Erasmus University Rotterdam

Erasmus School of Economics

Bachelor Thesis

International Bachelor Econometrics and Operations Research

Comparing the Reliability of Independent Component Analysis Algorithms for Neuroimaging Time Series

Author:

Rohit Khiara

Student Number:

523302rk

Supervisor:

Jeffrey Durieux

Second Assessor:

Utku Karaca

Abstract

Independent Component Analysis (ICA) for neuroimaging time series is a statistical technique which attempts to estimate brain activity for the purpose of medical diagnosis and treatment. However, algorithms which execute ICA often consist of stochastic elements, resulting in estimates of independent components which are not reliable. Thus, there are potentially adverse effects for those that rely on ICA techniques. As a solution, a method to determine the reliability of the estimated independent components is proposed in Himberg, Hyvärinen, and Esposito (2004): the Icasso software package, which provides clusters of the estimated components. These clusters serve as a visual measure of reliability.

Thus, this paper analyses three methods of ICA estimation - Icasso with a FastICA algorithm; Icasso with an Infomax algorithm; and a Minimum Spanning Tree approach with an ICA Entropy Bound Maximisation algorithm - to determine which one provides the most reliable estimates of independent components. The analysis is conducted on a Magnetoencephalograph (MEG) dataset, and a functional Magnetic Resonance Imaging (fMRI) dataset, through the use of the GIFT Toolbox for fMRI analysis in MATLAB.

The results of conducting this analysis indicate that Icasso with FastICA provides reliable estimates of independent components for both datasets, whilst serving as a computationally feasible method of independent component analysis.

July 2022

Contents

1 Introduction

Independent Component Analysis (ICA) is a statistical technique which aims to perform parameter estimation of data in a manner which minimises the statistical dependence between its components. More specifically, ICA provides a linear transformation of multivariate data such as random vectors. Then, each component of this representation is a statistically independent linear combination of the original variables. Variables or components are considered to be statistically independent if the information relating to one variable does not provide any information on the value of the second variable.

Given that ICA is a general-purpose technique, it has a variety of applications; it plays a prominent role in the healthcare sector as it is commonly utilised in the analysis and subsequent interpretation of brain imaging data. Hyvärinen and Oja (2000) provides an example of how ICA is used for the analysis of Electroencephalogram (EEG) data, which consist of recordings of electrical potentials within the brain from numerous locations on the scalp. The EEG provides these recordings by mixing underlying components of brain activity, where mixing refers to the creation of a probabilistic combination. Through ICA, it is possible to obtain independent components of said brain activity for the purpose of further analysis in diagnosing and treating patients or others.

For this reason, and given how widespread its implementation is, it is crucial that ICA as a technique provides estimated independent components that are statistically reliable. That is, multiple runs of the analysis should produce identical or similar results based on certain criteria, otherwise ICA cannot be used appropriately for any purpose. However, a common problem associated with ICA is that it contains stochastic elements, which result from ICA algorithms such as FastICA or Infomax that try to optimise an objective function. Himberg et al. (2004) states that this randomness results from the inability of these algorithms to find a global minimum or maximum while optimising the objective function, due to the high dimensionality of the signal space in which ICA operates. Consequently, multiple runs of the algorithm often produce results that are contrasting.

Thus, to investigate the reliability of the estimated independent components that are obtained from ICA, this paper presents a comparison between three ICA algorithms. Firstly, the procedure which is outlined in Himberg et al. (2004) is replicated. This replication involves an implementation of the Icasso method on Magnetoencephalography (MEG) data. It is important to note that Himberg et al. (2004) is the first piece of work to introduce and implement the Icasso method, which serves to assess the algorithmic and statistical reliability of estimated independent components. In addition, Icasso is then used on functional Magnetic Resonance Imaging (fMRI) data. By doing so, this paper aims to provide additional knowledge on whether Icasso is better suited to a certain type of neuroimaging time series.

The implementation of Icasso on these data employs the FastICA algorithm to select the parameters of the ICA estimation algorithm. To determine whether more reliable independent components for fMRI data can be estimated through Icasso, an Infomax algorithm is utilised instead of FastICA when selecting the parameters of ICA. This method allows for a comparison between FastICA and Infomax, which can potentially demonstrate the strengths of each algorithm. Langlois, Chartier, and Gosselin (2010) offers an extensive discussion and comparison of these algorithms.

Lastly, a novel approach for estimating and interpreting the statistical reliability of ICA for fMRI data, described in Du, Ma, Fu, Calhoun, and Adalı (2014), is considered and evaluated against the previously estimated ICA, which uses Icasso with FastICA and Infomax each. This novel approach uses a Generalised Assignment Problem (GAP) method with an ICA Entropy Bound Maximisation (ICA-EBM) algorithm in conjunction with a Minimum Spanning Tree (MST) to select and assess the reliability of the estimated independent components, thereby enabling a comparison between Icasso and the latter, which is a more recently developed algorithm. Ultimately, it can then be determined if this newer method provides an improved statistical reliability of the ICA algorithm for fMRI data relative to Icasso. To perform the analyses described above, an MEG dataset, obtained from Vigrio, Jousmki, Hmlinen, Hari, and Oja (1998) is used for the replication task in this paper, alongside an fMRI dataset which is obtained from TReNDS (2012).

To this end, this paper investigates the following research question:

Can the statistical reliability of the estimated independent components of ICA for neuroimaging time series be improved by using Icasso with FastICA, Icasso with Infomax, or a Minimum Spanning Tree algorithm with ICA-EBM?

Such analysis has both, theoretical and practical implications. By determining whether the estimated independent components that are obtained through ICA on neuroimaging time series achieve greater reliability through certain methods, this paper supplements existing knowledge, thereby contributing to the academic field and corresponding literature. In fact, Kachenoura, Albera, Senhadji, and Comon (2007) finds that appropriately selecting an algorithm whilst conducting ICA analysis in brain-computer interface (BCI) systems can significantly improve the capabilities of said systems. Therefore, it is necessary to continue testing both existing, traditional methods, as well as novel ones with varying datasets, to contribute to the information regarding ICA algorithms.

Furthermore, research on the reliability of ICA is likely to result in increasingly accurate clinical procedures, thus providing value for all stakeholders that are involved in the healthcare industry.

The remainder of this paper is structured as follows: Section 2 discusses relevant literature and presents additional sub-questions and hypotheses which are investigated in this research. Section 3 describes the datasets in greater detail, whilst Section 4 extensively outlines the statistical techniques and methods which are utilised to achieve the research aims of this paper. Accordingly, Section 5 provides the results and interpretation of conducting said research. Lastly, Section 6 summarises the paper and considers possibilities for further research.

2 Related Work

To investigate the research question in a more focused manner, the following sub-question is considered:

S1: *Does Icasso with FastICA provide more reliable estimated independent components than Icasso with Infomax when using fMRI data?*

As previously discussed, Icasso is a tool which determines the statistical reliability of estimated independent components. In order to perform this procedure, Icasso is commonly used with the computationally-efficient FastICA algorithm, a fixed-point algorithm which is introduced in Hyvarinen (1999). FastICA is responsible for computing and the estimated components based on certain user-inputted parameters such as an orthogonalisation approach, and non-linearity. Then, Icasso contains certain functionality to subsequently cluster and visualise the components that are obtained through FastICA, and the reliability of these estimates can be determined by analysing the clusters. FastICA primarily uses kurtosis or negentropy to estimate the independent components, as suggested in Sahonero-Alvarez and Calderon (2017). However, given the sensitivity of the kurtosis approach to outliers, the negentropy method is preferred when using FastICA. Section 4 elaborates upon this in further detail.

Alternatively, Icasso can be used with an Infomax algorithm, which is proposed in Bell and Sejnowski (1995). Infomax, as described in Hyvärinen and Oja (2000), computes the independent components by maximising the entropy of a non-linear function. Entropy, in the context of ICA, is a measure of uncertainty; Langlois et al. (2010) suggests that lower entropy corresponds to a greater amount of information relating to a given system. The FastICA and Infomax algorithms, and their respective implementations are described further in Section 4.

Existing research such as Ge et al. (2016) compares the performance of FastICA with *pow3* and *tanh* non-linearities to Infomax on fMRI data; the results indicate that the latter outperforms both specifications of FastICA in estimating components. Moreover, Ge et al. (2016) also notes that the *tanh* non-linearity is better suited to fMRI data, relative to *pow3*. Whilst the proposed research offers a comparison between FastICA and Infomax on a differing fMRI dataset in an attempt to corroborate the aforementioned findings, it also repeats the analysis presented in Himberg et al. (2004). As a result, it can be determined whether a certain FastICA non-linearity is better suited for MEG data, as is seemingly the case for fMRI data.

Additionally, Correa, Adalı, and Calhoun (2007) demonstrates that amongst a group of ICA algorithms (including FastICA), Infomax attains maximally independent components in the estimation of fMRI data, which is the purpose of conducting ICA. Furthermore, Esposito et al. (2002) draws a similar conclusion, suggesting that Infomax tends to outperform fixed-point algorithms such as FastICA in estimation of ICA models. Arya et al. (2003) suggests that the superior performance of Infomax relative to other ICA algorithms results from the fact that the former utilises mutual information, whereas the others such as FastICA use a fourth-order statistic.

