A statistical and computational framework for multiblock and multiway data analysis

Arnaud Gloaguen

université
PARIS-SACLAY

A statistical and computational framework for multiblock and multiway data analysis

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, Sciences et Technologies de l'Information et de la Communication (STIC)
Spécialité de doctorat: Traitement du Signal et des Images
Unité de recherche: Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, 91190, Gif-sur-Yvette, France
Université Paris-Saclay, CEA, Neurospin, 91191, Gif-sur-Yvette, France
Référent: Faculté des sciences d'Orsay

Thèse présentée et soutenue à Gif-sur-Yvette, le 23 septembre 2020, par

Arnaud GLOAGUEN

Composition du jury:

Julie Josse Professeure, Ecole Polytechnique, (CMAP)	Présidente
Eric Lock Maître de Conférence, University of Minnesota	Rapporteur & Examinateur
Hua Zhou Maître de Conférence, University of California, Los Angeles (UCLA)	Rapporteur & Examinateur
Christophe Ambroise Professeur, Université d'Evry Val d'Essonne, Université Paris-Saclay	Examinateur
Arthur Tenenhaus Professeur, CentraleSupélec, Université Paris-Saclay	Directeur de thèse
Vincent Frouin Directeur de Recherche, CEA NeuroSpin, Université Paris-Saclay	Co-directeur de thèse & Examinateur
Laurent Le Brusquet Enseignant-Chercheur, CentraleSupélec, Université Paris-Saclay	Invité
Cathy Philippe Ingénieure de Recherche, CEA NeuroSpin, Université Paris-Saclay	Invitée

Thèse de doctorat

NNT: 2020UPASG016

To my grandfather, Emile Gloaguen

Acknowledgments

First, I would like to acknowledge all the people that kindly supported me all along these years and made this work possible.

All my gratitude goes towards my thesis advisors/directors that followed me from the beginning until the very end of this PhD journey: Arthur Tenenhaus, Cathy Philippe, Laurent Le Brusquet and Vincent Frouin. Your passion for sciences, teaching and sharing your knowledge was a great source of inspiration and kept me motivated all along this thesis. Thank you Arthur for guiding me and for all these «Do you have a minute» talks that ended up being hours of inspiring discussions. I admire your optimism and generosity in your work and in life in general. Thank you Cathy for your exciting lessons about genetics and for always challenging me on the analysis I could make. Thank you Laurent for your availability and all the relevant advice you gave me on optimization. Thank you Vincent for welcoming me in the Brainomics team and for your enthusiasm to push forward the boundaries of imaging-genetics with the developments I could make.

I would also like to thank the people that reviewed/evaluated my PhD manuscript: Hua Zhou, Eric Frazer Lock, Julie Josse and Christophe Ambroise. Thank you for taking some of your precious time to review my work. It is a deep honor to have you as my jury members.

Thank you to all my colleagues from the Brainomics team at NeuroSpin and the Signal and Systems department of CentraleSupélec. Thank you also to the iCONICS team from the Brain Institute that hosted me here and there. Thank you to Léonie, Milad, Slim, the running team of NeuroSpin, the Geep's Lab for their tarot card games, Ghislaine and her contagious passion for neurosciences, Giulia for helping me analyze her wonderful EEG data, Loubna for our optimization talks, Vincent and Hervé for the inspiring discussions we had on data analysis, Nicolas and Raphael for this amazing conference trip we had to Venice. You all participated in making my working environment so wonderful.

I would also like to thank all my friends that were always supportive during this PhD: the «Switch'On», the «Dégaine», the «TMB running team», Jerôme, Aubrey and last but not least my amazing PhD roommates: Vincent and Hippolyte.

A particular thanks to all my family that encouraged me during all these years and of course to you, Sarah, my first supporter, sharing my life with you is a constant source of happiness.

Contents

List of Figures

List of Tables

List of Algorithms

General Introduction

Context & Motivations

MULTIDISCIPLINARY approaches are now common in scientific research and provide multiple and heterogeneous sources of measurements of a given phenomenon. These sources can be viewed as a collection of interconnected datasets acquired on the same set of individuals. The statistical analysis of multi-source datasets introduces new degrees of freedom, which raise questions beyond those related to exploiting each source separately. In the literature, this paradigm can be stated under several names as "learning from multimodal data", "data integration", "data fusion" or "multiblock data analysis". Typical examples are found in a large variety of fields such as biology, chemistry, sensory analysis, marketing, food research, where the common general objective is to identify variables of each block that are active in the relationships with other blocks. For instance, neuroimaging is increasingly recognized as an intermediate phenotype (endophenotype) to understand the complex path between genetics and behavioral or clinical phenotypes. In this imaging-genetics context, the goal is primarily to identify a set of genetic biomarkers that explains some neuroimaging variability which implies some modification of the behavior. The high number of measurements ($\sim$1M) in both genetic and neuroimaging data involves the computation of billions of associations.

In addition to this global multi-source structure, each source can be represented in the form of higher-order tensors or matrices. For instance, an anatomical Magnetic Resonance Imaging (MRI) is a three-dimensional image of the brain, so, by nature, a tensor. Another application is found in Electroencephalography (EEG) or Magnetoencephalography (MEG) that gives access respectively to the electric or magnetic brain waves. These waves are acquired from multiple sensors at the same time, leading to spatiotemporal data. When these two-dimensional spatiotemporal data are measured on different individuals, they become intrinsically a tensor. Taking into account the possible tensor structure of a source is mandatory in order to avoid altering the natural organization of the data and risking a loss of information.

The principle of parsimony is central to many areas of science: the simplest explanation to a given phenomenon should be preferred over more complicated ones. In statistics, it takes the form of variable selection. For instance, in the context of an imaging-genetic study of a neurodegenerative disease, it allows a subset of genetic variants to be identified as involved in the atrophy of specific regions of the brain.

Dedicated modeling algorithms able to cope with the inherent structural properties of such multi-source datasets are therefore mandatory for harnessing their complexity and provide relevant and robust information. The overall objective of the analysis of multi-source datasets includes extracting relevant information within massive amounts of variables spread across different sources, reducing dimensionality, synthesizing the information in an understandable way and displaying it for interpretation purposes.

Thesis Outline

The development of multivariate statistical methods for multi-source data constitutes the core of this work. All these developments find their foundations on Regularized Generalized Canonical Correlation Analysis (RGCCA) and as a matter extend it. RGCCA is a flexible framework for multiblock data analysis and grasps in a single optimization problem many well known multiblock methods. The RGCCA algorithm consists in a single yet very simple update repeated until convergence. If this update is gifted with certain conditions, the global convergence of the procedure is guaranteed (i.e. convergence of the algorithm towards a stationary point regardless the initialization). Throughout this work, we tried to preserve both the flexibility and the simplicity of the optimization framework of RGCCA. The second part of this work illustrates the versatility and usefulness of the proposed methods on five various studies: two imaging-genetic, two electroencephalography and one Raman Microscopy studies. In all these analyses, a focus is made on the interpretation of the results that is eased by considering explicitly the multiblock, tensor and sparse structures. This thesis is organized as follows:

Chapter 1: *Background Methods*

This chapter starts by describing Regularized Generalized Canonical Correlation analysis followed by an overview of multiblock methods - either particular case of RGCCA or not. In a second part, the mathematical foundations of tensor analysis are provided. Notations and operators used in the tensor literature are recalled. A brief presentation of the most popular multiway models is also given. A very general yet very simple optimization framework concludes this chapter. This optimization framework provides the algorithmic foundations of our developments. As we will see, this optimization framework offers a systematic approach for constructing globally convergent algorithms.

Chapter 2: *Global Regularized Generalized Canonical Correlation Analysis*

The objective of RGCCA is to find block components summarizing the relevant information between and within the blocks. RGCCA belongs to the family of the sequential multiblock component methods. It means that the components of each block are determined sequentially (R successive optimization problems have to be solved to extract R components per block). From an optimization point of view this strategy seems to be sub-optimal and we present, in Chapter 2, global RGCCA that allows all the components to be extracted simultaneously by solving a single optimization problem. The global RGCCA optimization problem is presented and we show that the corresponding algorithm is globally convergent. Sequential RGCCA and global RGCCA are compared on simulation experiments.

Chapter 3: *Multiway Generalized Canonical Correlation Analysis (MGCCA)*

Multiway Generalized Canonical Correlation Analysis (MGCCA) extends RGCCA to the joint analysis of a collection of higher-order tensors or matrices. Sequential MGCCA and global MGCCA optimization problems are proposed. For the sequential procedure, two strategies are developed to obtain higher level components. We propose two algorithms for global and sequential MGCCA that are globally convergent. The two approaches are compared on simulation experiments.

Chapter 4: *Structured Sparse Generalized Canonical Correlation Analysis*

A challenge in the multivariate analysis of heterogeneous datasets containing a large number of variables is the selection of relevant features. A version of RGCCA, called Sparse Generalized Canonical Correlation Analysis (SGCCA), enables to select the variables that interact the most between blocks. A novel and fast algorithm is derived to solve the SGCCA optimization problem efficiently. We demonstrate that this new algorithm is globally convergent. The variable selection of SGCCA relies on the ℓ_1 penalty which operates without further knowledge on the possible intra-block interactions between variables. SGCCA was thus enhanced by introducing structured sparse penalties (like group LASSO, sparse group, fused or elitist LASSO penalty) into the optimization process of SGCCA.

Chapter 5: *Multiblock and/or Multiway data studies*

Chapter 5 demonstrates the versatility and usefulness of RGCCA and MGCCA on five multiblock and/or multiway datasets. The first study investigates the influence of genetics on the normal aging brain from the United Kingdom Biobank (UKB) cohort. The second one is an imaging-genetic study on the Alzheimer's disease Neuroimaging Initiative (ADNI) that aims at understanding some mechanisms of the disease through several modalities (Genetics, Transcriptomics, longitudinal MRI, Clinical factors). The third study aims at analyzing the efficiency of a moisturizer from Raman microscopy. The two last studies aim at identifying brain areas implicated in the process of discrimination between close syllables in two- to three-month-old human infants from Electroencephalography (EEG).

Chapter 1

Background Methods

Chapter Outline

The background concepts used throughout this manuscript are presented in this chapter. Three main topics are addressed: multiblock component methods, multiway methods and the optimization framework under which the main algorithms of this work are developed.

1.1 Introduction

The field of multiblock data analysis starts with Canonical Correlation Analysis (CCA) [Hotelling, 1936] for analyzing the relationships between two sets of variables. Since then, the state of the art constantly evolved to cope with the challenges of analyzing the relationships between more than two sets of variables. Regularized Generalized Canonical Correlation Analysis (RGCCA) was proposed as a general framework to deal with such challenges. RGCCA, which is the starting point of all the methodological contributions described throughout this document, is detailed in Section 1.2.

One of the contribution of this work is the extension of RGCCA to multiway data. Multiway notations and most popular multiway models are detailed in sections 1.3 and 1.4 respectively.

Section 1.5 presents the optimization framework used to solve the RGCCA optimization problem. This framework offers the algorithmic foundations of the methodological developments of this work.

1.2 Multiblock Component Methods

In this section, we consider the case of several blocks of variables measured on the same set of individuals. The objective of multiblock component methods is to find block components summarizing the relevant information between and within the blocks. Each block component is defined as a weighted sum of the block variables. We consider methods where the weights are obtained by solving some optimization problem.

The purpose of the present section is to show that a remarkably large number of sequential multiblock component methods appear to be special cases of RGCCA. For all these methods, the same simple RGCCA algorithm can be used (with proper setting of the parameters).

1.2.1 Regularized Generalized Canonical Correlation Analysis (RGCCA)

Let $\mathbf{X}_1, \ldots, \mathbf{X}_l, \ldots, \mathbf{X}_L$ be a collection of L data matrices. Each $I \times J_l$ data matrix $\mathbf{X}_l = [\mathbf{x}_{l1}, \ldots, \mathbf{x}_{lJ_l}]$ is called a source or a block and represents a set of J_l variables observed on I individuals. The number and the nature of the variables may differ from one block to another, but the individuals must be the same across blocks. We assume that all variables are centered. The objective of RGCCA is to estimate block components $\mathbf{y}_l = \mathbf{X}_l \mathbf{w}_l$, $l = 1, \ldots, L$ (where the weight vector $\mathbf{w}_l$ is a column-vector with J_l elements) summarizing the relevant information between and within the blocks. The most recent formulation of the RGCCA optimization problem [Tenenhaus et al., 2017] is:

$$\begin{aligned} &\max_{\mathbf{w}_1, \ldots, \mathbf{w}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1} \mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) \\ &\text{s.t. } \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l = 1,\ l = 1, \ldots, L \end{aligned} \tag{1.1}$$

where:

- The design matrix $\mathbf{C} = \{c_{lk}\}$ is a symmetric $L \times L$ matrix of non-negative elements describing the network of connections between blocks that the user wants to take into account. Usually, $c_{lk} = 1$ for two connected blocks and 0 otherwise. These connections are undirected and act as soft correlations between linear combinations of variables of connected blocks. They can be used as an exploratory tool rather than a modeling one.

- The scheme function $g(x)$ is defined as any continuously differentiable convex function. Typical choices of g are the identity (leading to maximizing the sum of covariances between block components, a.k.a. Horst scheme), the absolute value (yielding maximization of the sum of the absolute values of the covariances, a.k.a. centroid scheme) or the square function (thereby maximizing the sum of squared covariances, a.k.a. factorial scheme).

- $\mathbf{M}_l$ is any $J_l \times J_l$ positive definite matrix.

In [Tenenhaus and Tenenhaus, 2011], the authors consider the following optimization problem:

$$\begin{aligned} &\max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) \\ &\text{s.t. } (1-\tau_l)\mathrm{var}(\mathbf{X}_l\mathbf{w}_l) + \tau_l\|\mathbf{w}_l\|^2,\ l = 1,\ldots,L \end{aligned} \tag{1.2}$$

where τ_l is the shrinkage parameter varying between 0 and 1. This optimization problem is recovered with optimization problem (1.1) by setting $\mathbf{M}_l = \tau_l\mathbf{I} + (1-\tau_l)I^{-1}\mathbf{X}_l^\top\mathbf{X}_l$. This constraint allows to give two interpretations of the RGCCA optimization problem.

Firstly, the shrinkage parameters τ_l interpolate smoothly between maximizing the covariance and maximizing the correlation. Setting the τ_l to 0 will force the block components to unit variance, in which case the covariance criterion boils down to the correlation. The correlation criterion is better in explaining the correlated structure across datasets, thus discarding the variance within each individual dataset. Setting τ_l to 1 will normalize the block weight vectors, which leads to the covariance criterion. A value between 0 and 1 will lead to a compromise between the two first options. Secondly, [Ledoit and Wolf, 2004] considers $\mathbf{M}_l$ as a shrinkage estimate of the true covariance matrix for block l. Various formulas for finding an optimal shrinkage constant τ_l have been proposed (see, for example, [Schäfer and Strimmer, 2005]).

Guidelines describing how to use RGCCA in practice are provided in [Garali et al., 2017].

The RGCCA algorithm is not detailed in this chapter but fully described in Chapter 2. Moreover, the RGCCA optimization problem (1.1) extracts only one component per block. The next chapter presents two strategies to extract more than one component per block.

1.2.2 Special Cases of RGCCA

From optimization problem (1.2), the term «Generalized» in the acronym of RGCCA embraces at least four notions. The first one relates to the generalization of two-block methods - including Canonical Correlation Analysis [Hotelling, 1936], Interbattery Factor Analysis [Tucker, 1958] and Redundancy Analysis [Van den Wollenberg, 1977] - to three or more sets of variables. The second one relates to the ability of taking into account some hypotheses on between-block connections: the user decides which blocks are connected and which are not. The third one relies on the choices of the shrinkage parameters allowing to capture both correlation or covariance-based criteria. The fourth one relates to the function g that enables to consider different functions of the covariance.

This generalization is embodied by a triplet of parameters : $(\mathrm{g}, \tau_l, \mathbf{C})$ and by the fact that an arbitrary number of blocks can be handled. This triplet of parameters offers a flexibility to RGCCA and allows to recover several known methods as particular cases, thus subsuming fifty years of multiblock component methods. Table 1.1 gives the correspondences between the triplet $(\mathrm{g}, \tau_l, \mathbf{C})$ and the corresponding methods. For a complete overview see [Tenenhaus et al., 2017].

In general, and especially for the covariance-based criterion, the data blocks might be pre-processed to ensure comparability between variables and blocks. To make variables comparable, standardization is applied (zero mean and unit variance). To make blocks comparable, a strategy is to divide each block by the square root of its number of variables. This two-step procedure leads to $\mathrm{Trace}(I^{-1}\mathbf{X}_l^\top \mathbf{X}_l) = 1$ for each block (i.e. the sum of the eigenvalues of the correlation matrix of $\mathbf{X}_l$ is equal to 1 whatever the block).

The next section presents several widely used multiblock methods. One main difference between those approaches and RGCCA is that the block components are extracted simultaneously. A version of RGCCA that enables to extract all the components at once is presented in the next chapter.

1.2.3 Alternative approaches

1.2.3.1 Simultaneous Component Analysis

Simultaneous Component Analysis (SCA) [Kiers, 1990, Kiers and ten Berge, 1989, Ten Berge et al., 1992], considers the following rank-R model:

$$\mathbf{X}_l = \mathbf{T}\mathbf{W}_l^\top + \mathbf{E}_l, \qquad l = 1, \ldots, L \tag{1.3}$$

where the $I \times R$ block components matrix $\mathbf{T}$ is shared across the the L datasets, $\mathbf{W}_l$ is the $J_l \times R$ specific block weight matrix and $\mathbf{E}_l$ is the $I \times J_l$ residual matrix. To estimate the model parameters $\mathbf{T}$ and $\mathbf{W}_l$, the following optimization problem is proposed:

$$\min_{\mathbf{T}, \mathbf{W}_1, \ldots, \mathbf{W}_L} \sum_{l=1}^{L} \left\| \mathbf{X}_l - \mathbf{T}\mathbf{W}_l^\top \right\|_F^2 . \tag{1.4}$$

An Alternative Least Squares (ALS) procedure is used to solve the optimization problem (1.4) [ten Berge, 1993]. It consists in alternating between the minimization of (1.4) over $\mathbf{T}$, keeping $\mathbf{W}_1, \ldots, \mathbf{W}_L$ fixed, and the other way round.

Table 1.1 – Special cases of RGCCA in a situation of $L \geq 2$ blocks. When τ_{L+1} is introduced, it is assumed that $\mathbf{X}_1, \ldots, \mathbf{X}_L$ are connected to a $(L+1)^{th}$ block defined as the concatenation of the blocks, $\mathbf{X}_{L+1} = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_L]$ and that τ_{L+1} corresponds to the shrinkage parameter associated with $\mathbf{X}_{L+1}$.

Methods	g(x)	τ_l	$\mathbf{C}$
Canonical Correlation Analysis [Hotelling, 1936]	x	$\tau_1 = \tau_2 = 0$	$\mathbf{C}_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
Interbattery Factor Analysis [Tucker, 1958]	x	$\tau_1 = \tau_2 = 1$	$\mathbf{C}_1$
Redundancy Analysis [Van den Wollenberg, 1977]	x	$\tau_1 = 1$ and $\tau_2 = 0$	$\mathbf{C}_1$
SUMCOR [Horst, 1961]	x	$\tau_l = 0,\ l = 1, \ldots, L$	$\mathbf{C}_2 = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ 1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 1 & \ldots & 1 & 1 \end{pmatrix}$
SSQCOR [Kettenring, 1971]	x^2	$\tau_l = 0,\ l = 1, \ldots, L$	$\mathbf{C}_2$
SABSCOR [Hanafi, 2007]	$\lvert x \rvert$	$\tau_l = 0,\ l = 1, \ldots, L$	$\mathbf{C}_2$
SUMCOV-1 [Van de Geer, 1984]	x	$\tau_l = 1,\ l = 1, \ldots, L$	$\mathbf{C}_2$
SSQCOV-1 [Hanafi and Kiers, 2006]	x^2	$\tau_l = 1,\ l = 1, \ldots, L$	$\mathbf{C}_2$
SABSCOV-1 [Kramer, 2007] [Tenenhaus and Tenenhaus, 2011]	$\lvert x \rvert$	$\tau_l = 1,\ l = 1, \ldots, L$	$\mathbf{C}_2$
SUMCOV-2 [Van de Geer, 1984]	x	$\tau_l = 1,\ l = 1, \ldots, L$	$\mathbf{C}_3 = \begin{pmatrix} 0 & 1 & \ldots & 1 \\ 1 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 1 \\ 1 & \ldots & 1 & 0 \end{pmatrix}$
SSQCOV-2 [Hanafi and Kiers, 2006]	x^2	$\tau_l = 1,\ l = 1, \ldots, L$	$\mathbf{C}_3$
Generalized CCA [Carroll, 1968]	x^2	$\tau_l = 1,\ l = 1, \ldots, L+1$	$\mathbf{C}_4 = \begin{pmatrix} 0 & \ldots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \ldots & 0 & 1 \\ 1 & \ldots & 1 & 0 \end{pmatrix}$
Generalized CCA [Carroll, 1968]	x^2	$\tau_l = 0,\ l = 1, \ldots, L_1,$ $\tau_l = 1,\ l = L_1 + 1, \ldots, L,$ and $\tau_{L+1} = 0$	$\mathbf{C}_4$
Hierarchical PCA [Wold et al.]	x^4	$\tau_l = 1,\ l = 1, \ldots, L,$ and $\tau_{L+1} = 0$	$\mathbf{C}_4$
Multiple Co-Inertia Analysis [Chessel and Hanafi, 1996]	x^2	$\tau_l = 1,\ l = 1, \ldots, L,$ and $\tau_{L+1} = 0$	$\mathbf{C}_4$
PLS path modeling-mode B [Wold, 1982]	$\lvert x \rvert$	$\tau_l = 0,\ l = 1, \ldots, L,$	$c_{lk} = 1$ for two connected blocks and 0 otherwise

1.2.3.2 SUM-PCA

To avoid the rotational indeterminacies of SCA it is possible to estimate model (1.3) under the identification constraints $\mathbf{T}^\top\mathbf{T} = \mathbf{I}_R$ which gives SUM-PCA [Smilde et al., 2003]. Several constraints used in the context of SCA to avoid such indeterminacies of the solution are discussed in [Deun et al., 2009]. Interestingly, the SUM-PCA solution can be found by performing a Principal Component Analysis (PCA) on the concatenated data matrix $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_L]$. One major limitation of SUM-PCA and SCA is that it does not allow for specific block components; which is the purpose of Joint and Individual Variation Explained (JIVE) [Lock et al., 2013] to be described.

1.2.3.3 Joint and Individual Variation Explained (JIVE)

Performing a rank-r_l PCA on each block individually corresponds to the model:

$$\mathbf{X}_l = \mathbf{T}_l\mathbf{W}_l^\top + \mathbf{E}_l, \qquad l = 1, \ldots, L, \tag{1.5}$$

where now each $\mathbf{T}_l$ is the $I \times J_l$ matrix of block components specific to each block.

JIVE can be viewed as an intermediate model between (1.3) and (1.5) as it corresponds to:

$$\mathbf{X}_l = \mathbf{T}_0\mathbf{W}_{0l}^\top + \mathbf{T}_l\mathbf{W}_l^\top + \mathbf{E}_l, \qquad l = 1, \ldots, L \tag{1.6}$$

where the block components matrix $\mathbf{T}_0$ is shared among all blocks, and the block components matrices $\mathbf{T}_l$ are block-specific so that $\mathbf{T}_0^\top\mathbf{T}_l = \mathbf{0}$. To estimate the model parameters $\mathbf{T}_0$, $\mathbf{W}_{0l}$, $\mathbf{T}_l$ and $\mathbf{W}_l$ the following optimization problem is considered:

$$\min_{\mathbf{J},\mathbf{A}_1,\ldots,\mathbf{A}_L} \|\mathbf{X} - \mathbf{J} - [\mathbf{A}_1, \ldots, \mathbf{A}_L]\|_F \tag{1.7}$$

$$\text{s.t.} \begin{cases} \text{rank}(\mathbf{J}) &= r, \\ \text{rank}(\mathbf{A}_l) &= r_l, \qquad l = 1, \ldots, L, \\ \mathbf{J}^\top\mathbf{A}_l &= \mathbf{0}_{J\times J_l}, \quad l = 1, \ldots, L, \end{cases} \tag{1.8}$$

where $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_L]$ is a $I \times J$ matrix ($J = \sum_l J_l$), $\mathbf{J} \in \mathbb{R}^{I\times J}$ is the joint structure and $\mathbf{A}_l \in \mathbb{R}^{I\times J_l}$ is the individual one.

The algorithm proposed to minimize (1.7) subject to (1.8) consists in alternating between the minimization of (1.7) subject to (1.8) over $\mathbf{J}$, keeping $\mathbf{A}_1, \ldots, \mathbf{A}_L$ fixed, and the other way round. Each optimization problem is solved by the Singular Value Decomposition (SVD) of a certain matrix. Hence, it means that $\mathbf{J} = \mathbf{U}\mathbf{\Delta}\mathbf{V}^\top$, where $\mathbf{U}$ and $\mathbf{V}$ are two orthonormal matrices of size $I \times r$ and $J \times r$ respectively, $\mathbf{\Delta}$ is a diagonal matrix of size r composed of non-negative elements and $\mathbf{V}$ can be written as $\mathbf{V} = \left[\mathbf{V}_1^\top, \ldots, \mathbf{V}_L^\top\right]^\top$ with $\mathbf{V}_l \in \mathbb{R}^{J_l\times r}$, $l = 1, \ldots, L$. Comparably, $\mathbf{A}_l = \mathbf{P}_l\mathbf{\Delta}_l\mathbf{Q}_l^\top$, where $\mathbf{P}_l$ and $\mathbf{Q}_l$ are two orthonormal matrices of size $I \times r_l$ and $J_l \times r_l$ respectively and $\mathbf{\Delta}_l$ is a diagonal matrix of size r_l composed of non-negative elements. Hence, the model (1.6) presented in the preamble is recovered with $\mathbf{T}_0 = \mathbf{U}\mathbf{\Delta}^{1/2}$, $\mathbf{W}_{0l} = \mathbf{V}_l\mathbf{\Delta}^{1/2}$, $\mathbf{T}_l = \mathbf{P}_l\mathbf{\Delta}_l^{1/2}$ and $\mathbf{W}_l = \mathbf{Q}_l\mathbf{\Delta}_l^{1/2}$.

The next section focuses on presenting the mathematical background and notations that are used in the tensor literature. The most popular multiway models are also briefly presented.

1.3 Multiway Notations and Operators

1.3.1 Notations

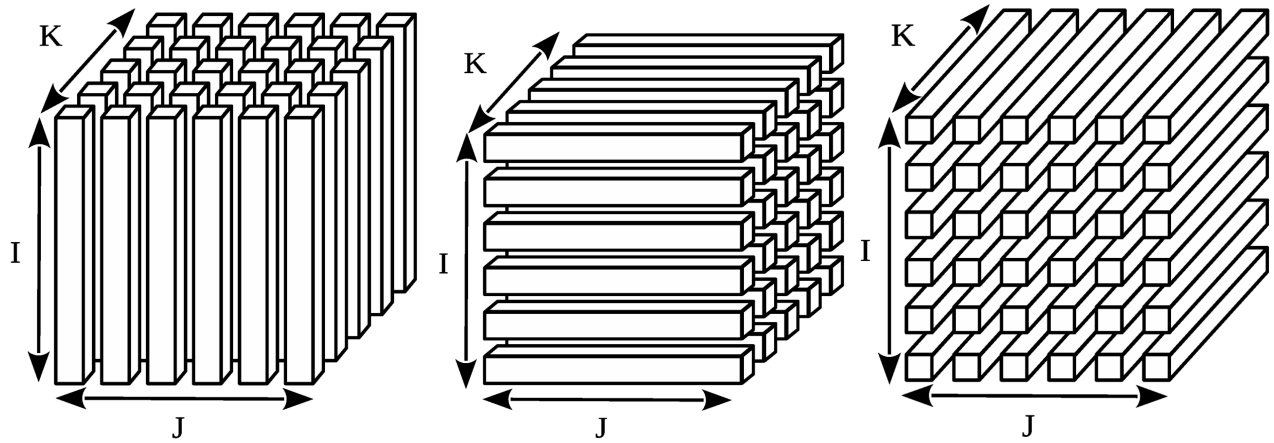
In the multiway literature, 1-way and 2-way arrays are used to refer to vectors and matrices respectively. When the number of ways/modes/orders is greater than 2, they are simply called N-way tensors/arrays. The main interest of this work is focused on 2 and 3-way arrays, even-though some fourth-order tensors might appear here and there. All along this manuscript, scalars are written as lowercase italic characters x, vectors as boldface lowercase characters $\mathbf{x}$, matrices as boldface uppercase characters $\mathbf{X}$ and three-way arrays as underlined boldface uppercase characters $\underline{\mathbf{X}}$, following the standardized notations and terminology proposed by [Kiers, 2000]. Tensor of order $N \geq 4$ are denoted by Euler script letters $\mathcal{X}$.

For sake of clarity, the vast majority of the concepts are presented in the case of a third-order tensor $\underline{\mathbf{X}}_l$, even though they can be easily extended to higher-order tensors. The running subscript l characterizing a block is discarded here in order to simplify the notations.

1.3.2 Fibers and slices

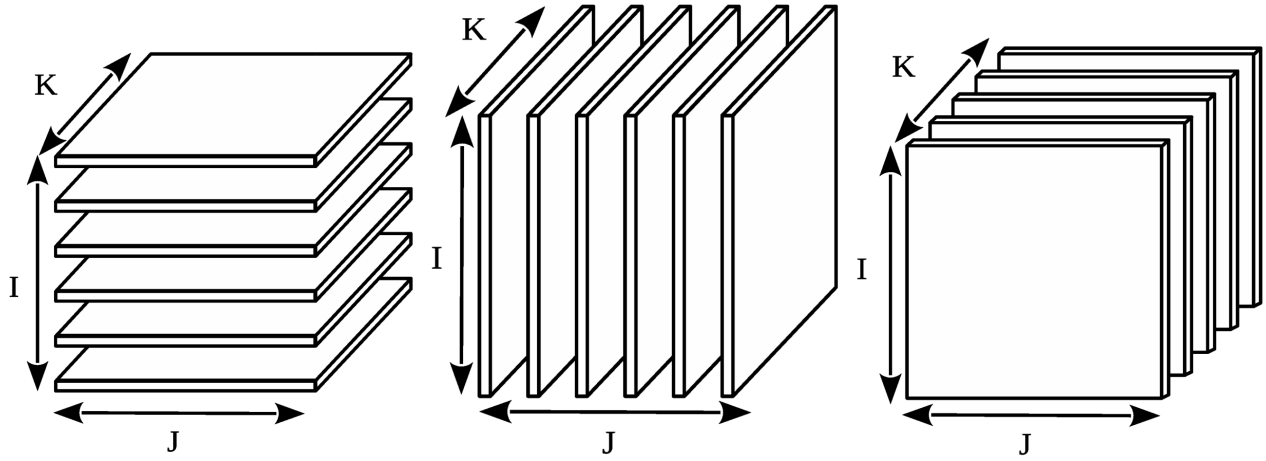
The mode fiber of a tensor is a vector defined by fixing all indices except for one. For example, for a matrix, columns and rows are respectively associated to mode-1 and mode-2 fibers of the matrix. For a third-order tensors $\underline{\mathbf{X}}$, column, row, and tube fibers are denoted by $\mathbf{x}_{.jk}$, $\mathbf{x}_{i.k}$, and $\mathbf{x}_{ij.}$, respectively; see Figure 1.3-1. When extracted from the tensor, fibers are always assumed to be oriented as column vectors.

Slices, in comparison to fibers, are two-dimensional sections of a tensor. They are defined by fixing all indices except for two. Figure 1.3-2 shows the horizontal, lateral, and frontal slices of a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$, denoted by $\mathbf{X}_{i..} \in \mathbb{R}^{J \times K}$, $\mathbf{X}_{.j.} \in \mathbb{R}^{I \times K}$, and $\mathbf{X}_{..k} \in \mathbb{R}^{I \times J}$, respectively.



(a) Mode-1 fibers (columns): $\mathbf{x}_{.jk}$ (b) Mode-2 fibers (rows): $\mathbf{x}_{i.k}$ (c) Mode-3 fibers (tubes): $\mathbf{x}_{ij.}$

Figure 1.3-1 – Third-order tensor mode fibers.



(a) Horizontal slices: $\mathbf{X}_{i..}$ (b) Lateral slices: $\mathbf{X}_{.j.}$ (c) Frontal slices: $\mathbf{X}_{..k}$

Figure 1.3-2 – Third-order tensor slices.

1.3.3 Matrix and Tensor Reshaping

1.3.3.1 Vectorization: Transforming an Array into a Vector

Commonly, the operator that reshapes a matrix $\mathbf{X} \in \mathbb{R}^{I \times J}$ into a column vector is noted as «vec». Among the various possibilities to vectorize a matrix, we adopt the column-wise vectorization. If $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_J]$, then $\text{vec}(\mathbf{X}) = \left[\mathbf{x}_1^\top, \ldots, \mathbf{x}_J^\top\right]^\top$ (also denoted $(\mathbf{x}_1; \ldots; \mathbf{x}_J)$).

An operator that maps tensors to vectors can also be defined. However, if this operation is needed, it will be done in two steps here: first a matricization (see next section) and then a vectorization.

1.3.3.2 Matricization: Transforming a Tensor into a Matrix

Matricization, also known as unfolding or flattening, consists in reshaping a tensor into a matrix. Here, only the case of mode-n matricization is presented, for a broader overview of matricization, see [Kolda, 2006]. The mode-n matricization of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ is denoted by $\mathbf{X}_{(n)}$ and arranges the mode-n fibers to be the columns of the resulting matrix. We only need to define in which order fibers are arranged. We again adopt the notation of [Kolda and Bader, 2009] that suggests to arrange them in the same order as their modes.

In order to better understand the matricization process, the case of a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{4 \times 3 \times 2}$ is taken. Its two frontal slices are:

$$\mathbf{X}_{..1} = \begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}, \ \mathbf{X}_{..2} = \begin{bmatrix} 13 & 17 & 21 \\ 14 & 18 & 22 \\ 15 & 19 & 23 \\ 16 & 20 & 24 \end{bmatrix} \tag{1.9}$$

Then, the mode-n matricizations for this particular tensor are:

$$\mathbf{X}_{(1)} = \begin{bmatrix} 1 & 5 & 9 & 13 & 17 & 21 \\ 2 & 6 & 10 & 14 & 18 & 22 \\ 3 & 7 & 11 & 15 & 19 & 23 \\ 4 & 8 & 12 & 16 & 20 & 24 \end{bmatrix}, \tag{1.10}$$

$$\mathbf{X}_{(2)} = \begin{bmatrix} 1 & 2 & 3 & 4 & 13 & 14 & 15 & 16 \\ 5 & 6 & 7 & 8 & 17 & 18 & 19 & 20 \\ 9 & 10 & 11 & 12 & 21 & 22 & 23 & 24 \end{bmatrix}, \tag{1.11}$$

$$\mathbf{X}_{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 \end{bmatrix}. \tag{1.12}$$

Therefore, the mode-1 matricization gives a matrix defined as the concatenation of the frontal slices next to each others $\mathbf{X}_{(1)} = [\mathbf{X}_{..1}, \ldots, \mathbf{X}_{..K}]$. The mode-2 matricization is the concatenation of the transposed frontal slices $\mathbf{X}_{(2)} = \left[\mathbf{X}_{..1}^\top, \ldots, \mathbf{X}_{..K}^\top\right]$. The mode-3 matricization is the concatenation of the transposed lateral slices $\mathbf{X}_{(3)} = \left[\mathbf{X}_{.1.}^\top, \ldots, \mathbf{X}_{.J.}^\top\right]$.

Another way of arranging the fibers when performing a mode-n matricization is presented in [Kiers, 2000]. In the case of a third-order tensor, it only differs from the arrangement presented above for the mode-2 matricization, which becomes for the example presented in (1.9):

$$\mathbf{X}_{(2)} = \begin{bmatrix} 1 & 13 & 2 & 14 & 3 & 15 & 4 & 16 \\ 5 & 17 & 6 & 18 & 7 & 19 & 8 & 20 \\ 9 & 21 & 10 & 22 & 11 & 23 & 12 & 24 \end{bmatrix}. \tag{1.13}$$

Therefore, this mode-2 matricization by [Kiers, 2000] is the concatenation of the horizontal slices $\mathbf{X}_{(2)} = [\mathbf{X}_{1..}, \ldots, \mathbf{X}_{I..}]$. This can also be performed by first cyclically permuting the modes of the tensor such that mode 1, 2, 3 respectively becomes modes 3, 1, 2 and then apply a mode-1 matricization. Even though this approach seems more natural, we have chosen the way of unfolding introduced by [Kolda and Bader, 2009] (and presented above) because it is associated with interesting formulas for unfolding the Tucker and the CANDECOMP/PARAFAC models (see section 1.3.4.2 equation (1.19) and 1.4.1 equation (1.27)).

1.3.4 Tensor-Matrix Operators

1.3.4.1 Between Vectors/Matrices operators

The Kronecker product $\otimes$ and the Khatri-Rao product $\odot$ are presented in this section.

The Kronecker product. For a general definition of the Kronecker product, let us consider two matrices $\mathbf{X} \in \mathbb{R}^{I \times J}$ and $\mathbf{Y} \in \mathbb{R}^{K \times L}$ such that $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_J]$, $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_L]$ and $(\mathbf{X})_{ij} = x_{ij}$, $\forall i \in [\![1; I]\!]$, $j \in [\![1; J]\!]$. Then, the Kronecker product between $\mathbf{X}$ of dimension $I \times J$ and $\mathbf{Y}$ of dimension $K \times L$ is a matrix of dimension $(IK) \times (JL)$ such that:

$$\mathbf{X} \otimes \mathbf{Y} = \begin{bmatrix} x_{11}\mathbf{Y} & \ldots & x_{1J}\mathbf{Y} \\ \vdots & & \vdots \\ x_{I1}\mathbf{Y} & \ldots & x_{IJ}\mathbf{Y} \end{bmatrix} = [\mathbf{x}_1 \otimes \mathbf{y}_1, \ldots, \mathbf{x}_1 \otimes \mathbf{y}_L, \mathbf{x}_2 \otimes \mathbf{y}_1, \ldots, \mathbf{x}_2 \otimes \mathbf{y}_L, \ldots, \mathbf{x}_J \otimes \mathbf{y}_L]. \tag{1.14}$$

The Kronecker product can be used between an arbitrary number of matrices/vectors. Considering $\mathbf{X} \in \mathbb{R}^{I\times J}$, $\mathbf{Y} \in \mathbb{R}^{K\times L}$, $\mathbf{W} \in \mathbb{R}^{J\times M}$, $\mathbf{Z} \in \mathbb{R}^{L\times N}$ and two non singular matrices $\mathbf{U} \in \mathbb{R}^{P\times P}$, $\mathbf{V} \in \mathbb{R}^{Q\times Q}$, then we have the following properties for the Kronecker Product (see [Kolda and Bader, 2009, Loan, 2000, Smilde et al., 2004]):

$$\begin{aligned}(\mathbf{X}\otimes\mathbf{Y})(\mathbf{W}\otimes\mathbf{Z}) &= (\mathbf{X}\mathbf{W}\otimes\mathbf{Y}\mathbf{Z})\\ (\mathbf{X}\otimes\mathbf{Y})^\top &= \left(\mathbf{X}^\top\otimes\mathbf{Y}^\top\right)\\ (\mathbf{U}\otimes\mathbf{V})^{-1} &= \mathbf{U}^{-1}\otimes\mathbf{V}^{-1}\end{aligned} \tag{1.15}$$

The Khatri-Rao product between two matrices is the column-wise Kronecker product and therefore is defined between matrices that have the same number of columns. More formally, the Khatri-Rao product between a matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_R]$ of dimension $I \times R$ and a matrix $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_R]$ of dimension $J \times R$ is a matrix of dimension $(IJ) \times R$ such that:

$$\mathbf{X}\odot\mathbf{Y} = [\mathbf{x}_1\otimes\mathbf{y}_1, \ldots, \mathbf{x}_R\otimes\mathbf{y}_R]\,. \tag{1.16}$$

The Khatri-Rao product can be used between an arbitrary number of matrices that have the same number of columns. Considering $\mathbf{X} \in \mathbb{R}^{I\times R}$ and $\mathbf{Y} \in \mathbb{R}^{J\times R}$, we have the following property for the Khatri-Rao Product (see [Kolda and Bader, 2009, Loan, 2000, Smilde et al., 2004]):

$$(\mathbf{X}\odot\mathbf{Y})^\top(\mathbf{X}\odot\mathbf{Y}) = \left(\mathbf{X}^\top\mathbf{X}\star\mathbf{Y}^\top\mathbf{Y}\right), \tag{1.17}$$

where $\star$ is the element-wise product between matrices, also called the Hadamard product.

1.3.4.2 Between Tensor and Matrix operator

The mode product between a tensor and a matrix is the generalization of the matrix product. Let us take the example of a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$ and a matrix $\mathbf{W} \in \mathbb{R}^{R\times J}$. The mode-2 product between $\underline{\mathbf{X}}$ of dimension $I \times J \times K$ and $\mathbf{W}$ of dimension $R \times J$ is written $\underline{\mathbf{X}} \times_2 \mathbf{W}$ and results in a tensor of dimension $I \times R \times K$ such that each of its elements are defined as (see [Lathauwer et al., 2000]):

$$(\underline{\mathbf{X}}\times_2\mathbf{W})_{irk} = \sum_{j=1}^{J} x_{ijk}w_{rj}, \quad \begin{cases}\forall i \in [\![1; I]\!]\\ \forall r \in [\![1; R]\!]\\ \forall k \in [\![1; K]\!]\end{cases} \tag{1.18}$$

The k^{th} frontal slices of the resulting tensor can then be expressed as $\mathbf{X}_{..k}\mathbf{W}^\top$.

Similarly, mode-1 and mode-3 products can be defined.

Considering $\underline{\mathbf{X}} \in \mathbb{R}^{I\times J\times K}$, $\mathbf{W}^I \in \mathbb{R}^{R_I\times I}$, $\mathbf{W}^J \in \mathbb{R}^{R_J\times J}$ and $\mathbf{W}^K \in \mathbb{R}^{R_K\times K}$, then we have the following property:

$$\begin{aligned}\underline{\mathbf{Y}} = \underline{\mathbf{X}}\times_1\mathbf{W}^I\times_2\mathbf{W}^J\times_3\mathbf{W}^K &\Leftrightarrow \mathbf{Y}_{(1)} = \mathbf{W}^I\mathbf{X}_{(1)}\left(\mathbf{W}^K\otimes\mathbf{W}^J\right)^\top\\ &\Leftrightarrow \mathbf{Y}_{(2)} = \mathbf{W}^J\mathbf{X}_{(2)}\left(\mathbf{W}^K\otimes\mathbf{W}^I\right)^\top\\ &\Leftrightarrow \mathbf{Y}_{(3)} = \mathbf{W}^K\mathbf{X}_{(3)}\left(\mathbf{W}^J\otimes\mathbf{W}^I\right)^\top,\end{aligned} \tag{1.19}$$

see [Kolda, 2006] for a proof of this property.

1.3.4.3 Operators Between Tensors

Two inner-products between tensors are presented in this section.

The inner-product between two tensors $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ of the same size $I \times J \times K$ denoted by $< \underline{\mathbf{X}}, \underline{\mathbf{Y}} >$ is defined as the sum of the product of all their elements:

$$\langle \underline{\mathbf{X}}, \underline{\mathbf{Y}} \rangle = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} x_{ijk} y_{ijk}. \tag{1.20}$$

The norm of a tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ is given by $\|\underline{\mathbf{X}}\|^2 =< \underline{\mathbf{X}}, \underline{\mathbf{X}} >= \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} x_{ijk}^2$. This is the so-called Frobenius norm of a tensor and it is referred as $\|.\|_F$.

