Politecnico di Milano
Department of Mathematics
Doctoral Programme In Mathematical Models and Methods in Engineering

---

Blind Source Separation methods for high-dimensional, massive, and complex data

Doctoral Dissertation of:
Paolo Zanini

Supervisor:
Prof. Piercesare Secchi

Co-advisor:
Dr. Simone Vantini

Tutor:
Prof. Piercesare Secchi

The Chair of the Doctoral Program:
Prof. Roberto Lucchetti

Year 2014 – XXVI Cycle

"The whole ocean is present at the back of each wave."

Swami Vivekananda

Contents

List of Figures

List of Tables

Introduction

The aim of this manuscript is to analyze Blind Source Separation problems for high-dimensional, massive, and complex data, and to propose new methods to face these statistical problems. Blind Source Separation problem consists of retrieving a set of unobserved source signals from a set of observed mixed signals, according to some a priori hypotheses on the sources and/or on the mixing process (see [12] for a wide and detailed description of BSS). Specifically we rely on the following model; let $\mathbf{X} \in \mathbb{R}^p$ be a random vector and assume the existence of a vector $\mathbf{S} \in \mathbb{R}^K$ representing K latent random sources and such that

$$\mathbf{X} = \mathcal{A}(\mathbf{S}), \tag{1}$$

where $\mathcal{A} : \mathbb{R}^K \to \mathbb{R}^p$ is an unknown mapping from $\mathbb{R}^K$ to $\mathbb{R}^p$ called mixing process. In this manuscript we consider two simplified assumptions related to model (1). In particular we assume $K \leq p$ and $\mathcal{A}$ is a linear process. Then model (1) reads

$$\mathbf{X} = A\mathbf{S}, \tag{2}$$

where A is an unknown $p \times K$ matrix of real numbers called mixing matrix. Therefore the columns of A constitute a basis of a K-dimensional subspace of $\mathbb{R}^p$; for this reason A is also called basis matrix. If the rows of the $n \times p$ matrix $\mathbb{X}$ collect n observed realizations $\mathbf{x}_1, ..., \mathbf{x}_n \in \mathbb{R}^p$ of the random vector $\mathbf{X}$ while the rows of the $n \times K$ matrix $\mathbb{S}$ represent the corresponding unobserved realizations of the latent random vector $\mathbf{S}$, model (2) implies that

$$\mathbb{X} = \mathbb{S}A'. \tag{3}$$

A BSS problem consists in estimating A and $\mathbb{S}$, given $\mathbb{X}$.

BSS problems are widespread in a lot of different fields. They became popular for those areas focused on temporal signals, like speech and audio signals, telecommunication systems and medical signal processing (e.g., electroencephalograms signals). Nowadays BSS methods are also frequently applied to more complex data, such as texts, images and tensors. Image processing, in particular, provides very different applications for BSS techniques. A typical example is the functional Magnetic Resonance Imaging (fMRI), a functional neuroimaging procedure that measures

brain activity and provides brain images that can be processed through BSS methods. Another possible application is the analysis of geo-referred images gathered by GIS systems.

All these applications share a common feature; the complexity of the available data is sharply increasing, due to the big improvements in the technology. Then, new statistical methods need to deal with high-dimensional, massive, and complex data, that often contain redundant information and hence make it difficult to extrapolate the relevant features. The crucial purpose is to find a space of small dimension where data can be easily analyzed without losing their significative features.

Many approaches are commonly used to solve a BSS problem. The most common is Principal Component Analysis (PCA) (see [23] for a detailed description). PCA is a powerful method to find optimal subspaces where to represent data, but it presents some drawbacks. In particular, PCA yields an orthonormal basis (i.e., the columns of A are orthogonal vectors); in many circumstances orthogonality is a desirable property but in some it introduces an artificial constraint not related to the phenomenological characteristics of the analyzed problem. For this reason basis elements provided by PCA might not represent physical features of the phenomenon under study. Indeed PCA is a model free method and this lack of assumptions might lead to solution that cannot be interpretable. The idea, instead, is to take into account some assumptions on model (3). These assumptions can be made on the source matrix $\mathbb{S}$ and/or on the basis matrix A. Among the methods that make assumptions on the sources the most popular is Independent Component Analysis (ICA) [22], which surmises the stochastic independence between the sources. Other approaches, instead, look for a sparse basis matrix, since some of the relevant features may involve a great number of the primitive variables describing the data set while others may be restricted only to a few. Hence a multi-resolution analysis is desirable. Among the others we cite Wavelets and Treelets [32, 29]. In a third group of BSS methods can be considered those methods which aim to find interpretable solution imposing some constraints on $\mathbb{S}$ and/or A in the estimate procedure. These constraints typically come from some a priori knowledge on the source matrix and the basis matrix. Among these methods we cite the Nonnegative Matrix Factorization (NMF) [27, 28], which solves the BSS problem imposing the nonnegativity constraint on both the elements of $\mathbb{S}$ and A.

In this manuscript we present new methods and interesting applications for BSS problems, particularly suited for high-dimensional, massive, and complex dataset. Specifically the manuscript is organized as follows.

Part I, Hierarchical Independent Component Analysis: in this part we present Hierarchical Independent Component Analysis (HICA), a

new method which simultaneously introduces sparsity and multi-resolution on the basis matrix A to obtain meaningful basis elements and sources. In particular in Chapter 1 we describe the algorithm and we prove some consistency results. We also show the efficiency of our method through some simulations. In Chapter 2 we apply HICA to an Electroencephalography (EEG) dataset, comparing the results provided by HICA with those obtained through other popular methods.

Part II, spatial colored Independent Component Analysis: in this part we present spatial colored Independent Component Analysis (scICA), a new method that extends ICA by exploiting the dependence structure within the sources, precisely when sources are generated by a spatial stochastic process. In Chapter 3 we introduce the new method and we show some simulations to validate it. In Chapter 4 we apply scICA to a geo-referenced dataset describing the mobile-phone traffic in the area of Milan, Italy. We compare scICA with other popular methods, also considering the HICA method described in Part I.

Part III, Alternating Least Square for Functional Data with equality and inequality constraints: in this part we analyze the resolution of the Nonnegative Matrix Factorization through an Alternating Least Square algorithm. In particular we focus on two issues. The first one is related on how to face the problem when different kind of constraints have to be imposed. The second regards the Alternating Least Square algorithm for functional data and, in particular, how to deal with the problem of misalignment of functional data. In Chapter 5 we describe the algorithm and the solutions we propose to deal with these two issues, while in Chapter 6 we show an application to the analysis of the gas chromatograms of chemical mixtures.

Part IV, Computational details: this part is dedicated to the computational details. Specifically we present the help of the R package fastHICA we developed to implement HICA method described in Part I.

Part I

Hierarchical Independent Component Analysis

Chapter 1

Methods and simulations

The statistical analysis of high-dimensional and complex data often requires the solution of two related issues: a data-driven dimensional reduction and a meaningful multiscale approximation. We look for a basis generating a space of small dimension where to represent data. We long for basis elements which are representative of the significant features of the phenomenon under study; some of these may involve a great number of the primitive variables describing the data set while others may be restricted only to a few. Hence a multi-resolution analysis is desirable. In this chapter we propose a new method for the construction of a multi-scale non-orthogonal data-driven basis.

We refers to the BSS model (3) and we focus on methods which aim to find multi-resolution basis matrix. PCA and ICA, two of the most common techniques to face BSS problems, provide basis elements defined globally, involving all the variables. Hence, if the interest is in catching multi-resolution behaviors, they are not suitable. Wavelets are commonly used (see, for instance, [32] and [37]) to generate a localized and multi-scale basis for data representation. Their main limitation is that the wavelet basis is not data-driven, since basis elements are fixed, regardless of the data. The Treelets algorithm is an efficient and recent approach that avoids this problem [29]. The Treelets algorithm generates a multi-scale orthonormal data-driven basis yielding a hierarchical tree that, at each level, represents data through an orthonormal basis. Thus the problem of interpretability of basis elements due to the exogenously imposed constraint of orthogonality still holds. We here propose a new approach able to provide a multi-scale non orthogonal data-driven basis through the integration between ICA and Treelets: we call it Hierarchical Independent Component Analysis (HICA).

The rest of the chapter is then organized as follows. In Section 1.1 we briefly describe Independent Component Analysis and the Treelets algorithm in order to introduce HICA in the second part of the section. In Section 1.2 we consider a procedure for data dimensional reduction with a

non-orthogonal basis that will be used in HICA. Then, in Section 1.3, we present some theoretical properties of the HICA method. Finally, in Section 1.4 we show some simulations which validate the algorithm proposed.

1.1 Hierarchical Independent Component Analysis

In the first part of this section we describe the main ideas concerning ICA and Treelets, since HICA is obtained by integrating these two approaches.

1.1.1 Independent Component Analysis

Independent Component Analysis is a method commonly used to solve Blind Source Separation problems. Consider model (2) and assume $K = p$. Given the data matrix $\mathbb{X}$, ICA looks for estimates of the basis matrix A and of the source matrix $\mathbb{S}$ in model (3), such that the columns of $\mathbb{S}$ could be taken as samples of the independent components of $\mathbf{S}$.

The ICA model presents two ambiguities. The first is label switching. The second is due to the fact that the independent components $S_1, ..., S_K$ of the vector $\mathbf{S}$ - i.e. the sources - are identifiable only up to multiplicative constants. Hence, for identifiability, the variances of the independent components are usually constrained to be 1; without loss of generality, we also assume that both the vector $\mathbf{X}$ and the vector $\mathbf{S}$ have zero mean. Moreover it is common to preprocess data by whitening $\mathbf{X}$ through a transformation matrix D. The covariance matrix of the transformed vector $\mathbf{Z} = D\mathbf{X}$ is required to be the identity, i.e. $E[\mathbf{Z}\mathbf{Z}'] = I$; for instance, $\mathbf{Z}$ is found by PCA on the standardized components of $\mathbf{X}$. Therefore model (2) becomes $\mathbf{Z} = (DA)\mathbf{S}$. Since $E[\mathbf{S}\mathbf{S}'] = I$, one then derives

$$I = E[\mathbf{Z}\mathbf{Z}'] = E[DA\mathbf{S}\mathbf{S}'A'D'] = DAE[\mathbf{S}\mathbf{S}']A'D' = (DA)(DA)'.$$

Hence $A^* = DA$ is orthogonal. Once the optimal rotation A^* has been found, A is obtained as $D^{-1}A^*$.

Existence of a basis for data representation through independent components is not guaranteed (differently from a representation through uncorrelated components which always exists, and it is found by PCA). In practical problems, the estimate of the matrix A^* is obtained through the minimization of the empirical dependence between the columns of $\mathbb{S}$. In [21] it is shown that A^* can be found by maximizing the non-gaussianity of the sources $S_1, ..., S_K$. This simplifies the ICA optimization problem and suggests some suitable numerical algorithm for its solution. In this manuscript all analyses regarding ICA will be carried out with the fastICA algorithm, which maximizes a non-gaussianity measures (e.g. the absolute value of the kurtosis) through a fast fixed-point procedure. (Details about the algorithm are presented in [21]).

Comparing the ICA solution with that provided by PCA, we note that while PCA yields a basis whose elements are conveniently arranged for dimensional reduction, this is not so for ICA which is useless for this purpose. A common approach to circumvent this difficulty, and to allow for the number K of independent components to be much smaller than the number p of primitive variables, is to first project data into the K-dimensional space generated by the first K principal directions. Then, ICA is carried out in this reduced K-dimensional space.

1.1.2 Treelets

The Treelets algorithm [29] generates a multi-scale orthonormal basis for data representation, like wavelets, but the basis is data-driven. The Treelets algorithm yields a hierarchical tree that, at each level, replaces the two more correlated variables through a pair-wise Principal Component decomposition. The procedure consists of an iterative algorithm with $p-1$ steps. At each step three operations are performed:

1. compute the correlations between couples of variables and search for the two variables with the highest correlation;
2. compute a Principal Component Analysis in the space of the two selected variables;
3. store the second principal direction, that will not be processed in the following step, while the first principal direction replaces the two original variables in the variables set.

At each level $l = 0, ..., p-1$, the algorithm provides a multi-resolution data-driven orthogonal basis $B^{(l)}$, able to catch internal structural features of the data.

1.1.3 The HICA algorithm

The two methods presented above are useful to reduce the complexity of high-dimensional problems and to detect relevant features of the data. However some problems still hold. ICA, as PCA, is a global method that produces a non-sparse basis. Hence it is not suitable for a multi-resolution analysis. Treelets provide an orthonormal basis, whose elements can be unrelated to the phenomenological characteristics of the problem under study. Hierarchical Independent Component Analysis, instead, aims at the construction of a multi-scale non orthogonal data-driven basis through the integration between ICA and Treelets. Basically it consists in replacing in the Treelets algorithm the pair-wise Principal Component Analysis step with a pair-wise Independent Component Analysis step. With respect to

this manuscript wording, we should indeed refer to Treelet analysis as Hierarchical Principal Component Analysis (HPCA). Anyhow we preferred to keep the authors' original wording (i.e., Treelets).

A more detailed description of the HICA algorithm is in order. First we need to define a suitable similarity measure between two random variables. According to the ICA procedure, we search for a measure that is greater when the dependence between two variables is larger. In particular we consider the distance correlation, a measure of dependence introduced in [47] and based on the distance covariance. Let X_1 and X_2 be two random variables and let $\phi_{X_1}(t)$ and $\phi_{X_2}(s)$ be their characteristic functions, while $\phi_{(X_1,X_2)}(t,s)$ is the characteristic function of the random vector $(X_1,\, X_2)'$. Then, the distance covariance between X_1 and X_2 is the non-negative number $\mathcal{V}(X_1, X_2)$ defined as

$$\mathcal{V}(X_1, X_2) = \left(\frac{1}{c^2} \int_{\mathbb{R}^2} \frac{|\phi_{(X_1,X_2)}(t,s) - \phi_{X_1}(t)\phi_{X_2}(s)|}{t^2 s^2} dtds \right)^{\frac{1}{2}},$$

where $c = \frac{\pi}{\Gamma(1)}$ and $\Gamma(\cdot)$ is the complete gamma function. If we indicate with $\mathcal{V}(X_1) = \mathcal{V}(X_1, X_1)$, the distance correlation between two random variables X_1 and X_2 is defined as

$$\mathcal{R}(X_1, X_2) = \frac{\mathcal{V}(X_1, X_2)}{\sqrt{\mathcal{V}(X_1)\mathcal{V}(X_2)}}.$$

Note $0 \leq \mathcal{R}(X_1, X_2) \leq 1$ and $\mathcal{R}(X_1, X_2)$ can be considered to be a measure of dependence between X_1 and X_2 in the sense that $\mathcal{R}(X_1, X_2)$ is equal to 0 if and only if X_1 and X_2 are independent random variables. Moreover distance variance and distance covariance have some properties that will be used in the following. In particular:

1. if X_1 and X_2 are independent random variables, then $\mathcal{V}(X_1 + X_2) \leq \mathcal{V}(X_1) + \mathcal{V}(X_2)$;
2. if $(X_{11}, X_{21})'$ and $(X_{12}, X_{22})'$ are independent random vectors, then $\mathcal{V}(X_{11} + X_{12}, X_{21} + X_{22}) \leq \mathcal{V}(X_{11}, X_{21}) + \mathcal{V}(X_{12}, X_{22})$.

We now describe the HICA algorithm. At level $l = 0$ of the hierarchical tree each component $X_1, ..., X_p$ of the random vector $\mathbf{X}$ is represented by itself, the basis matrix $B^{(0)}$ is the canonical basis of dimension p and the coordinates vector $\mathbf{Y}^{(0)} = (Y_1^{(0)}, ..., Y_p^{(0)})'$ corresponds to the primitive variables (i.e., $Y_i^{(0)} = X_i$). Define $\mathfrak{C}$ to be a set of indices of the active variables, initializing $\mathfrak{C}^{(0)} = \{1, ..., p\}$, and compute the sample similarity matrix $\widehat{R}^{(0)}$, where $\widehat{R}_{ij}^{(0)} = \mathcal{R}(Y_i^{(0)}, Y_j^{(0)})$. Then, for $l = 1, ..., p-1$, repeat the following three steps:

1. in the first step the two most similar variables are found. In particular set:
$$(\alpha, \beta) = \arg \max_{i<j \in \mathfrak{C}^{(l-1)}} \widehat{R}_{ij}^{(l-1)};$$

2. compute an Independent Component Analysis of the variables $Y_\alpha^{(l-1)}$ and $Y_\beta^{(l-1)}$:
$$\begin{aligned} Y_\alpha^{(l-1)} &= a_{11}^{(l)} S_1 + a_{12}^{(l)} S_2 \\ Y_\beta^{(l-1)} &= a_{21}^{(l)} S_1 + a_{22}^{(l)} S_2 \end{aligned} \tag{1.1}$$
The idea is to replace $Y_\alpha^{(l-1)}$ with S_1 and $Y_\beta^{(l-1)}$ with S_2. Hence define the matrix
$$\widetilde{A}^{(l)} = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & \widetilde{a}_{11}^{(l)} & \cdots & \widetilde{a}_{12}^{(l)} & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & \widetilde{a}_{21}^{(l)} & \cdots & \widetilde{a}_{22}^{(l)} & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix}$$
where $\widetilde{a}_{11}^{(l)}$ and $\widetilde{a}_{22}^{(l)}$ are, respectively, in position (α, α) and (β, β). The elements $\widetilde{a}_{ij}^{(l)}$ correspond to the $a_{ij}^{(l)}$ in (1.1), normalized such that $\widetilde{A}^{(l)}$ has columns with unitary norm. $\widetilde{A}^{(l)}$ represents the non orthogonal transformation identified by ICA. The new basis matrix and coordinates vector become $B^{(l)} = B^{(l-1)}\widetilde{A}^{(l)}$ and $\mathbf{Y}^{(l)} = (\widetilde{A}^{(l)})^{-1}\mathbf{Y}^{(l-1)}$ respectively. The similarity matrix $\widehat{R}^{(l)}$ is then updated accordingly;

3. order the new variables according to their variances. If the variance of $Y_\alpha^{(l)}$ is greater than the variance of $Y_\beta^{(l)}$, store the variable $Y_\beta^{(l)}$ and, at the next step, consider only $Y_\alpha^{(l)}$ as a possible candidate for a new aggregation. This corresponds to remove the index β from the set $\mathfrak{C}$, defining $\mathfrak{C}^{(l)} = \mathfrak{C}^{(l-1)} \setminus \{\beta\}$. Otherwise store $Y_\alpha^{(l)}$ and set $\mathfrak{C}^{(l)} = \mathfrak{C}^{(l-1)} \setminus \{\alpha\}$.

The algorithm provides, at each level l, a non orthogonal basis matrix $B^{(l)} = B^{(0)}\widetilde{A}^{(1)} \cdots \widetilde{A}^{(l)}$ - an estimate of the basis matrix A - and a coordinates vector $\mathbf{Y}^{(l)} = \widetilde{A}^{(l)^{-1}} \cdots \widetilde{A}^{(1)^{-1}}\mathbf{Y}^{(0)}$, which is an estimate of the scores matrix $\mathbb{S}$.

1.2 Selection of the level of the tree and dimensional reduction with a non-orthogonal basis

The HICA algorithm generates p different matrices $B^{(0)}, ..., B^{(p-1)}$ as estimates of the basis matrix A. Obviously one cannot take into account all these different estimates, but it is reasonable to choose only one (or some) of them for the analysis. The more natural choice is to consider the estimate related to the maximum height of the tree, $l = p - 1$, but alternatively one can choose any of the basis given at the different levels l. At a generic level l, $B^{(l)}$ is composed by the l elements stored in the previous steps and the $p - l$ elements corresponding to variables of the active set $\mathfrak{C}^{(l)}$ that would be ready for aggregation in the following steps. Let C^l be a partition of $\{1, ..., p\}$ in $p - l$ sets named C_i^l, with $i = 1, ..., p - l$. By construction each basis element of $B^{(l)}$ is defined on a different set C_i^l of the partition (i.e., the position of the non-zero values of the i-th basis element correspond to the indexes of the set C_i^l). Since at each level a new variable is generated as a linear combination of two variables of the active set, the number of sets that form the partition is reduced by aggregating two of them. Hence at a specific level l the basis elements stored in the previous steps of the algorithm are defined on subsets of the C_i^l. Therefore we can divide basis elements of $B^{(l)}$ into $p - l$ different groups, according to the $p - l$ different sets of the partition. For this reason we can relate the different basis $B^{(l)}$ to different degrees of sparsity, where the different degrees refer to the different cardinalities of partitions. In particular, the lower is the level l considered, the greater is the degree of sparsity of the basis taken into account (i.e., greater is the cardinality of the partition).

Once a specific basis $B^{(l)}$ is chosen, another important aspect to consider is dimensional reduction. In particular we need to select the dimension K (with $K \leq p$) of a suitable subspace to represent data, choosing only K basis elements.

To jointly face these two problems (i.e., the choice of the degree of sparsity and the K "best" basis elements) we consider the energy, an index related to the fraction of variance explained by a basis. We now first describe the energy index, focusing on the non trivial case of its evaluation for a non-orthogonal basis. Then we propose a strategy to choose a suitable dimension K to represent data in a reduced space and, given K, we show how to select a specific basis $B^{(l)}$ and its K basis elements.

1.2.1 The energy index

Consider a basis $A = [\mathbf{a}_1; ...; \mathbf{a}_p]$, not necessarily orthogonal. Let $\mathcal{I}_K = \{i_1, i_2, ..., i_K\}$ be one of the $\binom{p}{K}$ subsets of the index set $\{1, ..., p\}$ with cardinality K, and let $A_{\mathcal{I}_K} = [\mathbf{a}_{i_1}; ...; \mathbf{a}_{i_K}]$. Let $\mathbf{X}^{A_{\mathcal{I}_K}} = A_{\mathcal{I}_K}(A_{\mathcal{I}_K}^T A_{\mathcal{I}_K})^{-1} A_{\mathcal{I}_K}^T \mathbf{X}$

be the orthogonal projection of $\mathbf{X}$ on the space spanned by $A_{\mathcal{I}_K}$, where $\mathbf{X} \in \mathbb{R}^p$ is a random vector with zero mean. Then we define

$$\gamma(A_{\mathcal{I}_K}) = \frac{E[\|\mathbf{X}^{A_{\mathcal{I}_K}}\|^2]}{E[\|\mathbf{X}\|^2]} = \frac{tr(\Sigma A_{\mathcal{I}_K}(A_{\mathcal{I}_K}^T A_{\mathcal{I}_K})^{-1} A_{\mathcal{I}_K}^T)}{tr(\Sigma)}$$

being $\Sigma = E[\mathbf{X}\mathbf{X}^T] = Cov(\mathbf{X})$, and we call $\gamma(A_{\mathcal{I}_K})$ the energy associated to the basis $A_{\mathcal{I}_K}$. At this point we define $\Gamma_K(A)$ as the maximum energy among all the $\binom{p}{K}$ energies associated to the K-dimensional subspaces obtained from the basis matrix A:

$$\Gamma_K(A) = \max_{\mathcal{I}_K \subseteq \{1,\dots,p\}} \gamma(A_{\mathcal{I}_K}). \tag{1.2}$$

If A is non orthogonal the evaluation of $\Gamma_K(A)$ may become cumbersome. The non orthogonality, in fact, implies that the elements of the best $K-1$-dimensional space are not necessary a subset of the elements of the best K-dimensional space. Therefore we propose a forward selection strategy that can be easily computed and, in practical problems, produces reasonable approximations of the space with maximal energy $\Gamma_K(A)$. The strategy is suitable not only for HICA, but whenever dealing with non orthogonal basis. We start by calculating the energy $\gamma([\mathbf{a}_k])$ for each basis element and we set the maximum energy element as the first element of the basis. Let it be $\mathbf{a}_{(1)}$. Then we look for the second basis element, named $\mathbf{a}_{(2)}$, such that

$$\mathbf{a}_{(2)} = \arg \max_{\mathbf{a}_j \neq \mathbf{a}_{(1)}} \gamma([\mathbf{a}_{(1)}; \mathbf{a}_j]).$$

Once $\mathbf{a}_{(1)}, \dots, \mathbf{a}_{(k)}$ have been identified, $\mathbf{a}_{(k+1)}$ is found accordingly:

$$\mathbf{a}_{(k+1)} = \arg \max_{\mathbf{a}_j \neq \mathbf{a}_{(1)},\dots,\mathbf{a}_{(k)}} \gamma([\mathbf{a}_{(1)}; \dots; \mathbf{a}_{(k)}; \mathbf{a}_j])$$

and the procedure continues until $\mathbf{a}_{(K)}$ is found.

Remark 1.1 *If A is an orthonormal matrix, the exact solution of the optimization problem (1.2) can be found efficiently since we do not need to evaluate all the $\binom{p}{K}$ energies $\gamma(A_{\mathcal{I}_K})$. In fact, let $V = [\boldsymbol{v}_1; \dots; \boldsymbol{v}_p]$ be an orthonormal basis, $\Gamma_K(V)$ is found by computing, for $j = 1, \dots, p$,*

$$\gamma([\boldsymbol{v}_j]) = \frac{E[(\boldsymbol{v}_j^T \boldsymbol{X})^2]}{E[\|\boldsymbol{X}\|^2]} = \frac{\boldsymbol{v}_j^T \Sigma \boldsymbol{v}_j}{tr(\Sigma)} = \frac{\sum_{i=1}^p \lambda_i (\boldsymbol{v}_j^T \boldsymbol{e}_i)^2}{\sum_{i=1}^p \lambda_i}$$

where λ_i and $\boldsymbol{e}_i$ are the eigenvalues and the eigenvectors of Σ. After sorting the basis elements according to their energy, such that $\gamma([\boldsymbol{v}_{(1)}]) \geq \gamma([\boldsymbol{v}_{(2)}]) \geq \cdots \geq \gamma([\boldsymbol{v}_{(p)}])$, $\Gamma_K(V)$ is obtained by summing the first K energy terms. In particular:

$$\Gamma_K(V) = \frac{tr(\Sigma V_K V_K^T)}{tr(\Sigma)} = \sum_{k=1}^K \gamma([\boldsymbol{v}_{(k)}]),$$

where $V_K = [\boldsymbol{v}_{(1)}; ...; \boldsymbol{v}_{(K)}]$. *This is the same procedure adopted in [29] for finding the elements of the* K*-dimensional basis and also coincides with the criterium used in PCA to order the principal directions. Indeed, if* $E = [\boldsymbol{e}_1; ...; \boldsymbol{e}_p]$ *is the matrix whose columns are the eigenvectors of* Σ, $\gamma([\boldsymbol{e}_j]) = \frac{\sum_{i=1}^{p} \lambda_i (\boldsymbol{e}_j^T \boldsymbol{e}_i)^2}{\sum_{i=1}^{p} \lambda_i} = \frac{\lambda_j}{\sum_{i=1}^{p} \lambda_i}$ *and* $\Gamma_K(E) = \sum_{i=1}^{K} \gamma([\boldsymbol{e}_{(k)}]) = \frac{\sum_{k=1}^{K} \lambda_{(k)}}{\sum_{i=1}^{p} \lambda_i}$.

