Pattern Recognition Letters 33 (2012) 1695–1702

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

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

Feature selection from high-order tensorial data via sparse decomposition

Donghui Wang *, Shu Kong

Department of Computer Science and Technology, Zhejiang University, Hangzhou 310027, PR China

ARTICLE INFO

Article history:
Received 19 May 2011
Available online 21 June 2012

Communicated by G. Borgefors

Keywords:
Dimensionality reduction
Feature selection
Tensor decomposition
High-order principal component analysis
Sparse principal component analysis

ABSTRACT

Principal component analysis (PCA) suffers from the fact that each principal component (PC) is a linear combination of all the original variables, thus it is difficult to interpret the results. For this reason, sparse PCA (sPCA), which produces modified PCs with sparse loadings, arises to clear away this interpretation puzzlement. However, as a result of that sPCA is limited in handling vector-represented data, if we use sPCA to reduce the dimensionality and select significant features on the real-world data which are often naturally represented by high-order tensors, we have to reshape them into vectors beforehand, and this will destroy the intrinsic data structures and induce the curse of dimensionality. Focusing on this issue, in this paper, we address the problem to find a set of critical features with multi-directional sparse loadings directly from the tensorial data, and propose a novel method called sparse high-order PCA (sHOPCA) to derive a set of sparse loadings in multiple directions. The computational complexity analysis is also presented to illustrate the efficiency of sHOPCA. To evaluate the proposed sHOPCA, we perform several experiments on both synthetic and real-world datasets, and the experimental results demonstrate the merit of sHOPCA on sparse representation of high-order tensorial data.

1. Introduction

Principal component analysis (PCA) is a widely-used feature extraction and dimensionality reduction tool, and it is beneficial to better performance in some specific applications (Guyon and Elisseeff, 2003), as well as its variations and improved versions (Hoffmann, 2007; Chen and Zhu, 2004). Recently, PCA has been used in microarray data analysis (Jatin et al., 2002), in which each variable corresponds to a specific gene. However, when applying PCA to microarray data, it is difficult to interpret the results that each derived PC is a linear combination of all the genes or variables and the loadings are typically nonzero. As a result, PCA fails to discover the components of practical significance (Jeffers, 1967).

For this reason, researchers have developed several sparse approaches by imposing some sparsity constraints such as lasso (Tibshirani, 1996) and elastic net (Zou and Hastie, 2005) to the loadings, then some of the loadings are vanished and the key variables are selected. Roughly speaking, there are two families of sPCA methods in the literature (Dou et al., 2010). The first one uses the maximum-variance property of PCs, such as DSPCA (d'Aspremont et al., 2004), SCoTLASS (Jolliffe et al., 2003), sPCA-rSVD (Shen and Huang, 2008), sPCA-WTH (Witten et al., 2009), sPCA-DC (Sriperumbudur et al., 2007), etc. The other is based on regression-type problems such as sPCA-ZHT (Zou et al., 2006) and sPCA-OS (Dou et al., 2010). Even though these sPCA approaches can select some critical features, they are all limited to vector-represented data.

Besides sPCA methods, in the literatures, there are many other approaches to address the problem of feature selection, such as Laplacian Score (LapScor) (He et al., 2005), Multi-Cluster Feature Selection (MCFS) (Cai et al., 2010) and Minimum Redundancy Spectral Feature Selection (MRSF) (Zhao et al., 2010). These methods commonly use various graphs to characterize the manifold structure (Cai et al., 2007) at first and then select the features by ranking or regression steps. For example, LapScor computes the Laplacian score for each feature and then rank them, and both MCFS and MRSF exploit sparse constraints in multi-output regression. However, the performance of these methods is determined by the effectiveness of graph construction, and these methods only deal with vector-represented data.

Let's consider higher-order tensors, which naturally represent the real-world data, such as images and videos. Prior to applying sPCA, we have to vectorize these data in advance, which will induce the curse of dimensionality and destroy the intrinsic data structures, e.g. ignoring the special relationships between the pixels in the image (He et al., 2005). Moreover, compared with vector-represented learning methods, the tensor-represented one holds several advantages (Haiping et al., 2011): natural representation, preserving natural data structure, estimating fewer parameters, less small-sample-size problem and ability to handle massive data. Based on tensor algebra and its successful applications (Rittner et al., 2010; Rana et al., 2009; Andaló et al., 2010; Savas and Eldén,

* Corresponding author. Fax: +86 571 87951916.
E-mail addresses: dhwang@zju.edu.cn (D. Wang), aimerykong@zju.edu.cn (S. Kong).

