Algorithms for Sparse Higher Order Non-negative Matrix Factorization (HONMF)

Morten Mørup
www.imm.dtu.dk/∼mm
Informatics and Mathematical Modelling
Technical University of Denmark
Richard Petersens plads, building 321
DK-2800 Kgs Lyngby, Denmark
mm@imm.dtu.dk

Lars Kai Hansen
Informatics and Mathematical Modelling
Technical University of Denmark
Richard Petersens plads, building 321
DK-2800 Kgs Lyngby, Denmark
lkh@imm.dtu.dk

Sidse M. Arnfred
Department of Psychiatry
Hvidovre hospital
University Hospital of Copenhagen, Denmark
sidse.arnfred@hh.hosp.dk

Editor: n/a

Abstract

Higher order matrix (tensor) decompositions, are in frequent use today in a variety of fields including psychometric, chemometrics, image analysis, graph analysis and signal processing. For these higher order data the two most commonly used decompositions are the PARAFAC (also known as CANDECOMP) and the Tucker model. Often the data analyzed is non-negative and with good reason the components can also be assumed non-negative and their interactions additive. While the Tucker decomposition has been dominated by algorithms such as the Higher Order Singular Value Decomposition (HOSVD) the use of existing algorithms for non-negative Tucker decompositions has been limited since these decomposition does not in general yield unique decompositions. Presently, we extend the approach of Non-negative Matrix Factorization (NMF) to form algorithms for non-negative Tucker decomposition. Namely, a Higher Order NMF (HONMF). To improve uniqueness of the decompositions we develop updates that can impose sparseness in any combination of modalities. The algorithms for HONMF are tested on synthetic as well as real data revealing how sparseness indeed significantly improves uniqueness of the decomposition while also being useful for model selection.

Keywords: Tucker decomposition, Higher Order Non-negative Matrix Factorization (HONMF), Sparse Coding, PARAFAC, HOSVD.

1. Introduction

Higher order tensor decompositions are in frequent use today in a variety of fields including psychometric, chemometrics, image analysis, graph analysis and signal processing Kolda (2006). Tensors also called multidimensional matrices or multi-way arrays are higher order generalizations of vectors (first order tensors) and matrices (second order tensors), i.e. $\mathcal{X} \in \mathbb{C}^{I_1 \times I_2 \times \ldots \times I_N}$. The two most commonly used decompositions of higher order tensors are the PARAFAC (also known as CANDECOMP) Carroll and Chang (1970)Harshman (1970) and the Tucker model Tucker (1966).

The Tucker model is given as the decomposition

$$\mathcal{X}_{i_1,i_2,\ldots,i_N} \approx \mathcal{L}_{i_1,i_2,\ldots,i_N} = \sum_{j_1 j_2 \cdots j_N} \mathcal{G}_{j_1,j_2,\ldots,j_N} \mathbf{A}^{(1)}_{i_1,j_l} \mathbf{A}^{(2)}_{i_2,j_2} \cdot \ldots \cdot \mathbf{A}^{(N)}_{i_N,j_N}.$$

where $\mathcal{G} \in \mathbb{C}^{J_1 \times J_2 \times \ldots \times J_N}$ and $\mathbf{A}^{(n)} \in \mathbb{C}^{I_n \times J_n}$. By the use of the n-mode tensor product $\times_n$ given by

$$(\mathcal{Q} \times_n \mathbf{P})_{i_1,i_2,\ldots,j_n,\ldots i_N} = \sum_{i_n} \mathcal{Q}_{i_1,i_2,\ldots,i_n,\ldots i_N} \mathbf{P}_{j_n,i_n},$$

the model can also be stated as

$$\mathcal{X} \approx \mathcal{L} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 \ldots \times_N \mathbf{A}^{(N)}$$

Consequently, in the Tucker model the n^{th} modality is spanned by the vectors given by the columns of $\mathbf{A}^{(n)}$ while the vectors of each modality interact with the strength given by the core tensor $\mathcal{G}$ to reconstruct the data. As a result, the Tucker model account for all possible linear interaction across the vectors of the various modalities. The PARAFAC model is a special case of the Tucker model where the size of each modality of the core array $\mathcal{G}$ is the same, i.e. $J_1 = J_2 = \ldots = J_N$ while the only interaction are between columns of same indices such that the only non-zero elements are along the hyper-diagonal, i.e. $\mathcal{G}_{j_1,j_2,\ldots,j_N} \neq 0$ iff $j_1 = j_2 = \ldots = j_N$. Thus, the Tucker model is less restricted than the PARAFAC model. As a result, the Tucker model is not as the PARAFAC model in general unique Kruskal (1977); Sidiropoulos and Bro (2000) since a rotation of $\mathbf{A}^{(n)}$ can be compensated by a counter rotation of the core $\mathcal{G}$, i.e. $\mathcal{G} \times_n \mathbf{A}^{(n)} = (\mathcal{G} \times_n \mathbf{P}^{-1}) \times_n (\mathbf{A}^{(n)}\mathbf{P})$. In the following $\mathcal{X}^a_b$ will denote a tensor of the modalities a containing data of type b.

Lately, the Tucker model has among others been applied to

- spectroscopy data (Smilde et al. (2004); Andersson and Bro (1998) for instance $\mathcal{X}^{Batch\ number \times Time \times Spectra}_{Strength}$ Gurden et al. (2001); Nørgaard and Ridder (1994); Smilde et al. (1999))
- web mining ($\mathcal{X}^{Users \times Queries \times Wep\ pages}_{Click\ counts}$ Sun et al. (2005))
- image analysis ($\mathcal{X}^{People \times Views \times Illuminations \times Expressions \times Pixels}_{Image\ intensity}$ Vasilescu and Terzopoulos (2002); Wang and Ahuja (2003); Jia and Gong (2005) $\mathcal{X}^{Class \times Digits \times Pixels}_{Image\ intensity}$ Savas and Eldén (submitted))
- semantic differential data ($\mathcal{X}^{Judges \times Music\ pieces \times Scales}_{Grade}$Murakami and Kroonenberg (2003))

Common for all the data sets above is that they are all non-negative and the basis vectors/projections $\mathbf{A}^{(n)}$ and interactions $\mathcal{G}$ with good reason could have been assumed additive, i.e. non-negative.