Based on the preceding discussion, the following hypothesis is tested:

H1: *Icasso with Infomax will provide more reliable estimates of independent components than Icasso with FastICA for fMRI data.*

By answering this first sub-question, it will be clear whether Icasso with Infomax outperforms Icasso with FastICA. Whichever approach proves to be superior will be compared further to the MST approach with an ICA-EBM algorithm, leading to the consideration of an additional sub-question:

S2: *Does the Minimum Spanning Tree with ICA-EBM provide more reliable estimated independent components than Icasso with either Infomax or FastICA?*

The MST approach for estimating independent components uses the concept of T-maps. These maps are obtained by carrying out one-sample t-tests for each estimated independent component across the total runs of the algorithm. The purpose of these tests is to determine the reliability of the estimated components. The procedure through which the independent components are estimated involves a GAP, the Hungarian algorithm - proposed in Kuhn (1955) - and a MST. The algorithm subsequently identifies a best run as one with the highest correlation between the estimated components and their corresponding T-maps, thereby obtaining components which are consistently estimated across each run.

However, Long et al. (2018) highlights that the current academic knowledge regarding the consistency with which the MST algorithm chooses the best run is insufficient, and additionally finds that the algorithm is computationally intensive. On the other hand, Du et al. (2016) indicates that an MST-based method is able to "*produce components that tend to more accurately classify patients with schizophrenia than those generated using simpler models.*" Both, Levin-Schwartz, Calhoun, and Adalı (2017), as well as Du et al. (2014) support these claims, and suggest that the MST algorithm outperforms Icasso. Accordingly, the following hypothesis is tested:

H2: *The MST algorithm will provide more reliable estimated independent components relative to Icasso with Infomax or FastICA.*

Both hypotheses will be tested; this procedure is described in Section 4.

3 Data

To analyse the performance of Icasso with FastICA, an MEG dataset, obtained from Vigrio et al. (1998), is selected. The data consist of MEG signals that are recorded at 61 locations on a human scalp - the identity of the subject is undisclosed. Artifacts, or undesired interference that can potentially mask the MEG signals, are intentionally created and present in the dataset; for instance, to create myographic artifacts, subjects were told to bite their teeth for as long as 20 seconds. Further information regarding this process is available in the original paper. The recordings of MEG signals are two minutes in duration, and consist of $17,730$ samples. The MEG signals are measured in teslas; Hansen, Kringelbach, and Salmelin (2010) indicates that they typically range from a few femtotesla to a picotesla.

Icasso with FastICA and Icasso with Infomax are then compared, to determine which method provides more reliable independent components. A subsequent comparison with the estimated components that result from the Minimum Spanning Tree algorithm also follows. This analysis is conducted with the use of an fMRI dataset of three subjects, obtained from TReNDS (2012). The data consist of 220 pre-processed fMRI images for each of the three subjects, which are obtained from a visuomotor task. This task involves an event of 55 seconds in which the subjects are exposed to varying visual stimuli four times, resulting in a total of 220 seconds, which corresponds to the number of images.

Through the Minimum Description Length (MDL) principle, which is presented in Rissanen (1978), the number of independent components are estimated from the data. The MDL method is

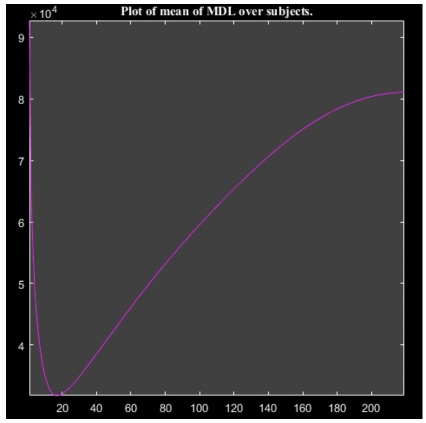


Figure 1: MDL plot

applied with independent and identically distributed sampling, and all three subjects are used to estimate the number of independent components. For each subject, the estimated components are computed, and the mean is taken. This process is visualised in Figure 1; the minimum of the curve corresponds to the number of estimated components. It follows that the mean number of estimated components over all subjects is 17; the standard deviation is 1.732, whilst the minimum and median are both 16. The maximum number of estimated components is found to be 19. Both datasets are super-Gaussian; the consequences of this are considered in Section 4, when presenting the FastICA and Infomax algorithms.

4 Methodology

This section discusses the methods through which the research goals are investigated. The concept of Independent Component Analysis is introduced in a technical framework, including the assumptions of the technique. Then, the FastICA and Infomax algorithms are presented through a discussion on non-Gaussianity and Mutual Information (MI), respectively. Upon establishing the details of these algorithms, the Icasso procedure is discussed, in addition to how it uses the aforementioned algorithms to estimate and measure the reliability of independent components. Lastly, this section explains the Minimum Spanning Tree algorithm and the manner in which it determines the reliability of independent components.

4.1 Independent Component Analysis

As previously discussed, ICA attempts to find a linear representation of data, such that the estimated components are statistically independent. Let $\mathbf{x} = [x_1, x_2, \ldots, x_m]^{\mathbf{T}}$ represent a random vector with m elements. These elements are mixtures (a probabilistic combination of two distributions) of m latent, and independent elements which belong to a random vector $\mathbf{s} = [s_1, s_2, \ldots, s_m]^{\mathbf{T}}$. The vector $\mathbf{s}$ is known as a vector of independent sources. Then, the ICA model is as follows

$$\mathbf{x} = \mathbf{A}\mathbf{s}\,, \tag{1}$$

where $\mathbf{A}$ is an $m \times m$ mixing matrix, which is assumed to be unknown. ICA attempts to estimate $\mathbf{A}$ and $\mathbf{s}$ using only the observed random vector $\mathbf{x}$ by determining a subsequent de-mixing matrix $\mathbf{W}$, which is the (pseudo)inverse of $\mathbf{A}$. This de-mixing matrix should provide an approximation $\mathbf{y}$ of $\mathbf{s}$, such that the ICA model becomes

$$\mathbf{y} = \mathbf{W}\mathbf{x}. \tag{2}$$

There are certain assumptions associated with the ICA model which must be taken into account. Firstly, it is assumed that each of the sources $s_i \in \mathbf{s}$ are independent and non-normally distributed; it follows that the number of sources and mixtures are equal, and the latter are linearly independent. However, the mixing matrix is not necessarily always square: ICA is able to estimate fewer components than the maximal number of possible components, thereby providing a matrix which is non-square. Moreover, it is assumed that the model is free of noise; that is, there are no errors in the dataset. Furthermore, the data are also assumed to have zero-mean. If this is not the case, Hyvärinen and Oja (2000) suggests demeaning the observable variables $x_1, x_2, \ldots, x_m$ by subtracting the sample mean $\bar{x}$, thereby centering the data. Lastly, the ICA model assumes that the distributions of the independent components are unknown, yet non-normally distributed - only one component is allowed to possess a Gaussian distribution. Under these assumptions, which are provided in Langlois et al. (2010), ICA is then achieved by either maximising non-Gaussianity, or minimising mutual information, resulting in the FastICA and Infomax algorithms, respectively.

4.1.1 Non-Gaussianity and FastICA

Maximising non-Gaussianity (or non-normality) is a popular technique in estimating independent components. Hyvarinen (1999) proposes two estimates of non-Gaussanity: kurtosis, and negentropy. These measures serve as objective functions that are to be maximised for ICA estimation. Kurtosis measures whether data are heavy-tailed or light-tailed, using a normal distribution as a reference; negentropy is a measure of the distance from normality for a given distribution. Formally, the negentropy $J(\mathbf{y})$ of a random vector $\mathbf{y}$ is defined as

$$J(\mathbf{y}) = H\left(\mathbf{y}_{gauss}\right) - H(\mathbf{y}) \ , \tag{3}$$

where $H(\cdot)$ is the differential entropy of a random vector $\cdot$ that has density $f(\cdot)$. Moreover, $\mathbf{y}_{gauss}$ is a Gaussian random variable which possesses an identical covariance matrix as $\mathbf{y}$. However, computation of negentropy is often infeasible, thus, Hyvärinen and Oja (2000) utilises an approximation of negentropy to measure non-Gaussianity. This approximation, shown in equation (4), is a contrast function; Comon (1994) defines these as functions which have the ability to obtain independent sources from a linear mixture. Accordingly, the FastICA algorithm, proposed in Hyvärinen and Oja (2000), maximises the contrast function

$$J(y) \propto \left[E\{G(y)\} - E\{G(\nu)\}\right]^2 \ , \tag{4}$$

where $E(\cdot)$ is the mean, y is a variable with zero-mean and unit variance, ν is a standardised Gaussian variable, and G is any non-quadratic function. Hyvärinen and Oja (2000) proposes two commonly used non-quadratic functions and their derivatives that provide robust estimators which are detailed in equations (5) and (6)

$$G_1(u) = \frac{1}{a_1} \log\cosh a_1 u \, , \qquad G_2(u) = -\exp\left\{\frac{-u^2}{2}\right\} \tag{5}$$

$$g_1(u) = \tanh(a_1 u) \ , \qquad g_2(u) = u \exp\left\{\frac{-u^2}{2}\right\}, \tag{6}$$

where $1 \leq a_1 \leq 2$; the constant a_1 is often set equal to 1. Equation (6) shows the *tanh* and *gauss* non-linearities, g_1 and g_2 respectively. These are used in the FastICA algorithm, described below.