1.2.2 Dimensional reduction and choice of a specific level l of the tree

We now focus on the energy index as a tool to perform dimensional reduction and to find the best basis between the p estimates provided by HICA. We first decide on the best value for K considering only the maximum height tree basis (i.e., considering only $\Gamma_K(B^{(p-1)})$). Once K has been determined, we compute $\Gamma_K(B^{(l)})$, for $l = 0, ..., p-1$, and we choose the best basis B_{best} according the same criterium adopted in [29] for Treelets:

$$B_{best} = \arg \max_{B_l : 0 \leq l \leq p-1} \Gamma_K(B_l).$$

This argmax is not necessarily unique. Indeed, at a specific level, say $l = p - k$, we have k elements (corresponding to the variables in the active set $\mathfrak{C}^{(p-k)}$) that in the following steps are merged together. It is straightforward to show that, if the best k-dimensional space was generated by these k elements, the quantity $\Gamma_k(B^{(p-k)})$ would not increase in the next levels, since, even if two of these elements are merged together, the space spanned by the new elements is the same. In general at level $p - k$ the best k-dimensional space need not be generated by the k active variables. However from the level when all variables of the active set $\mathfrak{C}^{(l)}$ constitute the best k-dimensional basis, the quantity $\Gamma_k(B^{(l)})$ does not increase. Hence we could have more than one basis with the same energy. The choice suggested in [29] is to take into account the basis with the smallest l. Such proceeding could however discard solutions which are able to better catch the underlying structure of the problem. For this reason we suggest to consider all basis with the same highest energy Γ_K. They might have different degrees of sparsity and this can make some of them more preferable. In the examples of Section 1.4 we will deepen the analysis of this issue.

1.3 Theoretical results

In this section we analyze the consistence of HICA when data are generated by K independent sources with disjoint supports plus some noise. Specifically we consider a situation where the p primitive variables are divided into K groups, with dependent variables within groups and independent

variables between groups. We want to show that HICA is well suited for representing and catching the underlying structure of this kind of data, providing at level $p-K$ loading vectors whose supports are defined on the different groups. We show this result in Lemma 1.2 and Theorem 1.1, after the discussion of a preliminary property in Lemma 1.1.

We start by dealing with an issue directly connected to the fact that the fastICA algorithm is grounded on non-gaussianity measures. In some special situations the directions maximizing kurtosis, a well-known non-gaussianity measure, can be found analytically, as it is proved in the following lemma.

Lemma 1.1 *Let T be a random variable such that $kurt(T) \neq 0$ and let E be a gaussian random variable. Set $\boldsymbol{Z} = (T, E)'$ and assume that T and E are independent. Let $\boldsymbol{w} = (w_1, w_2)'$ be a vector of unitary norm. The absolute value of the kurtosis of the random variable $\boldsymbol{w}'\boldsymbol{Z}$ is maximized by $\boldsymbol{w}_{max} = (1, 0)'$.*

Proof. For simplicity we consider T and E to be zero mean and unit variance random variables. The kurtosis of a zero mean and unit variance random variable Y is $kurt(Y) = E[Y^4] - 3$. If Y is gaussian, $kurt(Y) = 0$. Moreover if Y_1 and Y_2 are independent random variables and α e β real parameters, $kurt(\alpha Y_1 + \beta Y_2) = \alpha^4 kurt(Y_1) + \beta^4 kurt(Y_2)$. Hence:

$$\begin{aligned} &|kurt(\mathbf{w}'\mathbf{Z})| = |kurt(w_1 T + w_2 E)| = \\ &= |w_1^4 kurt(T) + w_2^4 kurt(E)| = |w_1^4 kurt(T)|. \end{aligned} \tag{1.3}$$

Since $kurt(T) \neq 0$, (1.3) is maximized by $w_1 = \pm 1$ (and $w_2 = 0$ because $\mathbf{w}$ is a vector of unitary norm). □

We now deal with p non-gaussian random variables identical but for an additive gaussian noise, in order to show that, in this particular case, HICA provides a constant loading vector at the final level $l = p - 1$, thus gathering the common component.

Lemma 1.2 *Let T be a random variable with 0 mean, $kurt(T) \neq 0$ and such that $\mathcal{V}(T) = 1$. Let $\boldsymbol{X} = (X_1, ..., X_p)' \in \mathbb{R}^p$ be a random vector such that, for $\sigma^2, \sigma_e^2 > 0$,*

$$X_i = \sigma^2 T + \sigma_e^2 E_i, \qquad i = 1, ..., p,$$

with E_i a random gaussian noise such that, for $i, j = 1, ...p$, $i \neq j$, $\mathcal{V}(E_i) = 1$, $\mathcal{V}(E_i, E_j) = 0$ and $\mathcal{V}(E_i, T) = 0$. At each level $1 \leq l \leq p-1$ the HICA decomposition reads:

$$B^{(l)} = [\boldsymbol{a}_1^{(l)}; ...; \boldsymbol{a}_{p-l}^{(l)}; \widetilde{\boldsymbol{a}}_1^{(l)}; ...; \widetilde{\boldsymbol{a}}_l^{(l)}]$$

$$\boldsymbol{Y}^{(l)} = (Y_1^{(l)}, ..., Y_{p-l}^{(l)}, \widetilde{Y}_1^{(l)}, ..., \widetilde{Y}_l^{(l)})'$$

where $\boldsymbol{a}_i^{(l)} = \frac{1}{\sqrt{|C_i^l|}} I_{C_i^l}$ and $Y_i^{(l)} = \frac{|C_i^l|}{\sqrt{|C_i^l|}} \sigma^2 T + \frac{\sigma_e^2}{\sqrt{|C_i^l|}} E_i^{(l)}$, with $\mathcal{V}(E_i^{(l)}) \leq |C_i^l|$ and $\mathcal{V}(Y_i^{(l)}) \leq \sqrt{|C_i^l|}(\sigma^2 + \sigma_e^2)$ $\forall i = 1, ..., p-l$ (the sets C_i^l have been defined

in Section 1.2). In particular, at the level $l = p-1$, $\boldsymbol{a}_1^{(p-1)} = (\frac{1}{\sqrt{p}}, ..., \frac{1}{\sqrt{p}})'$ *and* $Y_1^{(p-1)} = \frac{p}{\sqrt{p}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{p}}E_1^{(p-1)}$, *with* $\mathcal{V}(E_1^{(p-1)}) \leq p$ *and* $\mathcal{V}(Y_1^{(p-1)}) \leq \sqrt{p}(\sigma^2 + \sigma_e^2)$.

Proof. Suppose that the aggregation between variables follows the scheme: $\{\cdots\{\{X_1, X_2\}, X_3\}\cdots, X_p\}$. Hence, at level $l = 1$ we aggregate:

$$X_1 = \sigma^2 T + \sigma_e^2 E_1$$
$$X_2 = \sigma^2 T + \sigma_e^2 E_2.$$

The whitening procedure of ICA, transforms the vector $\mathbf{X} = (X_1\, X_2)'$ in a new vector $\mathbf{Z} = (Z_1\, Z_2)'$ such that

$$Z_1 = \frac{X_1+X_2}{a} = \frac{2\sigma^2 T+\sigma_e^2(E_1+E_2)}{a}$$
$$Z_2 = \frac{X_1-X_2}{b} = \frac{\sigma_e^2(E_1-E_2)}{b}$$

where a and b are, respectively, the standard deviations of $X_1 + X_2$ and $X_1 - X_2$. We observe that Z_1 is a non gaussian variable, while Z_2 is gaussian. Because of Lemma 1.1 the rotation found by fastICA in the whitened space coincides with the identity matrix. According to the selection criterium and taking into account the normalization of the matrix $\widetilde{A}^{(1)}$ in step 2 of the HICA algorithm, we obtain $\mathbf{a}_1^{(1)} = (\frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}0\cdots 0)'$ and $Y_1^{(1)} = \frac{1}{\sqrt{2}}X_1 + \frac{1}{\sqrt{2}}X_2 = \frac{2}{\sqrt{2}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{2}}E^{(1)}$, where $E^{(1)} = E_1 + E_2$ and $\mathcal{V}(E^{(1)}) \leq \mathcal{V}(E_1) + \mathcal{V}(E_2) \leq 2$. Furthermore $\mathcal{V}(Y_1^{(1)}) \leq \mathcal{V}(\frac{2}{\sqrt{2}}\sigma^2 T) + \mathcal{V}(\frac{\sigma_e^2}{\sqrt{2}}E_1^{(1)}) \leq \sqrt{2}(\sigma^2 + \sigma_e^2)$. At level $l = 2$ we aggregate:

$$Y_1^{(1)} = \frac{2}{\sqrt{2}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{2}}E_1^{(l)}$$
$$X_3 = \sigma^2 T + \sigma_e^2 E_3.$$

The whitening procedure provides a vector $\mathbf{Z} = (Z_1\, Z_2)'$ such that

$$Z_1 = \frac{\sqrt{2}Y_1^{(1)}+X_3}{a'} = \frac{3\sigma^2 T+\sigma_e^2(E_1^{(1)}+E_3)}{a'}$$
$$Z_2 = \frac{Y_1^{(1)}-\sqrt{2}X_3}{b'} = \frac{\sqrt{2}\sigma_e^2(E_1^{(1)}/2-E_3)}{b'}$$

where a' and b' are, respectively, the standard deviations of $\sqrt{2}Y_1^{(1)} + X_3$ and $Y_1^{(1)} - \sqrt{2}X_3$. Once again, Lemma 1.1 implies that the rotation provided by fastICA is the identity and according to the selection criterium and to the normalization of $\widetilde{A}^{(2)}$ we have

$$\mathbf{a}_1^{(2)} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \sqrt{\frac{2}{3}} \\ \frac{1}{\sqrt{3}} \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

and $Y_1^{(2)} = \sqrt{\frac{2}{3}}Y_1^{(1)} + \frac{1}{\sqrt{3}}X_3 = \frac{3}{\sqrt{3}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{3}}E_1^{(2)}$, where $E_1^{(2)} = E_1^{(1)} + E_3$ and $\mathcal{V}(E_1^{(2)}) \leq \mathcal{V}(E_1^{(1)}) + \mathcal{V}(E_3) \leq 3$. Moreover $\mathcal{V}(Y_1^{(2)}) \leq \mathcal{V}(\frac{3}{\sqrt{3}}\sigma^2 T) + \mathcal{V}(\frac{\sigma_e^2}{\sqrt{3}}E_1^{(2)}) \leq \sqrt{3}(\sigma^2 + \sigma_e^2)$. Iterating, we obtain the lemma when the aggregation scheme is $\{\cdots\{\{X_1, X_2\}, X_3\}\cdots, X_p\}$.

For a general aggregation scheme, at the level $l + 1 = 2, ..., p - 1$ we aggregate:

$$Y_i^{(l)} = \frac{m}{\sqrt{m}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{m}}E_i^{(l)}$$
$$Y_j^{(l)} = \frac{n}{\sqrt{n}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{n}}E_j^{(l)}$$

with $C_i^l \cap C_j^l = \emptyset$ and $m + n = l + 2$. The whitening procedure provides a vector $\mathbf{Z} = (Z_1\ Z_2)'$ such that

$$Z_1 = \frac{\sqrt{m}Y_i^{(l)} + \sqrt{n}Y_j^{(l)}}{a''} = \frac{(m+n)\sigma^2 T + \sigma_e^2(E_i^{(l)} + E_j^{(l)})}{a''}$$

$$Z_2 = \frac{\sqrt{n}Y_i^{(l)} - \sqrt{m}Y_j^{(l)}}{b''} = \frac{\sqrt{mn}(E_i^{(l)}/m - E_j^{(l)}/n)}{b''}$$

where a'' and b'' are, respectively, the standard deviations of $\sqrt{m}Y_i^{(l)} + \sqrt{n}Y_j^{(l)}$ and $\sqrt{n}Y_i^{(l)} - \sqrt{m}Y_j^{(l)}$. Then

$$\mathbf{a}_i^{(l+1)} = \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ \frac{1}{\sqrt{m}} & 0 \\ \vdots & \vdots \\ \frac{1}{\sqrt{m}} & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & \frac{1}{\sqrt{n}} \\ \vdots & \vdots \\ 0 & \frac{1}{\sqrt{n}} \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \sqrt{\frac{m}{m+n}} \\ \sqrt{\frac{n}{m+n}} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \frac{1}{\sqrt{m+n}} \\ \vdots \\ \frac{1}{\sqrt{m+n}} \\ 0 \\ \vdots \\ 0 \\ \frac{1}{\sqrt{m+n}} \\ \vdots \\ \frac{1}{\sqrt{m+n}} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

and $Y_i^{(l+1)} = \frac{\sqrt{m}Y_i^{(l)} + \sqrt{n}Y_j^{(l)}}{\sqrt{m+n}} = \frac{m+n}{\sqrt{m+n}}\sigma^2 T + \frac{\sigma_e^2}{\sqrt{m+n}}E_i^{(l+1)}$, where $E_i^{(l+1)} = E_i^{(l)} + E_j^{(l)}$ and $\mathcal{V}(E_i^{(l+1)}) \leq \mathcal{V}(E_i^{(l)}) + \mathcal{V}(E_j^{(l)}) \leq m+n$. Moreover $\mathcal{V}(Y_i^{(l+1)}) \leq \mathcal{V}(\frac{m+n}{\sqrt{m+n}}\sigma^2 T) + \mathcal{V}(\frac{\sigma_e^2}{\sqrt{m+n}}E_i^{(l+1)}) \leq \sqrt{m+n}(\sigma^2 + \sigma_e^2)$. The result now follows by induction. □

Lemma 1.2 is instrumental for proving the main theoretical result for HICA. If variables are dependent according to an approximate block structure where variables in the same block are exchangeable and strongly dependent while variables in different blocks are weakly dependent, then HICA is able to uncover this feature providing loading vectors constants on each block and null elsewhere.

Theorem 1.1 *Let $T_1, ..., T_K$ be random variables with 0 mean, non zero kurtosis and such that $\mathcal{V}(T_k) = 1$, $k = 1, ..., K$. Let $\boldsymbol{X} = (X_{11}, ..., X_{1p_1}, ..., X_{K1}, ..., X_{Kp_K})' \in \mathbb{R}^p$ be a random vector such that, for $\sigma_1^2, ..., \sigma_K^2, \sigma_e^2 > 0$,*

$$X_{ji} = \sigma_j^2 T_j + \sigma_e^2 E_{ji}$$

with E_{ji} random gaussian noise such that $\mathcal{V}(E_{ji}) = 1$, $\mathcal{V}(E_{ji}, E_{hl}) = 0$ and $\mathcal{V}(E_{ji}, T_h) = 0$ for $j, h = 1, ..., K$, $i = 1, ..., p_j$ and $l = 1, ..., p_h$. Furthermore set $\mathcal{V}(\sigma_j^2 T_j, \sigma_h^2 T_h) = \sigma_{jh}$ and assume that

$$\max_{1 \leq j,h \leq K} \left(\frac{\sigma_{jh}}{\sigma_j \sigma_h} \right) < \frac{c(\sigma_e)}{1 + \delta^2} \tag{1.4}$$

with $\delta = \frac{\sigma_e}{\min_{1\leq j\leq K}\sigma_j}$ *and* $c(\sigma_e)$ *a constant such that* $0 < c(\sigma_e) \leq 1$ *and* $c(\sigma_e) \overset{\sigma_e\to 0}{\longrightarrow} 1$. *Then, at level* $l = p - K$, *the HICA decomposition reads:*

$$B^{(p-K)} = [\boldsymbol{a}_1^{(p-K)}; ...; \boldsymbol{a}_K^{(p-K)}; \widetilde{\boldsymbol{a}}_1^{(p-K)}; ...; \widetilde{\boldsymbol{a}}_{p-K}^{(p-K)}]$$
$$\boldsymbol{Y}^{(p-K)} = (Y_1^{(p-K)}, ..., Y_K^{(p-K)}, \widetilde{Y}_1^{(p-K)}, ..., \widetilde{Y}_{p-K}^{(p-K)})'$$

where $\boldsymbol{a}_i^{(p-K)} = \frac{1}{\sqrt{|F_i|}} I_{F_i}$ *and* $Y_i^{(p-K)} = \frac{|F_i|}{\sqrt{|F_i|}}\sigma_i^2 T_i + \frac{\sigma_e^2}{\sqrt{|F_i|}} E_i^{(p-K)}$, *with* $\mathcal{V}(E_i^{(p-K)}) \leq |F_i|$, $\mathcal{V}(Y_i^{(p-K)}) \leq \sqrt{|F_i|}(\sigma_i^2 + \sigma_e^2)$ *and* $F_i = \{i1, ..., ip_i\}$, *for* $i = 1, ..., K$.

Proof. Assume that, at a generic level $l < p - K$ of the tree, random variables from different blocks have not been merged. Hence, from Lemma 1.2, any two variables in the active set $\mathfrak{C}$ have the form:

$$Y_u^{(l)} = \tfrac{m}{\sqrt{m}}\sigma_u^2 T_u + \tfrac{\sigma_e^2}{\sqrt{m}} E_u^{(l)}$$
$$Y_v^{(l)} = \tfrac{n}{\sqrt{n}}\sigma_v^2 T_v + \tfrac{\sigma_e^2}{\sqrt{n}} E_v^{(l)}$$

with $\mathbf{a}_u^{(l)} = \left(0\cdots 0\ \frac{1}{\sqrt{m}}\cdots\frac{1}{\sqrt{m}}\ 0\cdots 0\right)'$, $\mathbf{a}_v^{(l)} = \left(0\cdots 0\ \frac{1}{\sqrt{n}}\cdots\frac{1}{\sqrt{n}}\ 0\cdots 0\right)'$ and $\mathbf{a}_u^{(l)}$, $\mathbf{a}_v^{(l)}$ have non-zero elements relative to two disjoint subsets of indexes C_u^l, C_v^l with $|C_u^l| = m$, $|C_v^l| = n$. Let $\delta_k = \frac{\sigma_e}{\sigma_k}$. We now consider two different cases. In the first case $C_u^l \subseteq F_i$ e $C_v^l \subseteq F_j$ $(i \neq j)$. Hence:

$$Y_u^{(l)} = \tfrac{m}{\sqrt{m}}\sigma_i^2 T_i + \tfrac{\sigma_e^2}{\sqrt{m}} E_i^{(l)}$$
$$Y_v^{(l)} = \tfrac{n}{\sqrt{n}}\sigma_j^2 T_j + \tfrac{\sigma_e^2}{\sqrt{n}} E_j^{(l)}.$$

Let $\sqrt{m}\sigma_i^2 + \widetilde{\sigma}_m^2 = \mathcal{V}(Y_u^{(l)})$ $(\widetilde{\sigma}_m^2 \leq \sqrt{m}\sigma_e^2)$ and $\sqrt{n}\sigma_i^2 + \widetilde{\sigma}_n^2 = \mathcal{V}(Y_v^{(l)})$ $(\widetilde{\sigma}_n^2 \leq \sqrt{n}\sigma_e^2)$. In this case, the distance covariance and distance correlation between $Y_u^{(l)}$ and $Y_v^{(l)}$ are, respectively:

$$\begin{aligned}
\mathcal{V}(Y_u^{(l)}, Y_v^{(l)}) &\leq \mathcal{V}(\tfrac{m}{\sqrt{m}}\sigma_i^2 T_i, \tfrac{n}{\sqrt{n}}\sigma_j^2 T_j) + \mathcal{V}(\tfrac{\sigma_e^2}{\sqrt{m}} E_i^{(l)}, \tfrac{\sigma_e^2}{\sqrt{n}} E_j^{(l)}) \leq \sqrt[4]{mn}\sigma_{ij} \\
\mathcal{R}(Y_u^{(l)}, Y_v^{(l)}) &= \tfrac{\mathcal{V}(Y_u^{(l)}, Y_v^{(l)})}{\sqrt{\mathcal{V}(Y_u^{(l)})\mathcal{V}(Y_v^{(l)})}} \leq \tfrac{\sqrt[4]{mn}\sigma_{ij}}{\sqrt{\sqrt{m}\sigma_i^2 + \widetilde{\sigma}_m^2}\sqrt{\sqrt{n}\sigma_j^2 + \widetilde{\sigma}_n^2}} = \\
&= \tfrac{\sqrt[4]{mn}\sigma_{ij}}{\sqrt[4]{mn}\sigma_i\sigma_j\sqrt{1 + \frac{\widetilde{\sigma}_m^2}{\sqrt{m}\sigma_i^2}}\sqrt{1 + \frac{\widetilde{\sigma}_n^2}{\sqrt{n}\sigma_j^2}}} \leq \tfrac{\sigma_{ij}}{\sigma_i\sigma_j}.
\end{aligned}$$

In the second case C_u^l, C_v^l are subsets of the same F_k. Hence

$$Y_u^{(l)} = \tfrac{m}{\sqrt{m}}\sigma_k^2 T_k + \tfrac{\sigma_e^2}{\sqrt{m}} E_{k1}^{(l)}$$
$$Y_v^{(l)} = \tfrac{n}{\sqrt{n}}\sigma_k^2 T_k + \tfrac{\sigma_e^2}{\sqrt{n}} E_{k2}^{(l)}.$$

Let $\sqrt[4]{mn}\sigma_k^2 c(\sigma_e) = \mathcal{V}(Y_u^{(l)}, Y_v^{(l)}) \leq \sqrt[4]{mn}\sigma_k^2$, with $c(\sigma_e)$ a constant such that $0 < c(\sigma_e) \leq 1$ and $c(\sigma_e) \overset{\sigma_e\to 0}{\longrightarrow} 1$. Furthermore $\mathcal{V}(Y_u^{(l)}) \leq \sqrt{m}(\sigma_k^2 + \sigma_e^2)$ and $\mathcal{V}(Y_v^{(l)}) \leq \sqrt{n}(\sigma_k^2 + \sigma_e^2)$. Therefore the distance correlation between $Y_u^{(l)}$ and $Y_v^{(l)}$ are, respectively:

$$\begin{aligned}
\mathcal{R}(Y_u^{(l)}, Y_v^{(l)}) &= \tfrac{\mathcal{V}(Y_u^{(l)}, Y_v^{(l)})}{\sqrt{\mathcal{V}(Y_u^{(l)})\mathcal{V}(Y_v^{(l)})}} \geq \tfrac{\sqrt[4]{mn}\sigma_k^2 c(\sigma_e)}{\sqrt{\sqrt{m}(\sigma_k^2 + \sigma_e^2)}\sqrt{\sqrt{n}(\sigma_k^2 + \sigma_e^2)}} \geq \\
&\geq \tfrac{\sqrt[4]{mn}\sigma_k^2 c(\sigma_e)}{\sqrt[4]{mn}\sigma_k^2\sqrt{\left(1+\delta_k^2\right)^2}} = \tfrac{c(\sigma_e)}{1+\delta_k^2}.
\end{aligned}$$

Since, from (1.4), the maximum distance correlation between variables belonging to different blocks is lower than the minimum distance correlation between variables belonging to the same block, aggregation involves variables relative to the same block and this proves the theorem.

Furthermore, if the noise variance is not too large, the K dimensional space that explains the most part of the variability is that spanned by the K basis elements related to the K blocks. Then the energy criterium identifies those elements. □

1.4 Comparison among PCA, ICA, Treelets, and HICA on synthetic data

In this section we will present some simulated examples to compare PCA, ICA, Treelets, and HICA performances in different scenarios. For all scenarios we consider the following latent variable model:

$$\mathbf{X} = \sum_{k=1}^{3} \mathbf{a}_k S_k + \sigma \mathbf{E} \ , \tag{1.5}$$

where $\mathbf{X}$ is the observed p-variate random vector, $\mathbf{a}_k$ represent the columns of the basis matrix A (i.e., the unknown basis elements), S_k are unobserved non-gaussian random variables, and $\mathbf{E}$ is a p-variate gaussian vector (with $\mathbf{0}$ mean and identity covariance matrix) acting as a noise term. Our purpose is to use PCA, ICA, Treelets, and HICA to obtain an estimate for the basis matrix A from a sample of size n drawn from model (1.5).

In detail, we investigate four different scenarios exploring different structures of dependence and orthogonality of the components (i.e., dependent/indepedent sources S_k and orthogonal/non-orthogonal basis elements $\mathbf{a}_k$):

Scenario A: Orthogonal and independent latent components.

Scenario B: Orthogonal and dependent latent components.

Scenario C: Non-orthogonal and independent latent components.

Scenario D: Non-orthogonal and dependent latent components.

Below, we focus on scenarios B, C, and D, respectively. Scenario A is not discussed since, as expected, all four methods are effective in estimating the model in this trivial case.

All the analyses are carried out with the statistical software R [53].

Scenario B: Orthogonal and dependent latent components. We first consider an example similar to the one presented in [29] where $p = 10$ random variables are obtained by linear combinations of three dependent - and thus

correlated - random sources such that the basis elements $\mathbf{a}_1, \mathbf{a}_2$, and $\mathbf{a}_3$ are non-overlapping - and thus orthogonal -. In particular we set:

$$S_1 \sim U([0, b_1]) \quad \perp\!\!\!\perp \quad S_2 \sim U([0, b_2]), \qquad S_3 = c_1 S_1 + c_2 S_2,$$

with $b_1 = 20$, $b_2 = 15$, $c_1 = 2$, $c_2 = 1$, and $\sigma = 1$. The basis elements $\mathbf{a}_k$ are defined on disjoint subsets, specifically:

$$\mathbf{a}_1 = (1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0)',$$
$$\mathbf{a}_2 = (0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0)',$$
$$\mathbf{a}_3 = (0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1)'.$$

Finally, we sample $n = 1000$ independent realizations from the model.

