\ln(2\pi) + \ln|\boldsymbol{C}| + tr\left(\boldsymbol{C}^{-1}\boldsymbol{S}\right)\right\} \end{aligned} \quad (21)$$

in which $\boldsymbol{S}$ is the sample covariance.

B. Solution of mixing coefficients based on maximum likelihood

According to [16], the expectation of L_c for MPPCA is obtained by

$$\begin{aligned} \langle L_c\rangle = \sum_{n=1}^{N}\sum_{i=1}^{K} R_{ni}\left\{\ln\pi_i - \frac{d}{2}\ln\sigma_i^2 - \frac{1}{2}tr\left\{\left\langle \boldsymbol{x}_n^{(i)}\boldsymbol{x}_n^{(i)\mathrm{T}}\right\rangle\right\}\right. \\ -\frac{1}{2\sigma_i^2}\left\|\boldsymbol{t}_n - \boldsymbol{\mu}_i\right\|^2 + \frac{1}{\sigma_i^2}\left\langle \boldsymbol{x}_n^{(i)}\right\rangle^{\mathrm{T}}\boldsymbol{W}_i^{\mathrm{T}}(\boldsymbol{t}_n - \boldsymbol{\mu}_i) \\ \left.-\frac{1}{2\sigma_i^2}tr\left\{\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{W}_i\left\langle \boldsymbol{x}_n^{(i)}\boldsymbol{x}_n^{(i)\mathrm{T}}\right\rangle\right\}\right\} \end{aligned} \quad (22)$$

where $\langle\cdot\rangle$ denotes the expectation about the posterior distributions of $\boldsymbol{x}_n^{(i)}$. $\left\langle \boldsymbol{x}_n^{(i)}\right\rangle$ and $\left\langle \boldsymbol{x}_n^{(i)}\boldsymbol{x}_n^{(i)\mathrm{T}}\right\rangle$ are posterior mean and covariance matrices of ith local model, calculated as follows:

$$\left\langle \boldsymbol{x}_n^{(i)}\right\rangle = \boldsymbol{M}_i^{-1}\boldsymbol{W}_i^{\mathrm{T}}\left(\boldsymbol{t}_n - \boldsymbol{\mu}_i\right) \in R^q \quad (23)$$

$$\left\langle \boldsymbol{x}_n^{(i)}\boldsymbol{x}_n^{(i)\mathrm{T}}\right\rangle = \sigma_i^2\boldsymbol{M}_i^{-1} + \left\langle \boldsymbol{x}_n^{(i)}\right\rangle\left\langle \boldsymbol{x}_n^{(i)}\right\rangle^{\mathrm{T}} \in R^{q\times q} \quad (24)$$

Acknowledgements

This paper was supported by National Natural Science Foundation of China (Grant No.61873147). The authors are thankful to reviewers for their constructive comments that helped improve the quality of this paper.

References

[1] M. S. Afzal, W. Tan, T. Chen, Process monitoring for multimodal processes with mode-reachability constraints, IEEE Transactions on Industrial Electronics 64 (2017) 4325 – 4335.

[2] S. Zhang, C. Zhao, Concurrent analysis of variable correlation and data distribution for monitoring large-scale processes under varying operation conditions, Neurocomputing 349 (2019) 225–238.

[3] L. Ma, J. Dong, C. Hu, K. Peng, A novel decentralized detection framework for quality-related faults in manufacturing industrial processes, Neurocomputing 428 (2021) 30–41.

[4] K. Zhang, K. Peng, R. Chu, J. Dong, Implementing multivariate statistics-based process monitoring: A comparison of basic data modeling approaches, Neurocomputing 290 (2018) 172–184.

[5] J. Dong, C. Zhang, K. Peng, A novel industrial process monitoring method based on improved local tangent space alignment algorithm, Neurocomputing 405 (2020) 114–125.

[6] X. Shi, F. Nie, Z. Lai, Z. Guo, Robust principal component analysis via optimal mean by joint $l_{2,1}$ and schatten p-norms minimization, Neurocomputing 283 (2018) 205–213.

[7] L. Guo, P. Wu, S. Lou, J. Gao, Y. Liu, A multi-feature extraction technique based on principal component analysis for nonlinear dynamic process monitoring, Journal of Process Control 85 (2020) 159–172.

[8] B. Zhou, X. Gu, Multi-block statistics local kernel principal component analysis algorithm and its application in nonlinear process fault detection, Neurocomputing 376 (2020) 222–231.

[9] I. B. Khediri, M. Limam, C. Weihs, Variable window adaptive kernel principal component analysis for nonlinear nonstationary process monitoring, Computers & Industrial Engineering 61 (2011) 437–446.

[10] K. Wang, J. Chen, Z. Song, Performance analysis of dynamic PCA for closed-loop process monitoring and its improvement by output oversampling scheme, IEEE Transactions on Control Systems and Technology 27 (2019) 378–385.

[11] M. E. Tipping, C. M. Bishop, Probabilistic principal component analysis, Journal of the Royal Statistical Society: Series B 61 (1999) 611–622.

[12] D. Kim, I. B. Lee, Process monitoring based on probabilistic PCA, Chemometrics & Intelligent Laboratory Systems 67 (2003) 109–123.

[13] S. W. Choi, E. B. Martin, A. J. Morris, I.-B. Lee, Fault detection based on a maximum likelihood principal component analysis (PCA) mixture, Industrial & Engineering Chemistry Research 44 (2005) 2316–2327.

[14] K. Honda, H. Ichihashi, Regularized linear fuzzy clustering and probabilistic PCA mixture models, IEEE Transactions on Fuzzy System 13 (2005) 508–516.

[15] M. E. Tipping, C. M. Bishop, Mixtures of probabilistic principal component analyzers, Neural Computation 11 (1999) 443–482.

[16] J. Zhang, H. Chen, S. Chen, X. Hong, An improved mixture of probabilistic PCA for nonlinear data-driven process monitoring, IEEE Transactions on Cybernetics 49 (2019) 198–210.

[17] J. Katkar, T. Baraskar, V. R. Mankar, A novel approach for medical image segmentation using PCA and k-means clustering, in: International Conference on Applied & Theoretical Computing & Communication Technology, 2016.

[18] T. T. Cai, L. Zhang, A sparse PCA approach to clustering, arXiv: 1602.05236v1 (2016).

[19] J. Chen, Mixture principal component analysis models for process monitoring, Industrial & Engineering Chemistry Research 38 (1999) 1478–1488.

[20] H. K. Alaei, A new integrated on-line fuzzy clustering and segmentation methodology with adaptive pca approach for process monitoring

and fault detection and diagnosis, Soft Computing 17 (2013) 345–362.
[21] J. Jin, W. Wang, Important feature PCA for high dimensional clustering, Eprint Arxiv (2015).
[22] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society: Series B 39 (1977) 1–38.
[23] C. Boutsidis, M. Magdon-Ismail, Deterministic feature selection for k-means clustering, IEEE Transactions on Information Theory 59 (2013) 6099–6110.
[24] R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a data set via the gap statistic, Journal of The Royal Statistical Society Series B-statistical Methodology 63 (2001) 411–423.
[25] K. E. Pilario, M. Shafiee, Y. Cao, L. Lao, S.-H. Yang, A review of kernel methods for feature extraction in nonlinear process monitoring, Processes 8 (2019) 24.
[26] D. Cai, Spectral Regression: A Regression Framework for Efficient Regularized Subspace Learning, 2009.