The FastICA algorithm, in addition to assuming that the observable variables have zero mean, also assumes that they are whitened. This process entails a linear transformation of $\mathbf{x}$, such that a vector $\tilde{\mathbf{x}}$, with variance equal to one and uncorrelated components, is obtained. Then, the FastICA algorithm is as follows, described in Langlois et al. (2010)

Algorithm 1 FastICA Algorithm for one-unit

1: **Initialise:** $\mathbf{w_i}$, a column-vector of $\mathbf{W}$

2: $\mathbf{w_i}^+ = E\left(g\left(\mathbf{w_i}^\mathbf{T}\mathbf{X}\right)\right)\mathbf{w_i} - E\left(\mathbf{x}G\left(\mathbf{w_i}^\mathbf{T}\mathbf{X}\right)\right)$

3: $\mathbf{w_i} = \frac{\mathbf{w_i}^+}{\|\mathbf{w_i}^+\|}$

4: For $i = 1$, go to step 7, else continue to step 5

5: $\mathbf{w_i}^+ = \mathbf{w_i} - \sum_{\mathbf{j=1}}^{\mathbf{i=i-1}} \mathbf{w_i}^\mathbf{T}\mathbf{w_j}\mathbf{w_j}$

6: $\mathbf{w_i} = \frac{\mathbf{w_i}^+}{\|\mathbf{w_i}^+\|}$

7: If not converged, go back to step 2

8: Else go back to step 1 with $i = i + 1$ until all components are extracted

where $\mathbf{w_i}^+$ is a variable which is temporarily used to calculate $\mathbf{w_i}$, and $g(\cdot)$ is the derivative of G. Thus, this procedure outlines the manner in which the FastICA algorithm computes independent components.

4.1.2 Mutual Information and Infomax

Batina et al. (2011) defines MI as "*a general measure of the dependence between two random variables.*" More specifically, the MI, denoted as $I(\cdot)$ for a pair of random variables X and Y, is as follows

$$I(X;Y) = H[X,Y] - H\left[X|Y\right] - H\left[Y|X\right] , \tag{7}$$

where $H[X,Y]$ is the joint entropy of X and Y - the uncertainty which is associated with these variables. The conditional entropy, $H\left[X|Y\right]$, measures the uncertainty of X once Y is known. The last term in equation (7) can also be interpreted in a similar manner. Thus, this equation illustrates the manner in which the minimisation of mutual information between variables provides maximally independent components. Based on the minimisation of entropy, the Infomax algorithm presented in Amari, Cichocki, and Yang (1995) computes $\mathbf{W}$ in the following manner

Algorithm 2 Infomax Algorithm

1: **Initialise:** $\mathbf{W}(0)$, a random matrix

2: $\mathbf{W}(t+1) = \mathbf{W}(t) + \eta(t)\left(\mathbf{I} - f(\mathbf{y})\mathbf{y}^\mathbf{T}\right)\mathbf{W}(\mathbf{t})$

3: Repeat step 2 if not converged.

where t is a given step, $\eta(t)$ is a function that provides step size to update $\mathbf{W}$, $\mathbf{I}$ is an $m \times m$ identity matrix, and $f(\mathbf{y})$ is a non-linear function which depends on whether the distribution is super or sub-Gaussian. Note that $\mathbf{y}$ is defined in equation (2). Additionally, Shah, Arora, Robila, and Varshney (2002) defines super-Gaussian and sub-Gaussian variables as those with positive and negative kurtosis, respectively. Given that it is stated in Section 3 that the datasets are super-Gaussian, $f(\mathbf{y})$ is defined as

$$f(\mathbf{y}) = \tanh \mathbf{y}. \tag{8}$$

Hyvärinen and Oja (2000) demonstrates that computing $\mathbf{W}$ for which mutual information is minimised is approximately similar to maximising negentropy, albeit with the constraint that the estimates are uncorrelated. This assumption is used to make the computations more feasible.

4.2 Icasso

Sections 4.1.1 and 4.1.2 have thus far demonstrated the manner in which independent components are obtained. However, as previously discussed, it is also essential to estimate the reliability of these independent components; for this purpose, the Icasso software package is utilised. This package contains various functionality that assists in determining the statistical reliability of the estimates.

The Icasso algorithm works as follows: firstly, the parameters under which independent components are estimated are selected using either the FastICA or Infomax algorithms. Then, the selected ICA algorithm is run a certain number of times; in each run, the data are either bootstrapped, or the starting point of the optimisation is changed (random initial conditions). Consequently, estimated components are obtained. Next, using an agglomerative-clustering technique with an average-linkage criterion, the estimated components are clustered based on their mutual correlation coefficients r_{ij}. A two-dimensional plot of the clusters is produced by the Icasso software package, which offers insights regarding the reliability of the components: according to Himberg et al. (2004), reliability is represented through tight clusters, as these contain approximately similar estimates from numerous ICA runs.

It is pertinent to briefly elaborate upon some concepts which are used in Icasso - the first one being bootstrapping. Described in Horowitz (2001), bootstrapping is a statistical, computer-based resampling technique whereby the distribution of an estimator or test-statistic is computed. In the context of this research, the data sample is altered randomly through simulation, and a chosen ICA algorithm is run multiple times with the differing bootstrapped samples. This process enables a comparison between the original estimate and the spread of obtained estimates, thereby providing a way of analysing the reliability of estimates. However, the reliability can also be assessed through random initial conditions, or by using a combination of both, random initial conditions and bootstrapping. The manner in which these relate to reliability is discussed in Himberg et al. (2004).

Furthermore, agglomerative-clustering is a common type of hierarchical clustering, which groups data into clusters based on their similarities; the implementation of Icasso relies on this clustering method with a group-average linkage agglomeration strategy as opposed to single-link or complete-link strategies. The preference for group-average linkage stems from the fact that Himberg et al. (2004) reports that it is the strategy which is least sensitive to noise, and provides the most consistent performance.

The following sections will explain how Icasso is used in conjunction with the FastICA and Infomax algorithms for MEG and fMRI data.

4.2.1 MEG Data

To analyse the performance of Icasso on the MEG dataset described in Section 3, the FastICA algorithm is used with the following configurations, which are detailed in Himberg et al. (2004): (I) random initial conditions with third power (*pow3)* as non-linearity; (II) random initial conditions with *tanh* as non-linearity; and (III) random initial conditions and bootstrapping with *pow3* as non-linearity.

In order to reduce noise and prevent over-fitting, Principal Component Analysis (PCA) is firstly applied to the data to reduce their dimension from 122 to 20. The number of randomisations M is set equal to 15, and a symmetric decorrelation approach is selected, which ensures that the independent components are estimated in parallel. The results of the three configurations are evaluated on the basis of their visualised clusters, as well as a conservative cluster quality index I_q which is introduced in Himberg et al. (2004). The visualisations are obtained through the use of Curvilinear Component Analysis (CCA), which provides a representation of multidimensional data. This technique is introduced in Demartines and Hérault (1997), and functions as a neural network which is able to map any new point from an input space to an output space, or vice versa; it is able to do this in an uninterrupted fashion, thereby providing the visualisations of the clusters.

The visualised clusters can be interpreted as follows: tight clusters, such as a single point, represent independent components which are estimated reliably. Clusters with points that are more dispersed are wider, and thus represent less reliable independent components. The visualisation is supplemented by the I_q index, to ensure that any bias or subjectivity from purely visualising the clusters is eliminated. This index computes cluster quality "*as the difference between the average* *intracluster similarities and average intercluster similarities.*" Prior to providing a mathematical definition of I_q, certain notation is introduced in Table 1.

Table 1: Notation for the cluster quality index I_q.

Symbol	**Explanation**
C	Set of indices of all estimated components
C_m	Set of indices belonging to cluster m
$\|C_m\|$	Size of cluster m
C_{-m}	Set of indices that do not belong to cluster m
$\sigma_{ij} = \|r_{ij}\|$	Similarity between estimated independent components

Then, $I_q(C_m)$, the quality index of the set of indices belonging to cluster m, is defined as follows

$$I_q(C_m) = \frac{1}{|C_m|^2} \sum_{i,j \in C_m} \sigma_{i,j} - \frac{1}{|C_m||C_{-m}|} \sum_{i \in C_m} \sum_{j \in C_{-m}} \sigma_{i,j}. \tag{9}$$

I_q is equal to 1 for a cluster which is estimated reliably, and decreases as C_m becomes increasingly dispersed. The corresponding analysis is conducted using the packages Icasso (2010) and FastICA (2013); these are readily available for use with the MATLAB software. The following analysis, involving Icasso with FastICA and Infomax on fMRI data, and the Minimum Spanning Tree algorithm, will utilise GIFT (2017), which is implemented in MATLAB. This toolbox implements ICA for fMRI data.