This is an example in which neither PCA nor ICA is expected to target the correct model being the three sources neither uncorrelated nor independent. On the contrary, both Treelets and HICA can detect the correct model if the chosen level of aggregation is $l = 7$ (i.e., 3 disjoint supports) and the chosen number of latent sources is $K = 3$. As shown in the bottom panels of Figure 1.1, this choice of l and K is the one suggested by the criterion presented in [29] and is among the ones suggested by the criterion suggested in Section 1.2. This latter criterion supports indeed $K = 3$ and $l = 7, 8, 9$ as candidate values.

Scenario C: Non-orthogonal and independent latent components. The previous example presents a situation in which hierarchical methods (i.e., Treelets and ICA) can outperform non-hierarchical methods (i.e., PCA and ICA). We now consider a complementary scenario in which ICA-inspired methods (i.e., ICA and HICA) can outperform PCA-inspired methods (i.e., PCA and Treelets). In this scenario $p = 6$, the basis elements $\mathbf{a}_1$ and $\mathbf{a}_2$ are overlapping and non-orthogonal and sources S_1, S_2, and S_3 are independent. In particular:

$$S_1 \sim U([0, b_1]) \quad \perp\!\!\!\perp \quad S_2 \sim U([0, b_2]) \quad \perp\!\!\!\perp \quad S_3 \sim U([0, b_3],$$

with $b_1 = b_2 = b_3 = 20$ and $\sigma = 1$. The basis elements $\mathbf{a}_k$ are defined as follows:

$$\mathbf{a}_1 = (1\ 1\ 0\ 0\ 0\ 0)',$$
$$\mathbf{a}_2 = (1\ 1\ 1\ 1\ 0\ 0)',$$
$$\mathbf{a}_3 = (0\ 0\ 0\ 0\ 1\ 1)'.$$

Finally, we sample $n = 1000$ independent realizations from the model.

Of course in this scenario, PCA and Treelets cannot target the right solution being the basis elements non-orthogonal. ICA instead targets the right solution being the sources independent. Figure 1.2 shows that also

HICA can detect the right solution if $K = 3$ and $l = 4$ (i.e., 2 disjoint supports).

Note that criterion proposed in [29] would have suggested $K = 3$ and $l = 3$ (i.e., 3 disjoint supports) which would have taken to a misidentification of the model for HICA as well (see top panels of Figure 1.2). This example confirms what suggested in our criterion: once K is chosen, all values of l providing the maximal energy are candidate values and not just the minimum one. Good representations are indeed obtained using HICA with $K = 3$ and $l = 4, 5$.

Scenario D: Non-orthogonal and dependent latent components We finally present a situation in which HICA outperforms PCA, ICA, and Treelets. This last scenario is simply obtained by setting latent components both non-orthogonal and dependent. In this case indeed, PCA cannot target the correct model being the sources non-orthogonal and dependent, ICA cannot target the correct model being the sources dependent, Treelets cannot target the correct model being the sources non-orthogonal. HICA remains the only method having the chance to target the correct model.

We here set $p = 6$, the basis elements $\mathbf{a}_1$ and $\mathbf{a}_2$ are overlapping (and thus non-orthogonal) and the three sources S_1, S_2, and S_3 dependent. In particular:

$$S_1 \sim U([0, b_1]) \quad \perp\!\!\!\perp \quad S_2 \sim U([0, b_2]) \quad S_3 = S_1 + S_2 + U,$$

and $U \sim U([0, b_3]$, with $b_1 = b_2 = 20$, $b_3 = 1$, and $\sigma = 1$, while basis elements $\mathbf{a}_k$ are the same defined as in Scenario C.

As shown in the bottom panels of Figure 1.3, we can draw the same conclusions of Scenario C with respect to the choice of K and l: $K = 3$ and $l = 3, 4, 5$ are good candidate values. Once again (top panels of Figure 1.3) $l = 3$ (the value suggested by the criterion proposed in [29]) is not the best choice. Although in this case, neither HICA is able to exactly catch the right configuration, HICA with $K = 3$ and $l = 4$ of course provides the closest representation: second and third components are very well detected with some bias in the estimation of the first component.

These simulated examples suggest that when dealing non-Gaussian latent components (even non-orthogonal and/or dependent) HICA always performs better than or equally to PCA, ICA, and Treelets. Moreover, as expected by theory, they discourage the use of PCA and ICA when components are dependent and the use of PCA and Treelets when components are non-orthogonal. A summary of the "win situations" for the four methods that can be drawn from the simulations is reported in Table 1.1.

Simulations also show that the criterion proposed in [29] for the choice of K and l might take to a misdetection of the model. For a given value of

K the more proper approach seems indeed to consider as candidate values for the level of aggregation l all values providing the maximal energy and not necessarily the minimum one.

	PCA	ICA	Treelets	HICA
A: Orthogonal and independent	win	win	win	win
B: Orthogonal and dependent			win	win
C: Non-orthogonal and independent		win		win
D: Non-orthogonal and dependent				win

Table 1.1: Summary of the "win situations" for PCA, ICA, Treelets, and HICA with respect to orthogonality/non-orthogonality and dependence/independence of the latent components.

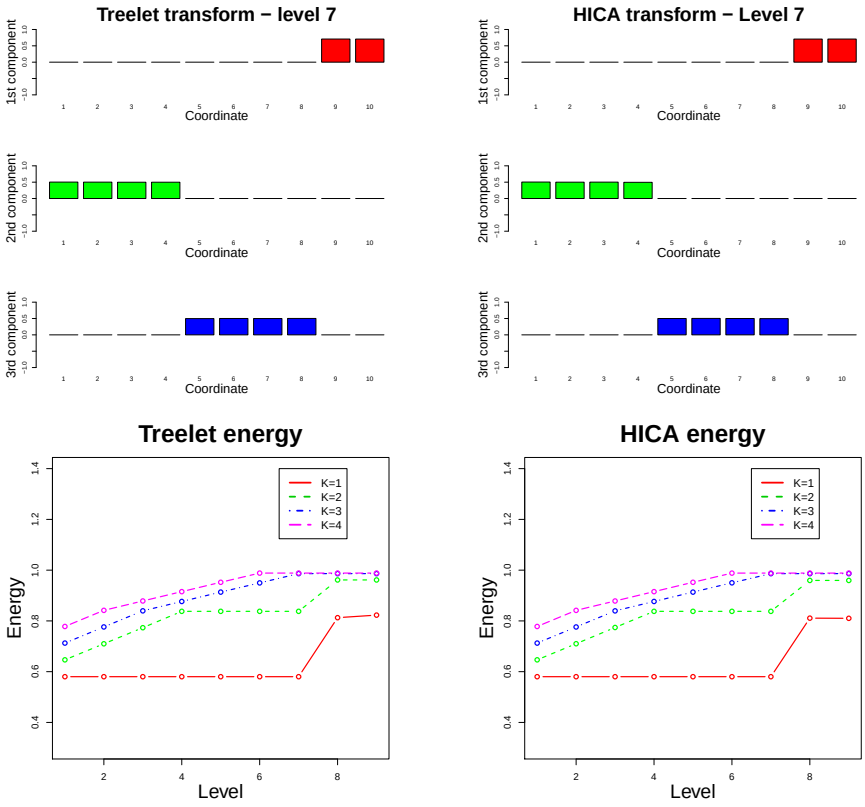

Figure 1.1: Scenario B: Orthogonal and dependent latent components. Top panels report the basis elements provided by Treelets (left) and HICA (right) when $K = 3$ and $l = 7$. The bottom panels report the energy as a function of l and K for Treelets decompositions (left) and HICA decompositions (right).

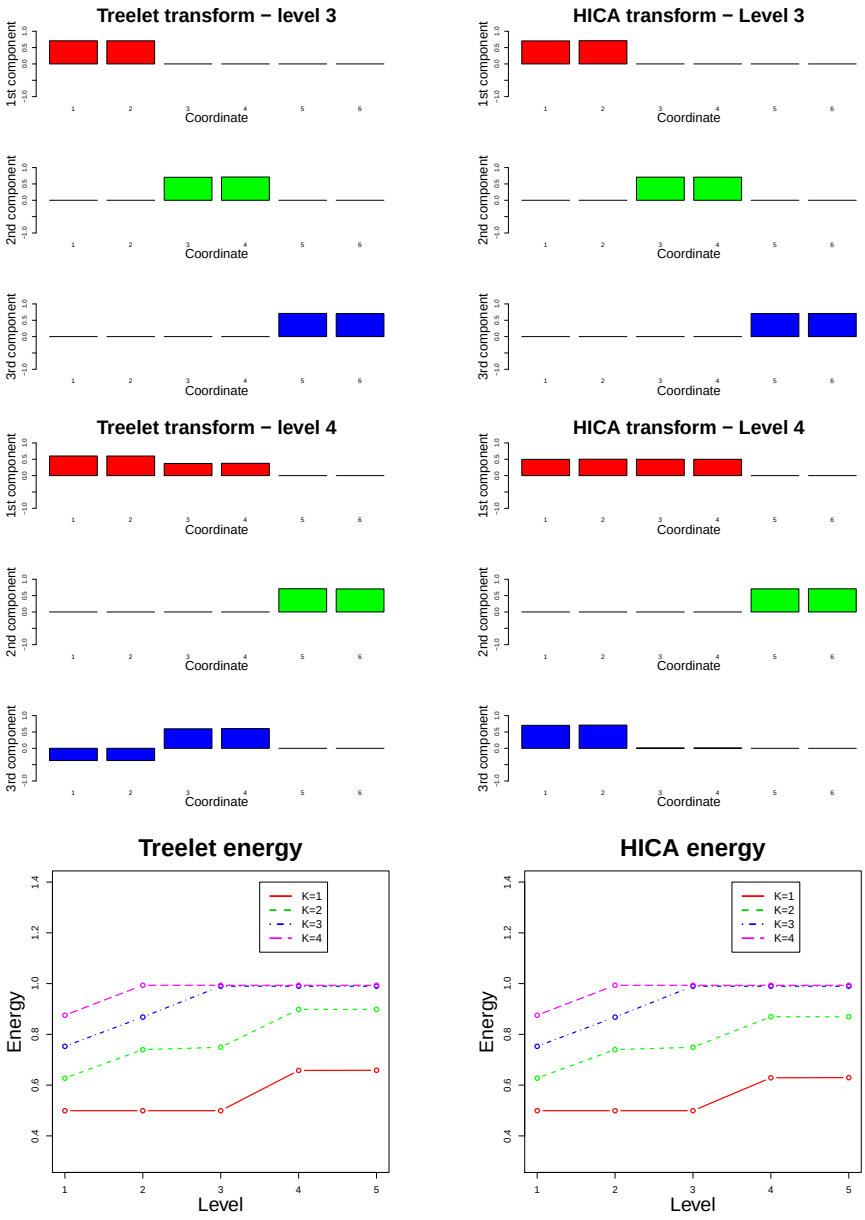

Figure 1.2: Scenario C: Non-orthogonal and independent latent components. Top panels report the basis elements provided by Treelets (left) and HICA (right) when $K = 3$ and $l = 3$. Middle panels report the basis elements provided by Treelets (left) and HICA (right) when $K = 3$ and $l = 4$. The bottom panels report the energy as a function of l and K for Treelets decompositions (left) and HICA decompositions (right).

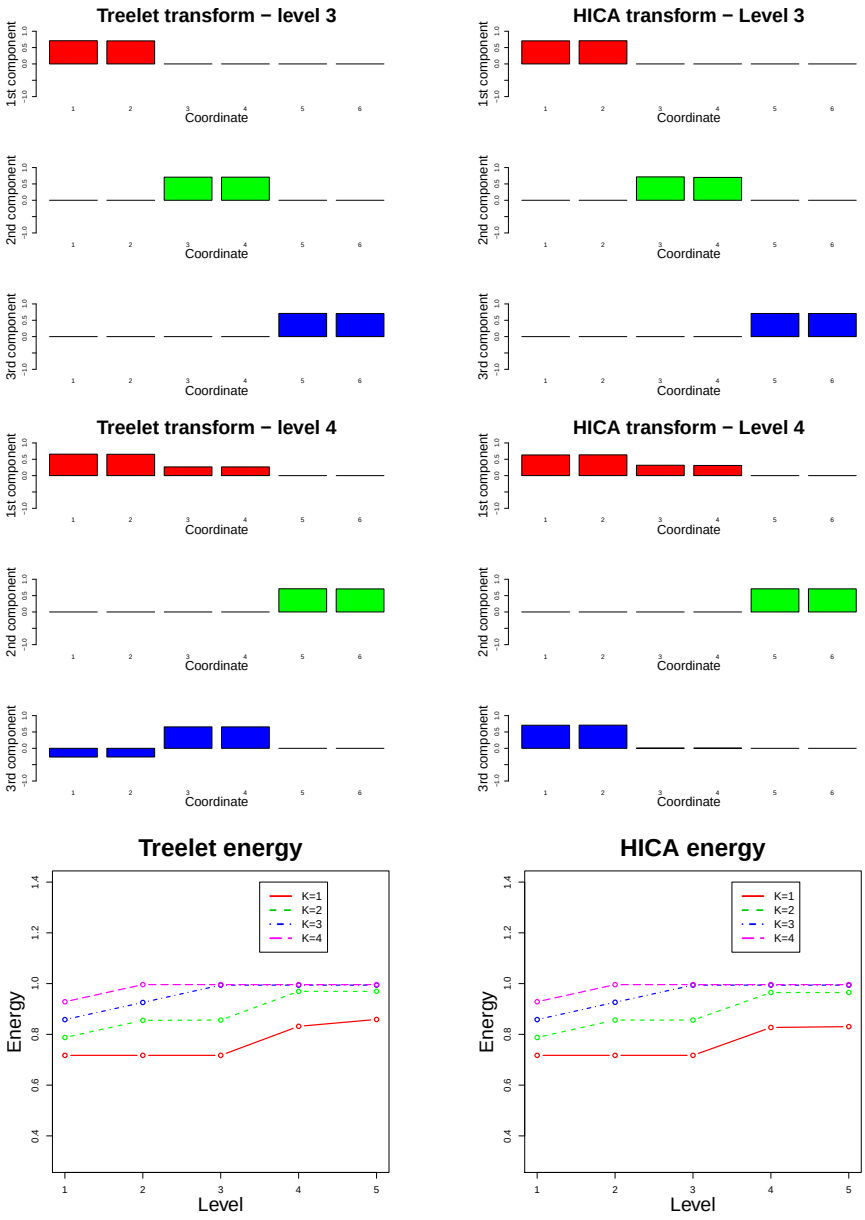

Figure 1.3: Scenario D: Non-orthogonal and dependent latent components. Top panels report the basis elements provided by Treelets (left) and HICA (right) when $K = 3$ and $l = 3$. Middle panels report the basis elements provided by Treelets (left) and HICA (right) when $K = 3$ and $l = 4$. The bottom panels report the energy as a function of l and K for Treelets decompositions (left) and HICA decompositions (right).

Chapter 2

Analysis of EEG signals

This chapter is dedicated to the analysis of a case study where HICA is applied to a BSS real data problem by analyzing EEG traces of patients affected by alcoholism. We compare HICA with other BSS techniques, specifically PCA, ICA and Treelets. We show how multi-resolution and non-orthogonal properties characterizing the HICA solution, allow to obtain interpretable and meaningful results that provide noticeable improvements in terms of phenomenological interpretation.

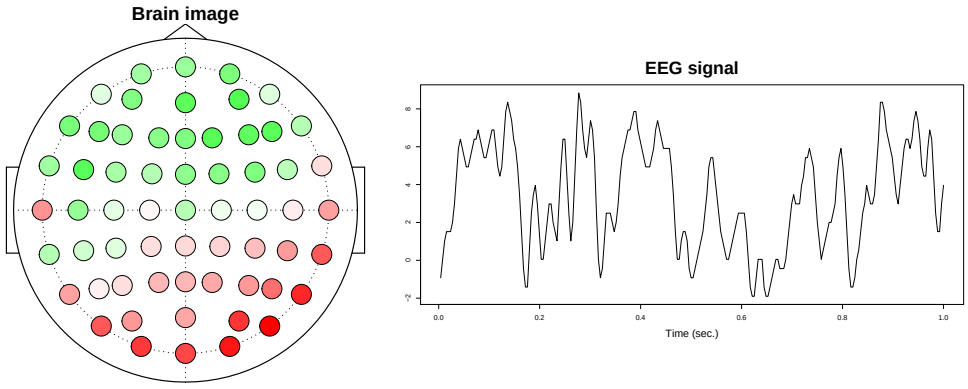

Figure 2.1: On the left: EEG brain signal at a fixed instant of time. On the right: EEG profile at a fixed electrode.

Data are courtesy of the online UCI Machine Learning Repository [8]. For each patient in the study, measurements from 61 electrodes out of 64 placed on the scalp are available. The electrodes are located at standard sites [44, 51]. For each electrode, the recorded signal measures the electrode electric potential with respect to some reference electrode and describes the electrical activity of the brain in the neighborhood of the electrode across time. In particular we analyze the brain signal related to one patient. The subject was exposed to two stimuli. Specifically, the patient was shown two

pictures chosen from the 1980 Snodgrass and Vanderwart set [46]. The two stimuli were presented in a matched condition (i.e., the subject has been asked to look at the same picture twice). For each electrode, we observe the signal at 256 equally spaced instants along a time span of 1 second. In Figure 2.1 the kind of data analyzed is shown. The analysis consists in the decomposition of the original variables through model (3). There are two different way to look at the data matrix $\mathbb{X}$. We can consider the instants of time as the observations. In this case we have a sample with $n = 256$ realizations of a random vector $\mathbf{X}$ in $\mathbb{R}^p$, where $p = 61$ is the number of electrodes studied. Otherwise we can look at electrodes as the observation. According this view $n = 61$ realizations of random vector of dimension $p = 256$ are observed. In literature there are some example of both these approaches. In particular, in the ICA framework, the first approach is called temporal ICA while the second one is named spatial ICA, because the sources are temporal and spatial processes, respectively. We follow the first approach, and we apply HICA method to the following model

$$x_{ij} = s_{i1}a_{j1} + ... + s_{iK}a_{jK},$$

where x_{ij} represents the signal at the ith time instant for the electrode j. According to this approach the columns of A contain the spatial maps of the brain and multi-resolution is a very interesting property for this data. In fact, some brain processes could involte the whole brain, while others activities involve only a specific part of brain. For this reason, multi-resolution methods seem particularly suitable for the analysis of this kind of data. Imposing multi-resolution property on the elements of A means that we could find a wide range of different behaviors characterizing the brain activity.

The rest of the chapter is organized as follow. In Section 2.1 we present the results obtained through the HICA algorithm described in Chapter 1. Then, in Section 2.2, we compare the results obtained through HICA with those provided by Treelets, ICA and PCA.

2.1 HICA analysis

In this section we present the results obtained applying HICA to the EEG dataset. As described in Section 1.2 we use the energy index in order to find a suitable K (dimension of the subspace) and l (level of the tree). In Figure 2.2 the energy for the last level $l = p - 1$ of the tree is shown. A precise value for K is not very clear (i.e, there is not an elbow). We chose $K = 5$ since it is sufficient to find interesting and meaningful results.

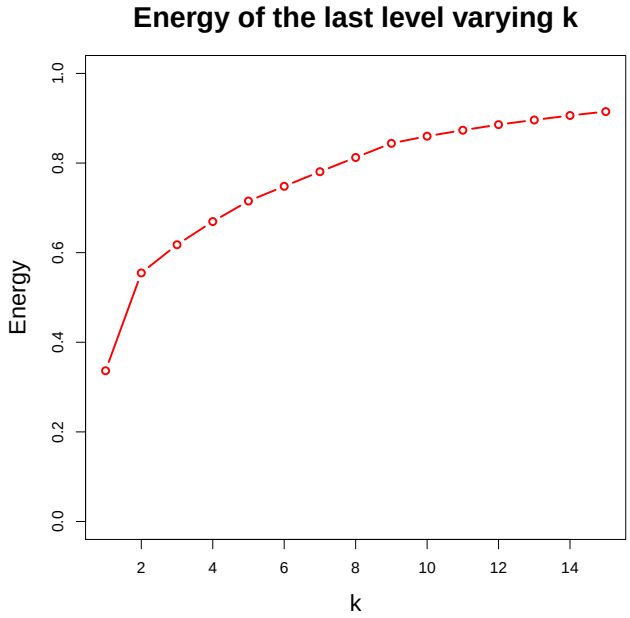

Figure 2.2: Energy for the last level $l = p - 1$ of the tree varying K.

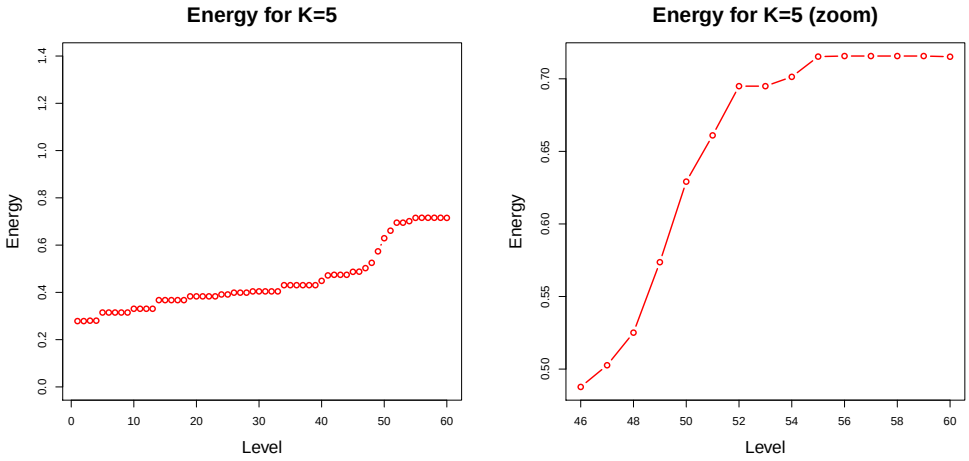

Figure 2.3: Energy for $K = 5$ varying the level l of the tree. In the panel on the left all the levels are shown. In the panel on the right a zoom of the last levels is depicted. The plateau is reached for $l = p - 6 = 55$.

Now we have tho chose a level l of the tree. In Figure 2.3 the energy for $K = 5$ varying l is depicted. From the panel on the right it is possible to see that the plateau is reached at $l = p - 6 = 55$. As we discussed in the Section 1.2, the choice between the levels in the plateau should be done

according the problem under study. In this case, for example, a basis with a higher degree of sparsity could be preferable, because some important brain activities involve a small part of the brain (i.e., a few variables). In Figure 2.4 we compare the lowest and the highest level of the plateau to see the main differences on the basis elements.

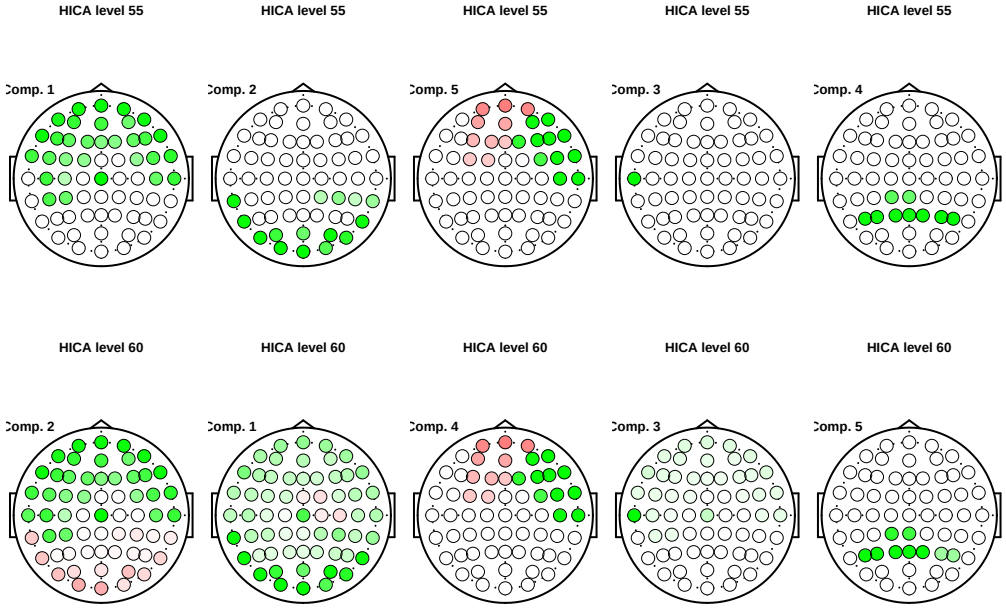

Figure 2.4: A comparison between the basis elements of level 55 (on the top) and level 60 (on the bottom).

Focusing on the first column of the figure, the component related to the lowest level identifies the associative activity in the frontal brain area, that is the area which processes the information related to similarities and differences between the two pictures. The corresponding component of the highest level, instead, present a contrast between the frontal and the occipital area, making the interpretation less clear. Another difference is highlighted by the second and the fifth column of Figure 2.4. The basis elements of level 55 are related to the occipital brain area. This information are split in two separate parts. The component shown in the second column is related to the primary visual cortex, the first area reached by visual information, which analyzes it in terms of shape and pattern recognition. Then the information flow goes to the internal area of the occipital hemisphere, highlighted by the component in the fifth column, which associates to the stimulus specific features like color, direction or origin. The level 60, instead, do not provide this separation. In particular, the second

component involve all the brain.

For these reasons a lower level seems to provide richer information. Hence we focus on level 55 for the comparison of HICA with other BBS popular techniques.

2.2 Comparison between HICA, Treelets, PCA and ICA

In this section we compare HICA with Treelets, PCA and ICA. We consider $K = 5$. For Treelets we select the level $l = 55$, as for HICA, in order to compare bases with the same degree of sparsity and, in both cases, we show the 5 components found by the energy criterium. Since $l = p - 6$, we expect to find basis elements whose supports are defined on no more than six different sets of variables. For PCA we show the first 5 principal components, while for ICA we present the results obtained exploiting the fastICA [21] algorithm selecting 5 sources. In Figure 2.5 we show some relevant basis elements identified by these methods for one patient.

Multi-resolution methods yield localized basis elements. This is a very interesting property, since it highlights components defined on localized brain regions and allows to identify more precisely the areas involved in the task. PCA and ICA, instead, yield more general and unspecific components, possibly difficult to read. Even when they seem to catch localized information, basis elements are not so clearly defined since they involve the entire set of variables. This is apparent in the fourth row of Figure 2.5, where HICA and Treelets select a single electrode (i.e., a single variable). This electrode clearly represents some noise either related to facial muscles activity or due to an unexpected saturation of the electrode. The related components identified by ICA and PCA, even though highlighting the same electrode, present more complex loadings diffused on other electrodes. The first row of Figure 2.5 reveals very similar components for HICA and and Treelets. Both analyses identify the associative activity in the frontal brain area, as described in Section 2.1. This crucial component is not caught by PCA and ICA. The main difference between HICA and Treelets regards instead the second and the fifth row in Figure 2.5. While Treelets yield an unfocused result, with components involving all the occipital cerebral hemisphere (i.e., one component averaging over the entire occipital part and the other contrasting the right and the left activity in the occipital part), HICA splits this information in two separate parts. As previously describes these different areas are responsible of specific activities, which only HICA is able to point out.

