Multiway Components Analysis Using Low-Rank Tensor Decompositions

Andrzej CICHOCKI†

†RIKEN Brain Science Institute, LABSP
Wako-shi, Japan
Email: a.cichocki@riken.jp

Abstract—Tensor decompositions (TDs) and tensor networks (TNs) are emerging and promising tools for distributed representation of large-scale data, feature extraction and data mining. In this paper we review briefly multiway components analysis, that is multilinear (tensor) generalizations of two-way components analysis, especially, PCA/SVD, ICA, NMF, SCA, CCA/PLS. We will also discuss some challenging problems and future perspectives related to big data analysis.

1. Introduction and Motivations

Tensors are adopted in diverse branches of science and data analysis such as signal and image processing, Psychometric, Chemometrics, Biometric, Quantum Physics/Information, Quantum Chemistry and Brain Science [1–7]. Modern scientific areas such as bioinformatics or computational brain science generate massive amounts of data collected in various forms of big, sparse tabular, graphs or networks with multiple aspects and high dimensionality. Tensors, which are multi-dimensional generalizations of matrices, provide often a meaningful sparse and distributed representation for such data. TDs and TNs provide some natural and flexible extensions of blind source separation (BSS) or, more generally, two-way (matrix) Components Analysis (2-way CA) to multiway components analysis (MWCA) methods [1, 4]. Moreover, TDs and TNs are potentially useful in dimensionality reduction and for analysis of linked (coupled) block of tensors [4].

A wealth of literature on (2-way) components analysis (CA) and BSS exists, especially on Principal Component Analysis (PCA), Independent Component Analysis (ICA), Sparse Component Analysis (SCA), Nonnegative Matrix Factorizations (NMF), and Morphological Component Analysis (MCA) [3, 8]. These techniques are maturing, and have been proven as enabling tools for BSS, feature extraction, classification, clustering, and 3D visualizations [3].

The "flattened view" provided by 2-way CA and matrix factorizations (PCA/SVD, NMF, SCA, MCA) may be inappropriate for large classes of real-world data which exhibit multiple couplings and cross-correlations. In this context, tensor decompositions give us the opportunity to develop more sophisticated models capturing multiple interactions and couplings, instead of standard pairwise interactions.

Our main motivation and objective in this paper is to review and further develop suitable tensor decomposition models and associated learning algorithms for large-scale multilinear BSS problems.

2. From Two-way to Multiway Components Analysis

Our adopted convenience is that tensors are denoted by bold underlined capital letters, e.g., $\underline{\mathbf{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, and that all data are real-valued. The order of a tensor is the number of its "modes", "ways" or "dimensions", which include space, time, frequency, trials, classes, and dictionaries. Matrices (2nd-order tensors) are denoted by boldface capital letters, e.g., $\mathbf{X}$, and vectors (1st-order tensors) by boldface lowercase letters; for instance the columns of the matrix $\mathbf{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_R] \in \mathbb{R}^{I \times R}$ are denoted by $\boldsymbol{a}_r$ and elements of a matrix (scalars) are denoted by lowercase letters, e.g., a_{ir}. We refer to [2,5] for more detail regarding the basic notations and tensor operations.

2.1. Constrained Matrix Factorizations and Decompositions – Two-Way Component Analysis

Two-way Components Analysis (2-way CA) exploits *a priori* knowledge about different characteristics, features or morphology of components (or source signals) [3] to find the hidden components thorough constrained matrix factorizations of the form

$$\mathbf{X} = \mathbf{A}\mathbf{B}^T + \mathbf{E} = \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r + \mathbf{E} = \sum_{r=1}^{R} \boldsymbol{a}_r \boldsymbol{b}_r^T + \mathbf{E}, \quad (1)$$

where the constraints imposed on factor matrices $\mathbf{A}$ and/or $\mathbf{B}$ include orthogonality, sparsity, statistical independence, nonnegativity or smoothness. The CA can be considered as a bilinear (2-way) factorization, where $\mathbf{X} \in \mathbb{R}^{I \times J}$ is a known matrix of observed data, $\mathbf{E} \in \mathbb{R}^{I \times J}$ represents residuals or noise, $\mathbf{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_R] \in \mathbb{R}^{I \times R}$ is the unknown (usually, full column rank R) mixing matrix with R basis vectors $\boldsymbol{a}_r \in \mathbb{R}^I$, and $\mathbf{B} = [\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_R] \in \mathbb{R}^{J \times R}$ is the matrix of unknown components (factors, latent variables, sources).