4.2.2 fMRI Data

To compare the performance of Icasso with FastICA to Icasso with Infomax, the former is computed with three non-linearities: *pow3*, *tanh*, and *gauss*. The obtained clusters from each method are compared based on a visualisation of the clusters and the I_q index. Moreover, another metric obtained from Himberg et al. (2004), the $R-$index, is used to determine the consistency of the estimates. Denoted by I_R, this index seeks for tight, well-separated clusters in the following manner

$$I_R = \frac{1}{L} \sum_{m=1}^{L} \frac{S_m^{in}}{S_m^{ex}}, \tag{10}$$

where L is the number of clusters, and

$$S_m^{in} = \frac{1}{|C_m|^2} \sum_{i,j \in C_m} d_{ij}, \qquad S_m^{ex} = \min_{m' \neq m} \frac{1}{|C_m||C'_m|} \sum_{i \in C_m} \sum_{j \in C'_m} d_{ij},$$

where $d_{ij} = 1 - \sigma_{ij}$. In addition, the minimum of the $R-$index helps determine an appropriate partition of clusters. For instance, in each configuration of Icasso which is used with the fMRI data, it holds from Section 3 that $L = 17$. To demonstrate how the $R-$index can be utilised, consider the following stylised example: upon obtaining the results, the $R-$index may contain a minimum at $L = 12$; then, it can be investigated how the results of the same configuration change when using the latter value of L.

The following Icasso configurations are tested with FastICA: (I) random initial conditions with each non-linearity (*pow3, tanh, and gauss*); (II) random initial conditions and bootstrapping with *pow3* as non-linearity. The FastICA option will be computed with a symmetrical approach. An Icasso with Infomax configuration is also considered. In total, five configurations are tested and subsequently compared.

For Icasso with Infomax, the default options of the GIFT toolbox are used; these are detailed in Appendix B. Ten Icasso runs are conducted with each algorithm; the GIFT toolbox also requires specifications for a minimum and maximum number of runs - these are set at eight and ten respectively, as recommended by the toolbox. The GIFT toolbox suggests a number of minimum runs based on the following formula

$$\mathit{MinimumRuns} = \lceil 0.8 \times \mathit{NumberOfRuns} \rceil \,, \tag{11}$$

where 10 is the number of runs, and $\lceil \cdot \rceil$ is the ceiling function.

4.3 Minimum Spanning Tree Algorithm

Like Icasso, the MST approach attempts to gauge the reliability of estimated independent components. Whereas Icasso uses the FastICA or Infomax algorithms to estimate the independent components, MST uses an Entropy Bound Minimisation (EBM) algorithm for ICA, introduced in Li and Adali (2010). ICA-EBM works as follows: numerical methods are used to bound the entropy of estimates, which is then minimised. A line-search method is used to obtain better convergence. ICA-EBM is an algorithm which does not use any parameters, and it employs four non-linearities in order to calculate the entropy bound; these are highlighted in equation (12) below

$$G\left(u\right) = u^4 \,, \qquad G\left(u\right) = \frac{|u|}{1+|u|} \,, \qquad G\left(u\right) = \frac{|u|}{10+|u|} \,, \qquad G\left(u\right) = \frac{u}{1+u^2}. \tag{12}$$

Like the Infomax algorithm, ICA-EBM minimises mutual information to estimate the de-mixing matrix $\mathbf{W}$, whilst using the same line-search algorithm as FastICA, detailed in Algorithm 1. This line-search iterates over different row vectors of $\mathbf{W}$ until convergence is achieved by means of the following criterion

$$1 - \max\left(\left|\left[diag\left(\mathbf{W}^{new}\mathbf{W}^{T}\right)\right]\right|\right) \leq \epsilon\,, \tag{13}$$

where $\epsilon = 0.0001$, and $diag\left(\cdot\right)$ is a matrix which only contains the diagonal elements of its argument on its diagonal, and zeros otherwise. Al-Ali, Chandran, and Naik (2021) describes the ICA-EBM algorithm, stating that the updated de-mixing matrix $\mathbf{W}^{new}$ is obtained in the following manner

$$\mathbf{W}^{new} = \left(\mathbf{W}\mathbf{W}^{T}\right)^{-\frac{1}{2}}\mathbf{W}\,, \tag{14}$$

by using symmetrical decorrelation to ensure that the matrix remains orthogonal. As previously discussed, the MST approach for ICA uses $T-$maps to select components. The estimates which are provided by Icasso are compared to the components obtained through the MST on the basis of the correlation between the components and their $T-$maps.

An additional measure is considered to compare the estimated independent components that are obtained through Icasso and MST - the modified RV-coefficient, which functions as a measure of matrix correlations for high-dimensional data. Introduced in Smilde, Kiers, Bijlsma, Rubingh, and Van Erk (2009), the modified RV-coefficient is calculated as follows (for matrices $\mathbf{X}$ and $\mathbf{Y}$)

$$RV_2\left(\mathbf{X},\mathbf{Y}\right) = \frac{Vec\left(\widetilde{\mathbf{X}\mathbf{X}'}\right)' Vec\left(\widetilde{\mathbf{Y}\mathbf{Y}'}\right)}{\sqrt{Vec\left(\widetilde{\mathbf{X}\mathbf{X}'}\right)' Vec\left(\widetilde{\mathbf{X}\mathbf{X}'}\right) \times Vec\left(\widetilde{\mathbf{Y}\mathbf{Y}'}\right)' Vec\left(\widetilde{\mathbf{Y}\mathbf{Y}'}\right)}}\,, \tag{15}$$

where $Vec\left(\cdot\right)$ is a vectorised version of its argument, and $\widetilde{\mathbf{X}\mathbf{X}'} = \mathbf{X}\mathbf{X}' - diag\left(\mathbf{X}\mathbf{X}'\right)$. It follows that $\widetilde{\mathbf{Y}\mathbf{Y}'}$ can be computed in a similar manner. The modified RV-coefficient ranges from (and including) -1 to 1, and $RV_2\left(\mathbf{X},\mathbf{Y}\right) = -1$, for instance, can be interpreted equivalently to that of a negative Pearson correlation coefficient. In the context of this research, RV_2 will be used to compute the correlation between the matrices of estimated independent components that result from Icasso, and the MST approach. Doing so provides some insight regarding the extent to which the estimated independent components from each method are similar.

To compare Icasso and MST, the following experimental design is adopted: ten runs of MST with ICA-EBM are performed on fMRI data; likewise to the previous analysis, the minimum and maximum runs are set at eight and ten, respectively based on equation (11). Then, the estimated components which are obtained through MST with ICA-EBM are compared to Icasso with either FastICA or Infomax through various measures. Thus, it can be determined whether one method provides more reliable components than the other.

5 Results

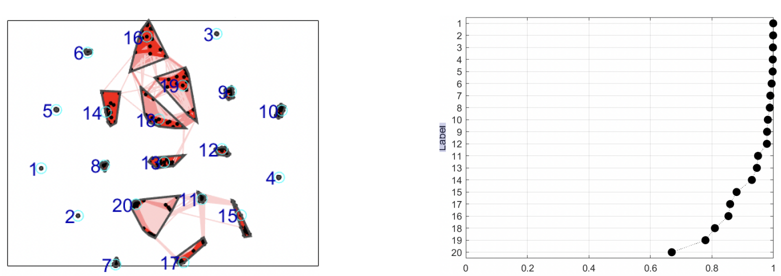
5.1 Icasso with MEG Data

This section discusses the results of running the Icasso software package with a FastICA algorithm on the MEG dataset which is described in Section 3. The outputs of three FastICA configurations are compared, to analyse which method provides the most reliable estimates of independent components. In doing so, this section functions as a replication of the research that is conducted in Himberg et al. (2004).

The experimental procedure is carried out using MATLAB R2021a with an Intel(R) Core(TM) $i5-9500$ CPU at 3.00 GHz using 16.0 GB RAM, with Windows 10 Enterprise. When running the analysis with the GIFT toolbox, the option to maximise performance is chosen each time.

5.1.1 Configuration I

The results of one Icasso run using the FastICA algorithm with random initial conditions and *pow3* non-linearity are shown below in Figure 2, which shows how the estimated independent components are clustered (Figure 2a), in addition to the quality of said clusters (Figure 2b). The latter shows that clusters $1-10$ (and excluding cluster 9) are valued at approximately 1, indicating that these clusters are reliably estimated. The visualised clusters corroborate these findings, as the clusters in question are represented tightly, often as single points. The remaining clusters eventually decline in their quality, and are characterised by wider clusters. The range of the I_q index is approximately 0.330; that is, the difference in the value between the quality of the best and worst cluster (0.999 and 0.669 respectively). These results can also be seen in Appendix A, which lists the values of the I_q index for each of the three configurations.



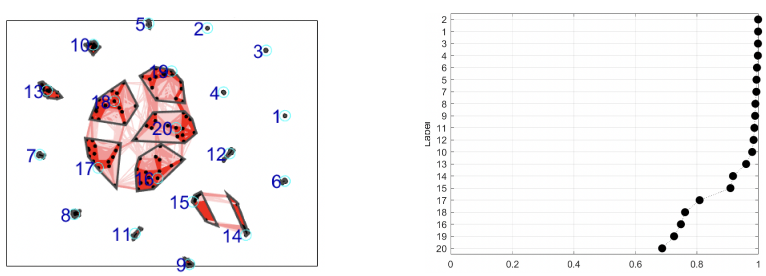
(a) Clusters **(b)** I_q index

Figure 2: Visualised results of Configuration I from one Icasso run

5.1.2 Configuration II

Figure 3 illustrates the results of one Icasso run with the second configuration; that is, with random initial conditions, and *tanh* as the chosen non-linearity. Here, the first twelve clusters are close to 1 in terms of the I_q index, whereas in Figure 2b, there is a slight kink between clusters 12 and 11, suggesting a decrease in the quality of the clusters. The range of the index is larger in the first configuration relative to configuration II, which implies that the latter provides more reliable independent components. Specifically, the range of the I_q index here is 0.312 (the minimum is 0.687; the maximum is 0.999). However, Figure 3b also shows lower I_q values for its last five estimates than Figure 2b. Thus, while the second configuration provides more reliable estimates for the first few components, this does not follow for the last few components - in this case, configuration I proves to be superior.

