Journal of Process Control 67 (2018) 112–128

Contents lists available at ScienceDirect

Journal of Process Control

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

Real-time fault detection and diagnosis using sparse principal component analysis☆

Shriram Gajjar[a], Murat Kulahci[b,c], Ahmet Palazoglu[a,*]

[a] Department of Chemical Engineering, University of California, Davis, CA 95616, USA
[b] Department of Informatics and Mathematical Modeling, Technical University of Denmark, Lyngby, Denmark
[c] Luleå University of Technology, Luleå, Sweden

ARTICLE INFO

Article history:
Received 15 June 2016
Received in revised form 1 January 2017
Accepted 10 March 2017
Available online 29 April 2017

Keywords:
Process surveillance
Latent structures
Multivariate statistical process monitoring
Tennessee Eastman process

ABSTRACT

With the emergence of smart factories, large volumes of process data are collected and stored at high sampling rates for improved energy efficiency, process monitoring and sustainability. The data collected in the course of enterprise-wide operations consists of information from broadly deployed sensors and other control equipment. Interpreting such large volumes of data with limited workforce is becoming an increasingly common challenge. Principal component analysis (PCA) is a widely accepted procedure for summarizing data while minimizing information loss. It does so by finding new variables, the principal components (PCs) that are linear combinations of the original variables in the dataset. However, interpreting PCs obtained from many variables from a large dataset is often challenging, especially in the context of fault detection and diagnosis studies. Sparse principal component analysis (SPCA) is a relatively recent technique proposed for producing PCs with sparse loadings via variance-sparsity trade-off. Using SPCA, some of the loadings on PCs can be restricted to zero. In this paper, we introduce a method to select the number of non-zero loadings in each PC while using SPCA. The proposed approach considerably improves the interpretability of PCs while minimizing the loss of total variance explained. Furthermore, we compare the performance of PCA- and SPCA-based techniques for fault detection and fault diagnosis. The key features of the methodology are assessed through a synthetic example and a comparative study of the benchmark Tennessee Eastman process.

1. Introduction

Smart production technologies that are implemented today have dramatically intensified data generation and collection through networked information-based technologies throughout the chemical industry and other manufacturing enterprises. The data generation and collection are so fast-paced that humans have to rely on computers for consuming as well as processing the data. It is thus imperative to develop dedicated algorithms and methods to improve process performance and facilitate process surveillance. These algorithms and methods should, at first, be able to unlock significant information from large datasets and, second, provide accurate means to reduce process variability and boost performance. Third, they should allow discovery of the underlying process dynamics that can substantially improve decision-making. Finally, steps can then be taken to move toward recommending preemptive actions (preventive decisions that are made before a failure occurs or is even observed).

Historically, multivariate statistical analysis and statistical process monitoring (SPM) techniques have been applied in a wide range of fields including genomics, signal processing, and various industrial processes [1–5]. Principal component analysis (PCA) is one of the most commonly used multivariate techniques with various applications ranging from image recognition to gene engineering to financial or climate data.

PCA preserves as much variability as possible of the dataset by finding a new set of variables or principal components (PCs) that are linear combinations of those in the original dataset that successively maximize variance and are uncorrelated with each other. These new sets of PCs are obtained by solving an eigenvalue problem. PCA captures the variance in m variables of the original dataset in a reduced dimension by l retained PCs. In most conventional settings, data is high dimensional but the underlying signal has a low-dimensional structure. Thus, l is often much smaller than m. In

☆ An early version of this paper was presented at DYCOPS 2016, Trondheim, Norway.

* Corresponding author.
E-mail address: anpalazoglu@ucdavis.edu (A. Palazoglu).

https://doi.org/10.1016/j.jprocont.2017.03.005
0959-1524/

PCA, since the derived PCs are uncorrelated, the distance between data points is preserved, leading to a diagonal covariance matrix. However, to satisfy these geometric constraints, most PCs contain non-zero loadings in all the coordinates. This, in turn, often complicates and confounds the interpretation of PCs, especially when the dimension m is large.

In most applications, the original variables in a dataset have a physical meaning and PCA is especially useful if the resulting PCs are composed of a small number of the original variables. For about a decade, improving the interpretability of PCs has been a topic of active research [6–10]. Rotation of PCs is a common practice wherein the rotated components are easier to interpret without any loss of information. Once PCs are rotated it is not possible to preserve the property that the components be pairwise uncorrelated and/or the loadings are orthogonal [8]. When rotating one also has to choose the normalization to preserve either orthogonality or zero correlation. In addition, different normalization criteria can lead to different quantitative results. Moreover, in conventional PCA, the variance captured by each PC decreases monotonically. However once the components are rotated this property does not always hold true.