Two-way components analysis (CA) refers to a class of signal processing techniques that decompose or encode superimposed or mixed signals into components with certain constraints or properties. The CA methods exploit *a priori* knowledge about the true nature or diversities of latent variables. By diversity, we refer to different characteristics, features or morphology of sources or hidden latent variables. For example, the columns of the matrix $\mathbf{B}$ that represent different data sources should be: as statistically independent as possible for ICA; as sparse as possible for SCA; take only nonnegative values for (NMF) [3].

Remark: Note that matrix factorizations have an inherent symmetry, Eq. (1) could be written as $\mathbf{X}^T \approx \mathbf{B}\mathbf{A}^T$, thus interchanging the roles of sources and mixing process.

Another virtue of components analysis comes from a representation of multiple-subject, multiple-task datasets by a set of data matrices $\mathbf{X}_k$, allowing us to perform simultaneous matrix factorizations:

$$\mathbf{X}_k \approx \mathbf{A}_k \mathbf{B}_k^T, \qquad (k = 1, 2, \ldots, K), \tag{2}$$

subject to various constraints. In the case of statistical independence constraints, the problem can be related to models of group ICA through suitable pre-processing, dimensionality reduction and post-processing procedures [3,9].

We will show how constrained matrix factorizations and components analysis (CA) models can be naturally generalized to multilinear models using constrained tensor decompositions, such as the Tucker and Canonical Polyadic Decomposition (CPD) models, as illustrated in Figs. 1 (a) and (b).

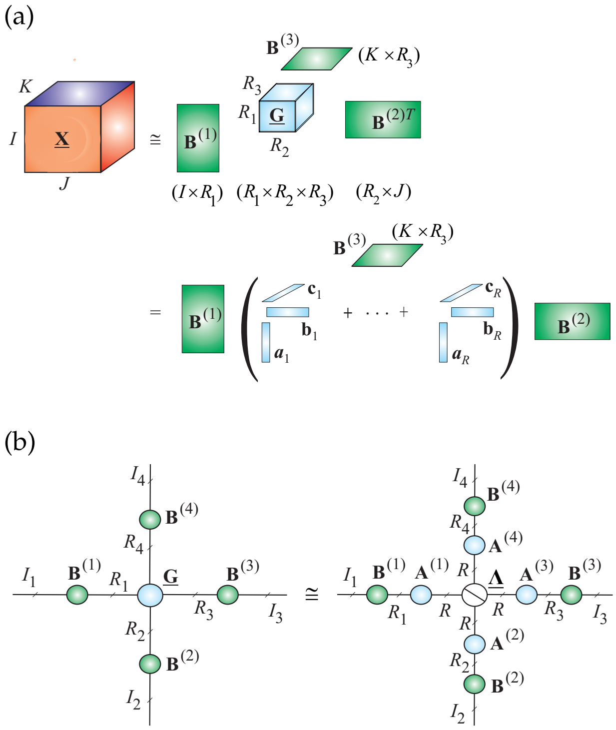

Figure 1: (a) Tucker decomposition of a 3rd-order tensor $\underline{\mathbf{X}} \cong \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \times_3 \mathbf{B}^{(3)}$. The objective is to compute constrained factor matrices $\mathbf{B}^{(n)}$ (whose columns represent specific components) and core tensor $\underline{\mathbf{G}}$, which indicates links between the components. In some applications, in the second stage, the core tensor is approximately factorized using the PARAFAC/CPD as $\underline{\mathbf{G}} \cong \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$. (b) TN diagrams for representation of the Tucker and the CP decompositions in two-stage procedure for a 4th-order tensor as: $\underline{\mathbf{X}} \cong \underline{\mathbf{G}} \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \cdots \times_4 \mathbf{B}^{(4)} = [\![\underline{\mathbf{G}}; \mathbf{B}^{(1)}, \mathbf{B}^{(2)}, \mathbf{B}^{(3)}, \mathbf{B}^{(4)}]\!] \cong (\underline{\mathbf{\Lambda}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_4 \mathbf{A}^{(4)}) \times_1 \mathbf{B}^{(1)} \times_2 \mathbf{B}^{(2)} \cdots \times_4 \mathbf{B}^{(4)} = [\![\underline{\mathbf{\Lambda}}; \mathbf{B}^{(1)}\mathbf{A}^{(1)}, \mathbf{B}^{(2)}\mathbf{A}^{(2)}, \mathbf{B}^{(3)}\mathbf{A}^{(3)}, \mathbf{B}^{(4)}\mathbf{A}^{(4)}]\!]$.

