First and second-order derivatives for CP and INDSCAL

Tendeiro, Jorge N.; Bennani-Dosse, M.; Ten Berge, Jos M. F.

Published in:
Chemometrics and Intelligent Laboratory Systems

DOI:
10.1016/j.chemolab.2010.05.013

Publication date:
2011

Chemometrics and Intelligent Laboratory Systems 106 (2011) 27–36

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

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

First and second-order derivatives for CP and INDSCAL[☆]

Jorge Tendeiro [a,*], Mohammed Bennani Dosse [b], Jos M.F. ten Berge [a]

[a] University of Groningen, The Netherlands
[b] University of Rennes 2, France

ARTICLE INFO

Article history:
Received 21 January 2010
Received in revised form 6 May 2010
Accepted 22 May 2010
Available online 27 May 2010

Keywords:
CANDECOMP
PARAFAC
INDSCAL
Jacobian matrix
Hessian matrix

ABSTRACT

In this paper we provide the means to analyse the second-order differential structure of optimization functions concerning CANDECOMP/PARAFAC and INDSCAL. Closed-form formulas are given under two types of constraint: unit-length columns or orthonormality of two of the three component matrices. Some numerical problems that might occur during the computation of the Jacobian and Hessian matrices are addressed. The use of these matrices is illustrated in three applications.

1. Introduction

Carroll and Chang [3] and Harshman [5] independently presented two identical methods to analyse three-way arrays. The former is CANDECOMP and the latter is PARAFAC; the method is now well known as CANDECOMP/PARAFAC or simply CP. Given a $p\times q\times m$ array $\underline{\mathbf{M}}$ with frontal $p\times q$ slices $\mathbf{M}_i$ ($i=1,\ldots,m$), CP aims at finding the component matrices $\mathbf{X}$ ($p\times r$), $\mathbf{Y}$ ($q\times r$) and $\mathbf{D}$ ($m\times r$) that minimize the function

$$f(\mathbf{X},\mathbf{Y},\mathbf{D}) = \sum_{i=1}^{m} ||\mathbf{M}_i - \mathbf{X}\mathbf{D}_i\mathbf{Y}'||^2, \tag{1.1}$$

where $\mathbf{D}_i$ is the diagonal matrix holding row i of $\mathbf{D}$ in the diagonal. Minimizing f can be done in various ways. Carroll and Chang [3] and Harshman [5] proposed an alternating least-squares method that has become known as the CP decomposition. However, other approaches have also been proposed. For instance, Paatero [12] has offered a conjugate gradient algorithm.

The CP decomposition starts by initializing $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{D}$, and alternately updates each component matrix while the others remain constant. Iterations are terminated when the relative improvement in f is smaller than a predefined threshold. It is not guaranteed that CP converges; if it does converge, it is not guaranteed that the global minimum is reached. To increase the chances of finding the seeked minimum it is desirable to start CP with several initialization values.

For the special case when the array has symmetric frontal slices, say $\mathbf{S}_1, \ldots, \mathbf{S}_m$ of order $p\times p$, Carroll and Chang [3] proposed INDSCAL, which minimizes the function

$$g(\mathbf{X},\mathbf{D}) = \sum_{i=1}^{m} ||\mathbf{S}_i - \mathbf{X}\mathbf{D}_i\mathbf{X}'||^2. \tag{1.2}$$

Since minimizing g directly seems difficult, Carroll and Chang [3] suggested minimizing f instead. They conjectured that, after convergence, $\mathbf{X}$ and $\mathbf{Y}$ will be equal or, at least, columnwise proportional (i.e., the columns of $\mathbf{Y}$ can be rescaled to match the columns of $\mathbf{X}$, while the columns of $\mathbf{D}$ absorb the inverse scaling). Such matrices will be referred to as being *equivalent*.

Carroll and Chang's conjecture seems to be valid in practical applications. However, counter-examples have already been constructed. Ten Berge and Kiers [16] proved that equivalence may be violated at global minima of f if the slices $\mathbf{S}_i$ are indefinite. When the slices are non-negative definite and $r=1$ then equivalence can be violated only at stationary points that do not correspond to global minima. Ten Berge and Kiers [16] conjectured that such stationary points would be local minima. However, Bennani Dosse and Ten Berge [1] proved that such stationary points must be saddle points. This was achieved by analysing the first and second-order derivatives of a specific optimization function derived from the loss function of CP. Notice that the result by Bennani Dosse and Ten Berge [1] concerns the case where $r=1$ component is used. The conjecture of Carroll and Chang seems to be an open issue when $r>1$ components are used. In this paper, we aimed at finding a second-order sufficient condition that classifies CP

☆ This research has been supported by POPH/FSE (Programa Operacional Potencial Humano/Fundo Social Europeu).

* Corresponding author. Heijmans Institute, University of Groningen, Grote Kruisstraat 2/1, 9712 TS Groningen, The Netherlands. Tel.: +31 503636953.
E-mail address: j.n.tendeiro@rug.nl (J. Tendeiro).

0169-7439/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.chemolab.2010.05.013

decompositions with $r \geq 1$ components as (local) optima or saddle points, see Section 3. With this tool at hand we conducted a simulation study which sheds some light on the equivalence problem, see Section 7. A similar second-order sufficient condition, this time applied to INDSCAL, was also derived, see Section 4.

We extended the research of Bennani Dosse and Ten Berge [1] to the case where $r>1$ components are extracted. First and second-order derivatives for optimization functions which follow directly from the loss functions of CP and INDSCAL were derived. The reason why the loss functions (1.1), (1.2) were not used is that it is possible to express **D** as a function of **X** and **Y** (for f) or as a function of **X** (for g) at stationary points, see Sections 3 and 4. This allows simplifying the optimization problem: the task of minimizing f and g will be replaced by maximizing simpler (= with less variables) optimization functions. Moreover, this is a necessary step if one is to use differential second-order conditions. The main reason is that the Hessian matrix is singular if no elimination of variables is performed, thus drawing inferences about minima and maxima is unwarranted.

Another source of freedom that needs to be controlled is directly related to the fact that the CP model is overparametrized. Namely, given diagonal matrices $\mathbf{\Lambda}_1$, $\mathbf{\Lambda}_2$, $\mathbf{\Lambda}_3$ such that $\mathbf{\Lambda}_1\mathbf{\Lambda}_2\mathbf{\Lambda}_3 = \mathbf{I}_r$, both (**X**, **Y**, **D**) and ($\mathbf{X\Lambda}_1$, $\mathbf{Y\Lambda}_2$, $\mathbf{D\Lambda}_3$) represent the same solution. This scaling indeterminacy is considered to be trivial in CP. Nevertheless it does pose a problem when optimizing f using differential tools since one has $f(\mathbf{X}, \mathbf{Y}, \mathbf{D}) = f(\mathbf{X\Lambda}_1, \mathbf{Y\Lambda}_2, \mathbf{D\Lambda}_3)$, ie, for each (**X**, **Y**, **D**) in the domain of f there is an infinity of points which are mapped onto $f(\mathbf{X}, \mathbf{Y}, \mathbf{D})$. This has the effect of making the second-order sufficient conditions useless, since in these conditions the Hessian matrix will invariably fail to be non-singular. Therefore, determining the nature of stationary points of f via its second-order differential structure becomes unfeasible under the current setting. Notice that a similar problem applies also to INDSCAL and its associate function g, since an INDSCAL solution is also characterized by scaling indeterminacy. Also, the new optimization functions that will be derived from f and g suffer from the same problem. Since the analysis of second-order structures is one of the goals in this paper, something had to be done to overcome this issue. Constraining the domain of the optimization functions is a possible solution to the problem discussed in the previous paragraph. We settled for two types of constraint: **X** and **Y** constrained to hold unit length columns (Case I), and **X** and **Y** constrained by orthonormality (Case II). The first constraint is a so-called identification constraint; it involves no loss of fit. The second constraint is active, thus a loss of fit is due to happen when compared to the unconstrained situation. Both constraints proved to eliminate the problem of singularity of the Hessian matrix in the vast majority of the cases. Some exceptions were found, as will be discussed in latter sections.

The utility of the second-order conditions that we present in this paper extends beyond the study of the equivalence problem. In fact, minimizing f is not a straightforward optimization problem. First of all, there is usually no closed-form solution. Moreover, a solution might not even *exist*. For example, the $2 \times 2 \times 2$ symmetric slice array analysed by Ten Berge, Kiers and De Leeuw [17] showed that the loss function (1.1) has an infimum which is not a minimum. More recently, Stegeman [13] showed that (1.1) does not have a minimum when $p \times p \times 2$ arrays of rank $p+1$ or higher are decomposed into p rank-1 arrays and a residual array (see also Stegeman [14] for a follow-up). Other problems that might affect the search and quality of an optimal solution for f are: preprocessing the data, the number of components to retain, the choice of the initialization values for the algorithm, slow convergence of the algorithm, existence, uniqueness or "illness" of the solution. The fact that CP does not always converge, or that it might converge to non-optimal points, raises questions concerning the nature of the limiting points of a CP sequence. Similar questions apply to INDSCAL solutions and target function (1.2). These observations reinforce the benefit of having available a tool like the one we propose in this paper. Since our tool allows to better characterize a CP or INDSCAL solution, we have a better insight into the nature of such solution. Specifically, if a solution proves to be a saddle point, then one is sure that it cannot correspond to the seeked global minimum. Therefore a new run of the algorithm is required, possibly with different (random) starting values.

There has been some research in the past concerning the study of the differential structure of the optimization function for CP. Paatero [11,12] has developed formulas for the Jacobian and Hessian matrices for loss function (1.1). However, our approach differs from Paatero's in two ways. Firstly, Paatero does not perform variable elimination. Secondly, Paatero derives a numerical approximation to the Hessian matrix, whereas we propose in the present paper Hessian matrices in closed form.

In this paper we will use matrix differential calculus; definitions and useful differential formulas are to be found in Section 2 and Appendix A. All formulas and necessary derivations for the Jacobian and Hessian matrices are the core of Sections 3 and 4.

The benefit of analysing second-order differential structures is first illustrated in Section 5, where we revisit the data analysed by Ten Berge, Kiers and De Leeuw [17]. It is shown that, for this data, saddle points occur very often when the ALS algorithm is initialized by randomly generating orthonormal component matrices **A**, **B**, **C**.

In Sections 6–8 we describe three simulation studies that were carried out. In the first study we used an algorithm for INDSCAL with orthogonality constraints (Ten Berge, Knol and Kiers [18]) to fit $3 \times 3 \times 3$ arrays with positive definite slices and also $3 \times 3 \times 3$ arrays with indefinite slices. We wished to detect non-optimal solutions, and to see whether they corresponded to saddle points or not. The goal was to further clarify the characterization of the solutions found in the simulation study of Ten Berge et al. [18]. In the second study we tried to see how a result of Bennani Dosse and Ten Berge ([1], pg. 306) extends to situations with $r>1$ components. In order to do this, we generated random arrays with positive definite slices and also with indefinite slices and then computed CP solutions with more than one component. We analysed the second-order information for each solution. The first goal was to check whether non-equivalence could occur at all. In case it would occur, we were interested in verifying whether such solutions correspond to saddle points (as it is proven to happen when $r=1$) or to local optima. The contrast between arrays with positive definite slices and indefinite slices was considered. In both situations we analysed features such as degeneracy and occurrence of different fit values. In the third study we expanded this type of analysis to CP solutions of non-symmetric slice arrays, for 29 different scenarios.

We finish the paper with a Discussion section, where some considerations about numerical stability of our second-order conditions are to be found. It is argued that differentiation might not be possible for degenerate solutions of CP or INDSCAL, since degenerate solutions correspond to points where the optimal functions are nearly non-differentiable. Some caution is therefore needed when analysing cases of this kind.

2. Derivatives of matrix functions with respect to matrix variables

2.1. Notation

Scalars will be denoted by lower case italic font (a, x, λ), vectors by lower case bold-face font ($\mathbf{a}$, $\mathbf{x}$, $\boldsymbol{\lambda}$), matrices by upper case bold-face font (**A**, **X**, $\mathbf{\Lambda}$), and arrays by underlined upper case bold-face font ($\underline{\mathbf{A}}$, $\underline{\mathbf{X}}$, $\underline{\mathbf{\Lambda}}$). Given matrix **X**, $\mathbf{x}_i$ denotes the i-th column of **X**. The only exceptions to this rule appear in definitions (3.2) and (4.2).