Due to the huge amount of data often present when dealing with tensors the efficacy of the algorithms used to estimate the Tucker model is of outmost importance Andersson and Bro (1998). Traditionally the Tucker model has been estimated using various alternating least square algorithms where the columns of $\mathbf{A}^{(n)}$ most often are assumed orthogonal Andersson and Bro (1998). Recently, an efficient algorithm for higher order singular value decomposition (HOSVD) based on solving N eigenvalue problems to estimate the Tucker model has been introduced Lathauwer et al. (2000). For the above mentioned data sets HOSVD was the most commonly used. Although algorithms for non-negative Tucker decompositions exist Bro and Andersson (2000) the decompositions are contrary to HOSVD not in general unique. Consequently, the lack of uniqueness hampers interpretability of potentially non-negative decompositions. For this reason the existing non-negative Tucker decompositions have been unattractive. Presently, we will develop efficient algorithms for non-negative Tucker decompositions based on easy implementable multiplicative updates, i.e. a higher order non-negative matrix factorization (HONMF) based on the approach of non negative matrix factorization (NMF) Lee and Seung (2000). To achieve unique decompositions we will incorporate sparsity constraints to the HONMF as suggested for NMF by Eggert and Korner (2004).

The paper is structured as follows: First the algorithms for HONMF will be derived including updates for sparsity constraints. Next, the algorithms abilities to identify the components of a synthetically generated data set will be demonstrated. Finally, the algorithm will be tested on a data set of wavelet transformed EEG-data previously explored by the PARAFAC model Mørup et al. (2006) and data obtained from a flow injection analysis Nørgaard and Ridder (1994); Smilde et al. (1999). The uniqueness of the decompositions of these data will be evaluated. Since the HOSVD recently has been the method the most employed, the current sparse HONMF will be contrasted to this algorithm. The existing algorithms for non-negative Tucker decompositions Bro and Andersson (2000); Bro and Jong (1997) give decompositions similar to the unconstrained HONMF based on LS.

2. Method

Lee and Seung gave two algorithms for (NMF) Lee and Seung (2000). They further showed how non-negative decompositions contrary to PCA/SVD give a part based representation Lee and Seung (1999). Recently, NMF has been extended to the PARAFAC decompositions FitzGerald et al. (2005); Parry and Essa (2006); Welling and Weber (2001); Mørup et al. (2006). However, to our knowledge no previous work has adapted the NMF approach to the Tucker model.

Consider the non-negative matrix factorization (NMF) problem Lee and Seung (2000):

$$\mathbf{V} \approx \mathbf{\Lambda} = \mathbf{W}\mathbf{H}$$

where $\mathbf{V} \in \mathbb{R}^{I \times J}$,$\mathbf{W} \in \mathbb{R}^{I \times D}$, and $\mathbf{H} \in \mathbb{R}^{D \times J}$ are non-negative. Lee and Seung (2000) devised two algorithms to find $\mathbf{W}$ and $\mathbf{H}$: For the least square error (LS) and the Kullback-Leibler divergence (KL) they proved that the recursive updates given at the top of Table 1 converge to a local minimum. These algorithms can be derived by minimizing the cost function using a gradient based search with step sizes appropriately chosen to give multiplicative updates.

However, the NMF decomposition is apart from trivial permutation and scaling not in general unique Donoho and Stodden (2003). If the data does not adequately span the positive orthant a rotation of the solution is possible violating uniqueness. Consequently, constraints in the form of sparseness has proven useful Hoyer (2002, 2004); Eggert and Korner (2004). Eggert and Korner (2004) derived an efficient algorithm for Sparse NMF based on multiplicative updates by penalizing values in $\mathbf{H}$ by a function $C_{sparse}(\mathbf{H})$ while keeping $\mathbf{W}$ normalized such that the sparsity is not achieved by simply letting $\mathbf{H}$ go to zero while $\mathbf{W}$ goes to infinity. Making the reconstruction invariant of this normalization, i.e. $\widetilde{\mathbf{\Lambda}} = \widetilde{\mathbf{W}}\mathbf{H}$ where $\widetilde{\mathbf{W}}_{i,d} = \frac{\mathbf{W}_{i,d}}{\sqrt{\sum_i \mathbf{W}_{i,d}}} = \frac{\mathbf{W}_{i,d}}{\|\mathbf{W}_d\|_2}$ they found the multiplicative updates for the LS-algorithm further adapted to the KL algorithm Mørup and Schmidt (2005) given at the bottom of Table 1.

In the following we will consider the Tucker model under non-negativity constraint, i.e $\mathcal{X}$, $\mathcal{G}$ and $\mathbf{A}^{(n)}$ are all non-negative. By turning 'matrizicing' $\mathcal{X}^{I_1 \times I_2 \times \ldots \times I_N}$ into a matrix, i.e. $\mathbf{X}_{(n)}^{I_n \times I_1 \ldots I_{n-1} I_{n+1} \ldots I_N}$ the Tucker model can be expressed in matrix notation as

$$\mathbf{X}_{(n)} \approx \mathbf{\Lambda}_{(n)} = \mathbf{A}^{(n)}\mathbf{G}_{(n)}(\mathbf{A}^{(N)} \otimes \ldots \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes \ldots \otimes \mathbf{A}^{(1)}) = \mathbf{A}^{(n)}\mathbf{Z}_{(n)},$$