2.2. Multiway Components Analysis Using Constrained and Unique Tucker and CPD/PARAFAC Decompositions

The Tucker decomposition, especially constrained Tucker is a basic and flexible tensor decomposition model (see Fig. 1 (a)). The multiway components analysis (MWCA) based on the Tucker-N model can be considered as a natural and simple extension of multilinear SVD and/or multilinear ICA, or NMF in which we apply any efficient CA/BSS algorithms, which often assure essential uniqueness of tensor decompositions [8,10].

Fig. 2 illustrates the basic concept of MWCA and its flexibility in choosing the mode-wise constraints; a Tucker representation of MWCA naturally accommodates such diversities in different modes [1]. Since multiway array data can be always interpreted in many different ways, some *a priori* knowledge is needed to determine which diversities, characteristics, features or properties represent true latent (hidden) components with physical meaning.

There are two possible approaches to interpret and implement constrained Tucker decompositions for MWCA. (1) the columns of the factor matrices $\mathbf{B}^{(n)}$ represent the desired latent variables, the core tensor $\underline{\mathbf{G}}$ has a role of "mixing process", modeling the links among the components from different modes, while the data tensor $\underline{\mathbf{X}}$ represents a collection of 1-D or 2-D mixing signals; (2) the core tensor represents the desired (but hidden) N-dimensional signal (e.g., 3D MRI image or 4D video), while the factor matrices represent mixing or filtering processes through e.g.,

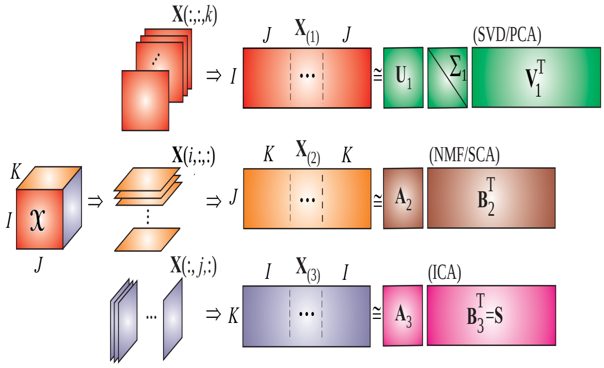

Figure 2: Illustration of Multiway Components Analysis (MWCA) for a third-order tensor, assuming that the components are: principal and orthogonal in the first mode, nonnegative and sparse in the second mode and statistically independent in the third mode. By unfolding a data tensor into matrices we perform PCA/SVD, NMF/SCA and ICA and explore link between them via the Tucker core tensor.

time-frequency transformations or wavelet dictionaries [4].

The MWCA based on the Tucker-N model can be computed directly in two steps: (1) for $n = 1, 2, \ldots, N$ perform model reduction and unfolding of data tensors sequentially and apply a suitable set of CA/BSS algorithms to reduced unfolding matrices $\tilde{\mathbf{X}}_{(n)}$, - in each mode we can apply different constraints and algorithms; (2) compute the core tensor using e.g., the inversion formula: $\hat{\underline{\mathbf{G}}} = \underline{\mathbf{X}} \times_1 \mathbf{B}^{(1)\dagger} \times_2 \mathbf{B}^{(2)\dagger} \cdots \times_N \mathbf{B}^{(N)\dagger}$ [10]. This step is quite important because core tensors illuminate complex links among the multiple components in different modes [3].

Particularly interesting is NMF approach and its generalization: Nonnegative Tucker decomposition (NTD) – powerful techniques to analyze multidimensional nonnegative matrix/tensor data, with the aim of giving sparse localized parts-based representation of high-dimensional objects. Recently, we have shown how low (multilinear) rank approximation (LRA) of matrices and tensors is able to significantly simplify the computation of the gradients of the cost function, upon which a family of efficient first-order NMF/NTD algorithms are developed [8]. Besides dramatically reducing the storage complexity and running time, the new algorithms are quite flexible and robust to noise because many well-established and efficient LRA methods can be easily applied. We will show how nonnegativity incorporating sparsity substantially relax the uniqueness conditions. The developed algorithms use the first-order information (gradients) only and are free of line search to search update steps (learning rates). The LRA procedure not only significantly reduces the computational complexity of subsequent nonnegative matrix/tensorfactorization procedure in terms of both time and space, but also substantially improves the robustness to noise and flexibility of NMF/NTD algorithms.