For given matrices **A** and **B**, $\mathbf{A}'$ is the transpose of **A**; tr(**A**) is the trace of **A**; vec(**A**) reshapes **A** into a column vector by stacking the columns in sequence, one below the other; $\mathbf{A} \otimes \mathbf{B}$, $\mathbf{A} * \mathbf{B}$ and $\mathbf{A} \odot \mathbf{B}$ denote the Kronecker, Hadamard and Khatri-Rao products of **A** and **B**, respectively; and $\mathrm{diag}_\mathrm{V}(\mathbf{A})$ is the column vector holding the diagonal of **A**. Given a vector **d**, $\mathrm{diag}_\mathrm{M}(\mathbf{d})$ denotes the diagonal matrix whose diagonal is equal to **d**. $\mathbf{I}_m$ is the identity matrix of order m; $\mathbf{0}_{mn}$ is the zero matrix of order $m \times n$; $\mathbf{C}_{mn}$ is the $mn \times mn$ commutation matrix, i.e., $\mathbf{C}_{mn}\mathrm{vec}(\mathbf{A}) = \mathrm{vec}(\mathbf{A}')$; $\mathbf{T}_n$ is the $n^2 \times n$ matrix with unit entries in position $((i-1)n+i,\ i)$ for $i=1,\ \ldots,\ n$ and zeroes elsewhere, and $\mathbf{E}_n=\mathbf{I}_{n^2}-\mathbf{T}_n\mathbf{T}_n'$. For example, for $n=3$

$$\mathbf{T}_3=\left[\begin{array}{ccc}1&0&0\\0&0&0\\0&0&0\\\hline 0&0&0\\0&1&0\\0&0&0\\\hline 0&0&0\\0&0&0\\0&0&1\end{array}\right],\quad \mathbf{E}_3=\left[\begin{array}{ccc|ccc|ccc}0&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0\\\hline 0&0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&1&0&0&0\\\hline 0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0&0\end{array}\right]. \tag{2.1}$$

In Appendices B and C there can be found several matrix functions ($\mathbf{F}_1$ to $\mathbf{F}_8$, and $\mathbf{G}_1$ to $\mathbf{G}_{16}$), as well as some associated derivatives. These functions will be used throughout the derivations in Sections 3 and 4.

2.2. Differentiation of functions with respect to matrix variables

The Jacobian matrix of f: $\mathbb{R}^n\rightarrow\mathbb{R}^m$ is the $m\times n$ matrix of partial derivatives whose entry $(i,\ j)$ is $\frac{\partial f_i(\mathbf{x})}{\partial x_j}$, for $\mathbf{x}\in\mathbb{R}^n$. This notion of Jacobian matrix can be extended to matrix functions with matrix variables: the Jacobian matrix of function $\mathbf{A}$: $\mathbb{R}^{p\times r}\rightarrow\mathbb{R}^{m\times n}$ is the $mn\times pr$ matrix given by $\frac{\partial\mathbf{A}}{\partial\mathbf{X}}=\frac{\partial\mathrm{vec}(\mathbf{A})}{\partial\mathrm{vec}(\mathbf{X})}$.

Given a scalar function f: $\mathbb{R}^n\rightarrow\mathbb{R}$, the associated Hessian matrix $\frac{\partial^2 f}{\partial\mathbf{X}^2}$ is the $n\times n$ matrix whose entry $(i,\ j)$ is $\frac{\partial^2 f(\mathbf{x})}{\partial x_j\partial x_i}$, for $\mathbf{x}\in\mathbb{R}^n$. The concept of Hessian matrix can be extended to scalar functions with matrix variables as follows: the Hessian matrix of function f: $\mathbb{R}^{p\times r}\rightarrow\mathbb{R}$ is the $pr\times pr$ matrix given by $\frac{\partial^2 f}{\partial\mathbf{X}^2}=\frac{\partial^2 f}{\partial\mathrm{vec}(\mathbf{X})^2}$.

This is how the partial derivatives will be arranged in the sequel. For example, the Jacobian matrix of a scalar function will be a row vector. Also, all differential formulae that will be introduced are adapted to this definition. Notice that there exist authors who choose to display the partial derivatives of the Jacobian and Hessian matrices in a different way than the one done in the present paper. See, for example, Magnus and Neudecker ([10], Chapter 9) for a discussion on this subject. Therefore, some caution is needed before going into the derivations of the next sections.

2.3. Matrix differentiation formulas

In Table A1 (see Appendix A) we summarize the most important formulas of matrix differentiation that are of use in this paper. In Tables B1 and C1 (see Appendices B and C) we define functions $\mathbf{F}_1-\mathbf{F}_8$ and $\mathbf{G}_1-\mathbf{G}_{16}$, for which we present the relevant partial derivatives. These functions appear useful during the differentiation process, as they simplify the presentation of our results. In some functions we add the superscript $^{(i)}$ to denote the dependency of the function on the value of $i=1,\ \ldots,\ m$.

3. Optimization of CP

The loss function of CP (1.1) can be written as

$$f(\mathbf{X},\mathbf{Y},\mathbf{D})=\sum_{i=1}^{m}\left(||\mathbf{M}_i||^2+\mathrm{tr}\left(\mathbf{Y}\mathbf{D}_i\mathbf{X}'\mathbf{X}\mathbf{D}_i\mathbf{Y}'\right)-2\mathrm{tr}\left(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i\right)\right). \tag{3.1}$$

At stationary points we have

$$\mathbf{d}_i=\text{transposed row } i \text{ of } \mathbf{D}=(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\mathrm{diag}_V(\mathbf{X}'\mathbf{M}_i\mathbf{Y}) \tag{3.2}$$

and

$$\mathrm{tr}(\mathbf{Y}\mathbf{D}_i\mathbf{X}'\mathbf{X}\mathbf{D}_i\mathbf{Y}')=\mathrm{tr}(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i). \tag{3.3}$$

Formula (3.2) can be directly derived from the equation $\frac{\partial f}{\partial\mathbf{D}_i}=0$; it allows to express $\mathbf{D}$ in terms of $\mathbf{X}$ and $\mathbf{Y}$. Equality (3.3) can be seen as follows: define $\mathbf{e}_i=\mathrm{diag}_V(\mathbf{Y}'\mathbf{M}'_i\mathbf{X})$, and verify that $\mathrm{tr}(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i)=\mathbf{e}_i'\mathbf{d}_i$, $\mathrm{tr}(\mathbf{X}'\mathbf{X}\mathbf{D}_i\mathbf{Y}'\mathbf{Y}\mathbf{D}_i)=\mathbf{d}_i'(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})\mathbf{d}_i=\mathbf{d}_i'\mathbf{e}_i$. Thus, to optimize the loss function of CP we can work with function

$$f(\mathbf{X},\mathbf{Y})=\sum_{i=1}^{m}\left(||\mathbf{M}_i||^2-\mathrm{tr}\left(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i\right)\right), \tag{3.4}$$

for $\mathbf{D}$ defined as in Eq. (3.2). Minimizing Eq. (3.4) is equivalent to maximizing

$$L^{\mathrm{CP}}(\mathbf{X},\mathbf{Y})=\sum_{i=1}^{m}\mathrm{tr}\left(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i\right), \tag{3.5}$$

for $\mathbf{D}$ defined by Eq. (3.2).

We wish to describe a sufficient condition for a stationary point of $L^{\mathrm{CP}}(\mathbf{X},\mathbf{Y})$ to be a (local) maximum. In order to do this, we will derive the Jacobian and Hessian matrices for $L^{\mathrm{CP}}(\mathbf{X},\mathbf{Y})$ in two different scenarios: $\mathbf{X}$,$\mathbf{Y}$ constrained to hold columns of unit length (Case I), and $\mathbf{X}$,$\mathbf{Y}$ constrained by orthonormality (Case II). The constrained situations will be dealt with by introducing Lagrange multipliers:

$$L_c^{\mathrm{CP}}(\mathbf{X},\mathbf{Y})=\sum_{i=1}^{m}\mathrm{tr}(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i)-\mathrm{tr}(\mathbf{\Lambda}(\mathbf{X}'\mathbf{X}-\mathbf{I}_r))-\mathrm{tr}(\mathbf{\Delta}(\mathbf{Y}'\mathbf{Y}-\mathbf{I}_r)). \tag{3.6}$$

In the case that $\mathbf{X}$ and $\mathbf{Y}$ are constrained to have unit length columns we have that $\mathbf{\Lambda}=\mathrm{Diag}([\lambda_i])$ and $\mathbf{\Delta}=\mathrm{Diag}([\delta_i])$ are diagonal $r\times r$ matrices holding Lagrange multipliers, and $\mathbf{D}$ is given by Eq. (3.2) with the diagonal of $\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y}$ filled with 1's. If $\mathbf{X}$ and $\mathbf{Y}$ are constrained by orthonormality then $\mathbf{\Lambda}=[\lambda_{ij}]$ and $\mathbf{\Delta}=[\delta_{ij}]$ are symmetric $r\times r$ matrices holding Lagrange multipliers and $\mathbf{D}$ is given by Eq. (3.2) with $\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y}=\mathbf{I}_r$.

3.1. Derivation of the Jacobian of L_c^{CP}

Define $\gamma_i=\mathrm{tr}(\mathbf{Y}'\mathbf{M}'_i\mathbf{X}\mathbf{D}_i)$; we have

$$\frac{\partial\gamma_i}{\partial\mathbf{X}}=\mathrm{vec}(\mathbf{I}_r)'(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)\left(\left(\mathbf{D}_i\otimes\mathbf{I}_p\right)+(\mathbf{I}_r\otimes\mathbf{X})\mathbf{T}_r\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\right). \tag{3.7}$$

In Case I the partial derivative of $\mathbf{d}_i$ with respect to $\mathbf{X}$ is

$$\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}=(\mathrm{diag}_V(\mathbf{X}'\mathbf{M}_i\mathbf{Y})'\otimes\mathbf{I}_r)\frac{\partial(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}}{\partial\mathbf{X}}+(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i); \tag{3.8}$$

see Appendix B for the derivation of $\frac{\partial(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}}{\partial\mathbf{X}}$. In Case II we have that

$$\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}=\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i). \tag{3.9}$$

Analogously,

$$\frac{\partial\gamma_i}{\partial\mathbf{Y}}=\mathrm{vec}(\mathbf{I}_r)'(\mathbf{I}_r\otimes\mathbf{X}'\mathbf{M}_i)\left(\left(\mathbf{D}_i\otimes\mathbf{I}_q\right)+(\mathbf{I}_r\otimes\mathbf{Y})\mathbf{T}_r\frac{\partial\mathbf{d}_i}{\partial\mathbf{Y}}\right). \tag{3.10}$$

In Case I the partial derivative of $\mathbf{d}_i$ with respect to $\mathbf{Y}$ is

$$\frac{\partial \mathbf{d}_i}{\partial \mathbf{Y}} = (\mathrm{diag_V}(\mathbf{X}'\mathbf{M}_i\mathbf{Y})'\otimes\mathbf{I}_r)\frac{\partial(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}}{\partial\mathbf{Y}} + (\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{X}'\mathbf{M}_i); \tag{3.11}$$

see Appendix B for the derivation of $\frac{\partial(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}}{\partial\mathbf{Y}}$. In Case II we have that

$$\frac{\partial \mathbf{d}_i}{\partial \mathbf{Y}} = \mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{X}'\mathbf{M}_i). \tag{3.12}$$

The Jacobian of L_c^{CP} is the $1\times(p+q)r$ row vector

$$\mathrm{Jac}\left(L_c^{\mathrm{CP}}\right) = \left[\sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{X}}\ \ \sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{Y}}\right] - 2\left[\mathrm{vec}(\mathbf{X})'(\mathbf{\Lambda}\otimes\mathbf{I}_p)\ \ \mathrm{vec}(\mathbf{Y})'(\mathbf{\Delta}\otimes\mathbf{I}_q)\right]. \tag{3.13}$$

3.2. Derivation of the Lagrange multipliers

To find expressions for the Lagrange multipliers as functions of $\mathbf{X}$ and $\mathbf{Y}$ we need to solve $\frac{\partial L_c^{\mathrm{CP}}}{\partial\mathbf{X}}=0$, $\frac{\partial L_c^{\mathrm{CP}}}{\partial\mathbf{Y}}=0$. We shall solve the first equation; the process is the same for the second one. Equation $\frac{\partial L_c^{\mathrm{CP}}}{\partial\mathbf{X}}=0$ is equivalent to $\sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{X}} = 2\mathrm{vec}(\mathbf{X})'(\mathbf{\Lambda}\otimes\mathbf{I}_p)$. This implies that

$$\sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{x}_k} = 2\sum_{j=1}^r\lambda_{jk}\mathbf{x}'_j, \tag{3.14}$$

for $k=1,\ldots, r$. In case I Eq. (3.14) becomes $\sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{x}_k}=2\lambda_{kk}\mathbf{x}'_k$, which implies that

$$\lambda_{kk} = \frac{1}{2}\sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{x}_k}\mathbf{x}_k. \tag{3.15}$$

In case II we have

$$\lambda_{jk} = \frac{1}{2}\sum_{i=1}^m\frac{\partial\gamma_i}{\partial\mathbf{x}_k}\mathbf{x}_j, \tag{3.16}$$

for $j=1,\ldots,r$.