$C_{LS}(\mathbf{V},\mathbf{\Lambda}) = \frac{1}{2}\sum_{ij}(\mathbf{V}_{i,j} - \mathbf{\Lambda}_{i,j})^2$	$C_{KL}(\mathbf{V},\mathbf{\Lambda}) = \sum_{ij} \mathbf{V}_{i,j} log\frac{\mathbf{V}_{i,j}}{\mathbf{\Lambda}_{i,j}} - \mathbf{V} + \mathbf{\Lambda}_{i,j}$
$\mathbf{W} \leftarrow \mathbf{W} \bullet \dfrac{\mathbf{V}\mathbf{H}^{\mathrm{T}}}{\mathbf{\Lambda}\mathbf{H}^{\mathrm{T}}} \quad \mathbf{H} \leftarrow \mathbf{H} \bullet \dfrac{\mathbf{W}^{\mathrm{T}}\mathbf{V}}{\mathbf{W}^{\mathrm{T}}\mathbf{\Lambda}}$	$\mathbf{W} \leftarrow \mathbf{W} \bullet \dfrac{\frac{\mathbf{V}}{\mathbf{\Lambda}}\mathbf{H}^{\mathrm{T}}}{\mathbf{1} \cdot \mathbf{H}^{\mathrm{T}}}, \quad \mathbf{H} \leftarrow \mathbf{H} \bullet \dfrac{\mathbf{W}^{\mathrm{T}}\frac{\mathbf{V}}{\mathbf{\Lambda}}}{\mathbf{W}^{\mathrm{T}} \cdot \mathbf{1}}$
$C_{SparseLS} = C_{LS}(\mathbf{V},\widetilde{\mathbf{\Lambda}}) + \beta C_{sparse}(\mathbf{H})$	$C_{SparseKL} = C_{KL}(\mathbf{V},\widetilde{\mathbf{\Lambda}}) + \beta C_{sparse}(\mathbf{H})$
$\mathbf{W} \leftarrow \widetilde{\mathbf{W}} \bullet \dfrac{\mathbf{V}\mathbf{H}^{\mathrm{T}} + \widetilde{\mathbf{W}} diag(\mathbf{1} \cdot \tilde{\mathbf{\Lambda}}\mathbf{H}^{\mathrm{T}} \bullet \widetilde{\mathbf{W}})}{\tilde{\mathbf{\Lambda}}\mathbf{H}^{\mathrm{T}} + \widetilde{\mathbf{W}} diag(\mathbf{1} \cdot \tilde{\mathbf{V}}\mathbf{H}^{\mathrm{T}} \bullet \widetilde{\mathbf{W}})}$ $\mathbf{H} \leftarrow \mathbf{H} \bullet \dfrac{\widetilde{\mathbf{W}}^{\mathrm{T}}\mathbf{V}}{\widetilde{\mathbf{W}}^{\mathrm{T}}\tilde{\mathbf{\Lambda}} + \beta\frac{\partial C_{sparse}(\mathbf{H})}{\partial \mathbf{H}}}$	$\mathbf{W} \leftarrow \widetilde{\mathbf{W}} \bullet \dfrac{\frac{\mathbf{V}}{\tilde{\mathbf{\Lambda}}}\mathbf{H}^{\mathrm{T}} + \widetilde{\mathbf{W}} diag(\mathbf{1} \cdot \mathbf{H}^{\mathrm{T}} \bullet \widetilde{\mathbf{W}})}{\mathbf{1} \cdot \mathbf{H}^{\mathrm{T}} + \widetilde{\mathbf{W}} diag(\mathbf{1} \cdot \frac{\mathbf{V}}{\tilde{\mathbf{\Lambda}}}\mathbf{H}^{\mathrm{T}} \bullet \widetilde{\mathbf{W}})}$ $\mathbf{H} \leftarrow \mathbf{H} \bullet \dfrac{\widetilde{\mathbf{W}}^{\mathrm{T}}\frac{\mathbf{V}}{\tilde{\mathbf{\Lambda}}}}{\widetilde{\mathbf{W}}^{\mathrm{T}} \cdot \mathbf{1} + \beta\frac{\partial C_{sparse}(\mathbf{H})}{\partial \mathbf{H}}}$

Table 1: The NMF updates (top) and Sparse NMF updates (bottom) given for LS in left column and KL in right column. $C_{sparse}(\mathbf{H})$ is the function used to penalize the elements in $\mathbf{H}$. In the following analysis we'll use $C_{sparse}(\mathbf{H}) = \|H\|_1$. Consequently $\frac{\partial C_{sparse}(\mathbf{H})}{\partial \mathbf{H}} = \mathbf{1}$. $A \bullet B$ and $\frac{A}{B}$ denotes element-wise multiplication and division respectively while $\widetilde{\mathbf{W}}_{i,d} = \frac{\mathbf{W}_{i,d}}{\|\mathbf{W}_d\|_2}$ and $\widetilde{\mathbf{\Lambda}} = \widetilde{\mathbf{W}}\mathbf{H}$.

where $\mathbf{Z}_{(n)} = \mathbf{G}_{(n)}(\mathbf{A}^{(N)} \otimes ... \otimes \mathbf{A}^{(n+1)} \otimes \mathbf{A}^{(n-1)} \otimes ... \otimes \mathbf{A}^{(1)})^T$. As a result, the updates of each of the factors $\mathbf{A}^{(n)}$ follows straight forward from the regular NMF updates by exchanging $\mathbf{W}$ with $\mathbf{A}$ and $\mathbf{H}$ with $\mathbf{Z}$ in the $\mathbf{W}$ update.

By lexicographical indexing of the elements in $\mathcal{X}$ and $\mathcal{G}$, i.e. $vec(\mathcal{X})$ and $vec(\mathcal{G})$ also the problem of finding the core $\mathcal{G}$ can be formulated in the framework of factor analysis Kolda (2006):

$$vec(\mathcal{X}) \approx vec(\mathcal{L}) = \mathbf{A}vec(\mathcal{G})$$

Where $\mathbf{A} = \mathbf{A}^{(1)} \otimes \mathbf{A}^{(2)} \otimes ... \otimes \mathbf{A}^{(N)}$. Consequently, also the update of $\mathcal{G}$ follows by the regular NMF updates exchanging $\mathbf{W}$ with $\mathbf{A}$ and $\mathbf{H}$ with $vec(\mathcal{G})$ in the $\mathbf{H}$ update. Finally, this update can be expressed in terms of the n-mode multiplication since

$$\mathbf{A}^T vec(\mathcal{X}) = vec(\mathcal{X} \times_1 \mathbf{A}^{(1)^T} \times_2 \mathbf{A}^{(2)^T} \times_3 ... \times_N \mathbf{A}^{(N)^T}).$$

The algorithms for HONMF are summarized in Table 2. Here $diag(\mathbf{v})$ is a matrix having the vector $\mathbf{v}$ along the diagonal while $\mathbf{1}$ and $\mathcal{T}$ is a matrix and a tensor having ones in all indices.