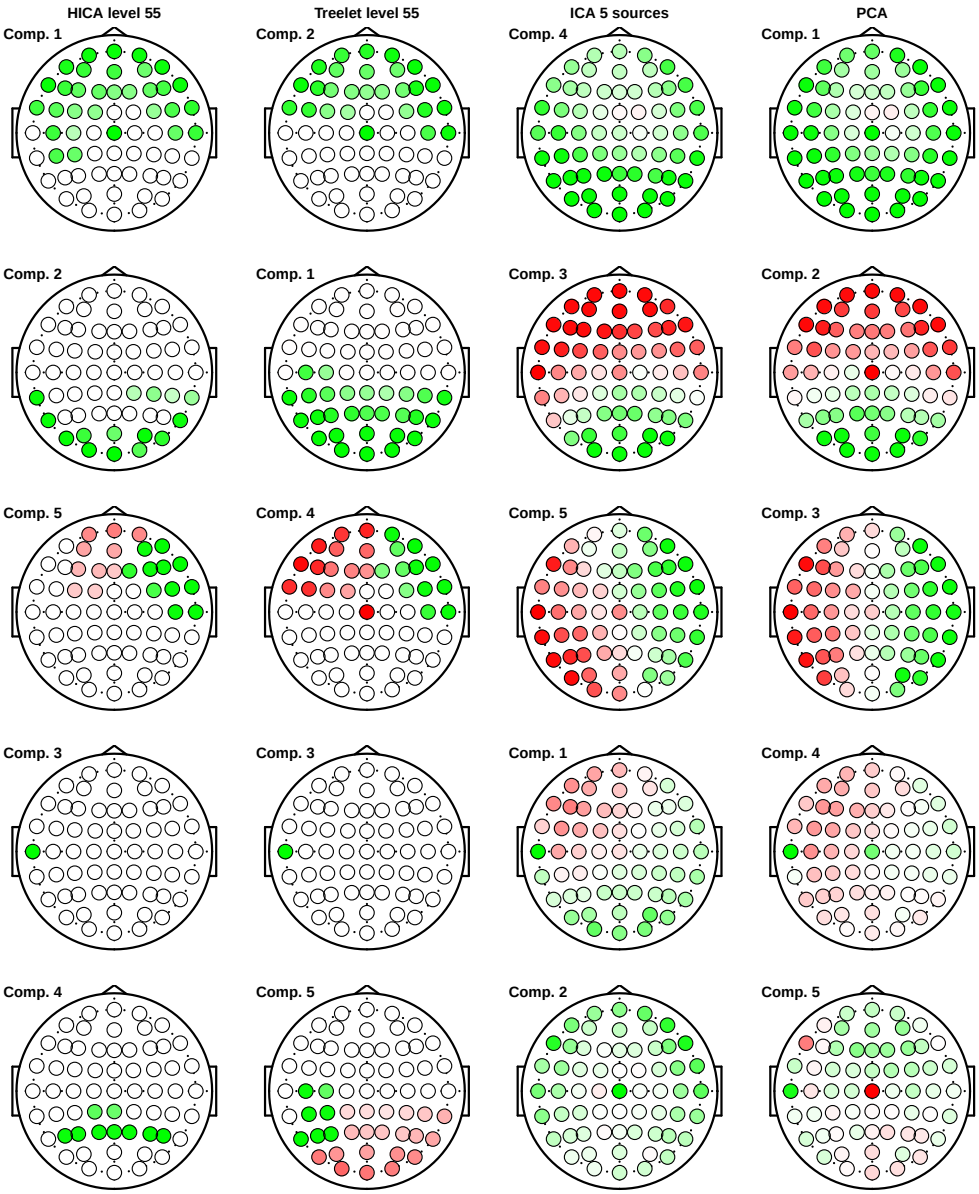

Figure 2.5: First five loadings found out by HICA (first column on the left), Treelets (second column), ICA (third column) and PCA (fourth column).

Part II

Spatial colored Independent Component Analysis

Chapter 3

Methods and simulations

Independent Component Analysis (ICA) is a data-driven method widely used to solve Blind Source Separation Problems [12]. We refer to the model (3) and we consider the case $K = p$. Hence, if we define the unmixing matrix $W = A^{-1}$, a BSS problem consists in estimating W, given $\mathbb{X}$, and recovering $\mathbb{S}$ through;

$$\mathbb{S} = \mathbb{X}W'$$

Being ICA a widespread approach for BSS problems a lot of methods have been developed to solve this problem. Two widely used techniques are infomax [9] and fastICA [21], where the unmixing matrix is estimated minimizing the mutual information (a measure of dependence) between the sources and this is equivalent to maximize the negentropy, a particular non-Gaussianity measure (indeed it is possible to show that the sources should not to be Gaussian distributed in order for the mixing matrix to be identifiable). These two algorithms rely only on the independence between the sources and try to estimate the marginal densities of the sources without any further assumption on the form of such densities.

Other methods, instead, make assumptions on the source densities. For example Independent Factor Analysis (IFA) [6, 35] models the independent components as mixtures of Gaussians, while Log-ICA and Lap-ICA [2] assume that the sources follow a Logistic and a Laplacian distribution, respectively.

All the above methods, while study the dependence structure between mixture variables trying to unmix the dependent signals in independent sources, do not exploit the possible correlation structure within the sources (and, then, within the mixtures). However in the real applications where ICA is commonly used, the signals are often autocorrelated, in time or space. For instance the typical framework where ICA has been introduced is the cocktail-party problem, where different microphones in a room register sounds produced by different sources. The goal here is to recover

the original audio signals through the time signals registered by the microphones. In this case a time correlation within the sources is present.

ICA is a method widely used also in the analysis of fMRI data [16, 34]. This kind of data consists in the registration of the brain activity in a certain number of cuboid elements, called voxels, for a period of time. Then both a spatial (between voxels) and temporal (between instants of time) dependence is present and there are two different approaches that can be applied. We can consider each spatial brain map at every instant of time a mixture of independent image components, or each temporal signal at every voxel a mixture of independent temporal sources. The former approach is called spatial ICA (sICA) while the latter is named temporal ICA (tICA) [34]. Some methods have been developed taking into account the correlation within the sources for the tICA approach. The method described in [38] is the first algorithm that considers the temporal autocorrelation of the sources, through the analysis of their spectral densities. However this method is based on the assumption that the spectral densities of the sources are known up to a scale parameter and this assumption is unrealistic in the real applications. Other methods, like AMUSE or TD-SEP algorithms (see [49, 52]), exploit the autocorrelation of the sources in the sense that they estimate the unmixing matrix W taking into account the independence between the sources at different lags. However they do not analyze the temporal structure within the single sources. Colored ICA (cICA) [30], instead, is an innovative procedure that takes into account the autocorrelation of the sources and it also works in the spectral domain, but in this case the knowledge of the spectral densities is not needed. Regarding the sICA approach in the literature there are no methods that involve the spatial autocorrelation of the sources in the evaluation of the independent components, imposing some stochastic spatial structure. In this chapter we provide a method to fill this lack.

Other spatiotemporal dataset that perfectly fit in the BSS framework are the geo-referred data, where the temporal changes of a certain quantity are measured on a specific geographic area. In particular we focus on the analysis of the Erlang, a dimensionless unit related to the mobile-phone network, in the urban area of Milan, Italy. sICA approach, in this case, is particularly interesting because allows to find out independent spatial maps related to different patterns that can be associated with specific activities within the city. The temporal profiles in the mixing matrix represent the temporal evolutions of such activities.

The chapter is then organized as follows. Firstly, in Section 3.1 we briefly present the spatial processes on lattices, introducing some simple models well known in literature and describing a non-parametric method to estimate the spectral density of spatial stochastic processes. Then, in Section 3.2 we describe in details the method and the algorithm we propose.

Since it extends the cICA method to the spatial case we call it spatial colored Independent Component Analysis (scICA). Finally, in Section 3.3 we present some simulations to validate scICA and to show the improvements due to take into account the spatial correlation within the sources.

3.1 Spatial models on lattices and their spectral representation

Let $\mathbf{s} \in \mathbb{R}^2$ be a generic location in a 2-dimensional Euclidean space and suppose that the potential datum $Z(\mathbf{s})$ at a spatial location $\mathbf{s}$ is a random quantity. If $\mathbf{s}$ varies over an index set $D \subseteq \mathbb{R}^2$, the spatial random field

$$\{Z(\mathbf{s}); \mathbf{s} \in D\} \tag{3.1}$$

is generated. A realization of (3.1) is denoted $\{z(\mathbf{s}); \mathbf{s} \in D\}$. We consider D a fixed regular collection of countably many points, say $D = \{\mathbf{s} = (u, v)' : u = ..., -1, 0, 1, ...; v = ..., -1, 0, 1, ...\}$. In this case (3.1) is called a spatial process on a lattice. We consider weakly-stationary processes, when the covariance $C(\mathbf{u})$ is defined, for every $\mathbf{u} \in \mathbb{Z}^2$, as

$$C(\mathbf{u}) = Cov(Z(\mathbf{s} + \mathbf{u}), Z(\mathbf{s})) \quad \forall\, \mathbf{s} \in D.$$

If the covariance values form an absolutely summable sequence, then we can define its Fourier Transform as:

$$f(\boldsymbol{\omega}) = \frac{1}{(2\pi)^2} \sum_{\mathbf{u} \in \mathbb{Z}^2} C(\mathbf{u}) e^{-i\mathbf{u}'\boldsymbol{\omega}}$$

with $(\omega_1, \omega_2)' = \boldsymbol{\omega} \in \Pi^2 = [-\pi, \pi] \times [-\pi, \pi]$. The function $f(\boldsymbol{\omega})$ is the spectral density of the stochastic process $Z(\mathbf{s})$. The covariance function at lag $\mathbf{u}$ can be recovered by the Inverse Fourier Transform of the spectral density as:

$$C(\mathbf{u}) = \int_{\Pi^2} f(\boldsymbol{\omega}) e^{i\mathbf{u}'\boldsymbol{\omega}} d\boldsymbol{\omega}.$$

Therefore covariance and spectral density form a Fourier pair (a detailed description of spatial stochastic processes and their properties can be found, for instance, in [13, 19]).

3.1.1 Spatial Autoregressive Moving-Average Models

We now introduce the class of Spatial Autoregressive Moving-Average (SARMA) models (see [19] for a complete description), considering the

following model for $Z(u,v)$:

$$\begin{aligned} Z(u,v) &= \sum_{j=-\infty}^{+\infty}\sum_{l=-\infty}^{+\infty}\phi_{jl}Z(u-j,v-l)+\epsilon(u,v) \\ \left(1-\sum_{j=-\infty}^{+\infty}\sum_{l=-\infty}^{+\infty}\phi_{jl}T_1^jT_2^l\right)Z(u,v) &= \epsilon(u,v) \\ \Phi(T_1,T_2)Z(u,v) &= \epsilon(u,v) \end{aligned} \tag{3.2}$$

where $\phi_{00}=0$, T_1 and T_2 are such that $T_1^{p_1}Z(u,v)=Z(u+p_1,v)$ and $T_2^{p_2}Z(u,v)=Z(u,v+p_2)$ and $\epsilon(u,v)$ is white noise with zero mean and variance σ^2. Model (3.2) is called Spatial Autoregressive (SAR) model. For example, if we consider a symmetric first-order model, $\Phi(T_1,T_2)$ reads:

$$\Phi(T_1,T_2)=1-\phi_1(T_1+T_1^{-1})-\phi_2(T_2+T_2^{-1}).$$

We take into account now a finite lattice with $n=n_1\cdot n_2$ sites $\{\mathbf{s}_1,...,\mathbf{s}_n\}$. We also define the random vector $\mathbf{Z}$ in $\mathbb{R}^n$ as $\mathbf{Z}=[Z(\mathbf{s}_1)\,Z(\mathbf{s}_2)\,...\,Z(\mathbf{s}_n)]'$ and the random vector $\boldsymbol{\epsilon}$ in $\mathbb{R}^n$ as $\boldsymbol{\epsilon}=[\epsilon(\mathbf{s}_1)\,\epsilon(\mathbf{s}_2)\,...\,\epsilon(\mathbf{s}_n)]'$, assuming that $\boldsymbol{\epsilon}$ is gaussian distributed with $\mathbf{0}$ mean and a (diagonal) covariance matrix Λ. Let $B=(b_{jl})$ be a matrix to be interpreted as the spatial-dependence matrix (the matrix of the coefficients ϕ_{jl}) with $b_{jj}=0$. Then, the SAR model for $\mathbf{Z}$ can be written as:

$$(I-B)\mathbf{Z}=\boldsymbol{\epsilon}.$$

Thus, it is easy to see that the distribution of $\boldsymbol{\epsilon}$ induces the distribution of $\mathbf{Z}$. Specifically:

$$\mathbf{Z}\sim N_n(\mathbf{0},(I-B)^{-1}\Lambda(I-B')^{-1}).$$

By the analogy with time-series models, it is possible to introduce Spatial Moving-Average (SMA) or Spatial Autoregressive Moving-Average (SARMA) processes:

$$\begin{aligned} Z(u,v) &= \Theta(T_1,T_2)\epsilon(u,v) \\ \Phi(T_1,T_2)Z(u,v) &= \Theta(T_1,T_2)\epsilon(u,v) \end{aligned}$$

where $\theta_{00}=1$. Defining $E=(e_{jl})$ the spatial dependence matrix of the coefficients θ_{jl} such that $e_{jj}=0$, the SMA model for $\mathbf{Z}$ can be written as:

$$\mathbf{Z}=(I-E)\boldsymbol{\epsilon}.$$

Hence

$$\mathbf{Z}\sim N_n(\mathbf{0},(I-E)\Lambda(I-E')).$$

The SARMA model for $\mathbf{Z}$

$$(I-B)\mathbf{Z}=(I-E)\boldsymbol{\epsilon}$$

provides

$$\mathbf{Z} \sim N_n(\mathbf{0}, (I-B)^{-1}(I-E)\Lambda(I-E')(I-B')^{-1}).$$

3.1.2 Spectral representation for SARMA models

Consider the general expression for SARMA model

$$\Phi(T_1, T_2)Z(u,v) = \Theta(T_1, T_2)\epsilon(u,v)$$

that can be reduce to the SAR or SMA model if $\Theta(T_1, T_2) = 1$ or $\Phi(T_1, T_2) = 1$ respectively.

It can be shown that the spectral density $f(\boldsymbol{\omega})$ of the stochastic process Z at a frequency $\boldsymbol{\omega} \in \Pi^2$ is given by:

$$f(\boldsymbol{\omega}) = \frac{\left|\sum_{j=-\infty}^{+\infty}\sum_{l=-\infty}^{+\infty}\theta_{jl}e^{-i(j,l)\cdot\boldsymbol{\omega}}\right|^2}{\left|1-\sum_{j=-\infty}^{+\infty}\sum_{l=-\infty}^{+\infty}\phi_{jl}e^{-i(j,l)\cdot\boldsymbol{\omega}}\right|^2}\frac{\sigma^2}{(2\pi)^2} = \frac{|A(\boldsymbol{\omega})|^2}{|B(\boldsymbol{\omega})|^2}f_\epsilon(\boldsymbol{\omega})$$

where $f_\epsilon(\boldsymbol{\omega}) = \sigma^2/(2\pi)^2 \; \forall\, \boldsymbol{\omega} \in \Pi^2$ is the spectral density of the white noise and $(j,l)\cdot\boldsymbol{\omega} = j\omega_1 + l\omega_2$.

3.1.3 Estimation of the spectral density based on Whittle log-likelihood

We now approach the problem of estimating the spectral density. In particular we focus on a non-parametric estimation of the spectral density based on Whittle log-likelihood [50]. For this reason we briefly introduce the spatial periodogram, an essential tool for the Whittle estimator.

The periodogram (also called sample spectral density) is a classical non-parametric estimator of the spectral density. For spatial process observed on a regular grid $D = \{\mathbf{s} = (s_1, s_2) : s_1 = 0, ..., n_1 - 1; s_2 = 0, ..., n_2 - 1\}$, $D \in \mathbb{R}^2$, $n = n_1 \cdot n_2$, the spatial periodogram at a frequency $\boldsymbol{\omega} \in \Pi^2$ is given by:

$$I(\boldsymbol{\omega}) = \frac{1}{(2\pi)^2 n}\left|\sum_{\mathbf{s}\in D} Z(\mathbf{s})\exp(-i\mathbf{s}'\boldsymbol{\omega})\right|^2.$$

The periodogram is usually computed at the set of bidimensional Fourier frequencies $\boldsymbol{\omega}_k = (\omega_{k_1}, \omega_{k_2})$:

$$\omega_{k_1} = \frac{2\pi k_1}{n_1} \quad k_1 = 0, \pm 1, ..., \pm m_1 \quad \text{where} \quad m_1 = \lceil \frac{(n_1 - 1)}{2} \rceil$$

$$\omega_{k_2} = \frac{2\pi k_2}{n_2} \quad k_2 = 0, \pm 1, ..., \pm m_2 \quad \text{where} \quad m_2 = \lceil \frac{(n_2 - 1)}{2} \rceil.$$

If we define the Discrete Fourier Transform of the data as:

$$J(\boldsymbol{\omega}) = \frac{1}{2\pi\sqrt{n}} \sum_{\mathbf{s} \in D} Z(\mathbf{s}) e^{-i\mathbf{s}'\boldsymbol{\omega}},$$

then the periodogram can be obtained as:

$$I(\boldsymbol{\omega}) = J(\boldsymbol{\omega})\overline{J(\boldsymbol{\omega})} = |J(\boldsymbol{\omega})|^2.$$

The spatial periodogram is an asymptotically unbiased estimator of the spectral density, but it is not consistent, since the variance is proportional to the square of the spectral density at each frequency. Nevertheless, the periodogram values at different frequencies are asymptotically uncorrelated [18]. To avoid this inconsistency problem one of the most popular method in the spectral parametric context is the Whittle estimation, based on an approximation to the Gaussian negative log-likelihood, and it uses the periodogram as a pilot estimate. For a parametric model of the spectral density f_θ, with $\theta \in \Theta \in \mathbb{R}^p$, the Whittle parameter estimator $\widehat{\theta}$ is given by:

$$\widehat{\theta} = \arg\min_{\theta} L(\theta, I),$$

where $L(\theta, I)$ denotes the Whittle log-likelihood

$$L(\theta, I) = \int_{\Pi^2} \left(\log f_\theta(\boldsymbol{\omega}) + \frac{I(\boldsymbol{\omega})}{f_\theta(\boldsymbol{\omega})} \right) d\boldsymbol{\omega}. \tag{3.3}$$

The log-likelihood (3.3) can be interpreted as the Kullback-Leibler divergence between I and f_θ. Note that, in practice, (3.3) is approximated by a discretized version:

$$\sum_k \left(\log f_\theta(\boldsymbol{\omega}_k) + \frac{I(\boldsymbol{\omega}_k)}{f_\theta(bo_k)} \right) \tag{3.4}$$

where the sum extends over all Fourier frequencies.

Based on the discrete approximation (3.4), it is possible to obtain a nonparametric estimator for the log-spectral density $m_\theta = \log f_\theta$ [14, 17]. It is easy to see that, minimizing (3.4) is equivalent to maximize in θ

$$\sum_k \left(Y_k - m_\theta(\boldsymbol{\omega}_k) - e^{Y_k - m_\theta(\boldsymbol{\omega}_k)} \right)$$

where Y_k denotes the log-periodogram value at the Fourier frequency $\boldsymbol{\omega}_k$. We consider the estimator obtained for the log-spectral density function m_θ by a multidimensional local linear kernel estimator. For any $\mathbf{x} \in \mathbb{R}^2$,

we approximate $m_\theta(\boldsymbol{\omega}_k)$ by the plane $a + \mathbf{b}'(\boldsymbol{\omega}_k - \mathbf{x})$. Then, we construct the local likelihood function

$$\sum_k \left(Y_k - a - \mathbf{b}'(\boldsymbol{\omega}_k - \mathbf{x}) - e^{Y_k - a - \mathbf{b}'(\boldsymbol{\omega}_k - \mathbf{x})} \right) K_H(\boldsymbol{\omega}_k - \mathbf{x}), \tag{3.5}$$

where the function K_H is a reescaled bidimensional kernel, H is a bidimensional bandwidth matrix and $K_H(\mathbf{x}) = |H|^{-1/2} K(H^{-1/2}\mathbf{x})$. The local maximum likelihood estimator $\widehat{m}_{LK}(H, \mathbf{x}) \equiv \widehat{m}_{LK}(\mathbf{x})$ of $m(\mathbf{x})$ is $\widehat{a}$ in the maximizer $(\widehat{a}, \widehat{b})$ of (3.5).

3.2 Spatial colored Independent Component Analysis

We consider now the BSS problem (3) assuming the sources to be spatial processes defined on a finite lattice D with n sites. Let $\mathbf{S} = (S_1, ..., S_p)'$ be a random vector in $\mathbb{R}^p$. We can define the spectral density and the periodogram of the jth source as $f_{S_j}(\boldsymbol{\omega})$ and $I(\boldsymbol{\omega}, S_j)$ respectively. Then the sources Whittle log-likelihood is given by

$$L(f_{\mathbf{S}}; \mathbf{S}) = \sum_{j=1}^{p} \sum_{k=1}^{n} \left(\frac{I(\boldsymbol{\omega}_k, S_j)}{f_{S_j}(\boldsymbol{\omega}_k)} + \ln(f_{S_j}(\boldsymbol{\omega}_k)) \right) \tag{3.6}$$

where $f_{\mathbf{S}}$ is the diagonal spectral density matrix of the sources (diagonal because the sources are assumed independent). In practice we do not observe the sources, but we observed the mixed spatial processes. So the log-likelihood (3.6) can be rewritten as

$$L(W, f_{\mathbf{S}}; \mathbf{X}) = \sum_{j=1}^{p} \sum_{k=1}^{n} \left(\frac{\mathbf{e}_j' W' I(\boldsymbol{\omega}_k, \mathbf{X}) W \mathbf{e}_j}{f_{S_j}(\boldsymbol{\omega}_k)} + \ln(f_{S_j}(\boldsymbol{\omega}_k)) \right) + n \ln |det(W)| \tag{3.7}$$

where $I(\boldsymbol{\omega}_k, \mathbf{X})$ is the matrix periodogram of the mixed signals at the Fourier frequency $\boldsymbol{\omega}_k$ and $\mathbf{e}_j = (0, ..., 0, 1, 0, ..., 0)'$ with the jth entry being 1. Then we basically need to estimate both the unmixing matrix W and the sources spectral density f_{S_j} for $j = 1, ..., p$. Therefore we implement an iterative algorithm, alternating a step where sources spectral densities are estimated with a step where an estimate $\widehat{W}$ of W is obtained. The iterative algorithm stops when the difference between $\widehat{W}_{new}$ and $\widehat{W}_{old}$ is under a convergence threshold, where the difference is measured with the Amari error (see [4] for details), a criterion widely used in ICA framework.

3.2.1 The iterative algorithm

Firstly we imagine the unmixing matrix W to be fixed. Then, the log-periodogram $Y(\boldsymbol{\omega}_k, S_j)$ can be easily evaluated for every $j = 1, ..., p$ and every $k = 1, ..., n$. Hence, for every $j = 1, ..., p$, the spectral density f_{S_j} can be estimated through the nonparametric method (3.5). In the cICA algorithm [30], differently, the parameter of the spectral density of the temporal sources are estimated through a parametric procedure and then the spectral densities are evaluated. In our framework, where sources are assumed to be spatial stochastic processes, a parametric approach could be to difficult to deal with, because of the nontrivial way to choose the order of the autoregressive and moving-average parts in SARMA models, and too restrictive, because of the very different features that spatial sources could present in real application. Nonparametric approach, although computationally expensive, allows us to take into account very different structures for the sources.

We now fix f_{S_j} for $j = 1, ..., p$. A typical procedure in ICA methods is to prewhite data [22]. In this way W is orthogonal and this allows us to drop the last term in (3.7). However we need to impose an orthogonality constraint on the unmixing matrix. Then, for every $j = 1, ..., p$, we minimize

$$\widetilde{L}(W, f_{\mathbf{S}}; \mathbf{X}) = \mathbf{w}_j'(A_k + \tau C_j)\mathbf{w}_j \tag{3.8}$$

where $\mathbf{w}_j = W\mathbf{e}_j$ is the jth column of W, $A_k = \sum_{k=1}^{n} \frac{I(\boldsymbol{\omega}_k, \mathbf{X})}{f_{S_j}(\boldsymbol{\omega}_w)}$, $C_j = \sum_{k \neq j} \mathbf{w}_k \mathbf{w}_k'$ and τ is a positive tuning parameter. Matrix C_j provides an orthogonality constraint in the sense that $\mathbf{w}_j' C_j \mathbf{w}_j = \sum_{k \neq j} \langle \mathbf{w}_j, \mathbf{w}_k \rangle^2$. This representation provides a straightforward estimate for $\mathbf{w}_j$. Indeed it is to see that $(A_k + \tau C_j)$ is symmetric and positive-definite. Hence the argmin of (3.8) is the eigenvector of $(A_k + \tau C_j)$ corresponding to the lowest eigenvalue. However, the problem of setting the tuning parameter τ still remains. It is important to point out that orthogonality has to be a constraint and not simply a penalization. For this reason we set an initial (small) value for τ and then we proceed in an alternating way as follows:

a) we obtain $\widehat{W}$ from (3.8);

b) if the orthogonality error is under a certain threshold, we remain with this estimate for W. Unless we repeat the step a) setting $\tau = 2\tau$.

The orthogonality error is measured by $\|\widehat{W}\widehat{W}' - I\|_F$, with $\|\cdot\|_F$ being the Frobenius norm.

We can finally summarize the iterative algorithm discuss in this section. Firstly we initialize $\widehat{W}$. Then, while the Amari error is greater than a certain threshold, we repeat the following steps:

1) we estimate the sources spectral density through the nonparametric algorithm (3.5);

2) we update $\widehat{W}$ according the minimization of (3.8), using the rule described above to impose the orthogonality constraint.

