Universiteit Leiden

Faculteit der Sociale Wetenschappen

Clusterwise Independent Component Analysis (C-ICA) for multi-subject fMRI data

A novel unsupervised method for assessing differences across subjects (groups) in functional connectivity patterns

Jeffrey Durieux

Master Thesis Psychology, Methodology and Statistics Unit, Institute of Psychology

Faculty of Social and Behavioral Sciences – Leiden University

Date: August 2015

Student number: 0921017

Supervisor: Dr. Tom F. Wilderjans

Second readers: Prof. dr. Mark de Rooij & Dr. Wouter D. Weeda

Acknowledgments

I would like to express my gratitude to my supervisor, Dr. Tom F. Wilderjans, for the continuous motivational support he has given to me during this project. This master thesis would not have been possible without his valuable feedback and guidance. I truly enjoyed the numerous meetings we had in office 3B18 and the insightful discussions at the ‘large table’ on the third floor of the faculty building. Secondly, I would also like to thank Prof. Dr. Mark de Rooij for giving me the opportunity to continue this project as a member of his team.

I would also like to thank my parents. Their unconditional support and genuine interest in my education, gave me an extra motivation to excel in both my Bachelor’s and Master’s degree. Most importantly, I would like to thank my girlfriend Nicole for her patience and love during this project. Words cannot express the importance of her support and encouragement.

Abstract

An important and emerging challenge in the field of neuropsychology pertains to revealing systematic differences (and similarities) between (groups of) patients in functional connectivity patterns. To this end, researchers often collect resting-state functional Magnetic Resonance Imaging (fMRI) data for multiple patients. One analysis strategy for this type of data consists of applying Independent Component Analysis (ICA) to the data of each patient separately; ICA is an analysis technique that decomposes a multivariate observed signal (from one subject) into a set of underlying independent source signals (i.e., spatial maps representing functional connectivity patterns) and their associated time courses. A major drawback of such a strategy is that each subject will be characterized by different connectivity patterns and time courses, which makes it very difficult to detect the systematic differences and similarities in connectivity patterns between (groups of) patients. Therefore, in this master thesis, an alternative, novel method, called *Clusterwise Independent Component Analysis* (CICA), is presented. The goal of this method is to cluster the patients into homogenous groups based on the similarities and differences in the functional connectivity patterns that characterize them. As such, patients allocated to the same cluster are assumed to have similar connectivity patterns, whereas patients belonging to different clusters will be described by qualitatively different connectivity patterns. To this end, the method combines an unsupervised clustering technique with ICA. In this thesis, after formulating the model expressions, an alternating least squares type of algorithm to estimate the C-ICA parameters is proposed, along with two procedures to tackle the non-trivial model selection problem (i.e., determining the optimal number of clusters and components). To evaluate the performance of the new CICA method, two extensive simulation study are conducted and the proposed model selection strategies are compared in a small third simulation study. Finally, directions for future research, including possible extensions of the CICA model, are presented. We hope this thesis to be a first, but decisive, step in the direction of the development of analysis techniques that allow for the detection of (potentially not yet known) disease subtypes (e.g., depression types) and/or subphases of neuropsychology disorders (e.g., dementia), which would imply a valuable advancement of neuropsychological research.

Table of Contents

Section 1. Introduction

In the behavioural sciences researchers often encounter three-way data, like, for example, resting-state multi-subject functional Magnetic Resonance Imaging (fMRI) data in which for a set of patients the BOLD response at various voxels is recorded over time. Such data, which are often collected in the context of neuropsychological studies on the neural basis of disorders (e.g., Alzheimer disease and Schizophrenia; see Dennis & Thompson, 2014; Shenton, Dickey, Frumin & McCarley, 2001), can be arranged in a three-dimensional array in which voxels represent the first dimension, time points the second and subjects the third one. As can be seen in Figure 1, multi-subject fMRI data can also be considered as multiple two-dimensional matrices (i.e., voxels by time points) where each matrix represents the fMRI data of a single individual.

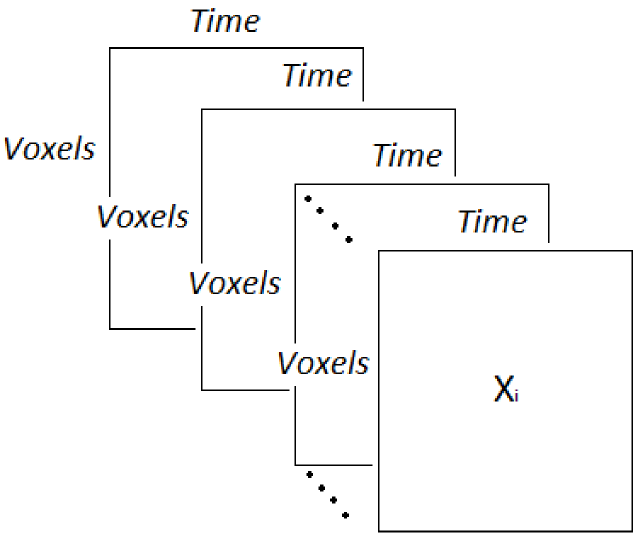

Figure 1. Graphical representation of multi-subject (resting-state) fMRI data consisting of I data blocks X_i $(i = 1 \dots I)$ with each data block $\boldsymbol{X}_i$ containing the BOLD response for subject i measured for different voxels over time.

An important and emerging challenge in the field of neuropsychology pertains to revealing differences in functional connectivity patterns (i.e. coordinated activity across brain regions) between (groups of) subjects. For example, identifying important (maybe even not yet known) changes in connectivity patterns for patients in consecutive stages of a neuropsychological disorder, like, for example Alzheimer’s disease, may substantially advance the scientific knowledge on the neural basis of such a disorder. An often used technique to disclose the most apparent connectivity patterns in resting-state fMRI data of a single subject is Independent Component Analysis (ICA; see Hyvärinen & Oja, 2000; Stone, 2004; Greicius, Srivastava, Reiss & Menon, 2004; Van de Ven, Formisano, Prvulovic, Roeder & Linden, 2004; Kiviniemi, Knatola, Jauhiainen, Hyvärinen & Tervonen, 2003). ICA is a relatively new decomposition technique that separates a multivariate signal (e.g., fMRI data) into statistically independent components. Moreover, ICA also discloses how the observed signal is obtained as a linear mixing of the underlying independent components (i.e., a linear mixing matrix). In the context of fMRI data, the independent components refer to spatial maps that can be interpreted as sets of voxels that are functionally connected (i.e., connectivity patterns), whereas the mixing matrix contains information regarding the underlying time course for each independent component. An advantage of using ICA (compared to using the linear model) for analysing fMRI data is that the underlying time courses should not be known in advance as they are determined during the analysis; this is especially interesting when working with resting-state fMRI data for which no expectations regarding the true time course of the BOLD signal can be postulated.

When analysing fMRI data of multiple subjects, one analysis strategy consists of performing ICA on each data set separately. A major drawback of such an approach is that the relationships (i.e., systematic differences and similarities) between different subjects are totally neglected. In particular, applying a separate ICA for each subject results in time courses and spatial maps that are specific for each subject, which makes it a difficult task to identify similarities and differences between these subjects in terms of functionally connected brain activity. In order to overcome this drawback, Beckmann and Smith (2005) proposed tensor Probabilistic ICA (tensor PICA). In tensor PICA, a multi-subject fMRI data set is decomposed as a trilinear product of three component matrices that represent (group) spatial maps, associated (group) time courses and subject specific weights (Guo & Pagnoni, 2008). As a consequence, tensor PICA results in a single set of spatial maps and prototypical time courses, which are shared among all subjects, and a set of subject specific weights which allow the predicted time courses to differ across subjects but only in a restrictive way (i.e., proportional time profiles that have the same shape as peaks are forced to occur at the same moment).

Although tensor PICA is clearly able to identify the similarities (i.e., group spatial maps) between the subjects under study, the method may obfuscate relevant differences across subjects. In particular, by assuming spatial maps to be the same for all subjects and time courses to be proportional to each other, crucial (qualitative) differences among the subjects may be overlooked. This may happen, for instance, when the population under study consists of groups of subjects that exhibit qualitative differences in brain functioning. In this regard, one may, for example, hypothesize that different stages of a disorder (e.g., Alzheimer) are characterized by substantial changes in functional connectivity patterns (i.e., spatial maps; see Gili et al., 2011). When this is true, it can be assumed that patients that are in a similar stage of a specific disorder have similar functional connectivity patterns, whereas patients that are in a different stage of this disorder may be characterized by connectivity patterns that are qualitatively different. In this case, a method that is able to identify qualitative differences in functional connectivity between groups of patients would provide a clear advantage over tensor PICA. In particular, such a method may account for the heterogeneity in functional connectivity patterns across (groups of) patients with a certain neurological disorder and, as such, may yield valuable insights into the development and prognosis of the pathology under study.

A promising way to uncover qualitative differences in functional connectivity between (groups of) patients consists of clustering (in an unsupervised way) the patients based on their underlying brain connectivity patterns. In particular, patients with similar connectivity patterns should be clustered together, whereas patients exhibiting connectivity patterns that are qualitatively different should be allocated to different clusters. Up to now, however, no such method exists that is able to disclose groups of patients that differ in functional connectivity patterns. The goal of this master thesis therefore is to develop a novel analysis method for multi-subject fMRI data that combines exploratory clustering techniques (i.e., unsupervised learning) with ICA in order to identify differences in connectivity patterns among (groups of) patients. In particular, the proposed method, which will be called Clusterwise Independent Component Analysis (C-ICA), will cluster the subjects into homogeneous groups based on the similarities and differences in their functional connectivity patterns. Note that by adopting an unsupervised method, additionally new insights may be obtained regarding the progressive phases of the disorder in question (e.g., differentiating phases in meaningful subphases or identifying phases that are characterized by changes in functional connectivity instead of or above changes in psychological and behavioural aspects of patients' functioning); these insights may complement or even contradict the existing consensus on the disease phases, which are often based on psychological and behavioural aspects of the functioning of the patients only.

The remainder of this thesis, which will be written in the format of an article, is organized as follows: in the next section, the basic principles of ICA estimation for single-subject fMRI data will be discussed and additionally the mathematical formulation of the new C-ICA model will be introduced. Next, in the *Data Analysi*s section, an appropriate algorithm to estimate the parameters of the C-ICA model will be presented, along with a short description of easy to use software for C-ICA and a procedure to select the optimal C-ICA model (i.e., number of clusters and components needed). In the fourth section, the performance of the C-ICA algorithm is evaluated by means of extensive simulation studies. Finally, implications of the C-ICA model and directions for future research will be discussed.

Section 2. Clusterwise Independent Component Analysis

In this section, an extensive formulation of the basic ICA model for single-subject fMRI data will be given. Here, several statistical principles and methods for ICA estimation will be explained. Finally, a short motivation for the novel C-ICA model will be presented, alongside the mathematical formulation of the C-ICA model.

2.1 The Independent Component Analysis framework for a single subjects' fMRI data

2.1.1 Linear representations of multivariate data

A commonly used analysis technique that is able to extract underlying patterns of functional connectivity from the resting-state fMRI data of a single subject is Independent Component Analysis (ICA; Hyvärinen & Oja, 2000; Stone, 2004). In ICA, a multivariate – observed – signal (e.g., the BOLD response for a set of voxels) is decomposed into a set of statistically independent – unobserved – source signals (e.g., correlated voxels which represent functionally connected brain regions) with their associated time courses. As such, ICA is able to separate systematic signal information (e.g., connectivity patterns, which usually appears in independent components) from noise and other – systematic but not relevant for the study – sources of variability (e.g. subtle head movements, cardiac pulsations) that usually compromise the BOLD signal.

Technically, ICA is a multivariate analysis technique that aims at finding a linear representation of non-Gaussian data in such a way that the statistical dependency between the underlying non-Gaussian components is minimized. In the basic ICA model (Jutten & Herault, 1991; Comon, 1992; Bell & Sejnowski, 1995; Hyvärinen & Oja, 2000; Stone, 2004), an (underlying) n-dimensional random vector of non-Gaussian independent source signals[1] $\boldsymbol{s} = (s_1, \dots, s_n)^{\mathrm{T}}$ is recovered from an n-dimensional random vector of observed signal mixtures $\boldsymbol{x} = (x_1, \dots, x_n)^{\mathrm{T}}$. The observed mixture signals in $\boldsymbol{x}$ are obtained by a linear mixing

[1] Throughout this thesis, the terms 'independent components' and 'source signals' will be used interchangeably.

by means of an $n \times n$ mixing matrix $\boldsymbol{A}$ (with elements a_{ij}) of the (independent) source signals in $\boldsymbol{s}$. Thus, the general ICA decomposition can be written as:

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{s} \tag{2.1}$$

It should be noticed that in the formulation of ICA (2.1) presented above, the mixing matrix $\boldsymbol{A}$ is considered to be square (i.e., n source components are derived from n mixture signals). In some applications, however, the goal may be to derive only q source signals from n observed mixture signals (with $q < n$), resulting in a non-square mixing matrix $\boldsymbol{A}$ (of size $n \times q$). In the non-square case, $\boldsymbol{x}$ cannot be perfectly decomposed into the product $\boldsymbol{A}\boldsymbol{s}$ and $\boldsymbol{s}$ cannot be computed through inverting the mixing matrix (see further). However, a decomposition into $\boldsymbol{A}\boldsymbol{s}$ (with $\boldsymbol{s}$ containing $q < n$ independent signals) can be sought that approximates $\boldsymbol{x}$ as close as possible (for example, in least-squares sense).

The unknown source signals $\boldsymbol{s}$ can be computed by multiplying the inverse of the mixing matrix $\boldsymbol{A}$ (i.e. unmixing matrix $\boldsymbol{W}$ with elements $\boldsymbol{w}_{ij}$) with the observed mixed signals $\boldsymbol{x}$:

$$\boldsymbol{s} = \boldsymbol{W}\boldsymbol{x} \tag{2.2}$$

Therefore, in order to find the underlying components $\boldsymbol{s}$, the unmixing matrix $\boldsymbol{W}$ has to be determined. To this end, several statistical principles could be used (for an overview of different principles, see Hyvärinen and Oja, 2000). For example, one could determine $\boldsymbol{W}$ such that the components are uncorrelated with each other. This procedure is known as principal component analysis (PCA). ICA, however, uses a more stringent statistical principle than the principle of uncorrelatedness that is adopted in PCA. In ICA, $\boldsymbol{W}$ is determined such that the underlying components are *statistically independent* from each other. Note that when the components are normally (i.e., Gaussian) distributed, uncorrelatedness implies independence (and vice versa); in the non-Gaussian case, however, independence implies uncorrelatedness but the reverse in general does not hold. Here, two important assumptions are necessary to make ICA estimation and the disclosure of the underlying components possible: (1) the underlying source signals are mutually statistically independent and (2) the underlying source signals are random variables from a distribution that does not resemble the Gaussian distribution.[2]

[2] For fMRI data, components pertaining to important systematic information in the signal are often non-Gaussian and can only be successfully separated from each other by imposing independence (i.e., uncorrelatedness does not suffice).

