Signal Processing 141 (2017) 204–216

Contents lists available at ScienceDirect

Signal Processing

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

Blind source separation of tensor-valued time series☆

Joni Virta[a,*], Klaus Nordhausen[a,b]

[a] Department of Mathematics and Statistics, University of Turku, 20014 Turku, Finland
[b] Institute of Statistics & Mathematical Methods in Economics, Vienna University of Technology, 1040 Vienna, Austria

ARTICLE INFO

Article history:
Received 19 April 2016
Revised 8 May 2017
Accepted 9 June 2017
Available online 10 June 2017

MSC:
62H12
62M10

Keywords:
FOBI
JADE
Multilinear algebra
SOBI

ABSTRACT

The blind source separation model for multivariate time series generally assumes that the observed series is a linear transformation of an unobserved series with temporally uncorrelated or independent components. Given the observations, the objective is to find a linear transformation that recovers the latent series. Several methods for accomplishing this exist and three particular ones are the classic SOBI and the recently proposed generalized FOBI (gFOBI) and generalized JADE (gJADE), each based on the use of joint lagged moments. In this paper we generalize the methodologies behind these algorithms for tensor-valued time series. We assume that our data consists of a tensor observed at each time point and that the observations are linear transformations of latent tensors we wish to estimate. The tensorial generalizations are shown to have particularly elegant forms and we show that each of them is Fisher consistent and orthogonal equivariant. Comparing the new methods with the original ones in various settings shows that the tensorial extensions are superior to both their vector-valued counterparts and to two existing tensorial dimension reduction methods for i.i.d. data. Finally, applications to fMRI-data and video processing show that the methods are capable of extracting relevant information from noisy high-dimensional data.

1. Introduction

1.1. Blind source separation and time series

In the classical *blind source separation (BSS)* model one assumes that the observed random vectors $\boldsymbol{x}_i$, $i = 1, \ldots, n$, are linear transformations of some latent vectors of interest, $\boldsymbol{x}_i = \boldsymbol{\Omega}\boldsymbol{z}_i$, where $\boldsymbol{\Omega} \in \mathbb{R}^{p\times p}$ is a full rank *mixing matrix*. Coupling the model with different sets of assumptions on $\boldsymbol{z}_i$ gives various well-known models: *(i)* assuming that $\boldsymbol{z}_i$ are i.i.d. and have mutually independent components yields the *independent component (IC) model, see e.g.* [15]; *(ii)* assuming $\boldsymbol{z}_i$ are i.i.d. and have spherical distribution yields an elliptical model for $\boldsymbol{x}_i$ [36] and *(iii)* as a special case of both previous, assuming that $\boldsymbol{z}_i$ are i.i.d. and have standard Gaussian distribution yields the general multivariate Gaussian distribution for $\boldsymbol{x}_i$.

In the context of time series it is natural to incorporate the time dependency of the data into the model structure and a commonly used BSS model [3] assumes that the observed series $\boldsymbol{x}_t$ is generated as

$$\boldsymbol{x}_t = \boldsymbol{\Omega}\boldsymbol{z}_t, \quad t = 0, \pm 1, \pm 2, \ldots, \tag{1}$$

where the latent time series $\boldsymbol{z}_t$ satisfies the following three assumptions.

(**V1**). $E[\boldsymbol{z}_t] = \boldsymbol{0}$
(**V2**). $cov[\boldsymbol{z}_t] = \boldsymbol{I}$
(**V3**). $E[\boldsymbol{z}_t\boldsymbol{z}_{t+\tau}^T] = \mathrm{E}[\boldsymbol{z}_{t+\tau}\boldsymbol{z}_t^T] = \boldsymbol{D}_\tau$ *is diagonal for all* $\tau = 1, 2, \ldots$.