Remark 3.1 *Another possibility to involve the orthogonality constraint is to use the Newton-Raphson method with Lagrange multiplier as presented in [30]. However in the framework analyzed in this paper, the nonparametric estimate of the spectral density could lead to bad conditioned Hessian matrix in the Newton-Raphson update. For this reason we prefer to estimate the unmixing matrix W through the criterium (3.8). In any case we point out that, in those situation where the Hessian matrix does not present bad conditioning problems, the results of the two approaches do not show relevant differences.*

Remark 3.2 *We presented here the particular case when $K = p$. To consider $K < p$ a typical procedure adopted in ICA method is to project data in the K-dimensional space identified by the first K principal direction. Then proceed with the estimate of the unmixing and of the mixing matrix in this space and finally recover the original mixing matrix by the first transformation.*

3.3 Simulation study

In this section we present some simulation studies, comparing the results obtained by scICA with those obtained by cICA and fastICA (the most popular ICA algorithm). To perform cICA algorithm, we vectorize the 2D processes and we consider them as 1D processes. We make this in order to compare cICA with scICA and to evaluate if taking into account the 2D dependence gives significative improvements with respect to consider the dependence only in one direction. fastICA algorithm, instead, is uses as a benchmark algorithm to implement ICA, since it is the most widespread method used in the literature. All simulations are carried out on a $n_1 \times n_2$ grid, with $n_1 = n_2 = 20$.

3.3.1 First simulation study: symmetric SARMA processes of the first order

We consider for this simulation two sources and two mixtures. We perform 100 different runs and for each run the mixing matrix C is generated randomly. The first source is generated according the following symmetric

SAR model of the first order:

$$Z(u,v) = \phi_1(Z(u-1,v)+Z(u+1,v))+\phi_2(Z(u,v-1)+Z(u,v+1))+\epsilon(u,v) \tag{3.9}$$

with $\phi_1 = 0.3$, $\phi_2 = 0.4$ and $\epsilon(u,v)$ a gaussian noise with zero mean and variance $\sigma^2 = 0.3^2$. The second source is generated according the following SMA model of the first order:

$$Z(u,v) = \epsilon(u,v) + \theta_1(\epsilon(u-1,v) + \epsilon(u+1,v)) + \theta_2(\epsilon(u,v-1) + \epsilon(u,v+1)) \tag{3.10}$$

with $\phi_1 = 0.25$, $\phi_2 = 0.3$ and $\epsilon(u,v)$ a gaussian noise with zero mean and variance $\sigma^2 = 0.3^2$. Then, data matrix $\mathbb{X}$ is generated according to the model (3). In the left panel of Figure 3.1 the boxplots of the Amari errors for every method considered are shown. Both scICA and cICA significantly outperform fastICA algorithm. The two colored methods seem comparable. However, if we consider the differences between the two errors for every run, we can observe that scICA is significantly better. In the right panel of Figure 3.1 the boxplot of the differences is depicted. Furthermore we report the p-value of the test to verify if the mean of the difference can be considered less than zero. The p-value is very low, equal to 0.00304, providing statistical evidence to reject the null hypothesis.

In BSS problems we are not interested only in a good estimate of the mixing matrix, but we also aim to reconstruct efficiently the sources. For this reason we evaluate for each run the error in estimating the sources, considering the mean of the errors over the 20×20 lattice. In Figure 3.2 we show the differences of the error over the 100 runs between scICA and cICA algorithm, both for the first and the second source. We also depict the p-value to verify if the mean of the differences could be considered lower than zero. We can observe that for the second source the p-value is around 0.05, providing us slight evidence to reject the null hypothesis, while for the first source the evidence is substantially stronger.

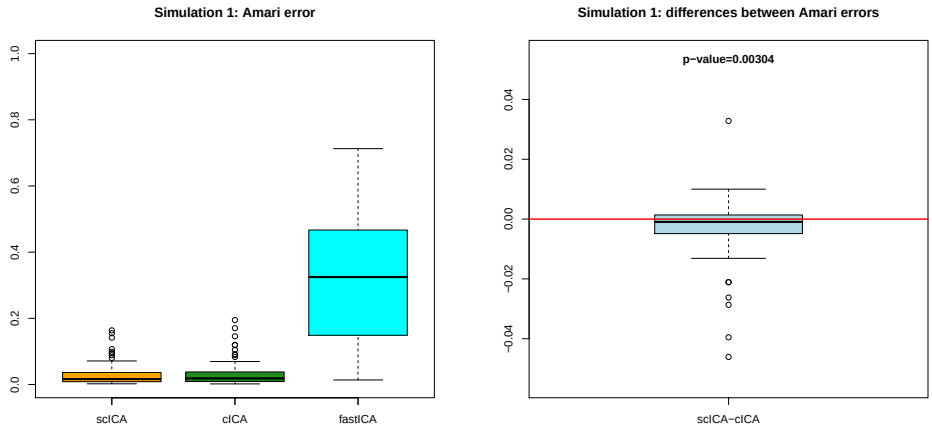

Figure 3.1: Simulation 1 - On the left panel: boxplot of the Amari error for the three methods considered. On the right panel: boxplot of the differences between scICA and cICA Amari error. The p-value to test if the mean of the difference can be considered lower than zero is shown above the boxplot.

Figure 3.2: Simulation 1 - On the left panel: boxplot of the differences of the errors between scICA and cICA algorithm in estimating the first source. The p-value to test if the mean of the difference can be considered lower than zero is shown above the boxplot. On the right panel: boxplot of the differences of the errors between scICA and cICA algorithm in estimating the second source. The p-value to test if the mean of the difference can be considered lower than zero is shown above the boxplot.

3.3.2 Second simulation study: spatial sources with irregular structure

We now take into account two sources, say S_1 and S_2 created artificially and showed in Figure 3.3.

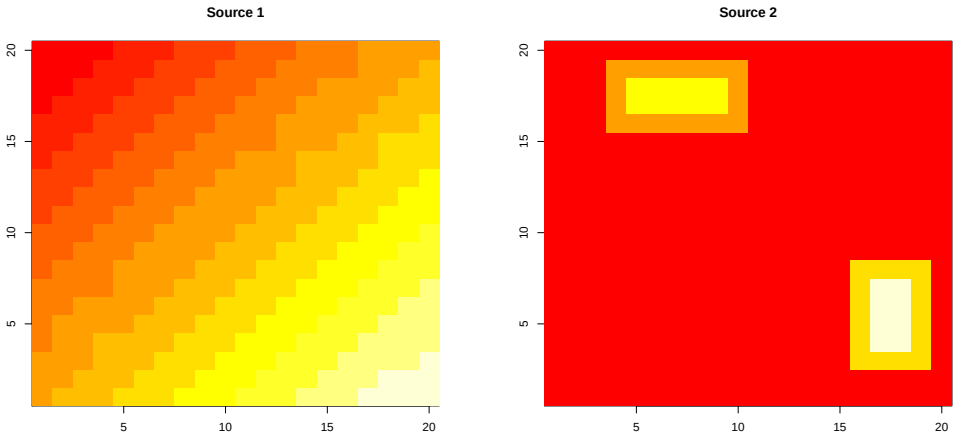

Figure 3.3: The two sources considered in the second simulation.

We perform 100 different runs, generating the mixing matrix randomly at each run and the data matrix according to the model (3), where the sources matrix is composed by the sources of Figure 3.3 plus some gaussian noise with zero mean and different variances for the two sources. Specifically $\sigma_1^2 = 2^2$ and $\sigma_2^2 = 0.1^2$.

The boxplots of the Amari errors for every method considered are depicted in the left panel of Figure 3.4. The two colored methods clearly outperform fastICA algorithm, as well as in the first simulation. Furthermore, in this case the improvements due to take into account the spatial structure of the sources seem even more evident. Indeed the p-value is significantly lower, as shown in the right panel of Figure 3.4. Comparing the estimate of the sources for the two colored method, is evident how scICA strongly outperform cICA, as highlighted by the extremely low p-values in Figure 3.5.

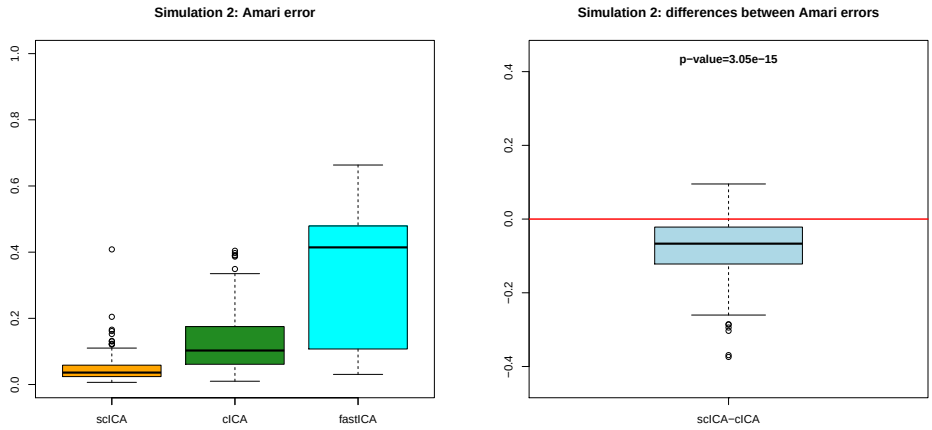

Figure 3.4: Simulation 2 - On the left panel: boxplot of the Amari error for the three methods considered. On the right panel: boxplot of the differences between scICA and cICA Amari error. The p-value to test if the mean of the difference can be considered lower than zero is shown above the boxplot.

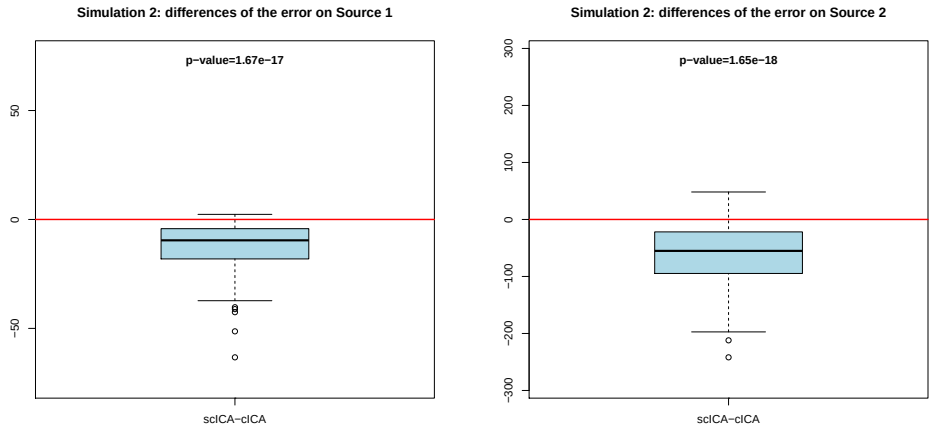

Figure 3.5: Simulation 2 - On the left panel: boxplot of the differences of the errors between scICA and cICA algorithm in estimating the first source. The p-value to test if the mean of the difference can be considered lower than zero is shown above the boxplot. On the right panel: boxplot of the differences of the errors between scICA and cICA algorithm in estimating the second source. The p-value to test if the mean of the difference can be considered lower than zero is shown above the boxplot.

Chapter 4

Analysis of Telecom data

In this chapter we analyze a mobile-phone traffic dataset, related to the metropolitan area of Milan (Italy), through different BSS techniques. In particular we focus on the Independent Component Analysis, comparing the well known fastICA algorithm (see e.g., [21] with the algorithm scICA we proposed in Chapter 3. Then we apply also the HICA algorithm described in Chapter 1, in order to analyze the behaviors of these different approaches. The first, ICA, addressed on the source matrix $\mathbb{S}$, while the second, HICA, on the basis matrix A.

The metropolitan area of Milan, located in the North of Italy is the fifth biggest metropolitan area of the entire Europe in terms of number of inhabitants. As all the large metropolitan areas, it is characterized by a consistent presence of working and residential/leisure activities. Indeed, the urban area of Milan provides nearly the 10% of the Italian gross domestic product and it is the most populated province of the country, with a density of more than 1000 inhabitants per km^2. An Organization foe Economic Co-operation and Development (OECD) review of 2006 (see [36] for the complete report) identified housing, transport and congestion as the principal limitation for the future development of the area. In particular most of the principal roads connecting the city of Milan with its suburbs have reached their saturation during the rushing hours. These aspects cause a lot of problems, above all in terms of pollution and economy. Although in recent years something has been done to decrease the congestion stimulating the use of different means of transport, like the public transports or car and bike-sharing systems, a deep analysis of the main features regarding working, residential and mobility activities is crucial for the well-being of the city. Indeed, as highlighted in [25] and [45], changes in management of mobility are a key point to understand times, places and modes of social life, thus structuring the urban areas. Traditional data sources for mobility and urban investigation are, for example, surveys or census. However these sources present a lot of limitations. Specifically they

are characterized by high costs or difficulty of data updating. Furthermore these kind of data are suitable to represent and to infer about static features of the urban life, but they less appropriate when the focus is on city dynamics and time/spatial dependent variations in intensity of urban spaces usages at different scales. Mobile phone network data are potentially an interesting tool in this direction, for the real-time monitoring of the urban dynamics. Indeed they have been widely analyzed in several experimental studies (see e.g, [40, 1, 20]). Since these studies are quite qualitative, our aim is to analyze this kind of data through suitable statistical methods. In particular these datasets perfectly fit in the BSS framework. Hence we eant to apply the methods described in this manuscript in order to retrieve meaningful and useful information for urban planning.

The rest of the chapter is organized as follows. In Section 4.1 we present the dataset and the pre-processing procedures adopted. Then, in Sections 4.2 and 4.3 we present the results obtained. Firstly following an ICA approach, comparing fastICA with scICA algorithm. Secondly following a multi-resolution approach applying the HICA algorithm. Finally, in Section 4.4, we summarize the results obtained with the different methods.

4.1 Telecom dataset: description and pre-processing

The dataset we analyze describes the mobile phone traffic on the metropolitan area of Milan. Data are courtesy of Telecom Italia, the biggest mobile phone Italian company, thanks to a research agreement between Telecom and the Politecnico di Milano. Telephone traffic is anonymously recorded as the average number of simultaneous contacts in a time unit. Then, Telecom elaborates these measurements by means a weighted interpolation, thus obtaining an evaluation of the phone traffic on a tessellation of the territory in rectangular areas (i.e., pixels). In the database, the metropolitan area of Milan is dividend into a lattice L_0 of 97×109 pixels (232m $\times$ 309m each). In order to make the analysis computationally faster we focus on a zoom on the municipality of Milan of 25×28 pixels. For each pixel of the covered area we observe the Erlang every 15 minutes for 14 days. The Erlang is a dimensionless unit calculated as the sum of the length of every call in a given time interval divided by the length of the interval (i.e., 15 minutes). For each pixel and for each quarter of an hour, this measure represents the average number of mobile phones simultaneously calling through the network, that, as a first approximation, can be considered proportional to the number of active people in that area at that time. The Erlang x_{ij} related to the pixel $l_i \in L_0$ and to the time interval

t_j (i.e., the jth quarter of a hour) is evaluated as

$$x_{ij} = \sum_{r=1}^{R} T_{ij}^r$$

where T_{ij}^r indicates the length in minutes of the time interval (or union of intervals) in which the rth mobile phone is calling while moving in the pixel l_i during the time interval t_j. R indicates the total number of potential network users. Hence these data describe a phenomenon in a 2D-space at different instants of time. This may be represented by a surface varying along time, as depicted in Figure 4.1.

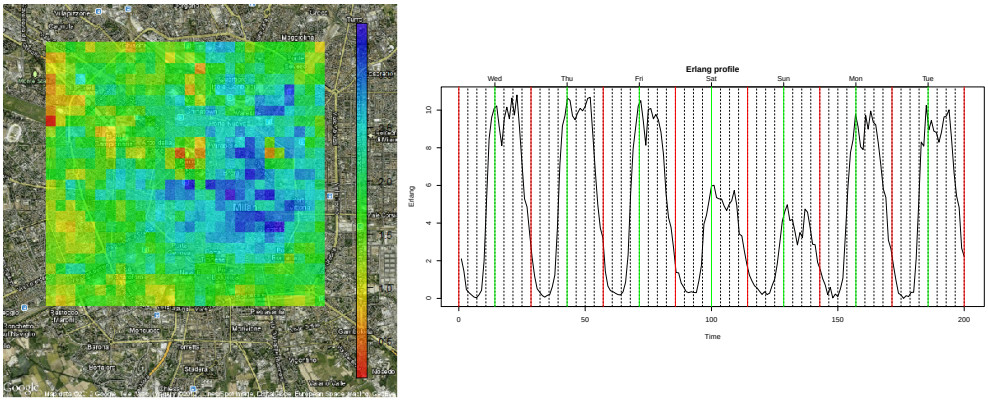

Figure 4.1: On the left: Erlang distribution on the lattice at a fixed instant of time. On the right: Erlang profile at a fixed pixel.

Aim of the analysis is to decompose the observed signal as a time-varying linear combination of a reduced number, say K, of time-invariant source surfaces. Specifically, for a fixed pixel l_i and a fixed time interval t_j:

$$x_{ij} = s_{i1}a_{j1} + ... + s_{iK}a_{jK},$$

where s_{ik} represents the contribution of the kth source in the pixel l_j and a_{jk} is the intensity of the kth source at the jth time interval. This problem fits in the BSS framework, indeed the purpose of the analysis is to represent $\mathbb{X}$ as the product of two matrices, a p x K matrix A and a n x K matrix $\mathbb{S}$, where each column of $\mathbb{S}$ represents the evaluation at the n pixels of the corresponding source surface and the element a_{jk} indicates the contribution of the kth surface at time j.

The Erlang data we deal with are recorded from March 18th to March 31st, 2009. Due to discontinuities in the information provided by the Telecom antennas, measurements are missing for some time intervals. Furthermore, they have been recorded for two weeks, even we can think that every week, if special events are not present, shares a common behavior. For

these reasons a pre-processing step is needed. We follow the analysis on this dataset presented in some recent works (see [33, 43] for the details). We perform a pixel-wise smoothing of the Erlang through a Fourier basis expansion of period one week. In this way we aim to represent the Erlang profiles ad a weighted sum of sines and cosines of increasing frequency. Formally we obtain:

$$x_i(t) = \frac{c_{i0}}{2} + \sum_{h=1}^{H} [\alpha_{ih} \cos(h\omega t) + \beta_{ih} \sin(h\omega t)],$$

where $t \in [0; T]$, $\omega = 2\pi/T$ and $T = 60 \times 24 \times 7$ is the period expressed in minutes. The coefficients c_0, α and β are estimated via ordinary least squares. To perform the analysis we need to sample the pre-processed Erlang at some time instants. We sample the measurements, for every pixel $i = 1, ..., n$ (with $n = 25 \times 28 = 700$), at $p = 200$ instants of time regularly spaced in the interval $[0; T]$. We use this dataset to perform our analyses.

4.2 Independent Component Analysis: results obtained through fastICA and scICA algorithms

In this section we focus our attention on the ICA framework. In this case the sources (i.e, the columns of $\mathbb{S}$) are spatial maps. Classical ICA methods, as fastICA, do not take into account this information. Hence we want to apply the scICA algorithm we proposed in Chapter 3 that, instead, exploits this information in the estimate of $\mathbb{S}$ and A. Furthermore we compare it with the well-known fastICA algorithm. In Figure 4.2, 4.3 and 4.4 we present three significative components identified by the two algorithms. Figure 4.2 seems to catch working activities. Indeed the temporal profiles, that are quite similar, are turned on during the daily hours of the working days more than during the daily hours of the weekend (the first day shown is Wednesday) and turned off during the nights. The spatial sources highlight the financial districts in the center of the city (i.e., the areas devoted to working activities). Figure 4.3 catches the behavior of the railway stations. Indeed both temporal profiles present a peak every working day around 6pm, when people take the train to come back home after work. However, while fastICA highlights in the spatial map only the Central railway station, that is the biggest Milanese station, scICA also catches the Garibaldi station (in the top central part of the map), another large station of the city.

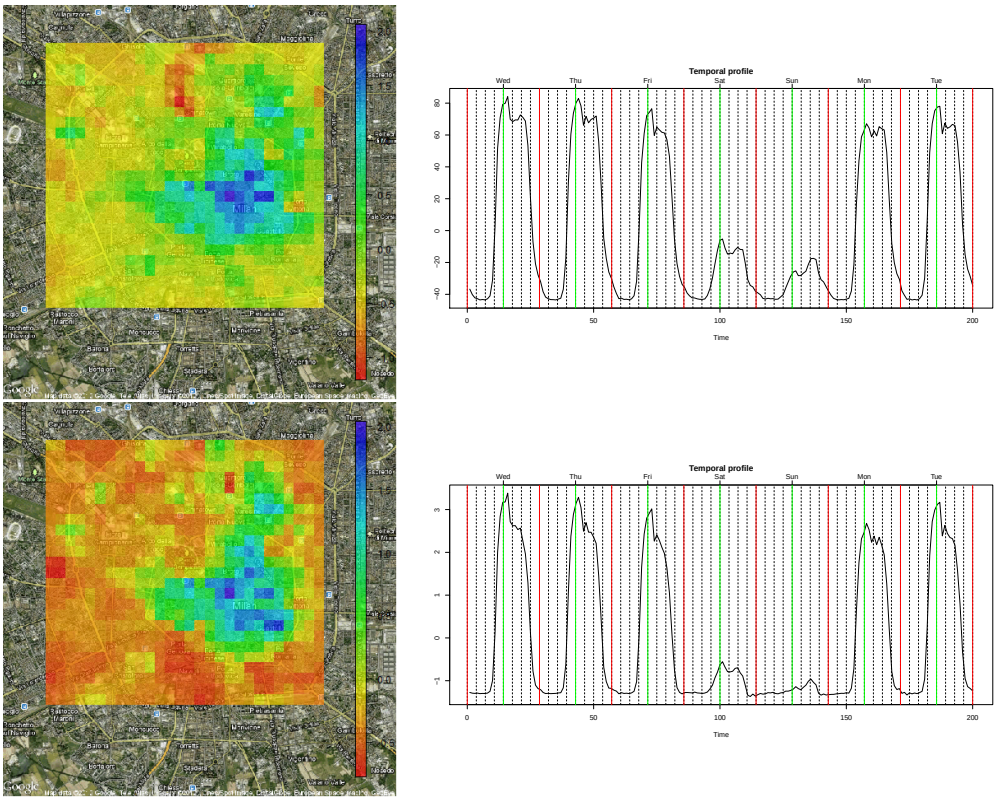

Figure 4.2: Working activities: the top panel presents surface (on the left) and temporal profile (on the right) identified by scICA. The bottom panel presents surface (on the left) and temporal profile (on the right) identified by fastICA. The surfaces catch the areas devoted to working activities. The temporal profiles are quite similar and they are turned on during the daily hours of the working days more than during the daily hours of the weekend and turned off during the nights.

Figure 4.4 presents the more interesting component to compare the two methods, where the improvements due to take into account the spatial dependence seem clear. The temporal profiles, indeed, are turned on during the daily hours of the working days more than during the weekend, with a peak around 6pm. The surfaces identify the areas around the center. This component seems to speak about the traffic after the work activities. However scICA component presents a more interesting surface, highlighting the big outflow streets, while fastICA seems able to highlight only the big ring around the center of the city.

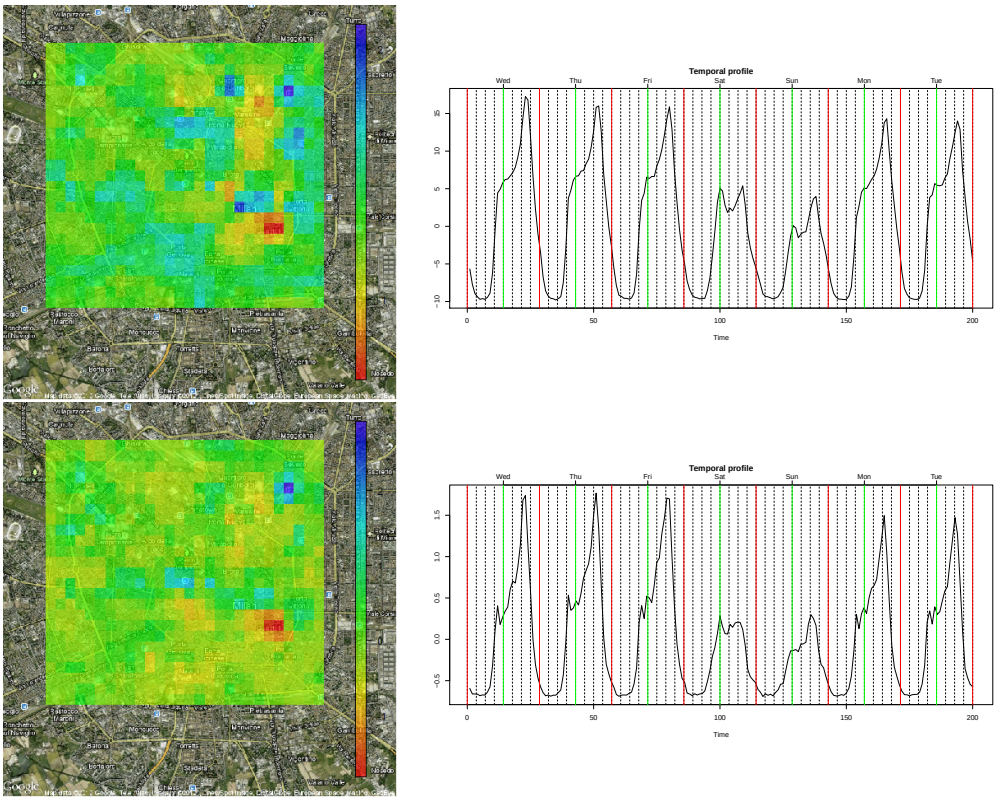

Figure 4.3: Railway stations: the top panel presents surface (on the left) and temporal profile (on the right) identified by scICA. The bottom panel presents surface (on the left) and temporal profile (on the right) identified by fastICA. Both temporal profiles present a peak around the 6pm of the working days. The fasICA source (bottom panel on the left) shows a single pixel with a high value on the Central railway station, the biggest railway station of Milan. The scICA source (top panel on the left) highlights the Central railway station, but also highlights Garibaldi railway station (in the central top part of the map), another large railway station of the city.