2.1.2 Principles of ICA estimation: independence and non-Gaussianity

Independence. The first principle that is used to estimate ICA is statistical independence. In particular, the source signals $\boldsymbol{s}$ are estimated such that they are as independent as possible. Intuitively, the source signals $\boldsymbol{s}$ are said to be independent when information on the value of s_i yields no information on the value of s_j for $i \neq j$ (Hyvärinen & Oja, 2000). More formally, statistical independence can be defined in terms of the joint and marginal probability density functions (pdf) of the s_i's. Statistical independence implies that the joint pdf of $\boldsymbol{s}$ equals the product of its marginal pdfs (i.e., the pdfs of the s_i's). In particular, consider two random (centered) variables s_1 and $s_{2,}$ let their joint pdf be denoted by $p_{s1s2}(s_1, s_2)$ and their marginal pdfs by $p_{s1}(s_1)$ and $p_{s2}(s_2)$, then statistical independence is defined as: $p_{s1s2}(s_1, s_2) = p_{s1}(s_1)\ p_{s2}(s_2)$.[3]

In practice, however, the exact pdf of a random variable is often unknown and lots of data are required to estimate such a pdf in an accurate way. As a way out, the expectation (of some function) of a given random variable, which can be easily estimated directly from the data, is derived and is used to check for statistical independence (Hyvärinen, Karhunen & Oja, 2001). For a given function g, the *expected value* of a function of the (continuous, centered) random variable s_1 is defined as:

$$E\{g(s_1)\} = \int_{-\infty}^{\infty} g(s_1) p_{s_1}(s_1) ds_1 \tag{2.3}$$

In the case of two random variables s_1 and s_2, given (arbitrary) functions $g_1$and g_2, the expected value of the joint density is given by:

$$E\{g_1(s_1)g_2(s_2)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g_1\,(s_1)g_2(s_2)\,p_{s1s2}(s_1, s_2)ds_1s_2 \tag{2.5}$$

Here the random variables s_1 and s_2 are said to be independent if and only if the expectation (i.e., first moment) of their joint pdf can be factorized into the (product of the) expectations of their marginal pdfs as follow:

$$\begin{aligned} E\{g_1(s_1)g_2(s_2)\} &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g_1\,(s_1)g_2(s_2)\,p_{s1s2}(s_1, s_2)ds_1s_2 \\ &= \int_{-\infty}^{\infty} g_1(s_1)p_{s_1}(s_1)ds_1 \int_{-\infty}^{\infty} g_2(s_2)p_{s2}(s_2)ds_2 \\ &= E\{g_1(s_1)\}E\{g_2(s_2)\} \end{aligned} \tag{2.6}$$

Additionally, if $s_1$and s_2 are independent then their covariance (i.e., joint variability) equals zero since $COV = E\{g_1(s_1)g_2(s_2)\} - E\{g_1(s_1)\}E\{g_2(s_2)\}$. Note that in case of statistical independence both properties hold for any possible functions $g_1$and g_2.

[3] In the case of n random variables, the joint pdf is a product of n terms, that is, $p_s(s_1, s_2, \dots, s_n) = \prod_{q=1}^{n} p_{s_q}(s_q)$.

As mentioned earlier, statistical independence (the property used in ICA) is more stringent than uncorrelatedness. Indeed, uncorrelatedness is obtained when equation 2.6 holds for all linear functions g_1 and g_2, whereas statistical independence requires that 2.6 holds for all possible functions/transformations g_1 and g_2 (Hyvärinen, Karhunen & Oja, 2001), with the class of linear functions being a subclass of the latter (much larger) class of functions.

Non-Gaussianity. A second principle that is used for ICA estimation is that of non-Gaussianity. For ICA estimation, it is not only necessary that the source signals $\boldsymbol{s}$ are assumed to be independent (see above) but the source signals $\boldsymbol{s}$ should also be non-Gaussian (i.e., follow a distribution that does not resemble the Normal distribution, like, for example, a Laplace or a uniform distribution). When components are Gaussian, by the Central Limit Theorem, any linear mixture of them is also Gaussian. As such, the underlying components cannot be identified in a unique way from the observed mixtures without extra knowledge regarding these underlying sources. Note that ensuring the components to be independent does not help as in the Gaussian case uncorrelatedness implies independence (and, as a consequence, ensuring independence does not imply a more stringent assumption). In fact, a multivariate distribution of independent (i.e., uncorrelated) Gaussian variables results in a density that is rotationally symmetric (Hyvärinen, Karhunen & Oja, 2001). A density that is rotationally symmetric contains no information about the directions of the columns of the mixing matrix $\boldsymbol{A}$, or likewise the unmixing matrix $\boldsymbol{W}$ (see equation 2.1 and 2.2 respectively), resulting in the elements in matrix $\boldsymbol{W}$ not being identifiable when independent components are Gaussian. Indeed, applying ICA estimation while the underlying components are in fact Gaussian will only result in a whitening of the data (i.e. yields uncorrelated components with unit variances) but does not yield a disclosure of the underlying independent components.

When the underlying independent components are in fact non-Gaussian, then ICA estimation is possible. Given, as following from the Central Limit Theorem, that the probability density function of a linear mixture of variables is approximately more Gaussian than the pdf of its constituent (source) variables, non-Gaussian independent components $\boldsymbol{s}$ can be estimated from their (more Gaussian linear) mixtures $\boldsymbol{x}$ by means of maximizing a measure of non-Gaussianity of the source signals $\boldsymbol{s}$. In this procedure, a linear combination $s_j = \sum_i w_{ij} x_i$ is sought such that the s_j's are as non-Gaussian as possible. To this end, the non-Gaussianity of s_j is quantified and this measure is optimized (see further in Section 2.1.3). It can be demonstrated that the local maxima of this non-Gaussianity measure yield the independent components $\boldsymbol{s}$ (Hyvärinen, Karhunen & Oja, 2001).

2.1.3 Maximizing non-Gaussianity to ensure independence and the fastICA algorithm

In order to retrieve the independent non-Gaussian signals $\boldsymbol{s}$ that underlie a set of observed mixture signals $\boldsymbol{x}$, both a measure of the non-Gaussianity of the source signals $\boldsymbol{s}$ and an optimization algorithm for maximizing this measure for non-Gaussianity are necessary. A classical measure of non-Gaussianity is kurtosis, which can be defined as the fourth moment $E[s^4]$ with s having unit variance and being zero-mean centered. Kurtosis is equal to zero for Gaussian random variables, negative for sub-Gaussian (platykurtic) and positive for super-Gaussian (leptokurtic) random variables. A drawback of using kurtosis as an objective function for ICA estimation is that it is very sensitive to outliers (Huber, 1985). Therefore, several other methods have been proposed to estimate ICA, with these methods being based on various approaches. For example, the infomax principle (Bell & Sejnowski, 1995) that is based on a maximum likelihood formulation of ICA can be used to estimate the ICA model. Another approach for estimating ICA consists of computing higher-order cumulant tensors (i.e., a generalization of calculating a covariance matrix) and finding the underlying components through a kind of eigenvalue decomposition. A popular algorithm in this regard is the JADE algorithm (Cardoso & Souloumiac, 1993). A third commonly used approach for identifying independent components, which will be used throughout this thesis, is maximizing negentropy. This quantity is related to differential entropy, a concept derived from information theory. A variable having a larger amount of "randomness" is said to have a larger entropy. From all (distributions of) random variables that have the same variance, Gaussian variables have the largest entropy. Negentropy, which is a normalized version of entropy, is always positive and equals zero if and only if a random variable is Gaussian. Further, negentropy is invariant to linear transformations (Comon, 1994; Hyvärinen, 1999; Hyvärinen & Oja, 2000).

As noted in Hyvärinen and Oja (2000), estimation of negentropy is a very difficult and noise-prone task as an (parametric or non-parametric) estimate of the full pdf is necessary. Therefore, Hyvärinen (1998, 2000) proposed to maximize an approximation of negentropy which is less computational intensive:

$$J(s) \approx k_1(E\{G_1(s)\})^2 + \ k_2(E\{G_2(s)\}^2 - E\{G_2(v)\})^2 \tag{2.7}$$

here, v is a standardized Gaussian variable, s is the independent component that is sought for, k_1 and k_2 are positive integers and G_1 and G_2 are non-quadratic contrast function that are defined as follow:

$$G_1(s) = \frac{1}{a_1} \log\cosh a_1 s, \text{ where } 1 \leq a_1 \leq 2 \tag{2.8}$$

and,

$$G_2(s) = -\exp(-s^2/2) \tag{2.9}$$

In cases where only one non-quadratic function $G(s)$ is used, the approximation from Equation 2.7 becomes:

$$J(s) \propto [E\{G(s)\} - E\{G(v)\}]^2 \tag{2.10}$$

where $G(s)$ can be substituted by either Equation 2.8 or by Equation 2.9.

Equation 2.8 is a general purpose contrast function, meaning that it is suitable for both sub-Gaussian and super-Gaussian components. Equation 2.9, at the contrary, is more suitable when components are known to be highly super-Gaussian and/or when robustness is important (Hyvärinen, 1999). Note that when it can be expected that the data contain no outliers, kurtosis can be used as a third contrast function:

$$G_3(s) = \frac{1}{4} s^4 \tag{2.11}$$

As the components are independent of each other, the optimization of the approximation of negentropy as presented in (2.7) and (2.10) can be performed by estimating all independent components simultaneously or by using a deflation procedure in which each of the independent components in $\boldsymbol{s} = \boldsymbol{Wx}$ is computed separately/sequentially (i.e., the optimization function has different maxima, with each maximum pertaining to a separate independent component; see Hyvärinen, Karhunen & Oja, 2001; Stone, 2004). In particular, for each component separately, a weight vector $\boldsymbol{w}$ is sought, by means of rotating $\boldsymbol{w}$, such that the orientation of $\boldsymbol{w}$ gives a maximum value of the negentropy approximation for the extracted component. To find the optimal rotation of $\boldsymbol{w}$, different approaches have been proposed. A first approach consists of using a brute force search (Stone, 2004), but this will become easily computationally intensive when more than two components need to be extracted. A solution here is to use a well-known optimization algorithm, like, for example, a gradient descent type of algorithm (Amari, Cichocki & Yang, 1996). Alternatively, Hyvärinen (1999) introduced *fastICA*, a fast and robust fixed-point algorithm that yields advantageous properties compared to gradient descent methods. In particular, *fastICA*, which will be used throughout this thesis, has a convergence rate (i.e., how fast – in terms of the number of iterations needed – the algorithm reaches the optimum as a function of the complexity of the problem) that is cubic or at least quadratic, which makes it faster than gradient descent methods for which convergence is only linear. Additionally, no step parameter needs to be chosen prior to the analysis, which makes the algorithm easier to use than gradient descent methods for which the performance may heavily depend on the choice of this step parameter. Note that a deflational procedure for ICA estimation closely resembles another multivariate method, called project pursuit, where projections of multivariate data are found that display interesting (i.e. non-Gaussian) information/distributions (Friedman, 1987; Huber, 1985) in the data. Extracting components one by one can be used for exploratory data analysis. Moreover, it also decreases computational load (Hyvärinen & Oja, 2000), which can be very advantageous when, for example, analyzing computationally challenging neuroimaging data sets.

In sum, the problem of finding a suitable linear representation of multivariate non-Gaussian data can be solved by using the statistical principles of statistical independence and non-Gaussianity. Additionally, after determining a measure that quantifies non-Gaussianity, an algorithm can be constructed that optimizes this measure for each component separately. In this thesis, non-Gaussianity will be measured by (an approximation of) negentropy (2.7) and *fastICA* will be used to optimize negentropy in a computationally efficient way.

2.2 Clusterwise Independent Component Analysis

In this section, the novel C-ICA model will be presented to uncover qualitative differences in functional connectivity patterns between groups of patients. In this model, the patients are clustered into homogenous groups based on the similarities and differences in the functional connectivity patterns underlying the data of each patient. To determine the connectivity patterns that are common for each patient group, ICA is performed on the concatenated data per group. A similar clustering strategy for identifying heterogeneity in the underlying association structure of multivariate data has already been successfully adopted in the context of simultaneous component analysis (De Roover, et al., 2012) and for Parafac (Wilderjans & Ceulemans, 2013).

2.2.1 Mathematical model formulation of C-ICA

In the C-ICA model it is assumed that I data blocks $\boldsymbol{X}_i$ $(i = 1, \dots, I)$ fall apart into R mutually exclusive clusters, with each data block $\boldsymbol{X}_i$ containing the fMRI data (J voxels by K time points) for subject i. In the C-ICA model, the time courses are allowed to differ for each data block (i.e., patient) but the spatial maps are set equal for all data blocks that belong to the same cluster. Thus, C-ICA is defined as:

$$\boldsymbol{X}_i = \sum_{r=1}^{R} \boldsymbol{p}_{ir}\, \boldsymbol{s}^r \boldsymbol{A}_i + \boldsymbol{E}_i \qquad (2.12)$$

where the elements $\boldsymbol{p}_{ir}$ denote the entries from the binary partition matrix $\boldsymbol{P}$ $(I \times R)$ which equal 1 when data block/person i is assigned to cluster r and 0 otherwise. Similar as in equation (2.1), $\boldsymbol{A}_i$ $(Q \times K)$ denotes the mixing matrix for subject i and $\boldsymbol{s}^r$ $(J \ \times Q)$ the (independent) source signals for cluster r $(r = 1, \ldots, R)$. Note that the signals in $\boldsymbol{s}^r$ are assumed to be the same for each data block i that belongs to cluster r. Additionally, the model contains an error term $\boldsymbol{E}_i$ for each data block i.

