Chemometrics and Intelligent Laboratory Systems 159 (2016) 31–44

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

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

Developments of two supervised maximum variance unfolding algorithms for process classification

Chihang Wei[a], Junghui Chen[b,*], Zhihuan Song[a,*]

[a] State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, 310027 Zhejiang, China
[b] Department of Chemical Engineering, Chung Yuan Christian University, Chungli, Taoyuan 32023, Taiwan, ROC

ARTICLE INFO

Keywords:
Dimension reduction
Maximum variance unfolding
Process classification
Semi-definite optimization

ABSTRACT

Maximum variance unfolding (MVU) has recently proven to be a powerful dimension reduction method for nonlinear data with numerous mutually correlated measured variables. However, in classification work, MVU performs poorly since it is an unsupervised method without considering class information of data. In this paper, two novel supervised maximum variance unfolding (SMVU) algorithms, SMVU1 and SMVU2 are developed respectively. They extend MVU to supervised methods. Both SMVU1 and SMVU2 not only aim to find the embeddings that unfold the manifold in the reduced dimensional space but also group the within-class samples together and separate between-class samples. In SMVU1, between-class manifold structures are defined by the class separation constraints, and within-class manifold structures to be unfolded are defined by the objective function; in SMVU2, between-class manifold structures to be unfolded and within-class manifold structures to be folded are put together in the objective function. Additionally, a novel kernel function approximation algorithm is developed based on the Nyström method to handle new samples. The effectiveness of the proposed methods are illustrated through a simple nonlinear system and a real industrial polyethylene process. The study results show the proposed SMVUs significantly outperform the conventional MVU in classification work.

1. Introduction

Automation in the operating refineries and chemical plants is consistently increasing as witnessed during the last few decades because of their large processing rates and complex configurations of unit operations [1]. Moreover, it is needed because of the increasing demand on consistent product quality and good process performance, lack of adequately skilled labor and the necessity to lower costs in order to keep pace with rapidly growing global competition. In the modern plants with the extensive use of distributed control systems, process data have become abundant and the measured variables are often characterized by high dimensions. However, most of these dimensions are unnecessary and there are often severe dependencies in the variables because of a set of constraints, like mass and energy balances, or operating policies [2–9]. This implies that the independent variables or the important information among these data typically lie in a low dimensional manifold. This means that only a few intrinsic variables are needed to characterize the data variations and dimension reduction is necessary to handle this kind of data. With these features, multivariate statistical process monitoring (MSPM) approaches have been proposed to learn the data structure, to reduce the dimension and to extract significant interests for performing supervisory tasks such as process monitoring, fault detection and diagnostics [2–9].

Principal component analysis (PCA) [10] is one of the earliest papers on dimension reduction in MSPM. Although PCA has been proven to be useful in the process monitoring, the linear assumption limits its applicable area and performance [11]. The kernel technique has also been applied. It can extend traditional linear dimension reduction methods to other nonlinear dimension reduction methods. Kernel PCA (KPCA) [11,12] was proposed to generalize PCA to the nonlinear case. However, the algorithm maps the high dimensional data onto a lower dimensional space without considering the manifold structure because the performance of KPCA largely depends on the kernel function, but appropriate selection of kernel functions has been sporadically discussed in the research literature. Unlike KPCA defined by an artificially determined kernel function, maximum variance unfolding (MVU) was proposed and it can automatically learn the kernel space from the input data instead. It has been proven to be a powerful dimension reduction method for nonlinear data with numerous mutually correlated measured variables [13,14]. Ideally, MVU represents the intrinsic dimension of the data. More importantly, the boundary of the distribution region of the training samples in the input

* Corresponding authors.
E-mail addresses: jason@wavenet.cycu.edu.tw (J. Chen), songzhihuan@zju.edu.cn (Z. Song).

http://dx.doi.org/10.1016/j.chemolab.2016.09.009
Received 2 March 2016; Received in revised form 27 September 2016; Accepted 29 September 2016
Available online 30 September 2016
0169-7439/

space is faithfully preserved. These features facilitate its process monitoring applications [15–17].

