Computational Statistics and Data Analysis 98 (2016) 1–18

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

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

Sparse Tucker2 analysis of three-way data subject to a constrained number of zero elements in a core array

Hiroki Ikemoto *, Kohei Adachi

Osaka University, Japan

ARTICLE INFO

Article history:
Received 13 March 2015
Received in revised form 1 December 2015
Accepted 7 December 2015
Available online 22 December 2015

Keywords:
Three-way principal component analysis
Tucker2
Parafac
Sparse principal component analysis
Sparse core arrays

ABSTRACT

Three-way principal component analysis (3WPCA) models have been developed for analyzing a three-way data array of objects × variables × sources. Among the 3WPCA models, the least restrictive is the Tucker2 model, in which an extended core array describes the source-specific relationships between the components underlying objects and those for variables. In contrast to Tucker2 with the core array unconstrained, the Parafac model is highly restrictive in that the core slices in the array are constrained to be diagonal matrices. In this paper, we propose a procedure by which a suitably constrained intermediate model between Tucker2 and Parafac can be found. In the proposed procedure, the Tucker2 loss function is minimized subject to a specified number of core elements being zero with their locations unknown; the optimal locations of zero elements and nonzero parameter values are simultaneously estimated. This technique can be called *sparse core Tucker2* (ScTucker2), as the matrices including a number of zeros are said to be sparse. We present an alternating least squares algorithm for ScTucker2 with a procedure for selecting a suitable number of zero elements. This procedure is assessed in a simulation study and illustrated with real data sets.

1. Introduction

A time-honored dimension-reduction method for a two-way data matrix is principal component analysis (PCA), which has been extended for analyzing a three-way data array $\{x_{ijk};\ i = 1, \ldots, I;\ j = 1, \ldots, J;\ k = 1, \ldots, K\}$, i.e., an I-objects $\times$ J-variables $\times$ K-sources array of x_{ijk}. Here, the terms "objects", "variables", and "sources" are used merely for the sake of distinguishing the three-modes in the array. Such extended procedures are generally called three-way PCA (3WPCA). The most popular 3WPCA procedures are Tucker2, Tucker3, and Parafac; the former two were originally proposed by Tucker (1966) and the latter was proposed by Harshman (1970). Among those procedures, Tucker2 is the most comprehensive, in that the other procedures are derived by imposing constraints in Tucker2 (Kroonenberg, 2008; Smilde et al., 2004): Tucker3 is a constrained version of Tucker2 (Kroonenberg, 1983; Murakami and Kroonenberg, 2003) and Parafac is a constrained one of Tucker3 (Kiers, 1991).

Using $\mathbf{X}_k = (x_{ijk})$ for the I-objects $\times$ J-variables data matrix for source k whose (i, j) element is x_{ijk}, the Tucker2 model can be expressed as

$$\mathbf{X}_k = \mathbf{A}\mathbf{H}_k\mathbf{B}' + \mathbf{E}_k, \tag{1}$$

* Correspondence to: Graduate School of Human Sciences, Osaka University, 1-2 Yamadaoka, Suita, Osaka 565-0871, Japan.
E-mail address: ikemoto@bm.hus.osaka-u.ac.jp (H. Ikemoto).

http://dx.doi.org/10.1016/j.csda.2015.12.007
0167-9473/

with $\mathbf{E}_k$ an error matrix (Tucker, 1966). The parameter matrices to be obtained in (1) are an I-objects $\times$ P-components matrix $\mathbf{A}$, a J-variables $\times$ Q-components matrix $\mathbf{B}$, and $P \times Q$ matrices $\mathbf{H}_k$, $k = 1, \ldots, K$, where $\mathbf{A}$ and $\mathbf{B}$ are constrained as

$$\mathbf{A}'\mathbf{A} = \mathbf{I}_p \quad \text{and} \quad \mathbf{B}'\mathbf{B} = \mathbf{I}_Q, \tag{2}$$

with $\mathbf{I}_P$ the $P \times P$ identity matrix. In the 3WPCA literature, $\mathbf{A}$ and $\mathbf{B}$ are called component matrices, while $\mathbf{H}_k$ ($P \times Q$) is called a core slice, and the $P \times QK$ block matrix $\mathbf{H} = [\mathbf{H}_1, \ \ldots, \ \mathbf{H}_K]$ is referred to as an extended core array (Kroonenberg, 2008; Smilde et al., 2004). In this paper, $\mathbf{H}$ is simply called a core array. The main features of the Tucker2 model (1) are summarized as follows: [1] $\mathbf{A}$ and $\mathbf{B}$ do not have a subscript k, but it is attached to $\mathbf{H}_k$, which implies that $\mathbf{A}$ and $\mathbf{B}$ are invariant across sources $k = 1, \ldots, K$, while $\mathbf{H} = [\mathbf{H}_1, \ \ldots, \ \mathbf{H}_K]$ serves to explain the differences in $\mathbf{X}_k$ across sources; and [2] each core slice $\mathbf{H}_k$ describes the relationships between the P components for objects (the columns in $\mathbf{A}$) and the Q components for variables (in $\mathbf{B}$) for source k (Adachi, 2011; Murakami and Kroonenberg, 2003).

In contrast to Tucker2, the most restrictive method among the popular 3WPCA procedures is Parafac, in which $\mathbf{H}_k$ is constrained to be a $P \times P$ diagonal matrix (Harshman, 1970). If $\mathbf{H}_k$ remains to be of $P \times Q$ and $\mathbf{D}_k$ is used for the diagonal matrix, the Parafac model is expressed as (1) with

$$\mathbf{H}_k = [\mathbf{D}_k, \ \mathbf{O}] \quad \text{for } P < Q, \qquad \mathbf{H}_k = [\mathbf{D}_k, \ \mathbf{O}]' \quad \text{for } P > Q, \quad \text{and} \quad \mathbf{H}_k = \mathbf{D}_k \quad \text{for } P = Q. \tag{3}$$

Here, $\mathbf{O}$ is the matrix of zeros by which $\mathbf{H}_k$ is completed. We can view the Tucker2 and Parafac models as two extremes in the sparseness of the core array $\mathbf{H} = [\mathbf{H}_1, \ldots, \mathbf{H}_K]$, where the sparseness refers to the number of zero elements in a matrix. In the Parafac model, $\mathbf{H}$ is very sparse; its $\mathbf{H}_k$ block is filled with zeros, except for the diagonal elements. On the other hand, $\mathbf{H} = [\mathbf{H}_1, \ldots, \mathbf{H}_K]$ is unconstrained in Tucker2, thus the elements of $\mathbf{H}$ are typically nonzero, i.e., not sparse.

The contrast in the sparseness of the core array shows that the unconstrained Tucker2 model fits a data set better than the Parafac one, while the array in Parafac might be too sparse for its model to fit a data set well. However, the sparser Parafac model is advantageous in model parsimony and interpretability; the number of parameters to be estimated is smaller and the core slice $\mathbf{H}_k$ can be interpreted only by focusing on its diagonal elements. Those arguments suggest that it is desirable for the core array $\mathbf{H}$ to have moderate sparseness. We thus propose a modified Tucker2 procedure with a sparseness constraint for $\mathbf{H}$, that is, a procedure for obtaining the optimal $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{H} = [\mathbf{H}_1, \ \ldots, \ \mathbf{H}_K]$ in (1) subject to (2), with the additional constraint

$$\mathrm{Sp}(\mathbf{H}) = c. \tag{4}$$

Here, $\mathrm{Sp}(\mathbf{H})$ denotes the number of zero elements in $\mathbf{H}$ and c is a specified integer. We further consider a strategy for selecting a suitable sparseness c. This procedure is referred to as *sparse core Tucker2* (ScTucker2).

In order to introduce ScTucker2, we have so far contrasted Tucker2 with Parafac, whose core array is structured as (3). However, ScTucker2 can provide any type of sparse core array without a specific structure, since only the number of zero elements in $\mathbf{H}$ is constrained with their locations unknown (to be estimated). For example, it may give the core array

$$\mathbf{H} = [\mathbf{H}_1, \ \mathbf{H}_2] = \left[\begin{array}{ccc|ccc} \# & 0 & 0 & 0 & 0 & \# \\ 0 & \# & 0 & \# & 0 & \# \\ 0 & 0 & \# & 0 & \# & 0 \end{array}\right] \tag{5}$$

for $\mathrm{Sp}(\mathbf{H}) = 11$ and $P = Q = 3$, where # indicates a nonzero element. As the core slice $\mathbf{H}_2$ is not a diagonal matrix, (5) cannot be located anywhere between Tucker2 and Parafac. Moreover, ScTucker2 may produce a core array sparser than that with Parafac, for example,

$$\mathbf{H} = [\mathbf{H}_1, \ \mathbf{H}_2] = \left[\begin{array}{ccc|ccc} \# & 0 & 0 & 0 & \# & 0 \\ 0 & \# & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \# \end{array}\right], \tag{6}$$

under a stronger sparseness constraint. Examples (5) and (6) illustrate that ScTucker2 can be viewed as a procedure for selecting a model without a special name such as Parafac.

A benefit of sparse core arrays, such as (5) and (6), is that we may only focus on a few nonzero elements for interpreting $\mathbf{H}$. Here, it should be noted that some elements of the core array in the ordinary Tucker2 can also be close to zero by applying rotation techniques (Kroonenberg, 2008; Smilde et al., 2004) to Tucker2 solutions, as they have rotational freedom. However, the rotation techniques have drawbacks, namely, that the elements in the resulting core array cannot equal exact zero, and the rotation is made without fitting a model to data, which is discussed in more detail and illustrated in Sections 6 and 7.

Although a sparse version of Tucker2 has never been considered so far, to the best of our knowledge, the standard PCA for two-way data has already been modified so as to give sparse solutions (Jolliffe et al., 2003; Zou et al., 2006; Trendafilov and Adachi, 2015). Such modified PCA procedures are generally called sparse PCA. The prevailing approach involves using penalty functions, i.e., combining the original PCA objective function with additional functions that penalize solutions for not being sparse (Trendafilov, 2014). A different approach without the use of a penalty function has been presented by Adachi and Trendafilov (in press). In their approach, the PCA objective function is optimized subject to the sparseness of the solution directly fixed to an integer, as in (4). We also use the latter approach in ScTucker2.

ScTucker2 can be related to Kiers' (1992) approach in which some elements of core array $\mathbf{H}$ are constrained to be zeros. But, the locations of zero elements are fixed, which differs from ScTucker2 with those locations estimated optimally.

However, Kiers' (1992) procedure is less restrictive in that the column-orthogonality constraint (2) is not imposed on component matrices. Whether or not constraint (2) should be imposed is an open question. Imposing it detracts from the goodness of fit of the model, but, on the other hand, it helps interpretation of the components as independent entities; and it allows for simple interpretation of fit contributions by the components. For these reasons, we imposed this constraint in ScTucker2.

The remaining parts of this paper are organized as follows: In Section 2, ScTucker2 is formulated in more detail, whose algorithm is then presented in Section 3. The methodology for selecting a suitable sparseness is described in Section 4. A simulation study for assessing the behaviors of ScTucker2 is reported in Section 5. ScTucker2 is compared with Tucker2 followed by rotation in Section 6, and is illustrated using real data examples in Section 7.

2. Sparse core Tucker2

The Tucker2 model (1) is rewritten as

$$\mathbf{X} = \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}') + \mathbf{E} \tag{1'}$$

with $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_K]$ $(I \times JK)$, $\mathbf{E} = [\mathbf{E}_1, \ldots, \mathbf{E}_K]$ $(I \times JK)$, and $\otimes$ denoting the Kronecker product (e.g., Harville, 1997). In the ordinary Tucker2, the least squares function is defined as the sum of squares of the errors in (1), as follows:

$$f = \|\mathbf{E}\|^2 = \|\mathbf{X} - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2, \tag{7}$$

with $\| \|^2$ denoting the squared Frobenius norm. Also in our proposed ScTucker2, the loss function is defined as (7), but the sparseness constraint (4) is imposed on the core array $\mathbf{H}$. ScTucker2 is thus formulated as minimizing (7) subject to (4) and the normalization condition (2):

$$\min_{\mathbf{A},\mathbf{B},\mathbf{H}} f = \|\mathbf{X} - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 \quad \text{subject to } \mathrm{Sp}(\mathbf{H}) = c, \qquad \mathbf{A}'\mathbf{A} = \mathbf{I}_P, \qquad \mathbf{B}'\mathbf{B} = \mathbf{I}_Q. \tag{8}$$

Using constraint (2), we can decompose the loss function (7) as

$$f = \|\mathbf{X} - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 = \|\mathbf{X} - \mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 + \|\mathbf{G} - \mathbf{H}\|^2, \tag{9}$$

with $\mathbf{G} = [\mathbf{G}_1, \ \ldots, \ \mathbf{G}_K]$ the $P \times QK$ block matrix whose kth block is the $P \times Q$ matrix defined as

$$\mathbf{G}_k = \mathbf{A}'\mathbf{X}_k\mathbf{B}, \quad \text{or equivalently,} \qquad \mathbf{G} = \mathbf{A}'\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B}). \tag{10}$$

The decomposition (9) is derived by rewriting (7) as

$$\begin{aligned} f &= \|\mathbf{X} - \mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}') + \mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}') - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 \\ &= \|\mathbf{X} - \mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 - 2\eta + \phi. \end{aligned} \tag{7'}$$

Here,

$$\eta = \mathrm{tr}\{\mathbf{X} - \mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}')\}'\{\mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}') - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\} \tag{11}$$

is found to be zero, as shown in Appendix A, and

$$\phi = \|\mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}') - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 = \|\mathbf{A}(\mathbf{G} - \mathbf{H})(\mathbf{I}_K \otimes \mathbf{B}')\|^2 = \|\mathbf{G} - \mathbf{H}\|^2 \tag{12}$$

can be derived using (2) and $(\mathbf{I}_K \otimes \mathbf{B}')(\mathbf{I}_K \otimes \mathbf{B}')' = \mathbf{I}_K \otimes (\mathbf{B}'\mathbf{B})$. The decomposition (9) shows that only the term $\phi = \phi(\mathbf{H}) = \|\mathbf{G} - \mathbf{H}\|^2$ is relevant to $\mathbf{H}$, which allows us to easily obtain the sparseness-constrained $\mathbf{H}$ that minimizes (7), as described in Section 3.

