Overlapping clusterwise simultaneous component analysis

De Roover, K.; Ceulemans, Eva; Giordani, Paolo

Published in:
Chemometrics & Intelligent Laboratory Systems

Document version:
Peer reviewed version

DOI:
10.1016/j.chemolab.2016.05.002

Publication date:
2016

Overlapping clusterwise simultaneous component analysis

Kim De Roover

KU Leuven

Eva Ceulemans

KU Leuven

Paolo Giordani

Sapienza University of Rome

Citation:
De Roover, K., Ceulemans, E., & Giordani, P. (in press). Overlapping clusterwise simultaneous component analysis. *Chemometrics and Intelligent Laboratory Systems*.

Author Notes:

Kim De Roover is a post-doctoral fellow of the Fund for Scientific Research Flanders (Belgium). The research leading to the results reported in this paper was sponsored in part by Belgian Federal Science Policy within the framework of the Interuniversity Attraction Poles program (IAP/P7/06), and by the Research Council of KU Leuven (GOA/15/003). Correspondence concerning this paper should be addressed to Kim De Roover, Quantitative Psychology and Individual Differences Research Group, Tiensestraat 102, B-3000 Leuven, Belgium. E-mail: Kim.DeRoover@ppw.kuleuven.be.

Abstract

When confronted with multivariate multiblock data (i.e., data in which the observations are nested within different data blocks that have the variables in common), it can be useful to synthesize the available information in terms of components and to inspect between-block similarities and differences in component structure. To this end, the clusterwise simultaneous component analysis (C-SCA) framework was developed across a series of papers: C-SCA partitions the data blocks into a limited number of mutually exclusive groups and performs separate SCA's per cluster. In this paper, we present a more general version of C-SCA. The key difference with the existing C-SCA methods is that the new method does not impose that the clusters are mutually exclusive, but allows for overlapping clusters. Therefore, the new method is called Overlapping Clusterwise Simultaneous Component Analysis (OC-SCA). Each of these clusters corresponds to a single component, such that all the data blocks that are assigned to a particular cluster have the associated component in common. Moreover, the more clusters a specific data block belongs to, the more complex the underlying component structure. A simulation study and an empirical application to emotion data are included in the paper.

Keywords: clusterwise simultaneous component analysis, SCA-IND, overlapping clustering

1. Introduction

Multivariate multiblock data are a set of matrices that have either the variable (column) mode in common, whereas the entities of the observation mode differ [1], or that have the observation (row) mode in common, whereas the variables differ. Examples of columnwise-coupled multiblock data can be found in several domains of research. In psychology, one may think of multiple emotion ratings of subjects from different age groups, or inhabitants of different countries (e.g., [2, 3]). In chemometrics, multiblock data may contain concentrations of chemical compounds in certain substances in different geographical areas, or measured with different measurement techniques, or from different raw material sources, etcetera (e.g., [4, 5, 6]). In economics, one can think of a questionnaire on work experience administered to workers belonging to different industries or countries (e.g., [7]). In marketing, an example is a survey on the liking of a food item administered to consumers of different countries (e.g., [8]). Examples of rowwise coupled multiblock data include multisource data in chemometrics (e.g., [9]). For the current paper, we will focus on columnwise coupled multiblock data. Adapting the method presented in this paper for rowwise coupled data is a possible direction for future research.

In all of the above cases, it can be useful to synthesize the available information in terms of components and to inspect similarities and differences in the component structures of the data blocks – which we will refer to as the 'within-block structures'. For this purpose, the clusterwise simultaneous component analysis (C-SCA) framework was developed in a series of papers by De Roover and colleagues [1, 10]. C-SCA builds on the assumption that, based on their within-block structure, the data blocks can be partitioned into a few mutually exclusive clusters. The cluster-specific component structures are revealed by applying simultaneous component analysis (SCA) [11, 12] to the data blocks that are assigned to the same cluster. C-SCA encompasses SCA and standard principal component analyses (PCA)

[13, 14] on the separate data blocks as special cases. The former is obtained when the number of clusters amounts to one, the latter when the number of clusters equals the number of blocks.

Several C-SCA variants have been proposed in the literature. One model feature that is varied is which particular SCA variant is used (SCA-ECP [1, 10], SCA-IND [15], or SCA-P [16]), and thus, which restrictions are imposed on the block-specific component variances and correlations. Moreover, variants differ in whether or not the number of extracted components is restricted to be the same across clusters [17]. Finally, a variant has been proposed that allows some of the extracted components to be shared by all clusters (i.e., common components) and thus distinguishes between common and cluster-specific components [18].

In this paper we will develop a more general version of C-SCA. The key principle of the new method is to seek for overlapping clusters, implying that a data block can be assigned to more than one cluster. Therefore, the method is called Overlapping Clusterwise Simultaneous Component Analysis (OC-SCA-IND; the reasons why we apply the SCA-IND restrictions will be elucidated in Section 2). Allowing for overlapping clusters may be helpful in many domains of research. For instance, in a cross-cultural data set, it is reasonable to think that, on the one hand, countries with the same language share a component and, on the other hand, countries with the same religion share another component, whereas countries will partially overlap in terms of religion and language.

Reconsidering the modelling features of the different C-SCA variants, OC-SCA encompasses several C-SCA variants as special cases. Regarding modelling between-block differences in the number of components, in OC-SCA-IND each cluster corresponds to one component. Consequently, the number of clusters to which a data block belongs gives an indication of the complexity of its underlying component structure. With respect to the

common versus cluster-specific nature of components, the number of data blocks that is assigned to a certain cluster reflects how common or specific the corresponding component is, allowing to model different degrees of commonness and specificity.

The paper is organized as follows. In Section 2, SCA-IND and C-SCA-IND are recapitulated. Section 3 is devoted to the new OC-SCA-IND model. The estimation procedure and how to select the optimal number of clusters (which equals the number of components) are discussed in Section 4. Sections 5 and 6 report a simulation study for evaluating the performance of OC-SCA-IND and the results of a real-life application, respectively. In both cases a comparison to the SCA-IND results is included. Finally, Section 7 contains some conclusions and points of discussion.

2. (Clusterwise) Simultaneous Component Analysis models

2.1. Data structure and preprocessing

Columnwise coupled multiblock data consist of I data blocks $\mathbf{X}_i$ ($N_i \times J$), $i = 1, \ldots, I$, containing the scores of N_i observations on J quantitative variables. We can vertically concatenate the data blocks $\mathbf{X}_i$, $i = 1, \ldots, I$, leading to the data matrix $\mathbf{X}$ ($N \times J$), where $N = \sum_{i=1}^{I} N_i$ denotes the total number of observations.

Prior to fitting the model to the data, these are usually preprocessed. Specifically, the data are first centered per data block to remove between-block differences in variable means, allowing us to focus on between-block differences in covariance structure. By scaling the data we subsequently eliminate artificial scale differences between variables. In SCA and C-SCA analysis, two scaling options are frequently used, namely autoscaling [19] and overall scaling [12]. In the former case every variable is normalized per data block (i.e., dividing the centered

data by the block-specific standard deviations), whereas in the latter case the variables are normalized across all data blocks (i.e., dividing by the overall standard deviations). Therefore, autoscaling should be preferred when one wants to focus on the within-block correlation structure, while overall scaling is recommended to inspect the within-block covariance structure. Since the IND version of SCA will be used, which allows for between-block differences in the variances of the components, overall scaling appears to be the most natural choice in this paper.

2.2. SCA-IND

An SCA model is formulated as

$$\mathbf{X}_i = \mathbf{F}_i\mathbf{B}' + \mathbf{E}_i, i = 1, ..., I, \quad (1)$$

where $\mathbf{F}_i$ ($N_i \times Q$) and $\mathbf{B}$ ($J \times Q$) are the component score matrix of data block i and the component loading matrix, respectively, where Q denotes the number of components, and $\mathbf{E}_i$ ($N_i \times J$) is the error matrix of data block i. As stated in the introduction, several variants have been proposed (i.e., SCA-ECP, SCA-IND, SCA-PF2, and SCA-P), that impose different restrictions on the variances and correlations of the block-specific component score matrices (for more details, see [12]). Generally speaking, the more restrictions are imposed, the less between-block differences are allowed for. Therefore, none of the variants is uniformly the best choice. Which variant is selected thus strongly depends on the data set under investigation. In this paper, we focus on SCA-IND (i.e., SCA with INDscal constraints), in which the block-specific component scores are uncorrelated. The variances of the component scores may differ across the blocks, but equal one across all blocks. Unlike SCA-ECP and SCA-P, SCA-IND has no rotational freedom (under mild assumptions), which makes interpretation simpler.

2.3. C-SCA-IND and other C-SCA variants

C-SCA models cluster the data blocks into *K* mutually exclusive groups and formulate a separate SCA model within each cluster. C-SCA [1, 10] was originally formulated as follows:

$$\mathbf{X}_i = \sum_{k=1}^{K} p_{ik}\mathbf{F}_i^{(k)}\mathbf{B}^{(k)\prime} + \mathbf{E}_i, i = 1, ..., I, \quad (2)$$

where $\mathbf{F}_i^{(k)}$ is the component score matrix of data block *i* when assigned to cluster *k*, $\mathbf{B}^{(k)}$ is the component loading matrix of cluster *k*. The matrices $\mathbf{F}_i^{(k)}$ and $\mathbf{B}^{(k)}$ have order ($N_i \times Q$) and ($J \times Q$), respectively, where *Q* denotes the number of cluster-specific components. Finally, the entries p_{ik} of the partition matrix **P** take values 1 (if data block *i* is assigned to cluster *k*) or 0 (otherwise). Moreover, it holds that $\sum_{k=1}^{K} p_{ik} = 1, i = 1, \ldots, I$. Hence, if $K = 1$, then $\mathbf{P} = \mathbf{1}$ (where **1** denotes a column vector of 1's) and C-SCA reduces to SCA.