Without loss of generality, (V1) implies that the observed series is centered and (V2) fixes the scales of the columns of $\boldsymbol{\Omega}$. After these two assumptions we can still freely change the signs and order of the elements of $\boldsymbol{z}_t$ and the corresponding columns of $\boldsymbol{\Omega}$ without altering the overall model. Thus the order and the signs of the latent series are unidentifiable which, however, is usually not a problem in practice. (V1)–(V3) together also imply that the time series $\boldsymbol{z}_t$ and $\boldsymbol{x}_t$ are weak second-order stationary and $\boldsymbol{x}_t$ satisfies $\mathrm{E}[\boldsymbol{x}_t] = \boldsymbol{0}$ and $\mathrm{Cov}[\boldsymbol{x}_t] = \boldsymbol{\Omega}\boldsymbol{\Omega}^T$.

BSS models for time series have become more and more popular in the recent years. A partial explanation for this is that general multivariate time series models are very demanding both theoretically and computationally and if one uses the BSS methodology as a preprocessing step the extracted sources can instead be further modelled separately using well-established univariate time series

☆ Funding: This work was supported by the Academy of Finland [grant 268703].
* Corresponding author.
E-mail addresses: joni.virta@utu.fi (J. Virta), klaus.nordhausen@tuwien.ac.at (K. Nordhausen).

http://dx.doi.org/10.1016/j.sigpro.2017.06.008
0165-1684/

methods. For some recent contributions following this type of approach, see for example [6,11,12,18,28,35,52].

1.2. Tensor-valued methods for time series

In the models discussed above a p-variate vector is observed at each time point. Modern data structures are however often more complex. For example, in many applications at each time point data might be observed which is better represented by a tensor. Such applications are for instance spatio-temporal data where at each time point usually a matrix is obtained or fMRI (functional Magnetic Resonance Imaging) data where for each time point a 3-dimensional tensor is recorded. But also video clip data can be seen as a time series where each frame is a matrix- or tensor-valued observation, depending on the number of colors used.

The most common approach to analyzing such data is to convert, following some convention, the tensor into a large vector and then apply standard multivariate methods for vector-valued data. Besides the often practical problem that the vectorized data might be of quite high dimension also information gets lost in this process. As for example [49] point out, after vectorizing the tensors the resulting vectors have a Kronecker structure. Ignoring this structure then means that a much larger number of parameters needs to be estimated.

For i.i.d. data this has recently led to extensive research where methods either model the Kronecker structure or work directly with the tensors. For some recent work on structured multivariate estimation see for example [19,40,49,51] and references therein. For some recent contributions for i.i.d. tensor methods see for example [17,26,54,56] and references therein.

Also independent component analysis (ICA) has already been considered in the context of tensors, some early works being [2,42,55]. A fully tensorial model-based approach was however only recently developed in [43,44] where tensorial versions of the well-known ICA methods Fourth order blind identification (FOBI) [8] and Joint approximate diagonalization of eigen-matrices (JADE) [9] were introduced.

Methods for tensor-valued time series seem however not to have gotten much attention yet although some first steps are for example [1,38,47]. But to the best of our knowledge no tensorial BSS-methods for dependent data have been considered so far. To fill this gap, we propose in this paper tensor extensions for three BSS methods meant for multivariate time series. The first method is called the *Second order blind identification* (SOBI) [3] and is based on using second-order information in the form of autocovariance matrices to separate the hidden source series. As such it is best suited for multivariate linear processes and may not work with models having trivial autocovariances, like for example stochastic volatility models. The recently proposed second and third methods, *generalized FOBI* (gFOBI) and *generalized JADE* (gJADE), are the exact opposite and operate on the joint fourth-order moments of the component series, see [30]. Thus the successful use of either requires non-trivial higher moments, ruling out for example standard ARMA models. The tensorial extensions of these three methods are respectively called TSOBI, TgFOBI and TgJADE and are discussed in Section 4.