The clusters in Figure 3a can be interpreted in a similar manner as before, with tight clusters (and single points) representing reliable estimates of independent components. It can be seen in Figure 3b that cluster 2 ranks above cluster 1, whilst the opposite holds for configuration I; Himberg et al. (2004) notes that this is often possible, especially in repeated runs of Icasso with varying settings.

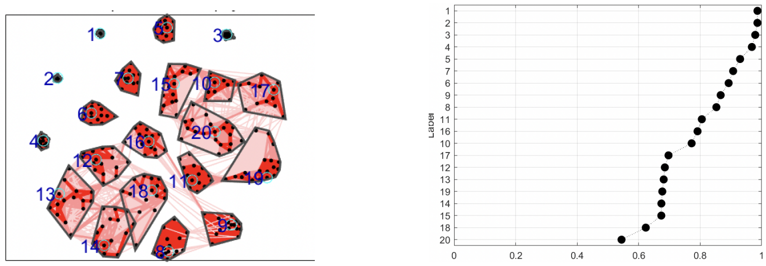


(a) Clusters **(b)** I_q index

Figure 3: Visualised results of Configuration II from one Icasso run

5.1.3 Configuration III

Configuration III, which utilises a combination of random initial conditions and bootstrapping with *pow*3 non-linearity as FastICA parameters, is run and the subsequent results are given in Figure 4. From both, Figures 4a and 4b, it is evident that the clusters are wider on average relative to the other two configurations. Moreover, whereas configurations I and II yielded at least ten clusters that were valued approximately 1 by the I_q index, configuration III provides only the first two clusters with a value close to 1. It is important to note however, that the I_q values of these first two clusters are lower than those belonging to the other configurations. For this particular configuration, the minimum value of the I_q index is 0.544, and the maximum is 0.986, thus the range of the I_q index is nearly 0.442 - a relatively large increase when compared to the previous Icasso specifications. The findings suggest that the particular FastICA parameters under consideration may not be suitable for MEG data.



(a) Clusters **(b)** I_q index

Figure 4: Visualised results of Configuration III from one Icasso run

Certain findings that are obtained from these three configurations match those of Himberg et al. (2004). For instance, in each case, the top four estimates are always ranked from $1 - 4$. Furthermore, clusters 5 and 6 are more reliably estimated under the conditions of configuration I than configurations II or III. Based on the analysis conducted in this section, an overall evaluation suggests that configuration I provides the most reliable estimates of independent components for MEG data.

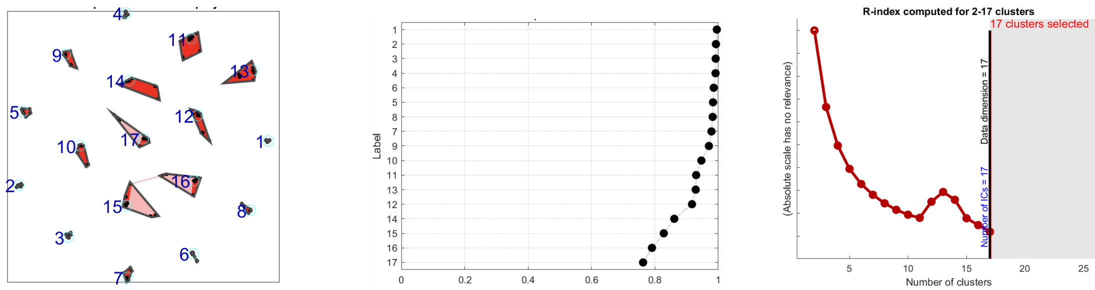
5.2 Icasso with fMRI Data

This section presents the findings when using Icasso with various algorithms and configurations on fMRI data. Firstly, the results of Icasso with FastICA and three non-linearities (*pow3*, *tanh*, and *gauss*) are discussed in that order. These results are then compared to Icasso with Infomax on the same fMRI data. Doing so will subsequently answer the first hypothesis and the relevant sub-question.

5.2.1 FastICA

Figure 5 illustrates the independent components that are obtained with the following Icasso specification: random initial conditions and *pow3* non-linearity. The configuration demonstrates reasonable performance in terms of its I_q index, and the visualised clusters are not too wide. Hence, the estimated independent components are relatively reliable, with the exception of the components labelled $14 - 17$ in Figure 5b. These components experience a sharper decline in cluster quality when compared to the other independent components.

The $R-$index in Figure 5c possesses a minimum at $L = 17$, which indicates that it is the most appropriate partition of clusters for the relevant specification; thus, it is not necessarily interesting to consider how the results would be affected for different values of L.



(a) Clusters

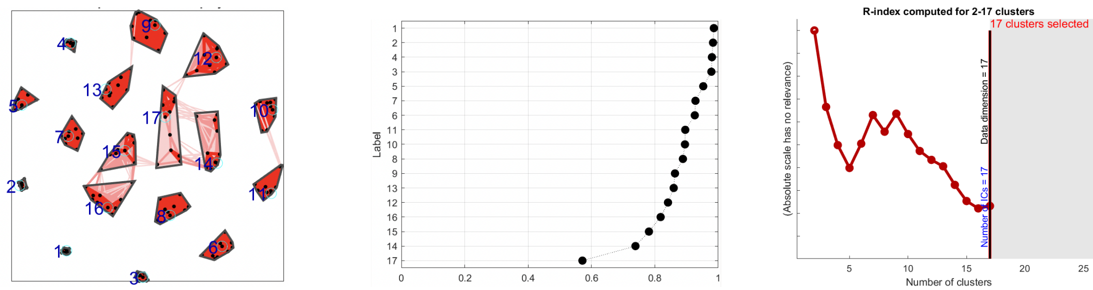
(b) I_q index

(c) $R-$index

Figure 5: Icasso results on fMRI data using random initial conditions and *pow3* non-linearity

On the other hand, Figure 6 shows the results of running Icasso with random initial conditions and *tanh* as the non-linearity of choice. Relative to the previous configuration, the estimated independent components that are obtained here are worse in quality, and therefore, lack reliability. It is interesting to note that for the analysis pertaining to MEG data in Section 5.1, the opposite result is noticed. That is, Icasso with *tanh* performed at least as well as Icasso with *pow3*. Nevertheless, Figures 6a and 6b show that only a small number of components, namely those from $1 - 10$ are reliably estimated, to some extent. Moreover, these findings contradict those presented in Ge et al. (2016), as well as the discussion in Section 2.

The corresponding $R-$index - shown in Figure 6c - has a minimum at $L = 16$, and so, Appendix C.1 presents the results of conducting Icasso with *tanh* and $L = 16$, accompanied by a brief discussion. The reason this analysis is conducted in further detail is to test if the configuration can outperform that of Icasso with *pow3*, as should be the case according to certain literature.

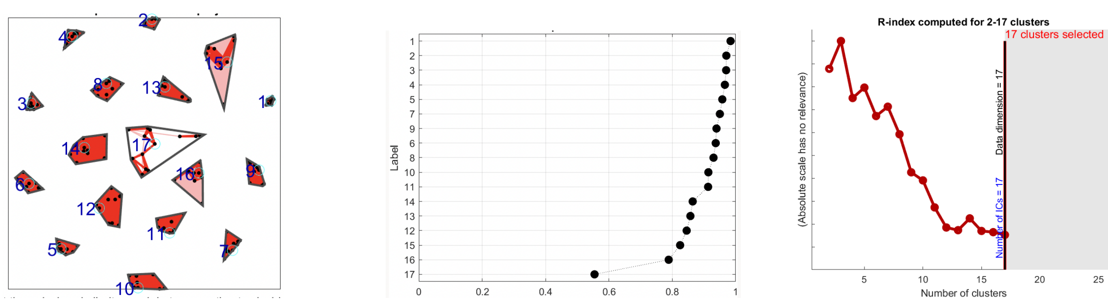


(a) Clusters **(b)** I_q index **(c)** $R-$index

Figure 6: Icasso results on fMRI data using random initial conditions and *tanh* non-linearity

When using Icasso with the *gauss* non-linearity, the estimated independent components are more reliable than Icasso with *tanh*, but still lacking in quality when compared to *pow3*, as shown by Figure 7. A greater number of components are reliably estimated, but there is a sharp decline for component 17. Furthermore, the first two configurations of this analysis provide more reliable estimates of the first four components than Icasso with *gauss*. A common pattern between each of the three configurations is that all top four estimates are always ranked $1-4$. It follows that while Icasso with *gauss* provides relatively more reliable components, Icasso with *tanh* is better at reliably estimating the first few components.

Likewise to Icasso with *pow3*, the $R-$index in Figure 7c has a minimum at $L = 17$, and so, any other case is not considered for further analysis.