The CPD/PARAFAC model, which can be considered as special case of the Tucker model with a diagonal core tensor, is usually unique by itself, and does not require constraints to impose uniqueness. However, if components in one or more modes are known to be e.g., nonnegative, orthogonal, statistically independent or sparse, these constraints should be incorporated to relax dramatically uniqueness conditions. More importantly, constraints may increase the accuracy and stability of the CPD algorithms and provide desired physical interpretability of components [1, 11].

3. Distributed Tucker Decomposition using Tensor Train Representation

In Figs. 3 (a) and (b) we have shown new distributed models for the Tucker-N model using tensor train and tensor chain decompositions (compare with QTT-Tucker model proposed in [12]). These model allow us to decompose a large-scale data into core tensors which represent useful hidden components or features.

The Tensor Train (TT) decompositions [12, 13], called also Matrix Product State/Operator (MPS/MPO) in quantum information theory, are the simplest TN models. In fact, the TT decompositions were rediscovered several times under different names: MPS/MPO, valence bond states and density matrix renormalization group (DMRG). The DMRG usually means not only tensor format but also powerfull computational algorithms. The matrix TT model (MPO) for 2Nth-order tensor $\underline{\mathbf{A}} \in \mathbb{R}^{I_1 \times J_1 \times \cdots I_N \times J_N}$, which is tensorized version of a large-scale matrix $\mathbf{A} \in \mathbb{R}^{I \times J}$, with $I = I_1 I_2 \cdots I_N$, $J = J_1 J_2 \cdots J_N$, can be described mathematically in the following general forms expressing via multilinear and outer products [2]:

$$\begin{aligned} \underline{\mathbf{A}} &\cong \underline{\mathbf{A}}^{(1)} \times_4^1 \underline{\mathbf{A}}^{(2)} \times_4^1 \cdots \times_4^1 \underline{\mathbf{A}}^{(N)} \\ &= \sum_{r_1, r_2, \ldots, r_{N-1}=1}^{R_1, R_2, \ldots, R_{N-1}} \mathbf{A}^{(1)}_{1, r_1} \circ \mathbf{A}^{(2)}_{r_1, r_2} \circ \cdots \circ \mathbf{A}^{(N)}_{r_{N-1}, 1} \end{aligned} \quad (3)$$

where the 4th-order cores are defined as $\underline{\mathbf{A}}^{(n)} \in \mathbb{R}^{R_{n-1} \times I_n \times J_n \times R_n}$, with $R_0 = R_N = 1$, $(n = 1, 2, \ldots, N)$.

The main advantage of distributed tensor decomposition is that the size of each of the core tensors in the internal tensor network structure is usually much smaller than the original Tucker core tensor, so consequently the total number of parameters can be re-