Moreover, a generalization of the inner-product between two tensors which only requires to have at least one mode of the same dimension (not necessarily the same mode) is presented (see [Bader and Kolda, 2006]). The inner-product between $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J_1 \times K_1}$ and $\underline{\mathbf{Y}} \in \mathbb{R}^{I_2 \times I \times K_2}$ along the first and second mode respectively is denoted by $\underline{\mathbf{X}} \times_2^1 \underline{\mathbf{Y}}$ (notation taken from [Zniyed et al., 2019]) and results in a tensor of size $J_1 \times K_1 \times I_2 \times K_2$, given by:

$$\left(\underline{\mathbf{X}} \times_2^1 \underline{\mathbf{Y}}\right)_{j_1 k_1 i_2 k_2} = \sum_{i=1}^{I} x_{i j_1 k_1} y_{i_2 i k_2} = \mathbf{x}_{.j_1 k_1}{}^{\top} \mathbf{y}_{i_2 . k_2} \tag{1.21}$$

This tensor concentrates all the possible inner-products between the mode-1 fibers of $\underline{\mathbf{X}}$ and the mode-2 fibers of $\underline{\mathbf{Y}}$ ($J_1 K_1 I_2 K_2$ in total) arranged in a «specific order» (see [Bader and Kolda, 2006] for details).

1.4 Classical Multiway Models

As specified in the introducing section 1.3.1, for easier readability, the models are presented in the case of a third-order tensor $\underline{\mathbf{X}}$, even-though they can be easily extended to higher-order tensors.

1.4.1 The CANDECOMP/PARAFAC (CP) decomposition

A third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ is a rank one tensor if there exists $\mathbf{w}^I \in \mathbb{R}^I$, $\mathbf{w}^J \in \mathbb{R}^J$ and $\mathbf{w}^K \in \mathbb{R}^K$ such that:

$$\underline{\mathbf{X}} = \mathbf{w}^I \circ \mathbf{w}^J \circ \mathbf{w}^K, \tag{1.22}$$

where $\circ$ stands for the vector outer product. This means that each element $x_{ijk} = (\underline{\mathbf{X}})_{ijk}$ of the tensor is the product of the corresponding vector elements:

$$x_{ijk} = w_i^I w_j^J w_k^K, \quad \begin{cases} \forall i \in [\![1; I]\!] \\ \forall j \in [\![1; J]\!] \\ \forall k \in [\![1; K]\!] \end{cases} \tag{1.23}$$

An explanation on how the outer product works can be seen on Figure 1.4-3.

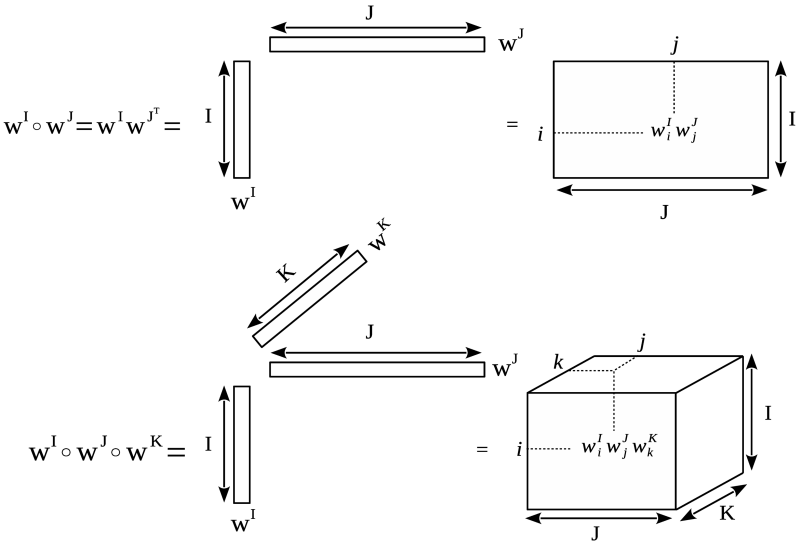


Figure 1.4-3 – Outer product.

Hence, a rank-R tensor can be written as a sum of rank-one tensors:

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{w}^{I,(r)} \circ \mathbf{w}^{J,(r)} \circ \mathbf{w}^{K,(r)} \Leftrightarrow x_{ijk} = \sum_{r=1}^{R} w_i^{I,(r)} w_j^{J,(r)} w_k^{K,(r)}, \quad \begin{cases} \forall i \in [\![1; I]\!] \\ \forall j \in [\![1; J]\!] \\ \forall k \in [\![1; K]\!] \end{cases} \tag{1.24}$$

where $\forall r \in [\![1; R]\!] : \mathbf{w}^{I,(r)} \in \mathbb{R}^I,\ \mathbf{w}^{J,(r)} \in \mathbb{R}^J$ and $\mathbf{w}^{K,(r)} \in \mathbb{R}^K$.

If we introduce matrices $\mathbf{W}^I = \left[\mathbf{w}^{I,(1)}, \ldots, \mathbf{w}^{I,(R)}\right]$, $\mathbf{W}^J = \left[\mathbf{w}^{J,(1)}, \ldots, \mathbf{w}^{J,(R)}\right]$ and $\mathbf{W}^K = \left[\mathbf{w}^{K,(1)}, \ldots, \mathbf{w}^{K,(R)}\right]$, a concise notation for (1.24) is $\underline{\mathbf{X}} = [\![\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!]$. It is often useful to assume that the columns of $\mathbf{W}^I$, $\mathbf{W}^J$ and $\mathbf{W}^K$ are normalized with the scaling absorbed into the vector $\boldsymbol{\lambda} \in \mathbb{R}^R$ such that:

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \lambda^{(r)} \mathbf{w}^{I,(r)} \circ \mathbf{w}^{J,(r)} \circ \mathbf{w}^{K,(r)} = [\![\boldsymbol{\lambda}; \mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!]. \tag{1.25}$$

As explained in [Kolda and Bader, 2009], the first proposition to decompose a tensor into a sum of rank-one tensors was made in 1927 by [Hitchcock, 1927, 1928]. The concept finally became popular after its third introduction, in 1970 to the psychometric community, in the form of CANDECOMP (canonical decomposition) by [Carroll and Chang, 1970] and PARAFAC (parallel factors) by [Harshman, 1970]. We refer to the CANDECOMP/PARAFAC decomposition as CP.

The CP decomposition factorizes a tensor into a sum of rank-one tensors. For example, when the fit of the CP decomposition is evaluated thanks to the Least Squares (LS), this leads to the following optimization problem:

$$\underset{\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K}{\operatorname{argmin}} \left\| \underline{\mathbf{X}} - [\![\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!] \right\|_F^2 \tag{1.26}$$

The CP decomposition is subsumed in Figure (1.4-4). This CP decomposition also presents useful

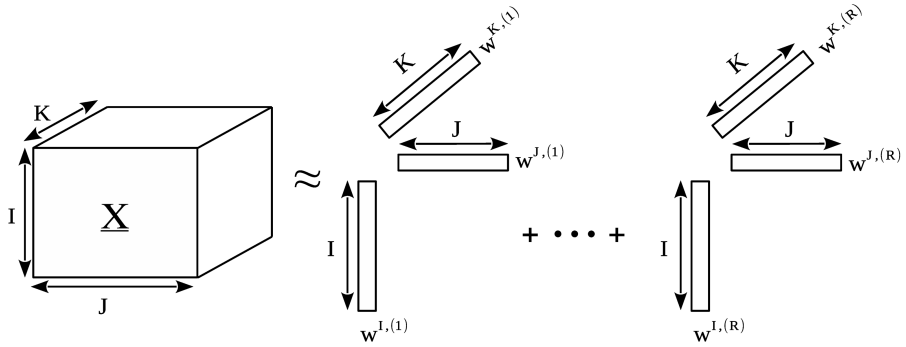


Figure 1.4-4 – CP decomposition of a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$.

unfolding properties. Indeed:

$$\begin{aligned} \underline{\mathbf{X}} = [\![\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!] &\Leftrightarrow \mathbf{X}_{(1)} = \mathbf{W}^I \left(\mathbf{W}^K \odot \mathbf{W}^J\right)^\top \\ &\Leftrightarrow \mathbf{X}_{(2)} = \mathbf{W}^J \left(\mathbf{W}^K \odot \mathbf{W}^I\right)^\top \\ &\Leftrightarrow \mathbf{X}_{(3)} = \mathbf{W}^K \left(\mathbf{W}^J \odot \mathbf{W}^I\right)^\top, \end{aligned} \tag{1.27}$$

As mentioned in [ten Berge, 1993], another formulation of the CP decomposition is possible. Indeed, let us consider a rank-R tensor, as described in equation (1.24). We can introduce $\mathbf{\Delta}_k$, $k = 1, \ldots, K$, which are diagonal matrices of size R such that the diagonal of $\mathbf{\Delta}_k$ is composed of the k^{th} row of $\mathbf{W}^K$. Thus, each frontal slice of $\underline{\mathbf{X}}$ can be re-written as $\mathbf{X}_{..k} = \mathbf{W}^I \mathbf{\Delta}_k \mathbf{W}^{J^\top}$. With this formulation, the CP decomposition optimization problem (1.26) can be re-written as:

$$\min_{\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K} \sum_{k=1}^{K} \left\| \mathbf{X}_{..k} - \mathbf{W}^I \mathbf{\Delta}_k \mathbf{W}^{J^\top} \right\|_F^2. \tag{1.28}$$

This formulation helps to see the CP decomposition as a version of a SCA model (cf. section 1.2.3.1) where blocks are replaced by frontal slices. This optimization problem is not strictly equivalent to the SCA one as the block weight matrices bear a specific structure.

A strong advantage of the CP model is the uniqueness of the decomposition under mild conditions (see [Kruskal, 1977]). These conditions are designed to avoid two indeterminacies in the CP model. Indeed, first, there is a permutation indeterminacy: if you apply the same column permutation to each factor matrix $\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K$, the CP model remains unchanged. Moreover, there is a factor indeterminacy: take $(a, b, c) \in \mathbb{R}^3$ such that $abc = 1$, then if you multiply the r^{th} column of $\mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K$ respectively by the scalar a, b, c, the CP decomposition is left unchanged. These two indeterminacies can be overcome quite easily.

A first drawback of the CP decomposition, common to almost all tensor factorization models, is that an equivalent of the Eckart–Young theorem [Eckart and Young, 1936] for matrix factorization is not possible. Indeed, is we perform a rank-R and a rank-Q (with $Q > R$) CP decomposition, then the first R columns of the matrix factors of the rank-Q decomposition are not necessarily equals to the R columns of the matrix factors of the rank-R decomposition. A second drawback is that this model is very constraining. Indeed, for example, the r^{th} column of a matrix factor is only interacting with the

the r^{th} columns of the other matrix factors, hence, each matrix factor has to have the same number of columns. Among other things, the Tucker model was proposed to overcome this last drawback.

1.4.2 The Tucker Model

This model was first presented in [Tucker, 1963, 1964]. In the case of a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$, the Tucker Model aims at factorizing this tensor as three mode products between a core tensor $\underline{\mathbf{G}} \in \mathbb{R}^{R_I \times R_J \times R_K}$ and three factor matrices $\mathbf{W}^I \in \mathbb{R}^{I \times R_I}$, $\mathbf{W}^J \in \mathbb{R}^{J \times R_J}$ and $\mathbf{W}^K \in \mathbb{R}^{K \times R_K}$ for each mode of the core tensor, resulting in the following approximate equation:

$$\underline{\mathbf{X}} \approx \underline{\mathbf{G}} \times_1 \mathbf{W}^I \times_2 \mathbf{W}^J \times_3 \mathbf{W}^K \Leftrightarrow x_{ijk} \approx \sum_{p=1}^{R_I} \sum_{q=1}^{R_J} \sum_{r=1}^{R_K} g_{pqr} w_{ip}^I w_{jq}^J w_{kr}^K, \quad \begin{cases} \forall i \in [\![1; I]\!] \\ \forall j \in [\![1; J]\!] \\ \forall k \in [\![1; K]\!] \end{cases} \tag{1.29}$$

Similarly to the CP decomposition, if we introduce matrices $\mathbf{W}^I = \left[\mathbf{w}^{I,(1)}, \ldots, \mathbf{w}^{I,(R_I)}\right]$, $\mathbf{W}^J = \left[\mathbf{w}^{J,(1)}, \ldots, \mathbf{w}^{J,(R_J)}\right]$ and $\mathbf{W}^K = \left[\mathbf{w}^{K,(1)}, \ldots, \mathbf{w}^{K,(R_K)}\right]$, then a concise notation for (1.29), introduced in [Kolda, 2006], is $\underline{\mathbf{X}} = [\![\underline{\mathbf{G}}; \mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!]$. The Tucker model is subsumed in Figure (1.4-5).

Moreover, another notation equivalent to (1.29) is possible:

$$\underline{\mathbf{X}} \approx \sum_{p=1}^{R_I} \sum_{q=1}^{R_J} \sum_{r=1}^{R_K} g_{pqr} \mathbf{w}^{I,(p)} \circ \mathbf{w}^{J,(q)} \circ \mathbf{w}^{K,(r)}. \tag{1.30}$$

This last formulation helps understanding the link between the CP decomposition and the Tucker decomposition. Indeed, if $\underline{\mathbf{G}}$ is superdiagonal, meaning $g_{pqr} = 0$, if $p \neq q \neq r$, then (1.30) becomes $\underline{\mathbf{X}} \approx \sum_{r=1}^{min(R_I, R_J, R_K)} g_{rrr} \mathbf{w}^{I,(r)} \circ \mathbf{w}^{J,(r)} \circ \mathbf{w}^{K,(r)}$, which is a CP decomposition. We also realize that the Tucker decomposition does allow every interaction between columns of the factor matrices.

The fit of the Tucker decomposition is evaluated in a Least Squares (LS) sense which leads to the following optimization problem:

$$\underset{\underline{\mathbf{G}}, \mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K}{\operatorname{argmin}} \left\| \underline{\mathbf{X}} - [\![\underline{\mathbf{G}}; \mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!] \right\|_F^2 \tag{1.31}$$

The major drawback of the Tucker model is its indeterminacy. Indeed, for any non-singular matrices $\mathbf{T} \in \mathbb{R}^{R_I \times R_I}$, $\mathbf{U} \in \mathbb{R}^{R_J \times R_J}$ and $\mathbf{V} \in \mathbb{R}^{R_K \times R_K}$, $[\![\underline{\mathbf{G}} \times_1 \mathbf{T} \times_2 \mathbf{U} \times_3 \mathbf{V}; \mathbf{W}^I \mathbf{T}^{-1}, \mathbf{W}^J \mathbf{U}^{-1}, \mathbf{W}^K \mathbf{V}^{-1}]\!] =$

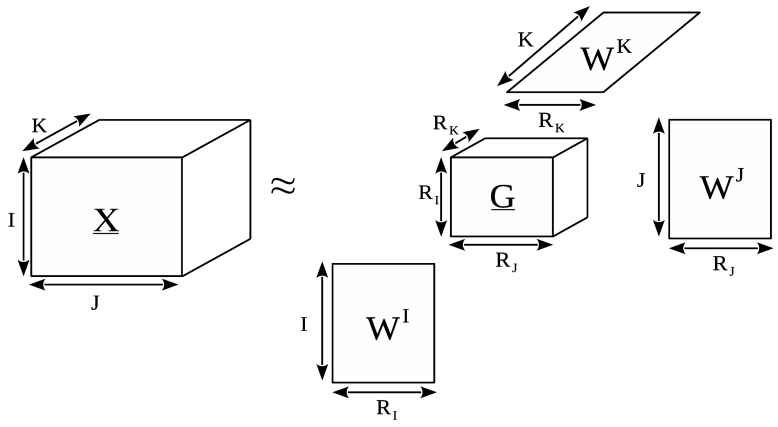


Figure 1.4-5 – Tucker decomposition of a third-order tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$.

$[\![\underline{\mathbf{G}}; \mathbf{W}^I, \mathbf{W}^J, \mathbf{W}^K]\!]$. Moreover, in comparison to CP decomposition, three parameters have to be set: R_I, R_J and R_K instead of one parameter R. Finally, the core tensor $\underline{\mathbf{G}}$ is often difficult to interpret.

1.4.3 The Coupled Matrix Tensor Factorization (CMTF)

The Coupled Matrix Tensor Factorization (CMTF) [Acar et al., 2011] proposes to jointly factorize a collection of tensors and matrices by coupling them through one or several modes. This model factorizes each tensor with a CP model and each matrix with a matrix factorization model. Moreover, the coupled modes share common factor matrices. This factorization model is usually presented in the case of a third-order tensor $\underline{\mathbf{X}}_1 \in \mathbb{R}^{I \times J_1 \times K_1}$ and a matrix $\mathbf{X}_2 \in \mathbb{R}^{I \times J_2}$ coupled in the first mode, which leads to the following optimization problem:

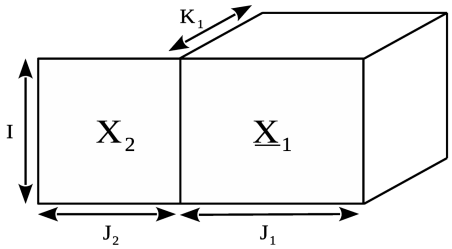
$$\underset{\mathbf{Y}, \mathbf{W}_1^J, \mathbf{W}_1^K, \mathbf{W}_2^J}{\operatorname{argmin}} \left\| \underline{\mathbf{X}}_1 - [\![\mathbf{Y}, \mathbf{W}_1^J, \mathbf{W}_1^K]\!] \right\|_F^2 + \left\| \mathbf{X}_2 - \mathbf{Y}\mathbf{W}_2^{J\top} \right\|_F^2, \tag{1.32}$$

where $\mathbf{W}_1^J \in \mathbb{R}^{J_1 \times R}, \mathbf{W}_1^K \in \mathbb{R}^{K_1 \times R}, \mathbf{W}_2^J \in \mathbb{R}^{J_2 \times R}$ and $\mathbf{Y} \in \mathbb{R}^{I \times R}$. This joint factorization is represented in Figure 1.4-6a. In a second example, two tensors $\underline{\mathbf{X}}_1 \in \mathbb{R}^{I \times J_1 \times K}$ and $\underline{\mathbf{X}}_2 \in \mathbb{R}^{I \times J \times K}$ are coupled in both their first and third modes and a matrix $\mathbf{X}_3 \in \mathbb{R}^{I_3 \times J}$ is coupled with the second mode of $\underline{\mathbf{X}}_2$. The optimization criterion, associated with this example can be expressed as:

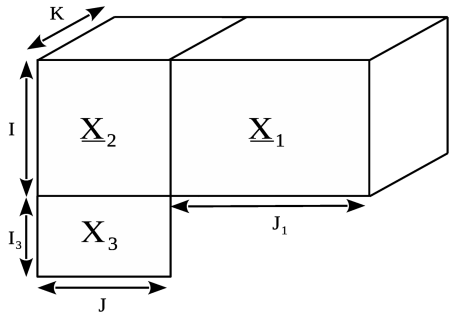
$$\underset{\mathbf{Y}, \mathbf{Z}, \mathbf{V}, \mathbf{W}_1^J, \mathbf{W}_3^I}{\operatorname{argmin}} \left\| \underline{\mathbf{X}}_1 - [\![\mathbf{Y}, \mathbf{W}_1^J, \mathbf{Z}]\!] \right\|_F^2 + \| \underline{\mathbf{X}}_2 - [\![\mathbf{Y}, \mathbf{V}, \mathbf{Z}]\!] \|_F^2 + \left\| \mathbf{X}_3 - \mathbf{W}_3^I \mathbf{V}^\top \right\|_F^2, \tag{1.33}$$

where $\mathbf{W}_1^J \in \mathbb{R}^{J_1 \times R}$, $\mathbf{W}_3^I \in \mathbb{R}^{I_3 \times R}$, $\mathbf{Y} \in \mathbb{R}^{I \times R}$, $\mathbf{Z} \in \mathbb{R}^{K \times R}$ and $\mathbf{V} \in \mathbb{R}^{J \times R}$. This joint factorization is represented in Figure 1.4-6b.

This model is both a multiway and a multiblock model. More recently, an Advanced CMTF (ACMTF) model was proposed in [Acar et al., 2014] to allow, with an ℓ_1-penalty, for each component of a common factor matrix to be either a common or a specific component to one of the block it is coupled with.



(a) Coupling of a third-order tensor $\underline{\mathbf{X}}_1 \in \mathbb{R}^{I \times J_1 \times K_1}$ and a matrix $\mathbf{X}_2 \in \mathbb{R}^{I \times J_2}$ in the first mode.



(b) Coupling of two tensors $\underline{\mathbf{X}}_1 \in \mathbb{R}^{I \times J_1 \times K}$, $\underline{\mathbf{X}}_2 \in \mathbb{R}^{I \times J \times K}$ in both their first and third modes and a matrix $\mathbf{X}_3 \in \mathbb{R}^{I_3 \times J}$ with the second mode of $\underline{\mathbf{X}}_2$.

Figure 1.4-6 – Two examples of the Coupled Matrix Tensor Factorization Model (CMTF).

1.5 Optimization Framework

The goal of this section is to present the optimization framework under which most of the algorithms derived in this document were designed. This framework has already been presented in [Tenenhaus et al., 2017] under the example of a spherical constraint. It is recalled here for a broader class of constraints.

1.5.1 Optimization Problem

This framework is proposed for the maximization of a continuously differentiable multi-convex function $f(\mathbf{v}_1, \ldots, \mathbf{v}_L) : \mathbb{R}^{J_1} \times \ldots \times \mathbb{R}^{J_L} \to \mathbb{R}$ (i.e. for each l, f is a convex function of $\mathbf{v}_l$ while all the other $\mathbf{v}_k$ are fixed) under the constraint that each $\mathbf{v}_l$ belongs to a compact set $\Omega_l \subset \mathbb{R}^{J_l}$. This general optimization problem can be formulated as follows:

$$\max_{\mathbf{v}_1,\ldots,\mathbf{v}_L} f(\mathbf{v}_1, \ldots, \mathbf{v}_L) \tag{1.34}$$

$$\text{s.t. } \mathbf{v}_l \in \Omega_l,\ l = 1, \ldots, L. \tag{1.35}$$

Remark on notations. For such function defined over a set of parameter vectors $(\mathbf{v}_1, \ldots, \mathbf{v}_L)$, we make no difference between the notations $f(\mathbf{v}_1, \ldots, \mathbf{v}_L)$ and $f(\mathbf{v})$, where $\mathbf{v}$ is the column vector $\mathbf{v} = \left(\mathbf{v}_1^\top, \ldots, \mathbf{v}_L^\top\right)^\top$ of size $J = \sum_{l=1}^{L} J_l$. Moreover, for the vertical concatenation of column vectors, the notation $\mathbf{v} = (\mathbf{v}_1; \ldots; \mathbf{v}_L)$ is preferred for the sake of simplification. This last formulation is also used to define a vertical concatenation of matrices. These notations are used all along this manuscript.

1.5.2 Algorithm

A simple, monotonically and globally convergent algorithm is presented for maximizing (1.34) subject to (1.35). An algorithm is globally convergent if, regardless of its initialization, it converges towards a stationary point. For an unconstrained optimization problem with a continuously differentiable objective function, a stationary point is a point where the derivative of the objective function is null. For a constrained optimization problem, a stationary point is a point where the derivative of the Lagrangian function associated with the problem is null. For such a point, the derivative of the objective function lies in the subspace defined by the derivative of each constraint. This condition is called the Karush-Kuhn-Tucker (KKT) condition.

The maximization of the function f defined over different parameter vectors $(\mathbf{v}_1, \ldots, \mathbf{v}_L)$, is approached by updating each of the parameter vectors in turn, keeping the others fixed. This update rule was recommended in [De Leeuw, 1994] and is called cyclic Block Coordinate Ascent (BCA).

In order to do so, let $\nabla_l f(\mathbf{v})$ be the partial gradient of $f(\mathbf{v})$ with respect to $\mathbf{v}_l$. We assume $\nabla_l f(\mathbf{v}) \neq \mathbf{0}$ in this manuscript. This assumption is not too binding as $\nabla_l f(\mathbf{v}) = \mathbf{0}$ characterizes the global minimum of $f(\mathbf{v}_1, \ldots, \mathbf{v}_L)$ with respect to $\mathbf{v}_l$ when the other vectors $\mathbf{v}1, \ldots, \mathbf{v}_{l-1}, \mathbf{v}_{l+1}, \ldots, \mathbf{v}_L$ are fixed. We want to find an update $\hat{\mathbf{v}}_l \in \Omega_l$ such that $f(\mathbf{v}) \leq f(\mathbf{v}_1, ..., \mathbf{v}_{l-1}, \hat{\mathbf{v}}_l, \mathbf{v}_{l+1}, ..., \mathbf{v}_L)$. As f is a continuously differentiable multi-convex function and considering that a convex function lies above its linear approximation at $\mathbf{v}_l$ for any $\tilde{\mathbf{v}}_l \in \Omega_l$, the following inequality holds:

$$f(\mathbf{v}_1, ..., \mathbf{v}_{l-1}, \tilde{\mathbf{v}}_l, \mathbf{v}_{l+1}, \ldots, \mathbf{v}_L) \geq f(\mathbf{v}) + \nabla_l f(\mathbf{v})^\top (\tilde{\mathbf{v}}_l - \mathbf{v}_l) := \ell_l(\tilde{\mathbf{v}}_l, \mathbf{v}) \tag{1.36}$$

On the right-hand side of the inequality (1.36), only the term $\nabla_l f(\mathbf{v})^\top \tilde{\mathbf{v}}_l$ is relevant to $\tilde{\mathbf{v}}_l$ and the solution that maximizes the minorizing function $\ell_l(\tilde{\mathbf{v}}_l, \mathbf{v})$ over $\tilde{\mathbf{v}}_l \in \Omega_l$ is obtained by considering the following optimization problem:

$$\hat{\mathbf{v}}_l = \underset{\tilde{\mathbf{v}}_l \in \Omega_l}{\operatorname{argmax}} \nabla_l f(\mathbf{v})^\top \tilde{\mathbf{v}}_l := r_l(\mathbf{v}). \tag{1.37}$$

The entire algorithm is subsumed in Algorithm 1.

Algorithm 1 Algorithm for the maximization of a continuously differentiable multi-convex function

1: **Result:** $\mathbf{v}_1^s, \ldots, \mathbf{v}_L^s$ (approximate solution of (1.34) subject to (1.35))
2: **Initialization:** choose random vector $\mathbf{v}_l^0 \in \Omega_l, l = 1, \ldots, L$, ε;
3: $s = 0$;
4: **repeat**
5: **for** $l = 1$ **to** L **do**
6: $$\mathbf{v}_l^{s+1} = r_l\left(\mathbf{v}_1^{s+1}, \ldots, \mathbf{v}_{l-1}^{s+1}, \mathbf{v}_l^s, \ldots, \mathbf{v}_L^s\right). \tag{1.38}$$
7: **end for**
8: $s = s + 1$;
9: **until** $f(\mathbf{v}_1^{s+1}, \ldots, \mathbf{v}_L^{s+1}) - f(\mathbf{v}_1^s, \ldots, \mathbf{v}_L^s) < \varepsilon$

1.5.3 Convergence Properties

To study the convergence properties of Algorithm 1, we introduce some notations: $\Omega = \Omega_1 \times \ldots \times \Omega_L$, $\mathbf{v} = (\mathbf{v}_1; ...; \mathbf{v}_L) \in \Omega$, $c_l : \Omega \mapsto \Omega$ is an operator defined as $c_l(\mathbf{v}) = (\mathbf{v}_1; ...; \mathbf{v}_{l-1}; r_l(\mathbf{v}); \mathbf{v}_{l+1}; ...; \mathbf{v}_L)$ with $r_l(\mathbf{v})$ introduced in equation (1.37) and $c : \Omega \mapsto \Omega$ is defined as $c = c_L \circ c_{L-1} \circ ... \circ c_1$, where $\circ$ stands for the function composition operator. We consider the sequence $\{\mathbf{v}^s = (\mathbf{v}_1^s; ...; \mathbf{v}_L^s)\}$ generated by Algorithm 1. Using the operator c, the «for loop» inside Algorithm 1 can be replaced by the following recurrence relation: $\mathbf{v}^{s+1} = c(\mathbf{v}^s)$. The convergence properties of Algorithm 1 are summarized in the following proposition:

Proposition 1.5.1. *Let $\{\mathbf{v}^s\}_{s=0}^\infty$ be any sequence generated by the recurrence relation $\mathbf{v}^{s+1} = c(\mathbf{v}^s)$ with $\mathbf{v}^0 \in \Omega$. Then, the following properties hold:*

(a) The sequence $\{f(\mathbf{v}^s)\}$ is monotonically increasing and therefore convergent as f is bounded on Ω. This result implies the monotonic convergence of Algorithm 1.

(b) If the infinite sequence $\{f(\mathbf{v}^s)\}$ involves a finite number of distinct terms, then the last distinct point satisfies $c(\mathbf{v}^s) = \mathbf{v}^s$ and therefore is a stationary point of problem (1.34).

(c) The limit of any convergent subsequence of $\{\mathbf{v}^s\}$ is a fixed point of c.

(d) $\lim_{s \to \infty} f(\mathbf{v}^s) = f(\mathbf{v}^\star)$, where $\mathbf{v}^\star$ is a fixed point of c.

(e) The sequence $\{\mathbf{v}^s = (\mathbf{v}_1^s; ...; \mathbf{v}_L^s)\}$, $l = 1, ..., L$, is asymptotically regular: $\lim_{s \to \infty} \sum_{l=1}^L \|\mathbf{v}_l^{s+1} - \mathbf{v}_l^s\| = 0$. This result implies that if the threshold ε for the stopping criterion in Algorithm 1 is made sufficiently small, the output of Algorithm 1 will be as close as wanted to a stationary point of (1.34).

(f) If the equation $\mathbf{v} = c(\mathbf{v})$ has a finite number of solutions, then the sequence $\{\mathbf{v}^s\}$ converges to one of them.

The goal is to demonstrate Proposition 1.5.1 that gathers all the convergence properties of Algorithm 1. For this purpose, the results given in the following lemma are useful.

Lemma 1.5.2. *Consider the set Ω, the function $f : \Omega \mapsto \mathbb{R}$ and the operator $c : \Omega \mapsto \Omega$ defined above. Then, the following properties hold:*

(i) Ω *is a compact set;*

(ii) c *is a continuous operator;*

(iii) $f(\mathbf{v}) \leq f(c(\mathbf{v}))$ *for any* $\mathbf{v} \in \Omega$*;*

(iv) If $f(\mathbf{v}) = f(c(\mathbf{v}))$*, then* $c(\mathbf{v}) = \mathbf{v}$*.*

Proof of Lemma 1.5.2.

Point *(i)* In this section, $\forall l,\ \Omega_l$ are assumed to be compact. As the Cartesian product of L compact sets is compact, $\Omega = \Omega_1 \times \ldots \times \Omega_L$ is compact.

Point *(ii)* We assume that $r_l(\mathbf{v})$ defined in equation (1.37) exists and is unique. As Ω_l is a compact set and l_l defined in equation (1.36) is a real-valued continuous function, Berge's maximum theorem applies and guarantees that the maximizer $r_l(\mathbf{v})$ of $l_l(\tilde{\mathbf{v}}_l, \mathbf{v})$ is continuous on Ω_l [Berge, 1966]. This implies that $c_l : \Omega \to \Omega$ is a continuous operator and that $c = c_L \circ c_{L-1} \circ \ldots \circ c_1$ is also continuous as composition of L continuous operators.

Point *(iii)* According to equation (1.36) based on multi-convexity of f and equation (1.37) that sets the definition of $r_l : \Omega \mapsto \Omega_l$, we know that:

$$f(\mathbf{v}) = \ell_l(\mathbf{v}_l, \mathbf{v}) \leq \ell_l(r_l(\mathbf{v}), \mathbf{v}) \leq f(\mathbf{v}_1, ..., \mathbf{v}_{l-1}, r_l(\mathbf{v}), \mathbf{v}_{l+1}, ..., \mathbf{v}_L) = f(c_l(\mathbf{v})). \tag{1.39}$$

This implies that updating $\mathbf{v}_l$ by $\hat{\mathbf{v}}_l = r_l(\mathbf{v})$ increases $f(\mathbf{v})$, or $f(\mathbf{v})$ stays the same. Moreover, the following inequality is deduced from (1.39) for each $l = 2, ..., L$:

$$f(c_{l-1} \circ ... \circ c_1(\mathbf{v})) \leq f(c_l \circ c_{l-1} \circ ... \circ c_1(\mathbf{v})). \tag{1.40}$$

This yields the desired inequalities for any $\mathbf{v} \in \Omega$:

$$f(\mathbf{v}) \leq f(c_1(\mathbf{v})) \leq f(c_2 \circ c_1(\mathbf{v})) \leq ... \leq f(c_L \circ ... \circ c_1(\mathbf{v})) = f(c(\mathbf{v})). \tag{1.41}$$

Point *(iv)* If $f(\mathbf{v}) = f(c(\mathbf{v}))$ for $\mathbf{v} \in \Omega$ then equation (1.41) implies

$$f(\mathbf{v}) = f(c_1(\mathbf{v})) = f(c_2 \circ c_1(\mathbf{v})) = ... = f(c_L \circ ... \circ c_1(\mathbf{v})) = f(c(\mathbf{v})). \tag{1.42}$$

Using equation (1.39), the equality $f(\mathbf{v}) = f(c_1(\mathbf{v}))$ implies $\ell_1(\mathbf{v}_1, \mathbf{v}) = \ell_1(r_1(\mathbf{v}), \mathbf{v})$ and therefore $\mathbf{v}_1 = r_1(\mathbf{v})$ as $r_1(\mathbf{v})$ is the unique maximizer of $\ell_1(r_1(\tilde{\mathbf{v}}_1), \mathbf{v})$ with respect to $\tilde{\mathbf{v}}_1 \in \Omega_1$. From this result, we deduce $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_L) = (r_1(\mathbf{v}), \mathbf{v}_2, \ldots, \mathbf{v}_L) = c_1(\mathbf{v})$ and then, by transitivity,

$$\mathbf{v} = c_1(\mathbf{v}) = c_2 \circ c_1(\mathbf{v}) = \ldots = c_L \circ \ldots \circ c_1(\mathbf{v}) = c(\mathbf{v}). \tag{1.43}$$

□

Proof of Proposition 1.5.1.

Point *(a)* Point (*iii*) of Lemma 1.5.2 implies that the sequence $f(\mathbf{v}^s)$ is monotonically increasing, and therefore, convergent as the continuous function f is bounded on the compact set Ω.

Point *(b)* If the infinite sequence $f(\mathbf{v}^s)$ has a finite number of distinct terms, it cannot be a strictly increasing sequence and consequently there exists some integer M such that $f(\mathbf{v}^0) < f(\mathbf{v}^1) < \ldots < f(\mathbf{v}^M) = (\mathbf{v}^{M+1})$. Then, Point (*iv*) of Lemma 1.5.2 implies that $\mathbf{v}^M$ is a fixed point of c.

Point *(c) to (f)* They are deduced from a direct application of Meyer's monotone convergence theorem (Theorem 3.1 in [Meyer, 1976]). This theorem gives quite general conditions under which a sequence $(\mathbf{v}^s)$ produced by an algorithm that monotonically increases a continuous objective function will converge. Meyer considered the case of a point-to-set operator $c : \Omega \mapsto \mathcal{P}(\Omega)$, where $\mathcal{P}(\Omega)$ is the set of all nonempty subsets of Ω. In this manuscript, c is a point-to-point operator and the conditions of Meyer's theorem reduce to the three following conditions (see [Fessler, 2004]): (1) c is a continuous operator; (2) c is strictly monotone (increasing) with respect to f ; and (3) c is uniformly compact on Ω. Condition (2) means that points (*iii*) and (*iv*) of Lemma 1.5.2 are verified. Condition (3) means that there exists a compact set $\mathcal{K}$ such that $c(\mathbf{v}) \in \mathcal{K}$ for all $\mathbf{v} \in \Omega$. According to Lemma 1.5.2, these three conditions are satisfied for Algorithm 1 and therefore, Meyer's theorem can be applied to any sequence $\mathbf{v}^s$ produced by the recurrence equation $\mathbf{v}^{s+1} = c(\mathbf{v}^s)$ with $\mathbf{v}^0 \in \Omega$. □

1.6 Conclusion

In this chapter, we have gathered the mathematical foundations that will be used throughout this manuscript. We have also described a very general and very simple optimization framework that enables to maximize a multi-convex function. It provides the algorithmic foundations of our developments. As we will see, especially in the next two chapters, this optimization framework offers a systematic approach for constructing globally convergent algorithms.

* * *
* *
*

Chapter 2

Global Regularized Generalized Canonical Correlation Analysis

Chapter Outline

The methods and principles contained in this chapter are the subject of a publication currently in preparation:

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Laurent Le Brusquet, Arthur Tenenhaus. *The Global Regularized Generalized Canonical Correlation Analysis.* In preparation.

2.1 Introduction

As described in Chapter 1, Regularized Generalized Canonical Correlation Analysis (RGCCA) is a general multiblock data analysis framework that encompasses several important multivariate analysis methods such as principal component analysis, partial least squares regression and several versions of generalized canonical correlation analysis and consensus PCA (see table 1.1 for an overview).

Regularized Generalized Canonical Correlation Analysis (RGCCA) belongs to the family of the sequential multiblock component methods [Tenenhaus et al., 2017]: at the first stage, block weight-vectors are computed as solution of some optimization problem; then, higher-stage block weight-vectors are computed on new blocks obtained by "deflation" of each block on the previous block components. Using a deflation procedure means that orthogonality constraints are imposed to the block components within each block. From an optimization point of view, this sequential approach may be seen as sub-optimal.

From that perspective, the global RGCCA optimization problem is proposed in this chapter. The objective of this approach is now to find a fixed number (say R) of components per block in one step by solving a single optimization problem. A globally convergent algorithm is proposed.

This chapter is structured as follows : for the sake of completeness, the optimization framework used to maximize the sequential RGCCA criterion is presented in details in section 2.2. Section 2.3 presents a novel RGCCA objective function (global RGCCA) that enables to compute all the components simultaneously. The global convergence of the global RGCCA algorithm is demonstrated. Finally, section 2.4 compares the sequential and the global approaches on simulation experiments and shows similar performances.

2.2 Sequential Regularized Generalized Canonical Correlation Analysis (RGCCA)

2.2.1 First-stage RGCCA block component

Let $\mathbf{X}_1, \ldots, \mathbf{X}_l, \ldots, \mathbf{X}_L$ be a collection of L data matrices. Each $I \times J_l$ data matrix $\mathbf{X}_l = [\mathbf{x}_{l1}, \ldots, \mathbf{x}_{lJ_l}]$ is a block and represents a set of J_l variables observed on I individuals. The number and the nature of the variables may differ from one block to another, but the individuals must be the same across blocks. We assume that all variables are centered. The most recent formulation of the RGCCA optimization problem [Tenenhaus et al., 2017] is:

$$\begin{aligned} \max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} & \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) \\ \text{s.t. } & \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l = 1,\ l = 1, \ldots, L \end{aligned} \tag{2.1}$$

where g, $\mathbf{C} \in \mathbb{R}^{L\times L}$ and $\mathbf{M}_l \in \mathbb{R}^{J_l \times J_l},\ l = 1, \ldots L$ are defined in Chapter 1, section 1.2.1. The optimization problem (2.1) can be simplified by considering the two following transforms $\mathbf{P}_l = I^{-1/2}\mathbf{X}_l\mathbf{M}_l^{-1/2}$

and $\mathbf{v}_l = \mathbf{M}_l^{1/2}\mathbf{w}_l$, which leads to:

$$\max_{\mathbf{v}_1,\ldots,\mathbf{v}_L} f(\mathbf{v}_1,\ldots,\mathbf{v}_L) = \sum_{k,l=1}^{L} c_{lk}\ \mathrm{g}\left(\mathbf{v}_k^\top \mathbf{P}_k^\top \mathbf{P}_l \mathbf{v}_l\right) \tag{2.2}$$

$$\text{s.t. } \mathbf{v}_l^\top \mathbf{v}_l = 1, l = 1,\ldots,L. \tag{2.3}$$

2.2.2 The RGCCA algorithm

The convexity and continuous differentiability of the scheme function g imply that the objective function f defined in (2.2) is a continuously differentiable multi-convex[1] function. Consequently, the maximization of (2.2) subject to (2.3) can be cast under the general optimization framework presented in section 1.5. Under this framework, the function f, defined in equation (2.2) over different parameter vectors $(\mathbf{v}_1,\ldots,\mathbf{v}_L)$, is maximized by updating each of the parameter vectors in turn, keeping the others fixed. Hence, we want to find an update $\hat{\mathbf{v}}_l \in \Omega_l = \left\{\mathbf{v}_l \in \mathbb{R}^{J_l}; \|\mathbf{v}_l\|_2 = 1\right\}$ such that $f(\mathbf{v}) \leq f(\mathbf{v}_1,...,\mathbf{v}_{l-1},\hat{\mathbf{v}}_l,\mathbf{v}_{l+1},...,\mathbf{v}_L)$, where $\mathbf{v} = (\mathbf{v}_1;\ldots;\mathbf{v}_L)$. Following section 1.5.2, this update is obtained by considering the following optimization problem:

$$\hat{\mathbf{v}}_l = \underset{\tilde{\mathbf{v}}_l \in \Omega_l}{\operatorname{argmax}}\ \nabla_l f(\mathbf{v})^\top \tilde{\mathbf{v}}_l = \frac{\nabla_l f(\mathbf{v})}{\|\nabla_l f(\mathbf{v})\|_2} := r_l(\mathbf{v}), \tag{2.4}$$

where $\nabla_l f(\mathbf{v})$ is the partial gradient of $f(\mathbf{v})$ with respect to $\mathbf{v}_l$:

$$\nabla_l f(\mathbf{v}) = 2\sum_{k=1}^{L} c_{lk}\ \mathrm{g}'(\mathbf{v}_l{}^\top \mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k)\mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k = \mathbf{P}_l^\top \mathbf{z}_l \tag{2.5}$$

where $\mathbf{z}_l$, called the inner component, is defined as $\mathbf{z}_l = 2\sum_{k=1}^{L} c_{lk}\ \mathrm{g}'(\mathbf{v}_l{}^\top \mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k)\mathbf{P}_k \mathbf{v}_k$. The entire RGCCA algorithm is subsumed in Algorithm 2.

Algorithm 2 Regularized Generalized Canonical Correlation Analysis (RGCCA) algorithm

1: **Data:** $\mathbf{X}_1,\ldots,\mathbf{X}_L$, $\mathbf{M}_1,\ldots,\mathbf{M}_L$, g, ε, $\mathbf{C}$
2: **Result:** $\mathbf{v}_1^s,\ldots,\mathbf{v}_L^s$ (solution of (2.2) subject to (2.3))
3: **Initialization:** random unit-norm $\mathbf{v}_l^0, l = 1,\ldots,L$, $s = 0$;
4: **repeat**
5: **for** $l = 1$ **to** L **do**
6: $\mathbf{v}_l^{s+1} = \dfrac{\nabla_l f(\mathbf{v}_1^{s+1},\ldots,\mathbf{v}_{l-1}^{s+1},\mathbf{v}_l^s,\mathbf{v}_{l+1}^s,\ldots,\mathbf{v}_L^s)}{\|\nabla_l f(\mathbf{v}_1^{s+1},\ldots,\mathbf{v}_{l-1}^{s+1},\mathbf{v}_l^s,\mathbf{v}_{l+1}^s,\ldots,\mathbf{v}_L^s)\|_2}$
7: **end for**
8: $s = s + 1$;
9: **until** $f(\mathbf{v}_1^{s+1},\ldots,\mathbf{v}_L^{s+1}) - f(\mathbf{v}_1^s,\ldots,\mathbf{v}_L^s) < \varepsilon$

At the end of the Algorithm 2, the original weight vectors $\mathbf{w}_l$ are recovered by $\mathbf{w}_l = (\mathbf{M}_l)^{-1/2}\mathbf{v}_l$.