3.3. Derivation of the Hessian of L_c^{CP}

Next we derive the second-order derivatives. Define the following constant matrices with respect to $\mathbf{X}$: $\mathbf{J}_1=\mathrm{vec}(\mathbf{I}_r)'(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)$; $\mathbf{J}_2=-\mathrm{diag_M}(\mathrm{vec}(\mathbf{Y}'\mathbf{Y}))\mathbf{E}_r(\mathbf{I}_{r^2}+\mathbf{C}_{rr})$; $\mathbf{J}_3=\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)$. It can be seen that

$$\begin{aligned}\frac{\partial^2\gamma_i}{\partial\mathbf{X}^2} = &\left(\mathbf{I}_{pr}\otimes\mathbf{J}_1\right)\Big[\left(\mathbf{I}_r\otimes\mathbf{C}_{pr}\otimes\mathbf{I}_p\right)\left(\mathbf{I}_{r^2}\otimes\mathrm{vec}\left(\mathbf{I}_p\right)\right)\mathbf{T}_r\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\\ &+\left(\left(\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\right)'\otimes\mathbf{I}_{pr}\right)\left(\mathbf{T}'_r\otimes\mathbf{I}_{pr}\right)\left(\mathbf{I}_r\otimes\mathbf{C}_{rr}\otimes\mathbf{I}_p\right)\left(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{pr}\right)\\ &+\left(\mathbf{I}_{pr}\otimes(\mathbf{I}_r\otimes\mathbf{X})\mathbf{T}_r\right)\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{X}^2}\Big],\end{aligned} \tag{3.17}$$

with $\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}$ given by Eq. (3.8) (in Case I) or Eq. (3.9) (in Case II). The term $\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{X}^2}$ is $\mathbf{0}_{pr^2,pr}$ in Case II; to derive $\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{X}^2}$ in Case I we start by rewriting Eq. (3.8):

$$\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}} = \mathbf{F}_4^{(i)}\mathbf{J}_2(\mathbf{I}_r\otimes\mathbf{X}') + (\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{J}_3, \tag{3.18}$$

where $\mathbf{F}_4^{(i)}=(\mathrm{diag_V}(\mathbf{X}'\mathbf{M}_i\mathbf{Y})'\otimes\mathbf{I}_r)((\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\otimes(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1})$, see Appendix B. We can now write

$$\begin{aligned}\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{X}^2} = &((\mathbf{I}_r\otimes\mathbf{X})\mathbf{J}'_2\otimes\mathbf{I}_r)\frac{\partial\mathbf{F}_4^{(i)}}{\partial\mathbf{X}} + \left(\mathbf{I}_{pr}\otimes\mathbf{F}_4^{(i)}\right)\left(\mathbf{I}_{pr}\otimes\mathbf{J}_2\right)\\ &\times\left(\mathbf{I}_r\otimes\mathbf{C}_{pr}\otimes\mathbf{I}_r\right)\left(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{pr}\right)\mathbf{C}_{pr} + (\mathbf{J}'_3\otimes\mathbf{I}_r)\frac{\partial(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}}{\partial\mathbf{X}}.\end{aligned} \tag{3.19}$$

We proceed in a similar way to derive the second-order derivatives with respect to $\mathbf{Y}$. Define the following constant matrices with respect to $\mathbf{Y}$: $\mathbf{K}_1=\mathrm{vec}(\mathbf{I}_r)'(\mathbf{I}_r\otimes\mathbf{X}'\mathbf{M}_i)$; $\mathbf{K}_2=-\mathrm{diag_M}(\mathrm{vec}(\mathbf{X}'\mathbf{X}))\mathbf{E}_r(\mathbf{I}_{r^2}+\mathbf{C}_{rr})$; $\mathbf{K}_3=\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{X}'\mathbf{M}_i)$. It can be seen that

$$\begin{aligned}\frac{\partial^2\gamma_i}{\partial\mathbf{Y}^2} = &\left(\mathbf{I}_{qr}\otimes\mathbf{K}_1\right)\Big[\left(\mathbf{I}_r\otimes\mathbf{C}_{qr}\otimes\mathbf{I}_q\right)\left(\mathbf{I}_{r^2}\otimes\mathrm{vec}\left(\mathbf{I}_q\right)\right)\mathbf{T}_r\frac{\partial\mathbf{d}_i}{\partial\mathbf{Y}}\\ &+\left(\left(\frac{\partial\mathbf{d}_i}{\partial\mathbf{Y}}\right)'\otimes\mathbf{I}_{qr}\right)\left(\mathbf{T}'_r\otimes\mathbf{I}_{qr}\right)\left(\mathbf{I}_r\otimes\mathbf{C}_{rr}\otimes\mathbf{I}_q\right)\left(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{qr}\right)\\ &+\left(\mathbf{I}_{qr}\otimes(\mathbf{I}_r\otimes\mathbf{Y})\mathbf{T}_r\right)\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{Y}^2}\Big],\end{aligned} \tag{3.20}$$

with $\frac{\partial\mathbf{d}_i}{\partial\mathbf{Y}}$ given by Eq. (3.11) (in Case I) or Eq. (3.12) (in Case II). The term $\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{Y}^2}$ is $\mathbf{0}_{qr^2,qr}$ in Case II; to derive $\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{Y}^2}$ in Case I we start by rewriting (3.11):

$$\frac{\partial\mathbf{d}_i}{\partial\mathbf{Y}} = \mathbf{F}_4^{(i)}\mathbf{K}_2(\mathbf{I}_r\otimes\mathbf{Y}') + (\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{K}_3. \tag{3.21}$$

We can now write:

$$\begin{aligned}\frac{\partial^2\mathbf{d}_i}{\partial\mathbf{Y}^2} = &((\mathbf{I}_r\otimes\mathbf{Y})\mathbf{K}'_2\otimes\mathbf{I}_r)\frac{\partial\mathbf{F}_4^{(i)}}{\partial\mathbf{Y}}\\ &+\left(\mathbf{I}_{qr}\otimes\mathbf{F}_4^{(i)}\right)\left(\mathbf{I}_{qr}\otimes\mathbf{K}_2\right)\left(\mathbf{I}_r\otimes\mathbf{C}_{qr}\otimes\mathbf{I}_r\right)\left(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{qr}\right)\mathbf{C}_{qr}\\ &+(\mathbf{K}'_3\otimes\mathbf{I}_r)\frac{\partial(\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}}{\partial\mathbf{Y}}.\end{aligned} \tag{3.22}$$

In order to derive the crossed derivative define the following constants with respect to $\mathbf{Y}$: $\mathbf{L}_1=\mathrm{vec}(\mathbf{I}_r)'$; $\mathbf{L}_2=(\mathbf{I}_r\otimes\mathbf{X})\mathbf{T}_r$; $\mathbf{L}_3=-\mathbf{E}_r(\mathbf{I}_{r^2}+\mathbf{C}_{rr})(\mathbf{I}_r\otimes\mathbf{X}')$. We can rewrite $\frac{\partial\gamma_i}{\partial\mathbf{X}}$:

$$\frac{\partial\gamma_i}{\partial\mathbf{X}} = \mathbf{L}_1(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)\left(\left(\mathbf{D}_i\otimes\mathbf{I}_p\right)+\mathbf{L}_2\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\right). \tag{3.23}$$

Differentiating $\frac{\partial\gamma_i}{\partial\mathbf{X}}$ with respect to $\mathbf{Y}$ gives us

$$\begin{aligned}\frac{\partial}{\partial\mathbf{Y}}\left(\frac{\partial\gamma_i}{\partial\mathbf{X}}\right) = &\left(\left(\mathbf{D}_i\otimes\mathbf{I}_p\right)+\left(\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\right)'\mathbf{L}'_2\right)\left(\mathbf{I}_{pr}\otimes\mathbf{L}_1\right)\frac{\partial(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)}{\partial\mathbf{Y}}\\ &+\left(\mathbf{I}_{pr}\otimes\mathbf{L}_1(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)\right)\left(\frac{\partial\left(\mathbf{D}_i\otimes\mathbf{I}_p\right)}{\partial\mathbf{Y}}+(\mathbf{I}_{pr}\otimes\mathbf{L}_2)\frac{\partial}{\partial\mathbf{Y}}\left(\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\right)\right),\end{aligned} \tag{3.24}$$

see Appendix B for the derivations of $\frac{\partial\left(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i\right)}{\partial\mathbf{Y}}$ and $\frac{\partial(\mathbf{D}_i\otimes\mathbf{I}_p)}{\partial\mathbf{Y}}$. We have $\frac{\partial}{\partial\mathbf{Y}}\left(\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}\right)$ left to derive. Start by rewriting $\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}}$:

$$\frac{\partial\mathbf{d}_i}{\partial\mathbf{X}} = \mathbf{F}_6^{(i)}\mathbf{L}_3 + (\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)\ \text{for Case I} \tag{3.25}$$

$$\frac{\partial \mathbf{d}_i}{\partial \mathbf{X}} = \mathbf{T}'_r(\mathbf{I}_r \otimes \mathbf{Y}'\mathbf{M}'_i) \text{ for Case II,} \tag{3.26}$$

where $\mathbf{F}_6^{(i)} = (\mathrm{diag}_V(\mathbf{X}'\mathbf{M}_i\mathbf{Y})' \otimes \mathbf{I}_r)\left((\mathbf{X}'\mathbf{X} * \mathbf{Y}'\mathbf{Y})^{-1} \otimes (\mathbf{X}'\mathbf{X} * \mathbf{Y}'\mathbf{Y})^{-1}\right)\mathrm{diag}_M(\mathrm{vec}(\mathbf{Y}'\mathbf{Y}))$.

We have that

$$\begin{aligned}\frac{\partial}{\partial \mathbf{Y}}\left(\frac{\partial \mathbf{d}_i}{\partial \mathbf{X}}\right) &= (\mathbf{L}'_3 \otimes \mathbf{I}_r)\frac{\partial \mathbf{F}_6^{(i)}}{\partial \mathbf{Y}} + ((\mathbf{I}_r \otimes \mathbf{M}_i\mathbf{Y})\mathbf{T}_r \otimes \mathbf{I}_r)\frac{\partial (\mathbf{X}'\mathbf{X} * \mathbf{Y}'\mathbf{Y})^{-1}}{\partial \mathbf{Y}} \\ &\quad + \left(\mathbf{I}_{pr} \otimes (\mathbf{X}'\mathbf{X} * \mathbf{Y}'\mathbf{Y})^{-1}\right)\left(\mathbf{I}_{pr} \otimes \mathbf{T}'_r\right)\frac{\partial (\mathbf{I}_r \otimes \mathbf{Y}'\mathbf{M}'_i)}{\partial \mathbf{Y}}\end{aligned} \tag{3.27}$$

for Case I, and

$$\frac{\partial}{\partial \mathbf{Y}}\left(\frac{\partial \mathbf{d}_i}{\partial \mathbf{X}}\right) = (\mathbf{I}_{pr} \otimes \mathbf{T}'_r)\frac{\partial (\mathbf{I}_r \otimes \mathbf{Y}'\mathbf{M}'_i)}{\partial \mathbf{Y}} \tag{3.28}$$

for Case II.