2.2.2 Ambiguities of the (C-)ICA model

The C-ICA model suffers from four sources of non-uniqueness/ambiguity (see Hyvärinen & Oja, 2000; Hyvärinen, Karhunen & Oja, 2001), with the first three also holding for the ICA model and the fourth one being specific for C-ICA. First, scaling ambiguity, which implies that scaling components in $\boldsymbol{s}^r$ can be compensated in $\boldsymbol{A}_i$ by counterscaling, resulting in the product $\boldsymbol{s}^r\boldsymbol{A}_i$ being unchanged. This is because in ICA both the mixing matrix and the independent components have to be estimated and their product shows up in the ICA (and C-ICA) model formulation (2.1). Here, any scalar multiplier applied to one of the sources $\boldsymbol{s}_q$ can be cancelled in $\boldsymbol{s}^r\boldsymbol{A}_i$ by dividing the corresponding row $\boldsymbol{a}_q$ of $\boldsymbol{A}$ by that scalar. As noted by Hyvärinen, Karhunen & Oja (2001), this non-uniqueness can be accounted for during ICA estimation by assuming that the independent components $\boldsymbol{s}$ all have unit variance (i.e., $E\{s_q^2\} = 1$). A second ambiguity is reflectional ambiguity, which pertains to the possibility to change the sign of an estimated independent component. Indeed, multiplying one of the estimated components (in $\boldsymbol{s}^r$) by -1 does not affect the ICA model (2.1) as long as this is compensated for in the associated $\boldsymbol{A}_i$'s (i.e., multiplying the associated time course with -1). Note that reflectional ambiguity is a special case of scaling ambiguity (i.e., scaling with a factor of -1). Fixing the variance of the independent components, however, does not solve for reflectional ambiguity. Third, since both the components $\boldsymbol{s}$ and mixing matrix $\boldsymbol{A}$ are unknown, the order of the components in the ICA model can be freely interchanged. To see why this is possible, consider the basic ICA model from (2.1) written in another form: $\boldsymbol{x} = \sum_{q=1}^{Q} \boldsymbol{a}_q \boldsymbol{s}_q$.

From this equation it should be clear that any permutation of the terms in the summation would not affect the model. A fourth ambiguity, which solely applies to the C-ICA model, is that the cluster indices of the $\boldsymbol{s}^r$'s can be permuted freely. Thus not only the independent components $\boldsymbol{s}$ can be permuted into $Q!$ ways (see third ambiguity) but also the clustered signals $\boldsymbol{s}^r$ can be permuted into $R!$ different ways.

Section 3. Data Analysis

3.1 Aim of C-ICA

Given a pre-specified number of clusters R (and, in the non-square mixing matrix case – see earlier – an a priori determined number of components Q), the aim of C-ICA is to estimate a partitioning matrix $\boldsymbol{P}$, mixing matrices $\boldsymbol{A}_i (i = 1, \dots, I)$ and source signals $\boldsymbol{s}^r$ $(r = 1, \dots, R)$ such that the C-ICA loss function is minimized:

$$L = \sum_{i=1}^{I} \|\boldsymbol{X}_i - \sum_{r=1}^{R} \boldsymbol{p}_{ir}\, \boldsymbol{s}^r \boldsymbol{A}_i\|^2 \tag{3.1}$$

Based on the value of the loss function L, for a particular C-ICA model (with estimates for $\boldsymbol{P}$, $\boldsymbol{A}_i$ and $\boldsymbol{s}^r$), a percentage of variance accounted for (VAF) can be computed as follows:

$$VAF = \frac{\|\underline{\boldsymbol{X}}\|^2 - L}{\|\underline{\boldsymbol{X}}\|^2} \times\ 100 \tag{3.2}$$

where $\|\underline{\boldsymbol{X}}\|^2$ indicates the sum of all squared elements (across all data blocks).

Before analysing a data set with C-ICA, it is advised to pre-process the data by row-wise centring each data block $\boldsymbol{X}_i$. As a consequence, for each subject i, the data for each voxel j have a mean of zero (across the K time points). Note that this is in accordance with ICA being defined for centred signals $\boldsymbol{x}$ (see Section 2.1.1).

3.2 The algorithm for C-ICA and software

In order to achieve an optimal clustering with the C-ICA model (and to determine the associated subject specific mixing matrices and group specific source signals), an alternating least-squares type of algorithm is constructed. Here, the algorithm alternates between updating partitioning matrix $\boldsymbol{P}$ (i.e. the cluster memberships) and re-estimating the (cluster specific) ICA parameters $\boldsymbol{A}_i$ and $\boldsymbol{s}^r$ until convergence is reached. More specifically, the C-ICA algorithm consists of the following four steps:

1. *Randomly initialize partition matrix $\boldsymbol{P}$.* Here each data block i is allocated to one of the clusters r, with all blocks having the same probability of being assigned to each cluster. Note that this procedure is repeated until none of the clusters is empty. After the initialization of $\boldsymbol{P}$, the cluster specific ICA parameters are estimated as explained in step 3 and the loss function (3.1) is evaluated.
2. *Update partition matrix $\boldsymbol{P}$ data block by data block.* Here the optimal cluster membership for data block i is determined by evaluating for each cluster r the fit of the data block i under consideration by means of the partition criterion $L_{ir} = \left\|\boldsymbol{X}_i - \widehat{\boldsymbol{X}_i}\right\|^2$; each person i is assigned to the cluster r for which L_{ir} is minimal. More specifically, here for each data block i in cluster r an estimated $\widehat{\boldsymbol{X}_i}$ is computed via the formula $\widehat{\boldsymbol{X}_i} = \boldsymbol{s}^r \boldsymbol{A}_i$, where $\boldsymbol{s}^r$ is given by the previous *fastICA* estimation under step 3 and $\boldsymbol{A_i}$ is computed via $\boldsymbol{A}_i = \boldsymbol{X}_i^T \boldsymbol{s}^r (\boldsymbol{s}^{rT} \boldsymbol{s}^r)^{-1}$ and $(\ldots)^{-1}$ denotes matrix inversion and $\boldsymbol{s}^T$ the transpose of a matrix/vector. Note that after reassigning all data blocks to their optimal cluster, it could occur that some clusters are empty. In order to avoid empty clusters, a procedure is applied that puts the data block with the worst fit (after reassigning all data blocks and updating the cluster specific ICA parameters) into the empty cluster.
3. *Estimate the ICA parameters for each cluster (and evaluate the loss function).* First, all data blocks that belong to cluster r are horizontally concatenated together into $\boldsymbol{X}^r$. Next, *fastICA* is performed on each of the concatenated data blocks $\boldsymbol{X}^r$ in order to estimate the *C-ICA* parameters $\boldsymbol{A}_i$ and $\boldsymbol{s}^r$. Here, *fastICA* uses the contrast function from equation (2.8) with an alpha value of 1 as this contrast function is suitable for both sub-Gaussian and super-Gaussian components (Hyvärinen and Oja, 2000). After computing the cluster specific ICA parameters, the loss function is evaluated.
4. *Convergence criterion.* Steps 2 and 3 are repeated until the decrease in the loss function value (3.1) between two evaluations is smaller than the convergence criterion of .000001.

Because the C-ICA algorithm, as is true for almost all clustering algorithms (Brusco, 2006), may end in a local optimal solution, a multi-start procedure may be advised. In this procedure, several different runs (e.g., 75) of the C-ICA algorithm are performed, each run starting with a different random initialization of the partition matrix $\boldsymbol{P}$ (see step 1); the solution with the lowest loss function value encountered across all runs is retained as the final solution. The aim of this procedure is to minimize the possibility that a local optimum of the C-ICA loss function is retained.

Implementation of the C-ICA algorithm in R software. The C-ICA algorithm and the procedure for model selection (see Section 3.3) are implemented in R-code (R Core Team, 2014). To perform the ICA on the concatenated data for each cluster, the ICA-function in the R package ‘fastICA’ is used (Marchini, Heaton & Ripley, 2013). To enforce the statistical independence of the source signals, this ICA-function optimizes negentropy by means of the fast and robust fixed-point *fastICA* algorithm (Hyvärinen, 1999).

3.3 Model selection

When performing C-ICA, the number of clusters R should be specified a priori. In general, however, no a priori information regarding the optimal number of clusters is present. A way to determine this number consists of running C-ICA with increasing numbers of clusters (e.g., from one up to six) and using a model selection heuristic to identify the optimal number of clusters. To this end, the CHull procedure (Wilderjans, Ceulemans, & Meers, 2013), which aims at finding a model that optimally balances model fit and model complexity, is proposed. In particular, CHull determines the hull solutions at the boundary of the convex hull of a model mis(fit) by model complexity plot in which all fitted solutions are presented. A final solution is retained by making a scree plot (Cattell, 1966) of the hull solutions and selecting in an automated way the solution lying at the elbow of this plot (for more information, see Wilderjans et al., 2013). To quantify model (mis)fit, the (optimal) value of loss function L is adopted, while different options exist to compute model complexity (e.g., the number of clusters, the number of estimated parameters). Note that when a decomposition into a smaller number of components is aimed at (i.e., non-square mixing matrix $\boldsymbol{A}$, see Section 2.1.1), also the optimal number of components Q should be identified. To address this problem, two strategies may be followed. First, when R and Q can be summarized into a single model complexity value (e.g., the number of estimated parameters), the CHull method may be used. Second, a sequential model selection procedure may be adopted that consists of the following two steps: (1) determine the optimal number of clusters R and (2) select, conditional upon the optimal R, the optimal number of components Q (for similar procedures, see De Roover, Ceulemans & Timmerman, 2012; Wilderjans and Ceulemans, 2013). In both steps, a procedure based on scree ratios may be used to determine the optimal number of clusters/components in an automated way. Here, for the first step, the scree ratio for a certain number of clusters r (keeping the number of components fixed at q) is computed as follow:

$$sr_{r|q} = \frac{L_{r-1,q} - L_{r,q}}{L_{r,q} - L_{r+1,q}}, \tag{3.3}$$

where L is the loss function value from equation (3.1) for a C-ICA model with r clusters and q components[4]. After computing equation (3.3) for each possible r $(r = 2, \ldots, R_{max} - 1)$ and each q $(q = 1, \ldots, Q_{max})$, the optimal number of clusters R is determined by averaging $sr_{r|q}$ over all considered number of components q and selecting the number of clusters r that has the largest averaged $sr_{r|q}$-ratio. Next, in the second step, conditional upon the optimal number of clusters R derived in step 1, the optimal number of components Q is determined by selecting the number of components q $(q = 2, \ldots, Q_{max} - 1)$ that maximizes the following $sr_{q|R}$-ratio:

$$sr_{q|R} = \frac{L_{q-1,R} - L_{q,R}}{L_{q,R} - L_{q+1,R}} \tag{3.4}$$

As in all model selection procedures, the final decision about model retention should also be based on the interpretability of the C-ICA solution.

[4] Note that in the first step it is not possible to compute a scree ratio for the smallest (i.e., $r = 1$) and largest (i.e., $r = R_{max}$) number of clusters.

Section 4. Simulation studies

In this section, three simulation studies are presented in which the performance of the C-ICA algorithm in terms of recovering the true partition and cluster specific ICA parameters is evaluated. In the first simulation study, the algorithm is tested when a correct number of clusters and independent components are specified for C-ICA. In the second simulation study, the performance of the C-ICA algorithm is examined under less favourable analysis conditions. In particular, here an incorrect number of clusters is specified. Lastly, a small simulation study is carried out to test the heuristic model selection strategy based on the CHull procedure and the proposed sequential model selection procedure (see Section 3.3). Here, C-ICA solutions are estimated for several values of r and q and it is determined whether the proposed model selection procedure identifies the true values for R and Q.

4.1 Simulation study 1

4.1.1 Problem

In the first simulation study, the C-ICA algorithm was evaluated under optimal conditions. In particular, data were generated from a C-ICA model with a known true number of clusters R and true number of independent components Q and these generated data sets were subjected to a C-ICA analysis using the true number of clusters and components. The C-ICA algorithm is evaluated with respect to goodness of recovery, that is, whether (1) the clustering of the data blocks ($\boldsymbol{P}$) can successfully be recovered, (2) the cluster specific source signals $\boldsymbol{s}^r$ can be correctly disclosed and (3) the time courses $\boldsymbol{A}_i$ can successfully be retrieved.

Furthermore, it is examined whether the performance of the algorithm depends on characteristics of the data (i.e., the number of elements in the independent components and the dimensionality of the mixing matrix) and/or on the complexity of the true underlying C-ICA model (i.e., the number of underlying clusters/components) and/or on the amount of noise in the data. Based on previous research (Brusco & Cradit, 2005; De Roover et al., 2011), expectations are that the C-ICA algorithm will perform better when there are more elements in the independent components (i.e., more available information). Furthermore, it can be postulated that the goodness of recovery will deteriorate with increasing complexity (i.e., more clusters and independent components) of the underlying C-ICA model (Milligan, Soon & Sokol, 1983, De Roover et al., 2011; Brusco & Cradit, 2005) and when the data contain more noise (Brusco & Cradit, 2005). Finally, regarding the dimensionality of the mixing matrix-factor, no clear expectations are available.

4.1.2 Design and procedure

In order to not have an overly complex design, the number of data blocks was fixed at 40 and only clusters of equal size (i.e., 40 divided by the number of clusters *R*) were considered. Furthermore, the five aforementioned factors were systematically varied in a full five-factorial randomized design in which all factors were considered as random factors (i.e., the selected values for each factor were considered as sampled at random from a wider population of values):

1. Number of elements in the independent components, at two levels: 500 and 2000
2. Number of independent components Q, at three levels: 2, 5 and 20
3. Number of (equally sized) clusters R, at two levels: 2 and 4
4. Dimension of mixing matrices $\boldsymbol{A}_i$, at 2 levels: square or non-square
5. The amount of noise *E* in the data, at three levels: 5 %, 20 % and 40 %.

With regard to the fourth factor, in the case of a square mixing matrix, the dimensionality of $\boldsymbol{A}_i$ depends on the number of independent components Q (i.e., either 2x2, 5x5 or 20x20). In the non-square conditions, however, the number of observed mixtures was held constant at 64 (i.e., a number larger than the number of independent components), resulting in the mixing matrix being either 2x64, 5x64 or 20x64. Furthermore, as in the general ICA model, the C-ICA model assumes that the independent components are non-Gaussian. Therefore the Q components for a particular cluster specific $\boldsymbol{s}^r$ were independently generated from one of the following four non-Gaussian distributions:

1. Uniform distribution
2. Laplace distribution
3. Bimodal distribution with equal peaks
4. Bimodal distribution with unequal peaks

Note that this implies that all independent components for a particular $\boldsymbol{s}^r$ were drawn from the same non-Gaussian distribution (e.g., the Q components of $\boldsymbol{s}^1$ following a Laplace distribution). When four clusters were generated, all aforementioned distributions were included (i.e., each of the four $\boldsymbol{s}^r$ was associated with a different distribution); in the conditions with two clusters, two non-Gaussian distributions were selected at random without replacement (i.e., $\boldsymbol{s}^1$ having one of the four distributions and $\boldsymbol{s}^2$ having another one).

All independent components were generated using the `R` function icasamp() from the `ica` package (Helwig, 2014). This function ensures that the generated components have a mean of zero. Moreover, the independent components were mixed with subject specific generated mixing matrices $\boldsymbol{A}_i$, which were drawn from a uniform distribution with mean zero. Additionally, a noise matrix $\boldsymbol{E}_i$ $(J \times K)$ was added to each data block $\boldsymbol{X}_i$ $(J \ \times K)$. Here, each noise matrix $\boldsymbol{E}_i$ was constructed by independently drawing numbers from $N(0,1)$. Next, the noise matrices were rescaled to ensure that computed across all data blocks $\boldsymbol{X}_i$ the data contained the required percentage of noise (i.e., 5%, 20% or 40%).