The advantage of imposing constraint (2) for ScTucker2 is shown by the fact that the decomposition of the loss function without using (2) does not give a simple function of $\mathbf{H}$ as (12) in which $\mathbf{H}$ is simply matched with $\mathbf{G}$. That is, even if (2) was not supposed, the loss function (7) can be decomposed as $f = \|\mathbf{X} - \mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 + \phi^{\#}$ with $\mathbf{G}^{\#} = \mathbf{A}^{+}\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B}'^{+})$ defined using $\mathbf{A}^{+}$ for the Moore–Penrose inverse of $\mathbf{A}$ and

$$\phi^{\#} = \|\mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}') - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2, \tag{13}$$

whose derivation is detailed in Appendix B. However, without constraint (2), $\mathbf{A}$ and $(\mathbf{I}_K \otimes \mathbf{B}')$ by which $\mathbf{H}$ is multiplied in (13) does not vanish, and (13) cannot be minimized over $\mathbf{H}$ subject to (4) in the manner to be shown in Section 3.

3. Algorithm

In order to obtain the solution for the ScTucker2 problem (8), we present an alternating least squares (ALS) algorithm in which the initialization of $\mathbf{A}$ and $\mathbf{B}$ is followed by alternately iterating the following steps:

[A-step] (7) is minimized over $\mathbf{A}$ subject to $\mathbf{A}'\mathbf{A} = \mathbf{I}_P$, while $\mathbf{H}$ and $\mathbf{B}$ are kept fixed.
[B-step] (7) is minimized over $\mathbf{B}$ subject to $\mathbf{B}'\mathbf{B} = \mathbf{I}_Q$, with $\mathbf{H}$ and $\mathbf{A}$ fixed.
[H-step] (7) is minimized over $\mathbf{H}$ subject to (4), with $\mathbf{A}$ and $\mathbf{B}$ kept fixed.

The minimizations in the A- and B-steps are attained in the same ways as those in the ordinary Tucker2 algorithm (Smilde et al., 2004, pp. 123–124). That is, the optimal $\mathbf{A}$ and $\mathbf{B}$ in those steps are given by

$$\mathbf{A} = \mathbf{K}\mathbf{L}', \tag{14}$$

$$\mathbf{B} = \mathbf{U}\mathbf{V}'. \tag{15}$$

Here, $\mathbf{K}$ and $\mathbf{L}$ are given through the singular value decomposition (SVD) of $\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B})\mathbf{H}'$, defined as $\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B})\mathbf{H}' = \mathbf{K}\boldsymbol{\Lambda}\mathbf{L}'$, with $\mathbf{K}'\mathbf{K} = \mathbf{L}'\mathbf{L} = \mathbf{I}_P$ and $\boldsymbol{\Lambda}$ a diagonal matrix, while $\mathbf{U}$ and $\mathbf{V}$ are obtained through the SVD of $\tilde{\mathbf{X}}(\mathbf{I}_K \otimes \mathbf{A})\tilde{\mathbf{H}}' = \mathbf{U}\boldsymbol{\Delta}\mathbf{V}'$, with $\tilde{\mathbf{X}} = [\mathbf{X}_1', \ldots, \mathbf{X}_K']$ $(J \times IK)$ and $\tilde{\mathbf{H}} = [\mathbf{H}_1', \ldots, \mathbf{H}_K']$ $(Q \times PK)$, $\mathbf{U}'\mathbf{U} = \mathbf{V}'\mathbf{V} = \mathbf{I}_Q$, and $\boldsymbol{\Delta}$ a diagonal matrix. The above solutions are derived using the fact that (7) is expanded as

$$f = \|\mathbf{X}\|^2 + \|\mathbf{H}\|^2 - 2\operatorname{tr}\mathbf{A}'\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B})\mathbf{H}' = \|\mathbf{X}\|^2 + \|\mathbf{H}\|^2 - 2\rho_A(\mathbf{A}), \tag{16}$$

and also as $\|\tilde{\mathbf{X}}\|^2 + \|\tilde{\mathbf{H}}\|^2 - 2\rho_B(\mathbf{B})$ with $\rho_A(\mathbf{A}) = \operatorname{tr}\mathbf{A}'\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B})\mathbf{H}'$ and $\rho_B(\mathbf{B}) = \operatorname{tr}\mathbf{B}'\tilde{\mathbf{X}}(\mathbf{I}_K \otimes \mathbf{A})\tilde{\mathbf{H}}'$; minimizing (7) is equivalent to the maximization of $\rho_A(\mathbf{A})$ and that of $\rho_B(\mathbf{B})$ subject to (2). Those are found to be attained for (14) and (15) using the theorems of ten Berge (1983, 1993).

The decomposition (9) shows that the H-step is attained by minimizing $\phi(\mathbf{H}) = \|\mathbf{G} - \mathbf{H}\|^2$ over $\mathbf{H}$ for fixed $\mathbf{A}$ and $\mathbf{B}$. Using $\mathbf{G} = (g_{pr})$ and $\mathbf{H} = (h_{pr})$ $(p = 1, \ldots, P;\ r = 1, \ \ldots, QK)$, the function $\phi(\mathbf{H})$ is rewritten as

$$\phi(\mathbf{H}) = \|\mathbf{G} - \mathbf{H}\|^2 = \sum_{(p,r)\in\mathbf{O}} g_{pr}^2 + \sum_{(p,r)\in\mathbf{O}^{\perp}} (g_{pr} - h_{pr})^2 \geq \sum_{(p,r)\in\mathbf{O}} g_{pr}^2. \tag{17}$$

Here, $\mathbf{O}$ denotes the set of c indices (p, r) indicating the locations of the elements h_{pr} to be zero. The complement set $\mathbf{O}^{\perp}$ is the set containing the $PQK - c$ indices (p, r) indicating the locations of the non-zero h_{pr}. The inequality (17) shows that $\phi(\mathbf{H})$ attains its lower limit $\sum_{(p,r)\in\mathbf{O}} g_{pr}^2$ when the non-zero elements h_{pr} with $(p,\ r) \in \mathbf{O}^{\perp}$ are taken as equal to the corresponding g_{pr}. Moreover, the limit $\sum_{(p,r)\in\mathbf{O}} g_{pr}^2$ is minimal when $\mathbf{O}$ contains the indices for the c smallest g_{pr}^2 among all squared elements of $\mathbf{G}$. Thus, the optimal $\mathbf{H} = (h_{pr})$ in the H-step is given by

$$h_{pr} = \begin{cases} 0 & \text{iff } g_{pr}^2 \leq g_{[c]}^2 \\ g_{pr} & \text{otherwise,} \end{cases} \tag{18}$$

with $g_{[c]}^2$ the cth smallest value among all g_{pr}^2.

It should be noted that the core $\mathbf{H}$ resulting in (18) satisfies $\mathbf{H}{\bullet}\mathbf{H} = \mathbf{G}{\bullet}\mathbf{H}$, with $\bullet$ denoting the Hadamard product (e.g., Harville, 1997). Using this equation and (10) in (16), we can find that the value of the loss function (7) after the H-step can be expressed as

$$f = \|\mathbf{X}\|^2 + \|\mathbf{H}\|^2 - 2\operatorname{tr}\mathbf{G}\mathbf{H}' = \|\mathbf{X}\|^2 - \|\mathbf{H}\|^2. \tag{19}$$

Its division by $\|\mathbf{X}\|^2$ gives

$$f_s = \frac{f}{\|\mathbf{X}\|^2} = 1 - \frac{\|\mathbf{H}\|^2}{\|\mathbf{X}\|^2}, \tag{20}$$

which is standardized so as to take a value within the range [0, 1]. This measure is convenient for testing convergence, which suggests that it should be tested immediately after the H-step.

The algorithm for ScTucker2 is thus listed as follows:

[1] Initialize $\mathbf{A}$ and $\mathbf{B}$.
[2] Update $\mathbf{H}$ with (18).
[3] Finish if convergence is reached; otherwise, go to [4].
[4] Update $\mathbf{A}$ with (14).
[5] Update $\mathbf{B}$ with (15) and go back to [2].

In Step [2], the convergence is defined as having a decrease in (20) from the previous round $(fs^{[t-1]} - fs^{[t]})$ that is less than 10^{-8}. The initialization in Step [1] is performed using the matrices $\mathbf{A}_0$ $(I \times \ P)$ and $\mathbf{B}_0$ $(J \times Q)$ whose elements are drawn from $U(-1, 1)$, with $U(\alpha,\ \beta)$ the uniform distribution over the range $[\alpha,\ \beta]$. The left singular vectors of $\mathbf{A}_0$ and $\mathbf{B}_0$ give the initial matrices for $\mathbf{A}$ and $\mathbf{B}$, respectively, with their satisfying (2). In order to avoid selecting a local minimizer as a solution, we run the above algorithm 200 times to choose the solution with the lowest loss function value as the optimal one.

4. Selection of sparseness

In this section, we consider how to choose the sparseness c in constraint (4). It can be viewed as a problem of selecting a model with suitable sparseness and good fitting for a data set. Between them, the following trade-off relationship exists: one minus (20) leads to the goodness-of-fit (GOF) index

$$GOF(c) = 1 - f_s = \frac{\|\mathbf{H}\|^2}{\|\mathbf{X}\|^2}, \tag{21}$$

which is a function of the sparseness c. With the decrease in c, (21) increases, and finally reaches the upper limit $GOF(0)$, which is the GOF of the ordinary Tucker2 solution with $c = 0$. We thus need a procedure for finding the best trade-off between GOF and sparseness. This can be attained using indices called information criteria, of which AIC and BIC are popular ones (Akaike, 1974; Schwarz, 1978). In those criteria, both GOF and the number of parameters to be estimated are considered, where the latter number is directly related to the sparseness c, since the c zero elements in $\mathbf{H}$ are excluded from those parameters whose values must be estimated.

The information criteria are defined with a maximum likelihood estimate (MLE). Though a ScTucker2 solution is a least squares estimate (LSE), it is also a MLE under the assumption that each element of $\mathbf{E}_k = (e_{ijk})$ is identically and independently distributed according to the normal distribution whose average is zero and whose variance equals σ^2:

$$e_{ijk} \sim N(0, \sigma^2). \tag{22}$$

This equivalence of LSE to MLE is proved in Appendix C with the maximum log-likelihood being given by

$$\log L_{\max} = -\frac{IJK}{2}\{\log[1 - GOF(c)] + const\}. \tag{23}$$

Here, const $= \log\|\mathbf{X}\|^2 + \log 2\pi - \log IJK + 1$ is a constant irrelevant to parameters.

Using (23), AIC and BIC are defined as functions of c:

$$AIC(c) = -2\log L_{\max} + 2\theta = -IJK\{\log[1 - GOF(c)] + const\} + 2(N - c), \tag{24}$$

$$BIC(c) = -2\log L_{\max} + \theta\log(IJK) = -IJK\{\log[1 - GOF(c)] + const\} + (N - c)\log(IJK), \tag{25}$$

where θ denotes the number of parameters to be estimated and $N = IP + JQ + KPQ$ is the total number of elements in $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{H}$, thus, $\theta = N - c$. A lower value of AIC/BIC indicates a better solution. In (24) and (25), we can find that those indices take lower values when $GOF(c)$ and c are greater. The optimal sparseness $\hat{c}$ is thus given by

$$\hat{c} = \underset{c_{\min} \le c \le c_{\max}}{\arg\min}\ BIC(c) \tag{26}$$

if BIC is used; otherwise $BIC(c)$ is replaced by $AIC(c)$ in (26). Here, the lower limit of c is $c_{\min} = P_0 + Q_0$ rather than zero, with $P_0 = P(P-1)/2$ and $Q_0 = Q(Q-1)/2$, since ScTucker2 solutions with $0 \le c \le c_{\min}$ can be regarded as equivalent to the Tucker2 solution with $c = 0$, for the reason described in Appendix D. On the other hand, the upper limit of c is chosen as $c_{\max} = KPQ - \max(P, Q)$ so that at least $\max(P, Q)$ elements of $\mathbf{H}$ are nonzero. Otherwise, at least one component in $\mathbf{A}$ or $\mathbf{B}$ disappears in the model part in the loss function (7), as explained in Appendix E.

5. Simulation study

We performed a simulation study in order to examine [1] how often the best solution is obtained, [2] how well parameter values are recovered by the ScTucker2 algorithm with c set to true values, and [3] how accurately the true sparseness c is identified by AIC/BIC. Their results are reported in Section 5.2, after the procedures for synthesizing and analyzing data sets are described in Section 5.1.

5.1. Procedures

With $I = J = K = 12$ and $P = Q = 3$, we generated a data matrix $\mathbf{X}$ according to the Tucker2 model (1) subject to (2) and (4). The true parameter and error matrices in those equations are synthesized as follows:

[1] Each element of $\mathbf{A}$ and $\mathbf{B}$ is drawn from $U(-1, 1)$ and the resulting $\mathbf{A}$ and $\mathbf{B}$ are transformed so as to satisfy (2), with $U(\alpha, \beta)$ the uniform distribution over the range $[\alpha, \beta]$.

[2] The true sparseness c is randomly set to an integer within the range $[0.5KPQ, 0.8KPQ]$, and the locations of c zero elements in $\mathbf{H}$ are randomly chosen subject to core slice $\mathbf{H}_k$ not being filled with zeros. The value of each nonzero element in $\mathbf{H}$ is drawn randomly from $U(0.5, 1)$ or $U(-1, -0.5)$.