MVU can learn the manifolds bottom up from the topology of the input data, but the data representatives on the learned manifold are not guaranteed to have desired properties such as classification. MVU only projects the raw data onto the manifold with lowest dimension without preserving the data topology at all when training samples with several classes are applied, the unfolded manifold structure of each class may be mixed together. Obviously, given this manifold, no linear classifier can easily separate the data samples of different classes since MVU is an unsupervised method. These features limit the applications of MVU in classification work while the MVU kernel can only be expected to perform well for large margin classification if the decision boundary on the unfolded manifold is approximately linear. There is no a priori reason, however, to expect this type of linear separability for different classes.

In the past, as plant operators and engineers often spent a substantial amount of time and efforts before the classes could be properly diagnosed, several algorithms for the automatic process classification had been developed based on the classification objective and data topology [7,18–21]. The major motivation of this paper is to develop two types of supervised MVUs. Supervised learning is the machine learning task of inferring a "machine" from the labeled training data. In supervised learning, each sample is a pair consisted of a set of input variables and a corresponding desired output value. A supervised learning algorithm analyzes the training data and produces an inferred "machine". The machine can be used for mapping new examples which are not the original training data to correctly determine the class labels. In this paper, the unsupervised MVU modeling is extended to two supervised ones, including Supervised Maximum Variance Unfolding 1 (SMVU1) and Supervised Maximum Variance Unfolding 2 (SMVU2). The purpose of the algorithms is to discover the natural boundary from the given data with the label annotations. Both algorithms aim at obtaining the separable unfolding of the low dimension and preserving the data structure at the same time. They are developed respectively based on the maximum variance embedding objective used in the existing semi-definite programming (SDP) algorithm [22] to unfold data and pull different class data apart. In SMVU1, between-class manifold structures are defined by the class separation constraints, and within-class manifold structures to be unfolded are defined by the objective function; in SMVU2, between-class manifold structures to be unfolded and within-class manifold structures to be folded are put together in the objective function. But the two optimization problems (SMVU1 and SMVU2) are not convex. A simplified optimization by reformulating the SMVU1 and SMVU2 problems in terms of the elements of the inner product matrix is developed. Then the computation kernel matrix is irrelevant to the dimension of the data. Despite the favorable properties of the kernel methods in terms of theory, empirical performance and flexibility, the constructed kernel matrix is valid for the training samples only; it cannot handle the new samples. In this work, the effectiveness of the Nyström method to scale kernel regression is used to get the approximate kernel function [23–25]. Then SMVU1 and SMVU2 algorithms learning from the labeled (or known) classes are developed. They can predict the unknown classes using the Bayesian classifier [26]. SMVU1 and SMVU2 use different types of constraints and objective functions respectively to construct between-class and within-class manifold structures. They have different scopes of applications. They will be described and discussed in detail later in this paper.

The rest of the paper is organized as follows. Section 2 gives the background knowledge of the MVU algorithm. Then the detailed SMVU1 and SMVU2 algorithms are discussed in Section 3; the differences among MVU, and SMVU1 and SMVU2 are also discussed in this section. Next a kernel approximation method is given in Section 4. Section 5 contains the SMVU1 and SMVU2 based process classification using the Gaussian naive classifier. Also, case studies of a simple nonlinear system and an industrial problem are presented to evaluate the proposed SMVU1 and SMVU2 algorithms in Section 6. Finally, conclusions are made.

2. Revisit of maximum variance unfolding

Before discussing the proposed algorithms, the basic background knowledge and concepts of MVU are introduced in this section. A manifold is a mathematical surface that behaves linear locally. Representing such data in its raw high dimensional-form is not necessary. Euclidean distances are only meaningful on a very local scale, and it is very hard to handle the whole data. Ideally one would like to have a representation that matches the intrinsic dimension of the data so that Euclidean distances can be globally meaningful.