Lastly, for each cell in the five-factorial design, 10 replication data sets were generated. Thus, in total, 2 (number of elements) × 3 (number of independent components) × 2 (number of clusters) × 2 (dimension of mixing matrix) × 3 (error) × 10 (replications) = 720 C-ICA data sets were generated. Each data set was analyzed with the C-ICA algorithm with 75 random starts, and the solution with the lowest value on the loss function L (see equation 3.1) was retained.

4.1.3 Results

Recovery of the clustering of the data blocks (***P***). In order to evaluate the goodness of recovery for the clustering of the data blocks, the Adjusted Rand Index (ARI; Hubert & Arabie, 1985) is computed between the true partition of the data blocks and the estimated partition. The ARI equals 1 if two partitions are identical and 0 when the overlap between both partitions is at chance level. The overall mean ARI, across all 720 data sets, equals .9823 (*SD* = .0714). Moreover, a perfect recovery of the partition was observed for 652 of the 720 data sets (i.e., 90.56%). It can be concluded that the C-ICA algorithm recovers the clustering of the data blocks to a very large extent.

To study how the recovery of the clustering of the data blocks changes as a function of the manipulated factors, Table 1 gives an overview of the mean ARI (and standard deviation of ARI) for each level of the five manipulated factors. From this table it can be seen that when the amount of noise is low (i.e., 5%), a perfect recovery is encountered for each data set. However, recovery slightly deteriorates when the amount of noise in the data increases (i.e., *M* = .9940 and *M* = .9530 for 20% and 40% of noise in the data, respectively). Moreover, recovery also decreases when (1) there are more clusters, (2) the mixing matrix becomes square, and (3) the number of elements in the components decreases. For the number of components, best recovery results are obtained for intermediate values of Q.

Table 1. Mean ARI and Tucker's congruence value (and standard deviation) for all levels of the manipulated factors

Factor	**Level**	***ARI***	***Tucker's Congruence***
Number of elements in independent components	500	.9807 (.0716)	.8835 (.1019)
	2000	.9840 (.0714)	.9033 (.0681)
Number of independent components Q	2	.9659 (.1046)	.9530 (.0515)
	5	.9940 (.0308)	.9231 (.0380)
	20	.9870 (.0551)	.8042 (.0795)
Number of clusters R	2	.9950 (.0520)	.8935 (.1020)
	4	.9696 (.0848)	.8933 (.0694)
Dimensionality of the mixing matrix	Square	.9694 (.0849)	.8814 (.0879)
	Non-square	.9953 (.0518)	.9054 (.0849)
Amount of noise in the data	5%	1 (.0000)	.8975 (.0805)
	20%	.9940 (.0423)	.8871 (.0926)
	40%	.9530 (.1107)	.8956 (.0880)

Additionally, to evaluate (the importance of) main and interaction effects of the manipulated design factors, an analysis of variance (ANOVA) with ARI as the dependent variable and the five factors from the design as the independent variables was performed. Here, only discussing significant effects (at $\alpha = .05$) with an intraclass correlation $\hat{\rho}$ (Haggard, 1958) larger than .10, this analysis reveals a considerable three-way interaction effect between the *number of clusters*, the *dimensionality of the mixing matrix* and the *amount of noise* ($\hat{\rho}$ = .43). As can be seen in Figure 2, when the number of clusters is large (right-hand panel of Figure 2), the recovery of the partitioning deteriorates when the mixing matrix becomes square and this especially when the data contain a large amount of noise. When only a small number of clusters underlies the data, however, the opposite is observed (see left-hand panel of Figure 2).

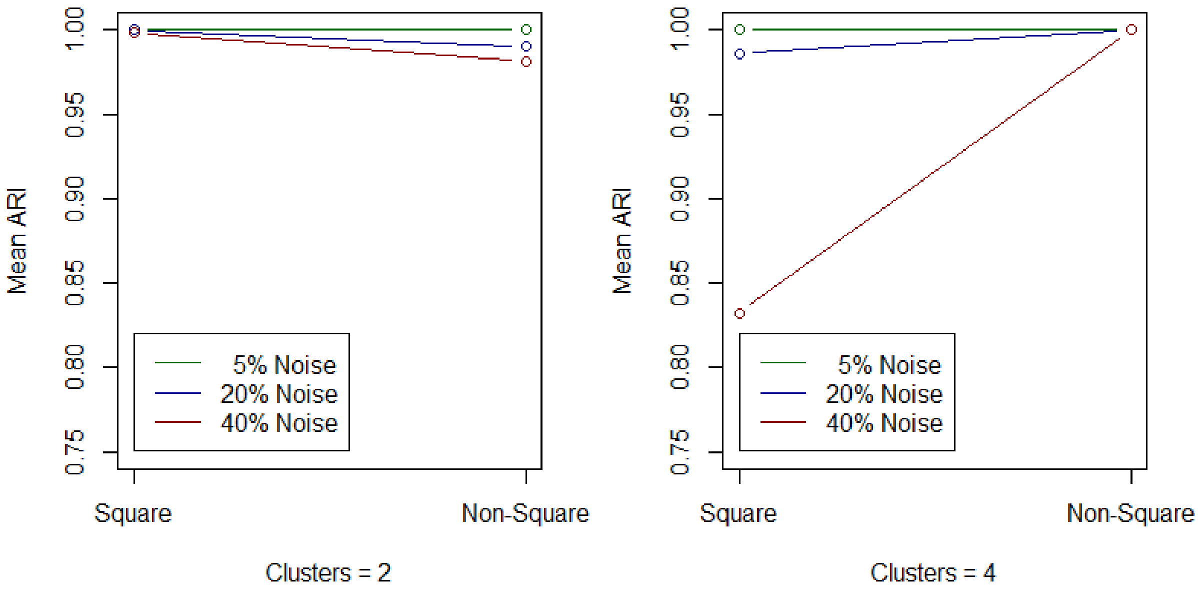

Figure 2. Recovery of the clustering of the data blocks (in terms of ARI) as a function of a three-way interaction between factors *Amount of noise, Dimensionality of the mixing matrix* and *Number of clusters*.

Recovery of the independent components ($\boldsymbol{s}^r$). To evaluate the extent to which the true independent components were recovered, for each component separately, the Tucker congruence coefficient (Tucker, 1951) is computed between the simulated independent component and the corresponding estimated independent component. To arrive at a single Tucker congruence value for each $\boldsymbol{s}^r$-matrix, for each of the Q components of that $\boldsymbol{s}^r$ the Tucker value is computed (after accounting for the C-ICA ambiguities – see further) and the mean across these Q obtained Tucker values is calculated. To obtain a single Tucker value for each generated data set, the (mean) Tucker values of the R $\boldsymbol{s}^r$'s were averaged. Tucker's congruence coefficient equals the normalized inner product between two vectors and ranges from -1 to 1, with 1 indicating perfect recovery.[5] A value in the range of .85-.94 denotes a fair similarity between the two vectors, whereas a value larger than .95 indicates that the two vectors are very similar (Lorenzo-Seva & ten Berge, 2006).

Determining the extent to which the simulated independent components are recovered by the estimated independent components is not straightforward as the C-ICA model suffers

[5] -1 indicates a perfect agreement between two vectors with the orientation of one of these vectors being reversed (i.e., a reflection).

from four ambiguities (see Section 2.2.2): scaling, reflectional and component and cluster permutational freedom. To take these ambiguities into account when computing Tucker's congruence, the following procedure was followed: (1) the absolute value of the Tucker coefficient is taken to account for reflectional freedom; (2) to account for the permutational freedom of both the components and the clusters, all possible combinations of cluster and component permutations are considered and for each combination of these permutations the associated mean Tucker congruence (averaged across components and $\boldsymbol{s}^r$'s) is computed. Next, the combination of cluster and component permutation with the largest Tucker congruence value is retained and the associated averaged Tucker value is reported. Note that as the Tucker coefficient is invariant under a scaling of the components with a positive scalar, this coefficient automatically accounts for the scaling ambiguity of the C-ICA model.

As the overall mean Tucker congruence equals .8934 (*SD* = .0871), it can be concluded that the C-ICA algorithm recovers the independent components reasonably well. The mean Tucker congruence value (and standard deviation) for each level of each factor can be found in Table 1. From this table it can be seen that the mean Tucker congruence especially varies as a function of the number of independent components Q, with recovery deteriorating when Q increases (*M* = .9530 versus *M* = .8042 for 2 and 20 components, respectively). Further, recovery of the independent components also decreases when (1) the number of elements in the independent components decreases, (2) the mixing matrix becomes square and (3) the data contain (more) noise.

To evaluate the (importance of the) effects of the manipulated design factors, an analysis of variance with the mean Tucker congruence value as the dependent variable and the five factors from the design as independent variables was performed. As in the analysis of variance with ARI as the dependent variable, only significant effects (at $\alpha = .05$) with an intraclass correlation larger than .10 are discussed. This analysis showed that there is a strong main effect of the *number of components* ($\hat{p}$ = .39): recovery decreases with an increasing number of independent components (see also Table 1). However, this main effect was involved in a sizeable five-way interaction ($\hat{p}$ = .15) between all design factors. The overall tendency of this complicated interaction effect can be summarized as follows: conditional on a large number of independent components (i.e., $Q = 20$), the recovery of the signals seems to increase when the mixing matrix becomes square, but only if the number of elements in the signals is small (i.e., 500). However, when the number of elements in the signals is large (i.e., 2000), the recovery of the signals deteriorates when the mixing matrix becomes square, with this improvement being more pronounced when the number of clusters is small (i.e. $R = 2$

compared to 4).

Recovery of the time courses ($\boldsymbol{A_i}$). In order to evaluate to what extent the time courses (i.e., mixing matrices $\boldsymbol{A_i}$) are recovered, Tucker's congruence coefficient (Tucker, 1958) is computed between each simulated and estimated mixing matrix. This measure, denoted by Tucker's mean $\boldsymbol{A}$ (TMA), is computed in a similar way as was done for determining the recovery of the independent components, herewith accounting for the C-ICA ambiguities (i.e., scaling and reflectional ambiguity and component permutational freedom but not cluster permutation freedom) in the same way as before (see earlier).

The mean TMA across all data sets is .7419 (*SD* = .1920). Therefore, it can be concluded that the C-ICA algorithm does not recover the associated time courses very well. In Table 2, in which the mean TMA for each level of the manipulated factors in the design is presented, it can be seen that recovery especially decreases when (1) there are more independent component underlying the data ($\hat{p}$ = .83) and (2) the mixing matrix becomes square ($\hat{p}$ = .11).

Table 2. Average Tucker's mean ***A*** value (and standard deviation) for all levels of the manipulated factors.

Factor	**Level**	***Tucker's mean* A (TMA)**
Number of elements in independent components	500	.7440 (.1927)
	2000	.7397 (1914)
Number of independent components Q	2	.9184 (.0871)
	5	.7959 (.1180)
	20	.5112 (.0410)
Number of clusters R	2	.7442 (.1936)
	4	.7395 (.1906)
Dimension of the mixing matrix	Square	.6621 (.1529)
	Non-square	.8216 (.1941)
Amount of noise in the data	5%	.7454 (.1943)
	20%	.7413 (.1918)
	40%	.7388 (.1905)

4.2 Simulation study 2

4.2.1 Problem and design

In the second simulation study, the performance of the C-ICA algorithm is examined under less favorable conditions, that is, with an incorrect number of clusters. To this end, data were simulated according to the same design and procedure as in the first simulation study, but now only taking 5 replications per cell of the design. Each of the 360 generated C-ICA data sets was analyzed (using 75 random starts) twice: (1) with the true number of clusters R and (2) with one cluster too many (i.e., R+1).

For this simulation study, the main interest is to what end specifying an incorrect number of clusters affects the recovery of the clustering. When too many clusters are extracted, expectations are that one true cluster will be split into two (or more) subclusters. Additionally, since the C-ICA algorithm ensures that empty clusters do not occur (see section 3.2), it is expected that specifying one cluster too many may result in one estimated cluster containing a single data block. This single data block truly belongs to another cluster, but has the worst fit for that cluster. This behavior of the C-ICA algorithm is especially expected when the data contain no noise (or at least a minimum amount of noise, i.e., 5%) as in that case the optimal partition into $R + 1$ clusters is a clustering where the worst fitting block from the R cluster solution is assigned to a separate (singleton) cluster.

4.2.2 Results

The mean ARI across all generated datasets equals .9752 (*SD* = .0914) when a correct number of clusters R is specified. As expected, when one cluster too many is specified (i.e., $R + 1$), the C-ICA algorithm recovers the partitioning of the data blocks to a smaller extent (*M* = .8992, *SD* = .0899). From Table 3, in which the mean ARI for each level of each design factor is presented for both R and R+1, it appears that when a correct number of clusters is specified, recovery decreases when (1) the number of independent components Q decreases, (2) the number of clusters R increases, (3) the mixing matrix becomes square, and (4) the data contain more noise. Note that these tendencies have also been observed in the first simulation study. An analysis of variance with ARI as the dependent variable shows a three-way interaction between the *number of clusters*, the *dimensionality of the mixing matrix* and the *amount of noise* ($\hat{p}$ = .23) that has also been found in simulation study 1 (albeit being stronger there). When $R + 1$ clusters are extracted, as can be seen in Table 3, recovery decreases when (1) the independent components have less elements, (2) the number of clusters decreases, and (3) the data are more noisy. Further, an analysis of variance with ARI as the dependent variable showed a significant five-way interaction effect ($\hat{p}$ = .19). The overall tendency of this complicated interaction effect can be summarized as follow: the recovery of the clustering – in general – improved when (1) less components were extracted, (2) less noise is present in the data, and (3) the mixing matrix is non-square. Moreover, when a large number of components was extracted, ARI generally deteriorated when many clusters were underlying the data. This deterioration was more pronounced when the independent components had a smaller number of elements.

In order to examine how frequent a single data block is allocated to a single cluster when one cluster too many is specified, a dichotomous variable *single membership* (being 1 when there is one estimated cluster that contains a single member and 0 otherwise) is computed for all 360 C-ICA data sets. As expected, the occurrence of a single data block in one estimated cluster was observed more frequent when noise is minimal (i.e., 73%, 58% and 37%, for 5%, 20% and 40% of noise, respectively).

Further, when only considering the data sets where no *single memberships* occurred, it was examined how often one true cluster was split into two estimated clusters (as opposed to being split into more than two clusters). Note that this implies that the other $R - 1$ true clusters were recovered perfectly. Results show that when one cluster too many (i.e., R+1) is estimated, the percentage of data sets where one true cluster is split into two estimated clusters, increases when noise decreases (i.e., 67%, 90% and 100% for 40%, 20% and 5% of noise, respectively).