[3] The error matrix $\mathbf{E}$ is filled with the standard normal variables multiplied by the constant that equalizes $\|\mathbf{E}\|^2/\|\mathbf{X}\|^2$ to 0.4 (a higher error level), 0.2 (a lower level) or 0.0 (an error-free setting).

We replicated the above procedures to have 100 $\mathbf{X}$'s.

Table 1
Statistics for the frequency of the best solution found during 200 runs.

Error	Error-free		Lower		Higher	
Index	AF_{BS}	$F_{\mathrm{BS}}^{\mathrm{true}}$	AF_{BS}	$F_{\mathrm{BS}}^{\mathrm{true}}$	AF_{BS}	$F_{\mathrm{BS}}^{\mathrm{true}}$
5%	45.1	12.0	42.7	12.0	42.8	6.0
25%	51.8	32.8	48.2	30.0	46.3	17.0
50%	72.7	71.0	63.7	72.0	57.7	50.0
75%	114.0	130.0	78.1	137.0	79.2	134.5
95%	146.9	183.1	107.7	182.1	91.2	172.5
Average	83.9	81.7	66.7	82.4	63.2	75.8

Table 2
Statistics for the differences between true and resulting parameters.

Error level	Index	MR_0	$\mathrm{MR}_\#$	AAD		
	Matrix	**H**	**H**	**H**	**A**	**B**
Error-Free	50%	0.000	0.000	0.000	0.000	0.000
	75%	0.000	0.000	0.000	0.000	0.000
	95%	0.000	0.000	0.000	0.000	0.000
	Average	0.000	0.000	0.000	0.000	0.000
Lower	50%	0.000	0.000	0.017	0.016	0.016
	75%	0.000	0.000	0.023	0.018	0.018
	95%	0.000	0.000	0.030	0.020	0.020
	Average	0.002	0.002	0.019	0.017	0.017
Higher	50%	0.000	0.000	0.029	0.026	0.028
	75%	0.000	0.000	0.043	0.029	0.030
	95%	0.018	0.023	0.055	0.035	0.036
	Average	0.005	0.007	0.033	0.027	0.030

For each data set, we set P and Q at the true values to perform ScTucker2 with the sparseness selection procedure in Section 4. As described in Section 3, the algorithm for ScTucker2 was run 200 times. We use $\mathbf{A}_{cu}$, $\mathbf{B}_{cu}$, and $\mathbf{H}_{cu}$ for the solutions of **A**, **B**, and **H** resulting from the uth run for a specific value of $\mathrm{Sp}(\mathbf{H}) = c$. That is, $\mathbf{A}_{cu}$, $\mathbf{B}_{cu}$, and $\mathbf{H}_{cu}$ $(c = c_{\min}, \ldots, c_{\max}; u = 1, \ldots, 200)$ were obtained for each data set. We express the optimal solution of **H** for a specific c as $\mathbf{H}_c = \mathrm{argmin}_{1\le u\le 200} f_s(\mathbf{H}_{cu})$ and the corresponding **A** and **B** as $\mathbf{A}_c$ and $\mathbf{B}_c$, respectively, where $f_s(\mathbf{H}_{cu})$ is the value of (20) for $\mathbf{H} = \mathbf{H}_{cu}$.

5.2. Results

In order to study how often the best solution is obtained out of 200 runs, we defined $\mathbf{H}_{cu}$ being the best as $f_s(\mathbf{H}_{cu}) - f_s(\mathbf{H}_c) \le 0.001$, with $0 \le f_s(\mathbf{H}) \le 1$, on the supposition that $f_s(\mathbf{H}_c)$ is the global minimum. For a specific c value, we counted the frequency F_c with which the best solution is obtained during the 200 runs, then obtained the averaged frequency of the best solution, $AF_{\mathrm{BS}} = \sum_{c=c_{\min}}^{c_{\max}} F_c/\Delta_c$, in the Δ_c solutions for each data set, where $\Delta_c = c_{\max} - c_{\min} + 1$. We further obtained the frequency $F_{\mathrm{BS}}^{\mathrm{true}} = F_c$ when c was the true sparseness. The percentiles and averages of AF_{BS} and $F_{\mathrm{BS}}^{\mathrm{true}}$ over 100 data sets are shown in Table 1. There, we can find that their averages indicate sufficient frequencies of the best solution and even the worst five percentile is six. We can thus consider that 200 is a reasonable number of runs for finding the optimal solution among the resulting ones.

Since ScTucker2 solutions have permutational and reflectional nonuniqueness, we accommodated $\mathbf{A}_c$, $\mathbf{B}_c$, and $\mathbf{H}_c$ for the true counterparts, as described in Appendix F. Let us use $\hat{\mathbf{H}}$, $\hat{\mathbf{A}}$, and $\hat{\mathbf{B}}$ for the solutions $\mathbf{H}_c$, $\mathbf{A}_c$, and $\mathbf{B}_c$ when c is the true sparseness. We define three indices for studying the recovery of the true $\mathbf{H} = (h_{pr})$ by $\hat{\mathbf{H}} = (\hat{h}_{pr})$. One is the *misidentification rate of zero elements* $\mathrm{MR}_0 = 1 - N_{00}/N_0$, where N_0 is the number of zero elements in **H** and N_{00} is the number of zero elements in $\hat{\mathbf{H}}$ whose true counterparts are also zero: $\hat{h}_{pr} = h_{pr} = 0$. In a parallel manner, we can define the *misidentification rate of nonzero elements* as $\mathrm{MR}_\# = 1 - N_{\#\#}/N_\#$ with $N_{\#\#}$ the number of nonzero elements in $\hat{\mathbf{H}}$ whose true counterparts are also nonzero. The other index is the *averaged absolute differences* $\mathrm{AAD}(\mathbf{H}) = (PQK)^{-1}\Sigma_p\Sigma_r|\hat{h}_{pr} - h_{pr}|$. We can also use $\mathrm{AAD}(\mathbf{A})$ and $\mathrm{AAD}(\mathbf{B})$ for measuring the recovery of **A** and **B**, respectively. Obviously, lower values in those indices stand for better recovery. Their percentiles and averages over 100 data sets are presented in Table 2. There, the results for MR_0 and $\mathrm{MR}_\#$ show that the locations of zero and nonzero elements were recovered very well; in particular, they were exactly identified for all error-free data sets and almost all ones of the lower error level. The recovery of the parameter values in **A**, **B**, and **H** was also found to be fairly good in the statistics of the AAD indices, with the recovery for the error-free data perfect. These results allow us to conclude that parameters can be well recovered in ScTucker2.

Finally, we studied the accuracy in the identification of sparseness, using c for the true sparseness and $\hat{c}$ for the sparseness selected by AIC or BIC. In this study, it is senseless to consider error-free data, since AIC and BIC cannot be defined for them:

Table 3
Statistics for the bias of sparseness selected by AIC and BIC.

Error	Lower		Higher	
Index	AIC	BIC	AIC	BIC
5%	−17.05	−2.00	−19.00	−2.00
25%	−14.00	−1.00	−15.00	−1.00
50%	−12.00	0.00	−12.00	0.00
75%	−9.75	0.00	−9.00	0.00
95%	−6.95	0.00	−6.00	1.00
Average	−12.03	−0.47	−12.39	−0.57

those data lead to f (the sum of squared errors (7)) being 0, which implies that the maximum log-likelihood (23) diverges toward infinity with $\sigma^2 = (IJK)^{-1}f = 0$ as found in Appendix C. We thus only analyzed error-perturbed data. For each data set, we obtain the bias $\hat{c} - c$, and its percentiles and average over the 100 data sets are shown in Table 3. There, we can find that the sparseness was selected almost correctly (on average) by BIC for both cases of lower and higher error levels. On the other hand, the sparseness was found to be considerably underestimated by AIC. We thus do not use AIC in real data analysis (Section 7).

6. Differences from Tucker2 followed by rotation

As mentioned in Section 1, the ordinary Tucker2 can produce a core array that approximates a sparse array, by exploiting the rotational freedom of solutions. This freedom is shown by the fact that the Tucker2 model (1) can be rewritten as $\mathbf{X}_k = \mathbf{AS}(\mathbf{S}'\mathbf{H}_k\mathbf{T})(\mathbf{T}'\mathbf{B}') + \mathbf{E}_k = \mathbf{A}^{\#}\mathbf{H}_k^{\#}\mathbf{B}^{\#\prime} + \mathbf{E}_k$. Here, $\mathbf{S}$ and $\mathbf{T}$ are arbitrary orthonormal matrices of $P \times P$ and $Q \times Q$, respectively. That is, $\mathbf{A}^{\#} = \mathbf{AS}$, $\mathbf{B}^{\#} = \mathbf{BT}$, and $\mathbf{H}_k^{\#} = \mathbf{S}'\mathbf{H}_k\mathbf{T}$ ($k = 1, \ldots, K$) which is rewritten as

$$\mathbf{H}^{\#} = \mathbf{S}'\mathbf{H}(\mathbf{I}_K \otimes \mathbf{T}) \quad \text{or} \quad \tilde{\mathbf{H}}^{\#} = \mathbf{T}'\tilde{\mathbf{H}}(\mathbf{I}_K \otimes \mathbf{S}) \tag{27}$$

with $\mathbf{H}^{\#} = [\mathbf{H}_1^{\#}, \ldots, \mathbf{H}_K^{\#}]$ ($P \times QK$) and $\tilde{\mathbf{H}} = [\mathbf{H}_1^{\#\prime}, \ldots, \mathbf{H}_K^{\#\prime}]$ ($Q \times PK$). Furthermore, the $\mathbf{A}$ and $\mathbf{B}$ in (2) can be replaced by $\mathbf{A}^{\#}$ and $\mathbf{B}^{\#}$, respectively; they also satisfy the orthonormality conditions indicated by (2). Those facts imply that if $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{H}_k$ are the optimal solutions, the rotated matrices in $\mathbf{A}^{\#} = \mathbf{AS}$, $\mathbf{B}^{\#} = \mathbf{BT}$, and (27) can also be so. Therefore, the following two-stage procedure can be used in Tucker2:

[1] Data fitting: (1) is fitted to $\mathbf{X}_k$ to obtain an initial solution of $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{H}_k$ ($k = 1, \ldots, K$).
[2] Rotation: A function of (27) is maximized or minimized over orthonormal $\mathbf{S}$ and $\mathbf{T}$ so that the resulting core array (27) has a simple structure.

Here, a simple structure refers to a property of a matrix that it has a number of elements close to zero (Browne, 2001; Thurstone, 1947). Giving such a structure to a matrix is restated as simplifying the matrix.

As a procedure for [2], Kiers' (1997) orthomax rotation procedure for three-way data can be used, which is formulated as maximizing

$$\kappa_1 \mathrm{Dp}(\mathbf{H}^{\#}, \tau_1) + \kappa_2 \mathrm{Dp}(\tilde{\mathbf{H}}^{\#}, \tau_2) = \kappa_1 \mathrm{Dp}((\mathbf{S}'\mathbf{H}(\mathbf{I}_K \otimes \mathbf{T}))', \tau_1) + \kappa_2 \mathrm{Dp}((\mathbf{T}'\tilde{\mathbf{H}}(\mathbf{I}_K \otimes \mathbf{S}))', \tau_2) \tag{28}$$

over orthonormal $\mathbf{S}$ and $\mathbf{T}$. Here, κ_1 and κ_2 are nonnegative constants and $\mathrm{Dp}(\mathbf{M}, \tau) = \|\mathbf{M}{\bullet}\mathbf{M}\|^2 - \tau n^{-1}\|\mathbf{1}_{N'}(\mathbf{M}{\bullet}\mathbf{M})\|^2$ expresses the dispersion of the squares of the elements in an $n \times m$ matrix $\mathbf{M}$ with $\tau \geq 0$. When $\tau = 1$, $\mathrm{Dp}(\mathbf{M}, \tau)$ is the well known varimax criterion (Kaiser, 1958). Though Kiers (1998a) has extended the procedure for also simplifying component matrices, the extended one is not treated in this paper, as we only consider simplifying/sparsifying the core array.

Kiers' (1998b) simplimax rotation for three-way data can also be used for [2]. In this method, a sparse target matrix $\mathbf{H}_{\mathrm{T}} = (h_{pr}^{\mathrm{T}})$ to be approximated by $\mathbf{H}^{\#} = (h_{pr}^{\#})$ is considered:

$$\|\mathbf{H}_{\mathrm{T}} - \mathbf{H}^{\#}\|^2 = \|\mathbf{H}_{\mathrm{T}} - \mathbf{S}'\mathbf{H}(\mathbf{I}_K \otimes \mathbf{T})\|^2 \tag{29}$$

is minimized over $\mathbf{S}$, $\mathbf{T}$, and $\mathbf{H}_{\mathrm{T}}$ subject to

$$\mathrm{Sp}(\mathbf{H}_{\mathrm{T}}) = s \tag{30}$$

with s a specified integer. This method is similar to ScTucker2 in that $\mathbf{H}_{\mathrm{T}}$ has the sparseness constraint (30) equivalent to (4): the locations of nonzero elements and their values in $\mathbf{H}_{\mathrm{T}}$ are optimally estimated. Indeed, $\mathbf{S}$ and $\mathbf{T}$ given, the minimization of (29) over $\mathbf{H}_{\mathrm{T}}$ under (30) is attained by the formula (18) with its h_{pr}, g_{pr}, and ${g_{[c]}}^2$ replaced by h_{pr}^{T}, $h_{pr}^{\#}$, and the sth smallest value among all $h_{pr}^{\#2}$, respectively. However, a difference is that the core array is constrained to be sparse in ScTucker2, while the target matrix (rather than the core array) is sparsified in the simplimax rotation. For selecting the $\mathrm{Sp}(\mathbf{H}_{\mathrm{T}})$ value in (30), a scree procedure is used, in which an increase in the value of (29) as a function of $\mathrm{Sp}(\mathbf{H}_{\mathrm{T}})$ is noted: $\mathrm{Sp}(\mathbf{H}_{\mathrm{T}})$ is selected whereafter (29) value shows a considerable relative increase (Kiers, 1994).