In the case of a single block ($L = 1$), Algorithm 2 is similar to the gradient-based algorithm proposed by [Journée et al., 2010] for maximizing a convex function of several variables with spherical constraints (see Problem 27, p. 529).

[1]When one element of the diagonal of the design matrix $\mathbf{C}$ is equal to 1, additional conditions have to be imposed on the scheme function g in order for f to still be multi-convex. For example, when g is twice differentiable, a sufficient condition is that $\forall x \in \mathbb{R}_+,\ \ g'(x) \geq 0$. All scheme functions g considered in this document respect this condition and the case where one element of the diagonal of the design matrix $\mathbf{C}$ is equal to 1 is never considered in our examples.

2.2.3 Convergence properties of the RGCCA algorithm

The convergence properties subsumed in Proposition 1.5.1 are satisfied for Algorithm 2. In order to show that Proposition 1.5.1 holds for the RGCCA algorithm, point (*i-iv*) of Lemma 1.5.2 are demonstrated below.

Proof of Lemma 1.5.2 for the RGCCA Algorithm.

Point *(i)* $\Omega_l = \left\{\mathbf{v}_l \in \mathbb{R}^{J_l}; \|\mathbf{v}_l\|_2 = 1\right\}$ is the ℓ_2-sphere of radius 1 and is a compact set. As $\Omega = \Omega_1 \times \ldots \times \Omega_L$ is the Cartesian product of L compact sets, it is compact.

Point *(ii)* $r_l(\mathbf{v})$ defined in equation (2.4) is the orthogonal projection of $\nabla_l f(\mathbf{v})$ onto the ℓ_2-sphere of radius 1. Under the assumption made in Chapter 1 section 1.5.2 paragraph 3, $r_l(\mathbf{v})$ exists and is unique.

Point *(iii)* The demonstration presented in Chapter 1 for point *(iii)* of Lemma 1.5.2 still holds here.

Point *(iv)* The proof is based on the uniqueness of $r_l(\mathbf{v})$ defined in equation (2.4). □

Therefore, whatever the starting point, Algorithm 2 converges towards a stationary point of the RGCCA optimization problem.

2.2.4 Higher-stage RGCCA block component

The optimization problem (2.1) is associated with the first component of RGCCA. A deflation procedure was proposed in order to extract more than one component.

In this section, let $\mathbf{y}_l^{(1)} = \mathbf{X}_l \mathbf{w}_l^{(1)},\ l = 1, \ldots, L$ be the first-stage block components solution of optimization problem (2.1). Seeking the second-stage block components $\mathbf{y}_l^{(2)} = \mathbf{X}_l \mathbf{w}_l^{(2)},\ l = 1, \ldots, L$, implies that some constraints must be added to the optimization problem. For example, orthogonality constraints can be considered, leading to the following formulation of the RGCCA optimization problem at the second stage:

$$\begin{aligned} &\max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) \\ &\text{s.t. } \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l = 1, \text{ and } \mathbf{w}_l^\top \mathbf{X}_l^\top \mathbf{y}_l^{(1)} = 0,\ l = 1, \ldots, L \end{aligned} \tag{2.6}$$

For each block, the resulting second-stage block component $\mathbf{y}_l^{(2)}$ is uncorrelated with the first-stage block component $\mathbf{y}_l^{(1)}$.

Problem (2.6) is easy to solve with the first-stage RGCCA algorithm by using a deflation procedure. This procedure consists in replacing a block $\mathbf{X}_l$ by the residual $\mathbf{X}_l^{(1)} = \mathbf{X}_l - \mathbf{y}_l^{(1)} \left(\mathbf{y}_l^{(1)^\top} \mathbf{y}_l^{(1)}\right)^{-1} \mathbf{y}_l^{(1)^\top} \mathbf{X}_l$ related to the regression of $\mathbf{X}_l$ on the first-stage block component $\mathbf{y}_l^{(1)}$. Moreover, as $\mathbf{y}_l^{(1)} = \mathbf{X}_l \mathbf{w}_l^{(1)}$, the range space of $\mathbf{X}_l^{(1)}$ is included in the range space of $\mathbf{X}_l$, meaning that any block component $\mathbf{y}_l$ belonging to the range space of $\mathbf{X}_l^{(1)}$ can also be expressed in term of the original block $\mathbf{X}_l$:

$$\mathbf{y}_l = \mathbf{X}_l^{(1)} \tilde{\mathbf{w}}_l = \mathbf{X}_l \mathbf{w}_l. \tag{2.7}$$

Furthermore, by assuming that each $\mathbf{X}_l$ is of full-rank, then $\mathbf{w}_l$ can be expressed in terms of $\tilde{\mathbf{w}}_l$: $\mathbf{w}_l = \left(\mathbf{X}_l^\top \mathbf{X}_l\right)^{-1} \mathbf{X}_l^\top \mathbf{X}_l^{(1)} \tilde{\mathbf{w}}_l$. Thus, the constraint $\mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l = 1$ can be rewritten in terms of $\tilde{\mathbf{w}}_j$:

$$1 = \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l = \tilde{\mathbf{w}}_l^\top {\mathbf{X}_l^{(1)}}^\top \mathbf{X}_l \left(\mathbf{X}_l^\top \mathbf{X}_l\right)^{-1} \mathbf{M}_l \left(\mathbf{X}_l^\top \mathbf{X}_l\right)^{-1} \mathbf{X}_l^\top \mathbf{X}_l^{(1)} \tilde{\mathbf{w}}_l \tag{2.8}$$

Setting $\mathbf{M}_l^{(1)} = {\mathbf{X}_l^{(1)}}^\top \mathbf{X}_l \left(\mathbf{X}_l^\top \mathbf{X}_l\right)^{-1} \mathbf{M}_l \left(\mathbf{X}_l^\top \mathbf{X}_l\right)^{-1} \mathbf{X}_l^\top \mathbf{X}_l^{(1)}$, optimization problem (2.6) becomes equivalent to:

$$\begin{aligned} &\max_{\tilde{\mathbf{w}}_1,\ldots,\tilde{\mathbf{w}}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1} \tilde{\mathbf{w}}_k^\top {\mathbf{X}_k^{(1)}}^\top \mathbf{X}_l^{(1)} \tilde{\mathbf{w}}_l\right) \\ &\text{s.t. } \tilde{\mathbf{w}}_l^\top \mathbf{M}_l^{(1)} \tilde{\mathbf{w}}_l = 1,\ l = 1,\ldots,L \end{aligned} \tag{2.9}$$

So the first-stage RGCCA algorithm can be used to solve (2.9) and leads to the block weight vector $\tilde{\mathbf{w}}_l^{(2)}$ and the second-stage block component $\mathbf{y}_l^{(2)} = \mathbf{X}_l^{(1)} \tilde{\mathbf{w}}_l^{(2)}$. This deflation procedure can be iterated in a very flexible way. For example, in a supervised situation where we want to predict a block based on other blocks, it might be interesting to apply this deflation procedure to all blocks except the one to predict.

However, maximizing successive criteria may be seen as suboptimal from an optimization point of view where a single global criterion might be preferred. Secondly, with this sequential procedure, if the first components are poorly estimated, this is going to affect the estimation of the following components, which is a major drawback. Thirdly, as seen previously, we have to assume that each block matrix $\mathbf{X}_l$ is of full-rank in order to properly define the constraint matrix $\mathbf{M}_l^{(1)}$. Nonetheless, this is not always true.

For those reasons, we propose the global RGCCA objective function that allows estimating all the block-components simultaneously.

2.3 Global RGCCA

The global RGCCA optimization problem is defined as the following optimization problem:

$$\max_{\mathbf{W}_1,\ldots,\mathbf{W}_L} f\left(\mathbf{W}_1,\ldots,\mathbf{W}_L\right) = \sum_{k,l=1}^{L} c_{kl} \operatorname{Tr}\Big(\ \mathrm{g}\left(I^{-1} \mathbf{W}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{W}_l\right)\Big) \tag{2.10}$$

$$\text{s.t. } \mathbf{W}_l^\top \mathbf{M}_l \mathbf{W}_l = \mathbf{I}_R,\ l = 1,\ldots,L. \tag{2.11}$$

where $\mathbf{W}_l = \left[\mathbf{w}_l^{(1)},\ldots,\mathbf{w}_l^{(R)}\right]$ is a $J_l \times R$ weight matrix defined as the concatenation of the R weight vectors $\mathbf{w}_l^{(r)}$. As previously, the design matrix $\mathbf{C} = \{c_{lk}\}$ is a symmetric $L \times L$ matrix of non-negative entries describing the network of connections between blocks that the user wants to take into account. Usually, $c_{lk} = 1$ for two connected blocks and 0 otherwise. The function g is convex, differentiable and element-wise from $\mathcal{M}_R(\mathbb{R})$ to $\mathcal{M}_R(\mathbb{R})$ (the set of real square matrices of size R). Constraint (2.11) is an orthonormal constraint on the weight matrix $\mathbf{W}_l$ in the metric space defined by the positive definite matrix $\mathbf{M}_l$.

As the Trace operator is used in the criterion of optimization problem (2.10), it means that it focuses only on maximizing the covariance between components of same levels. Indeed, for two connected blocks l and k, only ${\mathbf{y}_l^{(r)}}^\top \mathbf{y}_k^{(r)},\ r = 1,\ldots,R$ are part of the criterion, where $\mathbf{y}_l^{(r)}$ (resp. $\mathbf{y}_k^{(r)}$) is the r^{th} component of block l (resp. k).

The optimization criterion (2.10) can be simplified by considering the transformations $\mathbf{P}_l = I^{-1/2}\mathbf{X}_l\mathbf{M}_l^{-1/2},\ \mathbf{V}_l = \mathbf{M}_l^{1/2}\mathbf{W}_l,\ l = 1,\ldots,L$, which leads to:

$$\max_{\mathbf{V}_1,\ldots,\mathbf{V}_L} f(\mathbf{V}_1,\ldots,\mathbf{V}_L) = \sum_{k,l=1}^{L} c_{kl}\,\mathrm{Tr}\left(\mathrm{g}\left(\mathbf{V}_k^\top\mathbf{P}_k^\top\mathbf{P}_l\mathbf{V}_l\right)\right) \tag{2.12}$$

$$\text{s.t. } \mathbf{V}_l^\top\mathbf{V}_l = \mathbf{I}_R,\ l = 1,\ldots,L. \tag{2.13}$$

As g is element-wise and using the definition of the Trace operator for matrices, we have:

$$\mathrm{Tr}\left(\mathrm{g}\left(\mathbf{V}_k^\top\mathbf{P}_k^\top\mathbf{P}_l\mathbf{V}_l\right)\right) = \sum_{r=1}^{R}\mathrm{g}\left(\mathbf{v}_k^{(r)\top}\mathbf{P}_k^\top\mathbf{P}_l\mathbf{v}_l^{(r)}\right) \tag{2.14}$$

Therefore, the optimization problem (2.10) can be re-written as follows:

$$f(\mathbf{V}_1,\ldots,\mathbf{V}_L) = \sum_{k,l=1}^{L} c_{kl}\sum_{r=1}^{R}\mathrm{g}\left(\mathbf{v}_k^{(r)\top}\mathbf{P}_k^\top\mathbf{P}_l\mathbf{v}_l^{(r)}\right) \tag{2.15}$$

Furthermore, let us introduce $\mathbf{v}_l = \mathrm{vec}(\mathbf{V}_l) = \left(\mathbf{v}_1^{(1)};\ldots;\mathbf{v}_1^{(R)}\right)$ (see section 1.3.3.1 for details). Thus, the following equality stands: $\mathbf{v}_k^{(r)\top}\mathbf{P}_k^\top\mathbf{P}_l\mathbf{v}_l^{(r)} = \mathbf{v}_k^\top\left(\mathbf{J}_R^{(r)}\otimes\mathbf{P}_k^\top\mathbf{P}_l\right)\mathbf{v}_l,\ l = 1,\ldots,L$, where $\mathbf{J}_R^{(r)}$ is a diagonal matrix of size R such that all its elements are equal to zero except for $\left(\mathbf{J}_R^{(r)}\right)_{rr} = 1$. Hence, another formulation of f is possible:

$$f(\mathbf{V}_1,\ldots,\mathbf{V}_L) = \sum_{k,l=1}^{L} c_{kl}\sum_{r=1}^{R}\mathrm{g}\left(\mathbf{v}_k^\top\left(\mathbf{J}_R^{(r)}\otimes\mathbf{P}_k^\top\mathbf{P}_l\right)\mathbf{v}_l\right) \tag{2.16}$$

This last formulation emphasizes that f is multi-convex according to each $\mathbf{v}_l = \mathrm{vec}(\mathbf{V}_l),\ l = 1,\ldots,L$; that is, for each l, f is a convex function of $\mathbf{v}_l$ while all others $\mathbf{v}_k, k \neq l$ are fixed. In others words, f is multi-convex according to each $\mathbf{V}_l,\ l = 1,\ldots,L$.

In the next section, the global RGCCA algorithm proposed to solve (2.12) subject to (2.13) is detailed.

2.3.1 The Global RGCCA Algorithm

The objective function of the global RGCCA criterion is a continuously differentiable multi-convex function, meaning that we can use the optimization framework described in section 1.5 to solve the optimization problem (2.12)-(2.13). f defined in equation (2.12) over different parameter matrices $(\mathbf{V}_1,\ldots,\mathbf{V}_L)$, is approached by updating each of the parameter matrices in turn, keeping the others fixed. Let $\nabla_l^{(r)}f(\mathbf{V})$, be the partial gradient of $f(\mathbf{V})$ with respect to $\mathbf{v}_l^{(r)}$, where $\mathbf{V} = (\mathbf{V}_1;\ldots;\mathbf{V}_L)$. From equation (2.15), this partial gradient can be written as:

$$\nabla_l^{(r)}f(\mathbf{V}) = 2\sum_{k=1}^{L} c_{lk}\ \mathrm{g}'(\mathbf{v}_l^{(r)\top}\mathbf{P}_l^\top\mathbf{P}_k\mathbf{v}_k^{(r)})\mathbf{P}_l^\top\mathbf{P}_k\mathbf{v}_k^{(r)} = \mathbf{P}_l^\top\mathbf{z}_l^{(r)} \tag{2.17}$$

where $\mathbf{z}_l^{(r)}$ is defined as $\mathbf{z}_l^{(r)} = 2\sum_{k=1}^{L} c_{lk}\ \mathrm{g}'(\mathbf{v}_l^{(r)\top}\mathbf{P}_l^\top\mathbf{P}_k\mathbf{v}_k^{(r)})\mathbf{P}_k\mathbf{v}_k^{(r)}$. Let $\mathbf{Z}_l = \left[\mathbf{z}_l^{(1)},\ldots,\mathbf{z}_l^{(R)}\right]$, the partial gradient of $f(\mathbf{V})$ with respect to $\mathbf{V}_l$ can be written as:

$$\nabla_l f(\mathbf{V}) = \left[\nabla_l^{(1)}f(\mathbf{V}),\ldots,\nabla_l^{(R)}f(\mathbf{V})\right] = \left[\mathbf{P}_l^\top\mathbf{z}_l^{(1)},\ldots,\mathbf{P}_l^\top\mathbf{z}_l^{(R)}\right] = \mathbf{P}_l^\top\mathbf{Z}_l \tag{2.18}$$

We want to find an update $\hat{\mathbf{V}}_l \in \Omega_l = \left\{\mathbf{V}_l \in \mathbb{R}^{J_l \times R};\ \mathbf{V}_l^\top \mathbf{V}_l = \mathbf{I}_R\right\}$ such that $f(\mathbf{V}) \leq f(\mathbf{V}_1, \ldots, \mathbf{V}_{l-1}, \hat{\mathbf{V}}_l, \mathbf{V}_{l+1}, \ldots, \mathbf{V}_L)$. Considering that a convex function lies above its linear approximation at $\mathbf{V}_l$ for any $\tilde{\mathbf{V}}_l \in \Omega_l$, and introducing $v_{ij}^l = (\mathbf{V}_l)_{(ij)}$, the following inequality holds:

$$\begin{aligned} f(\mathbf{V}_1, ..., \mathbf{V}_{l-1}, \tilde{\mathbf{V}}_l, \mathbf{V}_{l+1}, \ldots, \mathbf{V}_L) &\geq f(\mathbf{V}) + \sum_{i=1}^{J_l} \sum_{j=1}^{R} \frac{\partial f}{\partial v_{ij}^l} (\tilde{v}_{ij}^l - v_{ij}^l) \\ &\geq f(\mathbf{V}) + \operatorname{Tr}\left(\nabla_l f(\mathbf{V})^\top \left(\tilde{\mathbf{V}}_l - \mathbf{V}_l\right)\right) := \ell_l(\tilde{\mathbf{V}}_l, \mathbf{V}) \end{aligned} \tag{2.19}$$

On the right-hand side of the inequality (2.19), only the term $\operatorname{Tr}\left(\nabla_l f(\mathbf{V})^\top \tilde{\mathbf{V}}_l\right)$ is relevant to $\tilde{\mathbf{V}}_l$ and the solution that maximizes the minorizing function $\ell_l(\tilde{\mathbf{V}}_l, \mathbf{V})$ over $\tilde{\mathbf{V}}_l \in \Omega_l$ is obtained by considering the following optimization problem:

$$\hat{\mathbf{V}}_l = \underset{\tilde{\mathbf{V}}_l^\top \tilde{\mathbf{V}}_l = \mathbf{I}_R}{\operatorname{argmax}} \operatorname{Tr}\left(\nabla_l f(\mathbf{V})^\top \tilde{\mathbf{V}}_l\right) := r_l(\mathbf{V}). \tag{2.20}$$

According to Theorem A.4.2 [Adachi, 2016], p. 270, solution of optimization problem (2.20) is:

$$\hat{\mathbf{V}}_l = \mathbf{Q}_l \mathbf{R}_l^\top, \tag{2.21}$$

where $\mathbf{Q}_l \in \mathbb{R}^{J_l \times R}$ and $\mathbf{R}_l \in \mathbb{R}^{R \times R}$ are given by the rank-R Singular Value Decomposition (SVD) of $\nabla_l f(\mathbf{V})$ defined as $\nabla_l f(\mathbf{V}) = \mathbf{Q}_l \boldsymbol{\Delta}_l \mathbf{R}_l^\top$, with $\mathbf{Q}_l^\top \mathbf{Q}_l = \mathbf{R}_l^\top \mathbf{R}_l = \mathbf{R}_l \mathbf{R}_l^\top = \mathbf{I}_R$ and $\boldsymbol{\Delta}_l$ a $R \times R$ diagonal matrix whose diagonal elements are all positive and in decreasing order.

The entire Global RGCCA algorithm is described in Algorithm 3.

Algorithm 3 Global Regularized Generalized Canonical Correlation Analysis algorithm

Data: $\mathbf{X}_1, \ldots, \mathbf{X}_L$, $\mathbf{M}_1, \ldots, \mathbf{M}_L$, g, ε, $\mathbf{C}$, R
Result: $\mathbf{V}_1^s, \ldots, \mathbf{V}_L^s$ (approximate solution of (2.12) subject to (2.13))
Initialization: choose random matrix $\mathbf{V}_l^0, l = 1, \ldots, L$, such that $\mathbf{V}_l^{0^\top} \mathbf{V}_l^0 = \mathbf{I}_R$;
$s = 0$
repeat
  for $l = 1$ **to** L **do**

$$\mathbf{V}_l^{s+1} = r_l\left(\mathbf{V}_1^{s+1}, ..., \mathbf{V}_{l-1}^{s+1}, \mathbf{V}_l^s, \mathbf{V}_{l+1}^s, ..., \mathbf{V}_L^s\right) = \mathbf{Q}_l^s \mathbf{R}_l^{s^\top} \tag{2.22}$$

  end for
  $s = s + 1$;
until $f(\mathbf{V}_1^{s+1}, \ldots, \mathbf{V}_L^{s+1}) - f(\mathbf{V}_1^s, \ldots, \mathbf{V}_L^s) < \varepsilon$

where $\mathbf{Q}_l^s \in \mathbb{R}^{J_l \times R}$ and $\mathbf{R}_l^s \in \mathbb{R}^{R \times R}$ are given by the rank-R Singular Value Decomposition (SVD) of $\nabla_l^s f(\mathbf{V}) = \sum_{k=1}^{l-1} \mathbf{P}_l^\top \mathbf{P}_k \mathbf{V}_k^{s+1} \mathbf{D}_{lk}^{s,s+1} + \sum_{k=l}^{L} \mathbf{P}_l^\top \mathbf{P}_k \mathbf{V}_k^s \mathbf{D}_{lk}^{s,s}$ of dimension $J_l \times R$ with $\mathbf{D}_{lk}^{s,t}$ a diagonal matrix of size R whose r^{th} element equals $2c_{lk}\ \mathrm{g}'\left(\mathbf{v}_l^{(r),s^\top} \mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k^{(r),t}\right)$.

The original weight matrix $\mathbf{W}_l$ is recovered by $\mathbf{W}_l = \mathbf{M}_l^{-1/2} \mathbf{V}_l$.

2.3.2 Convergence properties of the Global RGCCA algorithm

The operators defined in section 1.5.3 are extended here for matrices. $c_l : \Omega \mapsto \Omega$ is an operator defined as $c_l(\mathbf{V}) = (\mathbf{V}_1; \ldots; \mathbf{V}_{l-1}; r_l(\mathbf{V}); \mathbf{V}_{l+1}; \ldots; \mathbf{V}_L)$ with $r_l(\mathbf{V})$ introduced in equation (2.20) and $c : \Omega \mapsto \Omega$ is defined as $c = c_L \circ c_{L-1} \circ \ldots \circ c_1$, where $\circ$ stands for the function composition operator. We consider the sequence $\{\mathbf{V}^s = (\mathbf{V}_1^s; \ldots; \mathbf{V}_L^s)\}$ generated by Algorithm 3. Using the operator c, the «for loop» inside Algorithm 3 can be replaced by the following recurrence relation: $\mathbf{V}^{s+1} = c(\mathbf{V}^s)$.

The convergence properties subsumed in Proposition 1.5.1 are satisfied for Algorithm 3. In order to show that Proposition 1.5.1 holds for the global RGCCA algorithm, point (*i-iv*) of Lemma 1.5.2 are demonstrated below.

Proof of Lemma 1.5.2 for the global RGCCA Algorithm.

Point *(i)* $\Omega_l = \left\{\mathbf{V}_l \in \mathbb{R}^{J_l \times R};\ \mathbf{V}_l^\top \mathbf{V}_l = \mathbf{I}_R\right\}$ is the set of real orthonormal matrices $\mathcal{O}_{J_l \times R}$ of size $J_l \times R$ which is a compact set. Thus, Ω is compact as product of L compact sets.

Point *(ii)* As long as the solution of optimization problem (2.20) exists and is unique, the demonstration of the point (*ii*) of the Lemma 1.5.2 in Chapter 1 still holds. For global RGCCA, $r_l(\mathbf{V})$ is the rank-R SVD of $\nabla_l f(\mathbf{V})$ and suffers from a sign indeterminacy. However, this first drawback can be circumvented numerically by imposing for every iteration that each block weight vector $\mathbf{v}^{(r),s}$ is positively correlated with its corresponding value after the first iteration $\mathbf{v}_l^{(r),1}$. Moreover, the uniqueness breaks down when several non-null singular values of $\nabla_l f(\mathbf{V})$ are equal and some of their associated singular vectors lie within the chosen R-dimensional subspace and some others outside of it. Nonetheless, with real data, this hardly ever occurs and we will here disregard this possibility.

Point *(iii)* The demonstration presented in Chapter 1 for point *(iii)* of Lemma 1.5.2 still holds here.

Point *(iv)* The proof is based on the uniqueness of $r_l(\mathbf{V})$ defined in equation (2.4). Under mild conditions (see the discussion above), this point is satisfied for global RGCCA. □

Therefore, the RGCCA algorithm and the global RGCCA algorithm both bear the same convergence properties that are described in Proposition 1.5.1.

2.4 Simulation experiments

In this section, we compare the performances of global RGCCA with sequential RGCCA.

2.4.1 Data Generation

For this simulation experiment, we consider $L = 2$ blocks of the same dimension with $N = 200$ and $J_1 = J_2 = 30$. Each block is simulated according to the following generative matrix model:

$$\mathbf{X}_l = \eta \mathbf{Y}_l \mathbf{W}_l^\top + \frac{\|\mathbf{Y}_l \mathbf{W}_l^\top\|_F}{\|\mathbf{E}_l\|_F} \mathbf{E}_l,\ l = 1, 2, \tag{2.23}$$

where $\mathbf{W}_l \in \mathbb{R}^{J_l \times R^\star}$ is a randomly generated orthonormal matrix. In this experiment, we consider $R^\star = 4$ components. Furthermore, $[\mathbf{Y}_1, \mathbf{Y}_2] \in \mathbb{R}^{N \times 2R^\star}$ is randomly generated such that its columns are orthonormal, except for $\mathbf{y}_{11}^\top \mathbf{y}_{21} = 1,\ \mathbf{y}_{12}^\top \mathbf{y}_{22} = 0.8,\ \mathbf{y}_{13}^\top \mathbf{y}_{23} = 0.6,$ and $\mathbf{y}_{14}^\top \mathbf{y}_{24} = 0.4$.

The noise matrix $\mathbf{E}_l \in \mathbb{R}^{N \times J_l}$ is defined such that its entries are drawn from a standardized normal distribution. Finally, the Signal to Noise Ratio (SNR) is equal to $20log_{10}(\eta)$ which enables η to drive the SNR.

Let $\mathbf{W}_l$ and $\hat{\mathbf{W}}_l,\ l = 1, 2$ be respectively the original and the estimated block weight matrices. We quantify how well the estimated block weight matrices match the original ones using the accuracy (ACC) defined as:

$$ACC = \frac{1}{LR} \sum_{l=1}^{L} \sum_{r=1}^{R} |\hat{\mathbf{w}}_l^{(r)\top} \mathbf{w}_l^{(r)}|, \tag{2.24}$$

where $\hat{\mathbf{w}}_l^{(r)}$ and $\mathbf{w}_l^{(r)}$ are the r^{th} column of matrices $\hat{\mathbf{W}}_l$ and $\mathbf{W}_l$ respectively.

2.4.2 Results

We consider five values of $\eta \in \{0.2, 0.3, 1, 2, 5\}$. For each value of η, 100 different datasets were generated according to equation (2.23). For each dataset, global RGCCA and sequential RGCCA were applied to extract $R = 4$ components.

For the two procedures, $c_{12} = c_{21} = 1$ and $c_{11} = c_{22} = 0$, g was set to the square function (or the element-wise square function) and $\mathbf{M}_1 = \mathbf{M}_2 = \mathbf{I}_R$. As each method presents potentially many local maxima, multiple starts were performed (i.e., SVD-based initialization as well as 10 random starts) and the best solution was kept [Acar et al., 2013, ten Berge, 1993].

Moreover, inspired from [Acar et al., 2011], in order to evaluate to what extent global RGCCA is impacted by a misspecification of the number of factors to extract, global RGCCA is also evaluated in condition where $R = R^\star + 1 = 5$ components are extracted.

The measure of accuracy (ACC) defined in (2.24) was then computed for each dataset and each procedure. When $R^\star + 1$ components are extracted, the ACC is computed with the $R^\star = 4$ components leading to the highest ACC value. The mean and standard deviation (std) of ACC for each value of η are reported in table 2.1 along with the median (MD) of the number of iterations (and its interquartile range (IQR)) and the execution time for the best solution. The mean and standard deviation of the criterion value of each method are also reported (column CRIT).

It appears that regardless of the SNR, each procedure performs very similarly, with a slight improvement for global RGCCA in term of ACC and value of the criterion (CRIT). The ACC is always

improved with a higher value of SNR. For $\eta = 0.2$ and $\eta = 0.3$, global RGCCA with $R^\star + 1$ components performs even better than the other methods in term of ACC. The criterion of global RGCCA with $R^\star + 1$ components is always higher as it is computed with one extra component in comparison to the other methods. For $\eta = 1, 2, 5$, the results of Global RGCCA for either $R^\star$ or $R^\star + 1$ are almost equals in term of ACC and value of the criterion. Considering the number of iterations and the execution time, they are both always higher for global RGCCA with $R^\star$ components compared to the sequential RGCCA, with a factor of 2 for the number of iterations and 10 for the execution time. global RGCCA with $R^\star + 1$ components has almost the same number of iterations and execution time as global RGCCA with $R^\star$, slightly higher. For both sequential and global RGCCA, on each dataset, the different initializations lead to the same solution.

Table 2.1 – For each value $\eta \in \{0.2, 0.3, 1, 2, 5\}$, 100 datasets were generated. For each dataset, RGCCA was applied to extract $R = 4$ components either with a sequential or a global procedure. Global RGCCA was also applied to extract $R = 5$ components. For each method, the same stopping criterion is taken with $\varepsilon = 10^{-8}$. The mean and standard deviation (std) of ACC (defined in (2.24)) for each value of η are reported along with the median (MD) of the number of iterations (and its interquartile range (IQR)) and the execution time for the best solution. The mean and standard deviation of the criterion value of each method are also reported (column CRIT).

SNR	R	Algorithm	ACC (mean ± std)	Iter (MD - IQR)	Time(s) (mean ± std)	CRIT (mean ± std)
$\eta = 0.2$	$R^\star$	sequential RGCCA	0.311 ± 0.057	261 - 115	0.4 ± 0.1	(3.06 ± 0.23)1e-8
		global RGCCA	0.314 ± 0.064	650 - 490	5.1 ± 2.0	(3.07 ± 0.23)1e-8
	$R^\star + 1$	global RGCCA	0.326 ± 0.066	761 - 472	6.4 ± 2.6	(3.59 ± 0.25)1e-8
$\eta = 0.3$	$R^\star$	sequential RGCCA	0.505 ± 0.079	172 - 64	0.3 ± 0.1	(8.80 ± 0.86)1e-9
		global RGCCA	0.510 ± 0.076	382 - 365	3.3 ± 1.8	(8.81 ± 0.86)1e-9
	$R^\star + 1$	global RGCCA	0.516 ± 0.079	492 - 281	4.9 ± 2.4	(10.00 ± 0.88)1e-9
$\eta = 1$	$R^\star$	sequential RGCCA	0.953 ± 0.016	61 - 12	0.1 ± 0.0	(2.96 ± 0.14)1e-9
		global RGCCA	0.956 ± 0.014	113 - 55	1.0 ± 1.0	(2.97 ± 0.14)1e-9
	$R^\star + 1$	global RGCCA	0.956 ± 0.014	126 - 67	1.1 ± 0.9	(2.97 ± 0.14)1e-9
$\eta = 2$	$R^\star$	sequential RGCCA	0.989 ± 3e-3	56 - 4	0.1 ± 0.0	(2.77 ± 0.06)1e-9
		global RGCCA	0.990 ± 3e-3	105 - 38	0.6 ± 0.1	(2.77 ± 0.06)1e-9
	$R^\star + 1$	global RGCCA	0.990 ± 3e-3	102 - 36	0.8 ± 0.2	(2.78 ± 0.06)1e-9
$\eta = 5$	$R^\star$	sequential RGCCA	0.9983 ± 5e-4	54 - 5	0.1 ± 0.0	(2.72 ± 0.03)1e-9
		global RGCCA	0.9984 ± 5e-4	106 - 27	0.6 ± 0.1	(2.72 ± 0.03)1e-9
	$R^\star + 1$	global RGCCA	0.9984 ± 5e-4	104 - 32	0.7 ± 0.1	(2.72 ± 0.03)1e-9

2.5 Conclusion

In this Chapter, two strategies to compute higher-level components were presented: a sequential approach that relies on deflation and a global one that extracts all the components simultaneously. We have shown that the RGCCA algorithm has global convergence properties, which remains true for global RGCCA under mild conditions (i.e. uniqueness of the non-null singular values). Both approaches were compared on simulations and lead to very similar results.

* * *
* *
*

Chapter 3

Multiway Generalized Canonical Correlation Analysis (MGCCA)

Chapter Outline

The methods and principles described in this chapter were presented at national conferences or international journals and are also subject to publication in preparation :

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Laurent Le Brusquet, Arthur Tenenhaus. *Multiway Generalized Canonical Correlation Analysis.* Chimiométrie XIX, Paris, France, 2018. Oral.

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Giulia Gennari, Ghislaine Dehaene-Lambertz, Laurent Le Brusquet, Arthur Tenenhaus. *Multiway Generalized Canonical Correlation Analysis.* Biostatistics, 2020.

Fabien Girka, Pierrick Chevalier, **Arnaud Gloaguen**, Laurent Le Brusquet, Arthur Tenenhaus. *Rank-R Multiway Logistic Regression.* 52ème Journées de Statistique (JDS), France, 2020. Accepted.

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Laurent Le Brusquet, Arthur Tenenhaus. *The Global Multiway Generalized Canonical Correlation Analysis.* In preparation.

REGULARIZED Generalized Canonical Correlation Analysis (RGCCA) is presented in Chapters 1 and 2 as a general multiblock data analysis framework. In this chapter, we extend sequential RGCCA and global RGCCA to the case where at least one block has a tensor structure. These methods are called sequential Multiway Generalized Canonical Correlation Analysis and global Multiway Generalized Canonical Correlation Analysis (MGCCA). Two algorithms are proposed and convergence properties of these two algorithms are studied. The usefulness of MGCCA is shown on simulations and compared to competing methods.

3.1 Introduction

The literature of multi-source data analysis is rather unexplored but has seen renewed interest in the last few years. In the field of supervised methods, the generalized linear model has been extended to handle higher-order tensor [Zhou et al., 2013]. Multiple tasks regression has been adapted to the case where a task matrix is predicted by a higher-order tensor [Fu et al., 2014]. Regression has also been extended to predict a tensor on another tensor [Lock, 2018]. In the field of unsupervised tensor factorization, the Coupled Matrix Tensor Factorization (CMTF) approach has been studied in [Acar et al., 2011, 2013, 2014] and allows to jointly analyze datasets of different orders. Roughly speaking, CMTF can be seen as a multiway extension of simultaneous component analysis [Deun et al., 2009, Kiers and Berge, 1994]. The sources are modeled by fitting jointly PARAFAC models [Carroll and Chang, 1970, Harshman, 1970] to higher-order tensors and matrices. CMTF allows to define which modes (i.e. which dimensions of the tensors) are to be coupled. Another approach, proposed in [Smilde et al., 2000], allows factorizing tensors according to either a PARAFAC or a Tucker model [Tucker, 1963, 1964]. The coupling procedure is restricted to the first mode. More recently, Generalized Structured Component Analysis [Hwang and Takane, 2004] has been extended to the three-way configuration [Choi et al., 2018].

Canonical Correlation Analysis (CCA) [Hotelling, 1936] is one of the earliest model developed to capture relationships between two sets of variables. Several generalizations of CCA to more than two sets of variables have been proposed [Kettenring, 1971, Wold, 1982] and different types of regularizations have been added for more consistent estimations of the CCA parameters in high dimensional settings [Chen et al., 2012a, Leurgans et al., 1993, Vinod, 1976, Witten et al., 2009]. More recently, Regularized Generalized Canonical Correlation Analysis (RGCCA) has been proposed and subsumes many multiblock component methods as special cases (see [Tenenhaus and Tenenhaus, 2011, Tenenhaus et al., 2017] and Chapter 1 for an overview).

To the best of our knowledge, extensions of CCA to situation where at least one of the two sources is a higher-order tensor has been proposed for two blocks only [Kim and Cipolla, 2009, Lu, 2013, Min et al., 2019]. RGCCA is currently geared for the joint analysis of a set of data matrices. In this Chapter, Multiway Generalized Canonical Correlation Analysis (MGCCA) extends RGCCA to higher-order tensors. Preliminary work can be found in [Tenenhaus et al., 2015] where the higher-order structure of the sources is fully considered by adding appropriate Kronecker constraints within the RGCCA optimization problem.

This chapter starts by presenting the sequential MGCCA optimization problem in Section 3.2. The sequential algorithm and its convergence properties are respectively discussed in Sections 3.2.1 and 3.2.2. Section 3.2.3 details two strategies to obtain higher-level components. Section 3.2.4 presents the results of the sequential MGCCA on two simulated datasets. Then the global MGCCA optimization problem is presented in section 3.3 to obtain all the components simultaneously. The algorithm and the convergence properties of this global procedure are also discussed. Finally section 3.3.3 compares the sequential and the global approaches on simulations along with relevant methods.

3.2 The MGCCA optimization problem

For higher-order sources, the RGCCA notations introduced previously need to be extended. As presented in 1.3, we adopt the standardized notations and terminology proposed by [Kiers, 2000]. Let us consider L third-order tensors $\underline{\mathbf{X}}_1, \ldots, \underline{\mathbf{X}}_l, \ldots, \underline{\mathbf{X}}_L$. Each tensor $\underline{\mathbf{X}}_l$ is of dimension $I \times J_l \times K_l$ and represents a set of J_l variables observed at K_l occasions on I individuals. The number of frontal and lateral slices and the nature of the variables can differ from one tensor to another, but the individuals must be the same across tensors. Let $\mathbf{X}^l_{..k}$ be the k^{th} frontal slice of $\underline{\mathbf{X}}_l$ of dimension $I \times J_l$ and $\mathbf{X}^l_{.j.}$ be the j^{th} lateral slice of $\underline{\mathbf{X}}_l$ of dimension $I \times K_l$. Let $\mathbf{X}_l = [\mathbf{X}^l_{..1}, \ldots, \mathbf{X}^l_{..k_l}, \ldots, \mathbf{X}^l_{..K_l}]$ be the first mode matricized version of $\underline{\mathbf{X}}_l$. Each matrix $\mathbf{X}_l$ is of dimension $I \times J_l K_l$ and represents all the frontal slices of $\underline{\mathbf{X}}_l$ next to each other. In this Chapter, the lowercase characters i, j, k, l will be used as running indices respectively for the mode 1, 2, or 3 and for the tensor considered.

Relationships between tensors can be studied using the RGCCA optimization problem (2.1) applied to the matricized tensors $\mathbf{X}_1, \ldots, \mathbf{X}_L$ but the major drawback of this matricized based strategy is that the multiway structure of the data is not preserved. This leads to potentially very large $J_l K_l$ weight vectors to estimate. Moreover, the corresponding positive definite matrices $\mathbf{M}_l$ have prohibitive dimension $J_l K_l \times J_l K_l$. Finally the estimation procedure ignores the original three-way structure of the data both at the level of the weight vectors and at the level of $\mathbf{M}_l$. This may impair the relevance of the results as well as their interpretations. From that perspective, we propose *Multiway Generalized Canonical Correlation Analysis* (MGCCA) that specifically addresses the higher-order structure of the sources.

In order to consider the higher-order structure of some sources, the RGCCA optimization problem (2.1) is reformulated by incorporating Kronecker constraints, intensively used in the multiway literature [Bro, 1996, Kolda and Bader, 2009, Zhou et al., 2013]. The first stage of sequential MGCCA is

defined as the following optimization problem:

$$\begin{aligned} &\max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) \\ &\text{s.t. } \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l = 1 \text{ and } \mathbf{w}_l = \mathbf{w}_l^K \otimes \mathbf{w}_l^J, l = 1,\ldots,L. \end{aligned} \tag{3.1}$$

The weight vectors $\mathbf{w}_l, l = 1,\ldots,L$ are modeled as the Kronecker product between a weight vector $\mathbf{w}_l^K$ associated with the K_l frontal slices and a weight vector $\mathbf{w}_l^J$ associated with the J_l lateral slices: $\mathbf{w}_l = \mathbf{w}_l^K \otimes \mathbf{w}_l^J, l = 1,\ldots L$. The reformulation proposed in (3.1) using the Kronecker constraints applied to the weight vectors enables to rewrite the components as follows:

$$\mathbf{y}_l = \mathbf{X}_l\mathbf{w}_l = \mathbf{X}_l(\mathbf{w}_l^K \otimes \mathbf{w}_l^J) = \quad \mathbf{X}_l(\mathbf{I}_{K_l} \otimes \mathbf{w}_l^J)\mathbf{w}_l^K = \left(\sum_{j=1}^{J_l} w_{lj}^J \mathbf{X}_{.j.}^l\right)\mathbf{w}_l^K \tag{3.2}$$

From equation (3.2), it appears that the component $\mathbf{y}_l$ can be expressed as a linear combination of the columns of the matrix $\sum_{j=1}^{J_l} w_{lj}^J \mathbf{X}_{.j.}^l$ defined as a weighted mean of the lateral slices. In the same way, $\mathbf{y}_l$ can be expressed as a linear combination of the columns of $\sum_{k=1}^{K_l} w_{lk}^K \mathbf{X}_{..k}^l$ defined as a weighted mean of the frontal slices.

In addition, a Kronecker structure may also be imposed for $\mathbf{M}_l = \mathbf{M}_l^K \otimes \mathbf{M}_l^J$ where $\mathbf{M}_l^K$ and $\mathbf{M}_l^J$ are two positive definite matrices of dimensions $K_l \times K_l$ and $J_l \times J_l$, respectively. Then, the optimization problem (3.1) can be simplified by considering the two following transforms $\mathbf{P}_l = I^{-1/2}\mathbf{X}_l\mathbf{M}_l^{-1/2}$ and $\mathbf{v}_l = \mathbf{M}_l^{1/2}\mathbf{w}_l$, which can be re-written as:

$$\mathbf{v}_l = \mathbf{M}_l^{1/2}\mathbf{w}_l = (\mathbf{M}_l^K)^{1/2}\mathbf{w}_l^K \otimes (\mathbf{M}_l^J)^{1/2}\mathbf{w}_l^J = \mathbf{v}_l^K \otimes \mathbf{v}_l^J \tag{3.3}$$

Finally, the optimization problem (3.1) becomes:

$$\max_{\mathbf{v}_1,\ldots,\mathbf{v}_L} f(\mathbf{v}_1,\ldots,\mathbf{v}_L) = \sum_{k,l=1}^{L} c_{lk}\ \mathrm{g}\left(\mathbf{v}_k^\top \mathbf{P}_k^\top \mathbf{P}_l \mathbf{v}_l\right) \tag{3.4}$$

$$\text{s.t. } \mathbf{v}_l^\top \mathbf{v}_l = 1 \text{ and } \mathbf{v}_l = \mathbf{v}_l^K \otimes \mathbf{v}_l^J, l = 1,\ldots,L \quad . \tag{3.5}$$

These Kronecker constraints yield a more parsimonious model. Indeed, for each $l \in \{1,\ldots,L\}$, $J_l + K_l$ parameters have to be estimated instead of $J_l \times K_l$, which can be tremendously higher. Besides, distinct weight vectors $\mathbf{w}_l^K$ and $\mathbf{w}_l^J$ are estimated, which enables to interpret the effects of each mode separately. Furthermore, the Kronecker structure for $\mathbf{M}_l$ allows to better fit the three-way structure of the data and reduces the computational burden of the MGCCA algorithm. The sole requirement on $\mathbf{M}_l^K$ and $\mathbf{M}_l^J$ is the positive definiteness. The optimization problem (3.1) allows to recover well known multiway methods such as PARAFAC (first component) [Carroll and Chang, 1970, Harshman, 1970] and N-way Partial Least Squares (NPLS) [Bro, 1996] (see the appendix A sections A.1 and A.2 for more details).