Although C-SCA-ECP [1, 10] and C-SCA-P versions [16] have been proposed as well, we focus here on the C-SCA-IND variant [15]. This variant has no rotational freedom and, unlike C-SCA-P, forces all important between-block differences in the correlations of the variables to show up in the clustering. Moreover, the often too restrictive C-SCA-ECP assumption of equal component variances – implying that each component gets an equal weight in the solution for each data block – is avoided.

Regarding between-block differences in the complexity of the component structure, C-SCA models generally restrict the number of components to be the same across clusters. Since this assumption is often unrealistic, De Roover et al. [17] proposed a variant that allows for different numbers of cluster-specific components $Q^{(k)}$.

Finally, since all components are cluster-specific, it can be concluded that C-SCA models strongly focus on structural differences. However, in many cases, it is reasonable to expect that next to these differences, there will also be a lot of structural similarity. To better capture both aspects –similarities and differences– a C-SCA variant was proposed that allows for common components, shared by all clusters, as well as cluster-specific ones [18]. This model is formulated as follows:

$$\mathbf{X}_i = \mathbf{F}_{i,comm}\mathbf{B}_{comm}{}' + \sum_{k=1}^{K} p_{ik}\mathbf{F}_{i,spec}^{(k)}\mathbf{B}_{spec}^{(k)}{}' + \mathbf{E}_i, i = 1, ..., I, \quad (3)$$

where the subscripts '*comm*' and '*spec*' indicate 'common' and 'cluster-specific', respectively. $\mathbf{F}_{i,comm}$ ($\mathbf{F}_{i,spec}$) and $\mathbf{B}_{,comm}$ ($\mathbf{B}_{spec}$) are the common (cluster-specific) component score matrix for data block i and common (cluster-specific) component loading matrix, respectively. One drawback of CC-SCA is that the number of common components and cluster-specific ones has to be determined beforehand or selected later on by comparing the fit values of models with different numbers of common and cluster-specific components.

3. OC-SCA-IND model

The key feature of the new OC-SCA-IND model is that the clusters, which each correspond to one component, are allowed to overlap, rather than being mutually exclusive:

$$\mathbf{X}_i = \sum_{k=1}^{K} u_{ik}\mathbf{f}_i^{(k)}\mathbf{b}^{(k)\prime} + \mathbf{E}_i, i = 1, ..., I, \quad (4)$$

where $\mathbf{f}_i^{(k)}$ is the component score vector of data block i assigned to cluster k (i.e., the scores of the N_i observations in block i on the kth component) and $\mathbf{b}^{(k)}$ contains the loadings of the J variables on the component associated with cluster k. The vectors $\mathbf{f}_i^{(k)}$ and $\mathbf{b}^{(k)}$ have length N_i and J, respectively. Finally, u_{ik} is the generic entry of the binary overlapping matrix $\mathbf{U}$ ($I \times K$),

which takes values 1 (if data block *i* is assigned to cluster *k*) or 0 (implying that the *k*th component does not underlie data block *i*). When u_{ik} equals 0, we impose that $\mathbf{f}_i^{(k)} = \mathbf{0}_{N_i}$, where $\mathbf{0}_{N_i}$ denotes a vector of zeroes of length N_i. The overlapping nature of the clustering implies that each block can belong to multiple clusters: $\sum_{k=1}^{K} u_{ik} \geq 1, i = 1, \ldots, I$.

Consistently with SCA-IND, the components are uncorrelated per data block. Specifically, if we let $\mathbf{F}_i$ be the matrix of the component scores for block *i* obtained by juxtaposing next to each other the $\mathbf{f}_i^{(k)}$'s ($\mathbf{F}_i = \left[\mathbf{f}_i^{(1)} \quad \cdots \quad \mathbf{f}_i^{(K)}\right]$, *i* = 1,…, *I*), we impose the constraint $N_i^{-1}\mathbf{F}_i'\mathbf{F}_i = \mathbf{D}_i^2$, $i = 1, \ldots, I$, where $\mathbf{D}_i$ is a diagonal matrix holding the standard deviations of the component scores of block *i*. By letting the component variances vary across blocks, we take into account that the importance of a component may vary across the data blocks for which it is relevant. The orthogonality restrictions are useful from an interpretational as well as an estimation point of view, as we will explain. Note that the overlapping clusterwise models using the other SCA variants can be obtained by replacing these constraints by the ones associated with the desired variant [12].

Regarding between-block differences in the complexity of the underlying component structure, OC-SCA-IND extracts only one component per cluster. At first glance, this may appear to be a limitation, but one should note that this choice can be made without loss of generality, because of the overlapping nature of the clustering. If a specific subset of data blocks have two components in common that are not relevant for other data blocks, OC-SCA-IND deals with this by assigning all these data blocks to two clusters that correspond to the two components involved.

Regarding the common versus cluster-specific nature of the components, OC-SCA-IND allows to model all degrees of commonness. To further clarify this, let us consider the following example with *I* = 8 data blocks and *K* = 5 clusters. All the blocks belong to Cluster

1 and the remaining four clusters are composed by subsets of blocks (data blocks $\mathbf{X}_1$-$\mathbf{X}_4$ are assigned to Clusters 2 and 3, $\mathbf{X}_3$-$\mathbf{X}_6$ to Cluster 4 and $\mathbf{X}_7$-$\mathbf{X}_8$ to Cluster 5). It follows that

$$\mathbf{U} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

According to Equation 5 and taking into account Equation 4, the decomposition of the total data matrix $\mathbf{X}$ can be rewritten as (we omit the subscript for the $\mathbf{0}$ vectors):

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \mathbf{X}_3 \\ \mathbf{X}_4 \\ \mathbf{X}_5 \\ \mathbf{X}_6 \\ \mathbf{X}_7 \\ \mathbf{X}_8 \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1^{(1)} & \mathbf{f}_1^{(2)} & \mathbf{f}_1^{(3)} & \mathbf{0} & \mathbf{0} \\ \mathbf{f}_2^{(1)} & \mathbf{f}_2^{(2)} & \mathbf{f}_2^{(3)} & \mathbf{0} & \mathbf{0} \\ \mathbf{f}_3^{(1)} & \mathbf{f}_3^{(2)} & \mathbf{f}_3^{(3)} & \mathbf{f}_3^{(4)} & \mathbf{0} \\ \mathbf{f}_4^{(1)} & \mathbf{f}_4^{(2)} & \mathbf{f}_4^{(3)} & \mathbf{f}_4^{(4)} & \mathbf{0} \\ \mathbf{f}_5^{(1)} & \mathbf{0} & \mathbf{0} & \mathbf{f}_5^{(4)} & \mathbf{0} \\ \mathbf{f}_6^{(1)} & \mathbf{0} & \mathbf{0} & \mathbf{f}_6^{(4)} & \mathbf{0} \\ \mathbf{f}_7^{(1)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{f}_7^{(4)} \\ \mathbf{f}_8^{(1)} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{f}_8^{(4)} \end{bmatrix} \begin{bmatrix} \mathbf{b}^{(1)\prime} \\ \mathbf{b}^{(2)\prime} \\ \mathbf{b}^{(3)\prime} \\ \mathbf{b}^{(4)\prime} \\ \mathbf{b}^{(5)\prime} \end{bmatrix} + \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \mathbf{E}_3 \\ \mathbf{E}_4 \\ \mathbf{E}_5 \\ \mathbf{E}_6 \\ \mathbf{E}_7 \\ \mathbf{E}_8 \end{bmatrix}. \tag{6}$$

Since all the data blocks are assigned to Cluster 1, we can conclude that the associated component is common to all the data blocks. The first four blocks also constitute Clusters 2 and 3. The associated components are thus cluster-specific because they explain only a subset of data blocks. The structure of data blocks $\mathbf{X}_3$ and $\mathbf{X}_4$ is more complex, however, than that of data blocks $\mathbf{X}_1$ and $\mathbf{X}_2$. Therefore, $\mathbf{X}_3$ and $\mathbf{X}_4$ are also assigned to Cluster 4, next to data blocks $\mathbf{X}_5$ and $\mathbf{X}_6$. The associated component has the same degree of commonness as the components of clusters 2 and 3, since they are all relevant for four data blocks. Finally, data blocks $\mathbf{X}_7$ and $\mathbf{X}_8$ are assigned to Clusters 1 and 5. The fifth component therefore is the least common, since it only plays a role for two data blocks. It is important to note that, unlike CC-SCA, it is not

necessary to choose the nature of the components a priori (e.g., fit a model with three common components and two cluster-specific ones). Instead, the nature of the components can be determined post hoc by inspecting the binary overlapping matrix **U**.