According to the Sparse NMF some modalities can be kept sparse while the rest are normalized. Consequently, each or some of the $\mathbf{A}^{(n)}$ can be constrained sparse and/or $\mathcal{G}$, while re-normalizing the core and/or the other $\mathbf{A}^{(n)}$. As a result sparseness can be imposed on any combination of modalities including the core while normalizing the remaining modalities. In Table 2 the updates are given when sparsifying or normalizing a given modality. Here $\|\mathcal{G}\|_F = \sqrt{\sum_{j_1 j_2 \ldots, j_N} \mathcal{G}^2_{j_1,j_2,\ldots,j_N}}$ that is $\|\cdot\|_F$ is the regular Frobenious norm for matrices and tensors respectively as defined in Kolda (2006) while $\|\mathcal{G}\|_1 = \sum_{j_1 j_2 \ldots, j_N} \mathcal{G}_{j_1,j_2,\ldots,j_N}$. When normalizing each of the updated $\mathbf{A}^{(n)}$'s should be normalized after being updated, i.e. $\widetilde{\mathbf{A}}_{i_n,d} = \frac{\mathbf{A}_{i_n,d}}{\|\mathbf{A}_d\|_F}$ while the core normalized by $\widetilde{\mathcal{G}} = \frac{\mathcal{G}}{\|\mathcal{G}\|_F}$.

Notice,

$$\begin{aligned} C_{LS}(\mathbf{X}_{(1)}, \mathbf{\Lambda}_{(1)}) = C_{LS}(\mathbf{X}_{(2)}, \mathbf{\Lambda}_{(2)}) = ... = C_{LS}(\mathbf{X}_{(N)}, \mathbf{\Lambda}_{(N)}) &= C_{LS}(vec\mathcal{X}, \mathbf{A}vec(\mathcal{G})) \\ C_{KL}(\mathbf{X}_{(1)}, \mathbf{\Lambda}_{(1)}) = C_{KL}(\mathbf{X}_{(2)}, \mathbf{\Lambda}_{(2)}) = ... = C_{KL}(\mathbf{X}_{(N)}, \mathbf{\Lambda}_{(N)}) &= C_{KL}(vec\mathcal{X}, \mathbf{A}vec(\mathcal{G})). \end{aligned}$$

Consequently each of the updates above minimizes the same cost function. As a result, the convergence of the algorithms for HONMF without sparseness follow straight forward from the convergence

HONMF based on Least squares

```
1. Initialize all $\mathbf{A}^{(n)}$ and the core array $\mathcal{G}$ randomly.
2. For all n do
       $\mathbf{\Lambda}_{(n)} = \mathbf{A}^{(n)}\mathbf{Z}_{(n)}$
       $\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} \bullet \frac{\mathbf{X}_{(n)}\mathbf{Z}_{(n)}^T}{\mathbf{\Lambda}_{(n)}\mathbf{Z}_{(n)}^T}$
3. $\mathcal{L} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 ... \times_N \mathbf{A}^{(N)}$
   $\mathcal{B} = \mathcal{X} \times_1 \mathbf{A}^{(1)^T} \times_2 \mathbf{A}^{(2)^T} \times_3 ... \times_N \mathbf{A}^{(N)^T}$
   $\mathcal{C} = \mathcal{L} \times_1 \mathbf{A}^{(1)^T} \times_2 \mathbf{A}^{(2)^T} \times_3 ... \times_N \mathbf{A}^{(N)^T}$
   $\mathcal{G} \leftarrow \mathcal{G} \bullet \frac{\mathcal{B}}{\mathcal{C}}$
4. Repeat from step 2 until some convergence criterion has been satisfied
```

HONMF based on KL-divergence

```
1. Initialize all $\mathbf{A}^{(n)}$ and the core array $\mathcal{G}$ randomly.
2. For all n do
       $\mathbf{\Lambda}_{(n)} = \mathbf{A}^{(n)}\mathbf{Z}_{(n)}$
       $\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} \bullet \frac{\left(\frac{\mathbf{X}_{(n)}}{\mathbf{\Lambda}_{(n)}}\right)\mathbf{Z}_{(n)}^T}{\mathbf{\Lambda}_{(n)}\mathbf{Z}_{(n)}^T}$
3. $\mathcal{L} = \mathcal{G} \times_1 \mathbf{A}^{(1)} \times_2 \mathbf{A}^{(2)} \times_3 ... \times_N \mathbf{A}^{(N)}$
   $\mathcal{D} = \frac{\mathcal{X}}{\mathcal{L}} \times_1 \mathbf{A}^{(1)^T} \times_2 \mathbf{A}^{(2)^T} \times_3 ... \times_N \mathbf{A}^{(N)^T}$
   $\mathcal{E} = \mathcal{T} \times_1 \mathbf{A}^{(1)^T} \times_2 \mathbf{A}^{(2)^T} \times_3 ... \times_N \mathbf{A}^{(N)^T}$
   $\mathcal{G} \leftarrow \mathcal{G} \bullet \frac{\mathcal{D}}{\mathcal{E}}$
4. Repeat from step 2 until some convergence criterion has been satisfied.
```

Table 2: Algorithms for HONMF based on LS and KL minimization.