3.2.1 The MGCCA Algorithm

The MGCCA and RGCCA criteria only differ at the level of the constraints. Therefore, the general optimization framework presented in section 1.5 still applies for MGCCA. The update defined in

equation (2.4) for RGCCA is extended for MGCCA. This update finds $\hat{\mathbf{v}}_l \in \Omega_l$ by considering the following optimization problem:

$$\hat{\mathbf{v}}_l = \underset{\tilde{\mathbf{v}}_l \in \Omega_l}{\operatorname{argmax}} \ \nabla_l f(\mathbf{v})^\top \tilde{\mathbf{v}}_l = \underset{\tilde{\mathbf{v}}_l \in \Omega_l}{\operatorname{argmax}} \ \mathbf{z}_l^\top \mathbf{P}_l \tilde{\mathbf{v}}_l := r_l(\mathbf{v}), \tag{3.6}$$

where $\mathbf{v} = (\mathbf{v}_1; \ldots; \mathbf{v}_L)$, $\Omega_l = \left\{ \mathbf{v}_l \in \mathbb{R}^{K_l J_l};\ \mathbf{v}_l^\top \mathbf{v}_l = 1 \text{ and } \mathbf{v}_l = \mathbf{v}_l^K \otimes \mathbf{v}_l^J \right\}$ and $\nabla_l f(\mathbf{v})$ is the partial gradient of f with respect to $\mathbf{v}_l$ that can be found in equation (2.5) and $\mathbf{z}_l$ in the inner component first derived in equation (2.5) also.

The optimization problem (3.6) boils down to finding a pair of weight vectors $\mathbf{v}_l^K$ and $\mathbf{v}_l^J$ that produces a component $\mathbf{y}_l = \mathbf{P}_l \mathbf{v}_l$ with maximal scalar product with $\mathbf{z}_l$. The problem is equivalent to:

$$\begin{aligned}
\left(\mathbf{v}_l^K, \mathbf{v}_l^J\right) &= \underset{\substack{\mathbf{v}_l^K, \mathbf{v}_l^J \\ \|\mathbf{v}_l^K \otimes \mathbf{v}_l^J\|=1}}{\operatorname{argmax}} \ \mathbf{z}_l{}^\top \mathbf{P}_l (\mathbf{v}_l^K \otimes \mathbf{v}_l^J) = \underset{\substack{\mathbf{v}_l^K, \mathbf{v}_l^J \\ \|\mathbf{v}_l^K \otimes \mathbf{v}_l^J\|=1}}{\operatorname{argmax}} \ \mathbf{z}_l{}^\top \left[\sum_{k=1}^{K} v_{lk}^K \mathbf{P}_{..k}^l \right] \mathbf{v}_l^J \\
&= \underset{\substack{\mathbf{v}_l^K, \mathbf{v}_l^J \\ \|\mathbf{v}_l^K \otimes \mathbf{v}_l^J\|=1}}{\operatorname{argmax}} \ \left[\sum_{k=1}^{K} v_{lk}^K \mathbf{z}_l{}^\top \mathbf{P}_{..k}^l \right] \mathbf{v}_l^J = \underset{\substack{\mathbf{v}_l^K, \mathbf{v}_l^J \\ \|\mathbf{v}_l^K \otimes \mathbf{v}_l^J\|=1}}{\operatorname{argmax}} \ \mathbf{v}_l^K{}^\top \mathbf{Q}_l \mathbf{v}_l^J
\end{aligned} \tag{3.7}$$

where $\mathbf{Q}_l$ is a $K_l \times J_l$ matrix defined by $\mathbf{Q}_l = [(\mathbf{P}_{..1}^l)^\top \mathbf{z}_l, \ldots, (\mathbf{P}_{..K_l}^l)^\top \mathbf{z}_l]^\top$.
We deduce that $\mathbf{v}_l^K$ and $\mathbf{v}_l^J$, solution of the optimization problem (3.7), are the first left and right singular vectors of the matrix $\mathbf{Q}_l$ of dimension $K_l \times J_l$. The singular vectors $\mathbf{v}_l^K$ and $\mathbf{v}_l^J$ are unit-norm, thus satisfying the unit-norm constraint on $\mathbf{v}_l$. Note that a similar optimization procedure is found for NPLS [Bro, 1996]. The entire MGCCA algorithm is described in Algorithm 4.

Algorithm 4 Multiway Generalized Canonical Correlation Analysis (MGCCA) algorithm

1: **Data:** $\underline{\mathbf{X}}_1, \ldots, \underline{\mathbf{X}}_L$, $\mathbf{M}_1, \ldots, \mathbf{M}_L$, g, ε, $\mathbf{C}$
2: **Result:** $\mathbf{v}_1^s, \ldots, \mathbf{v}_L^s$ (solution of (3.4) subject to (3.5))
3: **Initialization:** $\mathbf{v}_l^0 = \mathbf{v}_l^{K,0} \otimes \mathbf{v}_l^{J,0}$, $l = 1, \ldots, L$, where $\mathbf{v}_l^{J,0}$, $\mathbf{v}_l^{K,0}$ are random unit-norm vectors, $s = 0$;
4: **repeat**
5: **for** $l = 1$ **to** L **do**

6: $$\mathbf{v}_l^{s+1} = r_l\left(\mathbf{v}_1^{s+1}, ..., \mathbf{v}_{l-1}^{s+1}, \mathbf{v}_l^s, \mathbf{v}_{l+1}^s, ..., \mathbf{v}_L^s\right) = (\mathbf{v}_l^K)^{s+1} \otimes (\mathbf{v}_l^J)^{s+1} \tag{3.8}$$

7: **end for**
8: $s = s + 1$;
9: **until** $f(\mathbf{v}_1^{s+1}, \ldots, \mathbf{v}_L^{s+1}) - f(\mathbf{v}_1^s, \ldots, \mathbf{v}_L^s) < \varepsilon$

where $(\mathbf{v}_l^K)^{s+1}$ and $(\mathbf{v}_l^J)^{s+1}$ are obtained as the first left and right singular vectors of the matrix $\mathbf{Q}_l = [(\mathbf{P}_{..1}^l)^\top \mathbf{z}_l^s, \ldots, (\mathbf{P}_{..K_l}^l)^\top \mathbf{z}_l^s]^\top$ of dimension $K_l \times J_l$ and
$\mathbf{z}_l^s = 2 \sum_{k=1}^{l-1} c_{lk}\ \mathrm{g}'(\mathbf{v}_l^{s\top} \mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k^{s+1}) \mathbf{P}_k \mathbf{v}_k^{s+1} + 2 \sum_{k=l}^{L} c_{lk}\ \mathrm{g}'(\mathbf{v}_l^{s\top} \mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k^s) \mathbf{P}_k \mathbf{v}_k^s$.

From equation (3.3), the original weight vectors $\mathbf{w}_l^K$ and $\mathbf{w}_l^J$ are recovered by $\mathbf{w}_l^K = (\mathbf{M}_l^K)^{-1/2} \mathbf{v}_l^K$ and $\mathbf{w}_l^J = (\mathbf{M}_l^J)^{-1/2} \mathbf{v}_l^J$.

3.2.2 Convergence properties of the MGCCA algorithm

In order to show that Proposition 1.5.1 holds for the MGCCA algorithm, point (*i-iv*) of Lemma 1.5.2 are verified below.

Proof of Lemma 1.5.2 for the MGCCA Algorithm.

Point *(i)* Let $\Omega_l^K = \left\{\mathbf{v}_l^K \in \mathbb{R}^{K_l}; \|\mathbf{v}_l^K\|_2 = 1\right\}$ and $\Omega_l^J = \left\{\mathbf{v}_l^J \in \mathbb{R}^{J_l}; \|\mathbf{v}_l^J\|_2 = 1\right\}$ be two compact sets and the continuous multilinear function f_l be defined as:

$$\begin{array}{rcl} f_l : \Omega_l^K \times \Omega_l^J & \rightarrow & \Omega_l \\ (\mathbf{v}_l^K, \mathbf{v}_l^J) & \mapsto & \mathbf{v}_l^K \otimes \mathbf{v}_l^J. \end{array}$$

Ω_l is compact as image of a compact set by the continuous function f_l. Consequently, Ω is compact as product of L compact sets.

Point *(ii)* As long as the solution of optimization problem (3.7) exists and is unique, the demonstration of the point (*ii*) of the Lemma 1.5.2 made in Chapter 1 still holds. For MGCCA, $r_l(\mathbf{v})$ is obtained as the Kronecker product between the first left and right singular vectors of the matrix $\mathbf{Q}_l = [(\mathbf{P}^l_{..1})^\top \mathbf{z}_l, \ldots, (\mathbf{P}^l_{..K_l})^\top \mathbf{z}_l]^\top$ of dimension $K_l \times J_l$. The first singular vectors of a matrix are defined up to their sign and up to a multiplicative constant. The sign indeterminacy is controlled just after the first iteration of the algorithm and the MGCCA algorithm guarantees normalized weight vectors. The uniqueness of the dominant singular value can easily be verified numerically at each iteration s. This procedure also guarantees that $(\mathbf{v}_l^K, \mathbf{v}_l^J)$, and so $(\mathbf{w}_l^K, \mathbf{w}_l^J)$ are identifiable. In section 3.2.4.1, on a simulated dataset, we monitor the Karush–Kuhn–Tucker (KKT) optimality conditions in order to assess numerically that this last assumption is not too constraining.

Point *(iii)* The demonstration presented in Chapter 1 for point *(iii)* of Lemma 1.5.2 still holds here.

Point *(iv)* The proof is based on the uniqueness of $r_l(\mathbf{v})$ defined in equation (3.7). Under mild conditions (see the discussion above), this point is satisfied for MGCCA. □

3.2.3 Higher-level components

At the end of Algorithm 4, the first-level weight vectors $\mathbf{w}_l^{(1)}, l = 1, \ldots, L$ solutions of the optimization problem (3.1) are obtained. Two strategies to determine higher-level weight vectors are presented. The first one yields orthogonal components and the second one yields orthogonal weight vectors.

3.2.3.1 Deflation procedure for orthogonal components

Deflation is the most straightforward way to add orthogonality constraints in many optimization problems encountered in multivariate analysis. This deflation procedure consists in replacing the data matrix $\mathbf{X}_l$ by its residual matrix $\mathbf{X}_l^{(1)}$ obtained by the regression of $\mathbf{X}_l$ on $\mathbf{y}_l^{(1)}$: $\mathbf{X}_l^{(1)} = \mathbf{X}_l - \mathbf{y}_l^{(1)} \left((\mathbf{y}_l^{(1)})^\top \mathbf{y}_l^{(1)}\right)^{-1} (\mathbf{y}_l^{(1)})^\top \mathbf{X}_l$. If $\mathbf{M}_l$ depends on $\mathbf{X}_l$, then $\mathbf{X}_l$ is replaced by its residual $\mathbf{X}_l^{(1)}$ in its

current calculation, otherwise $\mathbf{M}_l$ is left unchanged. In both cases, this matrix is defined as $\mathbf{M}_l^{(1)}$. The second-level MGCCA optimization problem is:

$$\begin{aligned} &\max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top (\mathbf{X}_k^{(1)})^\top \mathbf{X}_l^{(1)}\mathbf{w}_l\right) \\ &\text{s.t. } \mathbf{w}_l^\top \mathbf{M}_l^{(1)}\mathbf{w}_l = 1 \text{ and } \mathbf{w}_l = \mathbf{w}_l^K \otimes \mathbf{w}_l^J, l = 1,\ldots,L. \end{aligned} \tag{3.9}$$

For each $l = 1,\ldots,L$, the resulting component $\mathbf{y}_l^{(2)} = \mathbf{X}_l^{(1)}\mathbf{w}_l^{(2)}$ is orthogonal to $\mathbf{y}_l^{(1)}$. Components can thus be displayed in a Cartesian coordinate system. This may be interesting for generating a correlation circle per block in order to see the position of the variables in the components space and to understand which variables contribute to which components.

3.2.3.2 Orthogonality of the weight vectors

For each $l = 1,\ldots,L$, the second-level weight vector $\mathbf{w}_l^{(2)}$ is constrained to be orthogonal to the first-level weight vector $\mathbf{w}_l^{(1)}$. This novel constraint is written as:

$$\mathbf{w}_l^\top \mathbf{w}_l^{(1)} = 0 \Leftrightarrow \left\{ \mathbf{w}_l^{K^\top} \mathbf{w}_l^{K,(1)} = 0 \text{ or } \mathbf{w}_l^{J^\top} \mathbf{w}_l^{J,(1)} = 0 \right\}. \tag{3.10}$$

Therefore, the orthogonality can either be imposed on $\mathbf{w}_l^K$ or on $\mathbf{w}_l^J$. Hereafter, we discuss the case where orthogonality is imposed at the level of $\mathbf{w}_l^J$. The other case is recovered similarly.

Assume that for a given $\underline{\mathbf{X}}_l$, the r first weight vectors $\mathbf{w}_l^{J,(1)},\ldots,\mathbf{w}_l^{J,(r)}$ have already been computed and denote $\mathbf{W}_l^{J,(r)} = \left[\mathbf{w}_l^{J,(1)},\ldots,\mathbf{w}_l^{J,(r)}\right]$ the matrix that contains these weight vectors columnwise. Let $\mathbf{R}^{\perp}_{\mathbf{W}_l^{J,(r)}} = \mathbf{I}_{J_l\times J_l} - \mathbf{W}_l^{J,(r)}\left(\mathbf{W}_l^{J,(r)^\top}\mathbf{W}_l^{J,(r)}\right)^{-1}\mathbf{W}_l^{J,(r)^\top}$ be the projection matrix onto the orthogonal subspace defined by $\mathbf{W}_l^{J,(r)}$. Solving optimization problem (3.1) with the additional constraint that $\mathbf{W}_l^{J,(r)^\top}\mathbf{w}_l^J = \mathbf{0}$ is similar to say that there exists $\tilde{\mathbf{x}}_l^J \in \mathbb{R}^{J_l}$ such that $\mathbf{w}_l^J = \mathbf{R}^{\perp}_{\mathbf{W}_l^{J,(r)}}\tilde{\mathbf{x}}_l^J$.

The derivation of $\mathbf{w}_l$ is not straightforward since $\mathbf{R}^{\perp}_{\mathbf{W}_l^{J,(r)}}$ is of rank $J_l - r$, which implies that $\tilde{\mathbf{x}}_l^J$ is not unique. Nevertheless, $\mathbf{R}^{\perp}_{\mathbf{W}_l^{J,(r)}}$ is real and symmetric and can be decomposed as $\mathbf{U}_{\mathbf{W}_l^{J,(r)}}\mathbf{D}_{\mathbf{W}_l^{J,(r)}}\mathbf{U}^\top_{\mathbf{W}_l^{J,(r)}}$ with $\mathbf{U}_{\mathbf{W}_l^{J,(r)}}$ the $J_l \times (J_l - r)$ orthonormal matrix of eigenvectors and $\mathbf{D}_{\mathbf{W}_l^{J,(r)}}$ the $(J_l - r)\times(J_l - r)$ diagonal matrix whose elements are the non-null eigenvalues. Thus, there exists a unique $\mathbf{x}_l^J \in \mathbb{R}^{J_l - r}$ such that $\mathbf{w}_l^J = \mathbf{U}_{\mathbf{W}_l^{J,(r)}}\mathbf{x}_l^J$ and therefore, $\mathbf{w}_l$ can be written as:

$$\mathbf{w}_l \;=\; \mathbf{w}_l^K \otimes \mathbf{U}_{\mathbf{W}_l^{J,(r)}}\mathbf{x}_l^J = \left(\mathbf{I}_{K_l} \otimes \mathbf{U}_{\mathbf{W}_l^{J,(r)}}\right)\left(\mathbf{w}_l^K \otimes \mathbf{x}_l^J\right).$$

Therefore, from the two following equations

$$\begin{aligned} \mathbf{X}_l^{(r)} &= \mathbf{X}_l\left(\mathbf{I}_{K_l} \otimes \mathbf{U}_{\mathbf{W}_l^{J,(r)}}\right) \\ \mathbf{M}_l^{(r)} &= \left(\mathbf{I}_{K_l} \otimes \mathbf{U}^\top_{\mathbf{W}_l^{J,(r)}}\right)\left(\mathbf{M}_l^K \otimes \mathbf{M}_l^J\right)\left(\mathbf{I}_{K_l} \otimes \mathbf{U}_{\mathbf{W}_l^{J,(r)}}\right) = \mathbf{M}_l^K \otimes \left(\mathbf{U}^\top_{\mathbf{W}_l^{J,(r)}}\mathbf{M}_l^J\mathbf{U}_{\mathbf{W}_l^{J,(r)}}\right), \end{aligned}$$

the $(r+1)^{th}$ level MGCCA optimization problem is defined as:

$$\begin{aligned} &\max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top (\mathbf{X}_k^{(r)})^\top \mathbf{X}_l^{(r)}\mathbf{w}_l\right) \\ &\text{s.t. } \mathbf{w}_l^\top \mathbf{M}_l^{(r)}\mathbf{w}_l = 1 \text{ and } \mathbf{w}_l = \mathbf{w}_l^K \otimes \mathbf{x}_l^J, l = 1,\ldots,L, \end{aligned} \tag{3.11}$$

and the orthogonality of the weight vectors $\mathbf{w}_l^{J,(1)}, \ldots, \mathbf{w}_l^{J,(r+1)}$ is guaranteed. The optimization problem (3.11) gives for $l = 1, \ldots, L$, $\mathbf{w}_l^{K,(r+1)}$ and $\mathbf{x}_l^{J,(r+1)}$ and $\mathbf{w}_l^{J,(r+1)} = \mathbf{U}_{\mathbf{W}_l^{J,(r)}} \mathbf{x}_l^{J,(r+1)}$.

This procedure implies the SVD of $\mathbf{R}^{\perp}_{\mathbf{W}_l^{J,(r)}}$ or $\mathbf{R}^{\perp}_{\mathbf{W}_l^{K,(r)}}$, which is computationally advantageous when performed on the mode with the smallest dimension. Imposing orthogonality at the level of the weight vectors is interesting for evaluating the relative position of the individuals (e.g classification, clustering) in orthonormal bases defined in the variables space.

Similar deflation is proposed for multiway discriminant analysis in [Lechuga et al., 2016]. Moreover, in the framework of NPLS [Bro, 1996], a sequential approach to compute the subsequent components and weight vectors is also proposed in [Hanafi et al., 2015] where no orthogonality conditions are considered. Our deflation procedure both imposes orthogonality at the level of the weight vectors and respects the multi-way structure of the data.

3.2.4 Experiments

In this section, the performances of MGCCA are studied on simulations. The first simulation (section 3.2.4.1) evaluates to what extent the hypothesis made in the proof of the global convergence of Algorithm 4 (see section 3.2.2) is not too binding. The second simulation (section 3.2.4.2) compares the performances of MGCCA, RGCCA, Couple Matrix Tensor Factorization (CMTF) and PARAFAC.

3.2.4.1 Verification of the KKT optimality conditions

As mentioned in section 3.2.2, global convergence of Algorithm 4 relies on a uniqueness assumption. The objective of this first simulation is to check numerically that this assumption is not too constraining by monitoring the KKT optimality conditions. For that purpose, three tensors $\underline{\mathbf{X}}_l \in \mathbb{R}^{I \times J_l \times K_l}$, $l = 1, 2, 3$, following a $(R+1)$-component PARAFAC model are generated (cf. section 1.4.1):

$$\underline{\mathbf{X}}_l = \sum_{r=1}^{R+1} \mathbf{a}_l^r \circ \mathbf{b}_l^r \circ \mathbf{c}_l^r + \underline{\mathbf{E}}_l, \tag{3.12}$$

where $\circ$ stands for the outer product (see Figure 1.4-3 for more details). Dimension of each order are $I = 90$, $J_1 = 200, J_2 = 500, J_3 = 10^3$ and $K_1 = 5, K_2 = K_3 = 10$. For the first component and the first mode, every row of $\mathbf{A} = [\mathbf{a}_1^1, \mathbf{a}_2^1, \mathbf{a}_3^1]$ is independently drawn from a multivariate normal distribution with 0 mean and predefined correlation structure $\mathbf{\Sigma} = (\sigma_{lk})$, with $\sigma_{12} = \sigma_{13} = \sigma_{23} = 0.7$, 1 on the diagonal and 0 otherwise. All coordinates of all other generative vectors are drawn from a uniform distribution $\mathcal{U}[0, 1]$. Finally, $\underline{\mathbf{E}}_l = (e_{ijk})$ is a residual tensor with $e_{ijk} \sim \mathcal{N}(0, 4)$.

Let f be the objective function of the optimization problem (3.4) and $h_l = (\mathbf{v}_l^K \otimes \mathbf{v}_l^J)^\top (\mathbf{v}_l^K \otimes \mathbf{v}_l^J) - 1$ the constraint function associated with block l. Let $\mathbf{v} = \left(\mathbf{v}_1^K; \mathbf{v}_1^J; \ldots; \mathbf{v}_L^K; \mathbf{v}_L^J\right)$ and denote by $\nabla_{\mathbf{v}}\mathbf{h} = (\nabla_{\mathbf{v}} h_1, \ldots, \nabla_{\mathbf{v}} h_L)$ the matrix of partial gradient of each constraint with respect to $\mathbf{v}$. Let $\mathbf{\Pi} = \nabla_{\mathbf{v}}\mathbf{h} \left(\nabla_{\mathbf{v}}\mathbf{h}^\top \nabla_{\mathbf{v}}\mathbf{h}\right)^{-1} \nabla_{\mathbf{v}}\mathbf{h}^\top$ be the projection matrix on the subspace generated by the columns of $\nabla_{\mathbf{v}}\mathbf{h}$. As mentioned in the preamble of section 1.5.2, for a stationary point, the derivative of the objective function lies in the subspace defined by the derivative of each constraint. Thus, if the solution of Algorithm 4 is a stationary point then at convergence, the KKT optimality condition can be formulated as $(\mathbf{I} - \mathbf{\Pi}) \nabla_{\mathbf{v}} f = \mathbf{0}$, where vector $\nabla_{\mathbf{v}} f = \left(\nabla_{\mathbf{v}_1^K} f; \nabla_{\mathbf{v}_1^J} f; \ldots; \nabla_{\mathbf{v}_L^K} f; \nabla_{\mathbf{v}_L^J} f\right)$. For the

Table 3.1 – For each $R \in \{0, 1, 2, 5, 10\}$, this table reports the first iteration at which the value $KKT = \|(\mathbf{I} - \mathbf{\Pi})\nabla_{\mathbf{v}} f\|/\|\nabla_{\mathbf{v}} f\|$ is below a specific threshold (8 different thresholds considered) for all the 200 runs.

KKT	$\leq 10^{-1}$	$\leq 10^{-3}$	$\leq 10^{-5}$	$\leq 10^{-7}$	$\leq 10^{-9}$	$\leq 10^{-11}$	$\leq 10^{-13}$	$\leq 10^{-14}$
$R = 0$	8	11	15	18	22	25	29	31
$R = 1$	5	8	11	15	18	21	24	25
$R = 2$	6	10	13	17	20	24	27	29
$R = 5$	7	11	14	18	22	26	29	31
$R = 10$	10	18	25	33	41	48	56	60

sake of completeness, we mention that the partial gradients of f and h_l with respect to $\mathbf{v}_l^K$ and $\mathbf{v}_l^J$ are equal to $\nabla_{\mathbf{v}_l^K} f = \mathbf{Q}_l \mathbf{v}_l^J$, $\nabla_{\mathbf{v}_l^J} f = \mathbf{Q}_l^\top \mathbf{v}_l^K$, $\nabla_{\mathbf{v}_l^K} h_l = 2\|\mathbf{v}_l^J\|_2^2 \mathbf{v}_l^K$ and $\nabla_{\mathbf{v}_l^J} h_l = 2\|\mathbf{v}_l^K\|_2^2 \mathbf{v}_l^J$.

MGCCA was run 200 times with random initial weights on this simulated dataset to extract one component with $g(x) = x^2$ (a.k.a. factorial scheme), $c_{ij} = 1$, if $i \neq j$ (a.k.a. complete design), and $\mathbf{M}_l^K, \mathbf{M}_l^J$, $l = 1, \ldots, 3$ defined as identity matrices. For each run and for each iteration s of the MGCCA algorithm, the quantity $KKT = \|(\mathbf{I} - \mathbf{\Pi})\nabla_{\mathbf{v}} f\|/\|\nabla_{\mathbf{v}} f\|$ is computed. Experiment is repeated for $R \in \{0, 1, 2, 5, 10\}$. Table 3.1 reports the number of iterations s at which all the 200 runs are under a specific value of KKT for the different values of R. It appears that the KKT conditions are always satisfied in less than 60 iterations, meaning that whatever the initialization, the MGCCA algorithm converges to a stationary point.

3.2.4.2 Recovering interactions between tensors

This section aims at evaluating the ability of MGCCA to identify variables of each tensor responsible for the link between them. This time, $L = 2$ tensors are generated with the same dimension $J_1 = J_2 = K_1 = K_2 = 30$. In this experiment, rather than using a PARAFAC model, $I = 50$ samples are drawn from a centered multivariate normal distribution with covariance matrix $\mathbf{\Sigma}$ defined as:

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{bmatrix} + \begin{bmatrix} \mathbf{N}_{11} & 0 \\ 0 & \mathbf{N}_{22} \end{bmatrix} = \mathbf{S} + \mathbf{N} \tag{3.13}$$

where $\mathbf{\Sigma}_{lh} = \mathbf{W}_l \mathbf{\Delta} \mathbf{W}_h^\top$, for $l, h = 1, 2$, with $\mathbf{W}_l = \left[\mathbf{w}_l^{K,(1)} \otimes \mathbf{w}_l^{J,(1)}, \mathbf{w}_l^{K,(2)} \otimes \mathbf{w}_l^{J,(2)}\right]$ is a 900×2 matrix where $\mathbf{w}_l^{K,(r)}$, $\mathbf{w}_l^{J,(r)} \in \mathbb{R}^{30\times 1}$ for $l, r = 1, 2$. In this context, $\mathbf{S}$ is indeed positive definite and every $\mathbf{\Sigma}_{lh}$, $l, h = 1, 2$ is a square matrix of size 900. The 2×2 diagonal matrix $\mathbf{\Delta}$ with diagonal elements $\delta_1 = 0.7$ and $\delta_2 = 0.3$ is introduced to control the relative contributions of the two dimensions. Moreover, we add two positive definite noise matrices of size 900 defined as $\mathbf{N}_{ll} = \mathbf{U}_l \mathbf{\Lambda}_l \mathbf{U}_l^\top$, where $\mathbf{U}_l = \left[\mathbf{u}_l^{K,(1)} \otimes \mathbf{u}_l^{J,(1)}, \ldots, \mathbf{u}_l^{K,(30)} \otimes \mathbf{u}_l^{J,(30)}\right]$ is a random orthonormal matrix of size 900×30 with $\mathbf{u}_l^{K,(r)}$, $\mathbf{u}_l^{J,(r)} \in \mathbb{R}^{30\times 1}$ for $l = 1, 2$, $r \in [\![1, 30]\!]$ and $\mathbf{\Lambda}_l \in \mathbb{R}^{30\times 30}$ is a diagonal matrix whose diagonal elements are generated from $\mathcal{U}([10^{-10}; 1])$. Finally, the Signal to Noise Ratio is defined as $\text{SNR} = 20 \log_{10}(\|\mathbf{S}\|_F/\|\mathbf{N}\|_F)$ where $\|.\|_F$ stands for the Frobenius norm. The main objective of this experiment is to recover the interaction terms $\mathbf{w}_l^{J,(r)}$, $\mathbf{w}_l^{K,(r)}$, $l, r = 1, 2$. The noise matrix $\mathbf{N}$ was designed to represent variance terms of each of the two tensors separately. Thus in a low SNR case, PARAFAC can mostly identify variance terms of the noise matrix while CMTF and MGCCA are able to extract interaction terms hidden among variance terms.

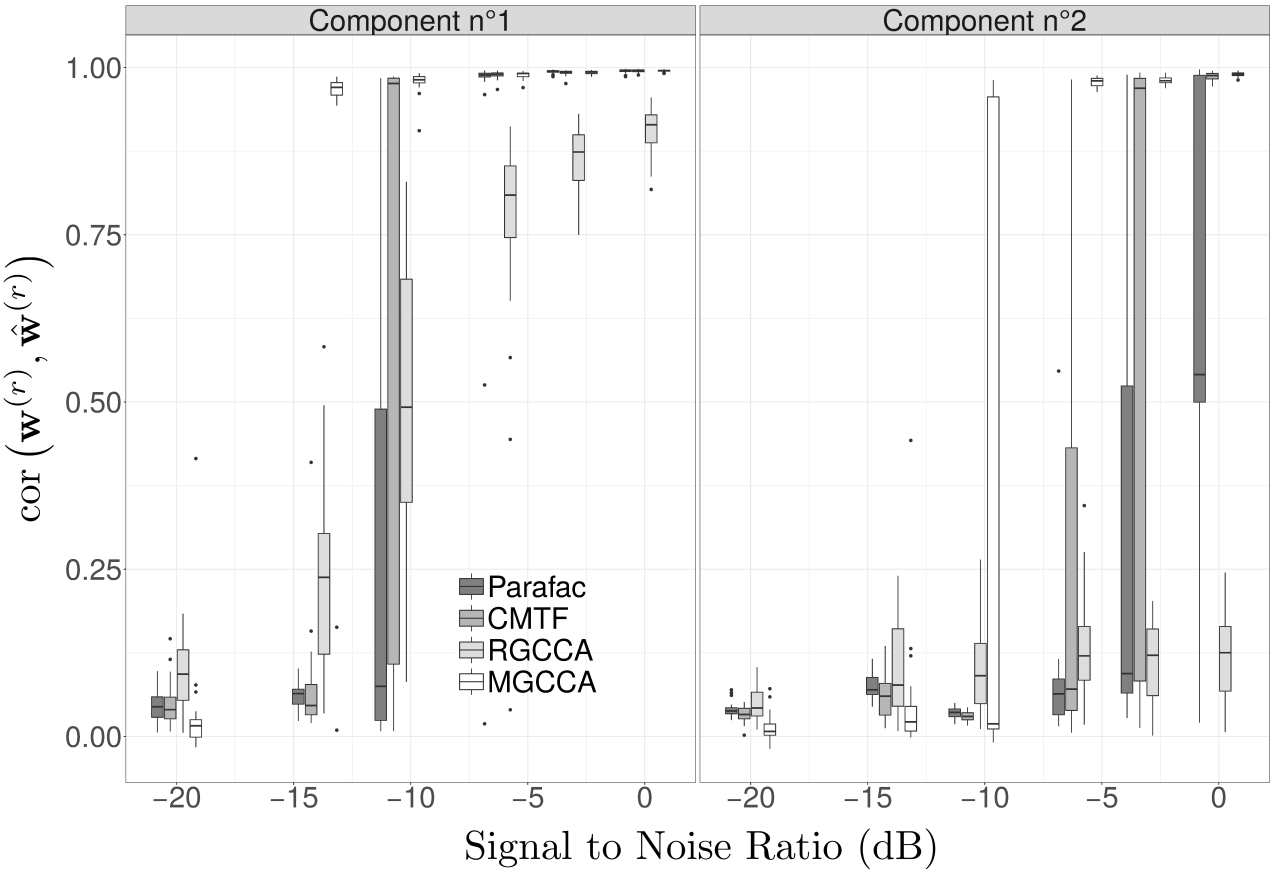


Figure 3.2-1 – Boxplots of cor $\left(\mathbf{w}^{(1)}, \hat{\mathbf{w}}^{(1)}\right)$ (left) and cor $\left(\mathbf{w}^{(2)}, \hat{\mathbf{w}}^{(2)}\right)$ (right) for PARAFAC, CMTF, MGCCA and RGCCA. 6 different levels of Signal to Noise Ratio, SNR $= 20 \log_{10}(\|\mathbf{S}\|_F / \|\mathbf{N}\|_F)$, ranging from -20 to 0 dB were evaluated.

MGCCA, RGCCA on the matricized tensors, CMTF and PARAFAC on each tensor separately, were evaluated and compared on these simulated datasets. The positive definite matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ that appear in the constraints of RGCCA and MGCCA optimization problems (see equation (2.1) and (3.1)) are set to the identity, $g(x) = x^2$ and $c_{12} = 1$. The R package `RGCCA` [Tenenhaus and Guillemot, 2017] was used for RGCCA and MGCCA. The MATLAB CMTF toolbox (v1.1) (http://www.models.life.ku.dk/joda/CMTF_Toolbox) was used for CMTF. The R package `multiway` [Helwig, 2018] was used for PARAFAC. The tolerance of the stopping criteria ε is set to 10^{-8} for each algorithm. As each method presents potentially many local minima/maxima, multiple starts were performed (i.e., SVD-based initialization as well as 100 random starts) and the best solution was kept [Acar et al., 2013, ten Berge, 1993]. For a SNR ranging from -20dB to 0dB, we generated 100 datasets according to the simulation protocol described above. MGCCA, RGCCA, CMTF and PARAFAC were applied and the efficiency of each method was measured, component by component, by considering, for each of the 100 datasets, the correlation between the estimate $\hat{\mathbf{w}}^{(r)} = \left(\hat{\mathbf{w}}_1^{(r)}; \hat{\mathbf{w}}_2^{(r)}\right)$ (where $\hat{\mathbf{w}}_l^{(r)} = \hat{\mathbf{w}}_l^{K,(r)} \otimes \hat{\mathbf{w}}_l^{J,(r)}$) and the true vector $\mathbf{w}^{(r)}$, for $r = 1, 2$. Boxplots of these correlations are reported in Figure 3.2-1. For RGCCA and MGCCA, only two components were estimated. For

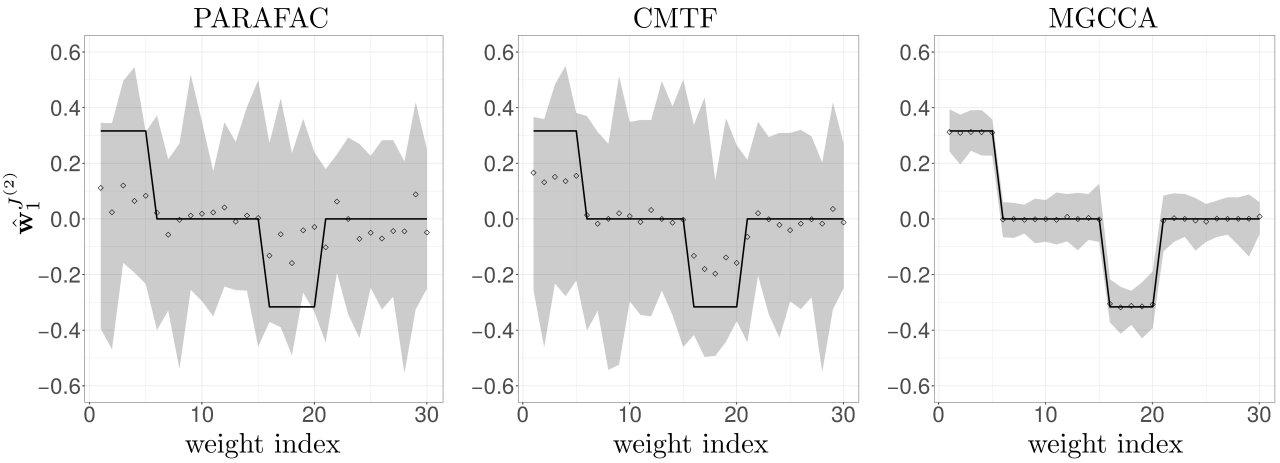


Figure 3.2-2 – For $SNR = -6$dB, estimated vectors $\hat{\mathbf{w}}_1^{J,(2)}$ is represented for the three multiway methods. Dots stand for the mean of the estimated vector. Black lines graph the true weights. 100 datasets were generated for this SNR. 10 worst runs for each method according to cor $\left(\mathbf{w}^{(2)}, \hat{\mathbf{w}}^{(2)}\right)$ were removed. For each element of this estimated vector, grey areas stand for the *min* and *max* of its distribution based on the 90 remaining runs. All vectors were normalized to unit norm.

MGCCA, the two procedures described in section 3.2.3 were evaluated and yield similar results. Results associated with orthogonal weight vectors are reported. For CMTF and PARAFAC, four different models were fitted with each time a different number of components (from 2 to 5). For both CMTF and PARAFAC, the couple of weight vectors maximally correlated with the truth are obtained with the model associated with 5 components. The corresponding correlations with the truth are reported. All the multiway methods recover the truth above a specific level of SNR. It appears that MGCCA recovers the truth at a level of SNR significantly lower that the other methods. RGCCA presents significantly lower results than MGCCA justifying the interest to integrate Kronecker constraints.

Figure 3.2-2 depicts the weight vectors $\hat{\mathbf{w}}_1^{J,(2)}$ estimated with PARAFAC, CMTF and MGCCA with $SNR = -6$dB. The non multiway nature of RGCCA precludes to report the RGCCA estimates. As expected, for this particular level of SNR, it is clear that MGCCA leads to less biased and more robust estimates. The overall execution time across SNR values and simulated datasets is $0.04 \pm 0.01s$ for MGCCA (median $\pm$ standard deviation), $0.7 \pm 0.2s$ for CMTF and $0.03 \pm 0.01s$ per tensor for PARAFAC. RGCCA is the only method with an execution time influenced by the SNR, going from $0.08 \pm 0.05s$ ($SNR = -20$ dB) to $0.04 \pm 0.01s$ ($SNR = 0$ dB).

3.3 Global MGCCA

In line of Global RGCCA (see section 2.3), we propose the global MGCCA optimization problem that can extract all the components simultaneously:

$$\max_{\mathbf{W}_1,\ldots,\mathbf{W}_L} \quad \sum_{k,l=1}^{L} c_{kl} \operatorname{Tr}\Big(\mathrm{g}\Big(I^{-1}\mathbf{W}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{W}_l\Big)\Big) \tag{3.14}$$

$$\text{s.t. } \mathbf{W}_l = \mathbf{W}_l^K \odot \mathbf{W}_l^J,\ l = 1,\ldots,L. \tag{3.15}$$

$$\mathbf{W}_l^{K^\top} \mathbf{M}_l^K \mathbf{W}_l^K = \mathbf{W}_l^{J^\top} \mathbf{M}_l^J \mathbf{W}_l^J = \mathbf{I}_R,\ l = 1,\ldots,L. \tag{3.16}$$

where the scheme function g is convex, differentiable and element-wise from $\mathcal{M}_R(\mathbb{R})$ to $\mathcal{M}_R(\mathbb{R})$ (the set of real square matrices of size R). $\mathbf{W}_l = \left[\mathbf{w}_l^{(1)},\ldots,\mathbf{w}_l^{(R)}\right]$ is a $J_l \times R$ matrix composed of the concatenation of the R weight vectors $\mathbf{w}_l^{(r)}$. In constraint (3.15), $\mathbf{W}_l^J = \left[\mathbf{w}_l^{J,(1)},\ldots,\mathbf{w}_l^{J,(R)}\right]$, $\mathbf{W}_l^K = \left[\mathbf{w}_l^{K,(1)},\ldots,\mathbf{w}_l^{K,(R)}\right]$. Moreover $\odot$ is the Khatri-Rao product. The Khatri-Rao product is the column-wise Kronecker product. So this constraint imposes that every weight vector $\mathbf{w}_l^{(r)}$ is modeled as the Kronecker product between a weight vector $\mathbf{w}_l^{K,(r)}$ associated with the K_l frontal slices and a weight vector $\mathbf{w}_l^{J,(r)}$ associated with the J_l lateral slices: $\mathbf{w}_l^{(r)} = \mathbf{w}_l^{K,(r)} \otimes \mathbf{w}_l^{J,(r)}$ (see section 1.3.4.1 for more details about the Khatri-Rao product). Constraint (3.16) is separated in two orthogonality constraints on the weight matrices $\mathbf{W}_l^J$ and $\mathbf{W}_l^K$ in the metric spaces defined by the positive definite matrices $\mathbf{M}_l^J$ and $\mathbf{M}_l^K$ respectively.

The reformulation proposed in (3.14) using the Khatri-Rao constraint applied to the weight matrices enables to rewrite each component associated with the block l and the level r as follows:

$$\mathbf{y}_l^{(r)} = \mathbf{X}_l \mathbf{w}_l^{(r)} = \mathbf{X}_l(\mathbf{w}_l^{K,(r)} \otimes \mathbf{w}_l^{J,(r)}) = \mathbf{X}_l(\mathbf{I}_{K_l} \otimes \mathbf{w}_l^{J,(r)})\mathbf{w}_l^{K,(r)} = \Big(\sum_{j=1}^{J_l} w_{lj}^{J,(r)} \mathbf{X}_{.j.}^l\Big)\mathbf{w}_l^{K,(r)}. \tag{3.17}$$

From equation (3.17), it appears that the component $\mathbf{y}_l^{(r)}$ can be expressed as a linear combination of the columns of the matrix $\sum_{j=1}^{J_l} w_{lj}^{J,(r)} \mathbf{X}_{.j.}^l$ defined as a weighted mean of the lateral slices. In the same way, $\mathbf{y}_l^{(r)}$ can be expressed as a linear combination of the columns of $\sum_{k=1}^{K_l} w_{lk}^{K,(r)} \mathbf{X}_{..k}^l$ defined as a weighted mean of the frontal slices.

In addition, the positive definite matrix $\mathbf{M}_l$ of size $J_l K_l$ is defined as: $\mathbf{M}_l = \mathbf{M}_l^K \otimes \mathbf{M}_l^J$. Then, the optimization problem (3.14) can be simplified by considering the two following transforms $\mathbf{P}_l = I^{-1/2}\mathbf{X}_l \mathbf{M}_l^{-1/2}$ and $\mathbf{V}_l = \mathbf{M}_l^{1/2}\mathbf{W}_l$, which can be re-written as:

$$\begin{aligned}
\mathbf{V}_l = \mathbf{M}_l^{1/2}\mathbf{W}_l &= \left[\left(\mathbf{M}_l^{K^{1/2}} \otimes \mathbf{M}_l^{J^{1/2}}\right)\left(\mathbf{w}_l^{K,(1)} \otimes \mathbf{w}_l^{J,(1)}\right),\ldots,\left(\mathbf{M}_l^{K^{1/2}} \otimes \mathbf{M}_l^{J^{1/2}}\right)\left(\mathbf{w}_l^{K,(R)} \otimes \mathbf{w}_l^{J,(R)}\right)\right] \\
&= \left[\left(\mathbf{M}_l^{K^{1/2}}\mathbf{w}_l^{K,(1)} \otimes \mathbf{M}_l^{J^{1/2}}\mathbf{w}_l^{J,(1)}\right),\ldots,\left(\mathbf{M}_l^{K^{1/2}}\mathbf{w}_l^{K,(R)} \otimes \mathbf{M}_l^{J^{1/2}}\mathbf{w}_l^{J,(R)}\right)\right] \\
&= \left[\mathbf{v}_l^{K,(1)} \otimes \mathbf{v}_l^{J,(1)},\ldots,\mathbf{v}_l^{K,(R)} \otimes \mathbf{v}_l^{J,(R)}\right] = \mathbf{V}_l^K \odot \mathbf{V}_l^J
\end{aligned} \tag{3.18}$$

Finally, the optimization problem (3.14) becomes:

$$\max_{\mathbf{V}_1,\ldots,\mathbf{V}_L} f(\mathbf{V}_1,\ldots,\mathbf{V}_L) = \sum_{k,l=1}^{L} c_{kl} \operatorname{Tr}\Big(\mathrm{g}\Big(\mathbf{V}_k^\top \mathbf{P}_k^\top \mathbf{P}_l \mathbf{V}_l\Big)\Big) \tag{3.19}$$

$$\text{s.t. } \mathbf{V}_l = \mathbf{V}_l^K \odot \mathbf{V}_l^J,\ l = 1,\ldots,L. \tag{3.20}$$

$$\mathbf{V}_l^{K^\top}\mathbf{V}_l^K = \mathbf{V}_l^{J^\top}\mathbf{V}_l^J = \mathbf{I}_R,\ l = 1,\ldots,L. \tag{3.21}$$

The objective function of global MGCCA is similar to the one of global RGCCA and can be re-expressed as:

$$f\left(\mathbf{V}_1,\ldots,\mathbf{V}_L\right)=\sum_{k,l=1}^{L}c_{kl}\sum_{r=1}^{R}\ \mathrm{g}\left(\mathbf{v}_k^{(r)^\top}\mathbf{P}_k^\top\mathbf{P}_l\mathbf{v}_l^{(r)}\right) \tag{3.22}$$

which emphasizes that f is a multi-convex function (see section 2.3 for more explanations).