1.3. Structure of the paper

In Section 2 we introduce the used notation and define various concepts of multilinear algebra needed in this paper. Although fairly easy to grasp and use, after we define the m-flattening of a tensor most tensor operations can be carried out conveniently in a matrix form. Section 3 reviews the theory of SOBI, gFOBI and gJADE and prepares the ground for their tensor versions in Section 4 where the corresponding theory and algorithms are discussed. In Section 5 we first use simulations to compare the presented methods with their vector-valued counterparts for vectorized data and then use the methods to process simulated fMRI-data and a color video. In both applications the proposed methodology is shown to extract the key elements of the signals in compressed form. In Section 6 we finally conclude with some discussion and the proofs are gathered in Appendix A.

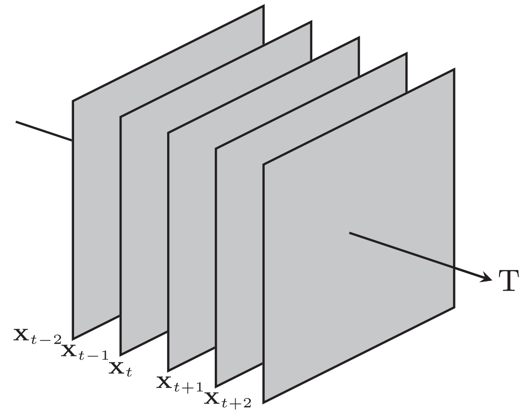

Fig. 1. Visualization of a tensor-valued time series. In the above scheme a matrix of the same size is observed at each time point and the resulting tensor-valued time series can be thought of as a video with frames corresponding to the individual observed matrices.

2. Notation and tensor algebra

2.1. Notation in general

Throughout the paper scalars are denoted by lower-case letters, $a, b, c, \ldots$, vectors by lower-case boldface letters, $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \ldots$, matrices by capital boldface letters, $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \ldots$, and tensors of general order by capital blackboard letters, $\mathbb{A}, \mathbb{B}, \mathbb{C}, \ldots$ (note that $\mathbb{R}$ still means the real line). The same convention on fonts is followed with random elements, but instead using the letters from the end of the alphabet, x, y, z, $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$, etc.

2.2. Regarding matrices

The standard basis vectors of $\mathbb{R}^p$ are denoted by $\boldsymbol{e}_i$, $i = 1, \ldots, p$, and using them we can construct the matrices $\boldsymbol{E}^{ij} := \boldsymbol{e}_i \boldsymbol{e}_j^T$, the only non-zero element of $\boldsymbol{E}^{ij}$ being the single one as the element (i, j). We further make use of the following sets of $p \times p$ matrices: $\mathcal{P}$, the set of all matrices with a single one in each row and column and rest of the entries zero; $\mathcal{J}$, the set of all diagonal matrices with the diagonal entries equal to ± 1; $\mathcal{D}$, the set of all diagonal matrices with positive diagonal elements and $\mathcal{C}$, the set of all matrices $\boldsymbol{C} = \boldsymbol{PJD}$ where $\boldsymbol{P} \in \mathcal{P}$, $\boldsymbol{J} \in \mathcal{J}$ and $\boldsymbol{D} \in \mathcal{D}$. The sets $\mathcal{P}, \mathcal{J}$ and $\mathcal{D}$ then respectively correspond to the sets of permutation matrices, heterogeneous sign-change matrices and heterogeneous scaling matrices. Finally, $\| \cdot \|$ is the Frobenius norm and by the equivalence $\boldsymbol{A} \equiv \boldsymbol{B}$ we mean that $\boldsymbol{A} = \boldsymbol{PJB}$ for some $\boldsymbol{P} \in \mathcal{P}$ and $\boldsymbol{J} \in \mathcal{J}$.

2.3. Regarding tensors

To manipulate tensors we next provide some basic tools of multilinear algebra, see also [16]. By a tensor-valued time series $\mathbb{X}_t$ we mean the set $\{\mathbb{X}_t\}_{t=-T}^{T}$ of realisations of a tensor-valued stochastic process $\mathbb{X}_t \in \mathbb{R}^{p_1 \times \cdots \times p_r}$ on some fixed set of time points $t = -T, \ldots, T$. That is, for each time point we observe a tensor of the same size, something akin to the frames of a video, see Fig. 1 for a visual representation.