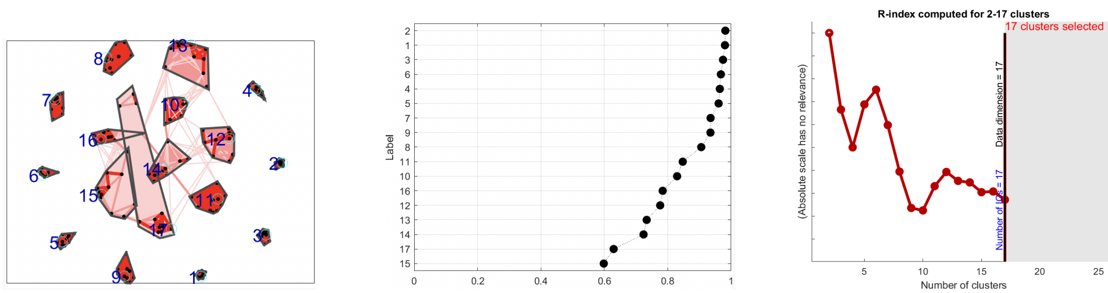


(a) Clusters **(b)** I_q index **(c)** $R-$index

Figure 7: Icasso results on fMRI data using random initial conditions and *gauss* non-linearity

Lastly, Figure 8 below illustrates the independent components of the following Icasso configuration: random initial conditions and bootstrapping with *pow3* as the non-linearity. From Figure 8a, it is visible that the obtained clusters are quite wide, and Figure 8b shows a pronounced and early decline in the quality of the clusters. The last seven components are fairly unreliable, as their I_q index value lies below 0.8. Moreover, in contrast to the other three Icasso specifications, component 4 does not belong to the top four estimates. Hence, this observation serves as further evidence of the fact that this configuration does not necessarily provide reliable estimates of the first few components either. As is the case with the Icasso results on MEG data from Section 5.1, the Icasso configuration with random initial conditions and bootstrapping provides the least reliable estimates of independent components.

For this configuration, Figure 8c shows that the $R-$index attains a minimum at $L = 10$. In addition, there is a noticeable decline in the cluster quality when moving from cluster 10 to cluster 16 in Figure 8b; accordingly, Appendix C.2 briefly considers the change in results when a fewer number of clusters are used, relative to the original specification of $L = 17$.



(a) Clusters **(b)** I_q index **(c)** $R-$index

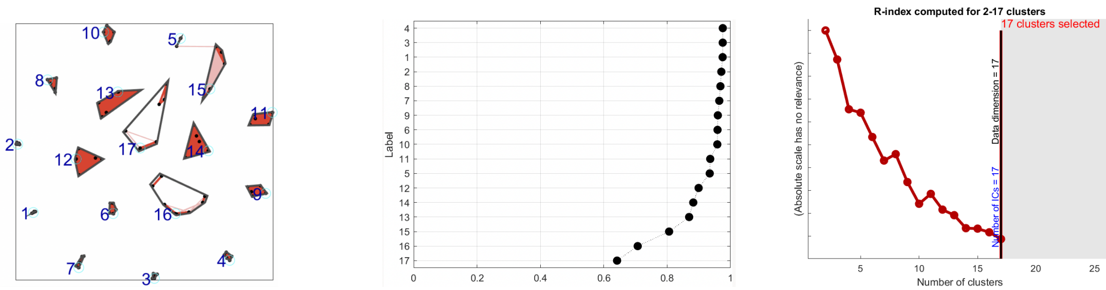
Figure 8: Icasso results on fMRI data using random initial conditions and bootstrapping with *pow3*

It follows that Icasso with random initial conditions and *pow3* as the non-linearity (presented in Figure 5) is the best performing specification from those which have been considered. Therefore, this approach will be compared further to Icasso with Infomax for fMRI data in the following section.

5.2.2 Infomax

Here, the estimated independent components of fMRI data yielded by Icasso with Infomax are considered. Like Icasso with the different FastICA configurations, ten runs are conducted, with the results presented in Figure 9 and its sub-figures. Like most of the Icasso with FastICA results, the top four estimates are ranked $1 - 4$, as shown by Figure 9b. However, the width of the clusters in Figure 9a indicates that while a few components are reliably estimated, when compared to those in Figure 5a, the latter has tighter clusters which suggests that it provides better reliability of estimates.

The $R-$index in Figure 9c shows that the optimal number of clusters is indeed 17, as is specified by the configuration. Thus, Icasso with Infomax is not tested for differing values of L.



(a) Clusters   **(b)** I_q index   **(c)** $R-$index

Figure 9: Icasso results on fMRI data using Infomax

This leads to a rejection of the first hypothesis; Icasso with Infomax does not provide more reliable estimates of independent components than Icasso with FastICA for these fMRI data. Whilst this conclusion is unexpected, Brunner, Naeem, Leeb, Graimann, and Pfurtscheller (2007) notes that such a result is possible, especially when PCA is conducted prior to ICA, which is indeed the case with the analysis presented in this section.

Accordingly, the Icasso with *pow3* configuration is compared to the MST with ICA-EBM approach, to determine which method provides more reliable estimates of independent components for fMRI data. Consequently, an answer to the second hypothesis will be obtained.

5.3 Minimum Spanning Tree Algorithm with fMRI Data

Ten runs of the ICA-EBM algorithm are performed, using the MST method as the choice of stability analysis rather than Icasso. As explained in Section 2, this approach identifies a best run according to the correlation between the estimated independent components and their $T-$maps. Accordingly, the ninth MST run is determined to be the best one; Figure 10 shows a box-plot of the relationship between the independent components and their $T-$maps across ten runs. The horizontal line in each box represents the median, whilst the cross (x) in the box shows the average. Explicitly, it is the average value of the correlation between the independent components and their clusters for each run. The corresponding value for the ninth run is 0.870, which is the highest across all runs. The numerical values of each run are available in Appendix D. The remainder of the box-plot can be interpreted in the following manner: the top and bottom edges of the box are the upper and lower quartiles respectively; the whiskers show the maximum and minimum values; lastly, individual points are outliers.

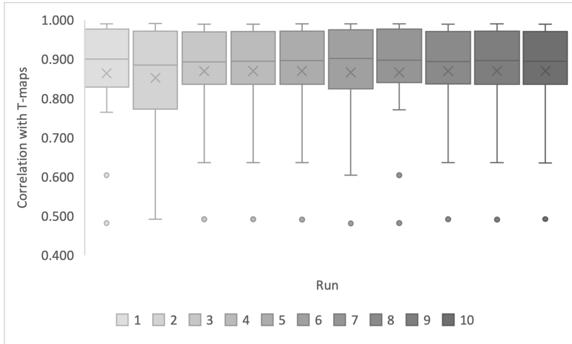


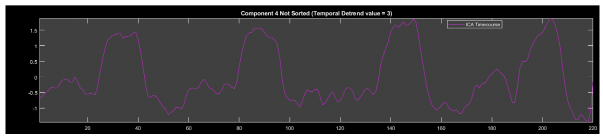
Figure 10: Correlations between estimated independent components and their $T-$maps across ten MST ICA-EBM runs

Based on the fact that Appendix D shows that component 4 has the highest correlation on average (0.990) amongst all the estimated components, it is inspected further in Figure 11; the sub-figures 11a and 11b show the time series and scans of component 4, respectively. It can be noted that in Figure 10, the correlations of component 4 are the maximum values - the uppermost whisker in each run. To compare the performance of MST to that of Icasso with *pow3*, Figure 11c shows the scans of component 4 that are obtained from using the latter method.

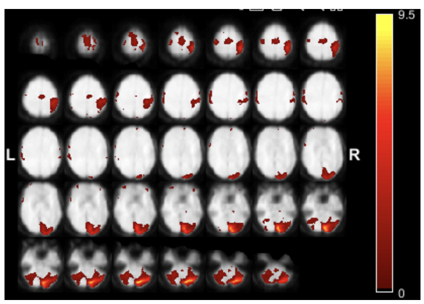
Computation of the RV_2 statistic provides a value of -0.0447, which indicates that both methods provide vastly different estimates of the independent components. This can be observed by the fact that from the best MST run, the first five components which have the highest correlation with their $T-$maps are components $4, 7, 6, 10$, and 12 in that order. However, based on Figure 5b, the five best estimated components from Icasso with *pow3* are $1, 2, 3, 4$, and 6 - both methods only have two components in common, which serves to illustrate the disparity in the estimated independent components.

Figure 11a displays a time-series plot of component 4; the sequential peaks and eventual decline of the time-series are associated with the introduction of a visual stimulus, which generates activity in the brain and is therefore responsible for the shape of the time-series. When comparing the scans of component 4, it can be seen that a greater degree of activation in the visual cortex is noticed through MST than Icasso with *pow3*. For instance, the scale of the heat-maps in Figures 11b and 11c ranges from 0 to 9.5 and 7.5 respectively, which indicates that the MST approach is able to identify greater levels of activity within the brain. However, Icasso with *pow3* measures a greater amount of activity in the visual cortex of the brain, as compared to the MST approach; this is particularly evident when looking at rows three and four of the respective sub-figures.