The above rotation methods might produce the core array similar to the sparse ones obtained by ScTucker2. However, Tucker2 followed by rotation differs from ScTucker2 in the following ways:

[A] The elements of rotated matrices cannot exactly be zero.

[B] Rotation does not involve data; the function in [2] is defined without $\mathbf{X}_k$.

These two properties are possessed by every rotation technique.

In spite of [A], we can regard the elements of small absolute values as zeros. This may be good enough for interpretational purposes. However, it needs a troublesome job of deciding a threshold, the values less than which are considered to be small enough to be neglected. For this job, we need a subjective decision. In contrast, such a job is unnecessary in ScTucker2: the Sp($\mathbf{H}$) elements in a core array become exactly zero.

The property [B] is found in (28) and (29), which are functions of a core array and the orthonormal matrices simplifying the array without a data matrix considered. That is, in Tucker2 before rotation, its model is optimally fitted to data, but the objective function in the subsequent rotation is optimized only for simplifying a solution. On the other hand, the Tucker2 model is fitted to data, subject to the sparseness constraint, in ScTucker2; and the appropriate sparseness can be selected using BIC which is defined by considering the fitness to a data set. However, such a measure is not available in rotation: though the sparseness value in (30) can be chosen by the scree procedure, the function (29) used in the procedure does not involve data.

Besides Kiers' (1997, 1998a,b) orthomax and simplimax rotation methods described above, some rotation methods have been proposed for Tucker2 solutions (Brouwer and Kroonenberg, 1991; Kroonenberg, 1983; Adachi, 2011). However, they have special purposes beyond giving simple structures to the resulting matrices. In the next examples, we thus compare ScTucker2 with Tucker2 followed by Kiers' (1997, 1998) orthomax and simplimax rotation. There, the former rotation is performed with $\kappa_1 = (QK)^{-1}$, $\kappa_2 = (PK)^{-1}$ and $\tau_1 = \tau_2 = 1$ in (28) set to one, while the latter is done with $\mathbf{S}$ and $\mathbf{T}$ constrained to be orthonormal so that $\mathbf{A}$ and $\mathbf{B}$ satisfy (2). Without this constraint, the simplimax rotation has been originally proposed (Kiers, 1994) and it can provide simpler core slices than the constrained version, but we use this version, as it is out of our scope whether or not (2) should be imposed and the use of the constrained version is thought to facilitate the comparisons to the ScTucker2 and the orthomax solutions with (2). In the next section, we further compare ScTucker2 with Parafac, which is formulated as minimizing (1) over $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{D}_k$ subject to (2) and (3), with the solution not having rotational freedom (Smilde et al., 2004). The constraint (2) is also dispensable in Parafac, i.e., its unconstrained version provides the solutions without rotational freedom, but they tend to degenerate (Smilde et al., 2004) which is also found in the next example.

7. Real data examples

We will now illustrate ScTucker2 and compare it with Tucker2 and Parafac using two real data sets. One of the data sets is a three-way array of impression ratings. The other data set is an array describing the physical growth patterns of girls. For the data sets, ScTucker2 is performed in the same procedure as in Section 5.

7.1. Impression rating data

The first example is Osgood and Luria's (1954, pp. 582–583) data array $\{y_{ijk}\}$ of 15-concepts $\times$ 10-adjectives $\times$ 6-occasions, which is available at http://three-mode.leidenuniv.nl. From that web page, we downloaded the data set. It contains rating values describing how strongly the concepts are associated with the adjectives. The rating was performed twice by a female suffering from dissociative identity disorder under each of the three assumed personalities, known as *Jane*, *Eve White*, and *Eve Black*, for a total of six occasions. That is, three personalities $\times$ two cases formed the six occasions. For the rating, a bipolar scale ranging from 1 to 7 was used, with each end of the scale corresponding to either an adjective or its antonym. For the analysis, 4 (the midpoint of 1 and 7) was subtracted from the raw value y_{ijk} and standardized with $x_{ijk} = (x_{ijk} - 4)/\{\Sigma_i \Sigma_j (x_{ijk} - 4)^2/150\}^{1/2}$, so that $x_{ijk} = 0$ implies a neutral rating and the sum of squares of x_{ijk} over i and j is equivalent across k.

ScTucker2, Tucker2 and Parafac were carried out for $\{x_{ijk}\}$ with the six occasions treated as sources. Here, P and Q were set to three, as Osgood and Luria (1954) have assumed that concepts and adjectives are distributed in a three-dimensional space. The Tucker2 solution was rotated by the orthomax and simplimax methods described in Section 6, where the scree procedure described in the next paragraph was used for the latter method. In ScTucker2, the BIC-based selection of the sparseness Sp($\mathbf{H}$) $= c$ showed the best Sp($\mathbf{H}$) being 37, with BIC $= 2048.9$. As described in Appendices C, D and F, BIC can also be defined for the Tucker2 and Parafac models under the normality assumption (22) for errors. The BIC values were 2201.6 and 2439.8 for the resulting Tucker2 and Parafac solutions. By comparing the three BIC values, we can find that the ScTucker2 model with Sp($\mathbf{H}$) $= 37$ is the best one for the data set among the models considered. The GOF values, which are defined as (21), were $GOF(37) = 0.715$ for the ScTucker2 solution and $GOF(0) = 0.733$ for the Tucker2 one. It should be noted that $GOF(37)$ is not so much lower than $GOF(0)$: the ScTucker2 solution approximates the data set well in spite of 37 elements' vanishing among the 54 ones in the core array. The GOF value for Parafac, defined as in Appendix G,

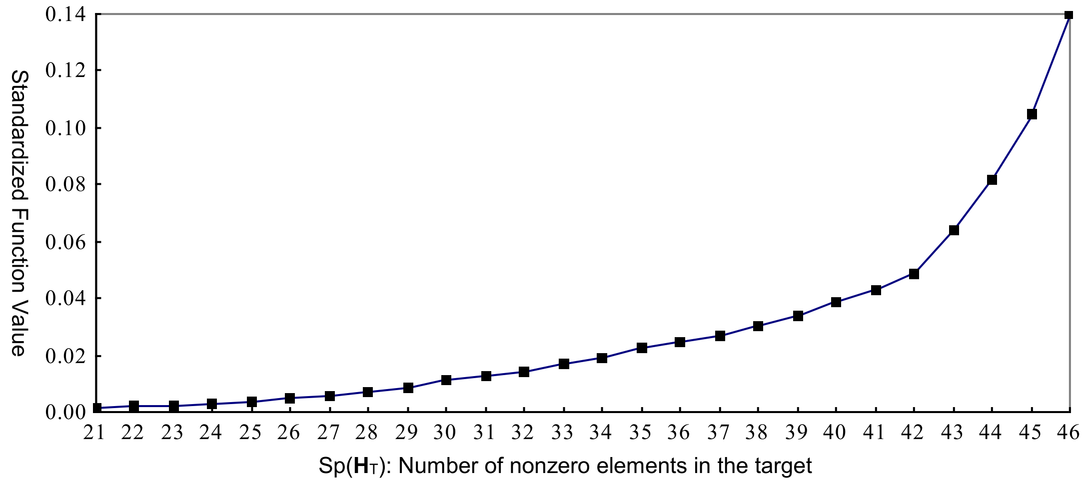

Fig. 1. Plot of the simplimax function values against $\text{Sp}(\mathbf{H}_\text{T})$ for impression rating data, with only the range of $\text{Sp}(\mathbf{H}_\text{T})$ shown within which a reasonable $\text{Sp}(\mathbf{H}_\text{T})$ is found.

was $GOF_\text{PA} = 0.564$, which suggests that the Parafac model does not approximate the data very well. The frequency of the best solution which is defined as in Section 5.2 was 73.8 in average and 23 for $\text{Sp}(\mathbf{H}) = 37$ (the selected sparseness). In Appendix H, we report the CPU time taken by the ScTucker2 and simplimax methods in which algorithms must be run for the multiple values of sparseness.

In the simplimax method, the scree procedure was used, in which the values of the simplimax function (29) were obtained for $\text{Sp}(\mathbf{H}_\text{T}) = c_\text{min}, \ldots, c_\text{max}$ for choosing the best $\text{Sp}(\mathbf{H}_\text{T})$ with c_min and c_max defined in Section 4. Fig. 1 shows the plot of the resulting function values. In the figure, the value is found to increase remarkably after $\text{Sp}(\mathbf{H}_\text{T}) = 42$, which allows us to choose $\text{Sp}(\mathbf{H}_\text{T}) = 42$. It is greater than $\text{Sp}(\mathbf{H}) = 37$, which allows us to hypothesize that the simplimax function values tend to suggest the choice of the $\text{Sp}(\mathbf{H}_\text{T})$ value greater than the true $\text{Sp}(\mathbf{H})$. This hypothesis is congruous to the result of a simulation study reported in Appendix I, though it cannot be concluded that the hypothesis was supported, as only one set of simulated data was examined in that study.

The resulting component matrices **A** and **B** are shown in Table 4. There, we can find that the values of the elements are very similar among the four solutions, and their components can thus be interpreted in the same manner. The "L, H, I" and "E, A, S" indicating the columns of **A** and **B** are the abbreviations for the names which we can give the corresponding components. That is, the three components for concepts in **A** can be related to (*daily*) *life* (L), *health* (H), and *illness* (I), respectively, from the signs of the elements and their absolute values. Those in **B** allow us to interpret the components for adjectives as standing for *evaluation* (E), *activity* (A), and (*the width of*) *scope* (S), respectively.

The six core slices $\mathbf{H}_1, \ldots, \mathbf{H}_6$ resulting in ScTucker2 and Tucker2 are presented in Table 5 together with the diagonal matrices $\mathbf{D}_1, \ldots, \mathbf{D}_6$ obtained by Parafac. There, bold font is used for the nonzero elements in the ScTucker2 solution and the corresponding ones in the Tucker2 and Parafac solutions. We can find that the values of the boldfaced elements are similar and their signs are equivalent among the four methods. However, the ScTucker2 solution obviously differs from that from Parafac with the off-diagonal elements of $\mathbf{D}_k$ restricted to zero. For example, in the ScTucker2 solution, the strong link of H to E in Eve Black 1 and 2 is found with the corresponding off-diagonal elements 8.4 and 9.9, but such a link cannot be found by Parafac. Furthermore, we should note that the ScTucker2 solution with $\text{Sp}(\mathbf{H}) = 37$ is sparser than the Parafac one with $\text{Sp}([\mathbf{D}_1, \ldots, \mathbf{D}_6]) = 36$. It demonstrates the flexibility of ScTucker2, in that the locations of zero elements are estimated optimally in contrast to Parafac with their locations fixed; the flexibility allowed ScTucker2 to find sparser core slices with better locations of nonzero elements.

On the other hand, the solutions of Tucker2 followed by rotation may approximate the ScTucker2 one if the elements whose absolute values are less than a threshold value in the former are regarded as zeros. However, such a value is difficult to choose in an objective manner. An approach may be to use the $\text{Sp}(\mathbf{H}_\text{T})$ value selected in the simplimax, i.e., to regard the $\text{Sp}(\mathbf{H}_\text{T})$ smallest absolute values as zeros, but $\text{Sp}(\mathbf{H}_\text{T})$ can be too great as shown in Appendix I, so that some elements of **H** may become unreasonably close to zero. Indeed, some of the elements in Table 5, which were estimated as nonzero in ScTucker2, are found to take the values close to zero in the simplimax solution and also in the orthomax one. For example, the element for the L and E components in Eve Black 2 takes −2.4 in the ScTucker2 solution, while the corresponding elements are −1.7 and −1.4 in the simplimax and orthomax solutions, respectively. They must be neglected as zero, if we use the threshold value by which the $\text{Sp}(\mathbf{H}_\text{T}) = 42$ smallest absolute values are regarded as zeros.

The interpretation of the ScTucker2 core slices $\mathbf{H}_k$ can be facilitated by representing them as the network diagrams in Fig. 2, which has been depicted in the following manner:

Table 4
Component matrices for impression rating data.