The Hessian of L_c^{CP} is the $(p+q)r \times (p+q)r$ symmetric matrix

$$\mathrm{Hess}\left(L_c^{\mathrm{CP}}\right) = \begin{bmatrix} \dfrac{\partial^2 L^{\mathrm{CP}}}{\partial \mathbf{X}^2} & \dfrac{\partial}{\partial \mathbf{Y}}\left(\dfrac{\partial L^{\mathrm{CP}}}{\partial \mathbf{X}}\right) \\ \dfrac{\partial}{\partial \mathbf{X}}\left(\dfrac{\partial L^{\mathrm{CP}}}{\partial \mathbf{Y}}\right) & \dfrac{\partial^2 L^{\mathrm{CP}}}{\partial \mathbf{Y}^2} \end{bmatrix} - 2\begin{bmatrix} \mathbf{\Lambda} \otimes \mathbf{I}_p & \mathbf{0} \\ \mathbf{0} & \mathbf{\Delta} \otimes \mathbf{I}_q \end{bmatrix}, \tag{3.29}$$

where $\frac{\partial^2 L^{\mathrm{CP}}}{\partial \mathbf{X}^2} = \sum_{i=1}^m \frac{\partial^2 \gamma_i}{\partial \mathbf{X}^2}$, $\frac{\partial^2 L^{\mathrm{CP}}}{\partial \mathbf{Y}^2} = \sum_{i=1}^m \frac{\partial^2 \gamma_i}{\partial \mathbf{Y}^2}$, $\frac{\partial}{\partial \mathbf{Y}}\left(\frac{\partial L^{\mathrm{CP}}}{\partial \mathbf{X}}\right) = \sum_{i=1}^m \frac{\partial}{\partial \mathbf{Y}}\left(\frac{\partial \gamma_i}{\partial \mathbf{X}}\right)$ and $\frac{\partial}{\partial \mathbf{X}}\left(\frac{\partial L^{\mathrm{CP}}}{\partial \mathbf{Y}}\right) = \left(\frac{\partial}{\partial \mathbf{Y}}\left(\frac{\partial L^{\mathrm{CP}}}{\partial \mathbf{X}}\right)\right)'$.

3.4. Sufficient second-order conditions

A sufficient condition for a stationary point of L_c^{CP} to be a maximum depends on the type of constraint:

- in Case I it is sufficient for a maximum that $\mathbf{W}'\frac{\partial^2 L_c^{\mathrm{CP}}}{\partial \mathbf{X}\partial \mathbf{Y}}\mathbf{W}$ is negative definite, where $\mathbf{W}$ is the $(p+q)r \times (p+q-2)r$ matrix whose columns span the subspace orthogonal to $\begin{bmatrix} \mathbf{I}_r \odot \mathbf{X} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_r \odot \mathbf{Y} \end{bmatrix}$;
- in Case II it is sufficient for a maximum that $\mathbf{W}'\frac{\partial^2 L_c^{\mathrm{CP}}}{\partial \mathbf{X}\partial \mathbf{Y}}\mathbf{W}$ is negative definite, where $\mathbf{W}$ is the $(p+q)r \times (p+q-r-1)r$ matrix whose columns span the subspace orthogonal to matrix

$$\left[\begin{array}{cc|cc} \mathbf{I}_r \odot \mathbf{X} & \mathbf{0} & \mathbf{H}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_r \odot \mathbf{X} & \mathbf{0} & \mathbf{H}_2 \end{array}\right], \tag{3.30}$$

where

$$\mathbf{H}_1 = \begin{bmatrix} \mathbf{x}_2 & \cdots & \mathbf{x}_r & & & & \cdots & \\ \mathbf{x}_1 & & & \mathbf{x}_3 & \cdots & \mathbf{x}_r & \cdots & \\ & & & \mathbf{x}_2 & & & \cdots & \\ & \ddots & & & \ddots & & \cdots & \\ & & & & & & \cdots & \mathbf{x}_r \\ & & \mathbf{x}_1 & & & \mathbf{x}_2 & \cdots & \mathbf{x}_{r-1} \end{bmatrix} \tag{3.31}$$

and $\mathbf{H}_2$ is similar to $\mathbf{H}_1$ with all occurrences of $\mathbf{x}$'s replaced by $\mathbf{y}$'s, Magnus and Neudecker ([10], Chapter 7).

4. Optimization of INDSCAL

In a similar fashion as was done for CP, we can reformulate the problem of minimizing the loss function (1.2) of INDSCAL as equivalent to the problem of maximizing

$$L^{\mathrm{IND}}(\mathbf{X}) = \sum_{i=1}^m \mathrm{tr}\left(\mathbf{X}'\mathbf{S}_i\mathbf{X}\tilde{\mathbf{D}}_i\right), \tag{4.1}$$

where $\tilde{\mathbf{D}}_i$ is the diagonal matrix holding

$$\tilde{\mathbf{d}}_i = \text{transposed row } i \text{ of } \tilde{\mathbf{D}} = (\mathbf{X}'\mathbf{X} * \mathbf{X}'\mathbf{X})^{-1}\mathrm{diag}_V(\mathbf{X}'\mathbf{S}_i\mathbf{X}) \tag{4.2}$$

in the diagonal. The Lagrangean is defined by

$$L_c^{\mathrm{IND}}(\mathbf{X}) = \sum_{i=1}^m \mathrm{tr}\left(\mathbf{X}'\mathbf{S}_i\mathbf{X}\tilde{\mathbf{D}}_i\right) - \mathrm{tr}(\mathbf{\Lambda}(\mathbf{X}'\mathbf{X} - \mathbf{I}_r)), \tag{4.3}$$

where $\mathbf{\Lambda} = \mathrm{Diag}([\lambda_i])$ is a diagonal $r \times r$ matrix holding Lagrange multipliers and $\tilde{\mathbf{D}}$ is given by Eq. (4.2) with the diagonal of $\mathbf{X}'\mathbf{X} * \mathbf{X}'\mathbf{X}$ filled with 1's in Case I, or $\mathbf{\Lambda} = [\lambda_{ij}]$ is a symmetric $r \times r$ matrix holding Lagrange multipliers and $\tilde{\mathbf{D}}$ is given by Eq. (4.2) with $\mathbf{X}'\mathbf{X} * \mathbf{X}'\mathbf{X} = \mathbf{I}_r$ in Case II.

4.1. Derivation of the Jacobian of L_c^{IND}

Define $\sigma_i = \mathrm{tr}(\mathbf{X}'\mathbf{S}_i\mathbf{X}\tilde{\mathbf{D}}_i)$. We have

$$\begin{aligned}\frac{\partial \sigma_i}{\partial \mathbf{X}} &= \mathrm{vec}(\mathbf{I}_r)'\Bigg(\left(\tilde{\mathbf{D}}_i\mathbf{X}' \otimes \mathbf{I}_r\right)(\mathbf{S}_i \otimes \mathbf{I}_r)\mathbf{C}_{pr} + (\mathbf{I}_r \otimes \mathbf{X}'\mathbf{S}_i) \\ &\quad \times \left(\left(\tilde{\mathbf{D}}_i \otimes \mathbf{I}_p\right) + (\mathbf{I}_r \otimes \mathbf{X})\mathbf{T}_r\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}}\right)\Bigg).\end{aligned} \tag{4.4}$$

In Case I the partial derivative of $\tilde{\mathbf{d}}_i$ with respect to $\mathbf{X}$ is

$$\begin{aligned}\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}} &= \left(\mathrm{diag}_V(\mathbf{X}'\mathbf{S}_i\mathbf{X})' \otimes \mathbf{I}_r\right)\frac{\partial (\mathbf{X}'\mathbf{X} * \mathbf{X}'\mathbf{X})^{-1}}{\partial \mathbf{X}} \\ &\quad + (\mathbf{X}'\mathbf{X} * \mathbf{X}'\mathbf{X})^{-1}\mathbf{T}'_r\left((\mathbf{X}'\mathbf{S}_i \otimes \mathbf{I}_r)\mathbf{C}_{pr} + (\mathbf{I}_r \otimes \mathbf{X}')(\mathbf{I}_r \otimes \mathbf{S}_i)\right),\end{aligned} \tag{4.5}$$

see Appendix C for the derivation of $\frac{\partial (\mathbf{X}'\mathbf{X} * \mathbf{X}'\mathbf{X})^{-1}}{\partial \mathbf{X}}$. In Case II we have that

$$\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}} = \mathbf{T}'_r\left((\mathbf{X}'\mathbf{S}_i \otimes \mathbf{I}_r)\mathbf{C}_{pr} + (\mathbf{I}_r \otimes \mathbf{X}')(\mathbf{I}_r \otimes \mathbf{S}_i)\right). \tag{4.6}$$

The Jacobian of L_c^{IND} is the $1 \times pr$ row vector

$$\mathrm{Jac}\left(L_c^{\mathrm{IND}}\right) = \sum_{i=1}^m \frac{\partial \sigma_i}{\partial \mathbf{X}} - 2\mathrm{vec}(\mathbf{X})'\left(\mathbf{\Lambda} \otimes \mathbf{I}_p\right). \tag{4.7}$$

4.2. Derivation of the Lagrange multipliers

Proceeding in a similar fashion as done in Section 3, it is straightforward to verify that

$$\lambda_{kk} = \frac{1}{2}\sum_{i=1}^m \frac{\partial \sigma_i}{\partial \mathbf{x}_k}\mathbf{x}_k \tag{4.8}$$

in Case I, and

$$\lambda_{jk} = \frac{1}{2}\sum_{i=1}^m \frac{\partial \sigma_i}{\partial \mathbf{x}_k}\mathbf{x}_j \tag{4.9}$$

in Case II ($j, k = 1, \ldots, r$).

4.3. Derivation of the Hessian of L_c^{IND}

Now define the following matrices which are constant with respect to $\mathbf{X}$: $\mathbf{N}_1 = \mathrm{vec}(\mathbf{I}_r)'$; $\mathbf{N}_2 = (\mathbf{S}_i \otimes \mathbf{I}_r)\mathbf{C}_{pr}$; $\mathbf{N}_3 = \mathbf{E}_r(\mathbf{I}_{r^2} + \mathbf{C}_{rr})$. We can rewrite

$$\frac{\partial \sigma_i}{\partial \mathbf{X}} = \mathbf{N}_1\left(\mathbf{G}_2^{(i)}\mathbf{N}_2 + \mathbf{G}_6^{(i)} + \mathbf{G}_7^{(i)}\mathbf{T}_r\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}}\right). \tag{4.10}$$

It can be seen that

$$\frac{\partial^2 \sigma_i}{\partial \mathbf{X}^2} = \left(\mathbf{I}_{pr}\otimes\mathbf{N}_1\right)\left((\mathbf{N}'_2\otimes\mathbf{I}_{r^2})\frac{\partial \mathbf{G}_2^{(i)}}{\partial \mathbf{X}} + \frac{\partial \mathbf{G}_6^{(i)}}{\partial \mathbf{X}} + \left(\left(\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}}\right)'\otimes\mathbf{I}_{r^2}\right)\right. \tag{4.11}$$
$$\left.\times\,(\mathbf{T}'_r\otimes\mathbf{I}_{r^2})\frac{\partial \mathbf{G}_7^{(i)}}{\partial \mathbf{X}} + \left(\mathbf{I}_{pr}\otimes\mathbf{G}_7^{(i)}\mathbf{T}_r\right)\frac{\partial^2 \tilde{\mathbf{d}}_i}{\partial \mathbf{X}^2}\right),$$

with $\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}}$ given by Eq. (4.5) (in Case I) or Eq. (4.6) (in Case II). To derive $\frac{\partial^2 \tilde{\mathbf{d}}_i}{\partial \mathbf{X}^2}$ in Case I we start by rewriting Eq. (4.5):

$$\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}} = 2\mathbf{G}_{14}^{(i)}\mathbf{N}_3\mathbf{G}'_5 + \mathbf{G}_8\mathbf{T}'_r\mathbf{G}_{16}^{(i)}. \tag{4.12}$$

It can be seen that

$$\frac{\partial^2 \tilde{\mathbf{d}}_i}{\partial \mathbf{X}^2} = 2(\mathbf{G}_5\mathbf{N}'_3\otimes\mathbf{I}_r)\frac{\partial \mathbf{G}_{14}^{(i)}}{\partial \mathbf{X}} + 2\left(\mathbf{I}_{pr}\otimes\mathbf{G}_{14}^{(i)}\right)\left(\mathbf{I}_{pr}\otimes\mathbf{N}_3\right)\mathbf{C}_{pr,r^2}\frac{\partial \mathbf{G}_5}{\partial \mathbf{X}} + \tag{4.13}$$

$$+\left(\left(\mathbf{G}_{16}^{(i)}\right)'\mathbf{T}_r\otimes\mathbf{I}_r\right)\frac{\partial \mathbf{G}_8}{\partial \mathbf{X}} + \left(\mathbf{I}_{pr}\otimes\mathbf{G}_8\right)\left(\mathbf{I}_{pr}\otimes\mathbf{T}'_r\right)\frac{\partial \mathbf{G}_{16}^{(i)}}{\partial \mathbf{X}}. \tag{4.14}$$

In Case II we have that

$$\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}} = \mathbf{T}'_r\mathbf{G}_{16}^{(i)}, \quad \frac{\partial^2 \tilde{\mathbf{d}}_i}{\partial \mathbf{X}^2} = \left(\mathbf{I}_{pr}\otimes\mathbf{T}'_r\right)\frac{\partial \mathbf{G}_{16}^{(i)}}{\partial \mathbf{X}}. \tag{4.15}$$

The Hessian of L_c^{IND} is the $pr\times pr$ symmetric matrix

$$\mathrm{Hess}\left(L_c^{\mathrm{IND}}\right) = \sum_{i=1}^{m}\frac{\partial^2 \sigma_i}{\partial \mathbf{X}^2} - 2\left(\mathbf{\Lambda}\otimes\mathbf{I}_p\right). \tag{4.16}$$

4.4. Sufficient second-order conditions

A sufficient condition for a stationary point of L_c^{IND} to be a maximum depends on the type of constraint:

- in Case I it is sufficient that $\mathbf{W}'\frac{\partial^2 L_c^{\mathrm{IND}}}{\partial\mathbf{X}\partial\mathbf{Y}}\mathbf{W}$ is negative definite, where $\mathbf{W}$ is the $pr\times(pr-r)$ matrix whose columns span the subspace orthogonal to $\mathbf{I}_r\odot\mathbf{X}$;
- in Case II it is sufficient that $\mathbf{W}'\frac{\partial^2 L_c^{\mathrm{IND}}}{\partial\mathbf{X}\partial\mathbf{Y}}\mathbf{W}$ is negative definite, where $\mathbf{W}$ is the $pr\times\left(pr-\frac{r(r+1)}{2}\right)$ matrix whose columns span the subspace orthogonal to matrix

$$[\mathbf{I}_r\odot\mathbf{X}\,|\,\mathbf{H}], \tag{4.17}$$

where $\mathbf{H}$ is the same as in Eq. (3.31).