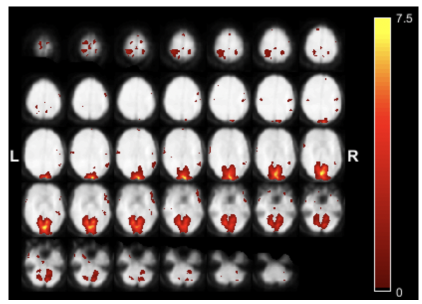
Whereas the GIFT toolbox does not possess functionality to measure the computation times of each method, the size of the output for each method can provide some insight about their computational feasibility. The file which is created by the GIFT toolbox which stores the results of the MST approach to estimate the independent components is 35.2 megabytes, whereas the Icasso with *pow3* file is 8.5 megabytes. These values suggest that Icasso is less (computationally) demanding in the manner through which it obtains estimates of independent components.



(a) Time Series*



(b) Scans*



(c) Scans from Icasso with *pow3*

Figure 11: Visualised results of component 4 from ten MST ICA-EBM runs*

Accordingly, it is not straightforward or trivial to declare whether one method outperforms the other. These results are able to draw attention to the differences in approach to ICA estimation, as well as the advantages and disadvantages of each method. The second sub-question attempts to determine whether MST provides more reliable estimates of independent components than Icasso - based on the obtained results, there are trade-offs associated with each approach, yet the performance and computational feasibility of Icasso (especially with *pow3*) makes it a viable option. The research in this paper goes to show that one method is not definitively, or strikingly better than the other when comparing MST and Icasso.

6 Conclusion

This paper investigates the following research question: *can the statistical reliability of the estimated independent components of ICA for neuroimaging time series be improved by using Icasso with FastICA, Icasso with Infomax, or a Minimum Spanning Tree algorithm with ICA-EBM?*

By conducting this research, the results show that when MEG data is concerned, Icasso with the FastICA configuration of random initial conditions and *pow3* non-linearity provides the most reliable estimates of independent components. For the fMRI data of choice in this research, Icasso with FastICA (same configuration as above) once again proves its strengths and outperforms Icasso with Infomax, by yielding more reliable estimates of independent components. Lastly, for the same fMRI data, Icasso with FastICA and MST with ICA-EBM both demonstrate reasonable performance in the estimation of independent components, with the former being more computationally efficient.

Taking into account the performance of Icasso with FastICA on both datasets that are utilised in this research, it stands out as the method of choice for independent component analysis, relative to Icasso with Infomax or MST with ICA-EBM. Thus, this answers the proposed research question, with several implications. For the academic field, this research provides more information as to how these methods perform for fMRI data, and discusses the suitability of different parameters and algorithms depending on the type of dataset. This additionally functions as an applied contribution of this body of work: there is more information as to how these methods behave for different data, and can ultimately assist in choosing the right algorithm when conducting independent component analysis. The consequent improvement in the reliability of estimates of neuroimaging data is likely to have medical and statistical benefits for all the involved stakeholders.

Nevertheless, directions for future research can consider more extensive comparisons of these methods, to increase the degree of knowledge within the industry. One way of performing better comparisons is through the use of simulated MEG and fMRI datasets, as opposed to subject data. An advantage of using simulated data is that it allows for the creation of data that exhibit certain properties; accordingly, it can be examined which ICA algorithms are better suited to datasets with said properties. In addition, the possibilities in terms of characteristics and the size of simulated data are vast, thus, these results will possess a greater degree of validity and generalisability, which the research conducted in this paper lacks. This is due to the fact that the two datasets which are employed in this paper are real MEG and fMRI data, and their nature cannot be adjusted to test the ICA algorithms to their full extent.

References

Al-Ali, A. K. H., Chandran, V., & Naik, G. R. (2021). Enhanced forensic speaker verification performance using the ICA-EBM algorithm under noisy and reverberant environments. *Evolutionary Intelligence*, *14*(4), 1475–1494.

Amari, S.-i., Cichocki, A., & Yang, H. (1995). A new learning algorithm for blind signal separation. *Advances in neural information processing systems*, *8*.

Arya, R., Calhoun, V., Roys, S., Adali, T., Greenspan, J., & Gullapalli, R. (2003). Comparative study of several multivariate fMRI processing methods: PCA, Factor Analysis, Infomax, FASTICA, MELODIC. *Procedings of the International Society for Magnetic Resonance in Medicine*, 1–1.

Batina, L., Gierlichs, B., Prouff, E., Rivain, M., Standaert, F.-X., & Veyrat-Charvillon, N. (2011). Mutual information analysis: a comprehensive study. *Journal of Cryptology*, *24*(2), 269–291.

Bell, A. J., & Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, *7*(6), 1129–1159.

Brunner, C., Naeem, M., Leeb, R., Graimann, B., & Pfurtscheller, G. (2007). Spatial filtering and selection of optimized components in four class motor imagery eeg data using independent components analysis. *Pattern recognition letters*, *28*(8), 957–964.

Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, *36*(3), 287–314.

Correa, N., Adalı, T., & Calhoun, V. D. (2007). Performance of blind source separation algorithms for fMRI analysis using a group ICA method. *Magnetic Resonance Imaging*, *25*(5), 684–694.

Demartines, P., & Hérault, J. (1997). Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on neural networks*, *8*(1), 148–154.

Du, W., Levin-Schwartz, Y., Fu, G.-S., Ma, S., Calhoun, V. D., & Adalı, T. (2016). The role of diversity in complex ICA algorithms for fMRI analysis. *Journal of neuroscience methods*, *264*, 129–135.

Du, W., Ma, S., Fu, G.-S., Calhoun, V. D., & Adalı, T. (2014). A novel approach for assessing reliability of ICA for FMRI analysis. In *2014 IEEE international conference on acoustics, speech and signal processing (Icassp)* (pp. 2084–2088).

Esposito, F., Formisano, E., Seifritz, E., Goebel, R., Morrone, R., Tedeschi, G., & Di Salle, F. (2002). Spatial independent component analysis of functional mri time-series: To what extent do results depend on the algorithm used? *Human brain mapping*, *16*(3), 146–157.

The FastICA package for MATLAB. (2013). Retrieved from `http://research.ics.aalto.fi/ica/fastica/` (The MathWorks, Natick, MA, USA)

fMRI Data - Example fMRI data is from a visuomotor paradigm. (2012). Retrieved from `https://trendscenter.org/trends/software/gift/software_download_links.html`

Ge, R., Wang, Y., Zhang, J., Yao, L., Zhang, H., & Long, Z. (2016). Improved FastICA algorithm in fMRI data analysis using the sparsity property of the sources. *Journal of neuroscience methods*, *263*, 103–114.

Group ICA of fMRI Toolbox (GIFT). (2017). Retrieved from `https://trendscenter.org/software/gift/` (The MathWorks, Natick, MA, USA)

Hansen, P., Kringelbach, M., & Salmelin, R. (2010). *MEG: an introduction to methods.* Oxford university press.

Himberg, J., Hyvärinen, A., & Esposito, F. (2004). Validating the independent components of neuroimaging time series via clustering and visualization. *Neuroimage*, *22*(3), 1214–1222.

Horowitz, J. L. (2001). The bootstrap. In *Handbook of econometrics* (Vol. 5, pp. 3159–3228). Elsevier.

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

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, *13*(4-5), 411–430.

Icasso: software for investigating the reliability of ICA estimates by clustering and visualization. (2010). Retrieved from `http://research.ics.aalto.fi/ica/icasso/` (The MathWorks, Natick, MA, USA)

Kachenoura, A., Albera, L., Senhadji, L., & Comon, P. (2007). ICA: a potential tool for BCI systems. *IEEE Signal Processing Magazine*, *25*(1), 57–68.

Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval research logistics quarterly*, *2*(1-2), 83–97.

Langlois, D., Chartier, S., & Gosselin, D. (2010). An introduction to independent component analysis: InfoMax and FastICA algorithms. *Tutorials in Quantitative Methods for Psychology*, *6*(1), 31–38.

Levin-Schwartz, Y., Calhoun, V. D., & Adalı, T. (2017). Quantifying the interaction and contribution of multiple datasets in fusion: Application to the detection of schizophrenia. *IEEE transactions on medical imaging*, *36*(7), 1385–1395.

Li, X.-L., & Adali, T. (2010). Independent component analysis by entropy bound minimization. *IEEE Transactions on Signal Processing*, *58*(10), 5151–5164.

Long, Q., Jia, C., Boukouvalas, Z., Gabrielson, B., Emge, D., & Adali, T. (2018). Consistent run selection for independent component analysis: Application to fMRI analysis. In *2018 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 2581–2585).

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*(5), 465–471.

Sahonero-Alvarez, G., & Calderon, H. (2017). A comparison of SOBI, FastICA, JADE and Infomax algorithms. In *Proceedings of the 8th international multi-conference on complexity, informatics and cybernetics* (pp. 17–22).

Shah, C. A., Arora, M. K., Robila, S. A., & Varshney, P. K. (2002). ICA mixture model based unsupervised classification of hyperspectral imagery. In *Applied imagery pattern recognition workshop, 2002. proceedings.* (pp. 29–35).

Smilde, A. K., Kiers, H. A., Bijlsma, S., Rubingh, C., & Van Erk, M. (2009). Matrix correlations for high-dimensional data: the modified rv-coefficient. *Bioinformatics*, *25*(3), 401–405.

Vigrio, R., Jousmki, V., Hmlinen, M., Hari, R., & Oja, E. (1998). Independent component analysis for identification of artifacts in magnetoencephalographic recordings. *Advances in neural information processing systems*, *10*, 229–235.