4.3 Multi-resolution analysis: results obtained through HICA algorithm

The two methods analyzed rely on some assumption on the sources. In particular the independence between the sources, with scICA method that takes into account also the spatial structure of the independent components. However also the structure of the basis matrix can be analyzed. In particular a multi-resolution analysis seems very useful in this contest, because temporal profile defined only on restricted time intervals could be as meaningful as temporal profile defined globally. Indeed, in the analyses already presented in the literature on this dataset [33] and [43], the treelet algorithm is performed.

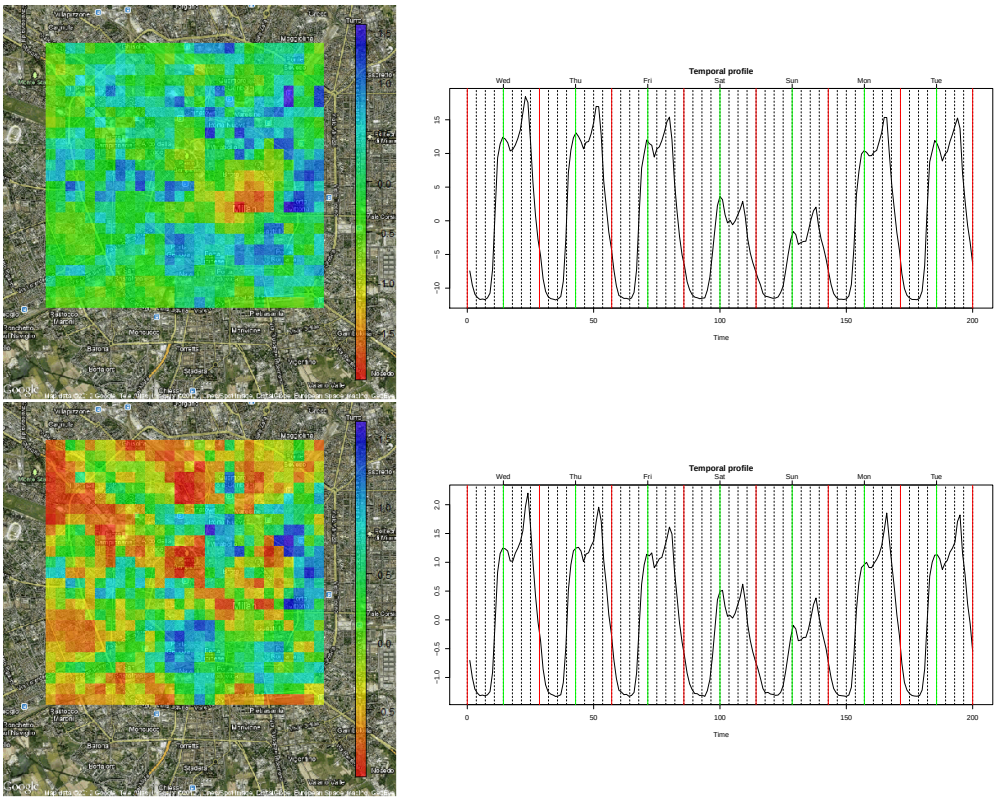

Figure 4.4: Traffic: the top panel presents surface (on the left) and temporal profile (on the right) identified by scICA. The bottom panel presents surface (on the left) and temporal profile (on the right) identified by fastICA. The temporal profiles are on during the daily hours of the working days more than the weekend, with a peak around 6pm. The surfaces identify the areas around the center. This component seems to speak about the traffic after the work activities. The scICA component presents a more interesting surface, highlighting the big outflow streets of the city.

Treelets, as we described in the introduction, are a multi-resolution data-driven basis and it is an efficient and computationally fast tool to decompose data. However we point out that, in BSS problems, they present some drawbacks. In Chapter 1 we proposed HICA, a treelet inspired algorithm particularly for BSS problems. Hence we apply to this dataset the HICA algorithm. In Figure 4.5 we show an interesting component catched by HICA. The spatial map highlights the Central railway station, a feature we also found in Section 4.2. However in this case the temporal profile is totally different. It presents two major peaks; the first on Friday around 6pm and the second on Sunday around 7pm. This component seems to represent the people which leave the city for the weekend.

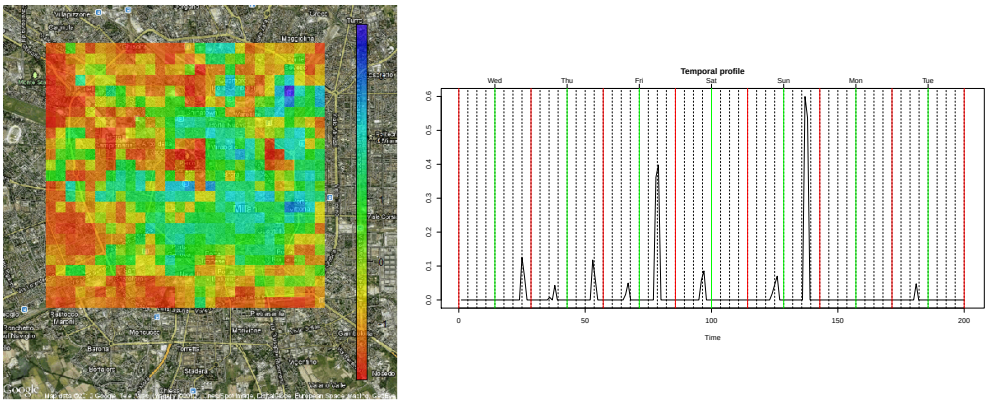

Figure 4.5: An example of surface (on the left) and temporal profile (on the right) identified by HICA. The temporal profile presents two biggest peaks, on Friday around 6pm and on Sunday around 7pm, while the source highlights the Central railway station. This component seems to represent the people which leave the city during the weekend.

4.4 Summary of the results

In this chapter we analyzed the Telecom dataset through different methods. The comparison between fastICA and scICA highlighted the improvements provided by the method we proposed in this manuscript. Indeed, taking into account the spatial dependence between the observations allows to obtain richer spatial components. In the component shown in Figure 4.3, for example, we are able to find of two similar railway stations of Milan, while only one of them were catched by the classical ICA algorithm. The components in Figure 4.4 of the two algorithms, instead, seem to describe the same feature (i.e., the traffic after the work activities). However through scICA algorithm we obtain a more clear spatial component, thanks to the fact we include the dependence between the pixels during the procedure to estimate spatial sources and temporal profiles. Furthermore, this case study is very useful to highlight the different perspectives and the different characteristics of BSS methods that rely on assumptions on the sources and BSS techniques that make assumptions on the mixing matrix. Indeed we showed how, using different methods, we can find out different temporal behaviors related to the same area. In particular, about the central railway station, we saw that in one case we are able to catch a global behavior (i.e., the peak every working day around 6pm, when people take the train to come back home after work), while in another case, using a multi-resolution approach, we are able to find a localized (in time) behavior (i.e., people which leave the city for the weekend). According to the problem under study and to desired achievements one method typology can be more useful than the other, but it is impossible to specify the best

approach overall.

Part III

Alternating Least Square for Functional Data with equality and inequality constraints

Chapter 5

Alternating Least Square

A very popular method to face BSS problems is Nonnegative Matrix Factorization (NMF), which has been shown to be a useful decomposition tool for multivariate data (see [27, 28]). In particular it is applied to imaging problem and text data mining. Furthermore it is widely used in the analysis of complex chemical, pharmaceutical or agricultural mixtures through spectrography, spectrometry, or chromatography. In this field it is often referred as Multivariate Curve Resolution (MCR), as described in [7, 15, 48]. Differently from other BSS methods, where an estimate of $\mathbb{S}$ and A is found making some assumptions on these matrices (e.g., independence between the sources or sparsity), NMF algorithms aim to find $\widehat{\mathbb{S}}$ and $\widehat{A}$, estimates of $\mathbb{S}$ and A respectively, minimizing the distance of $\widehat{\mathbb{S}}\widehat{A}'$ from the data matrix $\mathbb{X}$ imposing the nonnegativity constraint on all the variables. To minimize the distance the most common cost functions are the classic Frobenius norm

$$F(\mathbb{S}, A) = \|\mathbb{X} - \mathbb{S}A'\|_F^2 = \sum_{ij} (\mathbb{X}_{ij} - (\mathbb{S}A')_{ij})^2 \tag{5.1}$$

or the generalized Kullback-Leibler divergence

$$F(\mathbb{S}, A) = D(\mathbb{X}\|\mathbb{S}A') = \sum_{ij} (\mathbb{X}_{ij} \log \frac{\mathbb{X}_{ij}}{(\mathbb{S}A')_{ij}} - \mathbb{X}_{ij} + (\mathbb{S}A')_{ij}).$$

Both this functions vanish if and only if $\mathbb{X} = \mathbb{S}A'$. In this manuscript we focus on the Frobenius norm. Our goal is to minimize (5.1) subjecting to nonnegativity constraints on both the elements of $\mathbb{S}$ and A. Even considering the unconstraint problem, it is unrealistic to expect to find a global minimum for the objective function (5.1). Indeed, although it is convex in $\mathbb{S}$ only and in A only, it is not convex in both variable together. Hence an optimization algorithm is needed. A common way to solve this problem is to perform an alternating algorithm. In particular, keeping fix alternatively $\mathbb{S}$ and A, the current estimate for the other matrix is found

through a nonnegative least square (see [28]). This kind of algorithms are called Alternating Least Square (ALS). However, in practical problems, nonnegativity could not be the only constraint to take into account. Some works introduce regularization algorithms to impose sparsity constraints on $\mathbb{S}$ and A, in order to obtain simpler and more manageable solution (see for example [11]). In other cases can be interesting to impose equality constraints. For instance, when the rows of $\mathbb{X}$ gather the analysis of chemical compounds (i.e., gas chromatograms or mass spectrograms), the matrix $\mathbb{S}$ indicates the concentrations of the unknown referent elements of the matrix A that form the mixtures in $\mathbb{X}$. In this case each row of $\mathbb{S}$ needs to sum to 1, since represents the concentration profile of its related mixture. Generally speaking we can consider both equality and inequality constraint and we can formalize our problem in the following way:

Problem 5.1 *Given a $n \times K$ data matrix $\mathbb{X}$, we look for a $n \times K$ matrix $\widehat{\mathbb{S}}$ and a $p \times K$ matrix $\widehat{A}$, estimates of $\mathbb{S}$ and A respectively, such that*

$$\begin{aligned}(\widehat{\mathbb{S}}, \widehat{A}) = \quad & \arg\min \|\mathbb{X} - \mathbb{S}A'\|_F^2 \\ & u.c.\ H_k^a \boldsymbol{a}_k \geq \boldsymbol{h}_k^a \ \ \forall\, k = 1, ..., K \\ & \quad H_i^s \boldsymbol{s}_i \geq \boldsymbol{h}_i^s \ \ \forall\, i = 1, ..., n \\ & \quad W_k^a \boldsymbol{a}_k = \boldsymbol{w}_k^a \ \ \forall\, k = 1, ..., K \\ & \quad W_i^s \boldsymbol{s}_k = \boldsymbol{w}_i^s \ \ \forall\, i = 1, ..., n\end{aligned} \tag{5.2}$$

where $\mathbf{a}_k$ is the kth column of A, $\mathbf{s}_i$ is the ith row of $\mathbb{S}$ and H_k^a, $\mathbf{h}_k^a$, H_i^s, $\mathbf{h}_i^s$, W_k^a, $\mathbf{w}_k^a$, W_i^s and $\mathbf{w}_i^s$ identify the inequality and equality constraints for the columns of A and the rows of $\mathbb{S}$.

In the rest of the chapter we focus on two aspects to deal with. In Section 5.1 we analyze how to deal with the different kind of constraints. In particular we see that, in some specific situation, taking into account all the constraints minimizing the objective function provides worst solution. Hence we propose a procedure to avoid this problem. Then, in Section 5.2, we deal with the case $\mathbb{X}$ is a functional dataset, mainly focusing on the problem of registration of functional data and how registration affects the resolution of ALS.

5.1 Equality and inequality constraints

The most common way used in literature to solve the problem (5.2) is the ALS algorithm. As a preliminary step, starting estimates $\widehat{\mathbb{S}}^{(0)}$ and $\widehat{A}^{(0)}$ are set. Then, at the generic level l of the algorithm the following two steps are performed:

1) Updating of the estimate of $\mathbb{S}$

$$\widehat{\mathbb{S}}^{(l)} = \arg\min \|\mathbb{X} - \mathbb{S}\widehat{A'}^{(l-1)}\|_F^2$$

$$u.c.\ H_i^s \mathbf{s}_i \geq \mathbf{h}_i^s \ \ \forall\, i = 1, ..., n$$

$$W_i^s \mathbf{s}_k = \mathbf{w}_i^s \ \ \forall\, i = 1, ..., n$$

2) Updating of the estimate of A

$$\widehat{A}^{(l)} = \arg\min \|\mathbb{X} - \widehat{\mathbb{S}}^{(l)} A'\|_F^2$$

$$u.c.\ H_k^a \mathbf{a}_k \geq \mathbf{h}_k^a \ \ \forall\, k = 1, ..., K$$

$$W_k^a \mathbf{a}_k = \mathbf{w}_k^a \ \ \forall\, k = 1, ..., K$$

and these two steps are repeated until convergence. Then, at each level of the algorithm, two constrained optimization problem need to be solved. Inequality constraints involve positivity constraints and, for instance, other boundary constraints on the variables that can be provided by the specific problem under study. Equality constraints involve, for example, the constraint on the sum of the rows of $\mathbb{S}$ or other a priori information on the variables (e.g., we could know the exact value of some of the elements of $\mathbb{S}$ and/or A) that can be introduced in the optimization algorithm. In particular the constraint on the sum of the rows of $\mathbb{S}$ or on the rows of A is very important in the resolution of problem (5.2). Indeed, even in those situation where such a constraint has not a phenomenological interpretation, it is taken into account to make the solution identifiable. Let $(\widetilde{\mathbb{S}}, \widetilde{A})$ be a solution of problem (5.2). Then, every invertible $K \times K$ matrix L that keeps $\widetilde{\mathbb{S}}L$ and $L^{-1}\widetilde{A}$ in the feasible region provides an equally valid solution $(\overline{\mathbb{S}}, \overline{A})$, with $\overline{\mathbb{S}} = \widetilde{\mathbb{S}}L$ and $\overline{A} = L^{-1}\widetilde{A}$.

These constraints have to be treated differently. The introduction of some of them, indeed, could provide a constrained solution significantly different from the unconstrained one. In order to show clearly this problem we consider a simplified example. Our goal is to check the behavior of the estimate of the ith row of $\mathbb{S}$ under different constraints when the columns of A present collinearity. We consider $K = 2$, in order to graphically represent the isolines of the objective function. We analyze three different situations, summarized in Table 5.1.

	Objective function	Constraints
Case 1	$\sum_{j=1}^{p}(x_{ij} - s_{i1}a_{j1} - s_{i2}a_{j2})^2$	$s_{i1} + s_{i2} = 1$
Case 2	$\sum_{j=1}^{p}(x_{ij} - s_{i1}a_{j1} - s_{i2}a_{j2})^2$	$s_{i1}, s_{i2} \geq 0$
Case 3	$\sum_{j=1}^{p}(x_{ij} - s_{i1}a_{j1} - s_{i2}a_{j2})^2$	$s_{i2} = 0$

Table 5.1: The three different situations analyzed. For every case objective function and constraints are displayed.

In particular we are interested in analyzing what happens taking into account these three different constraints when the unconstrained solution is out, but close, from the feasible region. We fix $p = 12$ and we generate the columns of A randomly such that they have a high correlation. Then $\mathbb{X}$ is generated through the model (3) adding some noise and with $\mathbb{S}$ fixed such that the optimum lays close to the feasible region in the three different cases of Table 5.1.

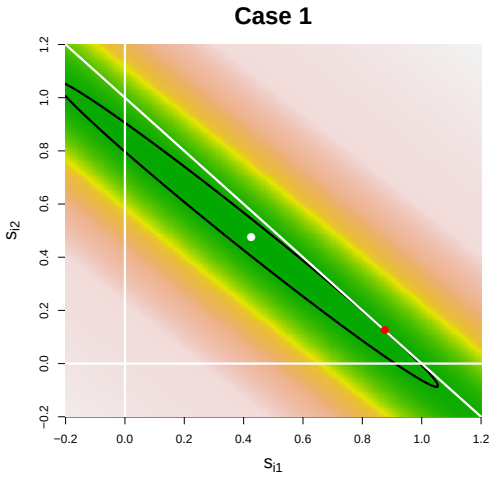

Figure 5.1: Case 1 - the constraint $s_{i1} + s_{i2} = 1$ is considered and in the graph the unconstrained solution (white point), the isoline tangent to the feasible region and the constrained solution (red point) are shown.

From Figure 5.1 it is clear that the introduction of the constraint of sum 1 leads to a constrained solution significantly far from the unconstrained solution, because of the shape of the isolines due to the collinearity between the columns of A. This problem is avoided in the other two cases, as depicted in Figure 5.2.

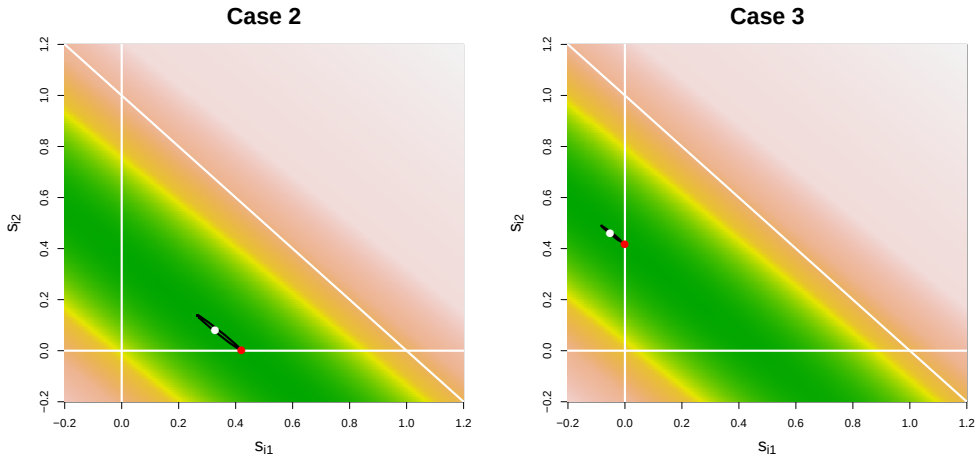

Figure 5.2: Case 2 and 3 - the constraints $s_{i1}, s_{i2} \geq 0$ (on the left) and $s_{i2} = 0$ (on the right) are considered and in the graphs the unconstrained solutions (white points), the isolines tangent to the feasible regions and the constrained solutions (red points) are shown.

The collinearity problem is generally solved in literature normalizing the unconstrained solution (or the constrained solution if other constraints are considered). Although it is a good compromise, such procedure does not take into account this knowledge about the problem. Then we propose to modify the objective function through a penalization. Hence we change the first step of the ALS algorithm in the following way:

1) Updating of the estimate of $\mathbb{S}$

$$\widehat{\mathbb{S}}^{(l)} = \arg\min \|\mathbb{X} - \mathbb{S}\widehat{A}'^{(l-1)}\|_F^2 + \lambda_s P(\mathbb{S})$$

$$u.c.\ H_i^s \mathbf{s}_i \geq \mathbf{h}_i^s \ \ \forall\, i = 1, ..., n$$

$$W_i^s \mathbf{s}_k = \mathbf{w}_i^s \ \ \forall\, i = 1, ..., n$$

2) Updating of the estimate of A

$$\widehat{A}^{(l)} = \arg\min \|\mathbb{X} - \widehat{\mathbb{S}}^{(l)} A'\|_F^2$$

$$u.c.\ H_k^a \mathbf{a}_k \geq \mathbf{h}_k^a \ \ \forall\, k = 1, ..., K$$

$$W_k^a \mathbf{a}_k = \mathbf{w}_k^a \ \ \forall\, k = 1, ..., K$$

where $P(\mathbb{S})$ penalizes, for every row of $\mathbb{S}$, the quantity $(\sum_{k=1}^{K}(s_{ik}) - 1)^2$.

Remark 5.1 *We presented the case where the constraint is the sum of the rows of* $\mathbb{S}$ *equal to 1, but this can be generalized to the matrix* A *and to the case where the sum is equal to a constant* $c \neq 1$.

Remark 5.2 *At every step of the ALS algorithm a constrained optimization has to be solved. Then, an optimization algorithm is needed. We considered the L-BFGS-M (Limited Memory BFGS), which allows box constraints and uses a limited-memory modification of the BFGS quasi-Newton method (see [10] for the detailed algorithm).*

5.2 The problem of registration for functional datasets

We now focus on the case where $\mathbb{X}$ presents functional features (see [39] for a detailed description of Functional Data Analysis). In the ALS problem, if $\mathbb{X}$ gather n functional data, the matrix A presents K functional reference elements and $\mathbb{S}$ the coefficients which generate the data. This is the case when the analysis of some chemical compounds is performed through spectrometry (or chromatography), for instance. In this case the spectrogram (or the chromatogram) is studied as a function. For example, in [5] mixtures of acetone and acrolein are analyzed through their gas chromatograms. In this case the chromatogram, as function of time, is the functional data. In [7], instead, spectrograms of several pharmaceutical samples are studied. In this case the functional data is represented by the spectrogram, as function of the wavelength. In literature there are a lot of issues related to Functional Data Analysis. In particular we deal with a problem often encountered with functional data, the misalignment of the data, and how it affects the ALS algorithm. In some situations the different functions follow a similar course, but the more important characteristics of this course happen at different time. In this case the alignment (or registration) is crucial to allow a correct analysis of the variability in the ordinate (amplitude variability). Then, there is the need to decouple the amplitude variability from the variability present in the abscissa (phase variability). If we are interesting in studying the amplitude variability, the misalignment has to be removed, in order to avoid the estimate of the K functional reference elements catching the phase variability. Even if this problem can be a crucial criticism in the ALS resolution, it has not been treated in literature. In particular, when functions depend on time, the misalignment is often critical for a correct resolution of the BSS problem, as we analyze in the following chapter.

A lot of methods for curve registration have been proposed and studied in the literature, exploiting, for example, self-modeling non linear regression methods or non linear mixed effects model (see [3, 26, 31] for the

details). Another line of research, instead, defines suitable similarity indexes between curves, and then the registration is performed maximizing their similarities by means of a Procustes procedure (see [41, 42, 24] for the detailed works). We consider the latter line of research and we propose to perform the registration of data as a preliminary step of the ALS algorithm.

What we propose is to perform a k-medoid alignment, described in [42], which aims at aligning k groups of functional data, performing alignment and clustering simultaneously. In our case we consider $k = 1$, since our attention is only for the registration problem. Then, once data have been aligned, we can proceed with the ALS described in Section 5.1.

Chapter 6

Analysis of chromatograms

In this chapter we apply the ALS algorithm to the analysis of the chromatograms of some chemical mixtures in order to retrieve the chromatograms of the original chemical compounds which generate the mixtures and the concentrations in each mixture. In particular we perform two different analyses. One exploiting a multivariate dataset obtained by a synthesis of the chromatograms and the other using the whole chromatogram (i.e., a functional data).

In this analysis we have $n = 16$ chemical mixtures of $K = 3$ original chemical compounds. The mixtures are analyzed through a gas-chromatography procedure. A gas chromatograph is a chemical analysis instrument for separating chemical elements in a complex mixture. A gas chromatograph uses a flow-through narrow tube, through which different chemical constituents of a mixture pass in a gas stream at different rates depending on their various chemical properties and their interaction with a specific column filling, called the stationary phase. When the elements exit at the end of the column, they are detected and identified. The function of the stationary phase in the column is to separate different chemical elements, causing each one to exit the column at a different time (retention time). Hence a chromatogram presents different peaks at different retention times. At each retention time is associated a specific chemical element and the area under the peak represents the quantity of its related element present in the analyzed mixture. One of the chromatogram analyzed in this chapter (i.e., a part of it related to a standard reference area) is shown in Figure 6.1.

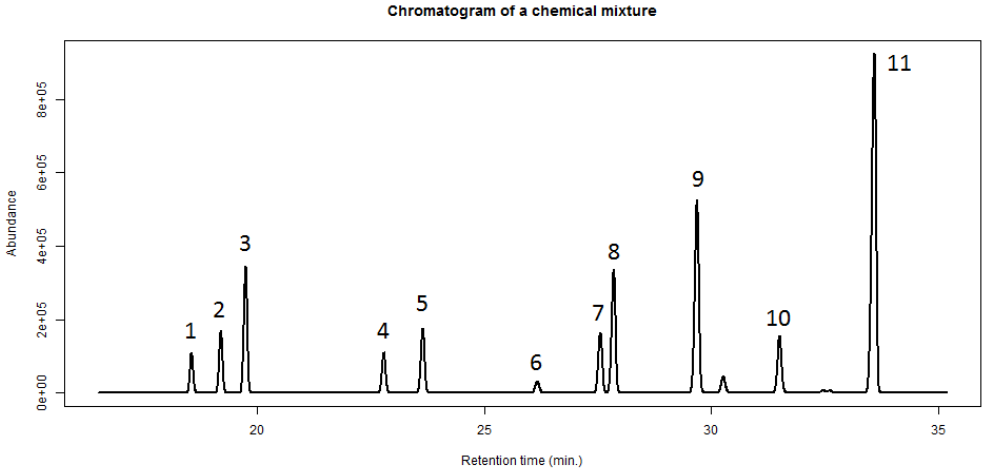

Figure 6.1: Example of chromatogram. The numbers from 1 to 11 highlight the peaks considered for the evaluation of the areas.

We obtain an evaluation of each chromatogram in $p = 4100$ instants of time. Due to the nature of the problem, the chromatogram of the ith mixture at the jth instant satisfies:

$$x_{ij} = \sum_{k=1}^{K} s_{ik} a_{jk} \tag{6.1}$$

where s_{ik} is the concentration of the kth original compound in the mixture i and a_{jk} is the value of the chromatogram of the kth original compound at the jth instant of time. Each variable s_{ik}, a_{jk} for $i = 1, ..., n$, $j = 1, ..., p$ and $k = 1, ..., K$ needs to be greater than zero. Then we have a further constraint on the elements s_{ik}. Specifically, for every $i = 1, ..., n$, we require $\sum_{k=1}^{K} s_{ik} = 1$. Since we use mixtures prepared in laboratory, we know both the chromatograms of the original compounds $a_1(t), ..., a_K(t)$ and the concentrations s_{ik}. Then we can use this dataset to test the procedures described in the previous chapter. Since the concentrations matrix has some elements equal to zero, we consider to know them, introducing other equality constraints $s_{ik} = 0$ for some i and some k.