In fact, the percentage of data sets in which either a *single membership* occurred or a true cluster was split into two clusters, increased from 79% to 96% to 100% when the noise decreased from 40% to 20% to 5%, respectively. It can therefore be concluded that when one cluster too many (i.e., $R + 1$) is specified, the C-ICA algorithm has a strong tendency to split a true cluster into two clusters or allocate exactly one data block to a single cluster; this tendency is more pronounced when the data only contain a small amount of noise.

Table 3. Mean ARI (and standard deviation) for R and R+1 specified clusters (for the same 360 generated C-ICA data sets) as a function of the manipulated factors.

Factor	**Level**	$\boldsymbol{R}$ **clusters**	$\boldsymbol{R+1}$ **clusters**
Number of elements in	500	.9797 (.0768)	.8860 (.0902)
independent components	2000	.9708 (.1039)	.9125 (.0879)
Number of	2	.9540 (.1317)	.8968 (.0944)
independent	5	.9881 (.0577)	.9141 (.0789)
components Q	20	.9836 (.0617)	.8868 (.0942)
Number of	2	.9902 (.0758)	.8680 (.0904)
clusters R	4	.9603 (.1027)	.9305 (.0779)
Dimension of	Square	.9603 (.1027)	.8973 (.0899)
the mixing matrix	Non-square	.9902 (.0758)	.9012 (.0901)
Amount of noise	5%	1 (.0000)	.9219 (.0763)
in the data	20%	.9983 (.0107)	.9133 (.0758)
	40%	.9274 (.1470)	.8625 (.1037)

4.3 Simulation Study 3

4.3.1 Problem and design

In the third simulation study, two heuristic model selection procedures, one based on CHull and a sequential procedure (see Section 3.3), were compared. The goal of this simulation study is to examine to what extent these model selection procedures are able to correctly identify the true number of clusters R and number of independent components Q underlying the C-ICA model. To this end, 4 C-ICA data sets were generated with either: (1) $R = 2$ or 4 true clusters, and (2) $Q = 2$ or 4 true independent components. The number of elements in each component was kept fixed at 2000. Additionally, all generated independent components were linearly mixed with a non-square mixing matrix of dimension $Q \times 64$. Lastly, the amount of noise in each data block was 5%. Components, mixing matrices and noise were generated as explained before (see Section 4.1.2).

For each of the 4 data sets, a C-ICA was performed with 75 (random) multiple starts and with q and r both varying from 1 to 5. In each analysis, the solution with the lowest loss function value (3.1) was retained. Thus, in total 4 × 5 × 5 = 100 C-ICA analyses were performed.

To quantify model complexity in the case when the CHull procedure was adopted, the total number of estimated parameters of a C-ICA solution was taken (i.e., total number of estimated elements across all $\boldsymbol{s}^{\boldsymbol{r}}$ and $\boldsymbol{A}_i$'s together). Furthermore, the value for loss function L (3.1) was used as a measure for model (mis)fit; this value was also used to compute scree ratio values for equations (3.3) and (3.4) in the case when a sequential model selection procedure was adopted.

4.3.2 Results

Results show that for the generated C-ICA data sets, the sequential model selection procedure outperforms CHull. In particular, the sequential method identified the correct model (i.e., the model with the true number of clusters R and components Q) in three out of four cases, whereas CHull never retained the correct model although the correct model was always located on the boundary of the convex hull. The correct model, however, was always the most complex hull model so that it could not be selected by CHull as CHull cannot select the most simple and the most complex hull model (see Section 3.3). As an illustration of this, Figure 3 shows the CHull plot for the generated data set with $R = 4$ and $Q = 4$ (CHull plots for the other data sets are presented in Appendix I, Figures 1-3). In this figure, one can see that CHull erroneously selects a model with $R = 2$ and $Q = 4$ (i.e., indicated by a green circle). However, the correct model (i.e., 4 clusters and 4 components) is model number 21. This model lies on the convex hull, but it cannot be selected by CHull as it is the last (most complex) model among all hull models.

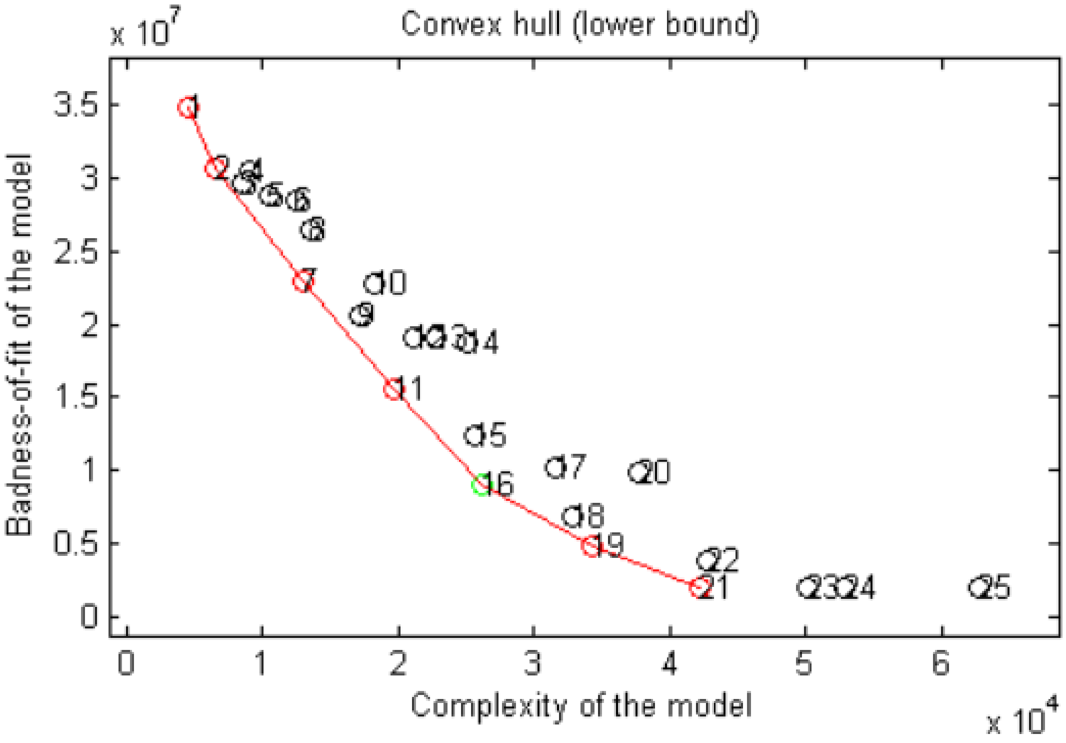

Figure 3. CHull plot for the generated data set with four true clusters ($R = 4$) and four true components ($Q = 4$). C-ICA analyses with r and q ranging from 1 up to 5 have been performed. The model indicated by a green circle ($R = 2, Q = 4$) is selected by CHull. Note that the true model is hull model number 21 ($R = 4, Q = 4$).

To illustrate the sequential model selection procedure, the results of this procedure for the generated data set with $R = 4$ and $Q = 4$ are presented in Table 4 and Figure 4 (results for the other data sets are presented in Appendix I, Tables 1-2 and Figures 4-5).

Table 4. Sequential procedure applied to the generated data set with four true clusters ($R = 4$) and four true components ($Q = 4$). Scree ratios $sr_{r|q}$ for the number of clusters r ($r = 2, \dots, R_{max} - 1$) given the number of components q ($q = 1, \dots, Q_{max}$) and the mean scree ratios over components are displayed. The largest scree ratio in each column is highlighted in bold.

Number of clusters ***R***	**q = 1**	**q = 2**	**q = 3**	**q = 4**	**q = 5**	**Mean *sr* over components**
2	**3.61**	3.40	3.50	3.32	4.05	3.55
3	1.55	1.45	1.46	1.48	1.60	1.51
4	1.71	**3.67**	**7.25**	**520.54**	**264.39**	**159.51**

Table 4 shows the computed scree ratios $sr_{r|q}$ from step 1 of the sequential procedure (see Section 3.3). As the highest mean scree ratio over components is obtained for $r = 4$ (i.e., mean $sr_{r|q}$ = 159.51), the optimal number of clusters equals 4. Only considering C-ICA solutions with four clusters, the scree ratio values $sr_{q|R=4}$ (see equation 3.4) are 1.08, 1.10 and 1081.84 for q = 2, q = 3 and q = 4, respectively, resulting in the selection of the solution with four components. Thus, according to the sequential model selection procedure, the model with R = 4 clusters and Q = 4 components should be retained, which is the correct model underlying the data. The same conclusion can be drawn when looking at Figure 4 which shows a scree plot in which the number of components q $(q = 1, \ldots, 5)$ is plotted against the loss function value (3.1) for C-ICA solutions with different numbers of clusters r $(r = 1, \ldots, 5)$. This figure clearly shows that the decrease is the loss function value levels off when more than 4 components are retained and that the four-cluster solutions fit best.

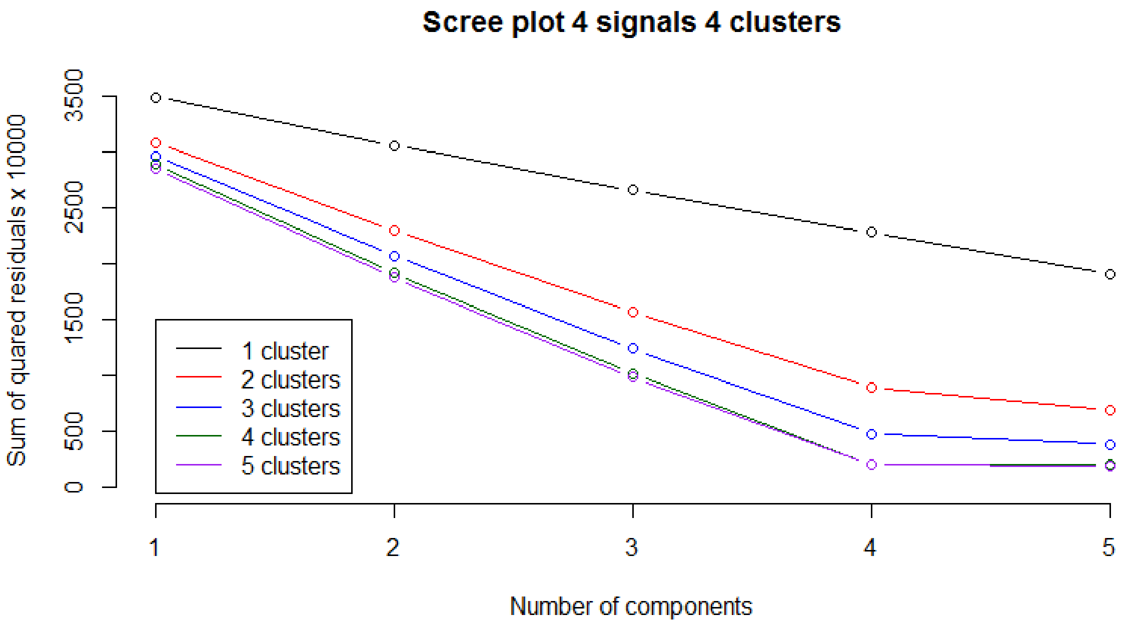

Figure 4. Scree plot for the generated data set with four true clusters $(R = 4)$ and four true components $(Q = 4)$. For all C-ICA solutions with the number of clusters and components varying between one and five, the number of components is plotted against the loss function value. Solutions with the same number of clusters are indicated in the same colour and connected by a line.

The sequential model selection procedure failed to select the correct model for the simulated C-ICA data set with $R = 4$ true clusters and $Q = 2$ true components. As shown in Table 5, the largest mean scree ratio sr over components is encountered for $r = 2$, implying that for this data set the optimal number of clusters should be 2. Calculating the scree ratios in the second step should therefore be conditional on $R = 2$, which is not the true number of clusters for this data set. Here the $sr_{q|R=2}$ values for the solution with 2, 3, and 4 components are 4.07, 1.15, and 492.54, respectively, resulting in the sequential procedure – incorrectly – indicating that the optimal number of components is 4. As such, the sequential procedure erroneously retains the solution with two clusters ($R = 2$) and four components ($Q = 4$).

Table 5. Sequential procedure applied to the generated data set with four true clusters ($R = 4$) and two true components ($Q = 2$). Scree ratios $sr_{r|q}$ for the number of clusters r ($r = 2, \dots, R_{max} - 1$) given the number of components q ($q = 1, \dots, Q_{max}$) and the mean scree ratios over components are displayed. The largest scree ratio in each column is highlighted in bold.

Number of clusters *R*	**q = 1**	**q = 2**	**q = 3**	**q = 4**	**q = 5**	**Mean *sr* over components**
2	3.77	3.86	6.18	**853.61**	**488.23**	**271.13**
3	**4.03**	1.23	1.15	1.23	1.46	1.82
4	.31	**950.70**	**281.01**	1.33	.76	246.82

However, the scree plot (see Figure 5) for this data set tells a different story. Here, for the number of clusters r going from 2 to 5, the decrease in loss functions values cease at $q = 2$ components indicated by the ‘elbow’) and not at $q = 4$ components as suggested in the second step of the sequential procedure. Moreover, the mean sr over all components for $r = 4$ (mean $sr = 246.82$; see Table 5) is only a bit smaller than the mean sr over components for $r = 2$ (mean $sr = 271.13$). Computing the scree ratios for $r = 4$ (i.e., $sr_{q|R=4}$ equals 2512.78, .85 and 1.05 for the solution with 2, 3, and 4 components, respectively) results in the selection of the solution with $Q = 2$ components (and $R = 4$ clusters), which corresponds with the true model underlying this data set.