In the next section, the global MGCCA algorithm is presented.

3.3.1 The Global MGCCA Algorithm

Once again, we make use of the optimization framework described in section 1.5 for solving the global MGCCA optimization problem. The update defined in equation (2.20) for global RGCCA is found again for global MGCCA with differences at the level of the constraints. The global MGCCA update is obtained as solution of the following optimization problem:

$$\hat{\mathbf{V}}_l=\underset{\tilde{\mathbf{V}}_l\in\Omega_l}{\mathrm{argmax}}\ \mathrm{Tr}\left(\nabla_l f(\mathbf{V})^\top\tilde{\mathbf{V}}_l\right)=\underset{\tilde{\mathbf{V}}_l\in\Omega_l}{\mathrm{argmax}}\ \mathrm{Tr}\left(\mathbf{Z}_l^\top\mathbf{P}_l\tilde{\mathbf{V}}_l\right), \tag{3.23}$$

where $\mathbf{V}=(\mathbf{V}_1;\ldots;\mathbf{V}_L)$ and $\Omega_l=\Omega_l^{\odot}\cap\left(\Omega_l^J\times\Omega_l^K\right)$, with $\Omega_l^{\odot}=\left\{\mathbf{V}_l\in\mathbb{R}^{J_lK_l\times R};\ \mathbf{V}_l=\mathbf{V}_l^K\odot\mathbf{V}_l^J\right\}$, $\Omega_l^J=\left\{\mathbf{V}_l^J\in\mathbb{R}^{J_l\times R};\ \mathbf{V}_l^{J^\top}\mathbf{V}_l^J=\mathbf{I}_R\right\}$ and $\Omega_l^K=\left\{\mathbf{V}_l^K\in\mathbb{R}^{K_l\times R};\ \mathbf{V}_l^{K^\top}\mathbf{V}_l^K=\mathbf{I}_R\right\}$. $\nabla_l f(\mathbf{V})$ is the partial gradient of f with respect to $\mathbf{V}_l$ that can be found in equation (2.18) and $\mathbf{Z}_l$ is the matrix composed of the concatenation of the R inner components first described in equation (2.18) also.

The optimization problem (3.23) boils down to finding a pair of weight matrices $\mathbf{V}_l^K$ and $\mathbf{V}_l^J$ such that the Trace of the matrix product between a component matrix $\mathbf{Y}_l=\mathbf{P}_l\tilde{\mathbf{V}}_l$ with an aggregated component matrix $\mathbf{Z}_l$ is maximal. The problem is equivalent to:

$$\left(\hat{\mathbf{V}}_l^J,\hat{\mathbf{V}}_l^K\right)=\underset{\left(\tilde{\mathbf{V}}_l^J,\tilde{\mathbf{V}}_l^K\right)\in\Omega_l^J\times\Omega_l^K}{\mathrm{argmax}}\ \mathrm{Tr}\left(\mathbf{Z}_l^\top\mathbf{P}_l\left(\tilde{\mathbf{V}}_l^K\odot\tilde{\mathbf{V}}_l^J\right)\right) \tag{3.24}$$

There is no analytical solution for the optimization problem (3.24) and the classical procedure consists in alternating between the maximization of the criterion according to $\tilde{\mathbf{V}}_l^J$, keeping $\tilde{\mathbf{V}}_l^K$ fixed, and inversely. Those two steps are repeated until convergence.

In the line of this alternating strategy, the update of $\mathbf{V}_l^J$ is presented hereafter (the case of $\mathbf{V}_l^K$ being similar).

$$\begin{aligned}\hat{\mathbf{V}}_l^J&=\underset{\tilde{\mathbf{V}}_l^J\in\Omega_l^J}{\mathrm{argmax}}\ \sum_{r=1}^{R}\mathbf{z}_l^{(r)^\top}\mathbf{P}_l\left(\tilde{\mathbf{v}}_l^{K,(r)}\otimes\tilde{\mathbf{v}}_l^{J,(r)}\right)=\underset{\tilde{\mathbf{V}}_l^J\in\Omega_l^J}{\mathrm{argmax}}\ \sum_{r=1}^{R}\mathbf{z}_l^{(r)^\top}\left(\sum_{k=1}^{K_l}\tilde{v}_{lk}^{K,(r)}\mathbf{P}_{..k}^l\right)\tilde{\mathbf{v}}_l^{J,(r)}\\&=\underset{\tilde{\mathbf{V}}_l^J\in\Omega_l^J}{\mathrm{argmax}}\ \sum_{r=1}^{R}\tilde{\mathbf{v}}_l^{J,(r)^\top}\left(\sum_{k=1}^{K_l}\tilde{v}_{lk}^{K,(r)}\mathbf{P}_{..k}^{l^\top}\right)\mathbf{z}_l^{(r)}=\underset{\tilde{\mathbf{V}}_l^J\in\Omega_l^J}{\mathrm{argmax}}\ \sum_{r=1}^{R}\tilde{\mathbf{v}}_l^{J,(r)^\top}\mathbf{P}_{l(2)}\left(\tilde{\mathbf{v}}_l^{K,(r)}\otimes\mathbf{I}_I\right)\mathbf{z}_l^{(r)}\\&=\underset{\tilde{\mathbf{V}}_l^J\in\Omega_l^J}{\mathrm{argmax}}\ \mathrm{Tr}\left(\tilde{\mathbf{V}}_l^{J^\top}\mathbf{P}_{l(2)}\left(\tilde{\mathbf{V}}_l^K\odot\mathbf{Z}_l\right)\right):=r_l^J(\mathbf{V}),\end{aligned} \tag{3.25}$$

where $\mathbf{P}_{l(2)}=[\mathbf{P}_{..1}^{l^\top},\ldots,\mathbf{P}_{..K_l}^{l^\top}]$ is the mode-2 matricization of the tensor $\underline{\mathbf{P}}_l$ whose k^{th} frontal slice is $\mathbf{P}_{..k}^l$ (see section 1.3.3.2 for details on matricization).

Hence, $\hat{\mathbf{V}}_l^J$ is given as:

$$\hat{\mathbf{V}}_l^J = \mathbf{Q}_l^J \mathbf{R}_l^{J^\top}, \tag{3.26}$$

where $\mathbf{Q}_l^J \in \mathbb{R}^{J_l \times R}$ and $\mathbf{R}_l^J \in \mathbb{R}^{R \times R}$ are given by the rank-R SVD of $\mathbf{P}_{l(2)} \left(\tilde{\mathbf{V}}_l^K \odot \mathbf{Z}_l \right)$ defined as $\mathbf{P}_{l(2)} \left(\tilde{\mathbf{V}}_l^K \odot \mathbf{Z}_l \right) = \mathbf{Q}_l^J \mathbf{\Delta}_l \mathbf{R}_l^{J^\top}$, with $\mathbf{Q}_l^{J^\top} \mathbf{Q}_l^J = \mathbf{R}_l^{J^\top} \mathbf{R}_l^J = \mathbf{R}_l^J \mathbf{R}_l^{J^\top} = \mathbf{I}_R$ and $\mathbf{\Delta}_l$ a $R \times R$ diagonal matrix whose diagonal elements are all positive and in decreasing order.

Similarly, if we introduce $\mathbf{P}_{l(3)} = [\mathbf{P}_{.1.}^{l^\top}, \ldots, \mathbf{P}_{.J_l.}^{l^\top}]$ the mode-3 matricization of $\underline{\mathbf{P}}_l$, then it can be shown that the update for $\mathbf{V}_l^K$ is:

$$\hat{\mathbf{V}}_l^K = \underset{\tilde{\mathbf{V}}_l^K \in \Omega_l^K}{\operatorname{argmax}} \ \operatorname{Tr} \left(\tilde{\mathbf{V}}_l^{K^\top} \mathbf{P}_{l(3)} \left(\tilde{\mathbf{V}}_l^J \odot \mathbf{Z}_l \right) \right) = \mathbf{Q}_l^K \mathbf{R}_l^{K^\top} := r_l^K(\mathbf{V}), \tag{3.27}$$

where $\mathbf{Q}_l^K \in \mathbb{R}^{K_l \times R}$ and $\mathbf{R}_l^K \in \mathbb{R}^{R \times R}$ are given by the rank-R SVD of $\mathbf{P}_{l(3)} \left(\tilde{\mathbf{V}}_l^J \odot \mathbf{Z}_l \right)$ defined as $\mathbf{P}_{l(3)} \left(\tilde{\mathbf{V}}_l^J \odot \mathbf{Z}_l \right) = \mathbf{Q}_l^K \mathbf{\Delta}_l \mathbf{R}_l^{K^\top}$.

Local solution for (3.24) is usually obtained by alternating (3.26) and (3.27) until convergence. However, instead of proposing an update $\left(\hat{\mathbf{V}}_l^J, \hat{\mathbf{V}}_l^K \right) \in \Omega_l^J \times \Omega_l^K$ that maximizes optimization problem (3.24), $\left(\hat{\mathbf{V}}_l^J, \hat{\mathbf{V}}_l^K \right)$ is found such that it simply increases the objective function. This principle, called generalized block relaxation by [De Leeuw, 1994], yields the following update:

$$r_l(\mathbf{V}) = r_l^K \left(\mathbf{V}_1, \ldots, \mathbf{V}_{l-1}, \mathbf{V}_l^K \odot r_l^J(\mathbf{V}), \mathbf{V}_{l+1}, \ldots, \mathbf{V}_L \right). \tag{3.28}$$

We highlight the fact that in (3.28) the update $\hat{\mathbf{V}}_l^J$ obtained by solving (3.25) is plugged into the computation of $\hat{\mathbf{V}}_l^K$ in (3.27). The entire global MGCCA algorithm is described in Algorithm 5.

Algorithm 5 Global Multiway Generalized Canonical Correlation Analysis algorithm

Data: $\underline{\mathbf{X}}_1, \ldots, \underline{\mathbf{X}}_L$, $\mathbf{M}_1, \ldots, \mathbf{M}_L$, g, ε, $\mathbf{C}$, R
Result: $\mathbf{V}_1^s, \ldots, \mathbf{V}_L^s$ (solution of (3.19) subject to (3.20) and (3.21))
Initialization: $\mathbf{V}_l^0 = \mathbf{V}_l^{K,0} \odot \mathbf{V}_l^{J,0}$, $l = 1, \ldots, L$, where $\mathbf{V}_l^{J,0}$, $\mathbf{V}_l^{K,0}$ are random orthonormal matrices of size $J_l \times R$ and $K_l \times R$ respectively;
$s = 0$
repeat
  for $l = 1$ **to** L **do**
    $\mathbf{V}_l^{s+1} = r_l \left(\mathbf{V}_1^{s+1}, ..., \mathbf{V}_{l-1}^{s+1}, \mathbf{V}_l^s, \mathbf{V}_{l+1}^s, ..., \mathbf{V}_L^s \right) = \mathbf{V}_l^{K,s+1} \odot \mathbf{V}_l^{J,s+1} = \left(\mathbf{Q}_l^{K,s} \mathbf{R}_l^{K,s^\top} \right) \odot \left(\mathbf{Q}_l^{J,s} \mathbf{R}_l^{J,s^\top} \right)$
  end for
  $s = s + 1$;
until $f(\mathbf{V}_1^{s+1}, \ldots, \mathbf{V}_L^{s+1}) - f(\mathbf{V}_1^s, \ldots, \mathbf{V}_L^s) < \varepsilon$

where $\mathbf{Q}_l^{J,s} \in \mathbb{R}^{J_l \times R}$ and $\mathbf{R}_l^{J,s} \in \mathbb{R}^{R \times R}$ are given by the rank-R SVD of $\mathbf{P}_{l(2)} \left(\mathbf{V}_l^{K,s} \odot \mathbf{Z}_l^s \right)$ of dimension $J_l \times R$ where $\mathbf{Z}_l^s = \sum_{k=1}^{l-1} \mathbf{P}_k \mathbf{V}_k^{s+1} \mathbf{D}_{lk}^{s,s+1} + \sum_{k=l}^{L} \mathbf{P}_k \mathbf{V}_k^s \mathbf{D}_{lk}^{s,s}$ of dimension $I \times R$ with $\mathbf{D}_{lk}^{s,t}$ a diagonal matrix of size R whose r^{th} element equals $2c_{lk}\ \mathrm{g}' \left(\mathbf{v}_l^{(r),s^\top} \mathbf{P}_l^\top \mathbf{P}_k \mathbf{v}_k^{(r),t} \right)$.

Moreover $\mathbf{Q}_l^{K,s} \in \mathbb{R}^{K_l \times R}$ and $\mathbf{R}_l^{K,s} \in \mathbb{R}^{R \times R}$ are given by the rank-R SVD of $\mathbf{P}_{l(3)} \left(\mathbf{V}_l^{J,s+1} \odot \mathbf{Z}_l^{s+1/2} \right)$ of dimension $K_l \times R$ and where $\mathbf{Z}_l^{s+1/2}$ is computed similarly to $\mathbf{Z}_l^s$ with $\mathbf{V}_l^s = \mathbf{V}_l^{K,s} \odot \mathbf{V}_l^{J,s+1}$.

At the end of Algorithm 5, the original weight matrices $\mathbf{W}_l^J$ and $\mathbf{W}_l^K$ are recovered by $\mathbf{W}_l^J = \mathbf{M}_l^{J^{-1/2}}\mathbf{V}_l^J$ and $\mathbf{W}_l^K = \mathbf{M}_l^{K^{-1/2}}\mathbf{V}_l^K$ respectively.

Remark. It is worth mentioning that the optimization problem (3.24) at the core of the global MGCCA algorithm is equivalent to:

$$\left(\hat{\mathbf{V}}_l^J, \hat{\mathbf{V}}_l^K\right) = \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmin}} \left\| \underline{\mathbf{P}}_l - [\![\mathbf{Z}_l, \tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K]\!] \right\|_F^2, \tag{3.29}$$

where $\underline{\mathbf{P}}_l = I^{-1/2}\underline{\mathbf{X}}_l \times_2 \mathbf{M}_l^{J^{-1/2}} \times_3 \mathbf{M}_l^{K^{-1/2}}$. The reader is referred to Appendix A, section A.3 for more details. Therefore, the core update of the global MGCCA procedure can be seen as a constrained rank-R CP decomposition where the first mode matrix factor is fixed and where the other factor matrices are constrained to be orthonormal. (cf. section 1.4.1 for the CP decomposition).

3.3.2 Convergence properties of the Global MGCCA algorithm

As detailed below, the convergence properties listed in Proposition 1.5.1 are fulfilled for Algorithm 5.

Proof of Lemma 1 for the global MGCCA algorithm.

Point *(i)* Let f_l be the continuous multilinear function defined as:

$$\begin{aligned} f_l : \Omega_l^K \times \Omega_l^J &\rightarrow \Omega_l \\ (\mathbf{V}_l^K, \mathbf{V}_l^J) &\mapsto \mathbf{V}_l^K \odot \mathbf{V}_l^J. \end{aligned}$$

Ω_l^J and Ω_l^K are the sets of real orthonormal matrices of $\mathcal{M}_{J_l \times R}(\mathbb{R})$ and $\mathcal{M}_{K_l \times R}(\mathbb{R})$ respectively. Those sets are compact. Therefore, Ω_l is a compact set as image of a compact set by the continuous function f_l. Finally, Ω is compact as product of L compact sets.

Point *(ii)* Assuming existence and uniqueness of the solution of the optimization problem (3.25) and (3.27), point (ii) of the Lemma 1.5.2 still holds. The uniqueness arguments that were used for global RGCCA are still valid for global MGCCA.

Point *(iii)* The demonstration presented in Chapter 1 for point *(iii)* of Lemma 1.5.2 still holds using very similar arguments.

Point *(iv)* The proof is based on the uniqueness of $r_l^J(\mathbf{V})$ and $r_l^K(\mathbf{V})$ defined in equation (3.25) and (3.27) respectively. Under mild conditions (see the discussion above), this point is satisfied for global MGCCA. □

3.3.3 Experiments

In this experiment, we compare the performances of global MGCCA, sequential MGCCA and the Coupled Matrix-Tensor Factorization (CMTF) approach. Two scenarii are considered: (i) a coupling of two third-order tensors (tensor/tensor scenario) and (ii) a coupling between a matrix and a third-order tensor (tensor/matrix scenario).

3.3.3.1 Data Generation

For the tensor/tensor scenario, we consider $L = 2$ blocks. Each block is a third-order tensor $\underline{\mathbf{X}}_l$ of dimension $50 \times 50 \times 50$ generated according to the following tensor model:

$$\underline{\mathbf{X}}_l = \eta[\![\boldsymbol{\lambda}; \mathbf{Y}; \mathbf{W}_l^J; \mathbf{W}_l^K]\!] + \frac{\left\|[\![\boldsymbol{\lambda}; \mathbf{Y}; \mathbf{W}_l^J; \mathbf{W}_l^K]\!]\right\|_F}{\|\underline{\mathbf{E}}_l\|_F}\underline{\mathbf{E}}_l, \tag{3.30}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^{R^\star}$. For more details about this notation, see Chapter 1, section 1.4.1.

For the tensor/matrix scenario, $L = 2$ blocks are also considered. On the one hand, a third-order tensor $\underline{\mathbf{X}}_l$ of dimension $50 \times 50 \times 50$ is generated according (3.30). On the other hand, a matrix $\mathbf{X}_l$ of dimension 50×50 is generated according to following matrix model:

$$\mathbf{X}_l = \eta \mathbf{Y}\boldsymbol{\Lambda}_l {\mathbf{W}_l^J}^\top + \frac{\left\|\mathbf{Y}\boldsymbol{\Lambda}_l {\mathbf{W}_l^J}^\top\right\|_F}{\|\mathbf{E}_l\|_F}\mathbf{E}_l, \tag{3.31}$$

where $\boldsymbol{\Lambda}_l$ is a diagonal matrix of dimension $R^\star$.

For these scenarii, each block is generated from $R^\star = 4$ components and $(\boldsymbol{\Lambda}_l)_1 = \lambda_1 = 1,\ (\boldsymbol{\Lambda}_l)_2 = \lambda_2 = 0.8,\ (\boldsymbol{\Lambda}_l)_3 = \lambda_3 = 0.6$ and $(\boldsymbol{\Lambda}_l)_4 = \lambda_4 = 0.4$.

In this experiment, blocks are coupled through the first mode with the same component matrix $\mathbf{Y} \in \mathbb{R}^{N \times R^\star}$ randomly generated such that its columns are centered, normalized and orthogonal. This is the same for $\mathbf{W}_l^J \in \mathbb{R}^{J_l \times R^\star}$, $\mathbf{W}_l^K \in \mathbb{R}^{K_l \times R^\star}$ and $\mathbf{W}_l \in \mathbb{R}^{J_l \times R^\star}$.

The noise matrix $\mathbf{E}_l \in \mathbb{R}^{N \times J_l}$ and the noise tensor $\underline{\mathbf{E}}_l \in \mathbb{R}^{N \times J_l \times K_l}$ are defined such that each of their entries are drawn from a standardized normal distribution. Finally, the Signal to Noise Ratio (SNR) is equal to $20log_{10}(\eta)$ which enables η to drive the SNR.

For the tensor/tensor scenario, let $\mathbf{W}_l^J, \mathbf{W}_l^K$ and $\hat{\mathbf{W}}_l^J, \hat{\mathbf{W}}_l^K$ be respectively the original and the estimated block weight matrices. We quantify how well the estimated block weight matrices match the original ones using the accuracy (ACC) defined as:

$$ACC = \frac{1}{2LR}\sum_{l=1}^{L}\sum_{r=1}^{R} |\hat{\mathbf{w}}_l^{J,(r)^\top}\mathbf{w}_l^{J,(r)}| + |\hat{\mathbf{w}}_l^{K,(r)^\top}\mathbf{w}_l^{K,(r)}|, \tag{3.32}$$

where $\hat{\mathbf{w}}_l^{J,(r)}$ and $\hat{\mathbf{w}}_l^{K,(r)}$ are the r^{th} column of matrices $\hat{\mathbf{W}}_l^J$ and $\hat{\mathbf{W}}_l^K$ respectively.

For the tensor/matrix scenario, let $\mathbf{W}_2$ and $\hat{\mathbf{W}}_2$, be respectively the original and the estimated block weight matrices. In this situation, the ACC is defined as:

$$ACC = \frac{1}{3R}\sum_{r=1}^{R} |\hat{\mathbf{w}}_1^{J,(r)^\top}\mathbf{w}_1^{J,(r)}| + |\hat{\mathbf{w}}_1^{K,(r)^\top}\mathbf{w}_l^{K,(r)}| + |\hat{\mathbf{w}}_2^{(r)^\top}\mathbf{w}_2^{(r)}|, \tag{3.33}$$

where $\hat{\mathbf{w}}_2^{(r)}$ is the r^{th} column of matrix $\hat{\mathbf{W}}_2$.

3.3.3.2 Results

We consider three values of $\eta \in \{0.5, 1, 2\}$. For each scenario (tensor/matrix or tensor/tensor), 100 datasets were generated according to equation (3.30) or (3.31). For each dataset, sequential MGCCA, global MGCCA and CMTF were applied to extract $R = R^\star$ components. For sequential MGCCA, orthogonality are imposed either at the level of components (c-MGCCA) or at the level of the second mode weight vectors (w-MGCCA).

For the three MGCCA procedures, $c_{12} = c_{21} = 1$ and $c_{11} = c_{22} = 0$, the function g was set to the square function (or the element-wise square function) and $\mathbf{M}_1 = \mathbf{M}_2 = \mathbf{I}_R$. Furthermore, CMTF was applied to couple only the first mode between the two blocks. The algorithm used for CMTF is based on a nonlinear conjugate gradient method (see [Acar et al., 2011]). As each method presents potential local minima/maxima, multiple starts were performed (i.e., SVD-based initialization as well as 10 random starts) and the best solution was kept [Acar et al., 2013, ten Berge, 1993].

Moreover, inspired from [Acar et al., 2011], in order to evaluate to what extent global methods are impacted by a misspecification of the number of factors to extract, global MGCCA and CMTF were also run with $R = R^\star + 1$.

The effectiveness of the methods is measured using the ACC metric (defined in equation (3.32) for the tensor/tensor and (3.33) for the tensor/matrix scenario) for each dataset and each procedure. In the case where $R^\star + 1$ components are extracted, the ACC is computed with the $R^\star = 4$ components leading to the highest ACC value. The mean and standard deviation (std) of ACC for each value of η are reported in tables 3.2 (scenario tensor/tensor) and 3.3 (scenario tensor/matrix), along with the median (MD) of the number of iterations (and its interquartile range (IQR)) and the execution time for the best solution.

For the tensor/tensor scenario, table 3.2, its appears that all methods performed well and managed to extract the relevant components. As expected, the mean of ACC increases with the SNR and its standard deviation decreases. Moreover, the execution time and the number of iterations decrease or stay the same when the SNR increases. Surprisingly, the computational time of CMTF with $R^\star + 1$ increases when the SNR increases. This phenomenon might be explained by the fact that it is more difficult to identify a non-existing component when the SNR is high. For global MGCCA and CMTF, extracting either $R^\star$ or $R^\star + 1$ components do not affect the performances. However, it increases the number of iterations and the execution time.

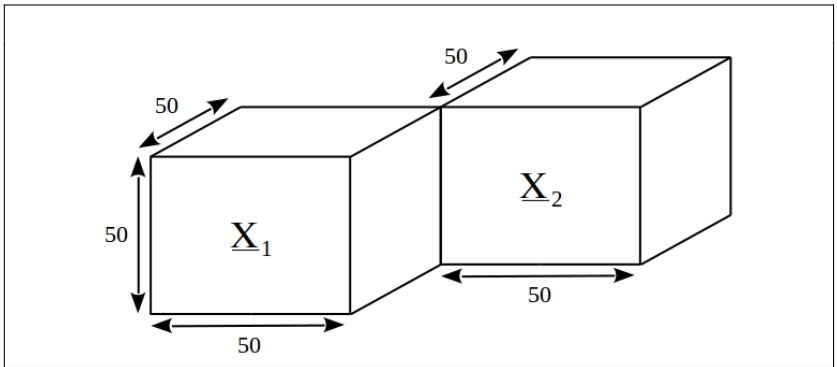
Results of the tensor/matrix scenario are reported in Table 3.3 and similar conclusions can be drawn. We can add that for $\eta = 0.5$, MGCCA methods perform better than CMTF whatever the measurements (ACC/Iter/Time). Moreover, for all the SNR, the performances of CMTF with $R^\star + 1$ components is worse than CMTF with $R^\star$ components, even though it still manages to extract correctly the components.

If we compare the two scenarii, it appears that for the same level of SNR, the ACC computed for the tensor/tensor case is globally higher than the ACC of the tensor/matrix case. This can be explained by the fact that the factorization of a matrix, as assumed in all those methods, suffers from a rotational indeterminacy. This rotational indeterminacy does not exist in the CP-model (but it does in the Tucker model), so coupling a tensor and a matrix help to overcome the rotational indeterminacy that does exist for the matrix. But still, this may explain the lower ACC results. Moreover, in term

of number of iterations and execution time, there seems to be no differences between the two scenarii, except for CMTF with $R^\star+1$ components, where an increase of a factor 2 can be observed. If we focus only on the MGCCA methods, and more specifically on w-MGCCA and global MGCCA that present almost the same orthogonality constraints, it appears that in the tensor/tensor case, global MGCCA is quicker than w-MGCCA by a factor 2. However, in the tensor/matrix scenario, they present almost the same execution time.

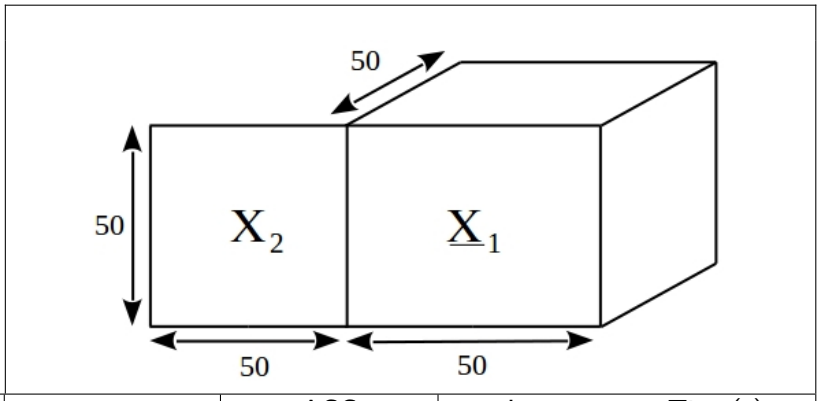
On this simulation, it is worth mentioning that MGCCA algorithms are not sensitive to the starting point as for each MGCCA model and each dataset, the different initializations lead to the same solution.

Table 3.2 – Scenario tensor/tensor. For each value $\eta \in \{0.5, 1, 2\}$, 100 datasets were generated. For each dataset, CMTF and MGCCA were compared. MGCCA was applied with either a deflation (c-MGCCA/w-MGCCA) or a global procedure. For CMTF and global MGCCA, $R^\star$ or $R^\star+1$ components were extracted, where $R^\star$ is the true number of factors. For each method, the same stopping criterion is taken with $\varepsilon = 10^{-8}$. The ACC (defined in equation (3.32)) was computed for each dataset and each procedure, along with the number of iterations and the execution time.



SNR	R	Algorithm	ACC (mean ± std)	Iter (MD - IQR)	Time(s) (mean ± std)
$\eta = 0.5$	$R^\star$	c-MGCCA	0.9950 ± 3e-4	15 - 1	0.5 ± 0.2
		w-MGCCA	0.9951 ± 3e-4	14 - 2	1.0 ± 1.1
		global MGCCA	0.9952 ± 3e-4	5 - 1	0.6 ± 0.2
		CMTF	0.9951 ± 3e-4	53 - 42	8.3 ± 5.9
	$R^\star + 1$	global MGCCA	0.9952 ± 3e-4	48 - 55	4.7 ± 1.6
		CMTF	0.9951 ± 3e-4	406 - 248	14.0 ± 5.5
$\eta = 1$	$R^\star$	c-MGCCA	0.99877± 9e-5	12 - 1	0.4 ± 0.1
		w-MGCCA	0.99879 ± 8e-5	11 - 2	1.4 ± 1.1
		global MGCCA	0.99881 ± 8e-5	4 - 0	0.5 ± 0.1
		CMTF	0.99877 ± 9e-5	56 - 29	3.9 ± 3.0
	$R^\star + 1$	global MGCCA	0.99881 ± 8e-5	29 - 19	2.4 ± 0.5
		CMTF	0.99877 ± 9e-5	555 - 348	20.9 ± 8.3
$\eta = 2$	$R^\star$	c-MGCCA	0.99969 ± 2e-5	12 - 0	0.5 ± 0.2
		w-MGCCA	0.99970 ± 2e-5	12 - 4	1.1 ± 1.1
		global MGCCA	0.99970 ± 2e-5	3 - 1	0.3 ± 0.1
		CMTF	0.99969 ± 2e-5	54 - 44	6.3 ± 9.5
	$R^\star + 1$	global MGCCA	0.99970 ± 2e-5	13 - 5	1.0 ± 0.2
		CMTF	0.99969 ± 2e-5	802 - 507	22.3 ± 8.5

Table 3.3 – Scenario tensor/matrix. For each value $\eta \in \{0.5, 1, 2\}$, 100 datasets were generated. For each dataset, CMTF and MGCCA were compared. MGCCA was applied with either a deflation (c-MGCCA/w-MGCCA) or a global procedure. For CMTF and global MGCCA, $R^\star$ or $R^\star$+1 components were extracted, where $R^\star$ is the true number of factors. For each method, the same stopping criterion is taken with $\varepsilon = 10^{-8}$. The ACC (defined in equation(3.33)) was computed for each dataset and each procedure, along with the number of iterations and the execution time.



SNR	R	Algorithm	ACC (mean ± std)	Iter (MD - IQR)	Time(s) (mean ± std)
$\eta = 0.5$	$R^\star$	c-MGCCA	0.941 ± 5e-3	20 - 2	0.3 ± 0.1
		w-MGCCA	0.941 ± 5e-3	20 - 2	0.4 ± 0.4
		global MGCCA	0.942 ± 5e-3	7 - 2	0.4 ± 0.1
		CMTF	0.91 ± 0.05	94 - 109	8.8 ± 7.9
	$R^\star + 1$	global MGCCA	0.942 ± 5e-3	40 - 34	3.1 ± 1.0
		CMTF	0.88 ± 0.03	481 - 264	37.1 ± 15.6
$\eta = 1$	$R^\star$	c-MGCCA	0.982 ± 2e-3	14 - 1	0.3 ± 0.1
		w-MGCCA	0.982 ± 2e-3	15 - 1	0.3 ± 0.3
		global MGCCA	0.982 ± 2e-3	5 - 1	0.2 ± 0.0
		CMTF	0.981 ± 2e-3	51 - 22	4.8 ± 2.7
	$R^\star + 1$	global MGCCA	0.982 ± 2e-3	29 - 20	2.2 ± 0.5
		CMTF	0.95 ± 0.02	540 - 244	42.0 ± 19.1
$\eta = 2$	$R^\star$	c-MGCCA	0.9951 ± 5e-4	12 - 1	0.3 ± 0.1
		w-MGCCA	0.9951 ± 5e-4	12 - 1	0.2 ± 0.2
		global MGCCA	0.9951 ± 5e-4	4 - 0	0.2 ± 0.0
		CMTF	0.9949 ± 5e-4	54 - 29	4.6 ± 2.0
	$R^\star + 1$	global MGCCA	0.9951 ± 5e-4	20 - 16	1.2 ± 0.3
		CMTF	0.97 ± 0.02	602 - 327	46.3 ± 19.8

3.3.3.3 Conclusion

For the two scenarii, all methods performed similarly. This experiment allows to see global MGCCA as a relevant alternative to CMTF, geared for coupling a collection of tensors and matrices.

In this experiment, orthonormal weight matrices were generated. This choice was made as global MGCCA can handle this constraint when $\mathbf{M}_l^J = \mathbf{I}_{J_l}$ and $\mathbf{M}_l^K = \mathbf{I}_{K_l}$, see optimization problem (3.14), constraint (3.16). Even though CMTF does not assume any orthogonality constraints on the factor matrices, the method fairly manages to estimate them. So in this simulation context, it might be

unnecessary to add orthogonality constraints. However, in a high-dimensional case ($I << K_l$ or $I << J_l$), we believe that enforcing such a constraint might help to better estimate the parameters.

Moreover, as mentioned in section 3.2.3, it might be interesting to evaluate the relative position of the individuals (e.g classification, clustering) in orthonormal bases defined in the variables space. Furthermore, we recall that orthogonality is imposed in the metric space of $\mathbf{M}_l$, which offers the possibility to impose a large variety of orthogonal constraints.

CMTF models the coupled modes by the same matrix and thus imposes identical components, which is a very strong constraint. In global MGCCA, this constraint is relaxed as we tend only to maximize the covariance between block component matrices.

In this chapter, simulation results are presented. The reader is referred to Chapter 5 for an intensive application of MGCCA on real experiments.

3.4 Conclusion and Future Works

In this Chapter, we introduced Multiway Generalized Canonical Correlation Analysis as a versatile framework for analyzing multi-way and multi-block data. Three strategies to determine higher-level components are presented. The first one yields orthogonal components and the second one yields orthogonal weight vectors. The latter respects the multi-way structure of the data. Those two approaches are sequential and rely on a deflation procedure. The third one is global and extracts all the components at the same time. Under mild conditions, the global convergence of sequential MGCCA and global MGCCA algorithms were proven. Furthermore, a Singular Value Decomposition constitutes the core update of each algorithm, which is simple to implement. The presentation of MGCCA is limited to three-way tensors and can be easily extended for higher-order tensors but would require introducing more complex notations.

The reliability of MGCCA to recover interactions between higher-order tensors was demonstrated on simulation studies and compared favorably against existing approaches. A strong improvement of the results compared to RGCCA is also worth noticing, supporting the relevance of adding Kronecker constraints to RGCCA. Moreover, both the sequential and global procedures were compared to CMTF and led to similar results, raising MGCCA to a potential alternative to CMTF. However, when at least two modes need to be coupled, CMTF is a relevant solution.

In appendix A.4 the link between CMTF and global MGCCA is studied. For that purpose, the fourth-order cross-covariance tensors $\mathcal{P}_{lk} = \underline{\mathbf{P}}_l \times_1^1 \underline{\mathbf{P}}_k$, for $1 \leq l < k \leq L$ of dimension $J_l \times K_l \times J_k \times K_k$ is introduced (see section 1.3.4.3 for more details about the operator $\times_1^1$), where $\underline{\mathbf{P}}_l$ is defined in equation (3.29). $\mathcal{P}_{lk}$ contains all 2-by-2 inner products between every mode-1 fibers of $\underline{\mathbf{P}}_l$ and $\underline{\mathbf{P}}_k$. Under additional conditions, it is possible to show that solving the global MGCCA optimization criterion is equivalent to solving a CMTF problem based on coupling all $\mathcal{P}_{lk}$ tensors for $1 \leq l < k \leq L$ along all 4 modes. This relation gives more insights on the two methods and open new ways of developments. Current work aims at studying this equivalence on simulations.

In [Min et al., 2019] a two-block Tensor CCA (TCCA) is presented along with multiple extensions. In this TCCA model, following our notations, the block TCCA components are computed as (see

equation (1.19) for more details):

$$\mathbf{y}_l = \sum_{r=1}^{R_l} \underline{\mathbf{X}}_l \times_2 \mathbf{w}_l^{J,(r)^\top} \times_3 \mathbf{w}_l^{K,(r)^\top} = \mathbf{X}_l \sum_{r=1}^{R_l} \mathbf{w}_l^{K,(r)} \otimes \mathbf{w}_l^{J,(r)}, \ l = 1, 2$$

In comparison to global MGCCA, where only the covariance between block components of the same level are taken into account, all conceivable inter-level covariances between block components are considered. Among other thing, this allows to extract different number of components per block. A BCA procedure is also used in TCCA, where at each step, a one-component CCA problem is solved. However, no orthogonal constraint is imposed for each block weight matrix. When only one component is sought, with the right specification of the $\mathbf{M}_l$ matrices, MGCCA and TCCA are identical.

Future works also include developing a sparse version of MGCCA. When sparse constraints are added to the global MGCCA criterion, they can take the form of an ℓ_1 penalty on the whole block matrix $\mathbf{V}_l = \mathbf{V}_l^K \odot \mathbf{V}_l^J$. This would allow sparsity to spread among variables, component levels and modes. Another possibility is to apply a group-LASSO penalty, where each variable, over each component level, form a group. This would result in the selection of entire rows of the block weight matrices. Such ideas were presented in the context of CCA in [Kanatsoulis et al., 2019].

* * *
* *
*

Chapter 4

Structured Sparse Generalized Canonical Correlation Analysis

Chapter Outline

The materials contained in this chapter were presented at national/international conferences or international journals and are also the subject of a publication in preparation :

Arnaud Gloaguen, Vincent Guillemot, Arthur Tenenhaus. *An efficient algorithm to satisfy l1 and l2 constraints.*49ème Journées de Statistique (JDS), Avignon, France, 2017. Oral.

Nicolas Guigui, Cathy Philippe, **Arnaud Gloaguen**, Slim Karkar, Vincent Guillemot, Tommy Löfstedt, Vincent Frouin. *Network Regularization in Imaging Genetics Improves Prediction Performances and Model Interpretability on Alzheimer's Disease.* Proceedings of the IEEE International Symposium of Biomedical Imaging, Venice, Italy, 2019. Oral.

Vincent Guillemot, Derek Beaton, **Arnaud Gloaguen**, Tommy Löfstedt, Brian Levine, Nicolas Raymond, Arthur Tenenhaus and Hervé Abdi. *A constrained singular value decomposition method that integrates sparsity and orthogonality.* PLoS ONE, Public Library of Science, 2019, 14 (3), pp.e0211463.

Vincent Guillemot, Julie Le Borgne, **Arnaud Gloaguen**, Arthur Tenenhaus, Gilbert Saporta, Sylvie Chollet, Derek Beaton, Hervé Abdi. *Sparse Multiple Correspondence Analysis.* 52ème Journées de Statistique (JDS), France, 2020. Submitted.

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Laurent Le Brusquet, Arthur Tenenhaus. *A MM Algorithm for Structured Sparse Generalized Canonical Correlation Analysis.* In preparation.

To improve the interpretability of the RGCCA model, an important task is to identify subsets of variables within each block that are active in the relationships among blocks. This variable selection step can be achieved by adding within the RGCCA optimization process different kinds of penalty promoting sparsity (ℓ_1) or structured sparsity (like group LASSO, sparse group, fused or elitist LASSO penalty); compared to competing methods, RGCCA enables to choose a specific penalty for each block according to its nature.

4.1 Introduction

To improve the interpretability of the RGCCA model, an important task is to identify subsets of variables within each block that are active in the relationships among blocks. This variable selection step can be achieved by adding within the RGCCA optimization process different kinds of penalty promoting sparsity (ℓ_1) or structured sparsity (like group LASSO, sparse group, fused or elitist LASSO penalty). In general, one might think to penalize the ℓ_0-pseudo-norm of the block weight vector to enforce variable selection. However, the resulting optimization problem becomes really hard to solve due to the combinatorial properties of the ℓ_0-pseudo-norm and its non-convexity. A relaxation of this problem was proposed by replacing the ℓ_0-pseudo-norm by its tightest convex envelop [Boyd and Vandenberghe, 2004], the ℓ_1-norm: $\|\mathbf{x}\|_1 = \sum_{j=1}^{J} |x_j|$. An ℓ_1-norm was added to the RGCCA optimization problem, this algorithm is called Sparse Generalized Canonical Correlation Analysis (SGCCA) [Tenenhaus et al., 2014]. At the heart of the SGCCA algorithm lies an optimization problem under both an ℓ_1 and an ℓ_2-norm constraint. Section 4.3, presents a new procedure for solving this problem. The convergence properties of SGCCA are also studied.

In dedicated problem, one might want to select together variables that are known to interact. This can be achieved by regularizing with norms that are different from the ℓ_1-norm. Section 4.4 presents a data integration framework adapted to most frequent sparsity-inducing norms.

4.2 Sparse Generalized Canonical Correlation Analysis (SGCCA)

A collection of L data matrices $\mathbf{X}_1, \ldots, \mathbf{X}_l, \ldots, \mathbf{X}_L$ is introduced. Each $I \times J_l$ data matrix $\mathbf{X}_l = [\mathbf{x}_{l1}, \ldots, \mathbf{x}_{lJ_l}]$ is a set of J_l variables observed on I individuals. The number and the nature of the variables may differ from one block to another, but the individuals must be the same across blocks. We assume that all variables are centered. A sparse version of RGCCA called SGCCA [Tenenhaus et al., 2014] was proposed to add an ℓ_1-norm constraint to the weights in order to perform variable selection. The optimization criterion of SGCCA can be written as:

$$\begin{aligned} \max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} f(\mathbf{w}_1, \ldots, \mathbf{w}_L) &= \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) \\ \text{s.t. } \mathbf{w}_l &\in \Omega_l,\ l = 1, \ldots, L, \end{aligned} \tag{4.1}$$

where function g, and the design matrix $\mathbf{C} \in \mathbb{R}^{L\times L}$ are defined in chapter 2 section 2.2.1.

In the original presentation of SGCCA , $\Omega_l = \{\mathbf{w}_l \in \mathbb{R}^{I\times J_l}; \|\mathbf{w}_l\|_2 = 1; \|\mathbf{w}_l\|_1 \leq s_l\}$, with $s_l \in \mathbb{R}_+^\star$. Here, in order to ease the convergence study of SGCCA, a slighty different set is considered:

$$\Omega_l = \{\mathbf{w}_l \in \mathbb{R}^{I\times J_l}; \|\mathbf{w}_l\|_2 \leq 1; \|\mathbf{w}_l\|_1 \leq s_l\}. \tag{4.2}$$

This set is defined as the intersection between the ℓ_2-ball of radius 1 and the ℓ_1-ball of radius $s_l \in \mathbb{R}_+^\star$ which are two convex sets. Hence, Ω_l is a convex set.

In comparison to RGCCA, $\mathbf{M}_l,\ l = 1, \ldots, L$, are all set to the identity in SGCCA optimization problem.

4.2.1 Intersection between the ℓ_2 and ℓ_1-spheres

In the rest of this chapter, the assumption is made that the ℓ_2-ball of radius 1 is not included in the ℓ_1-ball of radius s_l and the other way round. Otherwise systematically, only one of the two constraints is active. This assumption is true when the corresponding spheres intersect. When $J_l = 2$, the two borderline cases are shown on Figure 4.2-1a. This assumption can be translated into conditions on s_l.