The rotation methods cannot produce sparse loadings because they are designed to simplify the loadings while preserving a specified percentage of variance [11]. In recent years, several approaches have been proposed that obtain desired sparsity at the cost of explained variance. The advantage of the optimization approach is that it pushes some loadings to be exactly zero whereas a rotation usually does not. The simplified component technique (SCoT) is a penalized formulation wherein the explained variance is maximized while penalizing the number of non-zero loadings (NNZL) in a PC [12]. The successive components obtained using SCoT can be constrained to be uncorrelated with one another to obtain desired sparsity. Jolliffe and Uddin [12] demonstrated that SCoT drives many loadings to be identically zero and outperforms rotated PCA in terms of the varimax criterion [13]. However, SCoT suffers from having many local optima and the choice of the penalty function is problem specific. Jolliffe and Uddin [12] did not propose an algorithm or method to determine the best penalty but rather suggested examining a few penalty values to obtain the desired simplicity of the derived components.

Jolliffe et al. [14] proposed the Simplified Component Technique – LASSO (SCoTLASS) which adds a "least absolute shrinkage and selection operator" (LASSO) constraint to SCoT. The method SCoTLASS also modifies the original PCs by driving many loadings to exactly zero and has clear advantages over rotated PCA and SCoT. In SCoTLASS an extra constraint is introduced in the form of a bound on the sum of the absolute values of loadings in that component. This constraint shrinks some of the loadings on the components to be zero which makes it more favorable for variable selection. However, the introduction of the additional constraint requires a decision on a tuning parameter (t) that limits the search space for an optimal solution. Jolliffe et al. [14] solved the SCoTLASS by running the algorithm for a decreasing sequence of values of t resulting in loadings with different sparsities and correlations between the PCs. There is no satisfactory rule for selecting t even though the choice of t is crucial and has to be studied subjectively to obtain a suitable sparsity-variance tradeoff.

Shen and Huang [15] proposed obtaining sparse PCs using sparse PCA via regularized SVD (sPCA-rSVD) approach. They introduced regularization penalties to promote sparsity in PC loadings. They also suggested a cross validation and an *ad hoc* approach for selecting the degree of sparsity as the tuning parameter. Journée et al. [16] proposed a generalized power (GPower) method that treats sparse PCA with either LASSO or cardinality constraints to produce sparse loading vectors. On the basis of empirical comparisons presented in the literature, GPower approach appears to outperform other algorithms in computational speed.

There are several other methodologies proposed in the literature to obtain sparse loadings [4,14–20]. Trendafilov [21] and Jolliffe et al. [11] provided a review of main approaches and recent developments for improving the simplicity of the components. The approach used in this paper is the one introduced by Zou et al. [4] who obtained sparse loadings by reformulating PCA as a regression problem and imposing LASSO (elastic net) constraints on the L_1 norm of the regression coefficients (sparse loadings). This methodology known as sparse principal component analysis (SPCA) has several advantages such as it efficiently solves the optimization problem with a cost of a single least square fit, can be applied in the case when m is much larger than sample size and the desired NNZL can be specified for each component. This SPCA algorithm will be discussed in detail in the preliminaries section.

Once the desired sparse components are obtained, process monitoring task can be carried out. Liu et al. [22] offered the use of adaptive sparse PCA (ASPCA) for enhanced process monitoring and fault isolation. They developed a Bayesian information criterion for the selection of number of PCs and used Quasi-T^2 and SPE monitoring statistics for fault isolation. Recently, Yu et al. [23] proposed the use of robust, nonlinear and sparse PCA (RNSPCA) approach for fault diagnosis and robust feature discovery of industrial processes. With RNSPCA the nonlinear correlations in the process were captured using Spearman's and Kendall's tau correlation matrices. These correlation matrices were then used to obtain the sparse eigenvectors to reveal meaningful patters in the data. Yu et al. [23] observed slightly better or comparable fault detection rates as compared to kernel principal component analysis (KPCA), kernel independent component analysis (KICA) and robust nonlinear principal component analysis (RNPCA).

This paper, inspired by the studies mentioned above, outlines an approach in determining the NNZL on each PC when using SPCA. Second, we introduce a fault detection methodology and, to evaluate its performance, compare average run length (ARL) and fault detection rates using SPCA and PCA. The salient features of the proposed method are demonstrated through a synthetic example and the benchmark Tennessee Eastman process [24].

The paper is organized as follows: the next section briefly introduces PCA and SPCA concepts for the sake of completeness, followed by the introduction of the synthetic example and Tennessee Eastman benchmark process simulation case studies. The methodology for determining NNZL for each principal component is introduced next. Subsequently, the results obtained from SPCA on the case studies are compared with the conventional PCA. Finally, the conclusions and directions for future work are presented.

2. Preliminaries

2.1. Principal component analysis (PCA)

Mathematically, PCA is the eigenvector decomposition of the covariance or the correlation matrix obtained from data matrix $X \in \boldsymbol{R}^{n\times m}$ that contains n equally spaced (at same time interval) observations of m process variables and is scaled to zero mean and unit variance, into a transformed subspace of reduced dimension. The sample covariance matrix of X is defined as:

$$cov(X) = \Sigma = \frac{X^T X}{n-1} \tag{1}$$

The decomposition is then expressed as follows:

$$X = TP^T = \tilde{X} + E \tag{2}$$

where $T \in \boldsymbol{R}^{n\times m}$ and $P \in \boldsymbol{R}^{m\times m}$ are the score matrix and the loading matrix, respectively. The matrices $\tilde{X}$ and E represent the estimation