It should be clear that OC-SCA-IND encompasses SCA-IND and C-SCA-IND as special cases. Specifically, OC-SCA-IND is equivalent to C-SCA-IND if all columns of **U** are either identical or non-overlapping to the other columns. Moreover, if all **U** entries equal one, OC-SCA-IND boils down to SCA-IND. Therefore, one may doubt the added value of OC-SCA-IND over SCA-IND since, in SCA-IND, the block-specific component variances will in theory equal zero when a component is irrelevant to a certain data block. However, in practice, component variances will almost never equal zero in SCA-IND, as we will illustrate in Sections 5 and 6. Consequently, in SCA-IND, the component loadings may also be different than in OC-SCA-IND, since every data block has some influence on every component.

4. Model estimation and model selection

4.1. Objective function

We propose to use a penalized loss function when fitting OC-SCA-IND solutions with pre-specified numbers of clusters *K*. Without imposing a penalty, all data blocks are assigned to all clusters (yielding a SCA-IND model), because each component will account for some variance in each block. Building on [17], an *AIC*-based [20] loss function will be used (regarding the choice of *AIC*, see footnote 2 in [17]):

$$AIC = -2\text{loglik}(\mathbf{X} \mid \mathbf{M}) + 2\,fp, \tag{7}$$

where loglik(**X**|**M**) refers to the loglikelihood of data **X** given model **M** and *fp* denotes the number of free parameters to be estimated. Assuming the residuals $e_{n_i j}$ to be independent and identically distributed as $e_{n_i j} \sim N(0,\sigma^2)$, the OC-SCA-IND loglikelihood reads as follows:

$$\text{loglik}(\mathbf{X}\,|\,\mathbf{M}) = \log\left[\left(\frac{1}{2\pi\sigma^2}\right)^{\frac{NJ}{2}} \exp\left(-\frac{SSE}{2\sigma^2}\right)\right] = -\frac{NJ}{2}\log\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}SSE, \tag{8}$$

given that *SSE* is defined as

$$SSE = \sum_{i=1}^{I}\left\|\mathbf{X}_i - \sum_{k=1}^{K} u_{ik}\mathbf{f}_i^{(k)}\mathbf{b}^{(k)\prime}\right\|^2, \tag{9}$$

where ||·|| denotes the Frobenius norm. Using $\hat{\sigma}^2 = \frac{SSE}{NJ}$ as a post-hoc estimator of the error variance σ^2 [21], the loglikelihood becomes:

$$\begin{aligned}\text{loglik}(\mathbf{X}\,|\,\mathbf{M}) &= -\frac{NJ}{2}\log\left(\frac{2\pi SSE}{NJ}\right) - \frac{NJ}{2} \\ &= -\frac{NJ}{2}\left[1 + \log\left(2\pi\right) - \log\left(NJ\right) + \log\left(SSE\right)\right],\end{aligned} \tag{10}$$

where the first three terms are invariant across solutions and thus can be discarded during estimation.

The number of free parameters *fp* is given by:

$$fp = JK + \sum_{k=1}^{K}\left(N^{(k)} - 1\right) - \sum_{i=1}^{I}\left(Q_i(Q_i - 1)/2\right), \tag{11}$$

where Q_i indicates the number of components that are relevant for block *i* (i.e., $\sum_{k=1}^{K} u_{ik}$) and $N^{(k)}$ the total number of observations in cluster *k* (i.e., $N^{(k)} = \sum_{i=1}^{I} u_{ik}N_i$). The first and second terms of Equation 11 refer to the number of component loadings and scores respectively. The second term takes the restriction that each component has a variance of one over all blocks in the corresponding cluster into account. The third term corrects for the orthogonality restriction

per data block. The number of component loadings is invariant during model estimation and is thus discarded. Combining the remaining terms from Equations 10 and 11, the following penalized loss function L is obtained:

$$L = N\,J\log\left(SSE\right) + 2\left[\sum_{k=1}^{K}\left(N^{(k)}-1\right) - \sum_{i=1}^{I}\left(Q_i(Q_i-1)/2\right)\right]. \quad (12)$$

4.2. Model estimation

Building on the SCA-IND algorithm discussed in [12], we developed the following OC-SCA-IND algorithm.

1. Initialization:
    a. Randomly initialize the binary overlapping clustering matrix **U**, by sampling (with replacement) I cluster membership patterns from all possible patterns (excluding the pattern with zero assignments). If some of the obtained clusters are empty, sampling is repeated.
    b. For each data block i, the diagonal matrix $\mathbf{D}_i$ containing the block-specific component standard deviations (SD) is initialized by setting the standard deviations to one if the block is assigned to the corresponding cluster and to zero otherwise.
    c. Initialize all cluster-specific loading vectors $\mathbf{b}^{(k)}$, by conducting, for each cluster k, the Singular Value Decomposition (SVD) on the vertical concatenation $\mathbf{X}^{(k)}$ of the data blocks in that cluster: $\mathbf{X}^{(k)} = \mathbf{R}^{(k)}\mathbf{S}^{(k)}\mathbf{V}^{(k)\prime}$. Next, set $\mathbf{b}^{(k)}$ to $\sqrt{1/N^{(k)}}\mathbf{s}_1^{(k)}\mathbf{v}_1^{(k)}$ where $\mathbf{S}_1^{(k)}$ and $\mathbf{V}_1^{(k)}$ indicate the highest singular value and the associated right singular vector, respectively. Of course, strongly

overlapping clusters will have very similar loadings, but this is solved in the following steps.

2. Update the cluster-specific component scores and loadings. To this end, the following two substeps are iterated until the loss function L no longer decreases according to the convergence criterion ε (e.g., 1e × 10^{-6}).

   a. To update the component scores $\mathbf{f}_i^{(k)}$, the matrix $\mathbf{F}_i^*$, which is a reduced version of $\mathbf{F}_i$, containing only the scores on the components corresponding to the clusters to which the data block is assigned, is decomposed as $\mathbf{F}_i^* = \mathbf{P}_i^* \mathbf{D}_i^*$, where $\mathbf{P}_i^*$ holds the normalized component scores (i.e., with variances equal to one) and $\mathbf{D}_i^*$ the standard deviations (on the diagonal) of the components that are underlying data block i (i.e., $u_{ik} = 1$).

      i. Based on the SVD $\mathbf{X}_i \mathbf{B}^* \mathbf{D}_i^* = \mathbf{R}_i \mathbf{S}_i \mathbf{V}_i'$, $\mathbf{P}_i^*$ is updated as $\mathbf{P}_i^* = \sqrt{N_i}\mathbf{R}_i \mathbf{V}_i$. Note that $\mathbf{B}^*$ contains the loadings on the components which are applicable to data block i according to the clustering.[1]

      ii. The vector of component SD's $\mathbf{d}_i^*$ is computed by the regression step $\mathbf{d}_i^* = \left((\mathbf{G}'\mathbf{G})^{-1}\mathbf{G}'vec(\mathbf{X}_i)\right)$ with $\mathbf{G} = \begin{bmatrix} \mathbf{P}_i^* \bullet \mathbf{B}_1^* \\ \vdots \\ \mathbf{P}_i^* \bullet \mathbf{B}_j^* \\ \vdots \\ \mathbf{P}_i^* \bullet \mathbf{B}_J^* \end{bmatrix}$ where $\bullet$ denotes the elementwise product and $\mathbf{B}_j^*$ equals $\mathbf{1}_{N_i}\mathbf{b}_j^*$, with $\mathbf{1}_{N_i}$ denoting the

[1] OC-SCA-ECP is obtained by imposing that $\mathbf{D}_i^*$ is equal to an identity matrix for each data block and thus by performing the SVD $\mathbf{X}_i \mathbf{B}^* = \mathbf{R}_i \mathbf{S}_i \mathbf{V}_i'$ in this step. OC-SCA-P is obtained by updating the component scores using constrained least squares to impose $\mathbf{f}_i^{(k)} = \mathbf{0}_{N_i}$ when u_{ik} is equal to zero (see selectivity constraints in [22]). The penalty in the loss function (Equation 12) should be attuned accordingly, to properly account for these restrictions.

column vector of ones with length N_i and $\mathbf{b}_j^*$ equal to the j-th row of $\mathbf{B}^*$ [12]. When this regression step has been conducted for all data blocks, the resulting standard deviations are rescaled per cluster, so that the standard deviations across all the blocks within the cluster equal one[2]. This rescaling does not affect the loss function, because it can be compensated for in the loadings **B**. The loadings are not explicitly rescaled in Step 2a, however, because they are updated in Step 2b.

iii. $\mathbf{F}_i^*$ is calculated as $\mathbf{F}_i^* = \mathbf{P}_i^* \mathbf{D}_i^*$. Because $\mathbf{P}_i^*$ is columnwise orthonormal, $\mathbf{F}_i^*$ will be columnwise orthogonal.

iv. Insert the $\mathbf{F}_i^*$ estimates into the $\mathbf{F}_i$ matrices, which are vertically concatenated into the total component score matrix **F**.

b. The cluster-specific loading vectors $\mathbf{b}^{(k)}$ can be updated all at once by means of the regression step $\mathbf{B} = \left((\mathbf{F}'\mathbf{F})^{-1}\mathbf{F}'\mathbf{X}\right)'$, where **B** refers to the horizontal concatenation of the cluster-specific loading vectors $\mathbf{b}^{(k)}$.