5. Illustration: the KHL data

Ten Berge, Kiers and De Leeuw [17] analysed a contrived array which they christened "KHL data", due to previous work by Kruskal, Harshman and Lundy [8,9]. The KHL data is the $2\times2\times2$ array

$$\underline{\mathbf{X}} = \left[\begin{array}{cc|cc}1 & 0 & 0 & 1\\ 0 & -1 & 1 & 0\end{array}\right]. \tag{5.1}$$

We ran the ALS algorithm 200 times for $\underline{\mathbf{X}}$ with $r=2$ components. The component matrices were randomly initialized by orthonormal matrices. In all runs the algorithm halted on solutions with loss $f=2$. We wanted to test the nature of these solutions, i.e., whether these solutions correspond to minima and/or saddle points.

We computed the Jacobian and Hessian for each of the 200 solutions under unit length constraint. In general, each of the 200 solutions displays a similar behaviour: $\mathbf{A}$ has rank 1, $\mathbf{B}$ and $\mathbf{C}$ are orthonormal, Jac is approximately $\mathbf{0}_{1\times8}$ (its entries are usually in the order of 10^{-14}), and the Hess is of the form

$$\mathrm{Hess}_{\mathrm{CP}} = \begin{bmatrix}0 & 0 & a & b\\ 0 & 0 & -a & -b\\ a & -a & 0 & 0\\ b & -b & 0 & 0\end{bmatrix}, \tag{5.2}$$

for real numbers a, b. The eigenvalues of $\mathrm{Hess}_{\mathrm{CP}}$ are typically $\{0, 0, -\lambda, \lambda\}$, for real λ. Therefore, it can be concluded that each of the 200 solutions are, indeed, saddle points.

This example shows two things. On one hand, there exist cases for which the occurrence of saddle points is a severe problem, like the KHL data. On the other hand, it is relevant to have a tool available that diagnoses whether a solution is a saddle point. Once spotted, such solutions should be discarded at once.

Ten Berge, Kiers and De Leeuw [17] showed that the CP loss function (1.1) has infimum 1 when 2 components are extracted. This reinforces the fact that none of the 200 solutions that were found could correspond to the global minimum. However, in the absence of this information, the researcher would profit from knowing that all solutions were saddle points and therefore useless. This is possible by analysing the second-order differential structure as we have done here.

The KHL data is a contrived example. The question of whether similar behaviour is to be expected for real data is still unanswered. The applications discussed in Sections 6–8 are intended to better understand what happens in general.

6. Application I: INDSCAL under orthonormality constraint

Ten Berge et al. [18] discussed an algorithm for INDSCAL with orthogonality constraints referred to as the SVD-approach. This algorithm was originally deviced as a Varimax procedure based on an SVD, but Ten Berge [15] observed that the problem could be reformulated in terms of diagonalizing a set of symmetric matrices simultaneously. The SVD-approach provides a direct procedure to fit the INDSCAL model under orthogonality constraints.

The SVD-approach attempts to find a columnwise orthonormal $\mathbf{X}$ such that $L_c^{\mathrm{IND}}(\mathbf{X})$ is maximized; it proceeds as follows:

Step 1 Initialize $\mathbf{X}$ ($p\times r$ orthonormal).
Step 2 Compute $\mathbf{D}_i = \mathrm{diag}(\mathbf{X}'\mathbf{S}_i\mathbf{X})$, $i=1, \ldots, m$.
Step 3 Compute the SVD $\sum_{i=1}^{m}\mathbf{S}_i\mathbf{X}\mathbf{D}_i = \mathbf{PLQ}'$, and update $\mathbf{X}$ by $\mathbf{X}=\mathbf{PQ}'$.
Step 4 Repeat Steps 2 and 3 until the relative increase in $L_c^{\mathrm{IND}}(\mathbf{X})$ is smaller than a predefined convergence criterion.

The SVD-approach to INDSCAL has been proved to converge monotonically when the frontal slices of array $\underline{\mathbf{S}}$ are positive or negative semidefinite, Ten Berge et al. [18]. Thus, we will work with arrays holding semidefinite frontal slices in the remaining of this section.

Ten Berge et al. [18] ran some experiments where they argue that the SVD-approach to INDSCAL seems to be hampered by the occurrence of local maxima of L_c^{IND}. However, the possibility of the occurrence of saddle points was not considered. Notice that there exist contrived examples for which saddle points do occur. For example, consider the $2\times3\times3$ array with positive semidefinite slices (Ten Berge and Kiers [16])

$$\mathbf{S}_1 = \begin{bmatrix}3 & 1 & 0\\ 1 & 3 & 0\\ 0 & 0 & 0\end{bmatrix}, \quad \mathbf{S}_2 = \begin{bmatrix}3 & -1 & 0\\ -1 & 3 & 0\\ 0 & 0 & 1\end{bmatrix}. \tag{6.1}$$

The (orthonormally constrained) INDSCAL optimal solution with $r = 2$ components is

$$\mathbf{X} = \begin{bmatrix} \sqrt{.5} & -\sqrt{.5} \\ \sqrt{.5} & \sqrt{.5} \\ 0 & 0 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}; \tag{6.2}$$

it corresponds to the global minimum 1 of (1.2). There are, however, non-optimal orthonormally constrained INDSCAL solutions corresponding to saddle points. The following four solutions are stationary points of (1.2) that correspond to non-optimal values of (1.2) (5, 20, 22, 22, respectively). They are all saddle points.

$$\mathbf{X}^{(1)} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{D}^{(1)} = \begin{bmatrix} 3 & 3 \\ 3 & 3 \end{bmatrix} \tag{6.3}$$

$$\mathbf{X}^{(2)} = \begin{bmatrix} \sqrt{.5} & 0 \\ \sqrt{.5} & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{D}^{(2)} = \begin{bmatrix} 4 & 0 \\ 2 & 1 \end{bmatrix} \tag{6.4}$$

$$\mathbf{X}^{(3)} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{D}^{(3)} = \begin{bmatrix} 0 & 3 \\ 1 & 3 \end{bmatrix} \tag{6.5}$$

$$\mathbf{X}^{(4)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{D}^{(4)} = \begin{bmatrix} 3 & 0 \\ 3 & 1 \end{bmatrix}. \tag{6.6}$$

The fact that such non-optimal solutions exist does not imply that the SVD-approach algorithm will converge to them. This is precisely the point that we wanted to investigate in this application: is it possible that the SVD-approach algorithm converges to saddle points? The answer to this question can clarify the type of solutions that the SVD-approach usually finds, therefore the interpretation of the solution is further enriched.

A simulation study was carried out to test whether saddle points occur (software: *Matlab* R2008a). We randomly generated 150 $3\times3\times3$ symmetric slice arrays with positive definite slices. Each slice was generated as $\mathbf{M}'\mathbf{M}$, where $\mathbf{M}$ is a 3×3 matrix whose entries were uniformly generated from the interval $[-1, 1]$. For each array we ran the SVD-approach to INDSCAL with $r = 2$ components using 10 different random initializations for $\mathbf{X}$; each $\mathbf{X}$ was a 3×2 matrix whose entries were uniformly generated from the interval $[-1, 1]$; afterwards, $\mathbf{X}$ was orthonormalized via the Gram–Schmidt procedure. The convergence criterion was fixed at $1e-06$. After convergence, the Jacobian and Hessian for each INDSCAL solution ($\mathbf{X}$, $\mathbf{D}$) were computed, and we inspected whether $\mathbf{W}' \frac{\partial^2 L_c^{\mathrm{IND}}}{\partial \mathbf{X} \partial \mathbf{Y}} \mathbf{W}$ was negative definite or indefinite (second-order sufficient condition).

The same procedure was repeated, this time for arrays with positive semidefinite slices. Each slice was generated as $\mathbf{M}'\mathbf{M}$, where $\mathbf{M}$ is a 2×3 matrix whose entries were uniformly generated from the interval $[-1, 1]$.

All results were numerically stable, as expected. We verified that the SVD-approach for INDSCAL never halted on saddle points. Also, it was verified that local maxima occurred for 12 arrays (8% of the cases) for arrays with positive definite slices, whereas for arrays with positive semidefinite slices local maxima occurred for 17 arrays (11% of the cases). Although these results do not formally prove that convergence to saddle points is impossible, it can be concluded that there are no indications to that effect.

7. Application II: INDSCAL equivalence problem in CP formulation

Carroll and Chang [3] suggested running CP in order to fit INDSCAL because they conjectured that $\mathbf{X}$ and $\mathbf{Y}$ would end up equal or at least columnwise proportional if CP converged. If Carroll and Chang's conjecture is correct, CP can be used as an algorithm to compute INDSCAL solutions for symmetric slice arrays. This conjecture seems to be valid in practical applications. However, counter-examples have already been constructed. Ten Berge and Kiers [16] proved that equivalence may be violated at global minima of f if the slices $\mathbf{S}_i$ are not positive definite. They considered the array (Ten Berge and Kiers [16])

$$\mathbf{S}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{S}_2 = \begin{bmatrix} 0 & 0 & 2 \\ 0 & -2 & 0 \\ 2 & 0 & 0 \end{bmatrix}, \tag{7.1}$$

for which a global minimum of (1.1) with $r = 2$ components and $\mathbf{X}$ not equivalent to $\mathbf{Y}$ was presented. We ran CP 500 times with $r = 2$ and $r = 3$ components for the previous array. In both cases all runs converged to a global minimum of (1.1) with $\mathbf{X}$ and $\mathbf{Y}$ non-equivalent. When $r = 1$ the algorithm sometimes did converge to a solution with $\mathbf{X}$ and $\mathbf{Y}$ equivalent.

When the slices are nonnegative definite and $r = 1$ then equivalence can be violated only at stationary points that do not correspond to global minima. In this case, Ten Berge and Kiers [16] conjectured that such stationary points would correspond to local minima. However, Bennani Dosse and Ten Berge [1] proved that such stationary points can only be saddle points.

Bennani Dosse, Ten Berge and Tendeiro [2] showed that equivalence occurs when the components are constrained by orthonormality, the slices are positive semidefinite and the saliences are non-negative. It is still not clear whether non-equivalence occurs or not under circumstances different from these, or whether CP converges to saddle points or not. We conducted some simulations to try to understand what happens in cases where components are not orthonormal, slices are not necessarily positive semidefinite, and saliences are unconstrained. Eleven situations were considered, revolving around arrays with 3×3 symmetric slices: $2\times3\times3$ ($r = 2$), $3\times3\times3$ ($r = 2$, 3), $4\times3\times3$($r = 2$, 3), $5\times3\times3$ ($r = 2$, 3, 4), $6\times3\times3$ ($r = 2$, 3, 4). Both positive definite and indefinite slice arrays were considered.