		ScTucker2			Tucker2-Simplimax			Tucker2-Orthomax			PARAFAC		
		L	H	I	L	H	I	L	H	I	L	H	I
A	Love	0.34	−0.24	−0.03	0.32	−0.27	0.01	0.31	−0.28	0.00	0.38	−0.16	0.01
	My child	0.27	−0.16	0.16	0.25	−0.16	0.15	0.25	−0.17	0.14	0.24	−0.09	0.14
	My doctor	0.36	0.43	−0.05	0.40	0.41	−0.03	0.40	0.41	−0.02	0.27	0.50	−0.09
	Me	0.04	0.36	0.22	0.04	0.33	0.25	0.05	0.32	0.26	0.00	0.33	0.19
	My job	0.26	−0.26	0.09	0.23	−0.28	0.09	0.22	−0.29	0.07	0.31	−0.34	0.15
	Mental sick	0.19	−0.23	0.39	0.14	−0.25	0.43	0.14	−0.27	0.42	0.24	−0.13	0.45
	My mother	0.29	0.10	0.27	0.28	0.10	0.27	0.29	0.08	0.27	0.24	0.20	0.26
	Peace of mind	0.33	0.37	−0.23	0.37	0.33	−0.16	0.37	0.33	−0.15	0.33	0.40	−0.24
	Fraud	−0.24	0.21	0.40	−0.26	0.23	0.38	−0.25	0.22	0.39	−0.30	0.22	0.34
	My spouse	0.17	−0.23	−0.04	0.16	−0.25	−0.02	0.15	−0.25	−0.03	0.19	−0.22	−0.04
	Self control	0.33	0.09	−0.03	0.34	0.06	0.00	0.34	0.05	0.00	0.33	0.11	−0.01
	Hatred	−0.19	0.23	0.44	−0.20	0.24	0.45	−0.19	0.22	0.46	−0.22	0.20	0.42
	My father	0.35	0.22	0.14	0.35	0.20	0.15	0.36	0.19	0.15	0.28	0.25	0.14
	Confusion	0.05	−0.20	0.51	0.00	−0.20	0.51	0.00	−0.22	0.50	0.03	−0.11	0.52
	Sex	0.16	−0.29	−0.02	0.14	−0.33	0.03	0.13	−0.33	0.02	0.20	−0.24	0.00
		E	A	S	E	A	S	E	A	S	E	A	S
B	Hot	0.19	−0.35	0.19	0.21	−0.29	0.21	0.26	−0.25	0.21	0.12	−0.30	0.19
	Worthless	−0.43	−0.03	−0.01	−0.44	−0.09	0.01	−0.41	−0.16	0.01	−0.44	0.06	0.00
	Relaxed	0.13	−0.36	0.60	0.16	−0.28	0.60	0.21	−0.25	0.60	0.05	−0.30	0.63
	Large	0.39	0.27	−0.05	0.39	0.26	−0.09	0.34	0.32	−0.09	0.44	0.10	−0.10
	Fast	0.06	0.70	0.21	0.07	0.76	0.17	−0.06	0.76	0.17	0.32	0.73	0.12
	Clean	0.43	0.18	−0.09	0.43	0.18	−0.12	0.39	0.25	−0.12	0.44	−0.01	−0.10
	Tasty	0.32	−0.06	0.29	0.32	−0.09	0.29	0.33	−0.03	0.29	0.28	−0.17	0.30
	Weak	−0.38	0.00	−0.25	−0.37	0.06	−0.25	−0.38	−0.01	−0.25	−0.34	0.14	−0.27
	Deep	−0.33	0.01	0.44	−0.31	0.06	0.46	−0.31	0.00	0.46	−0.28	0.16	0.43
	Active	−0.26	0.40	0.44	−0.25	0.37	0.43	−0.31	0.32	0.43	−0.13	0.44	0.43

Table 5
Core slices across multiple personalities for impression rating data with blank cells denoting zero elements.

		ScTucker2				Tucker2-Simplimax				Tucker2-Orthomax				PARAFAC		
		E	A	S		E	A	S		E	A	S		E	A	S
	L	**9.0**			L	**8.9**	0.4	0.1	L	**8.7**	1.9	0.0	L	**8.7**		
Jane 1	H				H	−1.1	−0.7	−0.8	H	−1.2	−1.0	−0.6	H		−1.0	
	I			**−5.1**	I	1.2	1.0	**−5.0**	I	0.8	1.1	**−5.1**	I			**−5.1**
	L	**10.2**			L	**10.1**	−0.8	0.3	L	**10.1**	1.0	0.2	L	**9.5**		
Jane 2	H				H	−1.4	−0.8	−0.5	H	−1.5	−1.0	−0.3	H		−1.3	
	I			**−4.0**	I	0.8	−0.6	**−3.9**	I	0.8	−0.5	**−4.0**	I			**−3.9**
	L	**8.5**			L	**8.6**	0.6	−0.7	L	**8.3**	2.1	−0.7	L	**8.3**		
Eve White 1	H				H	−0.6	−0.5	1.3	H	−0.7	−0.7	1.5	H		−0.6	
	I	**−1.8**		**−4.9**	I	**−1.4**	0.3	**−4.9**	I	**−1.6**	0.0	**−4.8**	I			**−5.0**
	L	**8.5**			L	**8.6**	0.0	0.0	L	**8.5**	1.5	−0.1	L	**7.9**		
Eve White 2	H				H	0.5	−0.9	0.4	H	0.5	−0.8	0.6	H		−1.6	
	I			**−5.1**	I	−0.6	−0.6	**−5.2**	I	−0.6	−0.8	**−5.1**	I			**−5.0**
	L	***−1.6***			L	**−1.0**	0.5	−0.6	L	**−0.9**	0.2	−0.6	L	**−2.1**		
Eve Black 1	H	**8.4**	**−4.6**		H	**8.5**	**−4.5**	0.1	H	**9.2**	**−2.9**	0.1	H		**−6.8**	
	I				I	−0.3	−1.1	−1.0	I	0.3	−1.3	−1.0	I			−1.0
	L	**−2.4**		**−1.9**	L	**−1.7**	−0.6	**−1.7**	L	**−1.4**	−0.9	**−1.7**	L	**−3.6**		
Eve Black 2	H	**9.9**	**−3.8**		H	**10.1**	**−3.7**	0.2	H	**10.7**	**−1.7**	0.2	H		**−6.7**	
	I			**−1.6**	I	−1.0	−1.3	**−1.6**	I	−0.3	−1.5	**−1.6**	I			**−1.5**

[1] The components for concepts and those for adjectives are depicted as left and right boxes, respectively, with their abbreviations.

[2] No link exists between the components associated with $\tilde{h}_{pqk} = 0$, where $\tilde{h}_{pqk}$ is the (p, q) elements of $\mathbf{H}_k$.

[3] The components with $\tilde{h}_{pqk} > 0$ and those with $\tilde{h}_{pqk} < 0$ are linked by solid and dotted lines, respectively, with the $\tilde{h}_{pqk}$ values attached to the lines and their width proportional to the absolute values of $\tilde{h}_{pqk}$.

In Fig. 2, we can find that the component links shown by the resulting core slices are almost invariant across the two cases within the same personality. Further, the links are almost equivalent between Jane and Eve-White, but their links

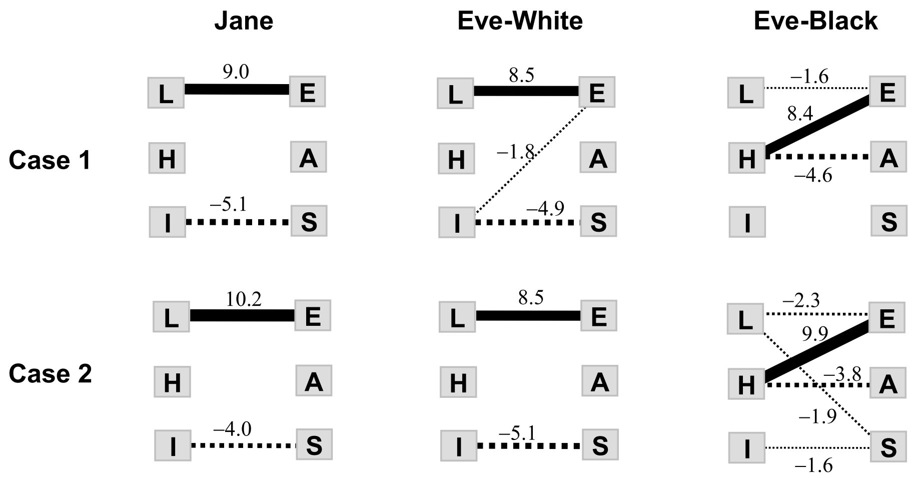

Fig. 2. Network representation of the core slices for ScTucker2 in Table 5.

differ from those for Eve-Black. That is, the core slices for Jane and Eve-White show a strong positive link between L (life) and E (evaluation) and a negative one between I (illness) and S (scope), without a link between H (heath) and A (activity). In contrast, L is rather negatively linked to E, and H is linked positively to E but negatively to A in the core slices for Eve-Black.

As already stated, the model with $\mathrm{Sp}(\mathbf{H}) = 37$ chosen in ScTucker2 is sparser than the Parafac one with $\mathrm{Sp}([\mathbf{D}_1, \ldots, \mathbf{D}_6]) = 36$. In spite of the number of parameters being smaller for the former model, its $GOF(37) = 0.715$ was higher than the latter $GOF_{\mathrm{PA}} = 0.564$, which suggests that the Parafac is not suitable to this data set. It can be shown clearly by the ScTucker2 solution with $\mathrm{Sp}(\mathbf{H}) = 36$, that is, by equalizing the sparseness to that of $[\mathbf{D}_1, \ldots, \mathbf{D}_6]$ in Parafac. The resulting $GOF(36) = 0.717$ for ScTucker2 allows us to state that it explains more amount of the variances in the data than Parafac by $100 \times (0.717 - 0.564) = 15.3\%$. The ScTucker2 with $\mathrm{Sp}(\mathbf{H}) = 36$ is shown in Table 6, whose core slices are found to differ from the diagonal matrices supposed in Parafac. Table 6 also demonstrates the stability of ScTucker2 solutions between different $\mathrm{Sp}(\mathbf{H})$. That is, the locations of nonzero elements for ScTucker2 core arrays are equivalent between $\mathrm{Sp}(\mathbf{H}) = 36$ (Table 6) and $\mathrm{Sp}(\mathbf{H}) = 37$ (Table 5), except the (2,3) element for Eve white 1 being nonzero for the former. Further, the values of the nonzero elements of the component matrices and core arrays in the ScTucker2 solution in Tables 4 and 5 are found to be almost equivalent to those in Table 6.

As stated in Section 6, Parafac can be performed without the constraint (2). This unconstrained Parafac showed high goodness of fit with $GOF_{\mathrm{PA}} = 0.724$, but the solution was degenerate and not useful: the resulting **A** and **B** were rank deficient with the maximum of the cosines between the column vectors in **A** being 0.97 and that for **B** 0.99. How degenerate the solution was can be found in Adachi (2013, pp. 47–49), where the same analysis has also been performed.

7.2. Physical growth data

The second example is the data set that describes the eight physical measurements of 30 normal French girls at ages 4, 5, …, 15 (Janssen et al., 1987). It gives 30-girls × 8-measures × 12-ages array of measurements which is available at http://three-mode.leidenuniv.nl. We standardized the raw value y_{ijk} in the array as $x_{ijk} = (y_{ijk} - \bar{y}_{.j.})/s_j$, with $\bar{y}_{.j.}$ and $s_{.j.}$ the average and standard deviation of y_{ijk} over $i = 1, \ldots, 30$ and $k = 1, \ldots, 12$, respectively. This standardization removes the influence of different scales among measures. We chose $P = 2$ and $Q = 3$, for the sake of illustrating a case with $P \neq Q$ and the consideration that ScTucker2 and Tucker 2 show satisfactorily high GOF as found below in spite of small P and Q values. Tucker2 with the two rotation methods and ScTucker2 were carried out for the data set. In ScTucker2, the sparseness $\mathrm{Sp}(\mathbf{H}) = 32$ was selected with the corresponding BIC value 1477.7. This was less than BIC = 1648.6 for the Tucker2 solution, showing that the ScTucker2 model is better for this data set. The $GOF(32)$ for the ScTucker2 solution was 0.930, which is almost equivalent to $GOF(0) = 0.932$ for Tucker2 solution. The frequency of the best solution which is defined as in Section 5.2 was 113.0 in average and 46 for $\mathrm{Sp}(\mathbf{H}) = 32$.

In the simplimax rotation, the scree selection of $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ was performed in the same procedure as in the last example. Fig. 3 shows the plot of the resulting values of the simplimax function. There, we can find some candidates for the best $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$. One is $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 56$, as the gradient of the curve before $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 56$ is different from the one after it. However, the increase in the value is remarkable after $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 58$ and furthermore remarkable after $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 59$. Here, we selected $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 56$, though in a subjective manner. This demonstrates a drawback of the simplimax rotation in which we must resort to the scree selection requiring a subjective decision. As in the last example, $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 56$ was greater than $\mathrm{Sp}(\mathbf{H}) = 32$ for ScTucker2, which gives the hypothesis that the selected $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ is too great. This hypothesis is congruous to the results of a simulation study in Appendix J.

Table 6
ScTucker2 solution with Sp(**H**) set to the sparseness of the Parafac core.

Component matrices				Core slices				
	L	H	I			E	A	S
Love	0.34	−0.24	−0.03		L	9.0		
My child	0.27	−0.16	0.16	Jane 1	H			
My doctor	0.37	0.43	−0.03		I			−5.1
Me	0.04	0.35	0.23			E	A	S
My job	0.26	−0.26	0.09		L	10.2		
Mental sick	0.18	−0.24	0.39	Jane 2	H			
My mother	0.29	0.10	0.27		I			−4.0
Peace of mind	0.33	0.37	−0.22			E	A	S
Fraud	−0.25	0.21	0.40		L	8.5		
My spouse	0.17	−0.23	−0.04	Eve White 1	H			1.3
Self control	0.33	0.09	−0.03		I	−1.8		−4.9
Hatred	−0.19	0.23	0.45			E	A	S
My father	0.35	0.22	0.15		L	8.5		
Confusion	0.04	−0.22	0.51	Eve White 2	H			
Sex	0.16	−0.29	−0.03		I			−5.1
	E	A	S			E	A	S
Hot	0.20	−0.35	0.19		L	−1.5		
Worthless	−0.43	−0.03	0.00	Eve Black 1	H	8.4	−4.6	
Relaxed	0.13	−0.36	0.60		I			
Large	0.39	0.27	−0.06			E	A	S
Fast	0.06	0.69	0.21		L	−2.3		−1.8
Clean	0.43	0.18	−0.09	Eve Black 2	H	9.9	−3.8	
Tasty	0.32	−0.06	0.29		I			−1.6
Weak	−0.38	0.00	−0.26					
Deep	−0.32	0.01	0.45					
Active	−0.25	0.40	0.45					

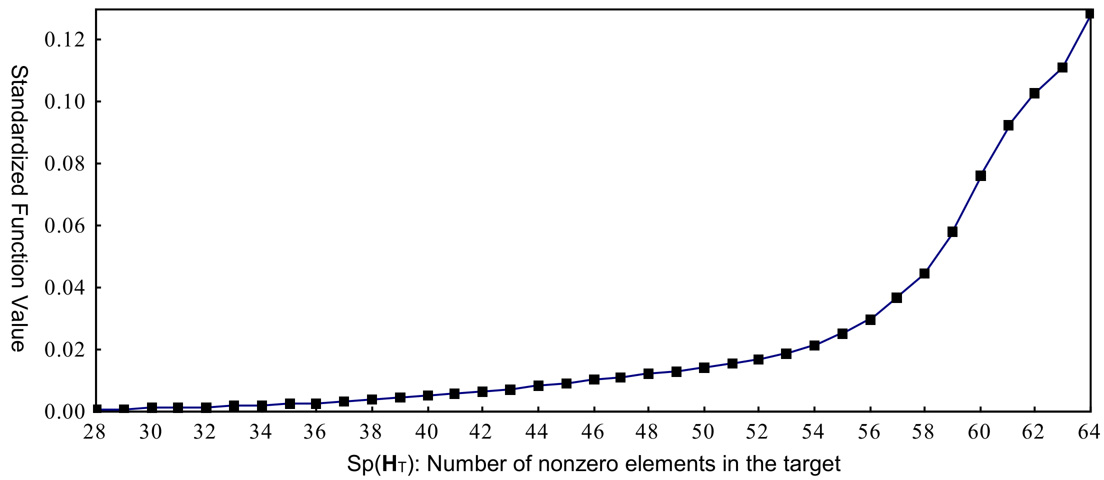

Fig. 3. Plot of the simplimax function values against Sp($\mathbf{H}_T$) for growth data, with only the range of Sp($\mathbf{H}_T$) shown within which reasonable Sp($\mathbf{H}_T$) is found.