3. Update the clustering matrix **U**. This update is done row per row (i.e., for each block separately) using a so-called 'greedy' approach [23]. First, evaluate all cluster membership patterns in which the block is assigned to one single cluster – e.g., if K = 3, [1 0 0], [0 1 0], and [0 0 1] – updating the component scores and loadings accordingly (i.e., performing a limited number of iterations of steps 2a and 2b). Retain the pattern with the lowest loss function value L. Next, evaluate in the same way whether it is beneficial to assign the block to an additional cluster. For instance, if the optimal assignment to a single cluster

[2] At this point, it is also checked whether some of the block-specific component SD's are smaller than 1e × 10^{-9}). Conceptually, this would imply that the block is assigned to a cluster while the corresponding component is not relevant to the block in question, which may occur, for instance, right after the random initialization of **U**. Technically, this would cause singularity or near-singularity problems. Thus, if this is the case, the SD in question is put to zero as well as the corresponding element of the clustering matrix **U**.

was [1 0 0], evaluate the patterns [1 1 0] and [1 0 1]. Retain the one with the best loss function value, and so on. Due to the penalty in the loss function, the loss function value may increase when adding an extra assignment, however, indicating that the increase in fit does not outweigh the increase in complexity. When this occurs, discard such additional assignments and cease the greedy update of the row. If the obtained loss function value after updating all cluster memberships is higher than the value before the update, the greedy approach failed, and an optimal update is performed instead, in which all $(2^K - 1)$ possible cluster memberships are evaluated for each block and the best one is retained[3].

4. Check for empty clusters. If one (or more) clusters are empty after step 3, an assignment to this empty cluster is tentatively added for each data block, updating the components by means of one iteration of substeps 2a and 2b. The data block for which this extra assignment is the least detrimental, is added to this cluster.
5. Repeat steps 2 to 4 until the loss function L no longer decreases according to the convergence criterion ε.

To reduce the risk of ending up in a local minimum, a multistart procedure with different random initializations of the clustering matrix **U** is used and the best-fitting solution (i.e., with the lowest L) is retained as the final solution.

4.3. Model selection

When using the algorithm described above, the number of clusters K has to be specified. Of course, the most appropriate number of clusters is in most cases unknown when analyzing real data and model selection needs to be performed. To this end, one may fit OC-

[3] An algorithm in which all clustering updates were optimal was also evaluated in the simulation study (Section 5), but it performed almost identical to the greedy one, whereas the computation time was more than three times longer. Therefore, we only consider the 'greedy algorithm' in the remainder of the paper.

SCA-IND models with different numbers of clusters *K* and use the scree test [24] to decide on the best number of clusters 'K^{best}' in terms of balance between model fit and complexity. Specifically, the goal of the scree test is to determine the number of clusters after which the increase in fit with additional clusters levels off and this is done by looking for an elbow in the scree plot. As a fit measure, we use the percentage of variance accounted for (*VAF*). Since the data is centered per data block, the *VAF* may be expressed as

$$VAF = \frac{\|\mathbf{X}\|^2 - SSE}{\|\mathbf{X}\|^2} \times 100\%. \tag{13}$$

Due to the overlapping nature of the clustering, the *VAF* will vary more irregularly in function of the number of clusters and, thus, the scree line may sometimes decrease. Therefore, we use the CHULL procedure (for more details, see [25-27]; for software, see [28]) to perform the scree test, which first looks for the convex hull of the scree plot and then selects the solution on the upper boundary of the hull that maximizes the following scree ratio:

$$sr_{(s)} = \frac{VAF_s - VAF_{s-1} / K_s - K_{s-1}}{VAF_{s+1} - VAF_s / K_{s+1} - K_s}, \tag{14}$$

where *s* refers to the *s*th solution on the hull[4]. In addition to using the CHULL procedure, one may also rely on a priori knowledge about the data or on the interpretability of the different models.

Given the *AIC*-based nature of the objective function (Equation 12), it may seem straightforward to use the *AIC* for model selection as well. However, for clusterwise SCA, it has been demonstrated that *AIC* performs badly with a strong tendency to overestimate the number of clusters to the extent that the highest number of clusters is usually selected [17].

[4] One could argue to use the number of free parameters (Equation 11) as the complexity of the models in the CHULL procedure, instead of the number of clusters. Problems with this strategy have been reported for other clusterwise SCA models [17], however, and a pilot study of this strategy for OC-SCA-IND indicated an inferior performance as well.

5. Simulation Study

In this section, a simulation study is discussed, aiming to evaluate the performance of the OC-SCA-IND algorithm and to examine its added value over the standard SCA-IND approach. Additionally, the performance of the CHULL procedure for selecting the number of overlapping clusters is assessed.

5.1. Design

In this simulation study, the number of variables J was fixed at 12. Furthermore, five factors were systematically varied in a complete factorial design:

1. the *number of data blocks I* at two levels: 20, 40[5];
2. the *number of observations per data block* N_i at three levels: $N_i \sim \mathrm{U}[15;20]$, $N_i \sim \mathrm{U}[30;70]$, $N_i \sim \mathrm{U}[80;120]$, with U indicating a discrete uniform distribution between the given numbers;
3. the *number of clusters K* at three levels: 2, 4, 6;
4. the *probability* P_{overlap} *that a data block belongs to more than one cluster* at three levels: .25, .50, .75;
5. the *error level e*, which is the expected proportion of error variance in the data blocks: .20, .40, .60.

[5] For a few replications per cell of the design, we also evaluated the performance of OC-SCA-IND and SCA-IND for only 10 data blocks. This decreased the performance of both methods proportionally (i.e., proportionally to the reported performance for 20 and 40 data blocks).

For each simulated data set, the clustering matrix **U** was generated by, first, splitting all possible cluster membership patterns into the overlapping and the non-overlapping ones, where the number of overlapping and non-overlapping ones is indicated by R_o and R_{no}, respectively. Then, I multinomial random numbers were sampled, indicating the different cluster membership patterns, with the multinomial probabilities equal to $P_{overlap}/R_o$ for the overlapping patterns and $(1 - P_{overlap})/R_{no}$ for the non-overlapping ones. Next, **U** was obtained by vertically concatenating the sampled cluster membership patterns in a random order[6].

The $J \times K$ loading matrix **B** was obtained by sampling the loadings uniformly between −1 and 1 and by rowwise rescaling them such that each row of **B** has a sum of squares equal to one. Each component score matrix $\mathbf{F}_i$ was randomly sampled from a multivariate normal distribution, with a mean vector of zeros and a diagonal variance-covariance matrix with the variances sampled between .25 and 1.75. Note that the scores on components that correspond to clusters to which the data block is not assigned were equal to zero. The residuals $\mathbf{E}_i$ were sampled from a standard normal distribution.

Next, the elements of **B** were multiplied by $\sqrt{1-e}$ whereas each $\mathbf{E}_i$ was rescaled by $\sqrt{e}$. Because **B** was (re)scaled over all components, and thus over clusters, a data block would only have an expected structural variance of $(1-e)$ when it was assigned to all clusters. This has two important consequences for the simulated data: (1) data blocks with more cluster assignments would have more structural variance and thus a more favorable expected error ratio, and (2) the expected structural variance over all data blocks would be influenced by the total number of cluster assignments. The former represents a realistic situation, since in real data sets the error ratio may also differ between the data blocks. The latter would cause the overall error ratio to be larger for data sets with more clusters (factor 3)

[6] This resulted in one common component (i.e., corresponding to a cluster containing all data blocks) for 29 out of 3,240 data sets and in two common components for two data sets.

and/or less cluster overlap (factor 4); thus, to safeguard the intended effect of factor 5, the error was rescaled once more to ensure that the overall error ratio was the required one (note that the between-block differences in error ratio are retained).

For each cell of the factorial design, 20 data matrices **X** were generated, yielding 3,240 data sets in total. Each data block $\mathbf{X}_i$ was columnwise centered and each data matrix **X** was columnwise rescaled to obtain unit variances over all data blocks.

To evaluate model estimation performance, each data matrix **X** is analyzed with the OC-SCA-IND algorithm, applying a convergence criterion ε equal to $1e \times 10^{-6}$ and using 25 random starts. To demonstrate the added value of OC-SCA-IND over SCA-IND, we also performed an SCA-IND analysis with *K* components and the same convergence criterion. Furthermore, to assess model selection performance of the proposed scree test, OC-SCA-IND models with one to eight clusters are estimated for each data matrix **X**, each time with 25 random starts, and the scree test was conducted. To repress the computational burden, these analyses are confined to the first five replications of each cell of the design.

5.2. Results

5.2.1. Model Estimation

We first discuss the sensitivity of the OC-SCA-IND algorithm to local minima. Then, we scrutinize the goodness-of-recovery of the clustering, the loadings and the block-specific component variances; for the latter two we also report the SCA-IND results. Finally, we inspect computation time.

5.2.1.1. Sensitivity to local minima

Even though we applied a multistart approach using 25 random starts, the retained solution may still be a local minimum. To evaluate the sensitivity of the algorithm to local

minima, the loss function value of the retained solutions (i.e., the best solution out of the 25 random starts) should be compared to that of the global minimum. Because the simulated data are perturbed with error and because sampling fluctuations can cause deviations from the OC-SCA-IND assumptions (e.g., orthogonality of the components per data block), the global minimum is unknown, however. Therefore, we used the solution that results from seeding the algorithm with the true clustering matrix **U** as a proxy of the global minimum. Specifically, we considered a solution to be a local minimum when the loss function is higher than that of the proxy and the associated clustering matrices differ. Only 38 local minima were found, i.e., for 1.17% of the simulated data sets. Most of these, i.e., 34, occurred in the conditions with six clusters.