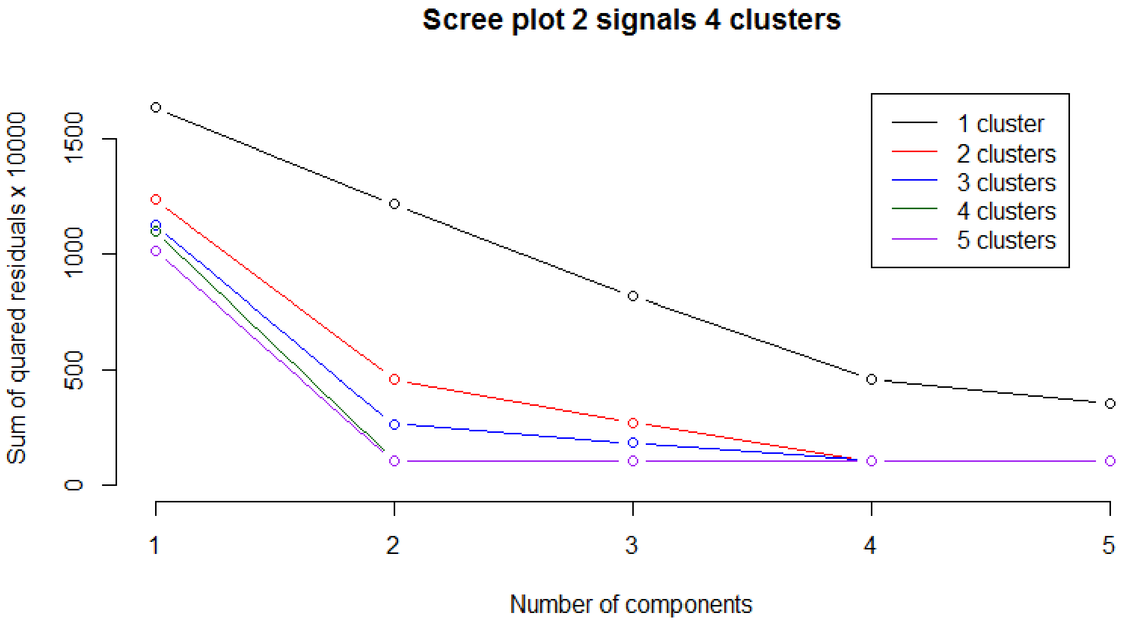

Figure 5. Scree plot for the generated data set with four true clusters ($R = 4$) and two true components ($Q = 2$) . For all C-ICA solutions with the number of clusters and components varying between one and five, the number of components is plotted against the loss function value. Solutions with the same number of clusters are indicated in the same colour and connected by a line.

The results for the C-ICA data set with 4 true clusters and 2 true components brought forth an important reminder for model selection procedures: when determining the optimal number of clusters and components for a range of C-ICA solutions, *both* the scree ratio values and the scree plot should be consulted. Additionally, a final decision about which C-ICA model to retain should always take the interpretability of the solution into account.

Section 5. Discussion

5.1 Discussion of the results

In this master thesis, a new model (C-ICA) was proposed that combines an exploratory clustering technique with ICA in order to identify differences (and similarities) in connectivity patterns between groups of patients (e.g., patients with Alzheimer's disease). In this model, patients are clustered into homogeneous groups such that patients in the same group have the same functional connectivity patterns (i.e., captured by the underlying independent components) and patients belonging to different groups can be characterized by means of connectivity patterns that are qualitatively different. Additionally, to estimate the parameters of the C-ICA model, an alternating least squares type of algorithm was constructed. Further, in order to determine the optimal number of clusters underlying a data set at hand, two model selection procedures were proposed. First, a CHull based procedure that determines the optimal number of components and clusters by balancing model (mis)fit and model complexity. Second, a two-step (sequential) procedure in which, first, the optimal number of clusters is determined, and, next, conditional on this optimal cluster number, the optimal number of components is selected; an automated scree test like procedure based on computing ratios is used in both steps. Finally, two extensive simulation studies were carried out to evaluate the performance of the C-ICA algorithm in terms of recovering the C-ICA parameters (i.e., the clustering, independent components and mixing matrix) and it was investigated whether performance depends on specific data characteristics. The two proposed model selection procedures were evaluated in a (smaller) third simulation study. In the following, the results of the three simulation studies are summarized and their implications are discussed.

Overview of the results of the three simulation studies. The first simulation study shows that the C-ICA algorithm performs well in recovering the underlying C-ICA parameters when a correct number of clusters R and components Q is selected. As expected, the C-ICA algorithm performs somewhat worse when the underlying C-ICA model is complex (i.e., consisting of many clusters) and when the data are very noisy. This worse performance, however, only is somewhat problematic in the case of a square mixing matrix. With respect to the recovery of the independent components, the performance of the C-ICA algorithm can be considered good. In line with previous studies on similar models (See Brusco & Cradit, 2005; De Roover et al., 2011; Milligan, Soon & Sokol, 1983), C-ICA, however, encounters some difficulties when the complexity of the underlying C-ICA model increases (i.e., more independent components). Additionally, C-ICA recovers the independent components better when these components contain a large number of elements and when the mixing matrix is non-square, a result that also has been observed for the recovery of the clustering. Regarding the recovery of the time courses, less good recovery results are obtained. The C-ICA algorithm especially performs weak when the underlying C-ICA model is complex (i.e., having many independent components) and when the underlying mixing matrix is square. Note that square mixing matrices also posed problems for the recovery of the clustering and the independent components.

The main goal of the second simulation study is to evaluate to what end the C-ICA algorithm can successfully recover the true cluster partition when an incorrect number of clusters is specified (i.e., $R + 1$). The results show that when one cluster too many is specified, the C-ICA algorithm often assigns exactly one data block (i.e., person) to one cluster or splits one true cluster into two (estimated) clusters. The C-ICA algorithm shows this tendency especially when the amount of noise in the data is low.

In the third simulation study, the two proposed model selection procedures are compared to each other for a limited number of data sets. It appears that the sequential procedure selects the correct model most of the time. When the procedure fails to identify the correct model, the two-step scree ratio test tells a different story than a scree plot analysis, highlighting the need of considering both pieces of information when selecting an optimal model. The CHull procedure always retains a less complex model than the true model. The true model, however, always lies on the convex hull, but is never selected as it is the most complex hull model.

Implications of the simulation results. The simulation studies demonstrated that the C-ICA algorithm uncovers the true partition to a very large extent. The independent components are recovered quite well by C-ICA, but the disclosure of the mixing matrices is not completely satisfactory. The good recovery of the clustering may in part be caused by the fact that true independent components that belong to the same cluster are drawn from the same distribution. As such, the true clusters are clearly separated (i.e., different) from each other as the signals from different clusters come from clearly different distributions. It may be expected that when two true clusters contain signals from the same distribution, it will be more difficult for the C-ICA algorithm to distinguish these two clusters from each other.

Regarding the independent components for a certain cluster, it should be noted that they are generated component by component by drawing entries from a pre-specified distribution. As such, it is not explicitly enforced that the components are statistically independent from each other. When the true components underlying a data set are not statistically independent, the C-ICA model is a misspecified model for that data set as C-ICA forces the estimated components to be independent. In that case, it cannot be expected that C-ICA fully discloses the true underlying components. It is our guess that the adopted generation procedure for the independent components does not always yield components that are fully statistically independent. When this is true, this may explain why the recovery performance of the independent components is a bit weaker than the performance for the clustering. Another factor that may negatively affect the estimation of the independent components is the (R implementation of the) *fastICA* algorithm that has been used. To the best of our knowledge, no information is available on the recovery performance of the function to perform ICA that has been adopted (i.e., the function 'fastICA' in the R package 'fastICA'; Marchini, Heaton & Ripley, 2013) in the third step of the C-ICA algorithm.

The recovery of the mixing matrices $\boldsymbol{A}_i$ is not completely satisfactory. In part, this may be a consequence of the fact that the independent components in $\boldsymbol{s}^r$ are not disclosed perfectly. Indeed, as both $\boldsymbol{A}_i$ and $\boldsymbol{s}^r$ show up in the C-ICA model formula (2.12), estimating one of both terms suboptimally will negatively impact the estimation of the other term in (2.12). Moreover, the estimation of the block specific mixing parameters in $\boldsymbol{A}_i$ relies on a (much) smaller amount of information/data than the estimation of the clustering $\boldsymbol{P}$ or the spatial maps in $\boldsymbol{s}^r$. Specifically, to estimate each $\boldsymbol{A}_i$, only the information in the corresponding $\boldsymbol{X}_i$ can be used, whereas the information in all $\boldsymbol{X}_i$ is used to estimate $\boldsymbol{P}$ and the information in all $\boldsymbol{X}_i$ belonging to cluster r to estimate the corresponding $\boldsymbol{s}^r$ $(r = 1, \dots, R)$. A possible way to improve the recovery of the $\boldsymbol{A}_i$'s is to assume that the $\boldsymbol{A}_i$ of patients belonging to the same cluster are proportional to each other (i.e., like in Tensor PICA, see Section 1 and 5.3). As such, the estimation of each $\boldsymbol{A}_i$ will be based on more information/data, which will positively impact their recovery.

In the second simulation study it was demonstrated that identifying the correct number of clusters R is important as recovery appears to deteriorate when R is chosen too large (with the same probably also being the case when R is taken too small). Having a good procedure for model selection is therefore of utmost importance. Therefore, in the third simulation study, two model selection procedures were compared. It appeared that the sequential procedure outperforms the CHull based method. Although the correct model always was located on the convex hull, CHull could not select the true model as this model was always the most complex hull model. A way to solve this problem with CHull is to include more complex models in the comparison so that the probability increases that also more complex models (than the true model) are located on the convex hull.[6] It can be concluded that, when applying C-ICA to empirical data, it is of crucial importance that enough complex models (i.e., with many clusters and/or components) are explored.

A general result encountered in the first simulation study is the better recovery performance for all C-ICA parameters when the mixing matrix is non-square. This is a fortunate result since in many applications one is only interested in deriving a smaller set of q signals from the n observed mixture signals (see Section 2.1). For example, in large data sets, which often are encountered in neuroimaging applications such as fMRI (and EEG, see Section 5.2), often the most important information is contained in a limited number of independent components only.

In the following, some directions for further research will be presented. First, a possible interesting application of the C-ICA model to multi-subject EEG data will be sketched. Next, an adaptation of the C-ICA algorithm and three possible extensions of the C-ICA model will be discussed. Some concluding remarks will close off this thesis.

5.2 Directions for future research I: Application of C-ICA to multi-subject EEG data

Although the current study focused on developing a model that is able to analyse multi-subject fMRI data, the C-ICA model may also be applied to other types of data as long as it can be assumed that there are groups in the data and that the underlying (group specific)

[6] Note that in the third simulation study, a (limited) number of more complex models than the true model was included in the comparison (i.e., the true R and Q was maximally 4, whereas models have been fitted with r and q going up to 5). Probably these more complex models were not retained as hull models because of the low amount of noise (i.e., 5%) that has been added to the data.

components are non-Gaussian and independent. For example, C-ICA may be used to study individual/group differences in multi-subject electroencephalography (EEG) data. Note that ICA has already successfully been used in many (single subject) EEG applications (see for example Makeig, Jung, Bell, Ghahremani & Sejnowski, 1997; Jung et al., 2001). In the following, a challenging possible application of C-ICA to multi-subject EEG data with respect to error monitoring will be sketched.

EEG and error monitoring. In order to analyse an empirical data set with C-ICA, multi-subject EEG recordings from a study conducted by Kowal et al. (in press) will be used. The researchers in this study wanted to investigate what the effect was of an acute dose of medicinal cannabis on the neural correlates of error monitoring, a cognitive process by which people are able to detect and adjust to errors accordingly (Botvinick, Braver, Barch, Carter & Cohen; 2001). Neuroimaging studies already demonstrated that the monitoring of errors can be assessed by EEG (see Falkenstein, Hohnsbein, Hoorman, & Blanke, 1991; Yeung, Botvinick & Cohen, 2004). More specifically, when a subject makes an erroneous response to a stimulus during, for example, a reaction time task, a specific EEG waveform is elicited from different parts of the brain. As the signal-to-noise ratio in a single EEG trial is typically very low, the EEG signal is averaged over many representations of the stimulus, relative to the onset of the stimulus (Ward, 2010). This averaged signal, called *event-related potential* (ERP), is of specific interest for many neurocognitive and neuropsychological research. Regarding error monitoring, for example, a specific ERP called the *error-related negativity* (ERN) occurs 50-100 ms after a person gave an erroneous response (Falkenstein et al., 1991; Yeung et al., 2004).

Characteristics of the EEG data. The data from Kowal et al. (in press) consist of EEG recordings from 55 chronic cannabis users. An analysis of variance showed that the participants were comparable with regard to demographic variables (e.g., sex and age). The main goal of the study was to investigate the impact of acute cannabis intoxication on error monitoring (represented by the ERN). To this end, a randomized, double blind, between-groups design was used where one group was given a placebo, another a low dose of 5.5 mg Delta-9-tetrahydrocannabinol (THC) and the last group a high dose of 22 mg THC. A previous analysis of this data showed that, compared to the placebo condition, participants in the high dose condition had a diminished ERN when an erroneous response was made during a task. Further, also a diminished amplitude for Pe[7], another ERP used in the study, was observed for both the high and low dose condition when comparing with the placebo condition.

Analysis of the EEG data with C-ICA. As mentioned before, ICA is able to analyse EEG data (Jung et al., 2001). In particular, ICA decomposes EEG data into independent components, which in turn, may represent a specific ERP. As mentioned by Groppe, Makeig and Kutas (2008), one of the most convincing extractions of independent sources related to an ERP, by means of ICA, is for the ERN (also see Debener et al., 2005). As the study of Kowal et al. (in press) mainly focuses on the ERN when studying the effect of cannabis use, C-ICA may be the ideal method to discover for this study group differences in ERN. Group differences may be expected as the study used an experimental design with three treatment groups (i.e., placebo, low and high dose group). As such, C-ICA can be used to test in an unsupervised way whether and how the groups differ in their ERN response.

Expectations regarding the result of applying C-ICA to the EEG data. When applying C-ICA to this data set, expectations are that the C-ICA algorithm is able to cluster the participants of the study into homogenous groups based on the similarities and differences in their ERP response. As the true clustering of the participants (i.e., the used randomization scheme) is known, this information can be used to investigate the efficacy of the clustering obtained by C-ICA. Herewith, it is assumed that the drug intervention truly brought forth a difference in spatial maps for the members of the various groups. In particular, it can be expected that C-ICA discloses the three treatment groups (i.e., placebo, low dose and high dose). On the other hand, it might happen that C-ICA only identifies two groups since only the high dose group differs significantly from the placebo group with respect to the ERN amplitude.[8] In particular, C-ICA may split the low dose group in two subgroups and may allocate one low dose subgroup to the placebo cluster and the other low dose subgroup to the high dose cluster. Of course, as C-ICA is an unsupervised method, it may also be the case that C-ICA indicates that there are no real differences between the participants (and the treatment groups) or that C-ICA identifies a clustering of the participants that does not match the treatment groups partition at all.

[7] Pe is a component that is observed relatively late (i.e., 200 to 500 ms) after an error response. Note that the ERN is measured 50 to 100 ms after an error response.

[8] Note that this does not necessarily mean that the ERN signal for the low dose group exactly matches the ERN signal of the placebo or the high dose group as there still may exist (smaller) differences in ERP between the low dose group and the other two groups.