The norm equivalence between $\|.\|_1$ and $\|.\|_2$ can be formulated as the following inequality:

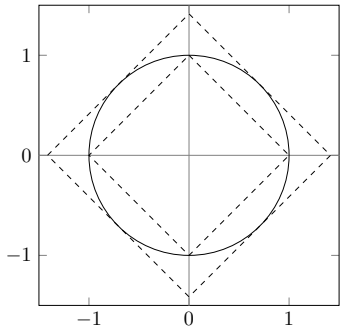
$$\forall \mathbf{x} \in \mathbb{R}^{J_l},\ \|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1 \leq \sqrt{J_l}\|\mathbf{x}\|_2. \tag{4.3}$$

This can be converted into a condition on s_l: $1 \leq s_l \leq \sqrt{J_l}$. When such condition is fulfilled, the ℓ_2-sphere of radius 1 and the ℓ_1-sphere of radius s_l intersect. This is depicted in figure 4.2-1b.

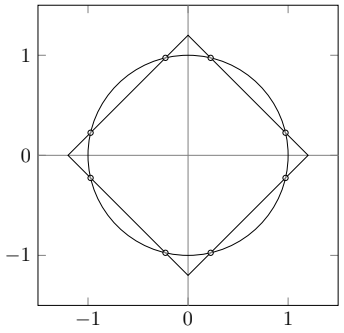
4.2.2 The SGCCA Algorithm

SGCCA and RGCCA have the same objective function so the general optimization framework presented in section 1.5 applies for SGCCA. Hence, under this framework, the update defined in equation (2.4) for RGCCA can be used again here but with Ω_l defined in (4.2). This update tries to find $\hat{\mathbf{w}}_l \in \Omega_l$ obtained by considering the optimization problem below:

$$\hat{\mathbf{w}}_l = \underset{\substack{\|\tilde{\mathbf{w}}_l\|_2\leq 1 \\ \|\tilde{\mathbf{w}}_l\|_1\leq s_l}}{\operatorname{argmax}}\ \nabla_l f(\mathbf{w})^\top \tilde{\mathbf{w}}_l := r_l(\mathbf{w}) \tag{4.4}$$



(a) The ℓ_2-sphere of radius 1 (continuous line) and the ℓ_1-spheres of radius $s_l = 1$ and $s_l = \sqrt{2}$ (dashed lines).



(b) The ℓ_2-sphere of radius 1 and the ℓ_1-sphere of radius $s_l = 1.2$. The small circles highlight the intersections between the two spheres.

Figure 4.2-1 – Conditions for the intersection between the ℓ_2 and ℓ_1-spheres.

where $\mathbf{w} = (\mathbf{w}_1; \ldots; \mathbf{w}_L)$ and $\nabla_l f(\mathbf{w})$ is the partial gradient of f with respect to $\mathbf{w}_l$ that can be found in equation (2.5).

According to [Witten et al., 2009], solution of (4.4) satisfies:

$$r_l(\mathbf{w}) = \hat{\mathbf{w}}_l = \frac{\mathcal{S}(\nabla_l f(\mathbf{w}), \lambda_l)}{\|\mathcal{S}(\nabla_l f(\mathbf{w}), \lambda_l)\|_2}, \text{ where } \lambda_l = \begin{cases} 0 \text{ if } & \frac{\|\nabla_l f(\mathbf{w})\|_1}{\|\nabla_l f(\mathbf{w})\|_2} \leq s_l \\ \text{find } \lambda_l \text{ such that } & \|\hat{\mathbf{w}}_l\|_1 = s_l \end{cases}, \tag{4.5}$$

where function $\mathcal{S}(., \lambda)$ is the soft-thresholding operator. When applied on a vector $\mathbf{x} \in \mathbb{R}^J$, this operator is defined as:

$$\mathbf{u} = \mathcal{S}(\mathbf{x}, \lambda) \Leftrightarrow u_j = \begin{cases} \text{sign}(x_j)(|x_j| - \lambda), & \text{if } |x_j| > \lambda \\ 0, & \text{if } |x_j| \leq \lambda \end{cases}, j = 1, \ldots, J. \tag{4.6}$$

In the case of a scalar $a \in \mathbb{R}$, Figure 4.2-2 depicts the function $\mathcal{S}(., \lambda)$ with $\lambda = 1$.

The entire SGCCA Algorithm is presented in Algorithm 6.

The next section proposes an algorithm to solve problem (4.5) at the heart of SGCCA.

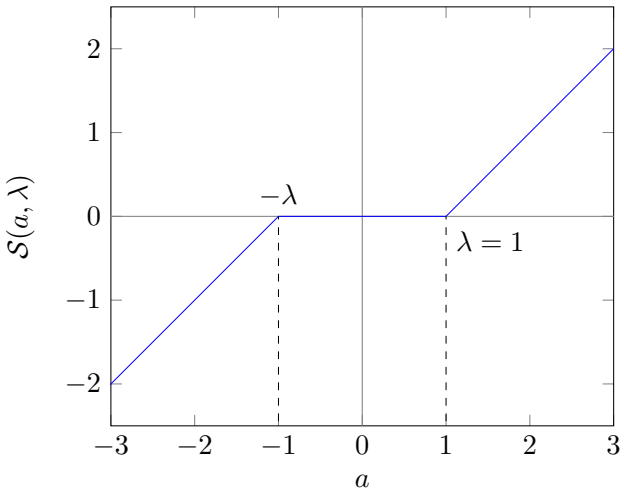


Figure 4.2-2 – Soft-thresholding operator $\mathcal{S}(a, \lambda)$ in the case where $a \in \mathbb{R}$ and $\lambda = 1$.

Algorithm 6 Sparse Generalized Canonical Correlation Analysis (SGCCA) algorithm

Data: $\mathbf{X}_1, \ldots, \mathbf{X}_L$, g, ε, $\mathbf{C}$, $s_1, \ldots, s_L$ and $\forall l,\ 1 \leq s_l \leq \sqrt{J_l}$
Result: $\mathbf{w}_1^s, \ldots, \mathbf{w}_L^s$ (solution of (4.1) subject to (4.2))
Initialization: choose random unit-norm $\mathbf{w}_l^0, l = 1, \ldots, L$.
$s = 0$;
repeat
  for $l = 1$ **to** L **do**

$$\mathbf{w}_l^{s+1} = r_l\left(\mathbf{w}_1^{s+1}, ..., \mathbf{w}_{l-1}^{s+1}, \mathbf{w}_l^s, \mathbf{w}_{l+1}^s, ..., \mathbf{w}_L^s\right) = \frac{\mathcal{S}(\nabla_l^s f, \lambda_l)}{\|\mathcal{S}(\nabla_l^s f, \lambda_l)\|_2}, \tag{4.7}$$

  end for
  $s = s + 1$;
until $f(\mathbf{w}_1^{s+1}, \ldots, \mathbf{w}_L^{s+1}) - f(\mathbf{w}_1^s, \ldots, \mathbf{w}_L^s) < \varepsilon$

where $\nabla_l^s f = \mathbf{X}_l^\top \left(2\sum_{k=1}^{l-1} c_{lk}\ \mathrm{g}'(\mathbf{w}_l^{s\top}\mathbf{X}_l^\top\mathbf{X}_k\mathbf{w}_k^{s+1})\mathbf{X}_k\mathbf{w}_k^{s+1} + 2\sum_{k=l}^{L} c_{lk}\ \mathrm{g}'(\mathbf{w}_l^{s\top}\mathbf{X}_l^\top\mathbf{X}_k\mathbf{w}_k^{s})\mathbf{X}_k\mathbf{w}_k^{s}\right)$.
$\lambda_l = 0$ if $\|\mathcal{S}(\nabla_l^s f, \lambda_l)\|_1/\|\mathcal{S}(\nabla_l^s f, \lambda_l)\|_2 \leq s_l$ and λ_l is chosen such that $\|\mathbf{w}_l^{s+1}\|_1 = s_l$ otherwise.

4.3 The update function of SGCCA

This section focuses on the update of the SGCCA algorithm presented in equation (4.4). This problem can be stated as follows:

$$\underset{\mathbf{x}\in\Omega}{\operatorname{argmax}}\ \mathbf{a}^\top\mathbf{x}, \tag{4.8}$$

where $\mathbf{a} \in \mathbb{R}^J$ and $\Omega = \left\{\mathbf{x} \in \mathbb{R}^J \mid \|\mathbf{x}\|_2 \leq 1 \text{ and } \|\mathbf{x}\|_1 \leq s\right\}$ with $s \in \mathbb{R}_+^\star$. As mentioned above, the solution of (4.8) satisfies $\mathbf{u} = \mathcal{S}(\mathbf{a}, \lambda)/\|\mathcal{S}(\mathbf{a}, \lambda)\|_2$, where $\lambda = 0$ if $\|\mathbf{a}\|_1/\|\mathbf{a}\|_2 \leq s$ and λ is chosen such that $\|\mathbf{u}\|_1 = s$ otherwise.

Several strategies such as Binary Search or the Projection On Convex Set algorithm (POCS), also known as alternating projection method [Boyd and Dattorro, 2003], can be used to determine λ verifying the ℓ_1-norm constraint. Here, an alternative approach inspired by [van den Berg et al., 2008] is proposed. In the entire section 4.3, the following assumption is made:

$$\operatorname{card}\left(\underset{i\in[\![1,J]\!]}{\operatorname{argmax}}\ |a_i|\right) = 1. \tag{4.9}$$

This assumption is equivalent to say that the maximum value of the vector $|\mathbf{a}|$ is reached for only one element. Appendix B, section B.1 justifies the importance of this assumption.

4.3.1 Projection algorithm onto the ℓ_1-norm ball

The proposed approach relies on a very efficient algorithm for projecting a point onto the ℓ_1-ball [van den Berg et al., 2008]. This approach is summarized below. Let $\tilde{\mathbf{a}}$ be the absolute value of $\mathbf{a}$ with its elements sorted in decreasing order. Further, we define the function $\varphi(\lambda) = \|\mathcal{S}(\mathbf{a}, \lambda)\|_1$ which is continuous, piecewise linear and decreasing from $\varphi(0) = \|\tilde{\mathbf{a}}\|_1$ to $\varphi(\tilde{a}_1) = 0$. Therefore, if $\|\mathbf{a}\|_1 \geq s$, as φ is continuous, there exists λ such that $\varphi(\lambda) = s$.

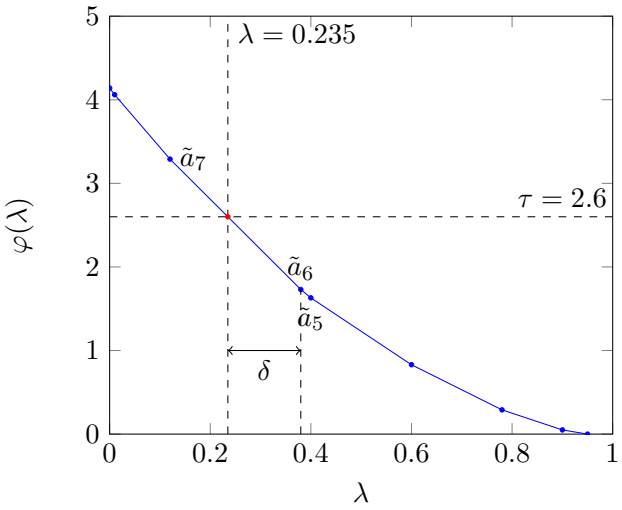


Figure 4.3-3 – Principle of the projection algorithm onto the ℓ_1-norm ball. $\varphi(\lambda)$ is represented in the case of $\tilde{\mathbf{a}} = (0.95, 0.9, 0.78, 0.6, 0.4, 0.38, 0.12, 0.01)$ and $s = 2.6$.

Hence, the projection algorithm reduces to 4 steps :

1. Take the absolute value of $\mathbf{a}$ and sort its elements in decreasing order to get $\tilde{\mathbf{a}}$.
2. Find i such that $\varphi(\tilde{a}_i) \leq s < \varphi(\tilde{a}_{i+1})$.
3. Find δ such that $\varphi(\tilde{a}_i - \delta) = s$. As $\varphi(\tilde{a}_i - \delta) = \sum_{j=1}^{i} \tilde{a}_j - i(\tilde{a}_i - \delta) = \varphi(\tilde{a}_i) + i\delta$ then $\delta = \frac{s - \varphi(\tilde{a}_i)}{i}$.
4. Compute $\mathcal{S}(\mathbf{a}, \lambda) = \text{sign}(\mathbf{a}) \max(|\mathbf{a}| - \lambda, 0)$ with $\lambda = \tilde{a}_i - \delta$.

The idea of this algorithm can be summarized by Figure 4.3-3 where $\varphi(\lambda)$ is represented in a specific case. This function is indeed piecewise linear. As mentioned above, $\tilde{\mathbf{a}}$ is the absolute value of $\mathbf{a}$ with its elements sorted in decreasing order. Then $\tilde{a}_6$ and $\tilde{a}_7$ are the adjacent elements of $\tilde{\mathbf{a}}$ that frame the optimal λ. From $\tilde{a}_6$, a linear interpolation is used to find δ such that $\lambda = \tilde{a}_6 - \delta$.

A similar algorithm has been proposed by [Candes and Romberg, 2005, Daubechies et al., 2008, Duchi et al., 2008].

4.3.2 Scalar product maximization under ℓ_1 and ℓ_2-norm constraints

A similar strategy was adopted when both ℓ_1 and ℓ_2-norm constraints have to be satisfied [Gloaguen et al., 2017]. The starting point is to consider the function $\psi(\lambda) = \|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_1 / \|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_2$. It has to be shown that $\psi(\lambda)$ is continuous and monotonically decreasing. Moreover, a new expression of δ has to be found.

To this end, the following two propositions were made.

Proposition 4.3.1. *The following function, defined on* $[0; \tilde{a}_2] \mapsto \mathbb{R}^+$*, is strictly decreasing:*

$$\psi(\lambda) = \frac{\|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_1}{\|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_2}, \tag{4.10}$$

with $\psi(0) = \|\mathbf{a}\|_1 / \|\mathbf{a}\|_2$ *and* $\psi(\tilde{a}_2) = 1$.

Remark. In Proposition 4.3.1, the function is defined on $[0;\tilde{a}_2]$ only. Indeed, $\forall\lambda \in [\tilde{a}_2;\tilde{a}_1[$, $\mathcal{S}(\tilde{\mathbf{a}},\lambda)$ is composed of only one non-null element which is $\tilde{a}_1 - \lambda$. Hence, $\|\mathcal{S}(\tilde{\mathbf{a}},\lambda)\|_1 = \|\mathcal{S}(\tilde{\mathbf{a}},\lambda)\|_2 = \tilde{a}_1 - \lambda$ and $\psi(\lambda) = 1$. So function ψ is constant on $[\tilde{a}_2;\tilde{a}_1[$ that is why it was excluded from its definition in order to create a strictly decreasing function. When $s = 1$, the entire interval $[\tilde{a}_2;\tilde{a}_1[$ is a solution for λ. However, on this interval, the only element that belongs to the interval of definition of ψ is $\tilde{a}_2$.

Proposition 4.3.2. *Giving $s \in [1;\sqrt{J}]$, the assumption that $\|\mathbf{a}\|_1/\|\mathbf{a}\|_2 > s$ is made. Then, there exists a unique $i \in [\![2;J]\!]$ and a unique $\delta \in [0;\tilde{a}_i - \tilde{a}_{i+1}[$ such that $\psi(\tilde{a}_i - \delta) = s$ and δ is a root of a second degree polynomial equation.*

Remark. In the previous proposition, the assumption that $\|\mathbf{a}\|_1/\|\mathbf{a}\|_2 > s$ is made. Indeed, if $\|\mathbf{a}\|_1/\|\mathbf{a}\|_2 \leq s$, then $\mathbf{a}/\|\mathbf{a}\|_2$ is solution of (4.8) as mentioned in the preamble of this section.

The demonstration of proposition 4.3.1 and 4.3.2 is available in appendix B.

Thanks to these two propositions, a four step algorithm is proposed for solving the optimization problem (4.8):

1. Take the absolute value of $\mathbf{a}$ and sort its elements in decreasing order to get $\tilde{\mathbf{a}}$.
2. Find i such that $\psi(\tilde{a}_i) \leq s < \psi(\tilde{a}_{i+1})$.
3. $\delta = \frac{\|\mathcal{S}(\tilde{\mathbf{a}},\tilde{a}_i)\|_2}{i}\left(s\sqrt{\frac{i-\psi(\tilde{a}_i)^2}{i-s^2}} - \psi(\tilde{a}_i)\right)$.
4. Compute $\mathcal{S}(\mathbf{a},\lambda) = \text{sign}(\mathbf{a})\max(|\mathbf{a}| - \lambda, 0)$ with $\lambda = \tilde{a}_i - \delta$.

A similar algorithm is proposed in [Thom and Palm, 2013].

Sorting the elements of $\mathbf{a}$ in step 1 implies that the complexity is at least in $\mathcal{O}(J\ln(J))$ with J the dimension of $\mathbf{a}$. A strategy preventing this sorting step is proposed in [van den Berg et al., 2008] which reduces the time complexity to $\mathcal{O}(J)$ for the projection onto the ℓ_1-ball. We used the same strategy for our implementation, but the complexity of the algorithm has not yet been assessed.

The next section provides a comparison of the proposed method, Binary Search, POCS, and the projection onto the ℓ_1-ball proposed by [van den Berg et al., 2008].

4.3.3 Results

Four methods were selected for the comparison: a binary search (Binary) algorithm to find λ solution of (4.5), a POCS algorithm consisting in alternating projection onto the ℓ_1 and the ℓ_2-norm balls, our method (Fast_l1_l2) and the projection on the ℓ_1-ball of radius s mentioned in [van den Berg et al., 2008] (Proj_l1). Proj_l1 does not solve the same problem as Binary/POCS/Fast_l1_l2. However, as Fast_l1_l2 is inspired from Proj_l1, we decided to include Proj_l1 in the comparison. Figure 4.3-4 (a) shows that Fast_l1_l2 performs very similarly to Proj_l1 and is almost 10 times faster than Binary and POCS. Figure 4.3-4 (b) depicts the runtime performances of the different methods as a function of J.

Fast_l1_l2 has been embedded within the SGCCA algorithm and compared to the original implementation with binary search. For this experiment, we applied SGCCA to the same 3-block real

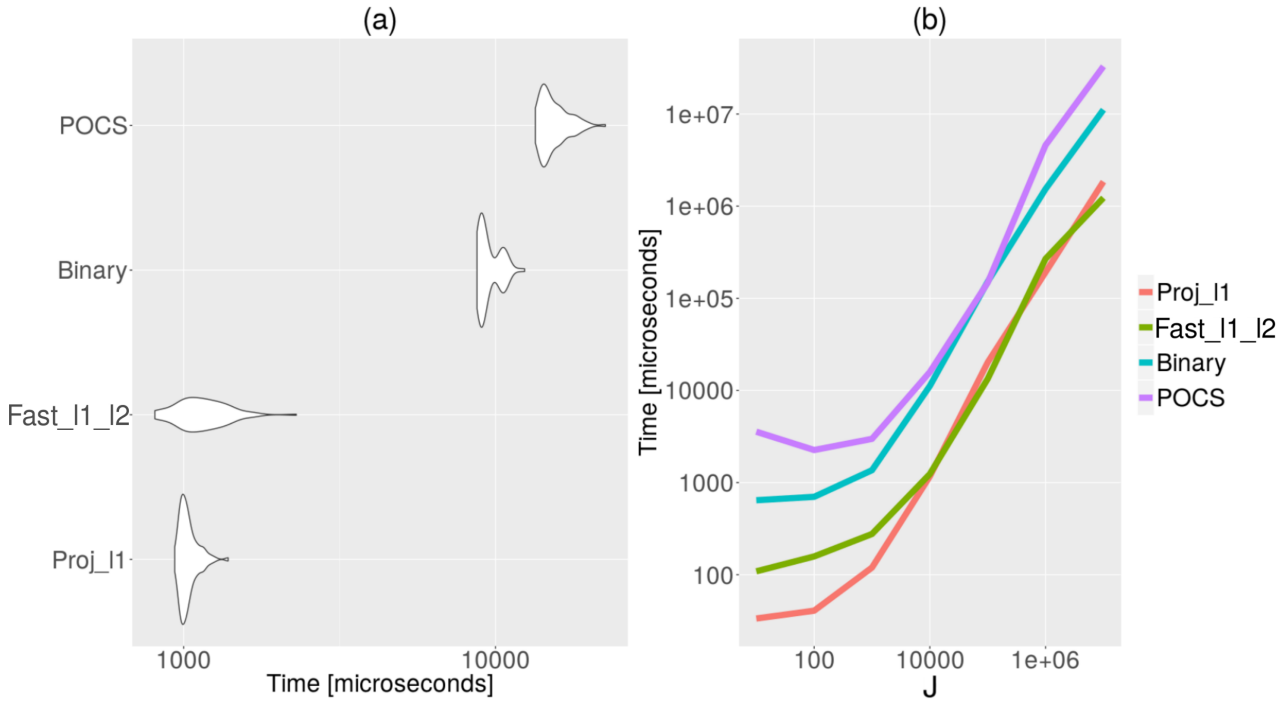


Figure 4.3-4 – (a): Violin plots of runtime for each method throughout 100 runs for a vector of length J =10.000. (b): Log-log plot representing the mean runtime over 20 runs for each method for different value of J. Each time computations are done with $s = 2.3$, on a vector generated thanks to a reduced and centered Gaussian law.

dataset of dimensions $J_1 = 15702$, $J_2 = 1229$, and $J_3 = 3$ measured on a cohort of 53 patients, as in the original article. For each of these two algorithms SGCCA was run 20 times. Binary search (resp. Fast_l1_l2) converged in $10.48s$ (resp. $7.62s$) on average with a standard deviation of $0.60s$ (resp. $0.29s$) on a mid-range laptop computer. We notice that the two implementations of the SGCCA algorithm converged to the same solution.

4.3.4 Convergence properties of SGCCA

The convergence properties subsumed in Proposition 1.5.1 are satisfied for Algorithm 6. In order to show that Proposition 1.5.1 holds for the SGCCA algorithm, Points (*i-iv*) of Lemma 1.5.2 are demonstrated below.

Proof of Lemma 1.5.2 for the SGCCA Algorithm.

Point *(i)* Ω_l defined in (4.2) is compact as intersection of two compact sets. We conclude that Ω is compact as product of L compact sets.

Point *(ii)* As long the solution of optimization problem (4.4) exists and is unique, the demonstration of the point (*ii*) of the Lemma 1.5.2 made in Chapter 1 still holds. Based on Proposition 4.3.2, if:

$$\text{card}\left(\underset{i\in[\![1,J]\!]}{\text{argmax}}\ |(\nabla_l^s f)_i|\right) = 1, \tag{4.11}$$

the solution of optimization problem (4.4) is unique (see Algorithm 6 for a definition of $\nabla_l^s f$).

Point *(iii)* The demonstration presented in Chapter 1 for point *(iii)* of Lemma 1.5.2 still holds here.

Point *(iv)* The proof is based on the uniqueness of $r_l(\mathbf{w})$ defined in equation (4.4). Under mild conditions (see the discussion above), this point is satisfied for SGCCA. □

So far, we were interested in the ℓ_1-norm in order to perform variable selection. However, such norm treat each variable individually without taking into account the interactions that may exist between them. For example, for intra-block variables that are known to interact, one might be interested in selecting them together. In the next section, SGCCA is enhanced with sparsity-inducing norms that enable encoding some additional structure about the variables.

4.4 Structured SGCCA

Recently, structured penalties were embedded into the objective function of SGCCA in order to select variables that are known to interact [Löfstedt et al., 2016]. To account for such structures, the following optimization problem is defined:

$$\max_{\mathbf{w}_1,\ldots,\mathbf{w}_L} f(\mathbf{w}_1,\ldots,\mathbf{w}_L) = \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) - \sum_{k=1}^{L} \psi_k p_k\left(\mathbf{w}_k\right) \tag{4.12}$$

$$\text{s.t. } \begin{cases} \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l \leq 1 \\ \|\mathbf{w}_l\|_1 \leq s_l \end{cases}, \ l = 1,\ldots,L, \tag{4.13}$$

where the functions $p_k,\ k = 1,\ldots,L$ are convex structured penalties (not necessarily differentiable) with corresponding regularization parameters ψ_k. In the framework proposed to solve (4.12), the function g is set to the identity to enforce the criterion to be multi-concave. This optimization procedure relies on a smoothing technique first described by [Nesterov, 2004], that provides an efficient method to smooth a non-differentiable function such that it becomes everywhere differentiable [Hadj-Selem et al., 2018]. This technique is very similar to the Smoothing Proximal-Gradient (SPG) method presented in [Chen et al., 2012b]. A slight modification of SPG was introduced in the framework of CCA for exploiting pre-given structures via structured-sparsity-inducing penalties [Chen and Liu, 2012, Chen et al., 2012a].

In these two approaches [Chen et al., 2012b, Löfstedt et al., 2016], the functions p_k have to be written as:

$$p_k(\mathbf{w}_k) = \max_{\boldsymbol{\alpha}\in\mathcal{K}} \boldsymbol{\alpha}^\top \mathbf{A}_k \mathbf{w}_k, \tag{4.14}$$

with $\mathcal{K}$ a compact convex set in a finite dimensional space and $\mathbf{A}_k$ a linear operator between two finite dimensional vector spaces. Functions $p_k,\ k = 1,\ldots,L$ are consequently convex as the maximization of a convex function over a convex set is convex. If formulation (4.14) is possible, Nesterov's smoothing technique introduces a parameter γ to assess the level of smoothness. Many classical penalties fall down to this formulation including group-lasso [Yuan and Lin, 2006], overlapping group lasso [Obozinski et al., 2011], total variation [Rudin et al., 1992], graph-constrained Elastic Net [Grosenick et al., 2009], graph-guided fusion penalty [Chen et al., 2012a, Kim and Cipolla, 2009] to name a few.

In this chapter a novel optimization framework is proposed to solve problem (4.12) that relies on the generalized block relaxation method and the Maximization by Minorization (MM) principle [De Leeuw, 1994, Lange, 2016]. In regards of the algorithms mentioned above, this new framework can handle any continuously differentiable function g and non-necessarily convex penalties. The only requirement for functions p_k is that they can be majorized by a quadratic and convex function.

This algorithmic framework has already been addressed in the literature in the context of CCA combined with specific structured sparse regularizations: graph-constrained Elastic Net [Du et al., 2015, 2016b], graph Octagonal Shrinkage and Clustering Algorithm for Regression (OSCAR) [Du et al., 2016a] and the truncated-ℓ_1 norm [Du et al., 2017] which is a surrogate function of the ℓ_0-pseudo-norm. For each penalty, a perturbation of the real penalty is considered to avoid dividing by 0. We propose a new framework that enables to solve (4.12) subject to (4.13) that is not associated with peculiar structured sparse penalties, that can handle an arbitrary number of blocks and any convex differentiable function g.

4.4.1 The Structured SGCCA Algorithm

The Structured SGCCA Algorithm combines the distance majorization method introduced in [Chi et al., 2013] and the principle of generalized Block Relaxation [De Leeuw, 1994].

4.4.1.1 Quadratic penalty method

The distance majorization method is a quadratic penalty method [Bertsekas et al., 2020] associated with a specific iterative algorithm.

Let $\Omega_l^1 = \{\mathbf{w}_l \in \mathbb{R}^{J_l}; \|\mathbf{w}_l\|_1 \leq s_l\}$, $\Omega_l^2 = \{\mathbf{w}_l \in \mathbb{R}^{J_l}; \mathbf{w}_l^\top \mathbf{M}_l \mathbf{w}_l \leq 1\}$ and $\Omega_l = \Omega_l^1 \cap \Omega_l^2$ be three convex sets. Let $\Omega = \Omega_1 \times ... \times \Omega_L$ be the set of the feasible solutions. Ω is convex as Cartesian product of convex sets. A quadratic penalty method replaces a constrained optimization problem by a series of unconstrained penalized problems. In the case of optimization problem (4.12) subject to (4.13), the following penalized unconstrained problem can be derived:

$$\underset{\mathbf{w}_1,\ldots,\mathbf{w}_L}{\operatorname{argmax}} h^\mu(\mathbf{w}_1,\ldots,\mathbf{w}_L) = \sum_{k,l=1}^{L} c_{kl}\ \mathrm{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right) - \sum_{k=1}^{L} \psi_k p_k\left(\mathbf{w}_k\right) - \frac{\mu}{2}\mathcal{C}_\Omega(\mathbf{w}_1,\ldots,\mathbf{w}_L), \quad (4.15)$$

where $\mathcal{C}_\Omega$ is a quadratic function penalizing the criterion when $(\mathbf{w}_1,\ldots,\mathbf{w}_L) \notin \Omega$. This penalty is nonzero when the constraints are violated and null otherwise. The penalty parameter $\mu \in \mathbb{R}_+^\star$ determines the severity of the penalty. In quadratic penalty methods, an iterative algorithm is used to solve (4.15) for any possible value of μ and the penalty parameter μ is gradually increased so that constraints (4.13) are more and more satisfied.

The following section focuses on designing an iterative algorithm in order to solve (4.15). This method was chosen among others because: (i) it allows dealing with unconstrained optimization problems usually easier than constrained ones, (ii) as mentioned in [Chi et al., 2013] this approach is particularly interesting when it is easy to project onto each separate set, but nontrivial to project onto their intersection.

4.4.1.2 Definition of the quadratic penalty function

Following [Chi et al., 2013], the quadratic penalty associated with the feasible constraints (4.13) is defined as:

$$\mathcal{C}_\Omega(\mathbf{w}_1, \ldots, \mathbf{w}_L) = \sum_{l=1}^{L} \sum_{j=1}^{2} \text{dist}(\mathbf{w}_l, \Omega_l^j)^2, \tag{4.16}$$

with:

$$\text{dist}(\mathbf{w}_l, \Omega_l^j)^2 = \inf_{\mathbf{x} \in \Omega_l^j} \|\mathbf{w}_l - \mathbf{x}\|_2^2 = \|\mathbf{w}_l - P_{\Omega_l^j}(\mathbf{w}_l)\|_2^2, \; l = 1, \ldots, L; \; j = 1, 2, \tag{4.17}$$

where $P_{\Omega_l^j}(\mathbf{w}_l)$ is the projection of $\mathbf{w}_l$ onto Ω_l^j.

4.4.1.3 Generalized Block Relaxation

The maximization of the optimization problem (4.15) over different parameter vectors $(\mathbf{w}_1, \ldots, \mathbf{w}_L)$, is approached by updating each of the parameter vectors in turn, keeping the others fixed as in a cyclic BCA framework [De Leeuw, 1994].

Let $\mathbf{w} = (\mathbf{w}_1; \ldots; \mathbf{w}_L)$ be a column vector and $f(\mathbf{w}) = \sum_{k,l=1}^{L} c_{kl}\, \text{g}\left(I^{-1}\mathbf{w}_k^\top \mathbf{X}_k^\top \mathbf{X}_l \mathbf{w}_l\right)$. Moreover, let $f_l(\tilde{\mathbf{w}}_l) = f(\mathbf{w}_1, ..., \mathbf{w}_{l-1}, \tilde{\mathbf{w}}_l, \mathbf{w}_{l+1}, \ldots, \mathbf{w}_L)$ be the function that solely depends on $\tilde{\mathbf{w}}_l \in \mathbb{R}^{J_l}$. In order to find an update $\hat{\mathbf{w}}_l \in \mathbb{R}^{J_l}$, cyclic block coordinate ascent suggests to consider the following optimization problem:

$$\hat{\mathbf{w}}_l = \underset{\tilde{\mathbf{w}}_l \in \mathbb{R}^{J_l}}{\text{argmax}}\; h_l^\mu(\tilde{\mathbf{w}}_l) = f_l(\tilde{\mathbf{w}}_l) - \psi_l p_l\left(\tilde{\mathbf{w}}_l\right) - \frac{\mu}{2} \sum_{j=1}^{2} \text{dist}(\tilde{\mathbf{w}}_l, \Omega_l^j)^2, \tag{4.18}$$

However, optimization problem (4.18) is still hard to solve, mainly due to the possible non-differentiability of function p_l. Thus, instead of proposing an update $\hat{\mathbf{w}}_l \in \mathbb{R}^{J_l}$ that maximizes $h_l^\mu(\tilde{\mathbf{w}}_l)$ over $\tilde{\mathbf{w}}_l \in \mathbb{R}^{J_l}$, $\hat{\mathbf{w}}_l$ is found such that it simply increases the objective function. As seen in Chapter 3 section 3.3.1, this principle is called generalized block relaxation by [De Leeuw, 1994] and constitutes the skeleton of the algorithm to be described.

The update that forces $h_l^\mu(\mathbf{w}_l) \leq h_l^\mu(\hat{\mathbf{w}}_l)$ is based on iterative minorization. This step is described in details thereafter. The method of iterative minorization for maximizing a function has been available in the literature for some time and has been rediscovered several times. Currently, it is perhaps best known under the name of MM-algorithms (minimization by majorization or maximization by minorization, [Hunter and Lange, 2004]).

4.4.1.4 The Maximization by Minorization Principle

Here, we consider $h_l^\mu(\tilde{\mathbf{w}}_l)$ that needs to be maximized over $\tilde{\mathbf{w}}_l \in \mathbb{R}^{J_l}$. The core of the method of iterative minorization is the use of a minorizing function called the surrogate function $\tilde{h}_l^\mu(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$ (with $\mathbf{w}_l$ being the previous estimate of $\tilde{\mathbf{w}}_l$) that has to satisfy the three following requirements:

(i). The minorizing surrogate function either touches or is smaller than the original function: $h_l^\mu(\tilde{\mathbf{w}}_l) \geq \tilde{h}_l^\mu(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$ (domination condition).

(ii). At the current estimate $\mathbf{w}_l$, the so-called supporting point, the two functions must touch: $h_l^\mu(\mathbf{w}_l) = \tilde{h}_l^\mu(\mathbf{w}_l|\mathbf{w}_l)$ (tangent condition).

(iii). $\tilde{h}_l^{\mu}(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$ should be a simple function in $\tilde{\mathbf{w}}_l$, often linear or quadratic which implies that the coordinates of its maximum $\hat{\mathbf{w}}_l$ should be easy to compute.

Consequently, we have $\tilde{h}_l^{\mu}(\hat{\mathbf{w}}_l|\mathbf{w}_l) \geq \tilde{h}_l^{\mu}(\mathbf{w}_l|\mathbf{w}_l)$. In addition, we have by construction, $h_l^{\mu}(\hat{\mathbf{w}}_l) \geq \tilde{h}_l^{\mu}(\hat{\mathbf{w}}_l|\mathbf{w}_l)$. Combining these inequalities gives the sandwich inequality:

$$h_l^{\mu}(\hat{\mathbf{w}}_l) \geq \tilde{h}_l^{\mu}(\hat{\mathbf{w}}_l|\mathbf{w}_l) \geq \tilde{h}_l^{\mu}(\mathbf{w}_l|\mathbf{w}_l) = h_l^{\mu}(\mathbf{w}_l), \tag{4.19}$$

showing that the update $\hat{\mathbf{w}}_l$ increases h_l^{μ} (or h_l^{μ} stays the same). This step constitutes a single iteration of a MM algorithm and in our case defines the update needed in Algorithm 7. Generalized block relaxation combined to the MM principal leads to the new optimization problem:

$$\hat{\mathbf{w}}_l = \underset{\tilde{\mathbf{w}}_l \in \mathbb{R}^{J_l}}{\operatorname{argmax}} \tilde{h}_l^{\mu}(\tilde{\mathbf{w}}_l|\mathbf{w}_l) := r_l(\mathbf{w}), \tag{4.20}$$

instead of optimization problem (4.18). Next section focuses on defining the surrogate function $\tilde{h}_l^{\mu}(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$.

4.4.1.5 Surrogate function

In this section, a minorizing surrogate function of $h_l^{\mu}(\tilde{\mathbf{w}}_l)$ anchored at $\mathbf{w}_l$ is derived. This minorizing surrogate function is defined as:

$$\tilde{h}_l^{\mu}(\tilde{\mathbf{w}}_l|\mathbf{w}_l) = \tilde{f}_l(\tilde{\mathbf{w}}_l|\mathbf{w}_l) - \psi_l \tilde{p}_l\left(\tilde{\mathbf{w}}_l|\mathbf{w}_l\right) - \frac{\mu}{2}\sum_{j=1}^{2} \widetilde{\operatorname{dist}}(\tilde{\mathbf{w}}_l, \Omega_l^j|\mathbf{w}_l)^2, \tag{4.21}$$

where $\tilde{f}_l(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$, $\tilde{p}_l\left(\tilde{\mathbf{w}}_l|\mathbf{w}_l\right)$ and $\widetilde{\operatorname{dist}}(\tilde{\mathbf{w}}_l, \Omega_l^j|\mathbf{w}_l)^2$ are the surrogate functions of the objective function $f_l(\tilde{\mathbf{w}}_l)$, the structured sparse penalty $p_l\left(\tilde{\mathbf{w}}_l\right)$ and the distance to the constraints $\operatorname{dist}(\tilde{\mathbf{w}}_l, \Omega_l^j)^2$ respectively. All these surrogates are anchored at $\mathbf{w}_l$. However, $\tilde{f}_l(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$ is a minorizing surrogate function while $\tilde{p}_l\left(\tilde{\mathbf{w}}_l|\mathbf{w}_l\right)$ and $\widetilde{\operatorname{dist}}(\tilde{\mathbf{w}}_l, \Omega_l^j|\mathbf{w}_l)^2$ are both majorizing surrogate functions as they are weighted by the opposite of a non-negative parameter. In the end, $\tilde{h}_l^{\mu}(\tilde{\mathbf{w}}_l|\mathbf{w}_l)$ is a minorizing surrogate function of $h_l^{\mu}(\tilde{\mathbf{w}}_l)$ anchored at $\mathbf{w}_l$.

Surrogate of the objective function. Using the multi-convexity of f and so the convexity of f_l, considering that a convex function lies above its linear approximation at $\mathbf{w}_l$ for any $\tilde{\mathbf{w}}_l \in \mathbb{R}^{J_l}$, the following inequality holds:

$$f_l(\tilde{\mathbf{w}}_l) \geq f_l(\mathbf{w}_l) + \nabla f_l(\mathbf{w}_l)^{\top}(\tilde{\mathbf{w}}_l - \mathbf{w}_l) := \tilde{f}_l(\tilde{\mathbf{w}}_l|\mathbf{w}_l) \tag{4.22}$$

with $\nabla f_l(\mathbf{w}_l) = \nabla_l f(\mathbf{w})$, the partial gradient of f with respect to $\mathbf{w}_l$ which is defined in (2.5) for example.

Surrogate of the distance to the constraints. The surrogate $\widetilde{\operatorname{dist}}(\tilde{\mathbf{w}}_l, \Omega_l^j|\mathbf{w}_l)^2$ is defined as in [Chi et al., 2013]:

$$\widetilde{\operatorname{dist}}(\tilde{\mathbf{w}}_l, \Omega_l^j|\mathbf{w}_l)^2 := \|\tilde{\mathbf{w}}_l - P_{\Omega_l^j}(\mathbf{w}_l)\|_2^2, \tag{4.23}$$

Indeed, requirements $(i-iii)$ mentioned in section 4.4.1.4 are satisfied by (4.23):

$$
\begin{aligned}
&(i). \quad \operatorname{dist}(\tilde{\mathbf{w}}_l, \Omega_l^j)^2 = \inf_{\mathbf{x}\in\Omega_l^j} \|\tilde{\mathbf{w}}_l - \mathbf{x}\|_2^2 \leq \|\tilde{\mathbf{w}}_l - P_{\Omega_l^j}(\mathbf{w}_l)\|_2^2 \\
&(ii). \quad \operatorname{dist}(\mathbf{w}_l, \Omega_l^j)^2 = \|\mathbf{w}_l - P_{\Omega_l^j}(\mathbf{w}_l)\|_2^2 \\
&(iii). \quad \operatorname*{argmin}_{\tilde{\mathbf{w}}_l\in\mathbb{R}^{J_l}} \|\tilde{\mathbf{w}}_l - P_{\Omega_l^j}(\mathbf{w}_l)\|_2^2 = P_{\Omega_l^j}(\mathbf{w}_l).
\end{aligned}
$$

Surrogate of the structured sparse penalty. The investigation is limited to the case where it is possible to find a quadratic and convex surrogate function for every penalty p_k. Further cases are mentioned in the discussion. As a matter of fact, the following general majorizing surrogate function can be defined:

$$
\tilde{p}_l\left(\tilde{\mathbf{w}}_l|\mathbf{w}_l\right) := p_l\left(\mathbf{w}_l\right) + \mathbf{b}_l^\top\left(\tilde{\mathbf{w}}_l - \mathbf{w}_l\right) + \left(\tilde{\mathbf{w}}_l - \mathbf{w}_l\right)^\top \mathbf{A}_l\left(\tilde{\mathbf{w}}_l - \mathbf{w}_l\right), \tag{4.24}
$$

where $\mathbf{b}_l \in \mathbb{R}^{J_l}$ and $\mathbf{A}_l \in \mathcal{S}_+^{J_l}$, the set of symmetric positive semidefinite matrices of size $J_l \times J_l$. In section 4.4.2.2 and in appendix C section C.2, examples of such quadratic and convex surrogate functions are given.

4.4.1.6 The Structured SGCCA algorithm

The resulting surrogate defined in (4.21) is strictly concave. Indeed, (4.22) is linear while (4.23) is strictly convex, (4.24) convex and both of them are weighted by the opposite of a non-negative parameter. The update $\hat{\mathbf{w}}_l \in \mathbb{R}^{J_l}$ is thus defined as the solution of a maximization problem introduced in (4.20) where the criterion is strictly concave. Hence this update is unique. It is obtained by finding $\hat{\mathbf{w}}_l \in \mathbb{R}^{J_l}$ such that $\nabla_{\tilde{\mathbf{w}}_l}\tilde{h}_l^\mu(\hat{\mathbf{w}}_l|\mathbf{w}_l) = \mathbf{0}_{J_l}$. It can be shown that this solution is:

$$
\hat{\mathbf{w}}_l = r_l(\mathbf{w}) = \frac{1}{2}\left[\mu\mathbf{I}_{J_l} + \psi_l\mathbf{A}_l\right]^{-1}\left(\nabla_l f(\mathbf{w}) + \psi_l\left(2\mathbf{A}_l\mathbf{w}_l - \mathbf{b}_l\right) + \mu\sum_{j=1}^{2} P_{\Omega_l^j}(\mathbf{w}_l)\right), \tag{4.25}
$$

where $[\mu\mathbf{I}_{J_l} + \psi_l\mathbf{A}_l]$ is invertible as $\mathbf{A}_l$ is symmetric positive semidefinite.

Based on update (4.25), the entire Structured SGCCA algorithm is presented in Algorithm 7.

Following the guidelines of [Chi et al., 2013], the penalization parameter μ is initialized at $\mu_0 = 1$ and updated such that $\mu_{p+1} = 2\mu_p + 1$.

4.4.2 Experiments

In this section, we compare the performances of RGCCA, SGCCA and structured SGCCA referred as MM_SGCCA. We tried hard to provide also a comparison with the structured SGCCA method presented in [Löfstedt et al., 2016] which is based on the smoothing framework of [Nesterov, 2004] and proximity operators. However, these results are not presented in the core of this manuscript as this proximal algorithm always reached the maximum number of iterations allowed. Though, the results of this proximal method are postponed in the Appendix C section C.3.