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2012.06.010

2007), we may ask, is it possible to select some key PCs that capture the maximum variability of the observations directly from these tensor-represented data? The answer is positive, and we believe it is a non-trivial work to find a method that avoids vectorizing process but captures the intrinsic variability by producing a set of sparse loadings.

Focusing on this issue, we first investigate high-order singular value decomposition (HOSVD) and high-order orthogonal iteration (HOOI). Based on the two algorithms, we propose a novel method called sparse high-order PCA (sHOPCA) to decompose the tensorial data with a set of selected features. In addition, we provide computational complexity analysis, which theoretically demonstrates the computational efficiency of sHOPCA. We also consider another problem of how to measure the explained variance in high-order data. By investigating the adjusted explained variance proposed in Shen and Huang (2008), we develop a general criteria to fit the measurement. Finally, to fairly evaluate our model, we perform several experiments both on synthetic and real-world benchmarks, and compare our sHOPCA with some popular sPCA approaches. The promising results demonstrate sHOPCA indeed discovers the intrinsic key features which capture more variability of the data under approximately the same compression ratio and sparse degree.

We begin our work by introducing the tensor algebra and notations used in this paper, which mainly follow Kolda and Bader (2009). Specially, $\mathscr{X}_{(k)}$ symbolizes the matrix corresponding to the flattened tensor along the kth mode and $\mathbf{I}_m$ denotes the $m \times m$ identity matrix. Moreover, the kth element in a sequence is denoted by a superscript in parentheses, e.g. $\mathbf{X}^{(k)}$ denotes the kth matrix in a sequence. We assume there are n observations, each one is represented as a *Nth-order tensor,* i.e. $\{\mathscr{X}_i \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}, i = 1, 2, \ldots, n\}$. Consequently, the sample set can be represented as an $(N+1)$th-order tensor $\mathscr{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N \times n}$.

2. High-order data decomposition

To generalize SVD for tensors, we first investigate SVD in the tensor viewpoint. As a matrix $\mathbf{X}$ has two vector spaces, i.e. a column space and a row space, then SVD decomposes $\mathbf{X}$ into its two vector spaces as $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T = \Sigma \times_1 \mathbf{U} \times_2 \mathbf{V}$, where $\mathbf{U}$ and $\mathbf{V}$ represent the orthogonal column space and row space, respectively. Obviously, SVD can be easily extended to a more generalized version, i.e. high-order SVD (HOSVD), which generates N associated vector spaces of a Nth-order tensor $\mathscr{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$:

$$\mathscr{X} \approx \mathscr{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times \cdots i \times_N \mathbf{U}^{(N)} = [\![\mathscr{G}; \mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(N)}]\!], \tag{1}$$

where the columnly orthogonal matrix $\mathbf{V}^{(k)} \in \mathbb{R}^{I_k \times J_k}$ represents the kth-mode vector space, and $\mathscr{G} \in \mathbb{R}^{I_1 \times \cdots J_N}$ is the core tensor of $\mathscr{X}$ which shows the interaction between different spaces, and $J_k <= I_k$. Although HOSVD has been already showed a convincing generalization of the matrix SVD, it is not optimal in terms of giving the best fit measured by the norm of the difference. However, it can be used as a good starting for a more efficient iterative algorithm called HOOI, which solves the following objective:

$$\begin{aligned} &\min_{\mathscr{G}, \mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(N)}} \|\mathscr{X} - [\![\mathscr{G}; \mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(N)}]\!]\|^2 \\ &\text{s.t. } \mathscr{G} \in \mathbb{R}^{J_1 \times \cdots \times J_N}, \\ &\mathbf{U}^{(k)} \in \mathbb{R}^{I_k \times J_k}, \quad \text{and} \quad \mathbf{U}^{(k)^T}\mathbf{U}^{(k)} = \mathbf{I}_{J_k}, \quad \text{for} \quad \forall k = 1, \ldots, N, \end{aligned} \tag{2}$$

where $\|\cdot\|$ denotes the Frobenius norm of a tensor. HOOI can be seen as an iterative optimization problem for HOSVD, which provides us with the basic formulation to derive modified sparse PC's and sparse loadings.