One hundred arrays were generated for each case. The positive definite slices were generated as $\mathbf{M}'\mathbf{M}$, where $\mathbf{M}$ is a 3×3 matrix whose entries were uniformly generated from the interval $[-1, 1]$. The entries of the diagonal and the upper-triangular parts of the indefinite slices were uniformly generated from the interval $[-1, 1]$; the lower-triangular part of each slice was filled in such that symmetry would occur. Each array was given 100 different random startups; the convergence criterium was set at $1e-08$. No constraint was imposed on the saliences in $\mathbf{D}$. A solution was declared degenerate when at least one of the non-diagonal entries of the so-called triple cosine matrix was below -0.95. Our main goal was to check whether non-equivalence occurred or not, and to what kind of stationary point it corresponded (local optimum or saddle point).

The Jacobian matrices associated to non-degenerate solutions were analysed. Its entries were relatively small (usually with modulus smaller than $1e-05$), thus analysing the second-order differential structure seems legitimate. We worked under unit length constraints. Occurrences of degeneracy and of different values for the loss function were registered. The results found are summarized in Tables 1 and 2. The variables read: NonEquiv = number of arrays for which at least one startup ended up with non-equivalent CP solution; Deg = number of arrays with degenerate solutions, within the 100 startups ($x+y$: x = all 100 startups are degenerate; y = < 100 startups are degenerate); SadPt = number of arrays for which at least one startup ended in a saddle point, non-degenerate (x/y: x = for CP's Hessian; y = for INDSCAL's Hessian); DiffFit = number of arrays with at least two different values for CP's loss function within the 100 startups, with at least one non-degenerate solution. Some special situations are marked with asterisks, as follows: (*) = the associated CP

Table 1
Arrays with positive definite slices.

Dim. array	# comps	NonEquiv	Deg	SadPt	DiffFit
$2\times3\times3$	$r=2$	–	–	–	4
$3\times3\times3$	$r=2$	–	–	–	12
	$r=3$	–	$2+2$	–	5
$4\times3\times3$	$r=2$	–	–	–	17
	$r=3$	–	$3+0$	–	8
$5\times3\times3$	$r=2$	–	–	–	10
	$r=3$	–	$0+1$	–	7
	$r=4$	–	$40+7$	–	20
$6\times3\times3$	$r=2$	–	–	–	10
	$r=3$	–	–	–	7
	$r=4$	–	$45+7$	2/2 (**)	7

solution is degenerate (all components are almost proportional); (**) = one or more of the eigenvalues of the Hessian are relatively small in magnitude (smaller than $1e-01$), indicating that the Hessian is nearly singular; (***) = for one startup of one array the Hessian for CP was nearly singular, but the Hessian for INDSCAL was negative definite.

The first observation to be made is that non-equivalence was never observed for non-degenerate solutions. Since no non-equivalent solution was found for arrays with positive definite slices, it was not possible to test whether the result of Bennani Dosse and Ten Berge [1] for arrays with positive definite slices does apply to cases with $r>1$ components. Also, saddle points were rarely observed. In addition, we can observe that arrays with indefinite slices are more prone to suffer from degeneracy, occurrence of saddle points, and multiple fit values.

The cases reported with (**) are situations where it is not clear whether we are facing a saddle point or not, since the Hessian matrix seems to be almost singular. These cases should be treated with care, since the second-order sufficient condition applies to non-singular Hessian matrices. It is not clear why such points occur. An anonymous reviewer suggested that the problem might be originated in rank-deficient component matrices. We verified that this was true for five of the situations reported by (**). It should be noted that these component matrices were estimated rather than randomly generated, and that these decompositions are not degenerate.

8. Application III: CP in general

A simulation study was conducted to inspect the occurrence of saddle points for CP solutions of generic arrays. Twenty nine situations were considered, for which uniqueness is proved to hold due to Kruskal's sufficient condition for uniqueness (Kruskal [7]). One hundred arrays were randomly generated for each situation, the entries being uniformly generated from the interval $[-1, 1]$. Each array was given 100 different random startups; the convergence criterium was set at $1e-08$. As before, we also registered the occurrences of degeneracies and multiple fit values. Both $r=1$ and $r>1$ situations were considered. We computed the Hessian under unit length constraint. The results found are summarized in Table 3.

Table 2
Arrays with indefinite slices.

Dim. array	# comps	NonEquiv	Deg	SadPt	DiffFit
$2\times3\times3$	$r=2$	–	$38+19$	2/1 (**)(***)	32
$3\times3\times3$	$r=2$	–	$21+39$	3/3	62
	$r=3$	3 (*)	$63+11$	1/1 (**)	19
$4\times3\times3$	$r=2$	–	$24+39$	–	64
	$r=3$	3 (*)	$55+15$	—	26
$5\times3\times3$	$r=2$	–	$23+39$	–	72
	$r=3$	2 (*)	$59+15$	1/1 (**)	25
	$r=4$	1 (*)	$81+5$	1/1 (**)	10
$6\times3\times3$	$r=2$	–	$20+48$	–	76
	$r=3$	–	$63+16$	1/1	27
	$r=4$	–	$86+5$	–	7

It can be seen that saddle points occur scarcely; almost all these occurrences relate to a nearly singular Hessian matrix. It is not clear why such solutions occur. In addition, we point out that retaining more components seems to have the effect of increasing the number of degenerate solutions.

9. Discussion

In this paper we dealt with first and second-order differential structures of optimization functions related to CP and INDSCAL. Our goal was to provide a tool to further characterize three-way solutions. Closed form formulas for the Jacobian and Hessian matrices were derived, under two different types of constraints.

Simulations that highlight the usefulness of Hessian structure were performed. The results of the simulations seem to tell that saddle points do not occur frequently, although they do occur with positive probability. In some cases the Hessian matrix showed to be ill-conditioned. The reasons for this phenomenon are still not clear and need further investigation.

Some numerical problems occur when we consider degenerate CP/INDSCAL solutions (Harshman and Lundy [6]). Typically, a degenerate solution is one where some components become more and more proportional, while some entries of these components become larger and larger, as the algorithm progresses. In a degenerate solution, the contributions of some of the degenerate components nearly cancel the contributions of other degenerate components, while the components together contribute to improve the fit.

The computation of the Jacobian and the Hessian matrices are free of numerical problems for CP/INDSCAL solutions which are not degenerate. However, degenerate solutions do lead to problems. These problems are more or less severe depending on how many degenerate components exist and how strong the degeneracy is. The core of this problem resides in matrix $\boldsymbol{\Gamma}=\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y}$ (for CP) and $\boldsymbol{\Gamma}=\mathbf{X}'\mathbf{X}*\mathbf{X}'\mathbf{X}$ (for INDSCAL), recall Eqs. (3.2) and (4.2). When a CP/INDSCAL solution is degenerate, $\boldsymbol{\Gamma}$ becomes almost rank deficient, which creates numerical problems when computing $\boldsymbol{\Gamma}^{-1}$. Equivalently, the problem is that the optimization function is (almost) non-differentiable at the

Table 3
Arrays with generic slices.

Dim. array	# comps	Deg	SadPt	DiffFit
$2\times2\times2$	$r=1$	–	–	13
$3\times2\times2$	$r=1$	–	–	25
	$r=2$	$18+2$	1 (**)	1
$3\times3\times2$	$r=1$	–	–	39
	$r=2$	$25+6$	–	14
$3\times3\times3$	$r=1$	–	–	43
	$r=2$	$19+15$	1	30
	$r=3$	$50+12$	1 (**)	17
$4\times2\times2$	$r=1$	–	1	32
	$r=2$	$22+2$	1 (**)	2
$4\times3\times2$	$r=1$	–	1	43
	$r=2$	$17+12$	–	22
	$r=3$	$54+4$	1,2 (**)	4
$4\times4\times2$	$r=1$	–	–	41
	$r=2$	$28+12$	1 (**)	26
	$r=3$	$53+8$	3 (**)	3
$4\times3\times3$	$r=1$	–	1 (**)	57
	$r=2$	$15+32$	1	54
	$r=3$	$52+15$	–	17
	$r=4$	$68+15$	1,1 (**)	17
$5\times2\times2$	$r=1$	–	–	36
	$r=2$	$22+0$	–	–
$5\times3\times2$	$r=1$	–	–	44
	$r=2$	$23+11$	1 (**)	29
	$r=3$	$59+0$	–	—
$5\times4\times2$	$r=1$	–	–	57
	$r=2$	$14+14$	1 (**)	27
	$r=3$	$47+11$	1 (**)	15
	$r=4$	$78+6$	1 (**)	7

point corresponding to a degenerate solution. The problem might vary from mild to severe, depending on how close or far is $\mathbf{\Gamma}$ from singularity. In some severe situations the computations might need to be completely disregarded. For instance, we observed solutions for which the severity of the deficiency of $\mathbf{\Gamma}$ leads to loss of symmetry of the Hessian, which is naturally a serious problem.

A CP solution is never degenerate under an orthonormality constraint on $\mathbf{X}$ and $\mathbf{Y}$, Harshman and Lundy [6]. Likewise, an INDSCAL solution is not degenerate if $\mathbf{X}$ is constrained by orthonormality. Therefore, orthonormality constraints typically avoid any numerical problems. In any other case, we advise to first check whether the solution at hand is degenerate or not. If the solution is not degenerate then the use of the formulas to compute the Jacobian and the Hessian is warranted. In case of degeneracy, one should do some prior analysis on the rank deficiency of $\mathbf{\Gamma}$. If the problem is not very severe, it is possible that $\mathbf{\Gamma}^{-1}$ is relatively well defined, and therefore all the computations will follow safely. A posterior test to the numerical stability of the process is, for example, to compute the Hessian matrix Hess and afterwards compute $\rho = \mathrm{tr}(\mathrm{Hess} - \mathrm{Hess}')$; large values of ρ (say, $\rho > 1e-20$) indicate that Hess is further from symmetry than it should. Therefore, Hess should be discarded in such cases.

Appendix A. Matrix differentiation formulas

Consider the following matrices: $\mathbf{A}$ $(m\times n)$; $\mathbf{B}$ $(p\times q)$; $\mathbf{d}$ $(n\times 1)$; $\mathbf{D} = \mathrm{diag}_\mathrm{M}(\mathbf{d})$.

Table A1 presents the most important formulas of matrix differentiation that are of use in this paper.

Table A1
Matrix differentiation formulas.

$\frac{\partial \mathbf{AB}}{\partial \mathbf{X}} = (\mathbf{B}'\otimes\mathbf{I}_m)\frac{\partial \mathbf{A}}{\partial \mathbf{X}} + (\mathbf{I}_q\otimes\mathbf{A})\frac{\partial \mathbf{B}}{\partial \mathbf{X}}$, if $n = p$	$\frac{\partial \mathrm{tr}(\mathbf{A})}{\partial \mathbf{X}} = \mathrm{vec}(\mathbf{I}_n)'\frac{\partial \mathbf{A}}{\partial \mathbf{X}}$, if $m = n$
$\frac{\partial \mathbf{A}(\mathbf{B}(\mathbf{X}))}{\partial \mathbf{X}} = \frac{\partial \mathbf{A}(\mathbf{Y})}{\partial \mathbf{Y}}\cdot\frac{\partial \mathbf{Y}}{\partial \mathbf{X}}$, where $\mathbf{Y} = \mathbf{B}(\mathbf{X})$	$\frac{\partial \mathbf{A}}{\partial \mathbf{A}} = \mathbf{I}_{mn}, \quad \frac{\partial \mathbf{A}'}{\partial \mathbf{A}} = \mathbf{C}_{mn}$
$\frac{\partial \mathbf{A}'\mathbf{A}}{\partial \mathbf{A}} = (\mathbf{I}_{n^2} + \mathbf{C}_{nn})(\mathbf{I}_n\otimes\mathbf{A}')$	$\frac{\partial \mathbf{A}*\mathbf{B}}{\partial \mathbf{X}} = \mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{B}))\frac{\partial \mathbf{A}}{\partial \mathbf{X}} + \mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{A}))\frac{\partial \mathbf{B}}{\partial \mathbf{X}}$
$\frac{\partial \mathbf{A}^{-1}}{\partial \mathbf{A}} = -\left(\left(\mathbf{A}^{-1}\right)'\otimes\mathbf{A}^{-1}\right)$, if $m = n$	$\frac{\partial \mathbf{D}}{\partial \mathbf{d}} = \mathbf{T}_n$
$\frac{\partial \mathbf{A}\otimes\mathbf{B}}{\partial \mathbf{A}} = \left(\mathbf{I}_n\otimes\mathbf{C}_{qm}\otimes\mathbf{I}_p\right)(\mathbf{I}_{mn}\otimes\mathrm{vec}(\mathbf{B}))$	$\frac{\partial \mathbf{D}}{\partial \mathbf{X}} = \mathbf{T}_n\frac{\partial \mathbf{d}}{\partial \mathbf{X}}$
$\frac{\partial \mathbf{A}\otimes\mathbf{B}}{\partial \mathbf{B}} = \left(\mathbf{I}_n\otimes\mathbf{C}_{qm}\otimes\mathbf{I}_p\right)(\mathrm{vec}(\mathbf{A})\otimes\mathbf{I}_{pq})$	$\frac{\partial \mathrm{vec}(\mathbf{A})}{\partial \mathbf{A}} = \mathbf{I}_{mn}$