Algorithm 7 Maximization by Minorization (MM) algorithm for Structured SGCCA

Data: $\mathbf{X}_1, \ldots, \mathbf{X}_L$, $\psi_1, \ldots, \psi_L$, $s_1, \ldots, s_L$, $\mathbf{M}_1, \ldots, \mathbf{M}_L$, g, ε_1, ε_2, $\mathbf{C}$
Result: $\mathbf{w}_1^s, \ldots, \mathbf{w}_L^s$ (approximate solution of (4.12) subject to (4.13))
Initialization: choose random normalized $\mathbf{w}_l^0, l = 1, \ldots, L$; $\mu_0 > 0$;
$s = 0$; $p = 0$;
repeat
  $(\mathbf{w}_1^0, \ldots, \mathbf{w}_L^0) \leftarrow (\mathbf{w}_1^s, \ldots, \mathbf{w}_L^s)$
  $s = 0$
  repeat
    for $l = 1$ **to** L **do**

$$\mathbf{w}_l^{s+1} = \frac{1}{2}\left[\mu_p \mathbf{I}_{J_l} + \psi_l \mathbf{A}_l^s\right]^{-1} \left(\nabla_l^s f + \psi_l \left(2\mathbf{A}_l^s \mathbf{w}_l^s - \mathbf{b}_l^s\right) + \mu_p \sum_{j=1}^{2} P_{\Omega_l^j}(\mathbf{w}_l^s) \right) \tag{4.26}$$

    end for
    $s \leftarrow s + 1$
  until $\|(\mathbf{w}_1^{s+1}, \ldots, \mathbf{w}_L^{s+1}) - (\mathbf{w}_1^s, \ldots, \mathbf{w}_L^s)\|_2 < \varepsilon_1$
  Choose new penalty parameter $\mu_{p+1} > \mu_p$
  $p \leftarrow p + 1$
until $\|(\mathbf{w}_1^s, \ldots, \mathbf{w}_L^s) - (\mathbf{w}_1^0, \ldots, \mathbf{w}_L^0)\|_2 < \varepsilon_2$

where $\nabla_l^s f = \mathbf{X}_l^\top \left(2\sum_{k=1}^{l-1} c_{lk}\ \mathrm{g}'(\mathbf{w}_l^{s\top}\mathbf{X}_l^\top \mathbf{X}_k \mathbf{w}_k^{s+1})\mathbf{X}_k \mathbf{w}_k^{s+1} + 2\sum_{k=l}^{L} c_{lk}\ \mathrm{g}'(\mathbf{w}_l^{s\top}\mathbf{X}_l^\top \mathbf{X}_k \mathbf{w}_k^{s})\mathbf{X}_k \mathbf{w}_k^{s}\right)$
Moreover, the symmetric positive semi-definite matrix $\mathbf{A}_l^s$ of size J_l and the column vector $\mathbf{b}_l^s$ of size J_l are defined such that $\tilde{p}_l(\mathbf{w}_l|\mathbf{w}_l^s) := p_l(\mathbf{w}_l^s) + \mathbf{b}_l^{s\top}(\mathbf{w}_l - \mathbf{w}_l^s) + (\mathbf{w}_l - \mathbf{w}_l^s)^\top \mathbf{A}_l^s (\mathbf{w}_l - \mathbf{w}_l^s)$ is a majorizing quadratic and convex surrogate function of p_l anchored at $\mathbf{w}_l^s$ (see section 4.4.2.2 and appendix C section C.2 for examples of such quadratic and convex surrogate functions).

4.4.2.1 Data Generation

For this experiment, we consider 2 blocks of dimensions $I = 50$, $J_1 = 150$ and $J_2 = 100$. Each block is generated according to the following matrix model:

$$\mathbf{X}_l = \eta \mathbf{y}_l \mathbf{w}_l^\top + \frac{\|\mathbf{y}_l \mathbf{w}_l^\top\|_F}{\|\mathbf{E}_l\|_F} \mathbf{E}_l,\ l = 1, 2$$

where each row of $[\mathbf{y}_1, \mathbf{y}_2] \in \mathbb{R}^{I \times 2}$ follows a multivariate normal distribution $\mathcal{N}(0, \mathbf{\Sigma})$, such that $(\mathbf{\Sigma})_{11} = (\mathbf{\Sigma})_{22} = 1$ and $(\mathbf{\Sigma})_{12} = (\mathbf{\Sigma})_{21} = 0.9$ ensuring a correlation of 0.9 between $\mathbf{y}_1$ and $\mathbf{y}_2$.

Moreover, $\mathbf{w}_1 \in \mathbb{R}^{J_1}$ and $\mathbf{w}_2 \in \mathbb{R}^{J_2}$ are generated with a pre-defined structure as follows:

- $\mathbf{w}_1$ is composed of multiple steps whose levels are defined randomly from the uniform distribution between -0.5 and 0.5. The width of each level is randomly chosen between 5 and 10. Then the elements of $\mathbf{w}_1$ are sorted in an increasing order, $\mathbf{w}_1$ is centered, normalized and soft-thresholded such that $\|\mathbf{w}_1\|_1 = 0.64\sqrt{J_1}$. This procedure allows to define $\mathbf{w}_1$ as a succession of plateaux of increasing level, where one level is null. This kind of structure is particularly well recovered through a Total Variation penalty (see next section).

- $\mathbf{w}_2$ is divided into 7 disjoint groups. All the elements of a group are set to specific constants. These constants are equal to 0 for groups 1, 5 and 7. $\mathbf{w}_2$ is also ℓ_2-normalized. This structure

is particularly well recovered through a group-LASSO penalty (see next section).

The noise matrix $\mathbf{E}_l \in \mathbb{R}^{I \times J_l}$ is defined such that its entries are drawn from a standardized normal distribution. The Signal to Noise Ratio (SNR) is equal to $20log_{10}(\eta)$ which enables η to drive the SNR.

4.4.2.2 Penalties and Surrogate functions

Considering the predefined within block structures, $\mathbf{w}_1$ was subject to a Total Variation (TV) penalty while $\mathbf{w}_2$ to a group-LASSO (GL) penalty.

The TV penalty, first introduced in [Rudin et al., 1992], is widely used as a tool in image denoising and restoration. It accounts for the spatial structure of images by encoding piecewise smoothness and enabling the recovery of homogeneous regions separated by sharp boundaries [Pierrefeu, 2018]. The TV penalty can be formulated as follows:

$$p_1(\mathbf{w}_1) = \sum_{j=1}^{J_l-1} |(\mathbf{w}_1)_{j+1} - (\mathbf{w}_1)_j| = \|\mathbf{D}_1\mathbf{w}_1\|_1, \tag{4.27}$$

where $\mathbf{D}_1 \in \mathbb{R}^{J_l-1 \times J_l}$ is defined such that $(\mathbf{D}_1)_{jj} = -1$, $(\mathbf{D}_1)_{jj+1} = 1$ and 0 elsewhere. A surrogate function of the TV penalty can be defined as (see appendix C section C.2.3 for more details):

$$\widetilde{p_1}(\mathbf{w}_1|\mathbf{w}_1^s) := \frac{1}{2}p_1(\mathbf{w}_1^s) + \frac{1}{2}\mathbf{w}_1^\top\mathbf{D}_1^\top\mathbf{\Delta}_1^s\mathbf{D}_1\mathbf{w}_1, \tag{4.28}$$

where $\mathbf{\Delta}_1^s$ is a diagonal matrix of size J_1-1 such that $(\mathbf{\Delta}_1^s)_{jj} = \frac{1}{|(\mathbf{w}_1^s)_{j+1}-(\mathbf{w}_1^s)_j|}$. This surrogate function is well defined whenever $(\mathbf{w}_1^s)_{j+1} \neq (\mathbf{w}_1^s)_j$.

The non-overlapping group-LASSO, first introduced in [Yuan and Lin, 2006], is the $\ell_{1,2}$-mixed norm. By introducing a partition $\mathcal{G}$ of $[\![1; J]\!]$ (meaning all the groups are disjoint and $\underset{g\in\mathcal{G}}{\cup} g = [\![1; J]\!]$), the group-LASSO penalty is defined as:

$$p_2(\mathbf{w}_2) = \sum_{g\in\mathcal{G}} \|(\mathbf{w}_2)_{i_g}\|_2, \tag{4.29}$$

where $(\mathbf{w}_2)_{i_g}$ is a subvector of $\mathbf{w}_2$ containing only the elements of the g^{th} group of $\mathcal{G}$. The group-LASSO penalty acts like the LASSO at the group level and an entire group of variables may drop out of jointly. A surrogate function of the group-LASSO penalty can be defined as (see appendix C section C.2.2 for more details):

$$\widetilde{p_2}(\mathbf{w}_2|\mathbf{w}_2^s) := \frac{1}{2}p_2(\mathbf{w}_2^s) + \frac{1}{2}\mathbf{w}_2^\top\mathbf{\Delta}_2^s\mathbf{w}_2, \tag{4.30}$$

where $\mathbf{\Delta}_2^s$ is a diagonal matrix of size J_2 such that $(\mathbf{\Delta}_2^s)_{jj} = \frac{1}{\|(\mathbf{w}_2^s)_{j_g}\|_2}$ if variable j is in group g. This surrogate function is defined $\forall \mathbf{w}_2^s \in \mathbb{R}^{J_2}$ such that $\forall g \in \mathcal{G},\ \|(\mathbf{w}_2^s)_{j_g}\|_2 \neq 0$.

4.4.2.3 Constraints and parameters

For all the methods, the design matrix $\mathbf{C}$ is defined such that $c_{12} = c_{21} = 1$ and $c_{11} = c_{22} = 0$ and the function g is set to the square function. Concerning the ℓ_2-norm constraints, for the two blocks and

all three methods, $\mathbf{M}_l = \mathrm{I}_{J_l}$. So, for MM_SGCCA, $\Omega_l^2 = \{\mathbf{w}_l \in \mathbb{R}^{J_l}; \mathbf{w}_l^\top \mathbf{w}_l \leq 1\},\ l = 1, 2$. An ℓ_1-norm constraint is imposed for the two blocks in SGCCA and only on the first block for MM_SGCCA. For RGCCA, no sparse constraint is imposed.

In the end, in this setting, RGCCA is parameter free. Two sparse parameters are tuned for SGCCA: s_1 and s_2. Three parameters are tuned for MM_SGCCA: s_1, ψ_1 (for the TV penalty) and ψ_2 (for the group-LASSO penalty).

In order to evaluate each method, several measurements a used. First, similarly to sections 2.4 and 3.3.3, a measurement of accuracy (ACC) is introduced:

$$ACC = \frac{1}{L} \sum_{l=1}^{L} |\hat{\mathbf{w}}_l^\top \mathbf{w}_l|, \tag{4.31}$$

where $\hat{\mathbf{w}}_l \in \mathbb{R}^{J_l}$ is the estimate of the true block-weight vector $\mathbf{w}_l$. Moreover, the Cohen's kappa [Cohen, 1960] is used to evaluate the support recovery. This indicator is computed as:

$$\kappa_l = \kappa\left(\mathbb{1}_{\hat{\mathbf{w}}_l}, \mathbb{1}_{\mathbf{w}_l}\right),\ l = 1, 2, \tag{4.32}$$

where $(\mathbb{1}_{\mathbf{w}})_i = 1$ if $|(\mathbf{w})_i| = 0$, and 0 otherwise. An element of a vector is considered as null if its absolute value is below the machine threshold (2.2×10^{-16}).

Parameters are tuned in order to maximize a weighted sum of ACC, κ_1 and κ_2.

All methods are initialized with the SVD of each block and the same stopping criterion is defined with $\varepsilon = 10^{-8}$. For MM_SGCCA, two stopping criteria have to be defined, one for the inner «for loop» ($\varepsilon_1 = 10^{-4}$) and one for the whole algorithm ($\varepsilon_2 = 10^{-8}$).

4.4.2.4 Computational considerations

Based on the surrogate functions defined in equations (4.28) and (4.30), the updates for a specific μ_p in Algorithm 7 becomes:

$$\mathbf{w}_1^{s+1} = \frac{1}{2}\left[\mu_p \mathbf{I}_{J_1} + \frac{\psi_1}{2}\mathbf{D}_1^\top \boldsymbol{\Delta}_1^s \mathbf{D}_1\right]^{-1} \left(\nabla_1^s f\left(\mathbf{w}_1^s, \mathbf{w}_2^s\right) + \mu_p \sum_{j=1}^{2} P_{\Omega_1^j}(\mathbf{w}_1^s)\right) \tag{4.33}$$

$$\mathbf{w}_2^{s+1} = \left[\mu_p \mathbf{I}_{J_2} + \psi_2 \boldsymbol{\Delta}_2^s\right]^{-1} \left(\nabla_2^s f\left(\mathbf{w}_1^{s+1}, \mathbf{w}_2^s\right) + \mu_p P_{\Omega_2^2}(\mathbf{w}_2^s)\right) \tag{4.34}$$

where $\boldsymbol{\Delta}_1^s$ and $\boldsymbol{\Delta}_2^s$ can be ill-conditioned (see their definition in section 4.4.2.2). In [Figueiredo et al., 2006, Selesnick, 2012], the authors overcome this issue using the matrix inversion lemma. With such lemma, these updates becomes:

$$\mathbf{w}_1^{s+1} = \frac{1}{2\mu_p}\left[\mathrm{I}_{J_1} + \mathbf{D}_1^\top \left(\frac{2\mu_p}{\psi_1}\boldsymbol{\Delta}_1^{s^{-1}} + \mathbf{D}_1\mathbf{D}_1^\top\right)^{-1}\mathbf{D}_1\right] \left(\nabla_1^s f\left(\mathbf{w}_1^s, \mathbf{w}_2^s\right) + \mu_p \sum_{j=1}^{2} P_{\Omega_1^j}(\mathbf{w}_1^s)\right) \tag{4.35}$$

$$\mathbf{w}_2^{s+1} = \frac{1}{\mu_p}\left[\mathrm{I}_{J_2} + \left(\frac{\mu_p}{\psi_2}\boldsymbol{\Delta}_2^{s^{-1}} + \mathrm{I}_{J_2}\right)^{-1}\right] \left(\nabla_2^s f\left(\mathbf{w}_1^{s+1}, \mathbf{w}_2^s\right) + \mu_p P_{\Omega_2^2}(\mathbf{w}_2^s)\right). \tag{4.36}$$

Hence, even if $\boldsymbol{\Delta}_1^s$ or $\boldsymbol{\Delta}_2^s$ are ill-conditioned, the algorithm is not affected as they are always inverted first.

4.4.2.5 Results

For each value of $\eta \in \{0.5, 1, 2\}$, 40 datasets are generated. RGCCA, SGCCA and MM_SGCCA are applied on each dataset. For each algorithm and value of η, mean and standard deviation of the ACC (defined in (4.31)) and of $\kappa_l,\ l = 1, 2$ (defined in (4.32)) are computed through datasets and reported in table 4.1. The median of the number of iterations of each algorithm, their interquartile range and the mean and standard deviation of the execution time are also presented in this table. On Figure 4.4-5 and 4.4-6, the weight vectors for the first and second block respectively are shown for each method.

In table 4.1, all methods performed quite similarly in regard of the ACC measurement, with an increase of the ACC with the SNR value. However, differences are observed concerning κ_1 and κ_2 that characterize how well null elements are recovered. First, their value is not reported for RGCCA due to the absence of sparse constraints in its optimization problem. Then, for MM_SGCCA, κ_1 and κ_2 are always higher than for SGCCA. MM_SGCCA even perfectly estimates the null elements when $\eta = 2$. This is a relief, the new algorithm proposed to handle structured sparse penalties indeed provides better results than SGCCA.

Still in table 4.1, from $\eta = 0.5$ to 2 for MM_SGCCA, the number of iterations and the execution time both decreases by a factor of 5 approximately. However, when $\eta = 2$, MM_SGCCA is still 10^3 times longer than SGCCA or RGCCA. For SGCCA and RGCCA, the number of iterations and the execution time are almost not affected by the SNR.

In figures 4.4-5 and 4.4-6, the rows are associated with a specific value of η and the columns with a specific method. It is interesting to visualize how the estimations evolve with SNR and methods. For example, only MM_SGCCA manages to catch almost all the right null elements for $\eta = 0.5$ for the two blocks. SGCCA only reaches this goal when $\eta = 2$. As mentioned earlier, RGCCA cannot perform this estimation as no sparse constraint is imposed.

Table 4.1 – For each value of $\eta \in \{0.5, 1, 2\}$, 40 datasets were generated. For each dataset, three methods are compared: RGCCA, SGCCA and MM_SGCCA. For each algorithm, the mean and standard deviation (std) of the ACC (defined in (4.31)) and of $\kappa_l,\ l = 1, 2$ (defined in (4.32)), the median (MD) of the number of iterations (Iter), their interquartile range (IQR) and the mean and standard deviation of the execution time (Time) are reported.

SNR	Algorithm	ACC (mean ± std)	κ_1 (mean ± std)	κ_2 (mean ± std)	Iter (MD - IQR)	Time (s) (mean ± std)
$\eta = 0.5$	RGCCA	0.952 ± 4e-3	\	\	6 - 0	0.03 ± 1e-2
	SGCCA	0.935 ± 5e-3	0.77 ± 4e-2	0.82 ± 5e-2	6 - 0	0.042 ± 7e-3
	MM_SGCCA	0.973 ± 6e-3	0.9 ± 0.1	1 ± 0	2180 - 1055	127 ± 35
$\eta = 1$	RGCCA	0.984 ± 2e-3	\	\	4 - 0	0.03 ± 1e-2
	SGCCA	0.981 ± 1e-3	0.90 ± 3e-2	0.97 ± 3e-2	4 - 0	0.033 ± 6e-3
	MM_SGCCA	0.9928 ± 7e-4	0.99 ± 1e-2	1 ± 0	740 - 127	46 ± 6
$\eta = 2$	RGCCA	0.992 ± 2e-3	\	\	4 - 0	0.04 ± 2e-2
	SGCCA	0.9944 ± 2e-4	0.99 ± 1e-2	1 ± 0	3 - 1	0.029 ± 6e-3
	MM_SGCCA	0.9977 ± 2e-4	1 ± 0	1 ± 0	426 - 52	28 ± 3

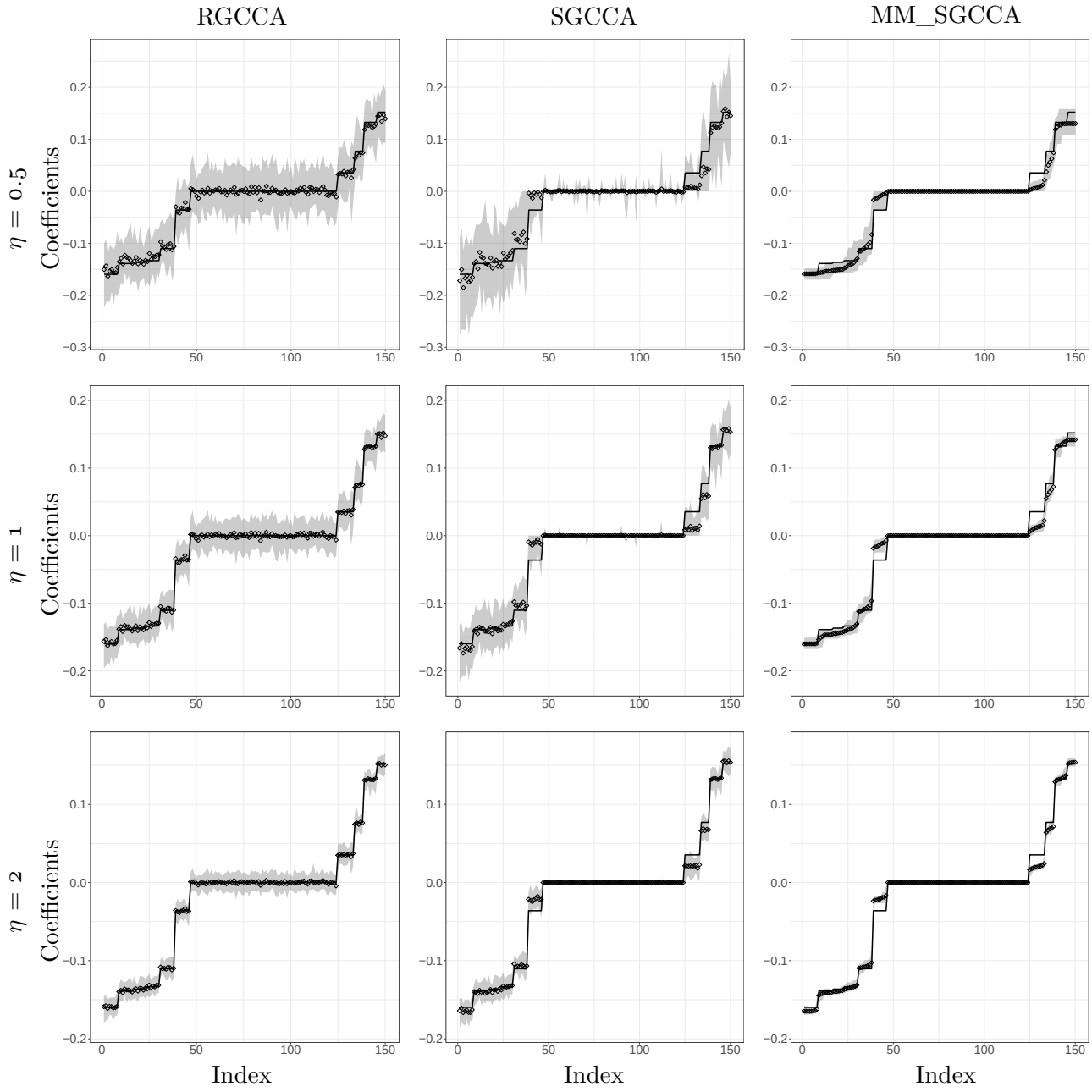


Figure 4.4-5 – Continuous lines correspond to the first block weight vector and points to its estimates obtained with RGCCA, SGCCA and MM_SGCCA. Each row is associated with a specific value of η (0.5, 1 and 2) arranged in increasing order and each column with a method. 4 worst runs for each method according to a weighted sum of ACC, κ_1 and κ_2 were removed. For each element of this estimated vector, grey areas stand for the min and max of its distribution based on the 36 remaining runs.

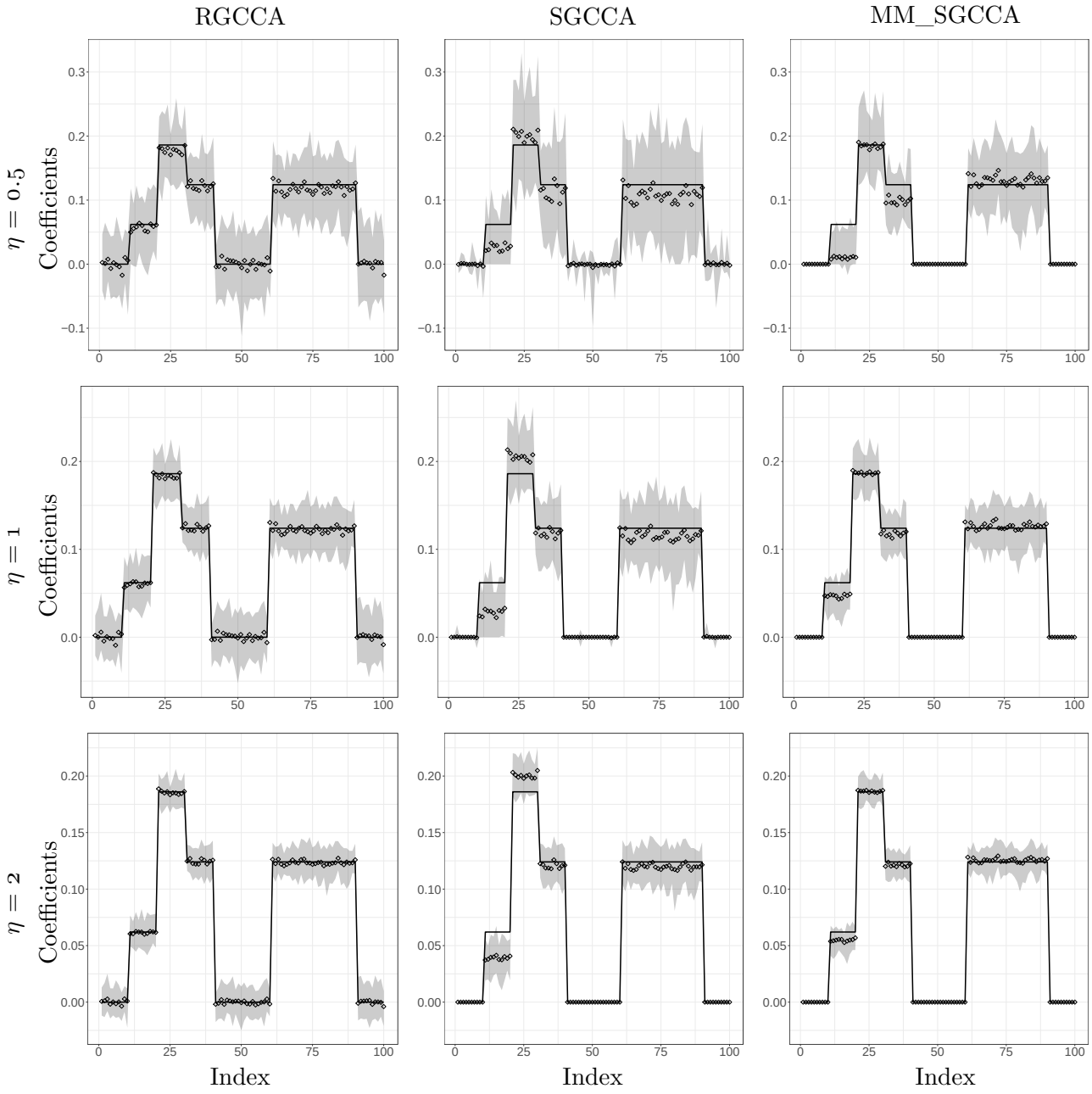


Figure 4.4-6 – Continuous lines correspond to the second block weight vector and points to its estimates obtained with RGCCA, SGCCA and MM_SGCCA. Each row is associated with a specific value of η (0.5, 1 and 2) arranged in increasing order and each column with a method. 4 worst runs for each method according to a weighted sum of ACC, κ_1 and κ_2 were removed. For each element of this estimated vector, grey areas stand for the min and max of its distribution based on the 36 remaining runs.

4.5 Conclusion and Discussion

In this Chapter, Fast_L1_L2 was proposed as a new procedure to solve efficiently the SGCCA optimization problem. This procedure is faster than binary search and POCS. Under mild conditions, we have shown that the corresponding SGCCA algorithm is globally convergent.

Moreover, a structured sparse version of RGCCA was presented. Structure sparse penalties allow to jointly select variables that are known to have intra-block interactions. A previous structured sparse version of RGCCA was proposed in [Löfstedt et al., 2016] in the framework of proximal algorithm. However, the function g was limited to the identity and sparse inducing penalties had to be convex. We propose a new algorithm that enables g to be any continuously differentiable convex function. In addition, as long as it is possible to find a quadratic and convex surrogate function of the sparse penalty, this procedure can be used. This algorithm relies on distance majorization, generalized BCA and MM principle. This approach was compared favorably with RGCCA and SGCCA. Comparison with other structured sparse CCA algorithms as [Chen and Liu, 2012, Chen et al., 2012a] is under way.

Here, only the construction of first component was addressed. Similarly to section 2.2.4 a deflation procedure can be used in structured SGCCA in order to obtain orthogonal higher-level components. This procedure consists in replacing a block $\mathbf{X}_l$ by the residual $\mathbf{X}_l^{(1)} = \mathbf{X}_l - \mathbf{y}_l^{(1)} \left(\mathbf{y}_l^{(1)^\top} \mathbf{y}_l^{(1)} \right)^{-1} \mathbf{y}_l^{(1)^\top} \mathbf{X}_l$ related to the regression of $\mathbf{X}_l$ on the first-stage block component $\mathbf{y}_l^{(1)}$.

As explained in section 4.4.2.2, if $\exists j$ such that $(\mathbf{w}_1^s)_{j+1} = (\mathbf{w}_1^s)_j$ or if $\exists g \in \mathcal{G}$ such that $\|(\mathbf{w}_2^s)_{j_g}\|_2 = 0$ then $\mathbf{\Delta}_1^s$ or $\mathbf{\Delta}_2^s$ are not defined. This is troublesome when sparse solution is sought. Indeed, no quadratic surrogate function of the scalar absolute value anchored at 0 can be found [de Leeuw and Lange, 2009]. As both group-LASSO and TV surrogate functions are derived from the surrogate function of the scalar absolute value (see appendix C), they inherit this issue. One solution is to consider a perturbation of the original penalties as discussed in [De Leeuw, 2018, Du et al., 2017, Yu et al., 2015]. This perturbation introduces a smoothing parameter in order to make the penalty everywhere differentiable. The MM principle is still needed as the perturbation is hard to minimize. Work in progress aims at integrating these perturbations inside the procedure to handle this issue. In practice, this issue is dealt with the matrix inversion lemma presented in section 4.4.2.4.

The general optimization framework presented in Chapter 1 cannot be used as such to demonstrate the global convergence of the Structured SGCCA algorithm. Indeed, the structure of this algorithm is different from the others as it is composed of an "inner loop", where the parameter μ that regulates the amount of the penalty associated with the feasible solutions is fixed, and an "outer loop", where μ is gradually increased to enforce the solution to be in the feasible set. The convergence of the "inner loop" can still be studied with the Meyer's theory. However, it needs to be shown that the sequences generated by this "inner loop" lies in a compact set, which is verified when the objective function is coercive. This approach is similar to the work undertaken in [Chi et al., 2013] to prove the global convergence of the distance majorization algorithm. In this article, convergence of the "outer loop" is also studied. Work in progress includes adapting this convergence study to our Structured SGCCA algorithm.

Here, as long as it is possible to find a majorizing surrogate function of p_k satisfying points $(i - iii)$

defined in section 4.4.1.4, Algorithm 7 is still valid. We limit the investigation to quadratic and convex surrogate function for every penalty p_k. We made this choice as quadratic and convex surrogates are the most common ones (see appendix C where surrogate functions of multiple structured sparse penalties are presented).

This project is still ongoing and not yet evaluated on a real dataset. To get an hint on its possibilities, the reader is referred to [Guigui et al., 2019] which applies the structured SGCCA algorithm proposed in [Löfstedt et al., 2016] on the Alzheimer's Disease Neuroimaging Initiative (ADNI), an open dataset on Alzheimer's disease. In this article, the interactions among three blocks are studied in a prediction context.

* * *
* *
*

Chapter 5

Multiblock and/or Multiway data studies

Chapter Outline

The methods and principles contained in this chapter were presented at national/international conferences or international journals and are also the subject of a publication in preparation:

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Laurent Le Brusquet, Arthur Tenenhaus. *Multiway Generalized Canonical Correlation Analysis.* Chimiométrie XIX, Paris, France, 2018. Oral.

Slim Karkar, **Arnaud Gloaguen**, Yann Le Guen, Morgane Pierre-Jean, Claire Dandine-Roulland, Edith Le Floch, Cathy Philippe, Arthur Tenenhaus and Vincent Frouin. *Multivariate Haplotype Analysis Of 96 Sulci Opening For 15,612 UK-Biobank Sujects.* Proceedings of the IEEE International Symposium of Biomedical Imaging, Venice, Italy, 2019. Poster.

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Giulia Gennari, Ghislaine Dehaene-Lambertz, Laurent Le Brusquet, Arthur Tenenhaus. *Multiway Generalized Canonical Correlation Analysis.* Biostatistics, 2020.

Fabien Girka, Pierrick Chevalier, **Arnaud Gloaguen**, Laurent Le Brusquet, Arthur Tenenhaus. *Rank-R Multiway Logistic Regression.* 52ème Journées de Statistique (JDS), France, 2020. Accepted.

The usefulness of RGCCA and MGCCA is shown over several applications. The first study investigates the influence of genetics on the normal aging brain from the United Kingdom Biobank (UKB) cohort. The second one is an imaging-genetic study on the Alzheimer's disease Neuroimaging Initiative (ADNI) that aims at understanding some mechanisms of the disease through several modalities (Genetics, Transcriptomics, longitudinal MRI, Clinical factors). The third study aims at analyzing the efficiency of a moisturizer from Raman microscopy. The two last studies aim at identifying brain areas implicated in the process of discrimination between close syllables in two- to three-month-old human infants from Electroencephalography (EEG).

5.1 A multivariate haplotype approach in imaging-genetics on the UK Biobank

At the heart of this work lies the United Kingdom Biobank (UKB) cohort. The UKB is a health research resource that aims at improving the prevention, diagnosis and treatment of a wide range of illnesses. Between the years 2006 and 2010, about 500.000 people aged between 45 and 73 years old, were recruited in the general population across Great Britain. The UKB cohort provides genotype data (800.000 SNPs), the digitized medical file (with daily updates) and even high quality brain imaging data (anatomical, functional and diffusion MRIs) for 20.000 of them, with a final goal of 100.000.

Imaging genetic studies of large general population cohorts such as UKB enable to assess the range of normal variations in brain structures. In this section, a two-block study is performed with RGCCA in order to investigate the influence of genetics on the normal aging brain.

5.1.1 Background in Imaging-genetics

All the concepts evoked in this section are rather basics and do not pretend to offer a general knowledge on genetics or neuroimaging but rather a short introduction in order to understand the following work.

5.1.1.1 Genetics

The DeoxyriboNucleic Acid (DNA). All human cells, except red blood cells, have, within the nucleus, a macro-molecule called the DeoxyriboNucleic Acid (DNA). This DNA bears the genetic information allowing the development, functioning, growth and reproduction processes. It is structured in chromosomes, 23 pairs in number, except for germ cells, offering a compact way of storing the genetic information. Each chromosome of a pair being inherited from either the mother or the father.

For a single chromosome, the DNA is composed of two strands. Each strand is a sequence of nucleotides. Four nucleotides are possible: cytosine (C), guanine (G), adenine (A) or thymine (T). The two strands of the DNA are linked by their nucleotides through hydrogen bonds. Only links between the nucleotides A and T or C and G are possible. We commonly talk about pairs of nucleotides. The human genome is composed of approximately 3.2 billions of pairs of nucleotides, also noted 3.2Gbp for Giga base pairs.

Genes, Alleles and SNPs. A gene is a fragment of DNA, composed of alternating exons and introns. Only exons are translated into proteins, according to the genetic code. Triplets of nucleotides

are called codons. Each codon codes for an amino acid, a sequence of amino acids is a protein. The genetic code is redundant, meaning that each amino acid can be coded by several codons, allowing some variability in the human genome. In the end, exons are coding parts of a gene while introns are non-coding parts. Variability occurs in non coding parts of the genome as well. The human genome is composed of approximately 23.000 genes. Flanking regions up-and down-stream of a gene do not code for proteins but can interact with proteins playing a role in the expression regulation of that gene.

In humans, each gene occurs twice: once on the maternal chromosome and once on the paternal one, possibly in two different versions. A version of a gene is called an allele. When the two alleles of a gene are identical, we say that the individual is homozygous for this gene. In opposite, when the alleles are different, the individual is heterozygous for this gene. When the terms allele/homozygous/heterozygous apply to a single nucleotide, it is called single nucleotide polymorphism (SNP). SNPs are part of the genome variability and can occur in coding and non coding part of the genome. The genotype of an individual is defined by the nucleotides for a pair of alleles. The phenotype is the set of behavioral, physiological, morphological and cellular features that can be observed or measured at a macroscopic scale. On the 3.2Gbp of the human DNA, only 1.5% constitutes the exome (the all set of exons of an individual) and around 80% is non-coding but is considered to be functional.

Genome Wide Association Studies (GWAS). A Genome Wide Association Studies (GWAS) aim at studying, in a homogeneous population called an ancestry, the correlation between genotypes and one or several phenotypes. Genotypes are usually evaluated at SNPs loci. As both alleles are considered, 3 of the following different states can be observed: A/T-A/T, C/G-C/G and A-T/C-G. The following conventional encoding can be encountered: 0, 1 or 2. 0 means that the participant is homozygous for the major allele at the considered locus, that is to say the most common allele in the population under study, 2 means the participant is homozygous for the minor allele, that is to say the less common allele, and 1 means heterozygous.

GWAS mainly investigate these correlations through univariate methods. This means that the influence of a SNP on the chosen phenotype is considered independently regarding the other SNPs. When the phenotype is quantitative (cortical thickness, blood cholesterol, etc), which is mostly the case in imaging-genetics, this influence is essentially studied through a linear model called an additive model in genetics. It is called additive because if a minor allele is present twice, the phenotype variation has to be twice higher or lower. From this additive model, a p-value can be derived. This p-value indicates if the linear coefficient is effectively different from zero. This model is then repeated with each SNP. As multiple tests are undertaken, we have to control the Family Wise Error Rate (FWER). The main correction employed in genetics is the Bonferroni correction. It consists in dividing the p-values by the number of tests undertaken. As the significant threshold of a single p-value is usually 5%, we can convert this correction to a new threshold for all the tests. In GWAS, the number of genetic variations considered is often around a million, which leads to a threshold of $5e-8$.

Haplotypes. SNP phasing aims at determining variants that are inherited together either from the mother or from the father. The genome of an individual is not strictly a combination between half of the genome of the mother and half of the genome of the father. This is due to the genetic recombination. In humans, genetic recombinations occur during the meiosis (the cell division that

generates the gametes). The main recombination that is of interest here is called the chromosomal crossover. During the meiosis, two chromosomes of the same pair can meet and thus exchange pieces of DNA sequences. Then, on a chromosome, we have what is called recombination hotspots, defined by a high recombination rate. The recombination rate is closely related to the linkage disequilibrium (LD). LD is the non-random association of alleles at different loci in a given population. SNPs are said to be in LD when the frequency of association of their different alleles is higher or lower than what would be expected if the loci were independent and associated randomly. Through the LD and the recombination rate, a genetic distance between loci can be defined. It is measured in centimorgan (cM). The lower the genetic distance between two loci, the stronger their link and the higher the probability that they were inherited together. Usually, between two recombination hotspots, the genetic distances between loci is low, meaning that the probability that this piece of sequence of DNA was inherited as it is high. For this kind of DNA chunks, we talk about haplotypic blocks. Thus, an haplotype is a set of genotyped SNPs located on a chromosome that are usually inherited together. This definition is not strict as it is still the topic of discussions. In the work considered here, we define haplotypes as an allele of a specific set of SNPs that may have been inherited together. We realize that in order to define such haplotypes, we have to know the phase of the SNPs.

5.1.1.2 Neuroimaging

Neurons. The human brain is composed on average of 170 billions of cells of which 86 billions are neurons. Neurons are nerve cells that are the basis unit of the nervous system. Neurons take care of the transmission of an electric signal called the Action Potential (AP). A neuron consists of a cellular body, where lies the nucleus, and of two types of extensions: the axon, that is unique and leads the AP away from the cellular body and the dendrites, that are numerous and carry the AP to the cellular body. Depending on the shape of the extensions, their number and localization, many classes of neurons have been defined. The transmission of an AP between two neurons is achieved through the synapse. In humans, synapses are essentially chemical and composed of a presynaptic, a postsynaptic cell and of a synaptic cleft between the two membranes. Among other things, presynaptic and postsynaptic cells can be either an axon or a dendrite. The axons are surrounded by a myelin sheath which accelerates the propagation speed of the AP.

Grey/White Matter. The human brain is located above the cerebellum and the brain stem. It is composed of two hemispheres and of the diencephalon which consists of structures that are on either side of the third ventricle, including the thalamus, the hypothalamus, the epithalamus and the subthalamus. In the two brain hemispheres, there are the grey and the white matter. The grey matter is located on the peripherical part of the brain and surrounds the white matter. Thus, it forms a kind of bark (cortex in Latin) around it. The grey matter mainly consists in the cellular body of the neurons. These cellular bodies are stacked on multiple layers of 3 to 6 cells leading to a thickness between 1 and 4.5 millimeters. Microscopically, it appears darker than the rest of the nerve tissue. The white matter consists in bundles of whitish-colored myelin-sheathed axonal fibers. The white matter allows to establish the connection between the cellular bodies of two distinct regions of the cortex.

Sulci. Geometrical, mechanical and genetic constraints have led to the phenomenon of cortical sulcation of the brain in the cranial cavity . The sulci are of variable depth and are delimited by ridges called gyri. The deepest sulci delimit the cortex into four lobes: the frontal, parietal, temporal and occipital lobes.

The CerebroSpinal Fluid (CSF). The human brain lies in a liquid called the CerebroSpinal Fluid (CSF). The main goal of this liquid is to absorb the shocks that could damage it. It is also used as a channel for the evacuation of waste produced by the brain. In the end, it plays an important role of immunological protection. Its composition reflects the physiological state of the brain: inflammation, infection, pharmacological molecules...

Atlases. A large part of the neuroimaging field tries to segment the brain and establish atlases. This segmentation allows for example to separate grey from white matter, to extract subcortical volumes (thalamus, hypothalamus...), to identify sulci... Once this segmentation is performed, atlases can be determined. For the ones we are interested in, they are mainly based on anatomical considerations. Grey matter or sulci based atlases can thus be found and help to divide the brain into Regions Of Interest (ROI). On these ROI, specific features can be computed: thickness of the grey matter, depth of a sulcus...

The Magnetic Resonance Imaging (MRI). The main tool used to study the brain anatomy is the Magnetic Resonance Imaging (MRI). It exploits the phenomenon of Nuclear Magnetic Resonance (NMR). The nucleus of some atoms has a spin magnetic moment. It is the case for the hydrogen atom which is present in large quantities in the human organism. When we apply a constant magnetic field to such atoms, their magnetic spin moment aligns with this magnetic field. In an MRI system the spin is then tilted perpendicularly to the constant field by an adapted radio-frequency wave. The return to alignment with the constant field of the spins following a precession trajectory creates the NMR signal picked up by the reading antennas. This signal has two components, parallel and perpendicular, which evolve according to an exponential of time constants T1 and T2 respectively. These constants reflect the concentrations of water molecules and the interactions these molecules have with their surroundings. In the end, an MRI allows the construction of T1 or T2 weighted maps that reflect the concentration or local environment of water molecules.

5.1.1.3 Preliminary Work

The analyses were conducted under UKB data application number #25251 on January 2018 release, consisting of 20.060 subjects with genotype data and brain T1-weighted MRI. The imaging quality control (QC) was performed by UKB following information described in [Alfaro-Almagro et al., 2018]. The UKB genetic data underwent also a stringent QC protocol, which was performed at the Wellcome Trust Centre for Human Genetics [Bycroft et al., 2017]. In the end, 15.612 subjects have been retained after QC protocol, British ancestry selection, and additional filtering for high heterogeneity, high missingness, first-degree relatedness and sex mismatch.

The UKB cohort is particularly suited to study natural and pathological aging, with a participants

mean age of 57 years, and a standard deviation of 8.2. Grey matter thickness is known to shrink with aging in both diseased and normal brains [Fjell and Walhovd, 2010, Ge et al., 2002, Lockhart and DeCarli, 2014]. A related effect is the cortical sulcus widening [Kochunov et al., 2005, Shen et al., 2018]. The width of a sulcus can be estimated using a feature called opening [Rivière et al., 2009] shown to be robustly related to grey matter thickness and which does not require spatial normalization nor regional atlas. The opening of a sulcus can be computed as the ratio of CSF volume contained in the sulcus and surface area of the sulcus. Heritability studies pointed to a dozen sulci that appeared to be under strong genetic control [Le Guen et al., 2017]. Furthermore, in [Le Guen et al., 2018], GWAS have identified a reproducible genetic marker associated with the opening of the left posterior cingulate sulcus (FCMpost_left). In this study, the sulcus opening phenotypes were studied in regards of 621.852 SNPs (see [Le Guen et al., 2018] for more details about their selection). See Figure 5.1-1 for a sum up of the results.