5.3 Directions for future research II: Algorithm adaptation and model extensions

Including rational starts in the multi-start procedure. As is true for all clustering techniques, the C-ICA algorithm may end up in a local optimum of the C-ICA loss function, which may produce a suboptimal solution and create misleading results (James, Witten, Hastie & Tibshirani, 2013; Steinley, 2003). It is therefore advised to always use C-ICA along with a multiple start procedure. In such a procedure, several C-ICA analyses are run, each run starting with a different initialization of the partitioning matrix $\boldsymbol{P}$, and the solution with the lowest loss function value encountered across all runs is retained. A first direction for future research pertains to the improvement of the multiple start procedure used in the C-ICA algorithm (see Section 3.2). In the simulation study, always a multi-start procedure using 75 random starts (i.e., data blocks are randomly assigned to clusters) has been used. As a random start, in general, will be far off from the optimal solution, it may be advised to also include other types of starting partitions in the multi-start procedure. For example, a *rational* starting partition may be used. The idea of a rational start is to search for a partitioning matrix $\boldsymbol{P}$ that is already close to the optimal $\boldsymbol{P}$. One possible way to arrive at such a 'good' $\boldsymbol{P}$ is to conduct an ICA (for a single subject) on each data block $\boldsymbol{X}_i$ ($i = 1, \dots, I$) separately. Then, for each pair of data blocks, the Tucker congruence coefficient is computed between the estimated independent components for each member of that pair. The resulting Tucker values are collected in a symmetric $I \times I$ (dis)similarity matrix. Finally, a K-means type of clustering with R clusters is performed on the resulting dissimilarity matrix. Alternatively, one can also perform a hierarchical clustering on the dissimilarity matrix. A partition can then be found by cutting the resulting tree at the required number of clusters R. Note that besides including a rational start in the multi-start procedure, one may also generate a set of pseudo-random starts and include these in the multi-start procedure (for an overview of different types of pseudo-random starts, see Ceulemans, Van Mechelen, & Leenen, 2007). A pseudo-random start may be obtained by slightly perturbing a rationally obtained start (e.g., allocating a small amount of data blocks, for example 10%, at random to a different cluster).

Semi-supervised C-ICA to incorporate a priori information on the clustering. When determining the optimal partition of the data blocks, the C-ICA algorithm only looks at the differences in underlying spatial maps between the data blocks (i.e., the clustering is performed in a fully unsupervised way). Sometimes, however, researchers may have some good ideas about parts of the optimal clustering (i.e., semi-supervised clustering). When, for example, above the EEG or fMRI data, also clinical and/or behavioural data (e.g., scores on a cognitive test) about the studied patients are available, the (dis)similarity in scores on these additional data can be used to identify sets of patients that should be allocated to the same cluster. In the same way, also pairs of patients could be determined that certainly do not belong to the same cluster. Note that a priori information on parts of the clustering can also be obtained when clinical/behavioural data are available for only a subset of the patients. In that case, the (dis)similarity in their data can be used to obtain a partition of the patients for which data are present and it can be enforced that this partition is maintained in the final partition which includes all patients (i.e., also those patients for which no clinical/behavioural data are present). A second direction for future research therefore pertains to an extension of the C-ICA model such that a priori (derived) information on the membership of (some of) the patients (or data blocks) can be incorporated in the analysis. A possible way to go here is to adapt the cluster re-assignment step of the C-ICA algorithm (i.e., step 2, see Section 3.2) in such a way that given pairs of patients are always assigned to the same cluster and other given patient pairs are forced to be allocated to different clusters.

Deflational C-ICA to detect common and distinctive spatial maps. A critical assumption of the C-ICA model is that the spatial maps underlying each cluster clearly differ between clusters. In some applications, however, this assumption may not hold as there may exist spatial maps that are shared by all patients (i.e., common spatial maps), together with spatial maps that are specific for a certain patient cluster (i.e., distinctive spatial maps). To search for common and distinctive spatial maps, a C-ICA model extension could be developed in which some spatial maps are forced to be the same across clusters and others are allowed to differ between clusters. To fit such a model to data, a deflational type of algorithm may be constructed that consists of two steps. First, the algorithm extracts a number Q_{equal} of independent components from *all* data blocks simultaneously (before any clustering); these components represent the common spatial maps. Then, in a second step, for each cluster separately (with the clustering and the optimal number of clusters R being unknown), a number Q_{diff} of components is extracted. These components, which represent the distinctive spatial maps, may differ between clusters. Note that the second step boils down to performing a C-ICA analysis on the data after removal of the common components. When applying this extended model to empirical data, a difficult model selection problem arises as not only the optimal number of clusters R and (distinctive) components Q_{diff} needs to be determined (as is the case for C-ICA), but it is also necessary to decide on the number of common components Q_{equal}.

Restriction of the time courses. In C-ICA, the time courses (i.e., $\boldsymbol{A}_i$'s) are allowed to differ for each data block/person. This implies, however, that many parameters have to be estimated (i.e., the entries of all $\boldsymbol{A}_i$'s), allowing noise to compromise the analysis results (i.e., the not so good recovery of the $\boldsymbol{A}_i$'s as observed in the first simulation study, see Section 4.1.3). A final direction for future research therefore involves in restricting the time courses. One way to go is to enforce the mixing matrices for patients belonging to the same cluster to be proportional to each other (see also Section 5.1). This makes sense as it may be assumed that patients in the same cluster show similar time courses. An advantage of this restriction is that there are less $\boldsymbol{A}_i$ parameters that have to be estimated, which considerably may reduce the risk of modelling noise. Note that such a model extension boils down to developing a clusterwise extension of the Tensor PICA model of Beckmann and Smith (2005).

5.4 Concluding remarks

The novel C-ICA model that was presented in this thesis combines an unsupervised clustering technique with a method that decomposes multivariate data into independent components. The goal of C-ICA is to identify differences (and similarities) in underlying connectivity patterns that are crucial for distinguishing patient groups. As such, valuable new insights with respect to the heterogeneity of an existing disease (e.g., patients suffering from different – maybe yet unknown– subtypes of depression or schizophrenia) or regarding the progressive phases of a disorder (e.g., phases of dementia) may be gained, herewith advancing neuropsychological knowledge and research. It is our hope that this thesis may be a first, but decisive, step in this direction.

References

Amari, S.-I., Cichocki, A., & Yang, H. H. (1996). A new learning algorithm for blind source separation. In D. D. Touretzky, M. C. Mozer, & M. E. Hasselmo (Eds.), *Advances in neural information processing systems 8* (pp. 757-763). Cambridge, MA, USA: MIT Press.

Beckmann, C. F., & Smith, S. M. (2005). Tensorial extensions of independent component analysis for multisubject FMRI analysis. *NeuroImage, 25,* 294-311.

Bell, A. J., & Sejnowski, T. J. (1995). An information-maximization approach to blind seperation and blind deconvolution. *Neural Computation, 7,* 1129-1159.

Botvinick, M. M., Braver, T. S., Barch, D. M., Carter, C. S., & Cohen, J. D. (2001). Conflict monitoring and cognitive control. *Psychological Review, 108*, 624-652.

Brusco, M. J. (2006). A repetitive branch-and-bound algorithm for minimum within-cluster sums of squares partitioning. *Psychometrika, 71*, 347-363.

Brusco, M. J., & Cradit, J. D. (2005). ConPar: A method for identifying groups of concordant subject proximity matrices for subsequent multidimensional scaling analysis. *Journal of Mathematical Psychology, 49*, 142-154.

Cardoso, J.-F., & Souloumiac, A. (1993). Blind beamforming for non Gaussian signals, *IEEE Transactions on Signal Processing, 140*(6), 362-370.

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research, 1*, 245-276.

Ceulemans, E., Van Mechelen, I., & Leenen, I. (2007). The local minima problem in hierarchical classes analysis: an evaluation of a simulated annealing algorithm and various multistart procedures. *Psychometrika, 72,* 377–391.

Comon, P. (1994). Independent Component Analysis, a new concept? *Signal Processing, 36,* 287-314.

Debener, S., Ullsperger, M., Siegel, M., Fiehler, K., von Cramon, D. Y., & Engel, A. K. (2005). Trial-by-trial coupling of concurrent electroencephalogram and functional magnetic resonance imaging identified the dynamics of performance monitoring. *The Journal of Neuroscience, 25*(50), 11730-11737.

Dennis, E. L., & Thompson, P. M. (2014). Functional brain connectivity using fMRI in aging Alzheimer’s disease. *Neuropsychology Review, 24*, 49-62.

De Roover, K., Ceulemans, E., & Timmerman, M. E. (2012). How to perform multiblock component analysis in practice. *Behavior Research Methods, 44*, 41-56.

De Roover, K., Ceulemans, E., Timmerman, M. E., Vansteelandt, K., Stouten, J., & Onghena, P. (2012). Clusterwise Simultaneous Component Analysis for analyzing structural differences in multivariate multiblock data. *Psychological Methods, 17*(1), 100-119.

Falkenstein, M., Hohnsbein, M., Hoorman, J., & Blanke, L. (1991). Effects of crossmodal divided attention on late ERP components: II. Error processing in choice reaction tasks. *Electroencephalography and Clinical Neurophysiology, 78*, 447-455.

Friedman, J. H. (1987). Exploratory projection pursuit. *Journal of the American Statistical Association, 82*(397), 249-266.

Gili, T., Cercignani, M., Serra, L., Perri, R., Giove, F., Maraviglia, B., Caltagirone, C., & Bozzali, M. (2011). Regional brain atrophy and functional disconnection across Alzheimer's disease evolution. *Journal of Neurology, Neurosurgery, and Psychiatry, 82*, 58-66.

Grecius, M. D., Srivastava, G., Reiss, A. L., & Menon, V. (2004). Default-mode network activity distinguishes Alzheimer's disease from healthy aging: Evidence from functional MRI. *Proceedings of the National Academy of Sciences of the United States of America, 101*(13), 4637-4642.

Groppe, D. M., Makeig, S., & Kutas, M. (2008). Independent component analysis of event-related potentials. *Cognitive Science Online, 6*(1), 1-44.

Guo, Y., & Pagnoni, G. (2008). A unified framework for group independent component analysis for multi-subject fMRI data. *NeuroImage, 42*, 1078-1093.

Haggard, E. A. (1958). *Intraclass correlation and the analysis of variance*. New York, NY, USA: Dryden.

Helwig, N. (2014). ica: Independent Component Analysis. R package version 1.0-0. http://CRAN.R-project.org/package=ica.

Huber, P. J. (1985). Projection pursuit. *The Annals of Statistics, 13*(2), 435-475.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*, 193-218.

Hyvärinen, A. (1998). New approximations of differential entropy for independent component analysis and projection pursuit. In M. I. Jordan, M. J. Kearns, & S. A. Solla (Eds). *Advances in Neural Information Processing Systems 10* (pp. 273-279). Cambridge MA, USA: MIT Press.

Hyvärinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks, 10*(3), 626-634.

Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent Component Analysis.* New York, NY: John Wiley & Sons.

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks, 13*, 411-430.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning: with applications in R.* New York, USA: Springer.

Jutten, C., & Herault, J. (1991). Blind separation of sources, Part 1: An adaptive algorithm based on neuromimetic architecture. *Signal Processing, 24*, 1-10.

Kiviniemi, V., Kantola, J.-H., Jauhiainen, J., Hyvärinen, A., & Tervonen, O. (2003). Independent component analysis of nondeterministic fMRI signal sources. *NeuroImage, 19*, 253-260.

Kowal, M., van Steenbergen, H., Colzato, L., Hazekamp, A., van der Wee, N., Manai, M., Durieux, J., & Hommel, B. (*in press*). Dose-dependent effects of cannabis on the neural correlates of error monitoring in frequent cannabis users. *European Neuropsychopharmacology*.

Lorenzo-Seva, U., & ten Berge, J. (2006). Tucker’s congruency coefficient as a meaningful index of factor similarity. *Methodology, 2*(2), 57-64.

Makeig, S., Jung, T.-P., Bell, A. J., Ghahremani, D., & Sejnowski, T. J. (1997). Blind separation of auditory event-related brain responses into independent components. *Proceedings of the National Academy of Sciences of the United States of America, 94,* 10979-10984.

Makeig, S., Jung, T.-P., Ghahremani, D., & Sejnowski, T. J. (2000). Independent component analysis of simulated ERP data. In T. Nakada (Ed.), *Integrated human brain science* (pp. 123-146). New York, USA: Elsevier.

Makeig, S., Westerfield, M., Jung, T.-P., Covington, J., Townsend, J., Sejnowski, T. J., & Courchesne, E. (1999). Functionally independent components of the late positive event-related potential during visual spatial attention. *Journal of Neuroscience 19*(7), 2665-2680.

Marchini, J. L., Heaton, C., & Ripley, B. D. (2013). fastICA: FastICA algorithms to perform

ICA and Projection Pursuit. R package version 1.2-0. http:// CRAN.R-project.org/package=fastICA.

Milligan, G. W., Soon, S. C., & Sokol, L. M. (1983). The effect of cluster size, dimensionality, and the number of clusters on recovery of true cluster structure. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 5*, 40-47.

R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Shenton, M. E., Dickey, C. C., Frumin, M., & McCarley, R. W. (2001). A review of MRI findings in schizophrenia. *Schizophrenia Research, 49*, 1-52.

Steinley, D. (2003). Local optima in *K*-Means clustering: What you don't know may hurt you. *Psychological Methods, 8*(3), 294-304.

Stone, J. V. (2004). *Independent Component Analysis: a tutorial introduction.* Cambridge, MA, USA: The MIT Press.

Tucker, L. R. (1951). *A method for synthesis of factor analysis studies* (Personnel Research Section Report No. 984). Washington, DC: U.S. Department of the Army.

Van de Ven, V. G., Formisano, E., Prvulovic, D., Roeder, C. H., & Linden, D. E. J. (2004). Functional connectivity as revealed by spatial independent component analysis of fMRI measurements during rest. *Human Brain Mapping, 22*, 165-178.

Ward, J. (2010). *The student's guide to cognitive neuroscience* (2nd edition). Hove and New York, USA: Psychology Press.

Wilderjans, T. F., & Ceulemans, E. (2013). Clusterwise Parafac to identify heterogeneity in three-way data. *Chemometrics and Intelligent Laboratory Systems, 129*, 87-97.

Wilderjans, T. F., Ceulemans, E., & Meers, K. (2013). CHull: A generic convex-hull-based model selection method. *Behavioral Research Methods, 45*, 1-15.

Yeung, N., Botvinick, M. M., & Cohen, J. D. (2004). The neural basis of error detection: conflict monitoring and the error-related negativity. *Psychological Review, 111*, 9931-959.