The formulas in the first column can be found in Fackler [4]. The formulas in the second column can be obtained as follows:

•

$$\frac{\partial \mathrm{tr}(\mathbf{A})}{\partial \mathbf{X}} = \frac{\partial \mathrm{tr}(\mathbf{A})}{\partial \mathbf{A}}\frac{\partial \mathbf{A}}{\partial \mathbf{X}} = \mathrm{vec}(\mathbf{I}_n)'\frac{\partial \mathbf{A}}{\partial \mathbf{X}}$$

•

$$\frac{\partial \mathbf{A}}{\partial \mathbf{A}} = \frac{\partial \mathrm{vec}(\mathbf{A})}{\partial \mathrm{vec}(\mathbf{A})} = \mathbf{I}_{mn}$$

$$\frac{\partial \mathbf{A}'}{\partial \mathbf{A}} = \frac{\partial \mathrm{vec}(\mathbf{A}')}{\partial \mathrm{vec}(\mathbf{A})} = \frac{\partial \mathbf{C}_{mn}\mathrm{vec}(\mathbf{A})}{\partial \mathrm{vec}(\mathbf{A})} = (\mathbf{I}_1\otimes\mathbf{C}_{mn})\mathbf{I}_{mn} = \mathbf{C}_{mn}$$

- entry (i,j) of $\mathbf{A}*\mathbf{B}$ is equal to $a_{ij}b_{ij}$, where both a_{ij} and b_{ij} are functions of $\mathbf{X}$. Therefore we can apply the rule to differentiate a product to each entry of $\mathbf{A}*\mathbf{B}$. The derivative of $\mathbf{A}*\mathbf{B}$ with respect to $\mathbf{X}$ when $\mathbf{B}$ is constant is equal to $\mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{B}))\frac{\partial \mathbf{A}}{\partial \mathbf{X}}$, and the derivative of $\mathbf{A}*\mathbf{B}$ with respect to $\mathbf{X}$ when $\mathbf{A}$ is constant is equal to $\mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{A}))\frac{\partial \mathbf{B}}{\partial \mathbf{X}}$.
- $\frac{\partial \mathbf{D}}{\partial \mathbf{d}} = \frac{\partial \mathrm{vec}(\mathbf{D})}{\partial \mathbf{d}}$ where $\mathrm{vec}(\mathbf{D})$ is the $n^2\times 1$ vector $[d_1 0\cdots 0|\cdots|0\cdots 0 d_n]'$. The derivative of $\mathrm{vec}(\mathbf{D})$ with respect to d_i is the zero vector except for entry $(i-1)n+i$ where it is 1. Collecting all these derivatives side by side in a $n^2\times n$ matrix gives $\mathbf{T}_n$.
- $\frac{\partial \mathbf{D}}{\partial \mathbf{X}} = \frac{\partial \mathbf{D}}{\partial \mathbf{d}}\frac{\partial \mathbf{d}}{\partial \mathbf{X}} = \mathbf{T}_n\frac{\partial \mathbf{d}}{\partial \mathbf{X}}$
- $\frac{\partial \mathrm{vec}(\mathbf{A})}{\partial \mathbf{A}} = \frac{\partial \mathrm{vec}(\mathbf{A})}{\partial \mathrm{vec}(\mathbf{A})} = \mathbf{I}_{mn}$

Appendix B

Notation: $\underline{\mathbf{M}}$ is a $p\times q\times m$ array with $p\times q$ frontal slices $\mathbf{M}_i$ $(i = 1, \ldots, m)$, $\mathbf{X}$ is a $p\times r$ matrix, $\mathbf{Y}$ is a $q\times r$ matrix. $\mathbf{D}_i$ is the diagonal matrix defined by Eq. (3.2).

Table B1 summarizes the expressions of functions $\mathbf{F}_1 - \mathbf{F}_8$. Also, the partial derivatives that are relevant for the paper are presented.

Table B1
Functions $\mathbf{F}_1$ to $\mathbf{F}_8$ and some of their partial derivatives.

Function	Derivative
$\mathbf{F}_1 = (\mathbf{X}'\mathbf{X}*\mathbf{Y}'\mathbf{Y})^{-1}$	In Case I ($\mathbf{X}$,$\mathbf{Y}$ constrained to hold unit length columns): $\frac{\partial \mathbf{F}_1}{\partial \mathbf{X}} = -(\mathbf{F}_1\otimes\mathbf{F}_1)\mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{Y}'\mathbf{Y}))\mathbf{E}_r(\mathbf{I}_{r^2} + \mathbf{C}_{rr})(\mathbf{I}_r\otimes\mathbf{X}')$ $\frac{\partial \mathbf{F}_1}{\partial \mathbf{Y}} = -(\mathbf{F}_1\otimes\mathbf{F}_1)\mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{X}'\mathbf{X}))\mathbf{E}_r(\mathbf{I}_{r^2} + \mathbf{C}_{rr})(\mathbf{I}_r\otimes\mathbf{Y}')$ In Case II ($\mathbf{X}$,$\mathbf{Y}$ constrained by orthonormality): $\frac{\partial \mathbf{F}_1}{\partial \mathbf{X}} = \mathbf{0}_{r^2,pr}$; $\frac{\partial \mathbf{F}_1}{\partial \mathbf{Y}} = \mathbf{0}_{r^2,qr}$
$\mathbf{F}_2^{(i)} = (\mathrm{diag}_\mathrm{V}(\mathbf{X}'\mathbf{M}_i\mathbf{Y})'\otimes\mathbf{I}_r)$	$\frac{\partial \mathbf{F}_2^{(i)}}{\partial \mathbf{X}} = (\mathbf{I}_r\otimes\mathrm{vec}(\mathbf{I}_r))\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i)$ $\frac{\partial \mathbf{F}_2^{(i)}}{\partial \mathbf{Y}} = (\mathbf{I}_r\otimes\mathrm{vec}(\mathbf{I}_r))\mathbf{T}'_r(\mathbf{I}_r\otimes\mathbf{X}'\mathbf{M}_i)$
$\mathbf{F}_3 = \mathbf{F}_1\otimes\mathbf{F}_1$	$\frac{\partial \mathbf{F}_3}{\partial \mathbf{X}} = (\mathbf{I}_r\otimes\mathbf{C}_{rr}\otimes\mathbf{I}_r)(\mathbf{I}_{r^4} + \mathbf{C}_{r^2r^2})(\mathbf{I}_{r^2}\otimes\mathrm{vec}(\mathbf{F}_1))\frac{\partial \mathbf{F}_1}{\partial \mathbf{X}}$ $\frac{\partial \mathbf{F}_3}{\partial \mathbf{Y}} = (\mathbf{I}_r\otimes\mathbf{C}_{rr}\otimes\mathbf{I}_r)(\mathbf{I}_{r^4} + \mathbf{C}_{r^2r^2})(\mathbf{I}_{r^2}\otimes\mathrm{vec}(\mathbf{F}_1))\frac{\partial \mathbf{F}_1}{\partial \mathbf{Y}}$
$\mathbf{F}_4^{(i)} = \mathbf{F}_2^{(i)}\mathbf{F}_3$	$\frac{\partial \mathbf{F}_4^{(i)}}{\partial \mathbf{X}} = (\mathbf{F}'_3\otimes\mathbf{I}_r)\frac{\partial \mathbf{F}_2^{(i)}}{\partial \mathbf{X}} + \left(\mathbf{I}_{r^2}\otimes\mathbf{F}_2^{(i)}\right)\frac{\partial \mathbf{F}_3}{\partial \mathbf{X}}$ $\frac{\partial \mathbf{F}_4^{(i)}}{\partial \mathbf{Y}} = (\mathbf{F}'_3\otimes\mathbf{I}_r)\frac{\partial \mathbf{F}_2^{(i)}}{\partial \mathbf{Y}} + \left(\mathbf{I}_{r^2}\otimes\mathbf{F}_2^{(i)}\right)\frac{\partial \mathbf{F}_3}{\partial \mathbf{Y}}$
$\mathbf{F}_5 = \mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{Y}'\mathbf{Y}))$	In Case I ($\mathbf{X}$,$\mathbf{Y}$ constrained to hold unit length columns): $\frac{\partial \mathbf{F}_5}{\partial \mathbf{Y}} = \mathbf{T}_{r^2}\mathbf{E}_r(\mathbf{I}_{r^2} + \mathbf{C}_{rr})(\mathbf{I}_r\otimes\mathbf{Y}')$ In Case II ($\mathbf{X}$,$\mathbf{Y}$ constrained by orthonormality): $\frac{\partial \mathbf{F}_5}{\partial \mathbf{Y}} = \mathbf{0}_{r^4,qr}$
$\mathbf{F}_6^{(i)} = \mathbf{F}_4^{(i)}\mathbf{F}_5$	$\frac{\partial \mathbf{F}_6^{(i)}}{\partial \mathbf{Y}} = (\mathbf{F}'_5\otimes\mathbf{I}_r)\frac{\partial \mathbf{F}_4^{(i)}}{\partial \mathbf{Y}} + \left(\mathbf{I}_{r^2}\otimes\mathbf{F}_4^{(i)}\right)\frac{\partial \mathbf{F}_5}{\partial \mathbf{Y}}$
$\mathbf{F}_7^{(i)} = \mathbf{I}_r\otimes\mathbf{Y}'\mathbf{M}'_i$	$\frac{\partial \mathbf{F}_7^{(i)}}{\partial \mathbf{Y}} = (\mathbf{I}_r\otimes\mathbf{C}_{pr}\otimes\mathbf{I}_r)(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{pr})(\mathbf{M}_i\otimes\mathbf{I}_r)\mathbf{C}_{qr}$
$\mathbf{F}_8^{(i)} = \mathbf{D}_i\otimes\mathbf{I}_p$	$\frac{\partial \mathbf{F}_8^{(i)}}{\partial \mathbf{Y}} = (\mathbf{I}_r\otimes\mathbf{C}_{pr}\otimes\mathbf{I}_p)(\mathbf{I}_{r^2}\otimes\mathrm{vec}(\mathbf{I}_p))\mathbf{T}_r\frac{\partial \mathbf{d}_i}{\partial \mathbf{Y}}$

Appendix C

Notation: $\underline{\mathbf{S}}$ is a $p\times p\times m$ array of symmetric frontal slices $\mathbf{S}_i$ $(i = 1, \ldots, m)$, $\mathbf{X}$ and $\mathbf{Y}$ are $p\times r$ matrices. $\tilde{\mathbf{D}}_i$ is the diagonal matrix defined by Eq. (4.2).

Table C1
Functions $\mathbf{G}_1$ to $\mathbf{G}_{16}$ and some of their partial derivatives.