Fig. 4 presents the loadings in the resulting component matrix **A**. There, the values of the elements are found to be similar among the three solutions, and their components can thus be interpreted in the same manner. The first component is interpreted as a *common* (C) component, as its loadings for all girls are similar positive values. In contrast, the second one can be named an individual difference (D) component, as the signs of its loadings allow us to classify the girls into two groups. Table 7 shows the resulting **B**. Among the solutions, small differences are found, but the same names can be given to the components. The first one can be called a *size* (S) component, as it shows positive loadings for all measures associated with physical sizes. The second can be named a length component, as it gives large positive loadings for the two length measures. The third component can be called a *parts* (P) component, as its loadings differ among physical parts: the loadings for the arm and chest circumferences are positive, while the loading for the head's circumference is negative and of a large absolute value.

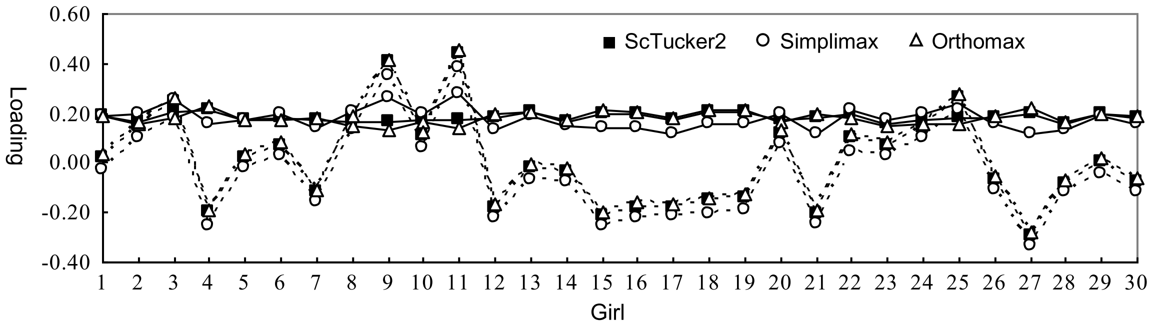

Fig. 4. The first and second component loadings connected by real and dotted lines, respectively.

Table 7
Component matrices for measures in growth data.

Measure	ScTucker2			Tucker2-Simplimax			Tucker2-Orthomax		
	S	L	P	S	L	P	S	L	P
Weight	0.33	0.14	0.31	0.37	−0.23	0.31	0.37	0.04	0.38
Length	0.16	0.65	−0.44	0.36	0.72	0.00	0.36	0.51	−0.51
Crown.rump.length	0.24	0.40	−0.04	0.36	0.28	0.13	0.37	0.29	−0.11
Head.circumference	0.57	−0.55	−0.52	0.33	−0.22	−0.85	0.31	−0.75	−0.46
Chest.circumference	0.31	0.16	0.46	0.36	−0.27	0.39	0.36	0.07	0.46
Arm	0.41	−0.17	0.46	0.34	−0.46	0.08	0.34	−0.28	0.38
Calf	0.31	0.19	−0.03	0.36	0.09	−0.01	0.36	0.05	−0.07
Pelvis	0.35	0.09	−0.15	0.36	0.05	−0.11	0.36	−0.05	−0.12

The core slices provided by ScTucker2 and the rotation following Tucker2 are presented in Table 8, where bold font is used for the nonzero elements in the ScTucker2 solution and the corresponding ones in the solutions of rotation. The values of (C, S) elements are found to be similar among the three solutions, where the (C, S) one stands for the element of each core slice associated with the C and S components. The value of that element is found to increase monotonically with age, which indicates that the growth in physical sizes is common to all girls. However, the (D, S) elements in the simplimax solution differ from those in the ScTucker2 and orthomax ones. The latter solutions allow us to find that how girls differ in sizes is invariant across ages, since the signs of the elements are equally negative and their absolute values are not very different among ages. However, such an interpretation cannot be made for the simplimax (D, S) elements, as their signs for the first three ages differ from the signs for the other ages. The differences of the ScTucker2 solution from the two rotation ones are found in the (C, L) and (C, P) elements. In the ScTucker2 solution, the value of the (C, L) element is found to increase monotonically with age, which can be reasonably interpreted as showing that the growth in lengths is common to all girls as those in sizes. On the other hand, such a clear interpretation cannot be made for the (C, L) elements in the rotation solutions, as [1] their values are close to zero and [2] the sign of the element for the final age is negative. The former result [1] is congruous to that the scree procedure suggested a greater value of $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ than $\mathrm{Sp}(\mathbf{H})$. The closeness of the rotated elements to zero is also found in almost all of (C, P) elements for which ScTucher2 provided nonzero values.

The result in Appendix J allows us to consider that the $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ value 56 selected in the simplimax rotation is too great, which suggests that the rotation with $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ set to $\mathrm{Sp}(\mathbf{H}) = 32$ selected in ScTucker2 may give a more reasonable solution. We thus performed the rotation with $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 32$, which provided the solution very similar to the ScTucker2 one as found in Table 9. It presents the resulting **B** and **H**, whose elements are similar to the ScTucker2 ones in Tables 7 and 8. Furthermore, the loadings in $\mathbf{A} = (a_{ip})$ given by the simplimax rotation were almost equivalent to those for ScTucker2 shown in Fig. 4, which is indicated by $(IP)^{-1}\Sigma_{i,p}|a_{ip}^{(\mathrm{spx})} - a_{ip}^{(\mathrm{sc})}| = 0.001$ with $a_{ip}^{(\mathrm{spx})}$ and $a_{ip}^{(\mathrm{sc})}$ the simplimax and ScTucker2 solutions of a_{ip}, respectively. Here, it should be remembered that the simplimax solution of **H** for $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 56$ differed from the ScTucker2 one as described in the last paragraph, but the change of $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ into 32 allows the simplimax solution to approach the ScTucker2 one. It suggests that ScTucker2 can help the selection of $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ in the simplimax rotation. At least, the BIC-based selection of sparseness in ScTucker2 allows us to avoid a subjective decision required by the scree selection in the simplimax.

8. Discussion

We proposed a sparse core Tucker2 (ScTucker2) method, in which the Tucker2 loss function is minimized subject to the sparseness constraint for the core array. In the alternating least squares algorithm for ScTucker2, the fact was used that the loss function is decomposed into a term irrelevant to the core array and its simple function, which allows us to easily attain the sparseness-constrained minimization of the Tucker2 loss function. We further presented a procedure for selecting a

Table 8
Core slices across ages for growth data with blank cells denoting zero elements.

Age		ScTucker2				Tucker2-Simplimax				Tucker2-Orthomax		
		S	L	P		S	L	P		S	L	P
4	C	**−19.0**	**−7.7**	**1.6**	C	**−20.6**	**−1.8**	1.4	C	**−20.1**	**−0.3**	2.3
	D	**−3.5**			D	**2.1**	1.1	0.9	D	**−4.8**	1.4	0.7
5	C	**−15.9**	**−6.5**		C	**−17.4**	**−0.9**	0.4	C	**−16.8**	**−0.4**	0.9
	D	**−3.7**			D	**0.9**	0.8	1.1	D	**−4.9**	1.2	0.6
6	C	**−12.8**	**−5.2**		C	**−14.2**	**−0.3**	0.0	C	**−13.5**	**−0.3**	0.1
	D	**−3.6**			D	**0.2**	0.6	1.3	D	**−4.5**	1.3	0.6
7	C	**−9.9**	**−3.9**	**−1.1**	C	**−11.3**	**0.4**	**−0.4**	C	**−10.3**	**−0.3**	**−0.7**
	D	**−4.2**			D	**−1.1**	0.6	1.5	D	**−4.8**	1.5	0.4
8	C	**−6.5**	**−2.6**	**−1.1**	C	**−7.8**	**0.6**	**−0.5**	C	**−6.7**	**−0.3**	**−0.9**
	D	**−4.4**			D	**−2.2**	0.7	1.6	D	**−4.6**	1.7	0.4
9	C	**−3.3**	**−1.2**	**−1.1**	C	**−4.6**	**0.9**	**−0.4**	C	**−3.2**	**−0.2**	**−1.0**
	D	**−4.6**			D	**−3.3**	0.8	1.6	D	**−4.5**	1.7	0.3
10	C				C	−1.0	0.8	−0.2	C	0.5	−0.1	−0.7
	D	**−4.7**			D	**−4.3**	1.0	1.1	D	**−4.4**	1.7	−0.1
11	C	**4.4**	**1.8**		C	**3.3**	**0.8**	0.0	C	**5.1**	**0.0**	−0.4
	D	**−5.3**			D	**−6.1**	1.2	0.8	D	**−4.7**	1.6	−0.4
12	C	**9.3**	**4.0**		C	**8.4**	**1.1**	0.2	C	**10.4**	**0.2**	−0.7
	D	**−5.4**			D	**−7.5**	1.0	1.0	D	**−4.3**	1.7	−0.2
13	C	**14.2**	**6.2**		C	**13.7**	**1.3**	0.7	C	**15.9**	**0.5**	−0.3
	D	**−5.4**			D	**−8.9**	1.2	0.6	D	**−3.9**	1.8	−0.5
14	C	**19.0**	**7.4**	**1.4**	C	**18.5**	**0.3**	**1.1**	C	**20.7**	**0.0**	**0.7**
	D	**−5.1**			D	**−9.7**	1.8	0.6	D	**−3.0**	2.0	−0.6
15	C	**22.1**	**7.9**	**2.1**	C	**21.7**	**−0.5**	**1.3**	C	**23.8**	**−0.6**	**1.3**
	D	**−4.4**			D	**−9.8**	1.9	0.6	D	**−2.0**	1.9	−0.4

Table 9
Simplimax solution with $Sp(\mathbf{H}_T)$ set to the sparseness selected in ScTucker2.

Component matrix **B**	S	L	P
Weight	0.33	0.13	0.40
Length	0.16	0.65	−0.45
Crown.rump.length	0.24	0.41	−0.07
Head.circumference	0.56	−0.55	−0.52
Chest.circumference	0.31	0.15	0.48
Arm	0.42	−0.18	0.36
Calf	0.32	0.18	−0.05
Pelvis	0.35	0.09	−0.11

Core slices in **H**									
Age		S	L	P	Age		S	L	P
4	C	−19.0	−7.6	1.5	10	C	0.1	0.1	−0.7
	D	−3.4	0.2	0.5		D	−4.7	0.0	−0.1
5	C	−15.9	−6.4	0.3	11	C	4.4	1.8	−0.3
	D	−3.7	−0.1	0.5		D	−5.3	−0.2	−0.4
6	C	−12.8	−5.1	−0.4	12	C	9.3	3.9	−0.3
	D	−3.6	−0.1	0.5		D	−5.4	−0.3	−0.2
7	C	−10.0	−3.9	−1.0	13	C	14.2	6.1	0.3
	D	−4.2	0.0	0.5		D	−5.5	−0.2	−0.5
8	C	−6.5	−2.5	−1.1	14	C	19.0	7.3	1.4
	D	−4.4	0.1	0.4		D	−5.1	0.2	−0.6
9	C	−3.4	−1.2	−1.1	15	C	22.1	7.8	2.1
	D	−4.6	0.1	0.4		D	−4.4	0.5	−0.5

suitable sparseness using AIC or BIC. A simulation study showed that the parameter matrices can be recovered well by ScTucker2 and the sparseness of the core array can be suitably chosen using BIC, but is underestimated by AIC. Real data examples demonstrated the benefits of using ScTucker2 rather than Tucker2 and Parafac.

In ScTucker2, the sparseness constraint is imposed only upon the core array. If the constraint was also imposed on component matrices **A** and **B**, the interpretation of ScTucker2 solutions would be further facilitated, as we might only focus on the nonzero elements in **A**, **B** and **H** for capturing the relationships of objects to components, those of variables to components, and inter-component relations. A modified Tucker2 procedure with a sparseness-constrained core array and component matrices remains to be developed for future studies.

Another remaining subject may be considering a sparse Tucker2 procedure in which the constraint (2) is not imposed on component matrices. As explained in Section 2, the loss function (13), which is more complex than the function (12) treated in this paper, must be minimized over **H**, if (2) is not imposed. To form an algorithm for that minimization may be an essential task for the subject.