	Normalized	Sparse
LS	$\mathbf{A}^{(n)} \leftarrow \widetilde{\mathbf{A}}^{(n)} \bullet \frac{\mathbf{X}_{(n)}\mathbf{Z}^T + \widetilde{\mathbf{A}}^{(n)} diag(\mathbf{1} \cdot \widetilde{\mathbf{\Lambda}}_{(n)}\mathbf{Z}^T \bullet \widetilde{\mathbf{A}}^{(n)})}{\widetilde{\mathbf{\Lambda}}_{(n)}\mathbf{Z}^T + \widetilde{\mathbf{A}}^{(n)} diag(\mathbf{1} \cdot \mathbf{X}_{(n)}\mathbf{Z}^T \bullet \widetilde{\mathbf{A}}^{(n)})}$	$\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} \bullet \frac{\mathbf{X}_{(n)}\mathbf{Z}_{(n)}^T}{\mathbf{\Lambda}_{(n)}\mathbf{Z}_{(n)}^T + \beta \frac{\partial C_{sparse}(\mathbf{A}^{(n)})}{\partial \mathbf{A}^{(n)}}}$
KL	$\mathbf{A}^{(n)} \leftarrow \widetilde{\mathbf{A}}^{(n)} \bullet \frac{\left(\frac{\mathbf{X}_{(n)}}{\widetilde{\mathbf{\Lambda}}_{(n)}}\right)\mathbf{Z}^T + \widetilde{\mathbf{A}}^{(n)} diag(\mathbf{1} \cdot \mathbf{1}\mathbf{Z} \bullet \widetilde{\mathbf{A}}^{(n)})}{\widetilde{\mathbf{\Lambda}}_{(n)}\mathbf{Z}^T + \widetilde{\mathbf{A}}^{(n)} diag\left(\mathbf{1} \cdot \left(\frac{\mathbf{X}_{(n)}}{\mathbf{\Lambda}_{(n)}}\right)\mathbf{Z}^T \bullet \widetilde{\mathbf{A}}^{(n)}\right)}$	$\mathbf{A}^{(n)} \leftarrow \mathbf{A}^{(n)} \bullet \frac{\left(\frac{\mathbf{X}_{(n)}}{\mathbf{\Lambda}_{(n)}}\right)\mathbf{Z}_{(n)}^T}{\mathbf{\Lambda}_{(n)}\mathbf{Z}_{(n)}^T + \beta \frac{\partial C_{sparse}(\mathbf{A}^{(n)})}{\partial \mathbf{A}^{(n)}}}$
LS	$\mathcal{G} \leftarrow \widetilde{\mathcal{G}} \bullet \frac{\mathcal{B} + \widetilde{\mathcal{G}} \lVert \mathcal{C} \bullet \widetilde{\mathcal{G}} \rVert_1}{\mathcal{C} + \widetilde{\mathcal{G}} \lVert \mathcal{B} \bullet \widetilde{\mathcal{G}} \rVert_1}$	$\mathcal{G} \leftarrow \mathcal{G} \bullet \frac{\mathcal{B}}{\mathcal{C} + \beta \frac{\partial C_{sparse}(\mathcal{G})}{\partial \mathcal{G}}}$
KL	$\mathcal{G} \leftarrow \widetilde{\mathcal{G}} \bullet \frac{\mathcal{D} + \widetilde{\mathcal{G}} \lVert \mathcal{E} \bullet \widetilde{\mathcal{G}} \rVert_1}{\mathcal{E} + \widetilde{\mathcal{G}} \lVert \mathcal{D} \bullet \widetilde{\mathcal{G}} \rVert_1}$	$\mathcal{G} \leftarrow \mathcal{G} \bullet \frac{\mathcal{D}}{\mathcal{E} + \beta \frac{\partial C_{sparse}(\mathcal{G})}{\partial \mathcal{G}}}$

Table 3: Updates when normalizing or imposing sparseness on the various modalities

of the regular NMF updates given in Lee and Seung (2000) since the estimation was formulated as a series of regular factor analysis problems minimizing the same cost function. However, the updates including sparseness has not yet been proved for regular NMF. Eggert and Korner (2004) and Mørup and Schmidt (2005) conjectured respectively the sparse LS and KL algorithms to be convergent. Although extensively tested we also never experienced any divergence of the updates above including sparseness.

The presently developed algorithms for HONMF are attractive for several reasons.

- The developed algorithms can yield unique non-negative decomposition by finding the sparsest representation of any combination of modalities.
- Often the tensor data has a lot of elements being zero Sun et al. (2005). Since the NMF updates are very tractable (i.e. no need for matrix inversion or eigenvalue decompositions) the algorithms are easily adapted to consider only the non-zero elements in $\mathcal{X}$.
- Contrary to the HOSVD structure can be forced into the model forming a supervised algorithm. For instance the core or some of the core elements can be fixed to force known interactions into the model.
- Each iteration of the HONMF is $\mathcal{O}(I_1 I_2 \cdot ... \cdot I_N J_1 J_2 \cdot ... \cdot J_N)$ i.e. grows linearly with the product of the size of $\mathcal{X}$ and $\mathcal{G}$ making the cost per iteration relatively cheap compared to existing algorithms for non-negative Tucker decomposition requiring an iterative check of the violation of non-negativity Bro and Andersson (2000); Bro and Jong (1997).
- Contrary to HOSVD the vectors across modalities interact in the estimation process. As a result, omitting vectors will change the existing vectors to account for more of the data.
- The NMF is known to cluster the data rather than projecting the data onto the dimensions accounting for most variance Lee and Seung (1999). This in many situations improves component interpretability.

Needless to say, the algorithms only works when data are non-negative and the components and interactions are considered purely additive. Admittedly, NMF is known to suffer from slow convergence Salakhutdinov et al. (2003). Consequently, the overall speed of the HONMF algorithm is not better than the existing non-unique algorithms.