Appendix I

CHull model selection results (CHull plots)

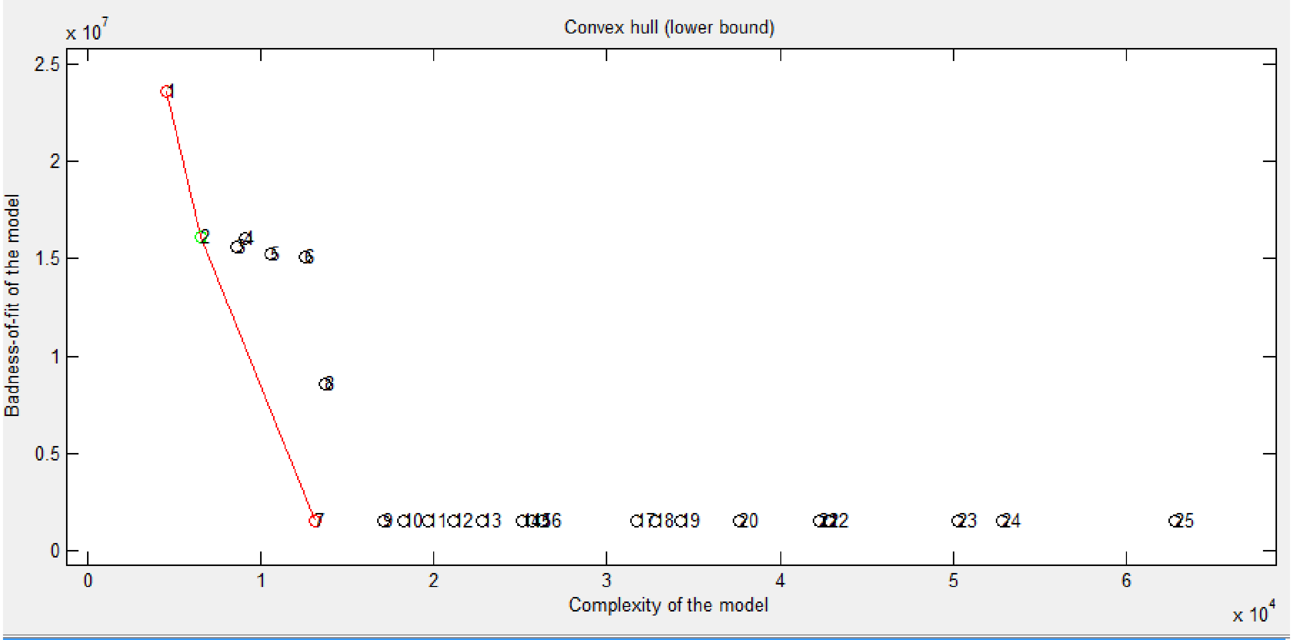

Figure 1. CHull plot for the generated data set with two true clusters ($R = 2$) and two true components ($Q = 2$). C-ICA analyses with r and q ranging from 1 up to 5 have been performed. The model indicated by a green circle ($R = 2, Q = 1$) is selected by CHull. Note that the true model is hull model number 7 ($R = 2, Q = 2$).

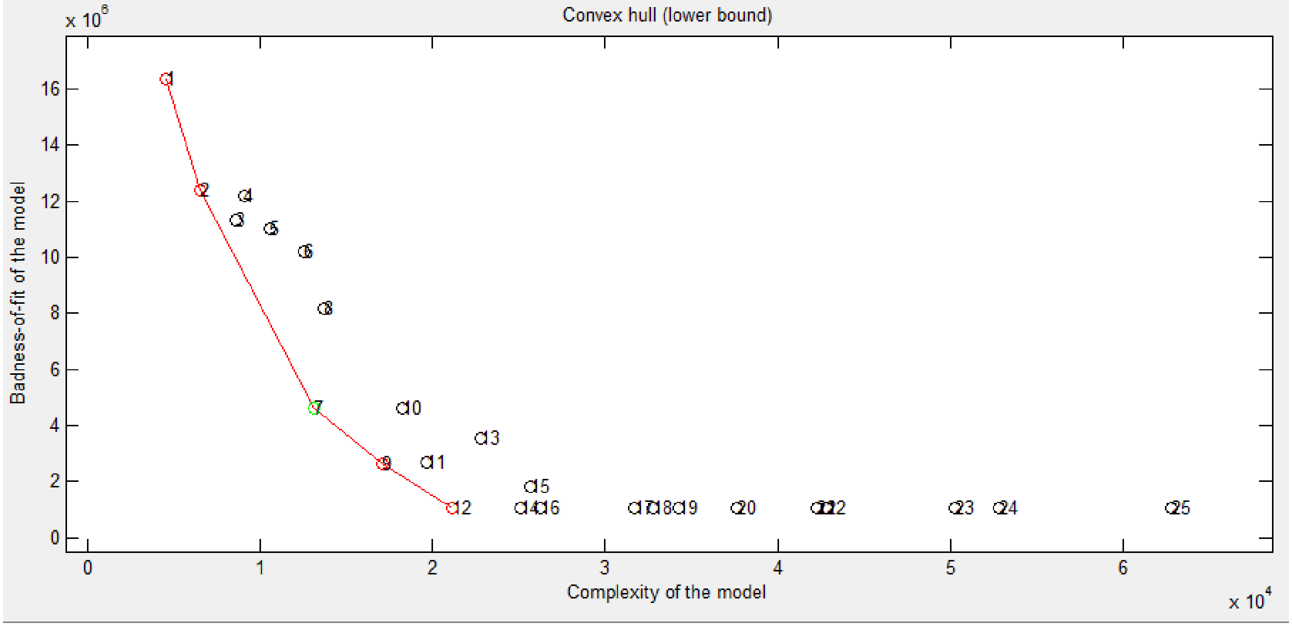

Figure 2. CHull plot for the generated data set with four true clusters ($R = 4$) and two true components ($Q = 2$). C-ICA analyses with r and q ranging from 1 up to 5 have been performed. The model indicated by a green circle ($R = 2, Q = 2$) is selected by CHull. Note that the true model is hull model number 12 ($R = 4, Q = 2$).

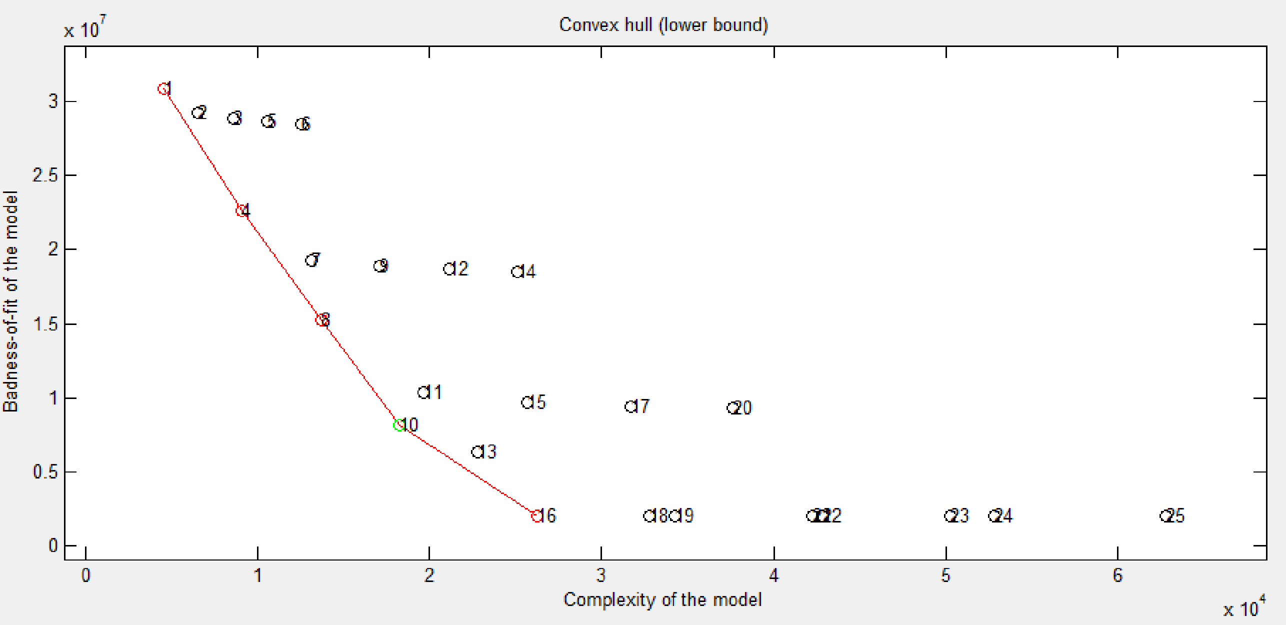

Figure 3. CHull plot for the generated data set with two true clusters ($R = 2$) and four true components ($Q = 4$). C-ICA analyses with r and q ranging from 1 up to 5 have been performed. The model indicated by a green circle ($R = 1, Q = 4$) is selected by CHull. Note that the true model is hull model number 16 ($R = 2, Q = 4$).

Sequential model selection results

Table 1. Sequential procedure applied to the generated data set with two true clusters ($R = 2$) and four true components ($Q = 4$). Scree ratios $sr_{r|q}$ for the number of clusters r ($r = 2, \ldots, R_{max} - 1$) given the number of components q ($q = 1, \ldots, Q_{max}$) and the mean scree ratios over components are displayed. The largest scree ratio in each column is highlighted in bold.

Number of clusters *R*	**q = 1**	**q = 2**	**q = 3**	**q = 4**	**q = 5**	**Mean *sr* over components**
2	3.99	**7.85**	**7.73**	**1100.46**	**558.81**	**335.17**
3	**2.10**	1.97	2.32	.95	1.05	1.68
4	1.48	1.38	2.21	1.10	1.04	1.44

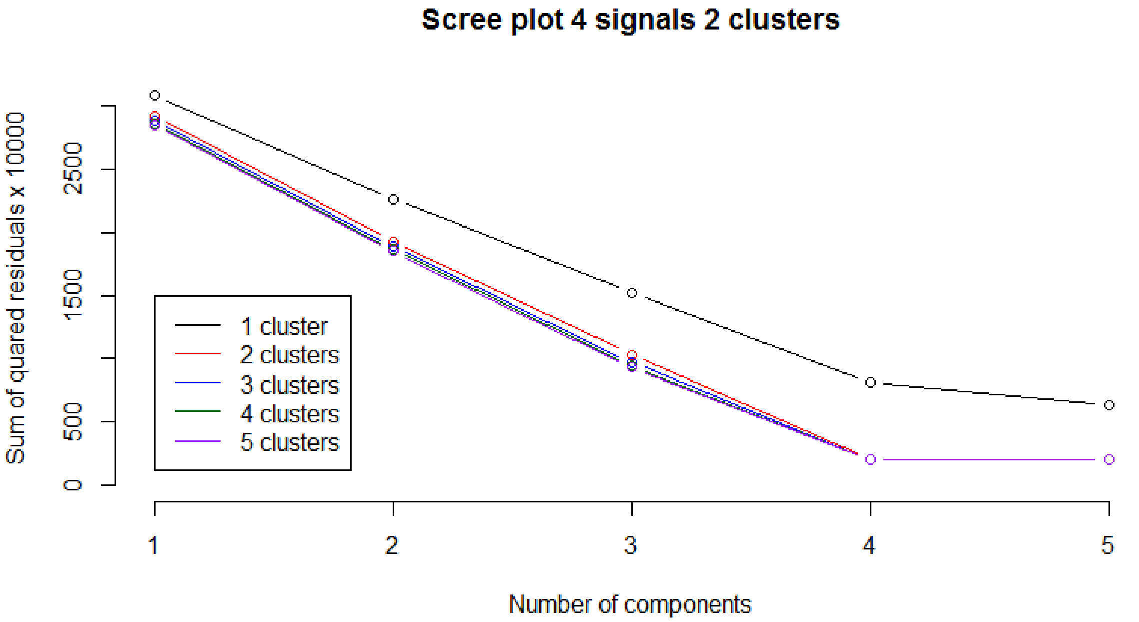

Figure 4. Scree plot for the generated data set with two true clusters ($R = 2$) and four true components ($Q = 4$) . For all C-ICA solutions with the number of clusters and components varying between one and five, the number of components is plotted against the loss function value. Solutions with the same number of clusters are indicated in the same colour and connected by a line.

Table 2. Sequential procedure applied to the generated data set with two true clusters ($R = 2$) and two true components ($Q = 2$). Scree ratios $sr_{r|q}$ for the number of clusters r ($r = 2, \dots, R_{max} - 1$) given the number of components q ($q = 1, \dots, Q_{max}$) and the mean scree ratios over components are displayed. The largest scree ratio in each column is highlighted in bold.

Number of clusters *R*	**q = 1**	**q = 2**	**q = 3**	**q = 4**	**q = 5**	**Mean *sr* over components**
2	**14.85**	**9896.33**	**2906.67**	**1.06**	1.07	**2564.00**
3	1.41	1.04	1.08	.98	.78	1.06
4	2.18	1.03	.98	.72	**1.30**	1.24

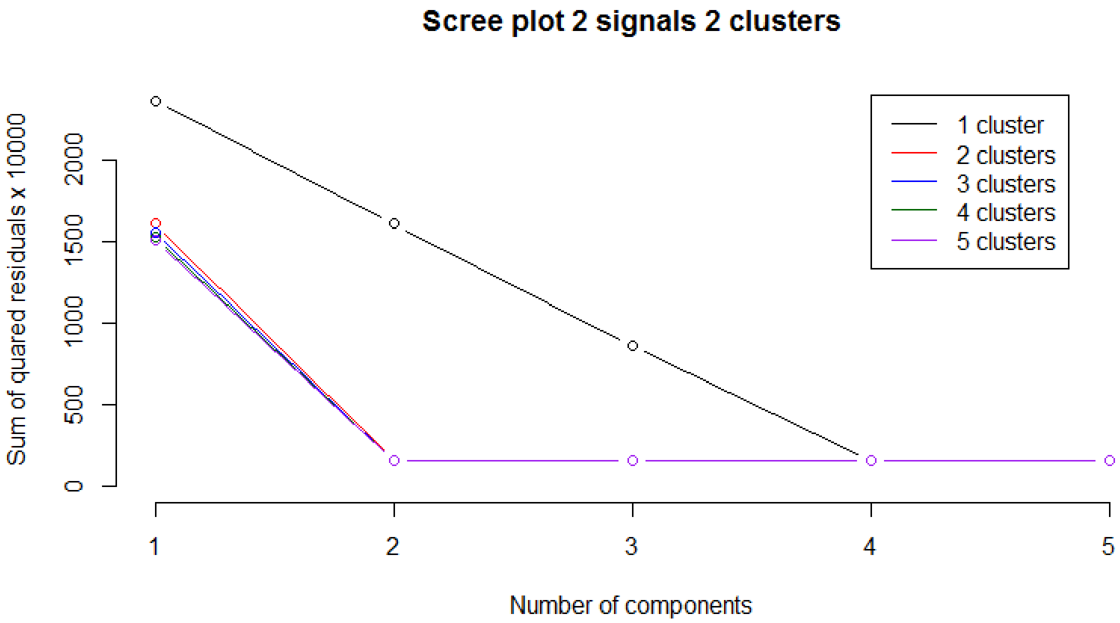

Figure 5. Scree plot for the generated data set with two true clusters ($R = 2$) and two true components ($Q = 2$). For all C-ICA solutions with the number of clusters and components varying between one and five, the number of components is plotted against the loss function value. Solutions with the same number of clusters are indicated in the same colour and connected by a line.