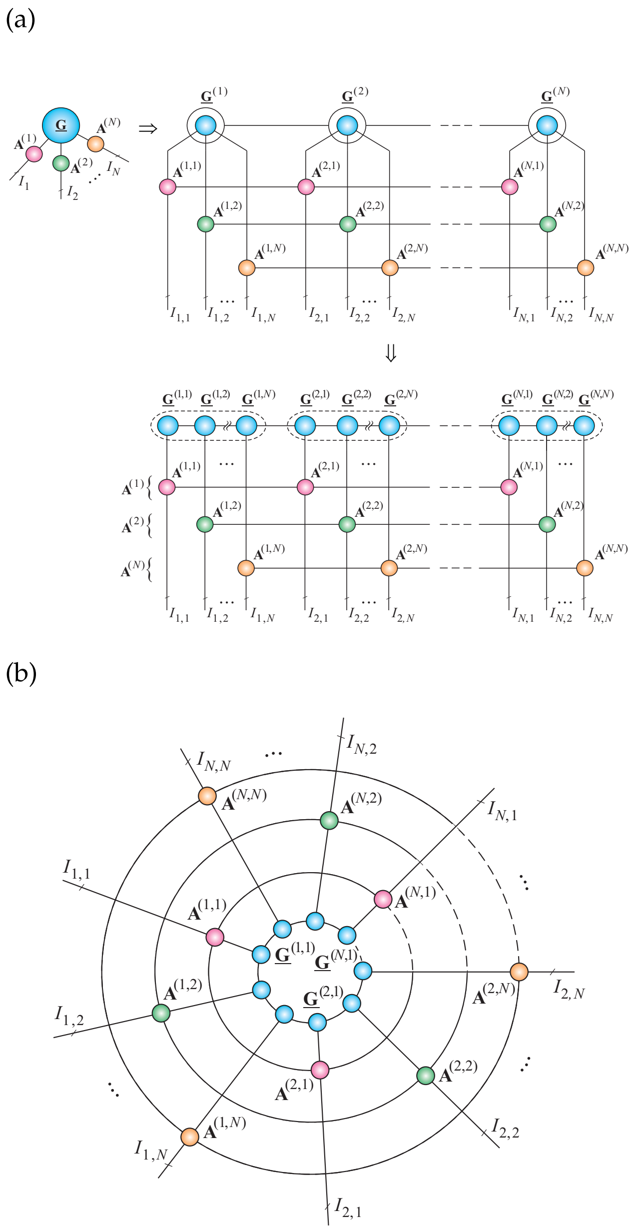

Figure 3: New distributed models of the Tucker-N decomposition $\underline{\mathbf{X}} = \underline{\mathbf{G}} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \cdots \times_N \mathbf{A}^{(N)} \in \mathbb{R}^{I_1 \times I_2 \times \cdots I_N}$, with $I_n = I_{1,n} I_{2,n} \cdots I_{N,n}$, $(n = 1, 2, \ldots, N)$ using: (a) the Tensor Train (TT) model and (b) the Tensor Chain (TC) model. The objective is to compute constrained cores $\underline{\mathbf{A}}^{(k,n)}$ and $\underline{\mathbf{G}}^{(k,n)}$ for $k, n = 1, 2, \ldots, N$

duced and model is suitable for big data representations [12] .

4. Conclusion

The main benefit of multiway (tensor) analysis methods is to incorporate various diversities or constraints in different modes or different factors matrices and core tensors, and thus naturally extend the standard (2-way) components analysis methods to multidimensional data. Furthermore, we developed powerful algorithms to analyze noisy, incomplete, missing data by using efficient low-rank tensor/matrix approximation techniques and by exploiting properties of distributed tensor decompositions.

References

[1] A. Cichocki, D. Mandic, C. Caiafa, A.-H. Phan, G. Zhou, Q. Zhao, and L. D. Lathauwer, "Multiway Component Analysis: Tensor Decompositions for Signal Processing Applications," *(in print)*, 2013.

[2] A. Cichocki, "Era of big data processing: A new approach via tensor networks and tensor decompositions," *CoRR*, vol. abs/1403.2048, 2014. [Online]. Available: http://arxiv.org/abs/1403.2048

[3] A. Cichocki, R. Zdunek, A.-H. Phan, and S. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Chichester: Wiley, 2009.

[4] A. Cichocki, "Tensors decompositions: New concepts for brain data analysis?" *Journal of Control, Measurement, and System Integration (SICE)*, vol. 47, no. 7, pp. 507–517, 2011.

[5] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, September 2009.

[6] S. Holtz, T. Rohwedder, and R. Schneider, "The alternating linear scheme for tensor optimization in the tensor train format," *SIAM J. Scientific Computing*, vol. 34, no. 2, pp. 683–713, 2012.

[7] A. Uschmajew, "Local convergence of the alternating least squares algorithm for canonical tensor approximation," *SIAM J. Matrix Anal. Appl.*, vol. 33, no. 2, pp. 639–652, 2012.

[8] G. Zhou, A. Cichocki, Q. Zhao, and S. Xie, "Efficient nonnegative Tucker decompositions: Algorithms and uniqueness," *CoRR*, vol. abs/1404.4412, 2014. [Online]. Available: http://arxiv.org/abs/1404.4412

[9] V. Calhoun, J. Liu, and T. Adali, "A review of group ICA for fMRI data and ICA for joint inference of imaging, genetic, and ERP data," *Neuroimage*, vol. 45, pp. 163–172, 2009.

[10] G. Zhou and A. Cichocki, "Fast and unique Tucker decompositions via multiway blind source separation," *Bulletin of Polish Academy of Science*, vol. 60, no. 3, pp. 389–407, 2012.

[11] ——, "Canonical Polyadic Decomposition based on a single mode blind source separation," *IEEE Signal Processing Letters*, vol. 19, no. 8, pp. 523–526, 2012.

[12] S. Dolgov and B. Khoromskij, "Two-level QTT-Tucker format for optimized tensor calculus," *SIAM J. Matrix Analysis Applications*, vol. 34, no. 2, pp. 593–623, 2013.

[13] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.