A I_q Index Values

Table 2: I_q index values presented in Figures 2b, 3b, and 4b.

Configuration					
(1)		(2)		(3)	
Cluster	I_q Index Value	Cluster	I_q Index Value	Cluster	I_q Index Value
1	0.999091452192222	2	0.998883521455306	1	0.986241148443553
2	0.998777591977537	1	0.998913943469279	2	0.986106221451974
3	0.998196645291537	3	0.998418213582830	3	0.979013503449766
4	0.997565750316272	4	0.998082403760301	4	0.968046009918210
5	0.997252302086494	6	0.993655807994599	5	0.929804055099880
6	0.993945047784676	5	0.995119565997337	7	0.892572662448049
7	0.989683193381351	7	0.993219540408289	6	0.907177609674446
8	0.987997741135066	8	0.989979135898155	9	0.852670862619801
10	0.979060900199316	9	0.989228145422494	8	0.866677416152906
9	0.981630962574155	11	0.979550843405942	11	0.772258532720923
12	0.949779035635608	12	0.986872366265747	16	0.805349179319062
11	0.978518779909557	10	0.984203547144850	10	0.685132691909284
13	0.945880371092350	13	0.959937612499798	17	0.681380178650770
14	0.929654344366549	14	0.917380300942012	12	0.673866779211718
15	0.880078063091993	15	0.908975220403593	13	0.673671357713141
17	0.853489864143088	17	0.747995257462368	19	0.791441621648217
16	0.859013705724211	18	0.808842623545147	14	0.697016430804388
18	0.809473697964739	16	0.761549378828881	15	0.623242822225613
19	0.778791492371745	19	0.725870894499537	18	0.676897475664531
20	0.669381166662484	20	0.687150587858765	20	0.544289696419271

B GIFT Parameters and Files

B.1 Icasso with FastICA

B.1.1 pow3, tanh, and gauss

TR in seconds = 1

number of estimated independent components: yes, MDL (iid) method

number of IC = 17

Autofill data reduction values: yes

Serial Group ICA

B.2 Infomax

(a) (b)

Figure 12: Infomax Parameters in the GIFT toolbox; results are visualised in Section 5.2.2

The MEG results are provided in folders named Configuration 1, 2, and 3, with an executable script and workspace. The following files contain the fMRI result structures (MATLAB data) of the analysis which are presented in Section 5:

1. IcassoFastICA_icasso_results: results of Icasso with *pow3* and random initial conditions

2. tanh.fastica_icasso_results: results of Icasso with *tanh* and random initial conditions

3. gauss.fastica_icasso_results: results of Icasso with *gauss* and random initial conditions
4. configuration2.fastica_icasso_results: results of Icasso with *pow3* and random initial conditions and bootstrapping
5. IcassoInfomax.infomax_icasso_results: results of Icasso with Infomax
6. mst.ebm_mst_results: results of MST with ICA-EBM

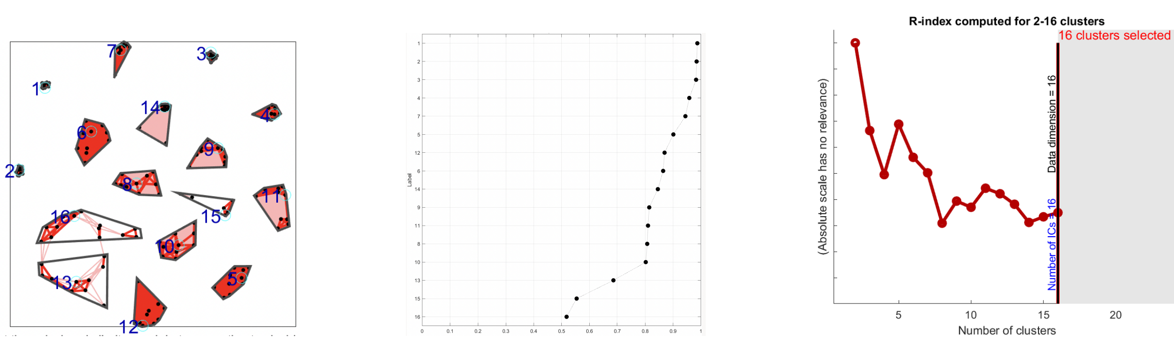
These results can be run with the GIFT toolbox in the following manner: Run Analysis in the toolbox, choose the file named *technique*_parameter_info.mat, where *technique* is the chosen ICA method and configuration such as Icasso with FastICA for instance, with *tanh* as the non-linearity.

C Icasso with Varying Numbers of Clusters for fMRI Data

These files are named tanh.fastica_icasso_results and both.fastica_icasso_results, respectively.

C.1 16 Clusters with tanh

Here, the results of running Icasso with random initial conditions, *tanh*, and 16 clusters are briefly discussed. There is no marked improvement in performance - that is, the quality of clusters is not necessarily better. Therefore, Icasso with *pow3* remains as the best performing configuration. Notably, the $R-$index shows a minimum at $L = 8$ in Figure 13c. However, this case is not considered, as it is evident that Icasso with *pow3* is able to estimate a greater number of independent components with more reliability. Thus, any further analysis regarding this specification is not conducted.

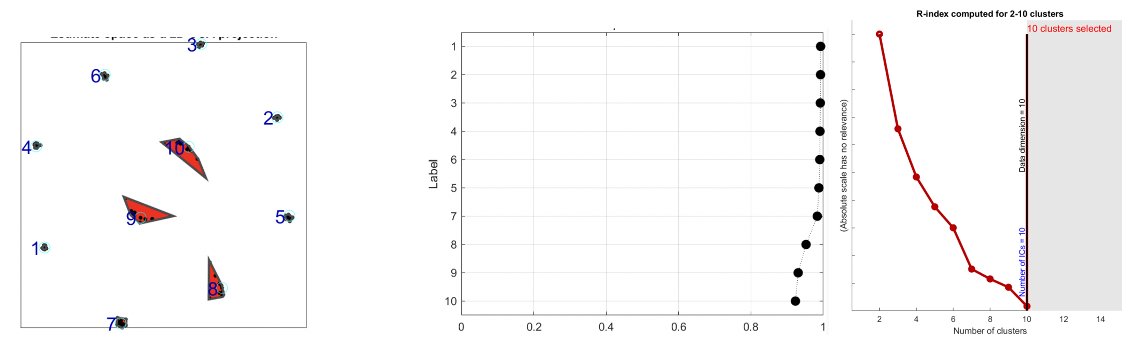


(a) Clusters **(b)** I_q index **(c)** $R-$index

Figure 13: Results using *tanh* with 16 clusters

C.2 10 Clusters with Random Initial Conditions and Bootstrapping

Presented below are the independent components that are estimated through Icasso with random initial conditions and bootstrapping, *pow3* non-linearity, and $L = 10$. Figure 14b illustrates that the quality of clusters is better than the original specification with $L = 17$. However, the trade-off between the number of components that are estimated versus those that are accurately estimated means that Icasso with *pow3* is still preferred.



(a) Clusters **(b)** I_q index **(c)** $R-$index

Figure 14: Results using random initial conditions and bootstrapping with 10 clusters

D MST Correlation Metrics

Table 3: Correlations between 17 independent components and their $T-$maps over 10 runs of MST with ICA-EBM. Runs 6 and 9 each consist of six components that have the highest correlation with their $T-$maps across a specific run. Components from run 9 have a higher correlation that those from run 6 in most cases, so run 9 is selected as the best.

	Run									
Independent Components	1	2	3	4	5	6	7	8	9	10
1	0.482	0.491	0.492	0.492	0.491	0.481	0.482	0.492	0.491	0.492
2	0.943	0.959	0.958	0.958	0.957	0.946	0.943	0.958	0.957	0.958
3	0.604	0.635	0.636	0.636	0.636	0.604	0.604	0.636	0.636	0.636
4	0.990	0.991	0.990	0.990	0.991	0.990	0.990	0.989	0.991	0.990
5	0.844	0.839	0.840	0.841	0.844	0.845	0.844	0.841	0.845	0.840
6	0.981	0.979	0.978	0.979	0.979	0.981	0.980	0.979	0.979	0.980
7	0.989	0.989	0.988	0.988	0.989	0.989	0.988	0.988	0.989	0.988
8	0.900	0.909	0.900	0.900	0.902	0.901	0.899	0.900	0.902	0.900
9	0.898	0.870	0.869	0.872	0.871	0.903	0.898	0.872	0.872	0.874
10	0.984	0.982	0.978	0.979	0.977	0.985	0.984	0.979	0.977	0.979
11	0.863	0.885	0.892	0.891	0.894	0.890	0.877	0.891	0.894	0.891
12	0.973	0.964	0.962	0.963	0.965	0.969	0.972	0.963	0.965	0.963
13	0.765	0.752	0.772	0.771	0.768	0.770	0.771	0.771	0.768	0.772
14	0.823	0.813	0.831	0.831	0.828	0.818	0.837	0.832	0.828	0.832
15	0.902	0.888	0.893	0.894	0.897	0.895	0.898	0.894	0.897	0.895
16	0.909	0.793	0.922	0.922	0.922	0.927	0.912	0.922	0.922	0.923
17	0.835	0.752	0.878	0.878	0.878	0.830	0.846	0.877	0.878	0.878