5.2.1.2. Goodness-of-cluster-recovery

To evaluate how well the OC-SCA-IND algorithm recovers the true clustering matrix $\mathbf{U}^{\mathrm{T}}$, we calculated the proportion of correctly recovered cluster assignments (*PCCA)*:

$$PCCA = 1 - \frac{\sum_{i=1}^{I}\sum_{k=1}^{K}\left|u_{ik}^{\mathrm{T}} - u_{ik}^{\mathrm{M}}\right|}{I \times K}, \tag{15}$$

where u_{ik}^{T} and u_{ik}^{M} refer to the elements of the true and estimated clustering matrices $\mathbf{U}^{\mathrm{T}}$ and $\mathbf{U}^{\mathrm{M}}$, respectively. To deal with the permutational freedom of the clusters, the *PCCA* was computed for all possible permutations of $\mathbf{U}^{\mathrm{M}}$ and the permutation that maximized the *PCCA* was retained. The overall mean *PCCA* equals .97 (*SD* = 0.04), with a minimum of .70. It is noteworthy that all *PCCA*-values smaller than .97 occurred in the conditions with only 15 to 20 observations per data block, six clusters and 60% error variance.

5.2.1.3. Goodness-of-loading-recovery

To quantify the goodness-of-loading recovery (*GOLR*), we calculated the following statistics:

$$GOLR_{\text{mean}} = \frac{\sum_{k=1}^{K} \varphi\left(\mathbf{b}^{(k)\text{T}}, \mathbf{b}^{(k)\text{M}}\right)}{K} \quad \text{and} \quad GOLR_{\text{min}} = \min_{k}\left(\varphi\left(\mathbf{b}^{(k)\text{T}}, \mathbf{b}^{(k)\text{M}}\right)\right) \tag{16}$$

with φ indicating the congruence coefficient[7] [29] and $\mathbf{b}^{(k)\text{T}}$ and $\mathbf{b}^{(k)\text{M}}$ denoting the component corresponding to the *k*th true and estimated cluster, respectively. The $GOLR_{\text{mean}}$ statistic quantifies the mean recovery over all components, whereas the $GOLR_{\text{min}}$ corresponds to the component with the worst recovery. The best permutation of $\mathbf{U}^{\text{M}}$ (see Section 5.2.1.2.) was used to permute the estimated components before calculating the *GOLR* value. $GOLR_{\text{mean}}$ and $GOLR_{\text{min}}$ take values between zero (no recovery at all) and one (perfect recovery), and – according to Lorenzo-Seva and ten Berge [30] – two components can be considered identical when their congruence coefficient is above .95. On average, $GOLR_{\text{mean}}$ has a value of .99 (*SD* = 0.03) whereas $GOLR_{\text{min}}$ takes on a value of .97 (*SD* = 0.11). $GOLR_{\text{min}}$ is smaller than .95 for 302 out of the 3,240 data sets, whereas 235 out of these 302 occurred in the conditions with 60% error variance.

Regarding SCA-IND, the $GOLR_{\text{mean}}$ and $GOLR_{\text{min}}$ of the SCA-IND loadings[8] amount to .97 (*SD* = 0.06) and .91 (*SD* = 0.21), on average, which is worse than those for OC-SCA-IND. Moreover, the $GOLR_{\text{mean}}$ and $GOLR_{\text{min}}$ of SCA-IND is lower than the one for OC-SCA-IND for no less than 2,912 (i.e., 90%) and 2,839 (i.e., 88%) out of the 3,240 data sets, respectively.

[7] The congruence coefficient [29] between two column vectors **x** and **y** is defined as their normalized inner product: $\varphi_{\mathbf{xy}} = \frac{\mathbf{x}'\mathbf{y}}{\sqrt{\mathbf{x}'\mathbf{x}}\sqrt{\mathbf{y}'\mathbf{y}}}$.

[8] For SCA-IND, the permutation of the estimated loadings maximizing $GOLR_{mean}$ was used for computing $GOLR_{mean}$ and $GOLR_{min}$.

5.2.1.4. Goodness-of-component-variance-recovery

To quantify how well the block-specific variances are recovered, we calculated the mean-squared-difference (*MSD*) between the true block-specific variances (including the zeros according to the true clustering) and the estimated block-specific variances (including the zeros according to the estimated clustering), using the best permutation of $\mathbf{U}^{M}$ (see Section 5.2.1.2). On average, the *MSD* was equal to 0.02 (*SD* = 0.03). With respect to the manipulated factors, *MSD* depends most on the number of observations per data block – mean *MSD* equal to 0.03, 0.01, and 0.01 for data sets with 15 to 20, 30 to 70, and 80 to 120 observations per data block, respectively – the error level of the data – mean *MSD* equal to 0.01, 0.01, and 0.04 in case of 20%, 40%, and 60% error variance, respectively – and, of course, whether or not the clustering is recovered correctly – mean *MSD* equal to 0.004 in case of a perfectly recovered clustering and 0.03 otherwise.

For SCA-IND, the *MSD* is, on average, equal to 0.23 (*SD* = 0.15), which is markedly higher than that of OC-SCA-IND. Here, the *MSD* depends mostly on the number of clusters – mean *MSD* equal to 0.07, 0.25, and 0.36 for two, four and six clusters, respectively – but also on the error level – mean *MSD* equal to 0.16, 0.22, and 0.30 for 20%, 40%, and 60% error – and the amount of cluster overlap – mean *MSD* equal to 0.29, 0.22, and 0.17 for the respective levels of cluster overlap. The estimates of the block-specific component variances that equal zero in the true data amount to 0.49 on average. These findings are probably due to the fitting of error variance, since, on average, the SCA-IND VAF is 9% larger than for OC-SCA-IND.

Another way of looking at the recovery of the component variances is quantifying how the relative differences between high and low (possibly zero) component variances are preserved in the estimated component variances for OC-SCA-IND and SCA-IND. To this

end, $GOVR_{mean}$ and $GOVR_{min}$ values were calculated, both for OC-SCA-IND and SCA-IND, where *GOVR* refers to 'goodness-of-variance-recovery'. These statistics are calculated as in Equation 16, replacing the loading vectors $\mathbf{b}^{(k)\mathrm{T}}$ and $\mathbf{b}^{(k)\mathrm{M}}$ by the $I \times 1$ vectors containing the true and estimated block-specific component variances for cluster *k*. For OC-SCA-IND, the average $GOVR_{mean}$ and $GOVR_{min}$ amount to .98 (*SD* = .05) and .95 (*SD* = .12), respectively. For SCA-IND, they amount to the markedly lower .90 (*SD* = .09) and .85 (*SD* = .15), respectively. The correlations between $GOVR_{mean}$ ($GOVR_{min}$) and $GOLR_{mean}$ ($GOLR_{min}$) are .87 (.86) and .76 (.78) for OC-SCA-IND and SCA-IND, respectively, indicating that – especially for SCA-IND – the recovery of the block-specific component variances is partly but not entirely explained by the recovery of the component loadings.

5.2.1.5. Computation time

The analyses were performed on a supercomputer consisting of INTEL XEON L5420 processors with a clock frequency of 2.5 GHz and with 8 GB RAM and took about 16 minutes per data set. The computation time is mostly influenced by the number of data blocks and the number of clusters. Specifically, the mean computation time was 6 minutes for 20 data blocks and 25 minutes for 40 data blocks, whereas the mean computation times for two, four and six clusters were 2, 12 and 33 minutes, respectively.

5.2.2. Model Selection

On average, the CHULL procedure selected the correct number of clusters for 590 or about 73% of the 810 data sets included in the model selection part of the simulation study. When we also take the second best solution into account as recommended by Ceulemans and

Kiers [25, 26], we obtain 81% correct selection. Given that some conditions are really difficult, this is a good result. The number of observations per data block, the number of clusters and the error level have important effects. Specifically, from Figure 1, we conclude that in case of two clusters the correct model is always the best or second best CHULL solution, whereas results deteriorate when the number of clusters increases. This effect of the number of clusters is reinforced by the number of observations per data block: having more information per block markedly improves model selection. Finally, model selection is worst for the data sets with 60% error variance. Thus, for data sets with low VAF% it is better to rely on substantive considerations and interpretability when selecting the most appropriate number of clusters.

[Insert Figure 1 about here]

6. Application

In this section, we present an empirical example from emotion research. Specifically, the data were gathered to study negative emotional granularity, which refers to the degree of differentiation between negative emotions in a subject's emotional experience [31]. Subjects who score low on negative emotional granularity are unable to differentiate between different negative emotions and thus feel overall negative without further nuance (i.e., all negative emotions co-occur), whereas subjects scoring high on emotional granularity describe their emotions in a more fine-grained way and will report specific negative emotions without the co-occurrence of all other negative emotions.

In the study, 42 subjects were asked to rate on a 7-point scale the extent to which 22 target persons (e.g., mother, father, partner, …) elicited 15 negative emotions, where the selected target persons obviously differ across subjects. Rows with missing data values were

removed, which led to the complete removal of one subject. Thus, the data being analyzed consists of 41 data blocks $\mathbf{X}_i$, one for each subject, where each data block holds the ratings of the 15 negative emotions for up to 22 target persons selected by subject *i*. The data blocks are columnwise centered, vertically concatenated and columnwise rescaled over all data blocks to achieve a total variance equal to one for each emotion.