Mixture	A	B	C
1	0.9	0.05	0.05
2	0.25	0	0.75
3	0.15	0.85	0
4	0.2	0.4	0.4
5	0	0.13	0.87
6	0.05	0	0.95
7	0.3	0.4	0.3
8	0	0.5	0.5
9	0.15	0.3	0.55
10	0.6	0.1	0.3
11	0.3	0.7	0
12	0.03	0.01	0.96
13	0.35	0.55	0.1
14	0.4	0.4	0.2
15	0.25	0.15	0.6
16	0.96	0.03	0.01

Mixture	A	B	C
1	X	X	X
2	X	0	X
3	X	X	0
4	X	X	X
5	0	X	X
6	X	0	X
7	X	X	X
8	0	X	X
9	X	X	X
10	X	X	X
11	X	X	0
12	X	X	X
13	X	X	X
14	X	X	X
15	X	X	X
16	X	X	X

Figure 6.2: Real concentrations matrix (on le left) and matrix containing the information known a priori (on the right).

The real concentrations matrix and the matrix with the information known a priori are shown in Figure 6.2. Our goal is to retrieve the chromatograms of the original compound and the concentrations matrix through the chromatograms of the mixtures, the a priori information and the suitable constraints. Hence we implement an ALS algorithm, taking into account the functional nature of the data. However we also analyze a multivariate data, coming from a synthesis of the chromatogram. In particular, since every peak of the chromatogram is associated to the retention time of a specific chemical element, the area of the peak provides the quantity of that specific element. An idea is to "vectorize" the functional data measuring the areas of some relevant peaks. In this analysis we take into account the areas of the 11 peaks highlighted in Figure 6.1. Before the comparison between these two different analyses it is worth to point out that the evaluation of the areas presents some drawbacks. Firstly it introduces a subjective choice of the points which identify the base of the peak for the evaluation

of the area. Then, in some situations, the separation between two or more peaks is not so clear. This is the case, for example, of the peaks 7 and 8 in Figure 6.1.

The rest of the chapter is organized as follows. In Section 6.1 we present a useful a simple procedure to choose the value of K. Then, in Sections 6.2 and 6.3, we present the results obtained using the vector of the areas and the whole chromatograms, respectively. Finally, in Section 6.4 we compare the multivariate and the functional approaches.

6.1 Choice of K

In this synthetic analysis all the variables are known. Hence we can check the goodness of our results. For instance we know the value of K (i.e., $K = 3$) of original compounds generating the mixtures. However, in the real problems, K is often unknown. Then, a procedure to find the right K is needed. In this particular situation the chromatograms of the mixtures $\mathbf{x}_1, ..., \mathbf{x}_n$ are generated by a convex linear combination of the chromatograms of the original compounds $\mathbf{a}_1, ..., \mathbf{a}_K$. This means that data lay in the $K-1$ dimensional simplex generated by $\mathbf{a}_1, ..., \mathbf{a}_K$. Hence data live in a space of dimension $K-1$.

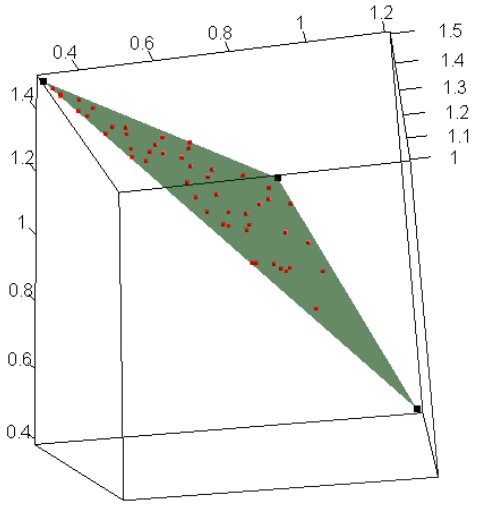

Figure 6.3: Toy example to show that data live in a $K-1$ dimensional space. In the picture the case with $K = 3$, where data live the 2D triangle generated by the columns of A, is shown.

In Figure 6.3 we can see a toy example with $K = p = 3$ and where data have been generated according to the model (6.1). The picture shows that

data lay in a 2-d triangle. Hence our proposal is to perform a Principal Component Analysis [23] to find the dimension h where data live. Then we set $K = h + 1$.

6.2 Analysis with multivariate data

In this section we analyze the chromatograms using the vector of the $p = 11$ areas highlighted in Figure 6.1. Firstly we perform a PCA to find the right value for K. In this case we know that $K = 3$. Hence, according to the Section 6.1, we should find an elbow at 2 Principal Components. However the graph on the left of Figure 6.4, seems to suggest $h = 4$ Principal Components (i.e., $K = 5$). This further artificial variability could have been introduced by some error in the measurements of the areas. Hence, some diagnostic tools are needed.

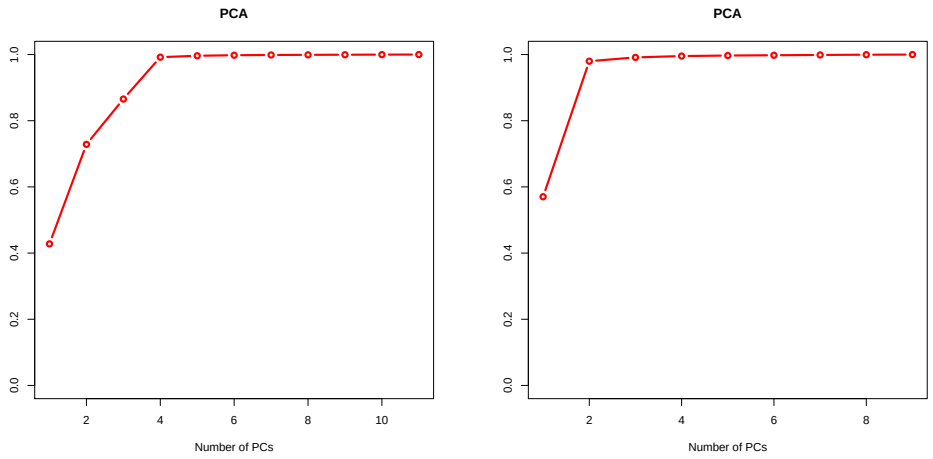

Figure 6.4: Portion of explained variance by the PCs of the original multivariate dataset (on the left) and the dataset after that the peaks 7 and 8 have been eliminated (on the right). In the panel on the right an elbow at $h = 2$ is evident, as expected.

6.2.1 Diagnostic tools

The more natural diagnostic tool is to consider how our estimates fit the data. In particular we focus on the difference

$$(x_{ij} - \sum_{k=1}^{K} \widehat{s}_{ik}\widehat{a}_{jk})^2, \tag{6.2}$$

checking (6.2) for every element of the data matrix $\mathbb{X}$ (i.e, for $i = 1, ..., n$ and $j = 1, ..., p$). The evaluation of the misfit can lead us to find anomalous

behaviors for some observations (mixtures) and/or some variables (peaks). Specifically, if we find out that the biggest misfit errors are all related to the same observation, we can interpretate it as anomalous and proceed in the analysis eliminating the observation (and the same can be done with a variable). In our problem, eliminating an observation correspond to eliminate a mixture. This means that we will not be able to estimate the concentrations related to that mixture. Eliminating a variable, on the contrary, means that we will not be able to estimate one peak area of the original compounds generating the mixtures. However, given that the new n and the new p remain greater than K, we can still estimate all the other quantities. In fact, the elimination of the aberrations should provide a more accurate estimate. Hence, giving up to find some quantities of interest is preferable if it allows us to improve the rest of the estimates.

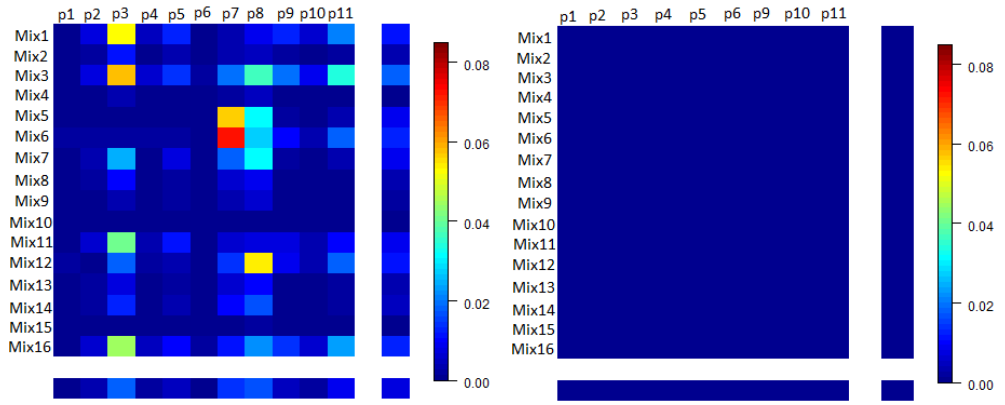

Figure 6.5: Left panel: evaluation of $\|\mathbb{X} - \widehat{\mathbb{X}}\|$ for the original data matrix. The biggest errors seem to occur for the peaks 3,7 and 8. Right panel: evaluation of $\|\mathbb{X} - \widehat{\mathbb{X}}\|$ after the elimination of peaks 7 and 8. Errors are small for every mixture and for every peak.

In the left panel of Figure 6.5 the evaluation of $\|\mathbb{X}-\widehat{\mathbb{X}}\|$ for every element of the matrix is shown. The biggest errors seem to occur for the peaks 3,7 and 8. Peaks 7 and 8, in particular, are those peaks whose separation is not so clear. This implies some difficulties in the measurements of the areas. For this reason we chose to eliminate only these two peaks and to check again the PCA and the misfit errors. In this case, looking at the right panel of Figure 6.4, there is a clear elbow at $h = 2$, as we expected. Regarding the misfit errors, from the right panel of Figure 6.5 it is possible to notice that the errors are small for every mixture and for every peak.

This analysis highlights the fact that, if the evaluation of the areas is not done correctly, data present incongruence that can lead to an inefficient resolution of the BBS problem. The diagnostic through the analysis of the misfit errors helps us to find these anomalies. Moreover we can also look

at the PCA as a diagnostic tool itself in those cases where K is known. Indeed, if we know the dimension of the space where data should live, we can use the PCA to verify that data live in the desired space.

We now analyze the results obtained in the estimate of $\mathbb{S}$ and A through the ALS algorithm described in the Section 5.1.

6.2.2 Analysis of results

We analyze the modified dataset after the elimination of the anomala peaks. Firstly we focus on the concentration matrix $\mathbb{S}$. In particular we analyze the introduction of the penalty related to the constraint on the rows of $\mathbb{S}$ (i.e., every row of $\mathbb{S}$ needs to sum up to 1), in order to verify if the penalty effectively provides improvements on the solution. In Figure 6.6 the error as a function of λ_s is shown. The error is measured as the mean of the estimation error of the unknown concentrations in the matrix $\mathbb{S}$. Figure 6.6 shows a clear improvement introducing a penalty in the objective function. In particular the error with $\lambda = 0$ is equal to 0.04366, while with λ optimum it is equal to 0.03532, reducing by a 20%.

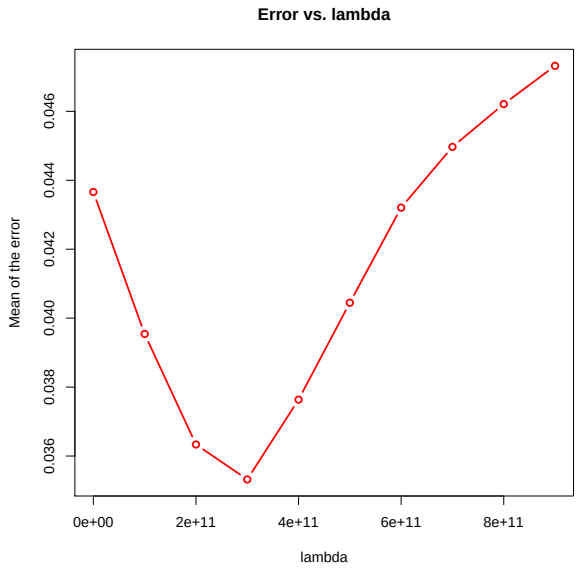

Figure 6.6: Estimation error of matrix $\mathbb{S}$ versus λ. There is a minimum for $\lambda = 3 * 10^{11}$ where the error is reduced by 20%.

In Figure 6.7 we compare $\widehat{\mathbb{S}}$ obtained for $\lambda = 0$ and for the λ optimum, with the true $\mathbb{S}$ and the main improvements provided by the penalized solution are highlighted.

Mix	A	B	C
1	0.9	0.05	0.05
2	0.25	0	0.75
3	0.15	0.85	0
4	0.2	0.4	0.4
5	0	0.13	0.87
6	0.05	0	0.95
7	0.3	0.4	0.3
8	0	0.5	0.5
9	0.15	0.3	0.55
10	**0.6**	**0.1**	**0.3**
11	0.3	0.7	0
12	0.03	0.01	0.96
13	0.35	0.55	0.1
14	**0.4**	**0.4**	**0.2**
15	**0.25**	**0.15**	**0.6**
16	0.96	0.03	0.01

Mix	A	B	C
1	0.94	0.06	0
2	0.29	0	0.71
3	0.07	0.93	0
4	0.2	0.41	0.39
5	0	0.05	0.95
6	0.03	0	0.97
7	0.3	0.39	0.31
8	0	0.35	0.65
9	0.11	0.36	0.53
10	**0.68**	**0**	**0.32**
11	0.25	0.75	0
12	0.03	0.0	0.97
13	0.34	0.56	0.11
14	**0.41**	**0.3**	**0.29**
15	**0.27**	**0.03**	**0.7**
16	0.99	0	0.01

Mix	A	B	C
1	0.94	0.06	0
2	0.29	0	0.71
3	0.07	0.93	0
4	0.21	0.39	0.4
5	0	0.04	0.96
6	0.03	0	0.97
7	0.3	0.4	0.31
8	0	0.35	0.65
9	0.1	0.37	0.53
10	**0.66**	**0.05**	**0.29**
11	0.25	0.75	0
12	0.02	0.04	0.94
13	0.33	0.58	0.09
14	**0.39**	**0.37**	**0.23**
15	**0.24**	**0.15**	**0.61**
16	0.99	0	0.01

Figure 6.7: Comparison of the true $\mathbb{S}$ (on the left) with $\widehat{\mathbb{S}}$ obtained for $\lambda = 0$ (in the middle) and for λ optimum (on the right). The main improvements due to the penalty are highlighted.

We now focus on the penalized solution to compare the true A with $\widehat{A}$, which represents the estimate of the peak areas of the original compounds. In Figure 6.8 the comparison is depicted and can be inferred that, for all the original compounds, the areas of the peaks are estimated very accurately. It is worth to point out that, since we removed two peaks after the diagnostic procedures, we are not able to estimate those areas for the original compounds generating the mixtures.

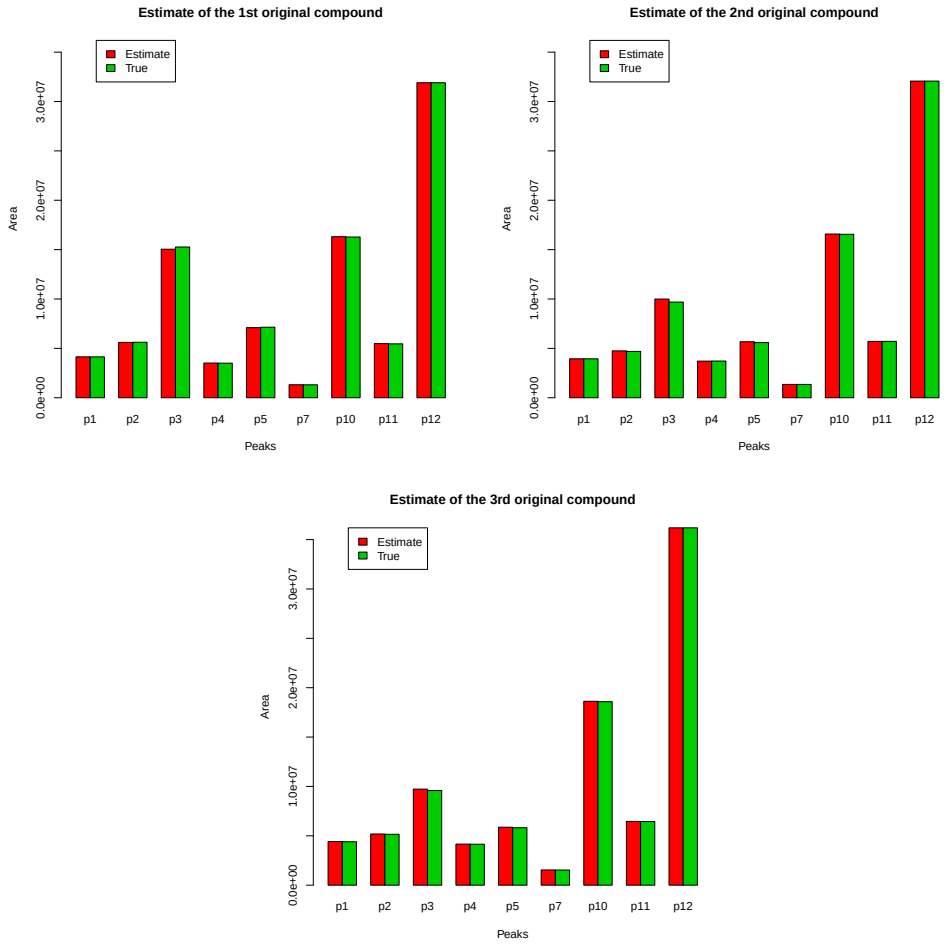

Figure 6.8: Comparison between $\widehat{A}$ (red) and A (green). In the top panel the first and the second compound are shown on the left and on the right respectively. In the bottom panel the third compound is depicted.

6.3 Analysis with functional data

We now consider for the analysis the whole chromatogram. For each mixture we evaluate its chromatogram at $p = 4100$ retention times and these objects are treated as a functions. Firstly, we can see from Figure 6.9 that data are misaligned. Since each peak is associated to a specific retention time and each retention time to a chemical elements, we expect to find the corresponding peaks among the different mixtures at the same abscissa. But this is not the case. The misalignment is probably due to slightly different condition of the machine during the experiment or, simply, is a measurement error. However we are not interested in catching such vari-

ability. Hence, the alignment step is needed. In Figure 6.10 the aligned chromatograms are depicted. Now the phase variability seems almost disappeared.

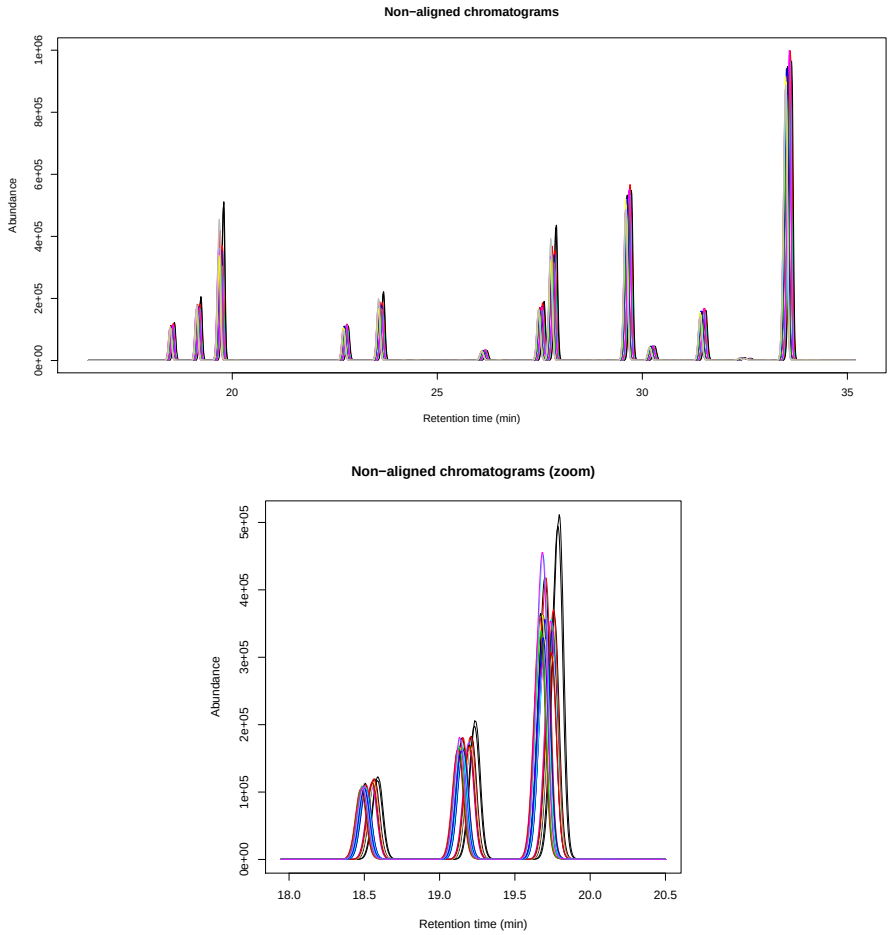

Figure 6.9: In the top panel all the chromatograms contained in $\mathbb{X}$ are shown and the problem of misalignment is evident. In the bottom panel only the first part of the chromatograms is depicted.

Therefore we can proceed applying the ALS algorithm to the aligned chromatograms. Before to do that, we check, through the PCA, if it is possibile to infer on the value of K, when it is unknown. The portion of variance explained by the PCs is shown in the left panel of Figure 6.11. It seems that the first PC is sufficient to explain all the variability of the problem. However, looking at the zoom of the picture in the right panel of Figure 6.11, an elbow at $h = 2$ can be observed. Hence, although the functional nature of the data is such that almost the entire variability can be expressed through the first principal component (the mean function), something on

the real value of K can be still inferred, even if less clearly than in the multivariate case.

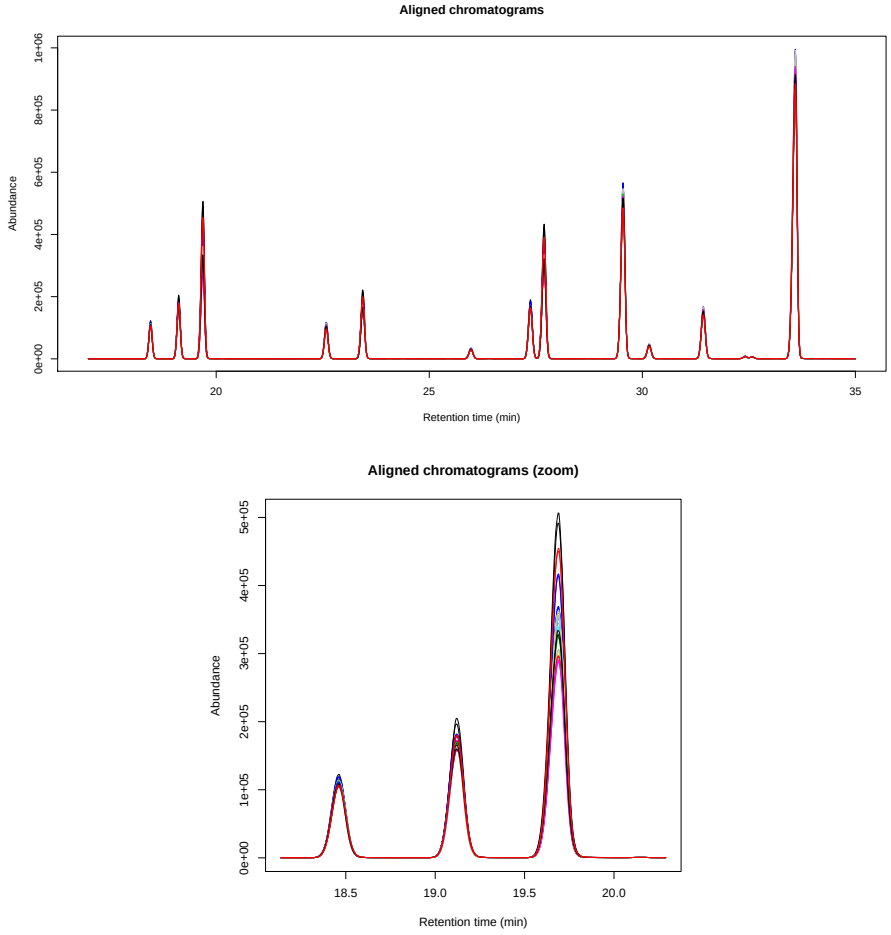

Figure 6.10: In the top panel the aligned chromatograms are shown. In the bottom panel only the first part of the chromatograms is depicted.

We now analyze the results obtained through the ALS algorithm. In this case the penalty does not provide any reasonable improvement. For this reason we consider only the unpenalized solution. In Figure 6.12 the true $\mathbb{S}$ is compared with its estimates obtained with the aligned and non aligned chromatograms. It is evident how the aligned solution is significantly better than the non aligned solution. The mean of the error over the unknown coefficients is, indeed, 0.0959 for the aligned solution and 0.3661 for the non aligned one. This makes clear the crucial part played by the registration in the resolution of the ALS algorithm.

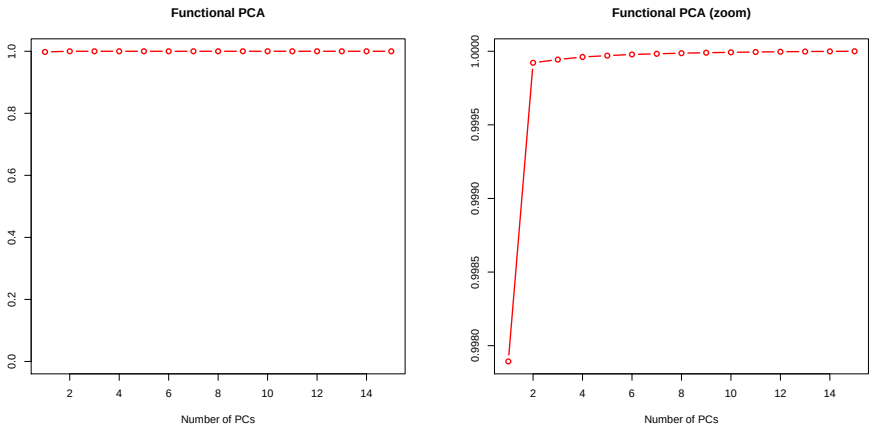

Figure 6.11: Portion of explained variance by the PCs of the functional dataset (on the left) and a zoom of that (on the right). In the panel on the right an elbow at $h = 2$ is evident, as expected.