Appendix A. Proof of $\eta = 0$

We can expand (11) as

$$\eta = \text{tr}\,\mathbf{X}'\mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}') - \text{tr}\,\mathbf{X}'\mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}') - \text{tr}(\mathbf{I}_K \otimes \mathbf{B}')'\mathbf{G}'\mathbf{A}'\mathbf{AG}(\mathbf{I}_K \otimes \mathbf{B}') + \text{tr}(\mathbf{I}_K \otimes \mathbf{B}')'\mathbf{G}'\mathbf{A}'\mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}'),$$

which is rewritten as

$$\eta = \text{tr}\,\mathbf{G}'\mathbf{G} - \text{tr}\,\mathbf{G}'\mathbf{H} - \text{tr}\,\mathbf{G}'\mathbf{A}'\mathbf{AG}\{\mathbf{I}_K \otimes (\mathbf{B}'\mathbf{B})\} + \text{tr}\,\mathbf{G}'\mathbf{A}'\mathbf{AH}\{\mathbf{I}_K \otimes (\mathbf{B}'\mathbf{B})\} \tag{A.1}$$

using $(\mathbf{I}_K \otimes \mathbf{B}')(\mathbf{I}_K \otimes \mathbf{B}')' = \mathbf{I}_K \otimes (\mathbf{B}'\mathbf{B})$ and (10). Further, the use of (2) in (A.1) leads to

$$\eta = \text{tr}\,\mathbf{G}'\mathbf{G} - \text{tr}\,\mathbf{G}'\mathbf{H} - \text{tr}\,\mathbf{G}'\mathbf{G} + \text{tr}\,\mathbf{G}'\mathbf{H} = 0.$$

Appendix B. Proof for the decomposition of (7) without using (2)

Loss function (7) can be rewritten as

$$\begin{aligned} f &= \|\mathbf{X} - \mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}') + \mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}') - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 \\ &= \|\mathbf{X} - \mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}')\|^2 - 2\eta^{\#} + \phi^{\#}. \end{aligned}$$

Here,

$$\begin{aligned} \eta^{\#} &= \text{tr}\{\mathbf{X} - \mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}')\}'\{\mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}') - \mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}')\} \\ &= \text{tr}\,\mathbf{\Phi}_1 - \text{tr}\,\mathbf{\Phi}_2 - \text{tr}\,\mathbf{\Phi}_3 + \text{tr}\,\mathbf{\Phi}_4 \end{aligned}$$

with

$$\mathbf{\Phi}_1 = \mathbf{X}'\mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}'), \qquad \mathbf{\Phi}_2 = \mathbf{X}'\mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}'),$$
$$\mathbf{\Phi}_3 = (\mathbf{I}_K \otimes \mathbf{B})\mathbf{G}^{\#\prime}\mathbf{A}'\mathbf{AG}^{\#}(\mathbf{I}_K \otimes \mathbf{B}'), \quad \text{and} \quad \mathbf{\Phi}_4 = (\mathbf{I}_K \otimes \mathbf{B})\mathbf{G}^{\#\prime}\mathbf{A}'\mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}').$$

Using $\mathbf{G}^{\#} = \mathbf{A}^{+}\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B}'^{+})$, we can find $\text{tr}\,\mathbf{\Phi}_1 = \text{tr}\,\mathbf{\Phi}_3$ and $\text{tr}\,\mathbf{\Phi}_2 = \text{tr}\,\mathbf{\Phi}_4$ to have $\eta^{\#} = 0$ as follows:

$$\begin{aligned} \text{tr}\,\mathbf{\Phi}_3 &= \text{tr}(\mathbf{I}_K \otimes \mathbf{B})\mathbf{G}^{\#\prime}\mathbf{A}'\mathbf{AA}^{+}\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B}'^{+})(\mathbf{I}_K \otimes \mathbf{B}') \\ &= \text{tr}\,\mathbf{G}^{\#\prime}(\mathbf{A}'\mathbf{AA}^{+})\mathbf{X}\{\mathbf{I}_K \otimes (\mathbf{B}'^{+}\mathbf{B}'\mathbf{B})\} = \text{tr}\,\mathbf{G}^{\#\prime}\mathbf{A}'\mathbf{X}(\mathbf{I}_K \otimes \mathbf{B}) = \text{tr}\,\mathbf{\Phi}_1, \\ \text{tr}\,\mathbf{\Phi}_4 &= \text{tr}(\mathbf{I}_K \otimes \mathbf{B})(\mathbf{I}_K \otimes \mathbf{B}^{+})\mathbf{X}'\mathbf{A}'^{+}\mathbf{A}'\mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}'). \\ &= \text{tr}\{\mathbf{I}_K \otimes (\mathbf{B}'\mathbf{BB}^{+})\}\mathbf{X}'(\mathbf{A}'^{+}\mathbf{A}'\mathbf{A})\mathbf{H} = \text{tr}(\mathbf{I}_K \otimes \mathbf{B}')\mathbf{X}'\mathbf{AH} = \text{tr}\,\mathbf{\Phi}_2, \end{aligned}$$

where we have used $\mathbf{A}' = \mathbf{A}^{+}\mathbf{AA}'$ (Magnus and Neudecker, 1991, p. 33).

Appendix C. Maximum likelihood Tucker2 and ScTucker2

Let $(\mathbf{AH}_k\mathbf{B}')_{ij}$ denote the (i, j) element of $\mathbf{AH}_k\mathbf{B}'$. Incorporating the assumption (22) for errors into (1) leads to x_{ijk} following $N((\mathbf{AH}_k\mathbf{B}')_{ij}, \sigma^2)$ independently across different i, j, k. Thus, the joint probability density of x_{ijk} over all i, j, k is given by

$$P(\mathbf{X}) = \prod_{i=1}^{I}\prod_{j=1}^{J}\prod_{k=1}^{K} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{[x_{ijk} - (\mathbf{AH}_k\mathbf{B}')_{ij}]^2}{2\sigma^2}\right\}.$$

The log likelihood for $\mathbf{X}$ can thus be written as

$$\begin{aligned} \log L &= \log P(\mathbf{X}) = -\frac{IJK}{2}\log 2\pi - \frac{IJK}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\Sigma_k\|\mathbf{X}_k - \mathbf{AH}_k\mathbf{B}'\|^2 \\ &= -\frac{IJK}{2}\log 2\pi - \frac{IJK}{2}\log\sigma^2 - \frac{1}{2\sigma^2}f, \end{aligned} \tag{C.2}$$

with f being the Tucker2 loss function defined in (7).

The equation $d\log L/d\sigma^2 = 0$ shows that the MLE of the variance satisfies $\sigma^2 = (IJK)^{-1}f$. Substituting this into (C.2) leads to

$$\log L = -\frac{IJK}{2}\log 2\pi - \frac{IJK}{2}\log\frac{f}{IJK} - \frac{IJK}{2} = -\frac{IJK}{2}(\log f + \log 2\pi - \log IJK + 1). \tag{C.3}$$

We can find the maximization of (C.3) to be equivalent to minimizing f or (7). That is, Tucker2 can also be formulated as maximizing (C.3) subject to (2) and incorporating (4) into this formulation leads to ScTucker2.

Using the equation $f = \|\mathbf{X}\|^2[1 - GOF_s(c)]$ derived from (20) and (21), the attained maximum of (C.3) is expressed as

$$\begin{aligned} \log L_{\max} &= -\frac{IJK}{2}\{\log\|\mathbf{X}\|^2[1 - GOF_s(c)] + \log 2\pi - \log IJK + 1\} \\ &= -\frac{IJK}{2}\{\log[1 - GOF_s(c)] + \text{const}\}, \end{aligned} \tag{C.4}$$

i.e., (23), with $\text{const} = \log\|\mathbf{X}\|^2 + \log 2\pi - \log IJK + 1$. The maximum log likelihood for ScTucker2 is given by (C.4) and that for Tucker2 is given by (C.4) with $c = 0$.

Appendix D. BIC for Tucker2

As described in Appendix C, Tucker2 can be formulated as an ML procedure and BIC can thus be defined for the Tucker2 model. Here, it should be cautioned that the Tucker2 model has rotational indeterminacy; its solutions can be rotated as in (27). This implies that a set of the rotation matrices $\mathbf{S}$ and $\mathbf{T}$ in (27) allows the $P_0 = P(P-1)/2$ elements of the rotated core array and its other $Q_0 = Q(Q-1)/2$ elements to take exact zero. Subtracting those numbers from the apparent number of parameters $N = IP + JQ + KPQ$ gives $N_{\mathrm{T2}} = N - P_0 - Q_0$, which is the number of parameters considered in BIC. Therefore, by substituting N_{T2} and $GOF(0)$ for $N - c$ and $GOF(c)$ in (25), BIC for Tucker 2 is expressed as

$$BIC = -IJK\{\log[1 - GOF(0)] + \text{const}\} + N_{\mathrm{T2}} \log(IJK).$$

Appendix E. Upper limit of sparseness

We show that at least one component in $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_P]$ or $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_Q]$ disappears in the model part in the ScTucker2 loss function (7), if $\mathrm{Sp}(\mathbf{H}) = c$ is larger than $KPQ - \max(P, Q)$, or equivalently, $\mathrm{Card}(\mathbf{H}) < \max(P, Q)$, with $\mathrm{Card}(\mathbf{H})$ the number of the nonzero elements in $\mathbf{H}$ (i.e., its cardinality). Here, we note that (7) is also expressed as

$$f = \sum_{k=1}^{K} \left\| \mathbf{X}_k - \sum_{p=1}^{P} \sum_{q=1}^{Q} h_{pqk} \mathbf{a}_p \mathbf{b}'_q \right\|^2 \tag{E.5}$$

with h_{pqk} the (p, q)th element of $\mathbf{H}_k$, and $P = \max(P, Q)$ is supposed without loss of generality.

If $\mathrm{Card}(\mathbf{H}) < P$, then $h_{p^*qk} = 0\,(q = 1, \ldots, Q;\ k = 1, \ldots, K)$ for at least one p^* and (E.5) can be rewritten as

$$f = \sum_{k=1}^{K} \left\| \mathbf{X}_k - \sum_{p \neq p^*} \sum_{q=1}^{Q} h_{pqk} \mathbf{a}_p \mathbf{b}'_q \right\|^2, \tag{E.6}$$

where the p^*th component is found to vanish in the model part: $\mathbf{a}_{p^*}$ is not fitted to data: An example for $P = 3,\ Q = 2,\ K = 2$, and $\mathrm{Card}(\mathbf{H}) = 2 < P$ is given by

$$\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2] = \left[\begin{array}{cc|cc} 0 & \# & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \# & 0 \end{array}\right]:$$

with $p^* = 2$. Then, (E.6) is expressed as $f = \|\mathbf{X}_1 - h_{121}\mathbf{a}_1\mathbf{b}'_2\|^2 + \|\mathbf{X}_2 - h_{312}\mathbf{a}_3\mathbf{b}'_1\|^2$. Here, $\mathbf{a}_2$ does not appear, which implies that the component $\mathbf{a}_2$ is unnecessary.

On the other hand, when $\mathrm{Card}(\mathbf{H})$ is at least P, all components in $\mathbf{A}$ and $\mathbf{B}$ can be fitted to data. An example for $P = 3$, $Q = 2,\ K = 2$, and $\mathrm{Card}(\mathbf{H}) = P$ is given by

$$\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2] = \left[\begin{array}{cc|cc} \# & 0 & 0 & 0 \\ 0 & 0 & 0 & \# \\ 0 & \# & 0 & 0 \end{array}\right]$$

which allows us to rewrite (7) as $f = \|\mathbf{X}_1 - (h_{111}\mathbf{a}_1\mathbf{b}'_1 + h_{321}\mathbf{a}_3\mathbf{b}'_2)\|^2 + \|\mathbf{X}_2 - h_{222}\mathbf{a}_2\mathbf{b}'_2\|^2$. Here, we can find that all components appear. However, $\mathrm{Card}(\mathbf{H}) \geq P$ does not guarantee that all components are fitted to $\mathbf{X}$. For example,

$$\mathbf{H} = [\mathbf{H}_1, \mathbf{H}_2] = \left[\begin{array}{cc|cc} \# & 0 & 0 & 0 \\ 0 & \# & 0 & \# \\ 0 & 0 & 0 & 0 \end{array}\right].$$

Then, $f = \|\mathbf{X}_1 - (h_{111}\mathbf{a}_1\mathbf{b}'_1 + h_{221}\mathbf{a}_2\mathbf{b}'_2)\|^2 + \|\mathbf{X}_2 - h_{222}\mathbf{a}_2\mathbf{b}'_2\|^2$, where $\mathbf{a}_3$ vanishes.

It is not problematic that a component disappears in the model part, in that it implies the component being unnecessary and thus might be regarded as a suggestion for reducing the number of components. However, this subject is beyond our scope. In this paper, we simply avoid the cases where at least a component inevitably disappears in the model, by setting $\mathrm{Card}(\mathbf{H}) > \max(P, Q)$, i.e., $c_{\max} = KPQ - \max(P, Q)$.

Appendix F. Accommodation of solutions

The model part in (7) can be rewritten as

$$\mathbf{AH}(\mathbf{I}_K \otimes \mathbf{B}') = \mathbf{A}\boldsymbol{\Pi}_P\boldsymbol{\Pi}'_P\mathbf{H}(\mathbf{I}_K \otimes \boldsymbol{\Pi}_Q)\{\mathbf{I}_K \otimes (\mathbf{B}\boldsymbol{\Pi}_Q)'\}$$

with $\boldsymbol{\Pi}_P = \boldsymbol{\Theta}_P\boldsymbol{\Omega}_P$. Here, $\boldsymbol{\Theta}_P$ is a $P \times P$ permutation matrix and $\boldsymbol{\Omega}_P$ is a $P \times P$ diagonal matrix with each diagonal element being one or minus one. That is, if $\mathbf{A}_c$, $\mathbf{B}_c$, and $\mathbf{H}_c$ are the optimal solutions of $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{H}$ for a specified c, then $\mathbf{A}_c\boldsymbol{\Pi}_P$, $\mathbf{B}_c\boldsymbol{\Pi}_Q$, and $\boldsymbol{\Pi}'_P\mathbf{H}_c(\mathbf{I}_K \otimes \boldsymbol{\Pi}_Q)$ are also the optimal ones. In the simulation study, we chose $\boldsymbol{\Pi}_P$ that minimizes the average of the absolute values of the differences between the elements of $\mathbf{A}_c\boldsymbol{\Pi}_P$ and the true counterparts. In a parallel manner, $\boldsymbol{\Pi}_Q$ was selected.