As is mostly the case in empirical research, we have no idea on the number of clusters to use. Therefore, we perform model selection by, first, performing OC-SCA-IND analyses with one up to eight clusters and, then, performing the CHULL procedure. Visual inspection of the scree plot in Figure 2 leads us to conclude that two or four clusters seem to be an appropriate number of clusters; this conclusion is corroborated by CHULL which retains these two solutions as the best ones. We therefore inspected both solutions and they extracted essentially the same information from the data. The four-cluster solution was more refined than the two-cluster one but also harder to interpret; thus, for reasons of parsimony, we will only discuss the two-cluster solution.

[Insert Figure 2 about here]

The loadings of the two-cluster OC-SCA-IND model are given in Table 1. The component of the first cluster has high loadings of all negative affect items – only the loading of ‘jealous’ is somewhat lower – which is why we labeled it ‘negative affect’. The component of the second cluster has a strongly negative loading of ‘jealous’ as well as positive high loadings of ‘bored’, ‘uneasy’, ‘angry’, ‘dislike’, ‘uncomfortable’, ‘disgust’ and ‘hatred’; thus, we labeled it ‘dislike versus jealousy’. Which subjects are assigned to which clusters may be read from the left portion of Table 2. From this table, it appears that 30 out of the 41 subjects are assigned to both clusters, whereas nine are only assigned to Cluster 1 and four only to Cluster 2. Thus, for the majority of the subjects both components are, at least to some extent,

underlying their emotional rating of the target persons. How strongly each of the components is underlying their data, i.e., the component variances for each subject, may be found in the right part of Table 2.

[Insert Tables 1 and 2 about here]

With respect to emotional granularity, emotion ratings that are only affected by the 'negative affect' component are clearly not granular at all, because they will be more or less overall negative. In contrast, the 'dislike versus jealousy' component differentiates between two groups of negative emotions (jealousy on the one hand and a number of dislike-related emotions on the other hand), whilst not being associated to some other emotions (i.e., loadings of almost zero for 'sad', 'fearful', and 'nervous'). Rating target persons based on this component thus seems to add some granularity to one's emotional experience.

To evaluate whether the structural differences between the subjects, as expressed by the assignments to Cluster 1 and/or Cluster 2, may indeed be interpreted as differences in emotional granularity, we related the cluster memberships to the average intraclass correlation coefficients (ICCs; [32, 33]) measuring absolute agreement, which were calculated across the negative emotions for each subject. This subject-specific measure quantifies whether the target persons elicit each negative emotion to exactly the same extent (i.e., absolute agreement). To this end, the subjects were divided into three subgroups: (1) the subjects only assigned to Cluster 1 (i.e., applying only the 'negative affect' component in their ratings), (2) the subjects only assigned to Cluster 2 (i.e., applying only the 'dislike vs. jealousy' component), and (3) the subjects assigned to both clusters (i.e., applying both the 'negative affect' and 'dislike vs. jealousy' component). Boxplots of the ICCs for the three subgroups are given in Figure 3. From this figure, it is obvious that the ICCs are higher for subgroup 2. Specifically, the mean ICCs are .88 (*SD* = 0.010), .69 (*SD* = 0.11) and .89 (*SD* = 0.05) for

subgroups 1 to 3, respectively. As higher ICC values indicate a lower granularity, subgroup 2 – i.e., the subjects applying only the 'dislike vs. jealousy' component in their emotional ratings – contains the most granular subjects, which corresponds to what we hypothesized earlier.

[Insert Figure 3 about here]

In Table 3, the component loadings and block-specific component variances are given for the SCA-IND model with two components for the emotional granularity data. The component loadings are essentially identical to the OC-SCA-IND ones in Table 1, i.e., the congruence coefficients between the SCA-IND and OC-SCA-IND loadings are equal to .9988 and .9979 for the two components, respectively. The structure of the block-specific variances is very unclear, however, in that the variances that are zero according to the OC-SCA-IND model are estimated with values as high as 0.68 in the SCA-IND model.

[Insert Table 3 about here]

7. Discussion

In this paper, OC-SCA-IND was proposed as an adaptation of the existing C-SCA models. The key feature of the new method is the overlapping clustering, whereas each cluster corresponds to a single component. Consequently, on the one hand, OC-SCA-IND provides a lot more flexibility in modeling the differences and similarities in the underlying components of the different data blocks. On the other hand, it comprises the existing C-SCA methods (and SCA) as special cases. Additionally, it may be conceived as a penalized version of SCA-IND, in that the penalty in the objective function forces some block-specific component variances to become zero, implying that these components are not underlying the data block in question.

This leads to a more parsimonious and insightful solution. Specifically, in OC-SCA-IND, the 'status' (i.e., common versus some degree of cluster-specificity) of the different components becomes clear when looking at the clustering matrix, while in SCA-IND one has to inspect the block-specific component variances – which will easily take on values larger than zero for all components, due to the fitting of error variance, as we illustrated in Sections 5 and 6. Consequently, in SCA-IND, the component estimates can sometimes be inferior to the ones obtained by OC-SCA-IND.

In Sections 2 and 3, we motivated the choice to only elaborate OC-SCA-IND for the current paper. Using other SCA variants may be interesting for some data sets, however. On the one hand, when between-block differences in component variances are not interesting or desirable, the more restrictive OC-SCA-ECP may be preferred. On the other hand, the less restrictive OC-SCA-P may be used when between-block differences in component correlations are of interest (in addition to differences in component variances). Note that, in both cases, rotational freedom is present for components that correspond to identical columns in **U** with no overlap to other columns. The performance of these variants will be evaluated in future research.

Another point of discussion may be the assumptions implied by the OC-SCA-IND objective function. Specifically, the residuals are assumed to be independently, identically and normally distributed. For empirical data, this assumption will often not hold. The robustness of the OC-SCA-IND model against violations of this assumption was not examined in the current paper. Previous work by Wilderjans et al. [21] on the influence of between-block differences in error variance on the performance of a stochastically extended SCA, indicated that the performance is only hampered when large differences in error variance are combined with large differences in the size of the data blocks; thus, we expect similar results for OC-SCA-IND. In any case, in future research, it would be useful to thoroughly examine the

robustness of OC-SCA-IND to between-block and between-variable differences in residual variance, non-normality of the residuals or dependences between the residuals. If proven to be non-robust, extensions or adaptations of OC-SCA-IND could be developed, pertaining to further refinements of the OC-SCA-IND objective function or a robust counterpart of OC-SCA-IND building on the work of Hubert and colleagues [34, 35]. Another possibility could be to avoid the assumptions all together by using a least squares loss function with a penalty like the group lasso [36, 37]. Yet, a disadvantage would be that the weight of the penalty has to be tuned.

References

[1] K. De Roover, E. Ceulemans, M.E. Timmerman, How to perform multiblock component analysis in practice, Behavior Research Methods 44 (2012) 41−56.

[2] M.P. Lawton, M.H. Kleban, D. Rajagopal, J. Dean, Dimensions of affective experience in three age groups, Psychology and Aging 7 (1992) 171–184.

[3] P. Kuppens, E. Ceulemans, M. E. Timmerman, E. Diener, C. Kim-Prieto, Universal intracultural and intercultural dimensions of the recalled frequency of emotional experience, Journal of Cross-Cultural Psychology 37 (2006) 491–515.

[4] M.C. Marcucci, F. Ferreres, A.R. Custódio, M.M.C. Ferreira, V.S. Bankova, C. García-Viguera, W.A. Bretz, Evaluation of phenolic compounds in Brazilian propolis from different geographic regions, Zeitschrift für Naturforsch 55C (2000) 76–81.

[5] B. Gutendorf, J. Westendorf, Comparison of an array of in vitro assays for the assessment of the estrogenic potential of natural and synthetic estrogens, phytoestrogens and xenoestrogens, Toxicology 166 (2001) 79–89.

[6] M. J. Ramos, C. M. Fernández, A. Casas, L. Rodríguez, Á. Pérez, Influence of fatty acid composition of raw materials on biodiesel properties, Bioresource Technology 100 (2009) 261–268.

[7] A. Sousa-Poza, A.A. Sousa-Poza, Well-being at work: A cross-national analysis of the levels and determinants of job satisfaction, Journal of Socio-Economics 29 (2000) 517–538.

[8] L.T. Wright, C. Nancorrow, P.M.H. Kwok, Food taste preferences and cultural influences on consumption, British Food Journal 103 (2001) 348–357.

[9] Lemmens K, Dhollander T, De Bie T, Monsieurs P, Engelen K, et al., Inferring transcriptional modules from ChIP-chip, motif and microarray data, Genome Biology 7 (2006) R37.

[10] K. De Roover, E. Ceulemans, M.E. Timmerman, K. Vansteelandt, J. Stouten, P. Onghena, Clusterwise simultaneous component analysis for analyzing structural differences in multivariate multiblock data, Psychological Methods 17 (2012) 100−119.

[11] H.A.L. Kiers, J.M.F. ten Berge, Hierarchical relations between methods for simultaneous components analysis and a technique for rotation to a simple simultaneous structure, British Journal of Mathematical and Statistical Psychology 47 (1994) 109–126.

[12] M.E. Timmerman, H.A.L. Kiers, Four simultaneous component models of multivariate time series from more than one subject to model intraindividual and interindividual differences, Psychometrika 86 (2003) 105–122.