3. Sparse high-order principal component analysis

3.1. Derivation

Let's first review the standard PCA. Denote 2nd-order dataset as $\mathbf{X} \in \mathbb{R}^{n \times p}$, where n and p are the number of observations and variables, respectively. If we use SVD to decompose $\mathbf{X}$ as $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, then the columns of $\mathbf{U}$ are the PCs of unit length with the columns of $\mathbf{V}$ as the loadings and the variance of the ith PC is the ith diagonal element of Σ. Usually the first q ($q < p$) PCs are chosen to represent the data, therefore dimensionality is reduced.

To address the drawback of PCA that it fails to discover the significant feature variables, researchers propose several sPCA methods by adding some sparse penalties on PCA. Based on the fact that PCA can be written as a regression-type optimization problem, Zou et al. (2006) propose a regression-based sPCA (dubbed sPCA-ZHT), which enables some loadings to be exactly zero and derives the modified PCs by solving:

$$\begin{aligned} (\hat{\mathbf{U}}, \hat{\mathbf{W}}) = \underset{\mathbf{U},\mathbf{W}}{\operatorname{argmin}} &\|\mathbf{X} - \mathbf{X}\mathbf{W}\mathbf{U}^T\|_2^2 + \lambda \sum_{j=1}^{q} \|\mathbf{w}_j\|_2^2 \\ &+ \sum_{j=1}^{q} \lambda_{1,j} \|\mathbf{w}_j\|_1 \\ &\text{s.t. } \mathbf{U}^T\mathbf{U} = \mathbf{I}_k, \end{aligned} \tag{3}$$

where $\mathbf{W}$ is the sparse loadings.

Consider an image dataset $\mathscr{X} \in \mathbb{R}^{I_1 \times I_2 \times n}$, consisting n grayscale images of $I_1 \times I_2$-pixel resolution. By HOSVD, we can decompose $\mathscr{X}$ as:

$$\mathscr{X} = \mathscr{G} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}. \tag{4}$$

Rewrite Eq. (4) in terms of the flattened tensor form along mode-3:

$$\mathscr{X}_{(3)} = \mathbf{U}^{(3)} \mathscr{G}_{(3)} (\mathbf{U}^{(2)} \otimes \mathbf{U}^{(1)})^T. \tag{5}$$

Obviously, if we use sPCA-ZHT to reduce dimensionality on the vectorized image dataset, we will get the same expression as Eq. (5), where the modified PCs and the corresponding loadings are $\mathbf{U}^{(3)}\mathscr{G}_{(3)}$ and $(\mathbf{U}^{(2)} \otimes \mathbf{U}^{(1)})$, respectively. Here, the two matrices $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ project the original dataset into a new tensor space by selecting and transforming variables. Then we may wonder how to get the sparse loadings $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ directly from the tensorial dataset, without reshaping each image into a huge vector. More intuitively, we are also interested in how to extend this to a much higher-order dataset.

3.2. Objective function

Given the high-order dataset as $\mathscr{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N \times n}$, consisting of n observations represented by a Nth-order tensor, the proposed sHOPCA aims to minimize the objective function f:

$$\begin{aligned} \left(\hat{\mathbf{U}}^{(j)}|_{j=1}^N, \hat{\mathbf{W}}^{(j)}|_{j=1}^N\right) = \underset{\substack{\mathbf{U}^{(j)}|_{j=1}^N \\ \mathbf{W}^{(j)}|_{j=1}^N}}{\operatorname{argmin}} \Big\{ & f \equiv \|\mathscr{X} - \mathscr{X} \times_1 (\mathbf{U}^{(1)}\mathbf{W}^{(1)^T}) \\ & \times \cdots \times_N (\mathbf{U}^{(N)}\mathbf{W}^{(N)^T})\|^2 + \sum_{k=1}^{N}\sum_{j=1}^{J_k} \lambda_{2,k} \|\mathbf{w}_j^{(k)}\|_2^2 \\ & + \sum_{k=1}^{N}\sum_{j=1}^{J_k} \lambda_{1,kj} \|\mathbf{w}_j^{(k)}\|_1 \Big\} \\ & \text{s.t. } \mathbf{U}^{(k)^T}\mathbf{U}^{(k)} = \mathbf{I}_{J_k}, \\ & k = 1, 2, \ldots, N, \end{aligned} \tag{6}$$