3. Results

The algorithms were tested on a synthetic data set consisting of 5 images of logical operators mixed through two modalities. The five images forming the third modality along with the mixing matrices of the two first modalities were created such that no rotational ambiguity was present between the factors and the Core. The Core was generated by $uniform(0,1)$ random numbers. The result of the decomposition of the synthetic data can be seen in Figure 1

The algorithms were also tested on a data set containing the inter trial phase coherence (ITPC) obtained from wavelet transformed electroencephalographic (EEG) data. This data set has previously been analyzed using PARAFAC and a detailed description of the data set can be found in Mørup et al. (2006). Briefly stated it consist of 14 subject recorded during a proprioceptive stimuli consisting of a weight change of right hand during odd trials and left hand during even trials giving a total of $14 \cdot 2 = 28$ trials. Consequently, the data has the following form $\mathcal{X}_{ITPCvalue}^{Channel \times Time-Frequency \times Trials}$. The results of a Tucker 3-3-3 model can be seen on Figure 2 while an evaluation of the uniqueness of the decompositions is given in Table 4

Finally, the algorithms were tested on a data set of $\mathcal{X}_{Strength}^{Spectra \times Time \times Batch}$ obtained from a flow injection analysis (FIA) system, see Nørgaard and Ridder (1994); Smilde et al. (1999). The data set

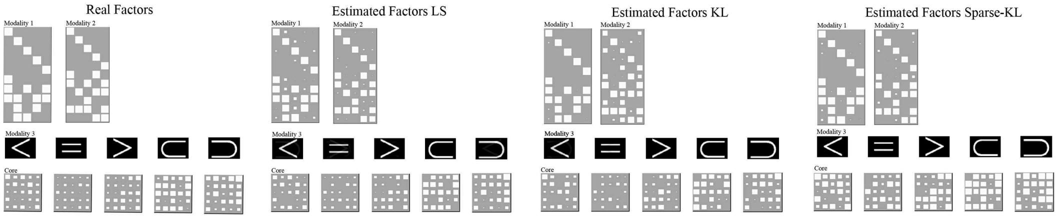

Figure 1: Examples of results obtained when analyzing the synthetic data. Leftmost panel: The true components forming the synthethic data. Middle left panel: Components obtained by the HONMF algorithm based on LS ($\beta = 0$ range of data [0;380]). Middle right panel: Components obtained by the HONMF algorithm based on KL. Rightmost panel: Components obtained by HONMF based on KL with sparseness on the three factor modalities ($\beta = 1$). All decompositions accounts for more than 99.99% of the variance. While the LS algorithm almost perfectly identifies all components the KL algorithm has problems identifying the components of modality 2 however, imposing sparseness the algorithm better identifies the components.

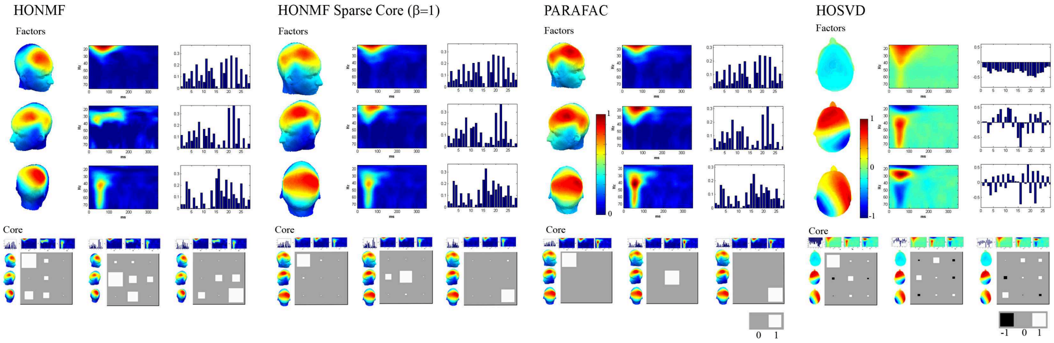

Figure 2: Analysis of the ITPC data of EEG consisting of 14 subjects undergoing weight change of right hand during odd trials and left hand during even trials. Leftmost panel: Example of result obtained when analyzing the data using HONMF. Middle left panel: Result when imposing sparseness on the core ($\beta = 1$, range of data [0;0.4]). Middle right panel: The results obtained from the PARAFAC model corresponding to a fixed Core having ones along the super diagonal. Rightmost panel: The results obtained using HOSVD. Clearly, the HONMF model approaches the PARAFAC model as sparseness is imposed on the Core. While the HONMF accounts for 49.3 % of the variance the sparse HONMF accounts for 49.11 % of the variance whereas the PARAFAC model accounts for 48.9 % of the variance.Finally, the HOSVD accounts for 58.9 % of the variance.

β	0	1	10	100
LS	**Channel :** $F1$: 0.7416±0.2990 (0.3743±0.1352) $F2$: 0.8453±0.1032 (0.3328±0.0897) $F3$: 0.8401±0.0945 (0.3976±0.0814) **Time − Frequency :** $F1$: 0.8906±0.1937 (0.3175±0.0867) $F2$: 0.9317±0.0716 (0.3077±0.0674) $F3$: 0.9313±0.0729 (0.3126±0.0851) **Trials :** $F1$: 0.9268±0.0910 (0.4050±0.1131) $F2$: 0.9538±0.0480 (0.4055±0.1215) $F3$: 0.8661±0.1609 (0.4835±0.0965) **Core :** 0.7420±0.1048 (0.2853±0.1776) **Explained variance :** 0.4912±0.0027	**Channel :** $F1$: 0.9464±0.0471 (0.3427±0.0949) $F2$: 0.9492±0.0541 (0.3932±0.1072) $F3$: 0.9595±0.0381 (0.3660±0.1116) **Time − Frequency :** $F1$: 0.9753±0.0212 (0.3111±0.0378) $F2$: 0.9258±0.1254 (0.3108±0.0378) $F3$: 0.9368±0.1312 (0.3277±0.0484) **Trials :** $F1$: 0.9657±0.0222 (0.3465±0.1702) $F2$: 0.9585±0.1485 (0.4852±0.0674) $F3$: 0.9664±0.1161 (0.4620±0.0674) **Core :** 0.9139±0.0383 (0.2793±0.1244) **Explained variance :** 0.4909±0.0017	**Channel :** $F1$: 1.000±0.000 (0.3813±0.1400) $F2$: 1.000±0.000 (0.3636±0.1631) $F3$: 1.000±0.000 (0.3417±0.1072) **Time − Frequency :** $F1$: 1.000±0.000 (0.2812±0.0380) $F2$: 1.000±0.000 (0.3259±0.0661) $F3$: 1.000±0.000 (0.3329±0.0555) **Trials :** $F1$: 1.000±0.000 (0.4268±0.1402) $F2$: 1.000±0.000 (0.3897±0.1815) $F3$: 1.000±0.000 (0.3947±0.1375) **Core :** 0.6963±0.3535 (0.3473±0.1470) **Explained variance :** 0.3695±0.0000	**Channel :** $F1$: 1.000±0.000 (0.3428±0.1195) $F2$: 1.000±3.4700.000 (0.3657±0.1406) $F3$: 1.000±3.3870.000 (0.3914±0.1305) **Time − Frequency :** $F1$: 1.000±0.000 (0.3327±0.0398) $F2$: 1.000±0.000 (0.3288±0.0417) $F3$: 1.000±0.000 (0.2935±0.0210) **Trials :** $F1$: 1.000±0.000 (0.3681±0.0972) $F2$: 1.000±0.000 (0.4116±0.1434) $F3$: 1.000±0.000 (0.4507±0.1347) **Core :** 0.3561±0.1493 (0.3094±0.1141) **Explained variance :** −0.2600±0.0000