[13] I.T. Jolliffe, Principal component analysis, New York: Springer, 1986.

[14] K. Pearson, On lines and planes of closest fit to systems of points in space, Philosophical Magazine 2 (1901) 559–572.

[15] K. De Roover, M.E. Timmerman, I. Van Mechelen, E. Ceulemans, On the added value of multiset methods for three-way data analysis, Chemometrics and Intelligent Laboratory Systems 129 (2013) 98–107.

[16] K. De Roover, E. Ceulemans, M.E. Timmerman, P. Onghena, A clusterwise simultaneous component method for capturing within-cluster differences in component variances and correlations, British Journal of Mathematical and Statistical Psychology 86 (2013) 81−102.

[17] K. De Roover, E. Ceulemans, M.E. Timmerman, J.B. Nezlek, P. Onghena, Modeling differences in the dimensionality of multiblock data by means of clusterwise simultaneous component analysis, Psychometrika 78 (2013) 648−668.

[18] K. De Roover, M.E. Timmerman, B. Mesquita, E. Ceulemans, Common and Cluster-Specific Simultaneous Component Analysis, Plos One, 8 (2013c), e62280, doi:10.1371/journal.pone.

[19] R. Bro, A.K. Smilde, Centering and scaling in component analysis, Psychometrika 17 (2003) 16–33.

[20] H. Akaike, A new look at the statistical model identification, IEEE Transactions on Automatic Control 19 (1974) 716–723.

[21] T.F. Wilderjans, E. Ceulemans, I. Van Mechelen, R.A. van den Berg, Simultaneous analysis of coupled data matrices subject to different amounts of noise, British Journal of Mathematical and Statistical Psychology 64 (2011) 277−290.

[22] R. Bro, Multi-way analysis in the food industry: models, algorithms, and applications, PhD thesis (1998).

[23] I. Leenen, I. Van Mechelen, An evaluation of two algorithms for hierarchical classes analysis, Journal of Classification, 18 (2001), 57–80.

[24] R.B. Cattell, The scree test for the number of factors, Multivariate Behavioral Research 1 (1966) 245–276.

[25] E. Ceulemans, H. A. L. Kiers, Selecting among three-mode principal component models of different types and complexities: A numerical convex hull based method, British Journal of Mathematical and Statistical Psychology 59 (2006) 133−150.

[26] E. Ceulemans, H. A. L. Kiers, Discriminating between strong and weak structures in three-mode principal component analysis, British Journal of Mathematical & Statistical Psychology 62 (2009) 601−620.

[27] E. Ceulemans, M.E. Timmerman, H.A.L. Kiers, The CHULL procedure for selecting among multilevel component solutions, Chemometrics and Intelligent Laboratory Systems 106 (2011) 12−20.

[28] T.F. Wilderjans, E. Ceulemans, K. Meers, CHull: A generic convex-hull-based model selection method, Behavior Research Methods 45 (2013) 1–15.

[29] L.R. Tucker, A method for synthesis of factor analysis studies (Personnel Research section Rep. No. 984), Washington, DC: Department of the Army, 1951.

[30] U. Lorenzo-Seva, J.M.F. ten Berge, Tucker's congruence coefficient as a meaningful index of factor similarity, Methodology 2 (2006) 57–64.

[31] L.F. Barrett, Discrete emotions or dimensions? The role of valence focus and arousal focus, Cognition and Emotion 12 (1998) 579–599.

[32] P.E. Shrout, J.L. Fleiss, Intraclass correlations: Uses in assessing rater reliability, Psychological Bulletin 86 (1979) 420–428.

[33] M.M. Tugade, B.L. Fredrickson, L.F. Barrett, Psychological resilience and positive emotional granularity: Examining the benefits of positive emotions on coping and health, Journal of Personality 72 (2004) 1161–1190.

[34] E. Ceulemans, M. Hubert, P. Rousseeuw, Robust multilevel simultaneous component analysis, Chemometrics and Intelligent Laboratory Systems 129 (2013) 33−39.

[35] M. Hubert, P. Rousseeuw, K. Vanden Branden, ROBPCA: a new approach to robust principal components analysis, Technometrics 47 (2005) 64–79.

[36] K. Van Deun, T.F. Wilderjans, R.A. Van Den Berg, A. Antoniadis, I. Van Mechelen, A flexible framework for sparse simultaneous component based data integration, BMC bioinformatics 12 (2011) 448.

[37] M. Yuan, Y. Lin, Model selection and estimation in regression with grouped variables, Journal of the Royal Statistical Society: Series B 68 (2006) 49–67.

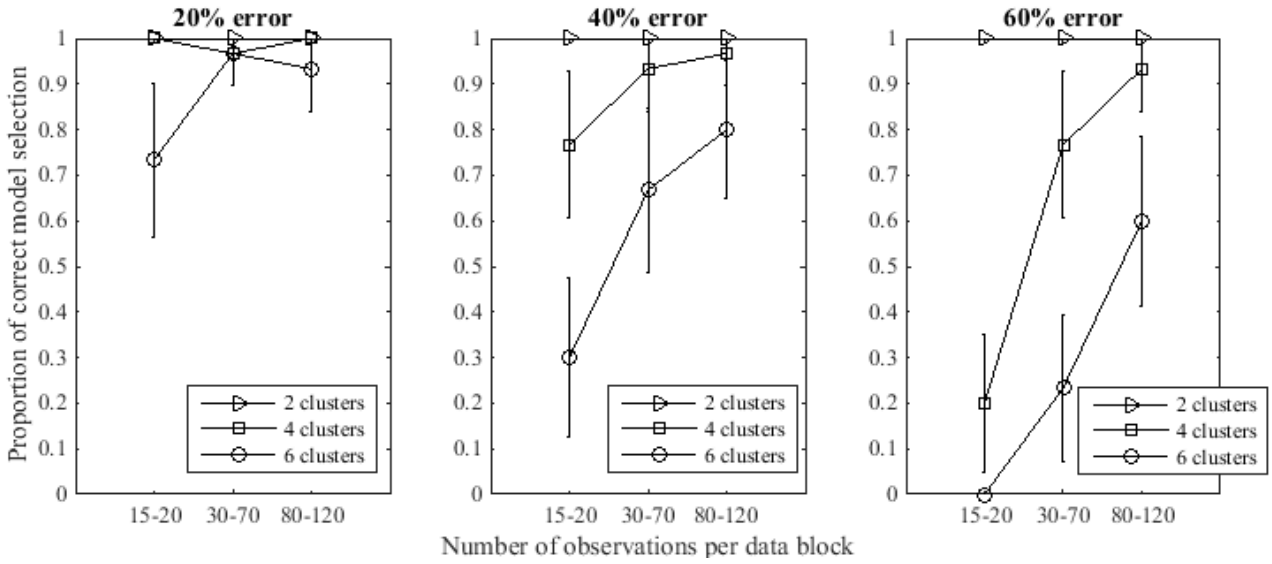

Figure 1. Mean values and associated 95% confidence intervals of the proportion of data sets with a correct model selection for OC-SCA-IND, i.e. the correct number of clusters is within the two best solutions according to the CHULL, as a function of the error level, the number of clusters, and the number of observations per data block.

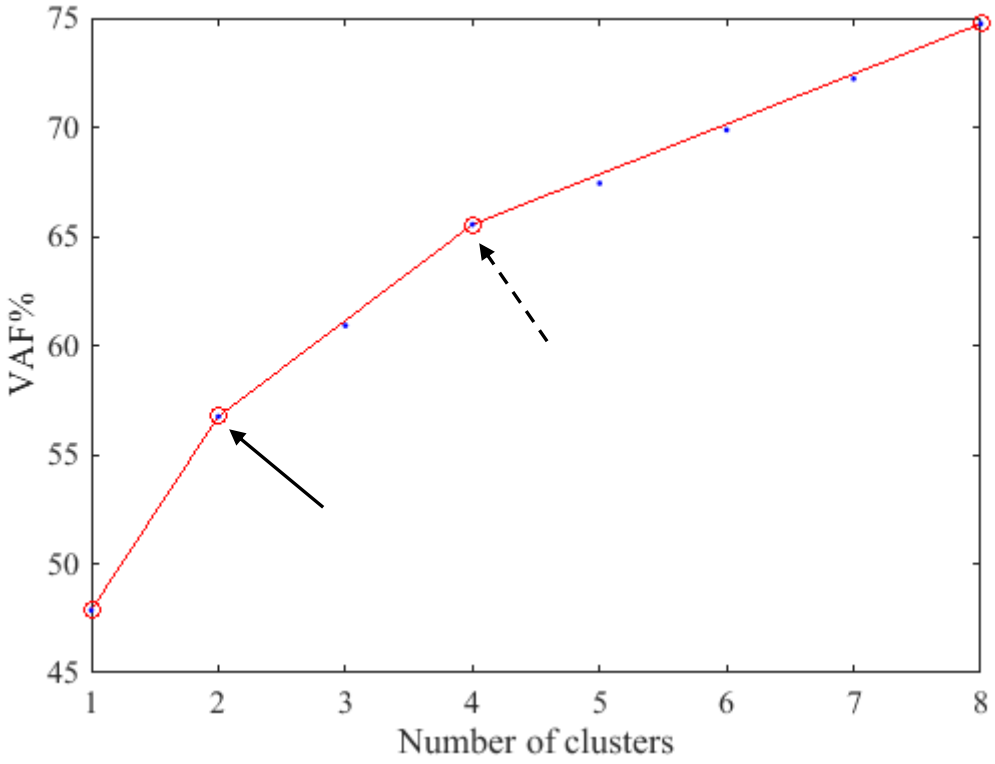

Figure 2. Scree plot with the percentage of variance accounted for (VAF%) for the OC-SCA-IND models with one up to eight clusters for the emotional granularity data. The convex hull according to the CHULL procedure is indicated by the red line and the solutions on the hull are indicated by a red circle. The solid arrow indicates the best solution according the CHULL procedure and the dashed arrow indicates the second best solution.