Mix	A	B	C
2	0.25	0	0.75
3	0.15	0.85	0
4	0.2	0.4	0.4
5	0	0.13	0.87
6	0.05	0	0.95
7	0.3	0.4	0.3
8	0	0.5	0.5
9	0.15	0.3	0.55
10	0.6	0.1	0.3
11	0.3	0.7	0
12	0.03	0.01	0.96
13	0.35	0.55	0.1
14	0.4	0.4	0.2
15	0.25	0.15	0.6
16	0.96	0.03	0.01

Mix	A	B	C
2	0.36	0	0.65
3	0.11	0.89	0
4	0.28	0.34	0.39
5	0	0.42	0.59
6	0.12	0	0.86
7	0.44	0.06	0.48
8	0	0.61	0.39
9	0.2	0.26	0.53
10	0.67	0.12	0.21
11	0.28	0.72	0
12	0	0.45	0.54
13	0.34	0.61	0.04
14	0.41	0.41	0.18
15	0.28	0.25	0.46
16	0.98	0.02	0

Mix	A	B	C
2	1	0	0
3	0.94	0.06	0
4	0.46	0.05	049
5	0	0	1
6	0.24	0	0.76
7	0	0.57	0.43
8	0	0.51	0.49
9	0	0.6	0.4
10	0	0.73	0.37
11	0.24	0.76	0
12	0	0.81	0.19
13	0	1	0
14	0	1	0
15	0	1	0
16	0	1	0

Figure 6.12: Comparison of the true $\mathbb{S}$ (on the left) with $\widehat{\mathbb{S}}$ obtained for the aligned (in the middle) and the non aligned (on the right) dataset. The estimated obtain with the aligned dataset in consistently better.

We now compare the true A with its estimate $\widehat{A}$, that is the estimate of the whole chromatogram of the original compounds generating the mixtures. In Figure 6.13 this comparison can be appreciated. We show only a part of the chromatogram in order to make the graph understandable. The estimate of the chromatograms of the original compounds seems to be very accurate.

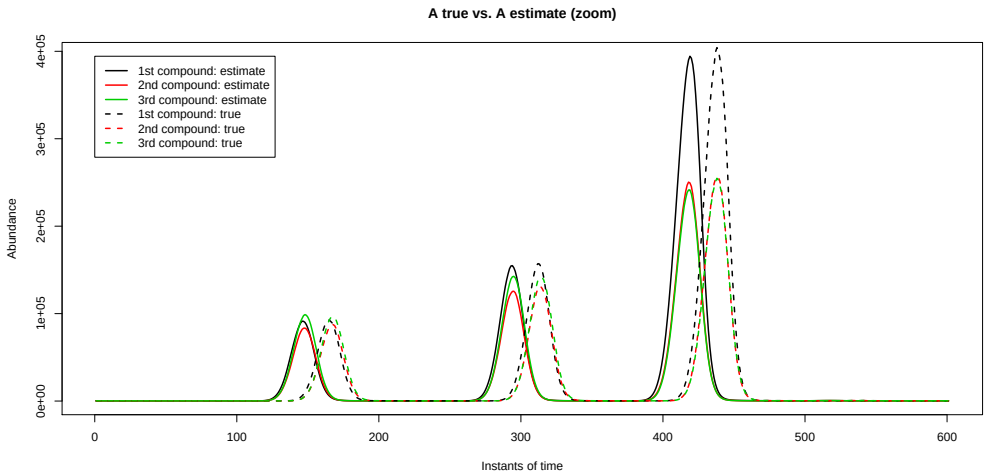

Figure 6.13: Comparison between the true A (dashed curves) with its estimate (solid curves). Curves are misaligned in order to make the comparison easier.

6.4 A comparison between multivariate and functional approach

In this section we compare the multivariate and the functional approach. Regarding the estimate of $\mathbb{S}$, shown in Figure 6.14, the multivariate analysis gives a very good result. The mean error over the unknown coefficients is equal to 0.03532. The estimate provided by the functional presents a slightly higher error, precisely equal to 0.0959. However this estimate is still acceptable. Moreover, looking at the estimates of Figure 6.14, it is possible to notice that the matrix provided by the functional approach, but for a few mixtures (i.e., mixtures number 5, 7 and 12) where the estimate is not very accurate, is absolutely comparable with the matrix given by the multivariate approach, for some mixtures even better. Although the analysis of the concentration estimates seems to make the multivariate approach preferable, it is worth to point out that it presents some drawbacks, as partially described in the previous sections. First of all it requires the manually evaluation of the peak areas and this procedure could introduce some unwanted variability. Sometimes the diagnostic tools can

solve this problem, but the functional approach, in this sense, is preferable, since it does not require any manual pre-processing operation. The functional approach only requires the alignment of the data, in order to avoid that the phase variability compromises the efficiency of the estimate. This procedure is done automatically through an alignment algorithm, like, for instance, the one we proposed in Section 5.2. The only drawback of this step is that data need to be perfectly aligned. In the application we described in this chapter, indeed, we removed the first mixture, since it presented a problem during the alignment step.

Mix	A	B	C
1	0.9	0.05	0.05
2	0.25	0	0.75
3	0.15	0.85	0
4	0.2	0.4	0.4
5	0	0.13	0.87
6	0.05	0	0.95
7	0.3	0.4	0.3
8	0	0.5	0.5
9	0.15	0.3	0.55
10	0.6	0.1	0.3
11	0.3	0.7	0
12	0.03	0.01	0.96
13	0.35	0.55	0.1
14	0.4	0.4	0.2
15	0.25	0.15	0.6
16	0.96	0.03	0.01

Mix	A	B	C
1	0.94	0.06	0
2	0.29	0	0.71
3	0.07	0.93	0
4	0.21	0.39	0.4
5	0	0.04	0.96
6	0.03	0	0.97
7	0.3	0.4	0.31
8	0	0.35	0.65
9	0.1	0.37	0.53
10	0.66	0.05	0.29
11	0.25	0.75	0
12	0.02	0.04	0.94
13	0.33	0.58	0.09
14	0.39	0.37	0.23
15	0.24	0.15	0.61
16	0.99	0	0.01

Mix	A	B	C
1	-	-	-
2	0.36	0	0.65
3	0.11	0.89	0
4	0.28	0.34	0.39
5	0	0.42	0.59
6	0.12	0	0.86
7	0.44	0.06	0.48
8	0	0.61	0.39
9	0.2	0.26	0.53
10	0.67	0.12	0.21
11	0.28	0.72	0
12	0	0.45	0.54
13	0.34	0.61	0.04
14	0.41	0.41	0.18
15	0.28	0.25	0.46
16	0.98	0.02	0

Figure 6.14: Comparison of the true $\mathbb{S}$ (on the left) with $\widehat{\mathbb{S}}$ obtained through the multivariate approach (in the middle) and the functional approach (on the right) dataset. The estimated obtain with the multivariate approach provide a slightly better result.

Another aspect which makes the functional approach preferable is that we use the entire chromatogram and not only a synthesis of it. This is important under at least two point of view. The first is that, when we use some peak areas instead of the whole function, if we select peaks where there is no variability between the original compounds, we cannot solve the

problem. However, since the original compounds are unknown, we cannot be sure to avoid this criticism. The functional approach, on the contrary, allows to use the entire chromatogram and to take into account all the differences between the original compounds, even considering larger areas than the standard reference part, where the divisions between peaks are not very clear and then the multivariate approach is unfeasible. The second advantage given by the functional approach is that it allows to estimate the whole chromatograms of the original compound generating the mixtures and not just the peak areas. If the interest is not focused simply on the concentrations this is an important aspect to take into account. Finally, the multivariate approach, since involves less variables, is computationally cheaper than the functional approach.

Part IV

Computational details

Package 'fastHICA'

November 22, 2013

Type Package

Title Hierarchical Independent Component Analysis: a multi-scale sparse non-orthogonal data-driven basis

Version 1.0

Date 2013-11-22

Author Piercesare Secchi, Simone Vantini, and Paolo Zanini

Maintainer Paolo Zanini <paolo.zanini@polimi.it>

Depends fastICA, extracat, grid, MASS, colorspace, hexbin, lattice,scales, ggplot2, reshape, plyr

Description This package implements HICA (Hierarchical Independent Component Analysis) algorithm. This approach, obtained through the integration between treelets and Independent Component Analysis, is able to provide a multi-scale non-orthogonal data-driven basis, whose elements have a phenomenological interpretation according to the problem under study.

License GPL (>= 2)

R topics documented:

`basis_hica` *Construction of the HICA basis*

Description

This function builds the HICA tree up to a prespecified height providing the corresponding non-orthogonal bases.

Usage

```
basis_hica(X, maxlev = dim(X)[2] - 1, dim.subset = 512)
```

Arguments

X	Data matrix with nrow(X) observations and ncol(X) variables.
maxlev	The maximum level of the tree. This must be an integer between 1 and ncol(X)-1. The default value is set to ncol(X)-1.
dim.subset	The dimension of the subset used for the evaluation of the similarity index (i.e., distance correlation). If this it is greater than nrow(X) all the observations are used, unless a random subsample of dim.subset observations is used. The default value is set to 512.

Value

X	data matrix.
basis	a list with maxlev elements. The ith element of the list contains the basis matrix provided at level i of the tree. Each column of the basis matrix represent a basis element.
aggregation	a matrix with maxlev rows and 3 columns. At each row the first two columns contain the variable indeces merged at the corresponding level of the tree. In the third column the distance correlation of the two merged variables is recorded.

Note

The distance correlation is evaluated through the function wdcor of the package "extracat". It becomes computational unfeasible if the number of observations is too large. For this reason it is possibile to choose the dimension of the subsample to be used in the evaluation of the similarity matrix. By default the dimension is set to 512.

Author(s)

Piercesare Secchi, Simone Vantini, and Paolo Zanini.

References

Secchi, Vantini, and Zanini (2013).

See Also

energy_hica, similarity_hica, extract_hica

Examples

```
## Not run:

############################################################
# Example 2 - Independent sources and overlapping loadings #
############################################################

c1=c(0,0,0,0,1,1)
c2=c(1,1,1,1,0,0)
c3=c(1,1,0,0,0,0)

s1=runif(1000,0,20)
s2=runif(1000,0,20)
s3=runif(1000,0,20)
```

```
# Here we generate the simulation dataset

X=s1%*%t(c1)+s2%*%t(c2)+s3%*%t(c3)+rnorm(6*1000,0,1)

X_in=t(t(X)-colMeans(X))  # Data-matrix whose columns have zero-mean

# Here we perform HICA on the simulation dataset

basis=basis_hica(X,5,1000)
energy=energy_hica(basis,6,5,plot=TRUE)

# We plot the 3 main components of HICA basis
# (according to the energy criterium) for 4th level.

ex4=extract_hica(energy,3,4)
loa4=ex4$C

windows()
par( mfrow = c(3,1))
barplot(loa4[,1], ylim = c(-1, 1),main="HICA transform - Level 4",
ylab="1st component",xlab="Coordinate",names.arg=1:6,col="red",mgp=c(2.5,1,0))
barplot(loa4[,2], ylim = c(-1, 1),ylab="2nd component",
xlab="Coordinate",names.arg=1:6,col="green",mgp=c(2.5,1,0))
barplot(loa4[,3], ylim = c(-1, 1),ylab="3rd component",
xlab="Coordinate",names.arg=1:6,col="blue",mgp=c(2.5,1,0))

## End(Not run)
```

`energy_hica` *Energy criterion*

Description

This function implements the energy criterion defined in Secchi, Vantini, and Zanini (2013).

Usage

```
energy_hica(HICA.obj, maxcomp = 1, nlevel = 1, plot = TRUE)
```

Arguments

`HICA.obj`	An object provided by the function `basis_hica`.
`maxcomp`	The maximum space dimension considered.
`nlevel`	The number of levels analyzed. Specifically the levels from p-`nlevel` to p-1 are analyzed, where p is the number of variables.
`plot`	A logical value. If TRUE the energy is plotted.

Details

This function computes the energy according the criterion presented in Secchi, Vantini and Zanini (2013). It is useful to find the best representation. It receives in input the output of the `basis_hica` function.

Value

energy	A matrix with maxcomp rows and p-1 columns, where p is the number of variables. In position (i,j) it contains the energy of the best i-dimensional space for the jth level of the tree. Only the last nlevel columns are filled.
components	A matrix with maxcomp rows and p-1 columns, where p is the number of variables. In position (i,j), it contains the index of th ith basis element for jth level of the tree. Only the last nlevel columns are filled.
HICA.obj	The same object, output of the function basis_hica, provided in input.

Author(s)

Piercesare Secchi, Simone Vantini, and Paolo Zanini

References

Secchi, Vantini, and Zanini (2013)

See Also

basis_hica, similarity_hica, extract_hica

---

extract_hica *Extraction of score and loading matrices.*

Description

This function extracts the score matrix and the loading matrix given the dimension of the subspace considered and the level of the tree chosen. Furthermore it provides the cumulant energies for the subspace extracted.

Usage

```
extract_hica(energy.obj, comp, level)
```

Arguments

energy.obj	An object provided by the function energy_hica.
comp	Dimension of the subspace.
level	Level of the tree.

Value

X	data matrix.
S	score data matrix.
C	loading matrix. Each column represent a basis element.
cum.energy	cumulant energy for the subspace extracted.

Author(s)

Piercesare Secchi, Simone Vantini, and Paolo Zanini.

References

Secchi, Vantini, and Zanini (2013).

See Also

basis_hica, similarity_hica, energy_hica

`similarity_hica` *Estimate of the similarity matrix*

Description

This function provides an estimate of the similarity matrix of the original data, before performing HICA algorithm.

Usage

```
similarity_hica(X, dim.subset = 512)
```

Arguments

`X`	Data matrix with nrow(X) observations and ncol(X) variables.
`dim.subset`	The dimension of the subset used for the evaluation of the similarity index (i.e., distance correlation). If this it is greater than nrow(X) all the observations are used, unless a random subset of `dim.subset` observations is used. The default value is set to 512.

Details

This function is auxiliary for the basis_hica function. Indeed its output is the estimate of the similarity matrix at the first step of the algorithm.

Value

`similarity_matrix`	similarity matrix of the original data.
`subset`	subset used for the evaluation of distance correlation between variables.

Note

The distance correlation is evaluated through the function wdcor of the package "extracat". It becomes computational unfeasible if the number of observations is too large. For this reason it is possibile to choose the dimension of the subsample to be used in the evaluation of the similarity matrix. By default the dimension is set to 512.

Author(s)

Piercesare Secchi, Simone Vantini, and Paolo Zanini.

References

Secchi, Vantini, and Zanini (2013).

See Also

basis_hica, energy_hica, extract_hica

Index

Conclusion

This manuscript focused on the resolution of Blind Source Separation (BSS) problems for high-dimensional, massive, and complex data. In particular we aimed to obtain solutions with a phenomenological interpretation according to the physics of the problem under study. We analyzed how, in literature, this purpose is not achieved through model-free methods (as, for instance, Principal Component Analysis) but making statistical or mathematical assumptions on the quantities of interest. In particular we considered three different approaches. The first one regarding statistical assumptions on the random sources (e.g, independence), the second one related on mathematical assumptions on the mixing matrix (e.g., sparsity or multi-scale property) and the last one based on the introduction of some constraints, generally provided by an a priori knowledge about the solution, on both the source matrix and the mixing matrix (e.g., nonnegativity constraint). Hence we developed our work in three different parts, each related to one of these approaches.

In the first part of the thesis we addressed our attention to the multi-resolution analysis of complex datasets. In particular we developed a new method, named Hierarchical Independent Component Analysis (HICA). It is built through the integration between treelets and Independent Component Analysis (ICA), and provides a multi-scale nonorthogonal data-driven basis. We also applied this new method to an EEG dataset of patients affected by alcoholism. We analyze one patient and we show how our method is able to provide meaningful and interpretable results and to give noticeable improvements with respect to other popular BSS methods.

In the second part we focused on the ICA framework. In particular we dealt with those situations where the sources are spatial stochastic processes on lattices. We built a new method, named spatial colored Independent Component Analysis (scICA), to take into account the spatial dependence of the observations. The key point of our method is to work in the frequency domain instead of the spatial domain, and the key statistical tool used to implement this method is the Whittle likelihood. We also applied the proposed method to a mobile phone traffic dataset, provided by Telecom

Italia. We compared scICA and fastICA (the more famous algorithm implementing ICA) algorithms and we show how our method is able, through the information carried out by the spatial dependence structure, to reveal some interesting population and mobility characteristics.

In the third part we faced some problems of the Alternating Least Square (ALS), an algorithm to solve Nonnegative Matrix Factorization (NMF) problem. Specifically we focused on two main problems to deal with. The first one is related to the implementation of the constraints. Indeed, nonnegativity is often not the only constraint one want to impose. In particular, equality constraints play an important role in a lot of applications. However we showed that a specific treatment for these kind of vincola is needed. Hence we proposed a procedure to deal with this kind of constraint (i.e., we introduce a suitable penalty in the objective function). The second aspect related to ALS is the analysis of functional dataset. In particular we dealt with the misalignment problem and we analyzed how misalignment can affect the solution. Then we present a real application where we studied some chemical mixtures through their gas chromatograms in order to retrieve the chromatograms of the original compounds generating the mixtures and the concentrations in each mixture.

All the analyses have been carried out through the statistical software R. In particular we also developed an R package, fastHICA, implementing the HICA algorithm we proposed. It has been published on the CRAN repository. The development of an R package implementing the scICA algorithm is also in order, although it not available on the CRAN repository yet. In the last part of this manuscript we presented the help of the fastHICA package.

Bibliography

[1] Ahas, R. and Mark, U. (2005). Location based services-new challenges for planning and public administration. *Futures* **37** 547–561.

[2] Allassoniere, S. and Younes, L. (2012). A stochastic algorithm for probabilistic independent component analysis. *The Annals of Applied Statistics 6* 125–160

[3] Altman, N.S. and Villareal, J.C. (2004). Self-modeling regression for longitudinal data with time-invariant covariates. *Canadian Journal of Statistics* **32** 251–268.

[4] Amari, S., Cichocki, A. and Yang, H. H. (1996). A new Learning Algorithm for Blind Signal Separation. *Neural Computation* **11** 1875–1883.

[5] Anbumalar, S., Anandanatarajan, R. and Rameshbabu, P. (2013). Sparse Non-negative Matrix Factorization and its Application in Overlapped Chromatograms Separation. *International Journal of Computer Applications* **63**.

[6] Attias, H. (1999). Independent Factor Analysis. *Neural Computation* **11** 803–851.

[7] Azzouz, T. and Tauler, R. (2008). Application of multivariate curve resolution alternating least squares (MCR-ALS) to the quantitative analysis of pharmaceutical and agricultural samples. *Talanta* **74** 1201–1210.

[8] Bache, K. and Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[9] Bell, A. and Sejnowski, T. J. (1995). An Information-Maximization Approach to Blind Source Separation and Blind Deconvolution. *Neural Computation* **7** 1129–1159.

[10] Byrd, R.H., Lu, P., Nocedal, J. and Zhu, C. (1995). A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM Journal on Scientific and Statistical Computing* **16** 1190-1208.

[11] Chicoki, A. and Zdunek, R. (2007). Regularized Alternating Least Squares Algorithms for Non-negative Matrix/Tensor Factorization. *Proceedings of the Fourth International Symposium on Neural Networks*.

[12] Comon, P. and Jutten, C. (2010). *Handbook of Blind Source Separation: Independent Component Analysis and Applications*. Academic Press.

[13] Cressie, N. (1993). *Statistics for Spatial Data*. Wiley, New York.

[14] Crujeiras, R. M. and Fernandez-Casal, R. (2010). On the estimation of the spectral density for continuos spatial processes. *Statistics: A Journal of Theoretical and Applied Statistics* **44** 587–600.

[15] de Juan, A. and Tauler, R. (2003). Chemometrics applied to unravel multicomponent processes and mixtures. Revisiting multivariate resolution. *Analytica Chimica Acta* **500** 195–210

[16] De Luca, M., Beckmann, C. F., De Stefano, N., Matthews, P. M. and Smith, S. M. (2006). fMRI resting state networks define distinct modes of long-distance interactions in the human brain. *Neuroimage* **29** 1359–1367.

[17] Fan, J. and Kreutzberger, E. (1998). Local smoothing for Spectral Density Estimation. *Scandinavian Journal of Statistics* **25** 359–369.

[18] Fuentes, M. (2002). Spectral methods for nonstationary spatial processes. *Biometrika* **89** 197–210.

[19] Gebizlioglu, O. L. (1998). Spatial processes: modelling, estimation, and hypothesis testing. *Commun. Fac. Sci. Univ. Ank. Ser. A1* **37** 67–94.

[20] Gonzalez, M. C., Hidalgo, C. A. and Barabasi, A. L. (2008). Understanding individual human mobility patterns. *Nature* **453** 779–782.

[21] Hyvarinen, A. and Oja, E. (2000). Independent Component Analysis: Algorithms and Applications. *Neural Networks* **13** 411–430.

[22] Hyvarinen, A., Karhunen, J. and Oja, E. (2001). *Independent Component Analysis*. Wiley, New York.

[23] Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer, New York.

[24] Kaziska, D. and Srivastava, A. (2007). Gait-based human recognition by classification of cyclostationary processes on nonlinear shape manifolds. *Journal of the American Statistical Association* **102** 1114–1128.

[25] Kaufmann, V. (2002). *Re-thinking mobility contemporary sociology*. Ashgate.

[26] Lawton, W.H., Sylvestre, E.A. and Maggio, M.S. (1972). Self modeling nonlinear regression. *Tecnhometrics* **14**, 513–532.

[27] Lee, D. D. and Seung, S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature.* **401**, 788–791.

[28] Lee, D. D. and Seung, S. (2001). Algorithms for Non-Negative Matrix Factorization. *Advances in Neural Information Processing Systems 13: Proceedings of the 2000 Conference*. MIT press 556–562.

[29] Lee, A. B., Nadler, B. and Wasserman L. (2008). Treelets - an adaptive multi-scale basis for sparse unordered data. *The Annals of Applied Statistics* **2** 435–471.

[30] Lee, S., Shen, H., Truong, Y., Lewis, M. and Huang, X. (2011). Independent Component Analysis Involving Autocorrelated Sources With an Application to Functional Magnetic Resonance Imaging. *Journal of American Statistical Association* **106** 1009–1024.

[31] Lindstrom, M.J. and Bates, D.M. (1990). Nonlinear mixed effects models for repeated measures data. *Biometrics* **46** 673–687.

[32] Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11** 674–693.

[33] Manfredini, F., Pucci, P., Secchi, P., Tagliolato, P., Vantini, S. and Vitelli, V. (2012). Treelet decomposition of mobile phone data for deriving city usage and mobility pattern in the Milan urban region. *MOX report 25/2012*, Department of Mathematics, Politecnico di Milano.

[34] McKeown, M. J. and Sejnowski, T. J. (1998). Independent Component Analysis of fMRI Data: Examining the Assumptions. *Human Brain Mapping* **6** 368–372.

[35] Moulines, E., Cardoso, J. and Gassiat, E. (1997). Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models. *IEEE International Conference on Acoustics, Speech and Signal Processing, 1997* **5** 3617–3620.

[36] OECD (2006). *Territorial Reviews: Milan, Italy*. OECD Publishing.

[37] Ogden, R. T. (1997). *Essential Wavelets for Statistical Applications and Data Analysis*. Birkhauser, Boston.

[38] Pham, D. and Garat, P. (1997). Blind Separation of Mixture of Independent Sources Through a Quasi-Maximum Likelihood Approach. *IEEE Transactions on Signal Processing* **45** 1712–1725.

[39] Ramsey, J. O. and Silverman, B. W. (2005). *Functional Data Analysis*. Springer.

[40] Ratti, C., Pulselli, R. M., Williams, S. and Frenchman, D. (2006). Mobile Landscapes: using location data from cell phones for urban analysis. *Environment and Planning B: Planning and Design* **33** 727–748.

[41] Sangalli, L.M., Secchi, P., Vantini, S. and Vitelli, V. (2010). K-mean alignment for curve clustering. *Computational Statistics and Data Analysis* **54** 1219–1233.

[42] Sangalli, L.M., Secchi, P., Vantini, S. and Vitelli, V. (2010). Functional clustering and alignment methods with applications. *Communications in Applied and Industrial Mathematics* **1** 205–224.

[43] Secchi, P., Vantini, S. and Vitelli, V. (2012). A Case Study on Spatially Dependent Functional Data: the Analysis of Mobile Network Data for the Metropolitan Area of Milan. *MOX report 43/2012*, Department of Mathematics, Politecnico di Milano.

[44] Sharbrough, F., Chatrian, G. E., Lesser, R.P., Lders, H., Nuwer, M. and Picton, T.W. (1991). American Electroencephalographic Society Guidelines for Standard Electrode Position Nomenclature. *Journal of Clinical Neurophysiology* **8** 200–202.

[45] Sheller, M. and Urry, J. (2006). The new mobilities paradigm. *Environment and Planning A* **38** 207–226.

[46] Snodgrass, J. G. and Vanderwart, M. (1980). A standardized set of 260 pictures: norms for the naming agreement, familiarity, and visual complexity. *Journal of Experimental Psychology: Human Learning and Memory* **6** 174–215.

[47] Szkely, G. J., Rizzo, M. L. and Bakirov, N. K. (2007). Measuring and testing dependence by correlation of distances. *The Annals of Statistics* **35** 2769–2794.

[48] Tauler, R. and Barcelo, D. (1993). Multivariate Curve Resolution applied to liquid-chromatograpy diode-array detection. *Trends in Analytical Chemistry* **12** 319–327.

[49] Tong, L., Liu, R. and Huang, Y. (1991). Indeterminacy and Identifiability of Blind Identification. *IEEE Transactions on Circuits and Systems* **38** 499–509.

[50] Whittle, P. (1952). Some results in time series analysis. *Skandivanisk Aktuarietidskrift* **35** 48–60.

[51] Zhang, X.L., Begleiter, H., Porjesz, B., Wang, W. and Litke, A. (1995). Event related potentials during object recognition tasks. *Brain Research Bulletin* **38** 531–538.

[52] Ziehe, A. and Muller, K. (1998). TDSEP – an efficient algorithm for blind separation using time structure. *Proceedings of ICANN 98* 675–680.

[53] R Development Core Team (2011). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.