Function	Derivative
$\mathbf{G}_1^{(i)} = \mathbf{X}'\mathbf{S}_i\mathbf{X}\tilde{\mathbf{D}}_i$	$\frac{\partial \mathbf{G}_1^{(i)}}{\partial \mathbf{X}} = \left(\tilde{\mathbf{D}}_i\mathbf{X}'\otimes\mathbf{I}_r\right)(\mathbf{S}_i\otimes\mathbf{I}_r)\mathbf{C}_{pr} + (\mathbf{I}_r\otimes\mathbf{X}'\mathbf{S}_i)\left(\left(\tilde{\mathbf{D}}_i\otimes\mathbf{I}_p\right) + (\mathbf{I}_r\otimes\mathbf{X})\mathbf{T}_r\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}}\right)$
$\mathbf{G}_2^{(i)} = \tilde{\mathbf{D}}_i\mathbf{X}'\otimes\mathbf{I}_r$	$\frac{\partial \mathbf{G}_2^{(i)}}{\partial \mathbf{X}} = (\mathbf{I}_p\otimes\mathbf{C}_{rr}\otimes\mathbf{I}_r)(\mathbf{I}_{pr}\otimes\mathrm{vec}(\mathbf{I}_r))\cdot\left((\mathbf{X}\otimes\mathbf{I}_r)\mathbf{T}_r\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}} + \left(\mathbf{I}_p\otimes\tilde{\mathbf{D}}_i\right)\mathbf{C}_{pr}\right)$
$\mathbf{G}_3^{(i)} = \mathbf{I}_r\otimes\mathbf{X}'\mathbf{S}_i$	$\frac{\partial \mathbf{G}_3^{(i)}}{\partial \mathbf{X}} = (\mathbf{I}_r\otimes\mathbf{C}_{pr}\otimes\mathbf{I}_r)(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{pr})(\mathbf{S}_i\otimes\mathbf{I}_r)\mathbf{C}_{pr}$
$\mathbf{G}_4^{(i)} = \tilde{\mathbf{D}}_i\otimes\mathbf{I}_p$	$\frac{\partial \mathbf{G}_4^{(i)}}{\partial \mathbf{X}} = (\mathbf{I}_r\otimes\mathbf{C}_{pr}\otimes\mathbf{I}_p)(\mathbf{I}_{r^2}\otimes\mathrm{vec}(\mathbf{I}_p))\mathbf{T}_r\frac{\partial \tilde{\mathbf{d}}_i}{\partial \mathbf{X}}$
$\mathbf{G}_5 = \mathbf{I}_r\otimes\mathbf{X}$	$\frac{\partial \mathbf{G}_5}{\partial \mathbf{X}} = (\mathbf{I}_r\otimes\mathbf{C}_{rr}\otimes\mathbf{I}_p)(\mathrm{vec}(\mathbf{I}_r)\otimes\mathbf{I}_{pr})$
$\mathbf{G}_6^{(i)} = \mathbf{G}_3^{(i)}\mathbf{G}_4^{(i)}$	$\frac{\partial \mathbf{G}_6^{(i)}}{\partial \mathbf{X}} = \left(\left(\mathbf{G}_4^{(i)}\right)'\otimes\mathbf{I}_{r^2}\right)\frac{\partial \mathbf{G}_3^{(i)}}{\partial \mathbf{X}} + \left(\mathbf{I}_{pr}\otimes\mathbf{G}_3^{(i)}\right)\frac{\partial \mathbf{G}_4^{(i)}}{\partial \mathbf{X}}$
$\mathbf{G}_7^{(i)} = \mathbf{G}_3^{(i)}\mathbf{G}_5$	$\frac{\partial \mathbf{G}_7^{(i)}}{\partial \mathbf{X}} = \left(\mathbf{G}'_5\otimes\mathbf{I}_{r^2}\right)\frac{\partial \mathbf{G}_3^{(i)}}{\partial \mathbf{X}} + \left(\mathbf{I}_{r^2}\otimes\mathbf{G}_3^{(i)}\right)\frac{\partial \mathbf{G}_5}{\partial \mathbf{X}}$
$\mathbf{G}_8 = (\mathbf{X}'\mathbf{X}*\mathbf{X}'\mathbf{X})^{-1}$	In Case I ($\mathbf{X}$,$\mathbf{Y}$ constrained to hold unit length columns): $\frac{\partial \mathbf{G}_8}{\partial \mathbf{X}} = -2(\mathbf{G}_8\otimes\mathbf{G}_8)\mathrm{diag}_\mathrm{M}(\mathrm{vec}(\mathbf{X}'\mathbf{X}))\mathbf{E}_r(\mathbf{I}_{r^2} + \mathbf{C}_{rr})\left(\mathbf{I}_r\otimes\mathbf{X}'\right)$ In Case II ($\mathbf{X}$,$\mathbf{Y}$ constrained by orthonormality): $\frac{\partial \mathbf{G}_8}{\partial \mathbf{X}} = \mathbf{0}_{r^2,pr}$
$\mathbf{G}_9^{(i)} = \mathrm{diag}_\mathrm{V}(\mathbf{X}'\mathbf{S}_i\mathbf{X})$	$\frac{\partial \mathbf{G}_9^{(i)}}{\partial \mathbf{X}} = \mathbf{T}'_r\left(\left(\mathbf{X}'\mathbf{S}_i\otimes\mathbf{I}_r\right)\mathbf{C}_{pr} + \left(\mathbf{I}_r\otimes\mathbf{X}'\right)(\mathbf{I}_r\otimes\mathbf{S}_i)\right)$

(continued on next page)

Table C1 (*continued*)

Function	Derivative
$\mathbf{G}_{10}^{(i)} = -((\mathbf{G}_9^{(i)})' \otimes \mathbf{I}_r)$	$\frac{\partial \mathbf{G}_{10}^{(i)}}{\partial \mathbf{X}} = -(\mathbf{I}_r \otimes \text{vec}(\mathbf{I}_r))\mathbf{T}'_r\left(\left(\mathbf{X}'\mathbf{S}_i \otimes \mathbf{I}_r\right)\mathbf{C}_{pr} + \left(\mathbf{I}_r \otimes \mathbf{X}'\right)(\mathbf{I}_r \otimes \mathbf{S}_i)\right)$
$\mathbf{G}_{11} = \mathbf{G}_8 \otimes \mathbf{G}_8$	$\frac{\partial \mathbf{G}_{11}}{\partial \mathbf{X}} = (\mathbf{I}_r \otimes \mathbf{C}_{rr} \otimes \mathbf{I}_r)(\mathbf{I}_{r^4} + \mathbf{C}_{r^2 r^2})(\mathbf{I}_{r^2} \otimes \text{vec}(\mathbf{G}_8))\frac{\partial \mathbf{G}_8}{\partial \mathbf{X}}$
$\mathbf{G}_{12}^{(i)} = \mathbf{G}_{10}^{(i)}\mathbf{G}_{11}$	$\frac{\partial \mathbf{G}_{12}^{(i)}}{\partial \mathbf{X}} = \left(\mathbf{G}'_{11} \otimes \mathbf{I}_r\right)\frac{\partial \mathbf{G}_{10}^{(i)}}{\partial \mathbf{X}} + \left(\mathbf{I}_{r^2} \otimes \mathbf{G}_{10}^{(i)}\right)\frac{\partial \mathbf{G}_{11}}{\partial \mathbf{X}}$
$\mathbf{G}_{13} = \text{diag}_{\text{M}}(\text{vec}(\mathbf{X}'\mathbf{X}))$	In Case I (**X**,**Y** constrained to hold unit length columns): $\frac{\partial \mathbf{G}_{13}}{\partial \mathbf{X}} = \mathbf{T}_{r^2}\mathbf{E}_r(\mathbf{I}_{r^2} + \mathbf{C}_{rr})(\mathbf{I}_r \otimes \mathbf{X}')$ In Case II (**X**,**Y** constrained by orthonormality): $\frac{\partial \mathbf{G}_{13}}{\partial \mathbf{X}} = \mathbf{0}_{r^4,pr}$
$\mathbf{G}_{14}^{(i)} = \mathbf{G}_{12}^{(i)}\mathbf{G}_{13}$	$\frac{\partial \mathbf{G}_{14}^{(i)}}{\partial \mathbf{X}} = (\mathbf{G}'_{13} \otimes \mathbf{I}_r)\frac{\partial \mathbf{G}_{12}^{(i)}}{\partial \mathbf{X}} + \left(\mathbf{I}_{r^2} \otimes \mathbf{G}_{12}^{(i)}\right)\frac{\partial \mathbf{G}_{13}}{\partial \mathbf{X}}$
$\mathbf{G}_{15}^{(i)} = \mathbf{X}'\mathbf{S}_i \otimes \mathbf{I}_r$	$\frac{\partial \mathbf{G}_{15}^{(i)}}{\partial \mathbf{X}} = \left(\mathbf{I}_p \otimes \mathbf{C}_{rr} \otimes \mathbf{I}_r\right)\left(\mathbf{I}_{pr} \otimes \text{vec}(\mathbf{I}_r)\right)(\mathbf{S}_i \otimes \mathbf{I}_r)\mathbf{C}_{pr}$
$\mathbf{G}_{16}^{(i)} = \mathbf{G}_{15}^{(i)}\mathbf{C}_{pr} + \mathbf{G}'_5(\mathbf{I}_r \otimes \mathbf{S}_i)$	$\frac{\partial \mathbf{G}_{16}^{(i)}}{\partial \mathbf{X}} = \left(\mathbf{C}'_{pr} \otimes \mathbf{I}_{r^2}\right)\frac{\partial \mathbf{G}_{15}^{(i)}}{\partial \mathbf{X}} + (\mathbf{I}_r \otimes \mathbf{S}_i \otimes \mathbf{I}_{r^2})\mathbf{C}_{pr,r^2}\frac{\partial \mathbf{G}_5}{\partial \mathbf{X}}$

Table C1 summarizes the expressions of functions $\mathbf{G}_1 - \mathbf{G}_{16}$, with the relevant partial derivatives.

References

[1] M. Bennani Dosse, J.M.F. Ten Berge, The assumption of proportional components when Candecomp is applied to symmetric matrices in the context of Indscal, Psychometrika 73 (2008) 303–307.

[2] M. Bennani Dosse, J.M.F. Ten Berge, J. Tendeiro, Some new results on orthogonally constrained Candecomp, submitted for publication.

[3] J.D. Carroll, J.J. Chang, Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart–Young decomposition, Psychometrika 35 (1970) 283–319.

[4] P.L. Fackler, Notes on Matrix Calculus, , 2009 Retrieved on 3 February 2009 from http://www4.ncsu.edu/~pfackler/MatCalc.pdf.

[5] R.A. Harshman, Foundations of the Parafac procedure: models and conditions for an "explanatory" multimodal factor analysis, UCLA Working Papers in Phonetics, 16, 1970, pp. 1–84.

[6] R.A. Harshman, M.E. Lundy, Data preprocessing and the extended PARAFAC model, in: H.G. Law, C.W. Snyder, J.A. Hattie, R.P. McDonald (Eds.), Research Methods for Multimode Data Analysis, Praeger, New York, 1984, pp. 216–284.

[7] J.B. Kruskal, Three-way arrays: rank and uniqueness of trilinear decompositions, with applications to arithmetic complexity and statistics, Linear Algebra and its Applications 18 (1977) 95–138.

[8] J.B. Kruskal, R.A. Harshman, M.E. Lundy, Some relationships between Tucker's three-mode factor analysis and PARAFAC/CANDECOMP, Paper Presented at the Annual Meeting of the Psychometric Society, Los Angeles, 1983.

[9] J.B. Kruskal, R.A. Harshman, M.E. Lundy, Several mathematical relationships between PARAFAC-CANDECOMP and three-mode factor analysis, Paper Presented at the Annual Meeting of the Classification Society, St. John's, Newfoundland, 1985.

[10] J.R. Magnus, H. Neudecker, Matrix Differential Calculus with Applications in Statistics and Econometrics, 3rd EditionJohn Wiley & Sons, Chichester/New York, 2007.

[11] P. Paatero, A weighted non-negative least squares algorithm for three-way 'PARAFAC' factor analysis, Chemometrics and Intelligent Laboratory systems 38 (1997) 223–242.

[12] P. Paatero, The multilinear engine — a table-driven, least squares program for solving multilinear problems, including the n-way parallel factor analysis model, Journal of Computational and Graphical Statistics 8 (1999) 854–888.

[13] A. Stegeman, Degeneracy in Candecomp/Parafac explained for $p \times p \times 2$ arrays of rank $p+1$ or higher, Psychometrika 71 (2006) 483–501.

[14] A. Stegeman, Degeneracy in Candecomp/Parafac and Indscal explained for several three-sliced arrays with a two-valued typical rank, Psychometrika 72 (2007) 601–619.

[15] J.M.F. Ten Berge, A joint treatment of Varimax rotation and the problem of diagonalizing symmetric matrices simultaneously in the least-squares sense, Psychometrika 49 (1984) 347–358.

[16] J.M.F. Ten Berge, H.A.L. Kiers, Some clarifications of the Candecomp algorithm applied to Indscal, Psychometrika 56 (1991) 317–326.

[17] J.M.F. Ten Berge, H.A.L. Kiers, J. De Leeuw, Explicit CANDECOMP/PARAFAC solutions for a contrived $2 \times 2 \times 2$ array of rank three, Psychometrika 53 (1988) 579–583.

[18] J.M.F. Ten Berge, D.L. Knol, H.A.L. Kiers, A treatment of the Orthomax rotation family in terms of diagonalization, and a re-examination of a singular value approach to Varimax rotation, Computational Statistics Quarterly 3 (1988) 207–217.