MVU, also known as semi-definite embedding, has recently been proposed as a special variation of KPCA utilized for nonlinear dimension reduction. The basic idea of MVU is based on the finding that the high dimensional data lie on a low dimensional manifold. The kernel matrix of KPCA, $\mathbf{K}$, is obtained by projecting the input data onto a higher dimensional feature space. The projection can be done by subjectively specifying a certain kernel function. Unlike KPCA, MVU directly constructs a kernel matrix from training samples so that the data manifold in the input space can be unfolded in the kernel feature space $\mathbf{\Phi}$ implicitly defined by $\mathbf{K}$. This also makes the manifold unfolded in the reduced space of MVU [13,14]. The problem of learning $\mathbf{K}$ is casted to an instance of SDP [22,27–29]. It is convex and it does not suffer from local optima. In particular, given an input set $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N, \mathbf{x}_n \in R^D$ and an unfolding space $\mathbf{Y} = \{\mathbf{y}_n = \mathbf{\Phi}(\mathbf{x}_n)\}_{n=1}^N$, $\mathbf{y}_n \in R^d$ where $d < D$, one considers a map $\mathbf{\Phi}$: $\mathbf{x} \to \mathbf{y}$ so that the outputs $\mathbf{Y}$ can be found and the inputs and the learned outputs are k-locally isometric, or at least approximately isometric. Here N is the number of samples while D and d are the dimensions of the input and the learned manifolds. Thus, the objective function is to unfold a manifold based on the observations, where any "fold" between two samples on a manifold serves to decrease the Euclidean distance between them. To unfold the manifold in $\mathbf{\Phi}$, an objective function that measures the sum of pairwise squared distances between the outputs $\{\mathbf{y}_n = \mathbf{\Phi}(\mathbf{x}_n)\}_{n=1}^N$ is maximized:

$$\max \Gamma = \max \frac{1}{2N} \sum_{ij} \|\mathbf{\Phi}(\mathbf{x}_i) - \mathbf{\Phi}(\mathbf{x}_j)\|^2 \tag{1}$$

By maximizing Eq. (1), the outputs are pulled as far apart as possible subject to some constraints, including isometry and centering.

- Isometry: This constraint is to preserve local manifold structure in the kernel space. The isometry between the discrete point sets can be translated into various sets of equality constraints on the inputs and the outputs. Let $\mathbf{S} \subset R^{N\times N}$ be a binary adjacency matrix which can tell whether there is an edge between $\mathbf{x}_i$ and $\mathbf{x}_j$ formed by pairwise connecting all the k-nearest neighbors. Thus, $\{\mathbf{x}_n\}_{n=1}^N$ and $\{\mathbf{y}_n = \mathbf{\Phi}(\mathbf{x}_n)\}_{n=1}^N$ are locally isometric if $\mathbf{x}_i$ and $\mathbf{x}_j$ are themselves neighbors ($\mathbf{S}_{ij} = 1$) or common neighbors of another point in the data set ($[\mathbf{S}^T\mathbf{S}]_{ij} = 1$). The local isometry constraints can be written as

$$\|\mathbf{\Phi}(\mathbf{x}_i) - \mathbf{\Phi}(\mathbf{x}_j)\|^2 = \|\mathbf{x}_i - \mathbf{x}_j\|^2 = D_{ij} \text{ for all } (i, j) \text{ with } \mathbf{S}_{ij} = 1 \text{ or } [\mathbf{S}^T\mathbf{S}]_{ij} = 1 \tag{2}$$

- Centering: the centering constraint,

$$\sum_i \mathbf{\Phi}(\mathbf{x}_i) = 0 \tag{3}$$

is also imposed to remove a translational degree of freedom from the final solution.

The optimization problem is to maximize the variance of the outputs $\{\mathbf{\Phi}(\mathbf{x}_n)\}_{n=1}^N$ (Eq. (1)) subject to the constraints that they are