Table 4: Mean correlation between the factors of 10 runs with sparseness imposed on the core array ranging from 0 to 100 here given for LS. In parenthesis are the correlations obtained by random (estimated by permutating the indices of the factors and calculating their correlation). Clearly imposing sparseness improves uniqueness (correlation between each decomposition) however if the sparseness imposed on the core is too strong all factors becomes identical only capturing the mean activity while the core is arbitrary due to the identical factors). The KL algortihm gave similar results.

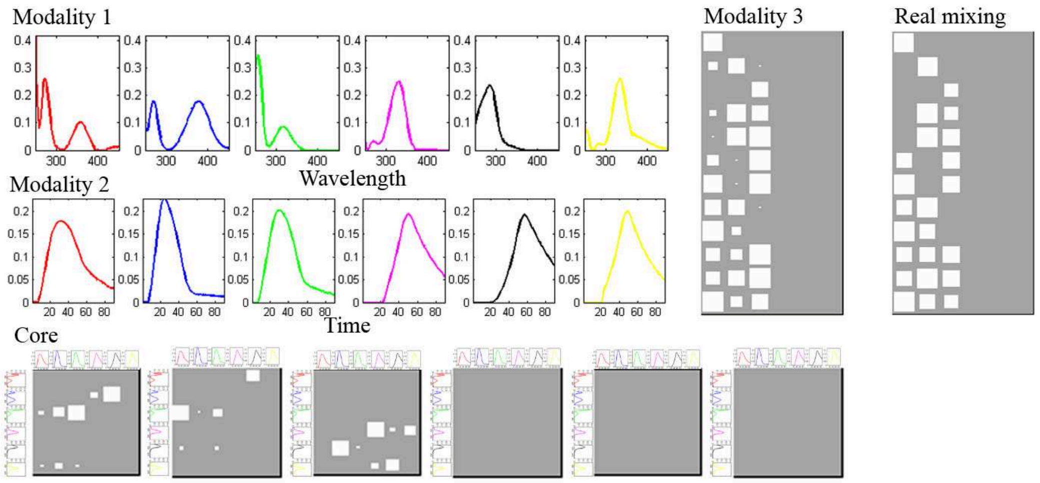

Figure 3: The result obtained analyzing the FIA data by a Tucker 6-6-6 model based on LS with sparsity on the Core and mixing modality ($\beta = 0.5$). 10 decompositions all resulted in very consistent results - mean correlation between the various components of each modality $0.9847 \pm 0.0396 (0.4008 \pm 0.1736)$. Furthermore, the estimated mixing was correlated by $0.9550 \pm 0.0648 (0.3258 \pm 0.1863)$ to the true mixing. The 10 decompositions on average explained 0.9972 ± 0.0007 of the variance.

has been analyzed through various supervised models using among other the prior knowledge of the concentration in each batch Nørgaard and Ridder (1994); Smilde et al. (1999). However, presently we employed a sparse HONMF to see if this algorithm could capture the underlying structure in the data unsupervised. To give an easy interpretable model and improve uniqueness of the batch concentrations found sparseness was imposed on both the core and batch modality ($\beta = 0.5$, range of data [0;0.637]) The results of the sparse Tucker 6-6-6 decomposition is given in Figure 3

4. Discussion

From the HONMF decomposition of the synthetically generated data set it was seen that the KL but especially LS captured well the true components. Although the factors found slightly deviate from the true factors especially on modality 2 imposing sparseness on the factors improved the algorithms ability to correctly identify the components as revealed for the KL results.

In the analysis of the ITPC of EEG data, it was seen in Table 4 that each unconstrained HONMF decomposition only was correlated by about 70-90%. However, when imposing sparseness on the core a more unique decomposition was achieved hence a correlation well above 90% between the components of the Factors and Core of the 10 decompositions while only slightly affecting the explained variance. However, by increasing sparseness too much only the mean activity was captured in all the components. Consequently the factors were all perfectly correlated to each other while the core could be arbitrarily chosen as long as the sum of the core elements remained the same. It was further seen that imposing sparseness on the data made the decomposition resemble the corresponding PARAFAC decomposition. This indicates that the PARAFAC decomposition rather than the full Tucker model is a reasonable model to the data. Consequently, the Tucker model with sparsity imposed on the core can indicate wether a PARAFAC or a Tucker model is the most reasonable model to the data at hand. Although the HOSVD accounts for more variance since cancellation of factors are allowed, the decomposition is difficult to interpret. While the last factor in the trial modality clearly differentiates between left and right side stimulation and the second and third scalp component differentiates between frontal parietal and left right activity the interpretation of the interactions between these components are difficult to resolve from the complex pattern of interaction given by the core. Consequently, although the HONMF model accounts for less variance it is easier to interpret since it clearly gives a more part based decomposition.