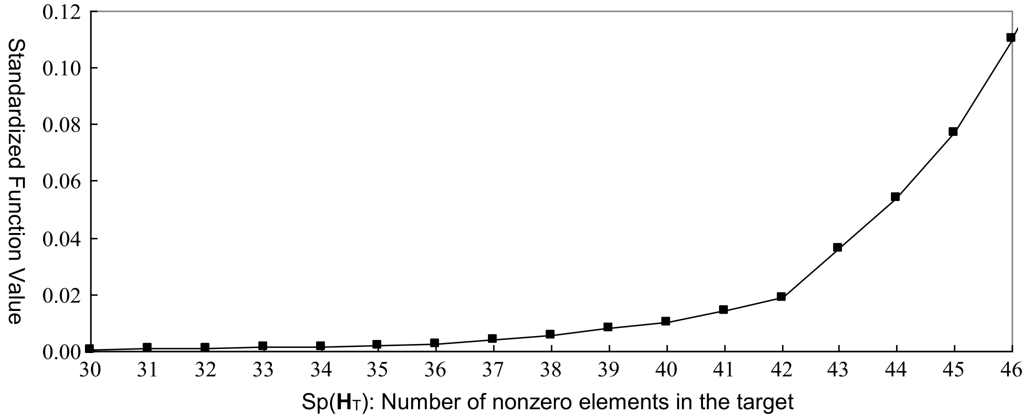

Fig. I.1. Plot of the simplimax function values against $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ for simulated data, with only the range of $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ shown within which a reasonable $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ is found.

Appendix G. Maximum likelihood Parafac

The maximum likelihood (ML) version of Parafac can be formulated merely by incorporating additional constraint (3) into the ML-based Tucker2 in Appendix C, i.e., by replacing $\mathbf{H}_k$ with a diagonal matrix $\mathbf{D}_{k\cdot}$. Thus, the log likelihood for Parafac is expressed as (C.2) with $(\mathbf{AH}_k\mathbf{B}')_{ij}$ replaced by $(\mathbf{AD}_k\mathbf{B}')_{ij}$ and the maximum log likelihood is given by

$$\mathrm{Log}L_{\max}^{[\mathrm{PA}]} = -\frac{IJK}{2}\{\log\|\mathbf{X}\|^2[1 - GOF_{\mathrm{PA}}] + \log 2\pi - \log IJK + 1\}$$
$$= -\frac{IJK}{2}\{\log[1 - GOF_{\mathrm{PA}}] + \mathrm{const}\}. \tag{G.7}$$

Here, $GOF_{\mathrm{PA}} = \|\mathbf{D}\|^2/\|\mathbf{X}\|^2$ with $\mathbf{D} = [\mathbf{D}_1, \ldots, \mathbf{D}_K]$. We can find that (E.5) is equivalent to (C.4), except *GOF*. Parafac does not have rotational indeterminacy and the number of parameters is $N_{\mathrm{PA}} = (I + J + K)P$. We thus have the following BIC for Parafac:

$$BIC = -IJK\{\log[1 - GOF_{\mathrm{PA}}] + \mathrm{const}\} + N_{\mathrm{PA}}\log(IJK),$$

by substituting N_{PA} and GOF_{PA} for $N - c$ and $GOF(c)$ in (25).

Appendix H. Computation time

The computations for ScTucker2 and the simplimax following Tucker2 were performed with the statistical package R (R Development Core Team, 2008) executing on a Core i7-2640M 2.8 GHz computer with 8 GB of memory. We used the function system.time to measure CPU times. As a result, the ScTucker2 and the simplimax procedures were found to take 1232 and one 1322 s, respectively, to provide the solutions over considered multiple values of sparseness, for the data set in Section 7.1. This result shows that ScTucker2 is a bit faster than the simplimax procedure. It was also found for the computations in Section 7.2: the ScTucker2 and simplimax procedures took 663 and 891 s, respectively.

Appendix I. Supplementary simulation for Section 7.1

To empirically test the hypothesis that the simplimax function values suggest choosing $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ greater than the true $\mathrm{Sp}(\mathbf{H})$, we performed the following simulation study: the simplimax solution for $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 42$ (Tables 4 and 5) with the 37 smallest absolute values of core elements replaced by zero was regarded as a set of the true parameter values with the true $\mathrm{Sp}(\mathbf{H}) = 37$. From them, we synthesized a data set through (1′) so that $\|\mathbf{E}\|^2/\|\mathbf{X}\|^2 = 0.1$. For the data set, BIC in ScTucker2 showed the best $\mathrm{Sp}(\mathbf{H})$ being 36 which is close to the true value 37, while the simplimax following Tucker2 gave the function values in Fig. I.1, which suggests to choose $\mathrm{Sp}(\mathbf{H}_\mathrm{T}) = 42$ greater than 37. This result supports our hypothesis. Such a greater $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ may make the recovery of $\mathbf{H}$ worse. Indeed, $\mathrm{AAD}(\mathbf{H})$ (the averaged absolute difference between the resulting and true $\mathbf{H}$ used in Section 5.2) for the simplimax solution was 0.215 which is greater than $\mathrm{AAD}(\mathbf{H}) = 0.129$ for ScTucker2. Further, we substituted zero into the elements of the simplimax $\mathbf{H}$ whose absolute values are less than the 43th smallest so that $\mathrm{Sp}(\mathbf{H})$ was 42 and matched with the chosen $\mathrm{Sp}(\mathbf{H}_\mathrm{T})$ value. The resulting $\mathbf{H}$ showed $\mathrm{AAD}(\mathbf{H}) = 0.168$, which is also greater than 0.129 for ScTucker2, though $\mathrm{AAD}(\mathbf{A})$ and $\mathrm{AAD}(\mathbf{B})$ were equivalent between the simplimax and ScTucker2 solutions.

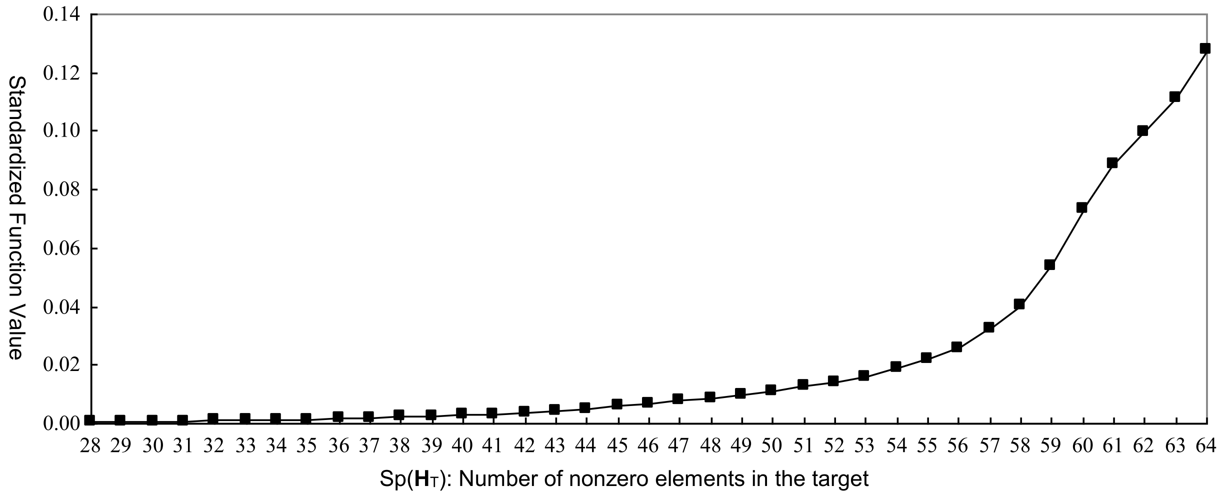

Fig. J.2. Plot of the simplimax function values against $\text{Sp}(\mathbf{H}_T)$ for simulated data, with only the range of $\text{Sp}(\mathbf{H}_T)$ shown within which reasonable $\text{Sp}(\mathbf{H}_T)$ is found.

Appendix J. Supplementary simulation for Section 7.2

To test the hypothesis that the simplimax function values suggest too great $\text{Sp}(\mathbf{H}_T)$, we performed the simulation study, whose procedure is the same as in Appendix I except that the true parameter values were set to those of the simplimax solutions shown in Fig. 4, Table 7, and Table 8 with the 32 smallest absolute values of $\mathbf{H}$ replaced by zeros, i.e., the true $\text{Sp}(\mathbf{H}) = 32$. For the data set, BIC in ScTucker2 showed the best $\text{Sp}(\mathbf{H})$ being 36, which was not very different from the true value, though it was overestimated. On the other hand, the simplimax following Tucker2 gave the function values in Fig. J.2. There, we cannot find a value that is clearly considered to be the best $\text{Sp}(\mathbf{H}_T)$, but for it $\text{Sp}(\mathbf{H}_T) = 58$ can be chosen, as the function value shows a considerable relative increase after $\text{Sp}(\mathbf{H}_T) = 58$, which is much greater than the true $\text{Sp}(\mathbf{H}) = 32$. It supports our hypothesis.

References

Adachi, K., 2011. Three-way Tucker2 component analysis solutions of stimuli × responses × individuals data with simple structure and the fewest core differences. Psychometrika 76, 285–305.
Adachi, K., 2013. A restrained condition number least squares technique with its applications to avoiding rank deficiency. J. Japanese Soc. Comput. Statist. 26, 39–51.
Adachi, K., Trendafilov, N.T., 2015. Sparse principal component analysis subject to prespecified cardinality of loadings. Comput. Statist. (in press) http://link.springer.com/article/10.1007%2Fs00180-015-0608-4.
Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Automat. Control 19, 716–723.
Brouwer, P., Kroonenberg, P.M., 1991. Some notes on the diagonalization of the extended three-mode cores matrix. J. Classification 8, 93–98.
Browne, M.W., 2001. An overview of analytic rotation in exploratory factor analysis. Multivariate Behav. Res. 36, 111–150.
Harshman, R.A., 1970. Foundations of the PARAFAC procedure: Models and conditions for an exploratory multi-mode factor analysis. UCLA Working Papers in Phonetics, Vol. 16, pp. 1–84.
Harville, D.A., 1997. Matrix Algebra from a Statistician's Perspective. Springer, New York.
Janssen, J., Marcotorchino, F., Proth, J.M., 1987. Data Analysis: The Ins and Outs of Solving Real Problems. Plenum, New York.
Jolliffe, I.T., Trendafilov, N.T., Uddin, M., 2003. A modified principal component technique based on the LASSO. J. Comput. Graph. Statist. 12, 531–547.
Kaiser, H.F., 1958. The varimax criterion for analytic rotation in factor analysis. Psychometrika 23, 187–200.
Kiers, H.A.L., 1991. Hierarchical relations among three-way methods. Psychometrika 56, 449–470.
Kiers, H.A.L., 1992. Tuckals core rotations and constrained Tuckals modeling. Statist. Appl. 4, 659–667.
Kiers, H.A.L., 1994. Simplimax: Oblique rotation to an optimal target with simple structure. Psychometrika 59, 567–579.
Kiers, H.A.L., 1997. Three-mode orthomax rotation. Psychometrika 62, 579–598.
Kiers, H.A.L., 1998a. Joint orthomax rotation of the core and component matrices resulting from three-mode principal component analysis. J. Classification 15, 245–263.
Kiers, H.A.L., 1998b. Three-way SIMPLIMAX for oblique rotation of the three-mode factor analysis core to simple structure. Comput. Statist. Data Anal. 28, 307–324.
Kroonenberg, P.M., 1983. Three-Mode Principal Component Analysis: Theory and Applications. DSWO Press, Leiden.
Kroonenberg, P.M., 2008. Applied Multiway Data Analysis. Wiley, Hoboken, NJ.
Magnus, J.R., Neudecker, H., 1991. Matrix Differential Calculus with Applications in Statistics and Econometrics, second ed. Wiley, Chichester.
Murakami, T., Kroonenberg, P.M., 2003. Three-mode models and individual differences in semantic differential data. Multivariate Behav. Res. 38, 247–283.
Osgood, C.E., Luria, Z., 1954. A blind analysis of a case of multiple personality. J. Abnorm. Soc. Psychol. 49, 579–591.
R Development Core Team, 2008. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN: 3-900051-07-0, URL: http://www.R-project.org.
Schwarz, G., 1978. Estimating the dimension of a model. Ann. Statist. 6, 461–464.
Smilde, A., Bro, R., Geladi, P., 2004. Multi-Way Analysis: Applications in the Chemical Sciences. Wiley, Chichester.
ten Berge, J.M.F., 1983. A generalization of Kristof's theorem on the trace of certain matrix products. Psychometrika 48, 519–523.
ten Berge, J.M.F., 1993. Least Squares Optimization in Multivariate Analysis. DSWO Press, Leiden, The Netherlands.
Thurstone, L.L., 1947. Multiple—Factor Analysis: A Development and Expansion of The Vectors of Mind. University of Chicago Press, Chicago.
Trendafilov, N.T., 2014. From simple structure to sparse components: A review. Comput. Statist. 29, 431–454.
Trendafilov, N.T., Adachi, K., 2015. Sparse versus simple structure loadings. Psychometrika 80, 776–790.
Tucker, L.R., 1966. Some mathematical notes on three-mode factor analysis. Psychometrika 31, 279–311.
Zou, D.M., Hastie, T., Tibshirani, R., 2006. Sparse principal component analysis. J. Comput. Graph. Statist. 15, 265–286.