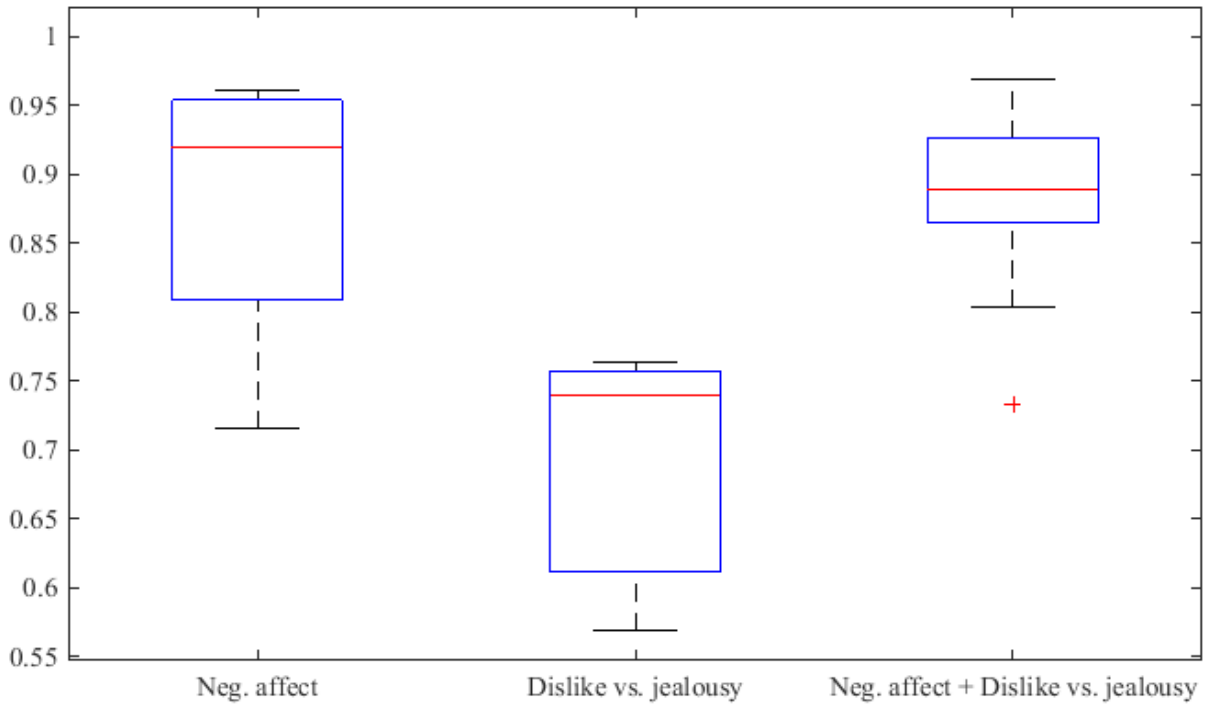

Figure 3. Boxplots of the intraclass correlation coefficients for (from left to right) the eight subjects only assigned to Cluster 1 (‘Neg. affect’) of the two-cluster OC-SCA-IND model, the three subjects only assigned to Cluster 2 (‘Dislike vs. jealousy’), and the 30 subjects assigned to both clusters (‘Neg. affect + dislike vs. jealousy’).

Table 1. Component loadings of the two-cluster OC-SCA-IND model for the emotional granularity data set. Loadings with an absolute value higher than .40 are printed in bold face.

	Component of Cluster 1 (38 subjects)	*Component of Cluster 2 (33 subj.)*
	Neg. affect	Dislike vs. jealousy
Bored	**.42**	**.71**
Uneasy	**.61**	**.43**
Miserable	**.68**	.12
Angry	**.67**	**.53**
Dislike	**.61**	**.68**
Inferior	**.61**	-.35
Sad	**.80**	-.01
Frustrated	**.67**	.37
Jealous	.36	**-.51**
Fearful	**.70**	.09
Nervous	**.76**	.00
Uncomfortable	**.70**	**.46**
Disgust	**.59**	**.75**
Upset	**.74**	.38
Hatred	**.59**	**.73**

Table 2. Clustering matrix (left) and subject-specific component variances (right) of the two-cluster OC-SCA-IND model for the emotional granularity data set.

	Cluster 1 (38 subjects)	*Cluster 2 (33 subjects)*	*Neg. affect (Cl. 1)*	*Dislike vs. jealousy (Cl. 2)*
Subject 1	1	0	1.35	0
Subject 2	1	1	1.22	1.29
Subject 3	1	1	1.33	0.67
Subject 4	1	0	0.72	0
Subject 5	1	1	1.09	1.12
Subject 6	1	1	0.98	0.93
Subject 7	1	1	0.51	1.31
Subject 8	1	1	0.93	0.92
Subject 9	1	1	1,21	0.76
Subject 10	1	1	0,70	1.29
Subject 11	1	1	0,96	0.64
Subject 12	1	0	0,94	0
Subject 13	1	1	1.22	1.11
Subject 14	1	0	0.91	0
Subject 15	0	1	0	0.89
Subject 16	1	1	0.93	1.26
Subject 17	1	1	0.96	0.70
Subject 18	1	1	0.81	0.71
Subject 19	0	1	0	1.40
Subject 20	1	0	1.42	0
Subject 21	1	1	1.07	0.83
Subject 22	1	1	0.66	0.67
Subject 23	1	1	0.76	1.48
Subject 24	0	1	0	0.93
Subject 25	1	1	1.22	0.94
Subject 26	1	0	0.73	0
Subject 27	1	1	0.78	0.71
Subject 28	1	1	0.53	1.20
Subject 29	1	1	0.76	0.83
Subject 30	1	1	1.21	0.80
Subject 31	1	1	1.44	0.86
Subject 32	1	1	0.60	0.80
Subject 33	1	1	0.56	0.84
Subject 34	1	1	1.18	1.29
Subject 35	1	1	1.25	1.01
Subject 36	1	1	0.44	1.01
Subject 37	1	0	0.35	0
Subject 38	1	0	1.39	0
Subject 39	1	1	0.48	1.04
Subject 40	1	1	0.77	1.00
Subject 41	1	1	0.80	0.72

Table 3. Component loadings (left) and subject-specific component variances (right) of the two-component SCA-IND model for the emotional granularity data set. Loadings with an absolute value higher than .40 and variances that are zero in the OC-SCA-IND model are printed in bold face.

	Component loadings			Subj.-spec. component variances	
	Neg. affect	*Dislike vs. jealousy*		*Neg. affect*	*Dislike vs. jealousy*
Bored	.34	**.70**	Subj. 1	1.36	**0.68**
Uneasy	**.54**	**.46**	Subj. 2	1.22	1.46
Miserable	**.64**	.17	Subj. 3	1.42	0.73
Angry	**.62**	**.50**	Subj. 4	0.75	**0.52**
Dislike	**.55**	**.65**	Subj. 5	1.15	1.17
Inferior	**.62**	-.29	Subj. 6	1.01	1.01
Sad	**.78**	.04	Subj. 7	0.53	1.36
Frustrated	**.63**	.37	Subj. 8	0.97	0.99
Jealous	**.40**	**-.47**	Subj. 9	1.27	0.86
Fearful	**.67**	.11	Subj. 10	0.70	1.37
Nervous	**.73**	.03	Subj. 11	1.03	0.68
Uncomfortable	**.65**	**.46**	Subj. 12	0.97	**0.56**
Disgust	**.53**	**.70**	Subj. 13	1.22	1.28
Upset	**.69**	.38	Subj. 14	0.93	**0.41**
Hatred	**.53**	**.69**	Subj. 15	**0.31**	0.91
			Subj. 16	0.89	1.44
			Subj. 17	1.02	0.74
			Subj. 18	0.83	0.81
			Subj. 19	**0.44**	1.41
			Subj. 20	1.48	**0.48**
			Subj. 21	1.11	0.90
			Subj. 22	0.69	0.71
			Subj. 23	0.76	1.59
			Subj. 24	**0.35**	0.93
			Subj. 25	1.29	1.01
			Subj. 26	0.73	**0.61**
			Subj. 27	0.83	0.75
			Subj. 28	0.54	1.26
			Subj. 29	0.80	0.87
			Subj. 30	1.31	0.84
			Subj. 31	1.51	0.95
			Subj. 32	0.63	0.84
			Subj. 33	0.54	0.94
			Subj. 34	1.12	1.54
			Subj. 35	1.28	1.15
			Subj. 36	1.49	1.15
			Subj. 37	0.30	**0.40**
			Subj. 38	1.45	**0.53**
			Subj. 39	0.51	1.07
			Subj. 40	0.79	1.08

	Subj. 41	0.85	0.76