Finally, the analysis of the FIA data gave a very consistent decomposition. By imposing sparseness on the core and Batch modality the model captured well the true concentrations in the batch as well as giving a sparse Core improving the interpretatibility. Consequently, imposing sparseness could turn of unnecessary excess factors as well as capturing the true structure in the data unsupervised rather than resorting to supervised approaches as previously done Nørgaard and Ridder (1994); Smilde et al. (1999) presently capturing well the true concentrations in the batches.

Admittedly, the present HONMF has two drawback. The choices of β is not obvious while to some extent impacting the decompositions found. Furthermore, NMF and therefore also HONMF is known to suffer from slow convergence Salakhutdinov et al. (2003). Presently, the algorithmss were accelerated as proposed for NMF by Salakhutdinov et al. (2003).

5. Conclusion

It is our strong belief that the HONMF algorithms proposed will be useful in the analysis of a variety of higher order data. Presently, the HONMF gave a more easy interpretable decomposition than the HOSVD. Furthermore, imposing constraints of sparseness significantly improved the uniqueness of the decomposition as well as being a tool for model selection.

References

Claus A. Andersson and Rasmus Bro. Improving the speed of multi-way algorithms: Part i. tucker3. *Chemometrics and Intelligent Laboratory Systems*, 42:93–103, 1998.

Rasmus Bro and Claus A. Andersson. The n-way toolbox for matlab. *Chemometrics and Intelligent Laboratory Systems*, 52:1–4, 2000.

Rasmus Bro Bro and Sijmen De Jong. A fast non-negativity-constrained least squares algorithm. *Journal of Chemometrics*, 11(5):393–401, 1997.

J. D. Carroll and J. J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35:283–319, 1970.

David Donoho and Victoria Stodden. When does non-negative matrix factorization give a correct decomposition into parts? *NIPS*, 2003.

J. Eggert and E. Korner. Sparse coding and nmf. In *Neural Networks*, volume 4, pages 2529–2533, 2004.

D. FitzGerald, M. Cranitch, and E. Coyle. Non-negative tensor factorisation for sound source separation. In *proceedings of Irish Signals and Systems Conference*, pages 8–12, 2005.

S. P. Gurden, J. A. Westerhuis, S. Bijlsma, and A. K. Smilde. Modelling of spectroscopic batch process data using grey models to incorporate external information. *Journal of Chemometrics*, 15:101–121, 2001.

R. A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970.

P.O. Hoyer. Non-negative sparse coding. *Neural Networks for Signal Processing, 2002. Proceedings of the 2002 12th IEEE Workshop on*, pages 557–565, 2002.

P.O. Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 2004.

Kui Jia and Shaogang Gong. Multi-modal tensor face for simultaneous super-resolution and recognition. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*, pages 1683–1690, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2334-X-02.

Tamara G. Kolda. Multilinear operators for higher-order decompositions. Technical Report SAND2006-2081, tr:sandreport, April 2006.

J.B Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra Appl.*, 18:95–138, 1977.

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. Multilinear singular value decomposition. *SIAM J. MATRIX ANAL. APPL.*, 21(4):1253Ű1278, 2000.

Daniel D Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

D.D. Lee and H.S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–91, 1999. ISSN 00280836.

M. Mørup, L. K. Hansen, J. Parnas, and S. M. Arnfred. Decomposing the time-frequency representation of EEG using non-negative matrix and multi-way factorization. Technical report, 2006. URL `http://www2.imm.dtu.dk/pubdb/p.php?4144`.

M. Mørup and M. N. Schmidt. Nonnegative matrix factor 2-D deconvolution (nmf2d) and sparse nmf2d. Technical report, Institute for Mathematical Modelling, Technical University of Denmark, 2005.

Takashi Murakami and Pieter M. Kroonenberg. Three-mode models and individual differences in semantic differential data. *Multivariate Behavioral Research*, 38(2):247–283, 2003.

L Nørgaard and C. Ridder. Rank annihilation factor analysis applied to flow injection analysis with photodiode-array detection. *Chemometrics and Intelligent Laboratory Systems*, 23(1):107–114, 1994.

R. Parry, Mitchell and Irfan Essa. Estimating the spatial position of spectral components in audio. In *proceedings ICA2006*, pages 666– 673, 2006.

Ruslan Salakhutdinov, Sam Roweis, and Zoubin Ghahramani. On the convergence of bound optimization algorithms. In *Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence (UAI-03)*, pages 509–516, San Francisco, CA, 2003. Morgan Kaufmann Publishers.

Berkant Savas and Lars Eldén. Handwritten digit classification using higher order singular value decomposition. *Pattern Recognition*, submitted.

Nicholas D. Sidiropoulos and Rasmus Bro. On the uniqueness of multilinear decomposition of n-way arrays. *Journal of Chemometrics*, 14:229–239, 2000.

Age Smilde, Rasmus Bro, and Paul Geladi. *Multi-way Analysis: Applications in the Chemical Sciences.* Wiley, August 2004. ISBN ISBN: 0-471-98691-7.

Age K. Smilde Smilde, Roma Tauller, Javier Saurina, and Rasmus Bro. Calibration methods for complex second-order data. *Analytica Chimica Acta*, 398:237–251, 1999.

Jian-Tao Sun, Hua-Jun Zeng, Huan Liu, Yuchang Lu, and Zheng Chen. Cubesvd: a novel approach to personalized web search. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 382–390, New York, NY, USA, 2005. ACM Press. ISBN 1-59593-046-9.

L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

M. A. O. Vasilescu and Demetri Terzopoulos. Multilinear analysis of image ensembles: Tensorfaces. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part I*, pages 447–460, London, UK, 2002. Springer-Verlag. ISBN 3-540-43745-2.

Hongcheng Wang and Narendra Ahuja. Facial expression decomposition. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 958, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1950-4.

Max Welling and Markus Weber. Positive tensor factorization. *Pattern Recogn. Lett.*, 22(12):1255–1261, 2001. ISSN 0167-8655.