On Figure 5.1-1.a, for two specific SNPs (rs864736 and rs59084003), the p-value of their association with ten selected sulci openings is presented. It appears that both of them are strongly associated with the FCMpost_left sulcus opening. On Figure 5.1-1.b, the corresponding Locuszoom display [Pruim et al., 2010] is shown, focusing on a 500kb window on chromosome 1 around the two considered SNPs. Both of them appears to be in the upstream region of the KCNK2 gene. The correlation coefficient between each represented SNP and either rs864736 (circles) or rs59084003 (triangles) is also reported. Both rs864736 and rs59084003 pass the genomic threshold of significant association with the opening of FCMpost_left after correction for multiple testing. However, they appear to be strongly correlated and are located close to a recombination hotspot. Moreover, 5 other SNPs (the ones named in Figure 5.1-1.b), seem also to have an influence on the opening of FCMpost_left sulcus.

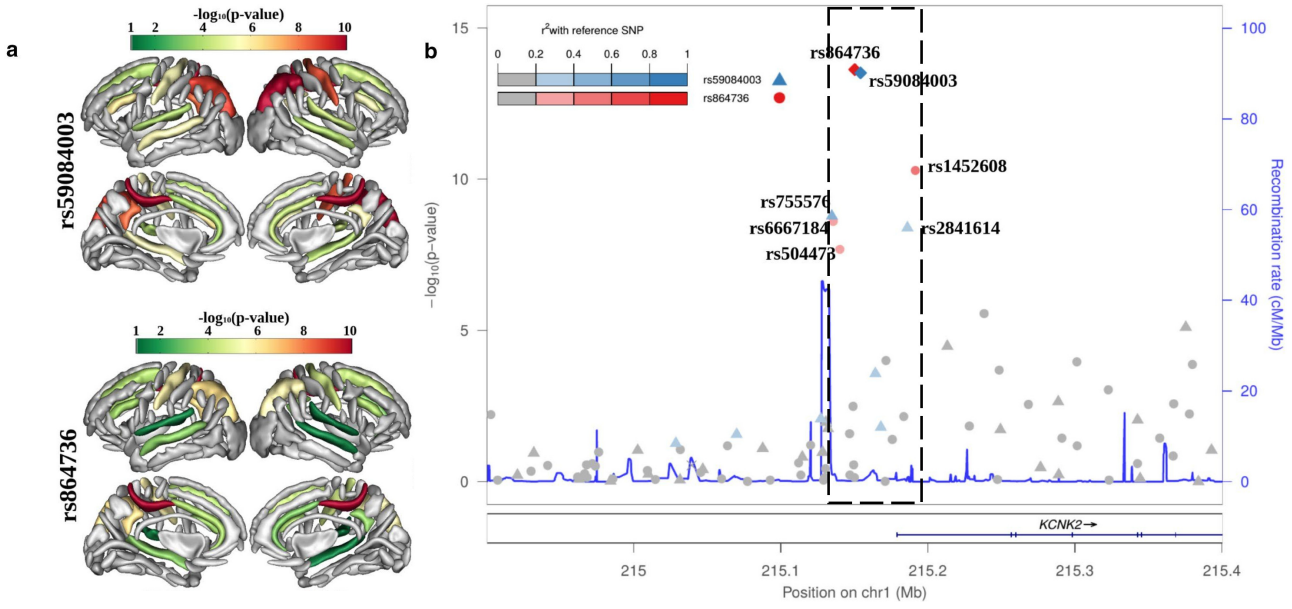


Figure 5.1-1 – Taken from [Le Guen et al., 2018]. GWAS hits upstream of KCNK2 regulating the sulcal opening. **a**: the log10 (p-value) of each SNPs (rs864736 and rs59084003) mapped onto the nominally significant sulci among the ten considered; **b**: Locuszoom display [Pruim et al., 2010] of the phenotype-variants association for the region upstream of KCNK2 with the left posterior cingulate sulcus opening as a phenotype.

5.1.2 Haplotype multivariate association analysis

In this section, we propose to analyze in a multivariate setup the associations between sets of genetic variants and multiple sulci widths. The genetic variants we consider are sets of SNPs of known phase called haplotypes, taken from the upstream region of the KCNK2 gene. To the best of our knowledge, multivariate analysis in imaging genetics has never been used in haplotype studies.

GWAS use a univariate approach and as such, suffer from several drawbacks, in particular the use of an unduly conservative multiple test correction and the fact that the correlation structure of the genome is not accounted for. In the context of complex traits, where individual variant effect size is expected to be small, only SNPs that are frequent in the population can significantly be associated with the phenotype. Moreover, univariate analyses are unable to model or predict the role of a genetic variant within the genomic region. Finally, univariate approaches are inadequate in situations where a set of variants are jointly associated with multiple phenotypes (pleiotropy). Using a multiple phenotype multivariate approach, we propose to alleviate these drawbacks by simultaneously analyzing one hundred related phenotypes and to model interactions between genetic variants within the same genomic segment.

The work undertaken is still under the UKB data application #25251 and uses the exact same participants and features as in Le Guen et al. [2018].

5.1.2.1 Imaging

For each selected subject, the brain mask of the T1-weighted image is obtained using SPM8 software (fil.ion.ucl.ac.uk/spm). Next, individual brain images were segmented into grey matter, white matter and CSF in BrainVisa. Finally, individual sulci were extracted using Morphologist, the sulcus identification pipeline of BrainVisa, to automatically segment [Fischer et al., 2012] and label [Perrot et al., 2011] 126 brain sulci. We retained the 96 most sample-wide identified sulci: a sulci was retained if it was missing in less than 1000 individuals (94.6% presence rate, see [Borne et al., 2018]). For each retained sulcus and for each subject, sulcus width or opening (the average distance between both banks) was estimated as the ratio of CSF volume and surface area of the sulcus [Le Guen et al., 2017, 2018].

5.1.2.2 Genetics

Genotyping data in UKB (UK Biobank Axiom Array) contains 820.967 SNPs. In such data, for a given SNP, the variant status is obtained without knowing if it lays on the paternal or maternal chromosome for a heterozygous subject. This raises an issue when one wants to use the chain of consecutive SNPs. In the 2018 release of UK Biobank, the so-called "phased data" are available for the 500.000 subjects. With this pre-processing, the succession of SNPs alleles is inferred in contiguous small regions of maternal or paternal chromosomes. Based on the results of a GWAS [Le Guen et al., 2018] where SNP rs864736 is found to be associated with the opening of several sulci, we chose a genomic region of 55.8 kbp on chromosome 1, which contained all the SNPs in LD with rs864736 (i.e., SNPs which are supposed to be inherited together from the same parent). This region consists in 18

SNPs that are inside the black-dashed box in Figure 5.1-1.b. Using the "phased data" of our 15.612 subjects in this region, we derived all the haplotypes with a length of 3 to 18 SNPs.

The construction of the haplotypes matrix is explained in (5.1). In this example, 3 SNPs are considered and two subjects. A subject is defined as $S_{i,j}$, where i refers to the subject and j to the chromosome considered. This time, as the SNPs are phased, they are only coded in 0 or 1 if they are present in the major or minor allele respectively. Then, the observed haplotypes are derived. Here, two haplotypes appear: 011 or 100. The first subject is heterozygous for this 3-locus example because (s)he has the two haplotypes, when the second subject is homozygous for the second haplotype.

$$\underbrace{\begin{array}{c|ccc|cc} S_{i,j} & \text{SNP}_1 & \text{SNP}_2 & \text{SNP}_3 & h_1 & h_2 \\ S_{1,1} & 0 & 1 & 1 & 1 & 0 \\ S_{1,2} & 1 & 0 & 0 & 0 & 1 \\ S_{2,1} & 1 & 0 & 0 & 0 & 1 \\ S_{2,2} & 1 & 0 & 0 & 0 & 1 \end{array}}_{\text{UKB haplotypes}} \longrightarrow \underbrace{\begin{array}{c|cc} S_i & h_1 & h_2 \\ S_1 & 1 & 1 \\ S_2 & 0 & 2 \end{array}}_{\text{H}} \tag{5.1}$$

In the end, after considering all the possible haplotypes with a length of 3 to 18 SNPs, and filtered out the less frequent ones (less than 1% as for the Minor Allele Frequency (MAF) of a SNP), 604 haplotypes are retained.

5.1.2.3 RGCCA

The interplay between neuroimaging and genetic data is uncovered using Regularized Generalized Canonical Correlation Analysis (RGCCA) (see Chapter 2 for more details). The first block, denoted $\mathbf{X}_1$, is related to neuroimaging and is defined by $J_1 = 96$ sulci measured on $I = 15.612$ individuals. The second block $\mathbf{X}_2$ is related to genetic information and is defined by $J_2 = 604$ haplotypes measured on the same set of I individuals. Function g is set to the square function.

For the two blocks, $\mathbf{M}_l = \tau_l \mathbf{I} + \frac{(1-\tau_l)}{I-1}\mathbf{X}_l^\top \mathbf{X}_l$, $l = 1, 2$, where τ_l is a scalar between 0 and 1. $\mathbf{M}_l$ can be considered as a shrinkage estimate of the true variance-covariance matrix Σ_{ll} [Ledoit and Wolf, 2004]. [Schäfer and Strimmer, 2005] gives an analytical formula for the optimal τ_l that minimizes the mean square error between the true covariance matrix Σ_{ll} for block l and its estimate $\mathbf{M}_l$ (see Chapter 1 section 1.2.1 for more details).

5.1.2.4 Bootstrap procedure and missing data imputation

A balanced bootstrap procedure [Gleason, 1988] is used to assess the reliability of block weight vectors estimated by RGCCA. For that purpose, $B = 2000$ bootstrap samples are considered. Some sulci were not detected in all individuals, therefore a simple regression imputation strategy is used to avoid missing values in each bootstrap sample of sulci opening data. The regression model is built to predict each opening value from the covariates Age, Sex, and the 10 first components of UK Biobank-provided multidimensional scaling. Residuals where reported in a new $I \times J_1$ matrix, where subjects with missing sulci (i.e. where not accounted for in the regression model) are set to 0. This procedure allows both to impute missing values and remove effects of covariates which are confounding factors in

our case. Finally, each residual bootstrap sample is standardized within each block in order to make the variables comparable. To make blocks comparable, each block was divided by the square root of its number of variables [Tenenhaus et al., 2017]. The RGCCA package (freely available at CRAN: cran.r-project.org) was then used to yield the weight vectors $\mathbf{w}_1^b$ and $\mathbf{w}_2^b$ for each bootstrap sample $b = 1, \ldots, 2000$.

Thanks to this bootstrap procedure, a distribution for each weight $w_{lj},\ l = 1, \ldots, L; j = 1, 2$ is obtained. A weight element w_{lj} is considered relevant if zero is excluded from $\left[\min_{b\in[\![1,B]\!]}(w_{l,j}^b), \max_{b\in[\![1,B]\!]}(w_{l,j}^b)\right]$, where $w_{l,j}^b$ is the estimate associated with the b^{th} bootstrap sample of the j^{th} element of the weight vector corresponding to the block l.

5.1.3 Results

Figure 5.1-2 represents the weights $\mathbf{w}_l,\ l = 1, 2$ computed with RGCCA. Only relevant weights according to the procedure described in section 5.1.2.4 are represented.

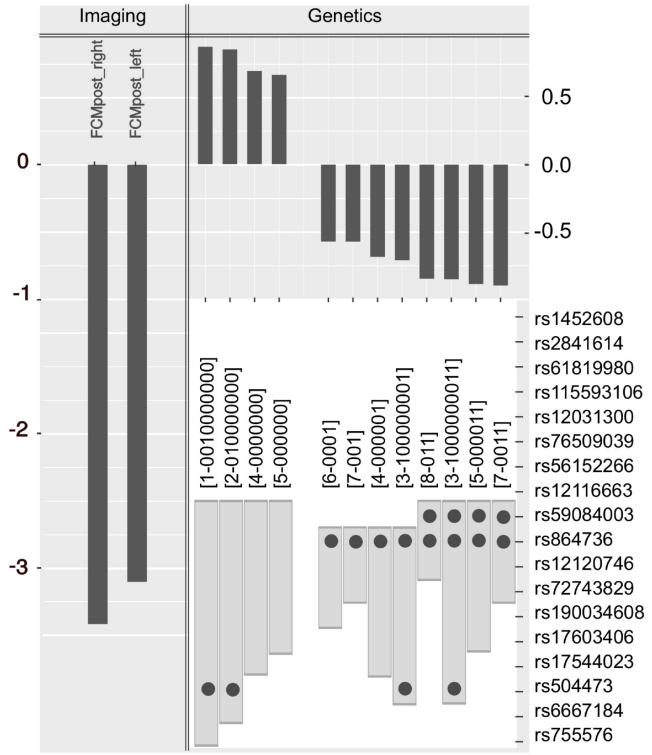


Figure 5.1-2 – (Left): Weight vector $\mathbf{w}_1$ associated with the selected variables of the imaging block. Selected features were the bilateral posterior cingulate sulci ; (Top, Right): Weight vector $\mathbf{w}_2$ associated with the selected variables of the genetic block. (Bottom, Right): SNP composition of selected haplotypes: light grey bars show the extent of the sequence and dots indicate the location of alternative alleles (see text for details).

5.1.3.1 Selected variables for imaging block

Figure 5.1-2 (top, left) shows a barplot of the weights associated with the 2 selected features of the imaging block. The selected variables correspond to the bilateral posterior cingulate sulci. The left

posterior cingulate sulcus opening was reported significantly associated with rs864736 in the univariate approach.

5.1.3.2 Selected variables for genetic block

Figure 5.1-2 (top, right) depicts the weights associated with the 12 selected haplotypes (in decreasing order). Figure 5.1-2 (bottom, right) gives the composition of the haplotypes: selected sequences of variants are represented as a grey box. In each sequence, grey dots indicate the alternative alleles. Haplotypes are named as follows: [index of starting SNP - sequence of variants], e.g. $[4 - 000001]$ refers to the haplotype that starts on position #4 with 5 reference alleles and a single alternative allele at position #9. Selected haplotypes included various combinations of variants (from 3 to 10), however none of the haplotypes included variants between position 11 to 18.

5.1.3.3 Findings interpretation

We will interpret the sign of the weights using haplotype $[4-000001]$ and left posterior cingulate sulcus (FCMpost_left) as an example. These both variables have negative weights in the model meaning that they are negatively correlated to their block component $\mathbf{y}_l$, $l = 1, 2$. However, over the $B = 2000$ bootstrap samples, correlation between $\mathbf{y}_1$ and $\mathbf{y}_2$ was always negative. To summarize, the presence of haplotype $[4 - 000001]$ is associated with a lower opening for FCMpost_left: haplotypes with a negative weight have a protective effect on the sulcus opening w.r.t aging. Opposite conclusions are drawn for haplotypes with a positive weight in the model.

5.1.4 Conclusion and future works

Previous studies by our group identified SNP rs864736 (and marginally rs59084003) as significantly associated with sulcus opening and grey matter thickness for left posterior cingulate, Intra-Parietal and Central sulci. Here, we proposed a multivariate model for haplotype associations with multiple quantitative traits that successfully recovered this previously known associations, and gained substantial knowledge regarding the genomic region and associated sulci. We present three new findings: 1) only the genomic region located upstream of rs864736 and rs59084003 seems to be implicated in the association ; 2) haplotype combinations are explanatory variables regarding posterior cingulate sulcus in both hemispheres ; and 3) an alternative allele at the third position (rs504473) seems to be associated with an antagonistic effect w.r.t rs864736 and rs59084003. Future works will extend this approach to gene clusters, gene pathways and larger intergenic regions to detect regulating patterns that interact with the observed phenotypes.

This method relies on a critical variable selection procedure based on bootstrap resampling. This procedure has shown to be sensitive to strongly co-linear variables, therefore we intend to propose several developments that could enhance this step. First, using a tree-like representation of haplotypes, we could regularize or combine variables, thus allowing us to keep more observations for the model estimation. Second, using block sparsity and regularization, multivariate procedures such as sparse group-LASSO could better account for co-linearity of the variables. For such analysis, sparse versions of RGCCA (see [Löfstedt et al., 2016, Tenenhaus et al., 2014] or Chapter 4) will be used. In the

context of imaging genetics, we argue that insights provided by multivariate approaches are key in uncovering the complex interactions between genes, structure and function.

5.2 A longitudinal imaging-genetic approach to predict Alzheimer's disease conversion

The dataset presented in this section was obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairments (MCI) and early Alzheimer's disease (AD). For up-to-date information, see www.adni-info.org.

5.2.1 Cohort description

The studied cohort was composed of 72 control subjects (CTL), 64 patients diagnosed with Alzheimer's disease (AD), 39 patients with Mild Cognitive Impairments (MCI), and 55 MCI who converted to AD during the study (MCIc). These participants have been recruited in the ADNI1 cohort.

We considered the following data types: longitudinal T1-weighted MRI, single-nucleotide polymorphisms (SNPs) and cerebrospinal fluid (CSF). On the MRI scans, Freesurfer (v5.3) has been run to extract the cortical thickness of different regions of interest and the volume of different subcortical regions based on the Destrieux (aparc 2009) atlas [Fischl et al., 2004]. Furthermore, these features were extracted for 4 different visits (time points): at baseline (bl), 6 months (m6), 12 (m12) and 24 (m24). A higher number of visits could have been chosen, however it would have reduced the number of subjects considered as only the participants present at all visits were kept. As a result, from the longitudinal T1-weighted MRI, two tensors were created, one associated with the cortical thickness ($\underline{\mathbf{X}}_1 \in \mathbb{R}^{230\times150\times4}$) and one with the subcortical volumes ($\underline{\mathbf{X}}_2 \in \mathbb{R}^{230\times59\times4}$). The genetic block ($\mathbf{X}_3 \in \mathbb{R}^{230\times148}$) comprises 148 genotyped SNPs related to known AD genes (ABCA7, MS4A6A, MS4A4E, EPHA1, CD33, CD2AP, BIN1, CR1, PICALM, CLU, GERAD2, APOE, PSEN1, PSEN2). The concentration of Amyloid-β (Aβ) 42 and 40 peptide in the CSF at baseline were used as the fourth block ($\mathbf{X}_4 \in \mathbb{R}^{230\times2}$). They were extracted from upennplasma.csv contained in the R package ADNIMERGE (https://adni.bitbucket.io/). The fifth block ($\mathbf{X}_5$) is a group coding matrix containing the disease status for each subject: CTL, AD, MCI or MCIc.

5.2.2 Question asked

In [Westman et al., 2012], a model was adjusted to discriminate between AD and CTL volunteers and then applied to predict if an MCI will convert to AD or not. It is a way to evaluate the robustness of the biomarkers: some biomarkers could be detected very early and enable some preventive management of the disease. The same approach is adopted here over different methods.

5.2.3 Methods

Several MGCCA or RGCCA models are trained on this dataset, with a Cross-Validation (CV) procedure, in order to learn how to discriminate between AD and CTL subjects. Then, these trained models are used to predict if a MCI will convert to AD or not.

For each trained model, $g(x) = x^2$ and two components are extracted from the cortical thickness and the subcortical volumes blocks (with different strategies, see below), while only one component is extracted from the CSF / SNPs / group coding matrix. The specificity of each trained model along with the Cross-Validation (CV) procedure are explained below.

Normalization. The Combat normalization procedure, first developed to remove batch effect in microarray expression data [Johnson et al., 2007], was used to remove the scanner and site effects on imaging blocks $\underline{\mathbf{X}}_1$ and $\underline{\mathbf{X}}_2$ [Fortin et al., 2018]. ComBat was applied both on the dataset composed of only the AD and CTL volunteers and only the MCI and MCIc volunteers separately. Each time, the center ID is taken as site variable and age as biological covariate. Then a standardization procedure (centering and reducing to unit variance) was nested inside the CV procedure.

MGCCA. MGCCA is used with the two sequential strategies to extract the two components of $\underline{\mathbf{X}}_1$ and $\underline{\mathbf{X}}_2$. These strategies yields (i) orthogonal components (c-MGCCA) or (ii) orthogonal mode-2 (features mode) weight vectors (w-MGCCA). In addition to these two deflation procedures, two different design matrices $\mathbf{C}$ are evaluated: either (i) all blocks are connected to one another (a.k.a. complete design: C) or (ii) all blocks are connected only to the 5^{th} block, the diagnosis block (a.k.a. hierarchical design: H). In the end, 4 different models are trained: c-MGCCA C, c-MGCCA H, w-MGCCA C and w-MGCCA H.

For c-MGCCA models, $\mathbf{M}_l$ is a $J_l K_l \times J_l K_l$ block diagonal matrix, where the k^{th} block is $\mathbf{M}_l^k = \tau_l \mathbf{I}_{J_l} + (1-\tau_l)\mathbf{X}_{..k}^{l^\top}\mathbf{X}_{..k}^l$, with $\tau_l \in [0;1]$. This structure was chosen because it comes down to apply RGCCA constraint seen in Chapter 2 to each frontal slice $\mathbf{X}_{..k}^l$ separately.

For w-MGCCA models, as seen in section 3.2.3.2, $\mathbf{M}_l$ needs to have a Kronecker structure. Thus, $\mathbf{M}_l = \mathbf{M}_l^K \otimes \mathbf{M}_l^J$ with $\mathbf{M}_l^J = \tau_l \mathbf{I}_{J_l} + (1-\tau_l) I^{-1} \left(\sum_{k=1}^{K_l} \mathbf{X}_{..k}^{l^\top} \right) \left(\sum_{k=1}^{K_l} \mathbf{X}_{..k}^{l} \right)$ and $\mathbf{M}_l^K = \tau_l \mathbf{I}_{K_l} + (1-\tau_l) I^{-1} \left(\sum_{j=1}^{J_l} \mathbf{X}_{.j.}^{l^\top} \right) \left(\sum_{j=1}^{J_l} \mathbf{X}_{.j.}^{l} \right)$, where $\tau_l \in [0;1]$. With this structure, $\mathbf{M}_l^J$ (resp. $\mathbf{M}_l^K$) represents an estimation of the covariance matrix associated with the sum of the frontal slices (resp. lateral slices).

RGCCA. RGCCA is used with a sequential procedure to extract two components out of the mode-1 matricization of $\underline{\mathbf{X}}_1$ and $\underline{\mathbf{X}}_2$ (and one component out of the other blocks). Moreover, all blocks are connected only to the 5^{th} block, the diagnosis block (a.k.a. hierarchical design: H). Finally, $\mathbf{M}_l = \tau_l \mathbf{I} + (1-\tau_l) I^{-1} \mathbf{X}_l^\top \mathbf{X}_l$, where $\tau_l \in [0;1]$.

Hyperparameters. For each $\mathbf{M}_l$ presented above, one parameter τ_l has to be tuned. For low-dimensional blocks (CSF and SNPs), it was fixed to 10^{-5} in order for these blocks to give more importance to the correlation with the block component of the group coding matrix. For the group

coding matrix, $\tau_5 = 0$. In the end, only τ_1 and τ_2 are tuned through a logarithmic grid of 10 values between 10^{-6} and 1.

Cross validation. The cross validation is performed only on the AD and CTL subjects (136 participants in total) in order to learn how to discriminate between the two groups. A 10-folds Monte-Carlo Cross-Validation (MCCV) procedure is used to tune parameters. For each fold, the subsampling is stratified between a training (80%) and testing set (20%). Each MGCCA and RGCCA model is learned on a train set and leads to 7 components: 2 for the cortical thickness, 2 for the subcortical volumes and 1 for each of the CSF, SNPs and group coding matrix. Then, these components, except for the group coding matrix one, are joined together into a 6-column matrix. A linear discriminant analysis (LDA) is applied onto this 6-column matrix to predict between AD and CTL. Finally, each trained model (MGCCA or RGCCA model and LDA) is applied to the corresponding test set and leads to a prediction accuracy. At the end of the 10-fold MCCV, the parameter set that performed best according to the mean of prediction accuracy, across all test sets, is chosen. In Table 5.1, column «Test AD/CTL» presents the mean of prediction accuracy across the 10 test sets for the optimized parameters.

Prediction for MCI/MCIc. With the optimized parameter set, each model is run on the dataset composed of all AD and CTL subjects. Then, as explained earlier, a 6-column matrix is created by joining the 2 components associated with the cortical thickness and the subcortical volumes, the SNPs component and the CSF component. Thus, a LDA and a k-nearest-neighbors (KNN) are trained on this 6-column matrix to predict for AD vs. CTL. Finally, each trained model (MGCCA or RGCCA model and LDA or KNN) is applied on the dataset composed of only MCI and MCIc subjects (96 participants). In Table 5.1, prediction results to discriminate between MCI and MCIc based on tuned models for the discrimination between AD and CTL are in column «Prediction MCI/MCIc». Columns «LDA» and «KNN» refer to the accuracy of prediction in the MCI/MCIc task with these two methods. «AUC» column refers to Area Under the Curve computation based on the output of LDA results.

5.2.4 Results

Results are presented in Table 5.1. MGCCA performs slightly better in test than RGCCA and performs better to predict MCI/MCIc.

Among MGCCA settings, results for the MCI/MCIc task are better for c-MGCCA over w-MGCCA. This is due to an overfitting. The τ_l parameters selected for w-MGCCA in the case of the AD/CTL task present good results in test but poor generalization for the MCI/MCIc task. In general, these results are quite similar to those reported in the literature, see [Guigui et al., 2019], where similar methods are employed to analyze the ADNI cohort.

The best results of MGCCA are obtained with the complete connection and the orthogonality at the component level. An attempt to interpret the weights estimate is provided. For the AD/CTL task, the prediction power (evaluated with a LDA) in term of accuracy in test for the 2 components of the cortical thickness block are 0.86 and 0.53, for the 2 components of the subcortical volumes 0.87 and 0.53, for the SNPs block 0.63 and for the CSF block 0.55. We choose to focus our interpretation

Table 5.1 – Results for the 10-folds MCCV stratified between Train (80%) and Test (20%). Prediction accuracy are presented for the test sets on the AD/CTL task. All trained models are then applied to predict if a MCI volunteer will convert or not to AD (Prediction MCI/MCIc). MGCCA and RGCCA are compared with different settings. H: hierarchical design, C: complete design. c-MGCCA: deflation with orthogonality between components. w-MGCCA: deflation with orthogonality between weights $\mathbf{w}_l^J$.

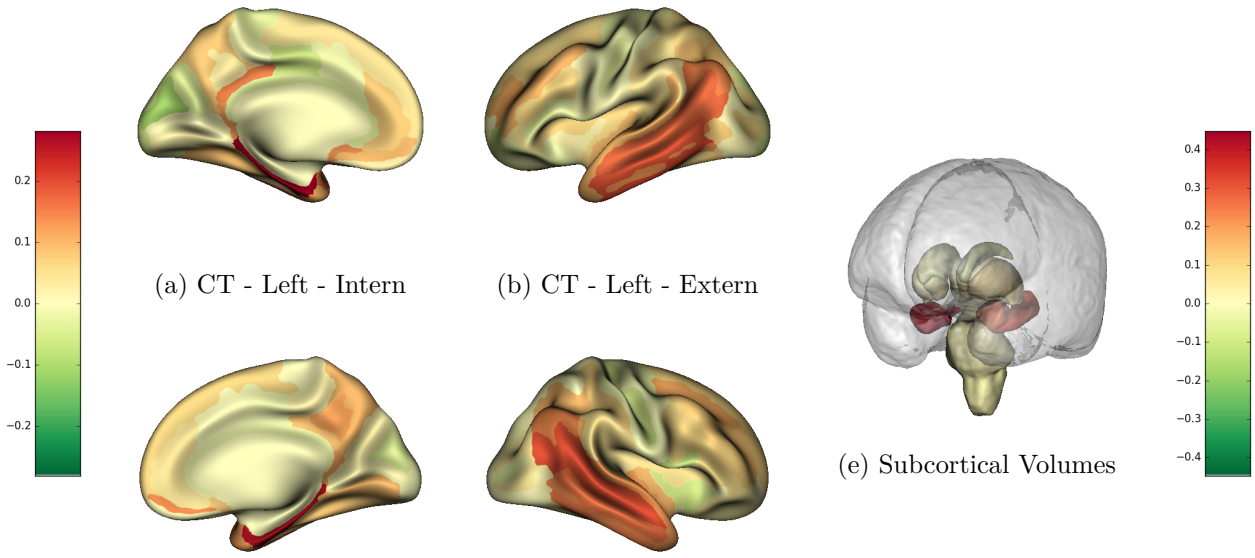
Method	Mean	Median	Std	LDA	KNN	AUC
		Test AD/CTL			Prediction MCI/MCIc	
sequential RGCCA H	0.84	0.84	0.06	0.62	0.60	0.65
c-MGCCA H	0.89	**0.89**	0.05	0.69	**0.72**	**0.73**
c-MGCCA C	**0.90**	**0.89**	**0.04**	**0.70**	0.68	**0.73**
w-MGCCA H	0.88	0.88	0.06	0.56	0.59	0.65
w-MGCCA C	0.88	0.88	0.06	0.57	0.60	0.65

on the two components with high prediction power, so the first component of the cortical thickness and the subcortical volumes.

Figure 5.2-3 depicts the mode-2 block weight vectors (features dimension) associated with the cortical thickness and the subcortical volumes. For the cortical thickness (Figures 5.2-3.a, b, c and d), the coefficients associated with the highest magnitude are located in the left and right temporal lobes. For the subcortical volumes (Figures 5.2-3.e), the coefficients associated with the highest magnitude are located in the left and right Hippocampus and Amygdala. These results are similar to those reported in the literature [Guigui et al., 2019, Lorenzi et al., 2018]. If now we take a look at the mode-3 block weight vectors (time dimension), they are both composed of four values (bsl, m6, m12, m24) and equal for the cortical thickness to: 0.494, 0.493, 0.507 0.507 and for the subcortical volumes to: 0.493, 0.489, 0.507, 0.510. These weights are relatively close to each other so it is hard to conclude. In the following paragraphs, a procedure is presented to enhance the interpretation of these results. This new interpretation focuses on AD and CTL volunteers.

A common practice in CCA is to look at the components. An interesting advance with MGCCA is that it is possible to compute mode components. Let us define $\underline{\mathbf{X}}_1^N$ and $\underline{\mathbf{X}}_2^N$ the normalized blocks of the cortical thickness and the subcortical volumes composed only of the AD and CTL subjects. The normalization consists in both the Combat normalization and the standardization mentioned earlier. It is possible to compute the mode-2 components as $\mathbf{Y}_l^J = \mathbf{X}_l^N \left(\mathbf{w}_l^K \otimes \mathbf{I}_{J_l}\right),\ l = 1, 2$, which is homogeneous to the second mode of $\underline{\mathbf{X}}_l^N$. We can even go further and compute $\mathbf{y}_l^{J,AD}$ and $\mathbf{y}_l^{J,CTL}$ respectively the median through the AD and CTL subjects for all the elements of $\mathbf{Y}_l^J$.

On Figure 5.2-4, $\mathbf{y}_l^{J,CTL} - \mathbf{y}_l^{J,AD},\ l = 1, 2$ are represented. To begin with, all these elements are positive except for 2 Regions Of Interest (ROI) of the cortical thickness, even though their distribution is not significantly different from 0 according to a t-test. It means that this component managed to catch a neurodegenerative effect characterizing Alzheimer's disease. For the cortical thickness (Figures 5.2-4.a, b, c and d), the difference between CTL and AD is higher in the left and right temporal lobes, particularly in the left. The left frontal lobe seems also highlighted in this difference. For the subcortical volumes (Figures 5.2-4.e), the difference is higher in the left and right Hippocampus and Amygdala. The left and right Accumbens and the left Putamen are also underlined.

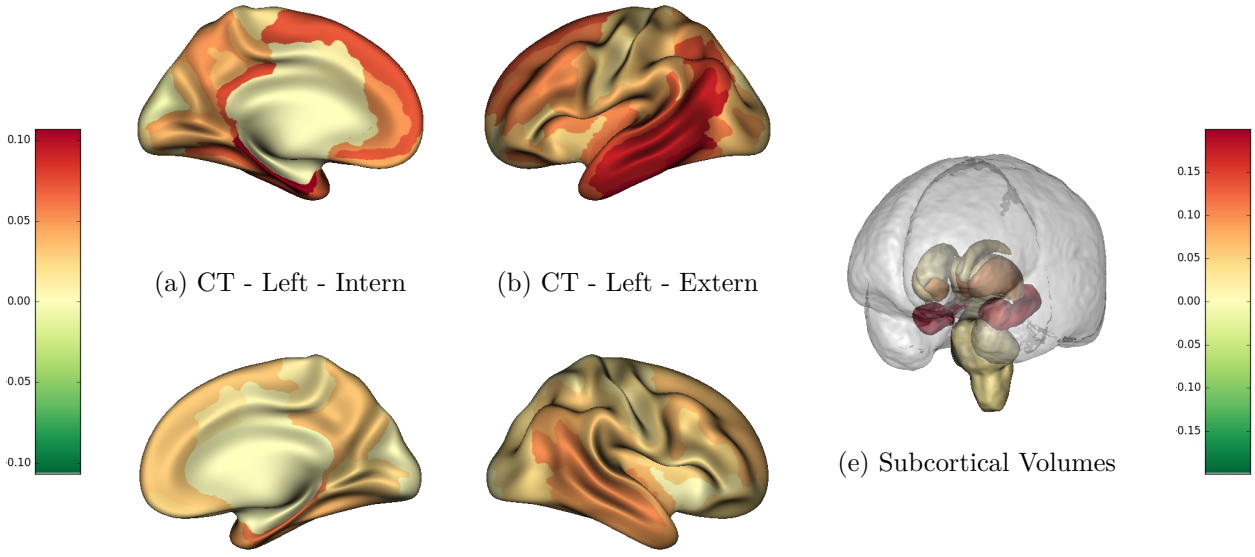


(a) CT - Left - Intern (b) CT - Left - Extern

(e) Subcortical Volumes

(c) CT - Right - Intern (d) CT - Right - Extern

Figure 5.2-3 – Mode-2 weight vectors for c-MGCCA (deflation with orthogonality between components) complete (all blocks connected) associated with the Cortical Thickness (CT): a, b, c, d; and the Subcortical volumes: e. Each block is associated with its own magnitude scale



(a) CT - Left - Intern (b) CT - Left - Extern

(e) Subcortical Volumes

(c) CT - Right - Intern (d) CT - Right - Extern

Figure 5.2-4 – Representation of $\mathbf{y}_l^{J,CTL} - \mathbf{y}_l^{J,AD}$, $l = 1, 2$ (see the text for more details about their construction) computed for c-MGCCA (deflation with orthogonality between components) complete (all blocks connected) associated with ($l = 1$) the Cortical Thickness (CT): a, b, c, d; and ($l = 2$) the Subcortical volumes: e. To compute $\mathbf{y}_l^{J,CTL} - \mathbf{y}_l^{J,AD}$, $l = 1, 2$, only AD and CTL volunteers were considered. Each block is associated with its own magnitude scale

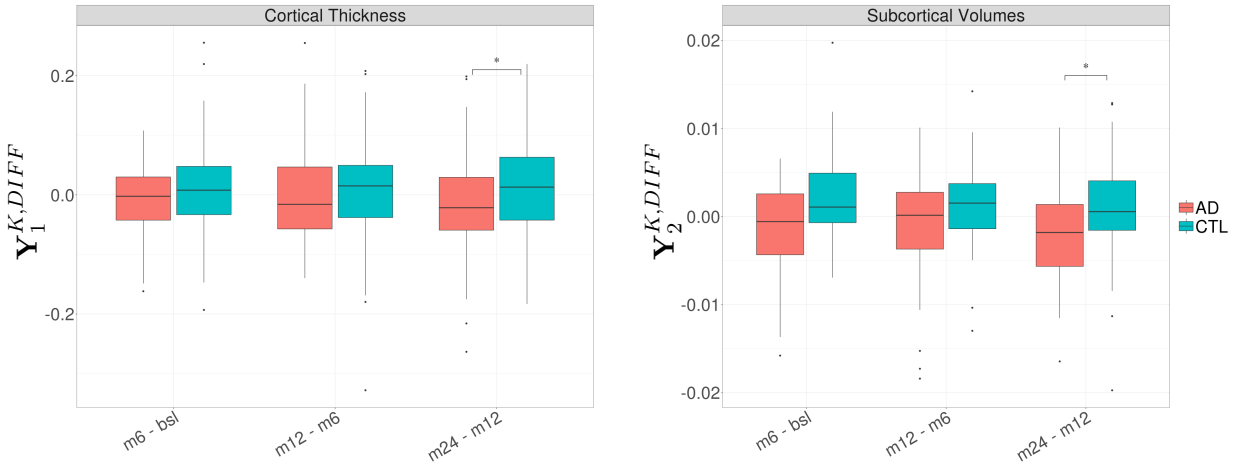


Figure 5.2-5 – Boxplot of each of the three columns of both $\mathbf{Y}_l^{K,DIFF}$, $l = 1, 2$ represented for either the AD or CTL subjects (see text for more details). $\mathbf{Y}_l^{K,DIFF}$, $l = 1, 2$ was computed for c-MGCCA (deflation with orthogonality between components) complete (all blocks connected) associated with ($l = 1$) the Cortical Thickness (CT): left; and ($l = 2$) the Subcortical volumes: right.

Similarly the mode-3 components can be defined as $\mathbf{Y}_l^K = \mathbf{X}_l^N \left(\mathbf{I}_{K_l} \otimes \mathbf{w}_l^J\right)$, $l = 1, 2$, which is homogeneous to the third mode of $\underline{\mathbf{X}}_l^N$. This time, instead of directly computing the median through a class of subject, for each individual, the difference between two consecutive visits is calculated. For example, in the case of the subcortical volumes, the evolution of the mode-3 component between the m6 and m12 visits is computed as: $\mathbf{y}_{2,3}^K - \mathbf{y}_{2,2}^K$. The resulting matrices are defined as $\mathbf{Y}_l^{K,DIFF}$, $l = 1, 2$ and are both composed of 3 columns (m6 - bsl, m12 - m6, m24 - m12). On Figure 5.2-5, the distribution of each column of $\mathbf{Y}_l^{K,DIFF}$, $l = 1, 2$ is represented with a distinction between the AD and CTL subjects. A t-test is applied for each time difference and block between the AD and CTL and after a Bonferroni correction, it appears that for both the Cortical thickness and the Subcortical Volumes, the mean of the two distributions are significantly different for (m24-m12) with a confidence level of 0.05.

Finally, on Figure 5.2-5, the difference between the mean of the CTL and AD subjects is positive for each time difference considered and increases with time. If we recall that on Figure 5.2-4, $\mathbf{y}_l^{J,CTL} - \mathbf{y}_l^{J,AD}$, $l = 1, 2$ is almost always positive, it means that these components managed to extract a neurodegenerative effect worsening over time.

On this particular example, a multiway multiblock method performed better than an only multiblock method. When multiway structure is taken into account, results are more interpretable because it is possible to separately analyze the effects of the cortical thickness (or subcortical volumes) and the time. Indeed, it appears that the third and fourth visits (m12 and m24) play a greater role in the discrimination between AD and CTL which is expected because it is a neurodegenerative disease. As time goes by, brain regions that suffer from atrophy will be more and more different from the same regions in control subjects.

5.2.5 Conclusion

In the context of the Alzheimer's Disease, MGCCA provided components that caught a neurodegenerative effect characterizing AD patients. The interpretation of the mode component was particularly interesting in this analysis in order to determine the different affected regions in the brain.

Even though the results are at the same scale as those reported in the literature (see [Guigui et al., 2019, Westman et al., 2012]), they are not better. This might be explained by a smaller number of available subjects (with required data) than in these previous studies. Indeed, we only took into account subjects without any missing observations in the five blocks considered and also in the four visits selected to build the two tensors. In this context, it seems particularly interesting to develop an extension of MGCCA that can handle missing values, which would allow to keep subjects with only few missing data.

Another possible explanation for these results is the fact that for the study of brain degeneration, multiway constraints might be too strong. Indeed, by using a multiway method to study this longitudinal dataset, we made the assumption that all brain regions taken into account degenerate at the same speed, which is not true. A sparse extension of MGCCA might come in hand to overcome this drawback. This would allow each extracted component to focus on the degeneration of a specific group of brain regions.

5.3 Raman Microscopy Data

5.3.1 The Raman Microscopy

When a molecule is illuminated by an intense monochromatic light source (a laser for example), most of the photons diffused by this molecule have the same wavelength as the source of excitation. However, 1 over 100 millions of the photons are diffused with a different energy and hence a different wavelength. This is called the Raman effect.

This light spectrum diffused by a molecule after its monochromatic excitation is called the Raman spectrum. It uniquely characterizes the molecule. Moreover, the intensity of the different light lines measured is proportional to the concentration of the molecule in the studied environment.

The combination of the Raman effect with a confocal microscope helps to analyze a specific volume of the studied sample. Indeed, this association both allows to focus the light on a designated volume of the sample and to spatially filter the light diffused back by it in order to restrain the measurements to the desired space. This method is called either confocal Raman spectroscopy or Raman microscopy. For more information about the Raman microscopy, the reader is referred to [Roig, 2015].

In a nutshell, the Raman microscopy is a non invasive technique that allows to study the molecular composition of different layers of a sample. This technology was used to collect the data of interest in this section.

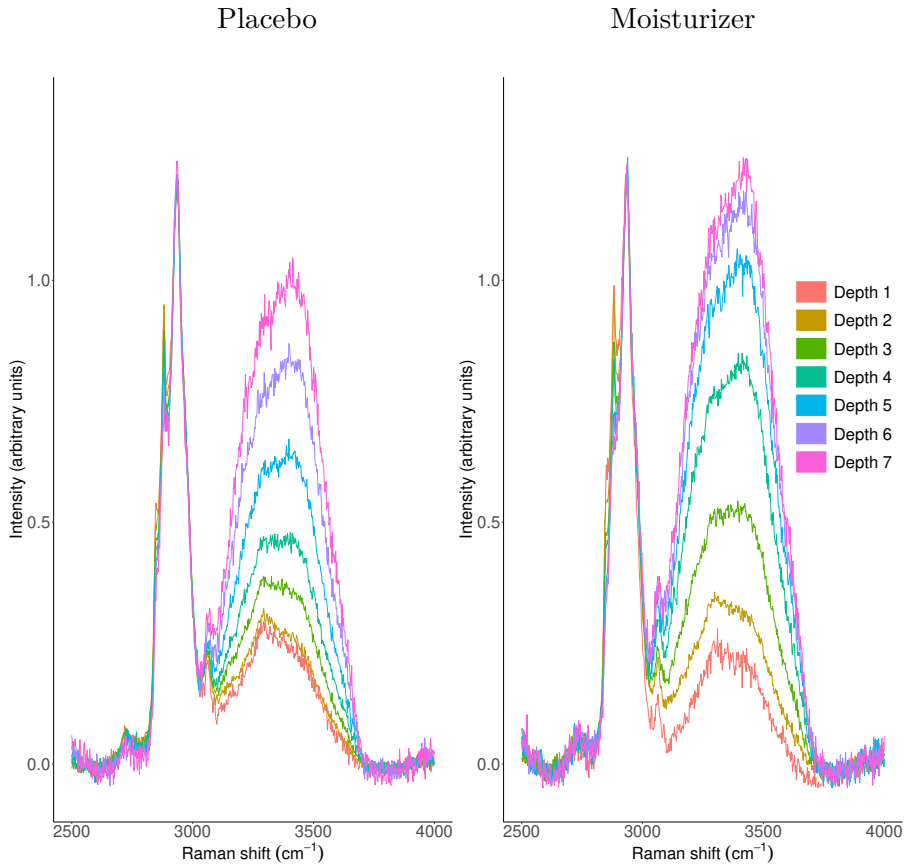


Figure 5.3-6 – Raman spectra associated with the T3 visit of a specific subject in the study. The left panel is associated with the arm that received a placebo and the right panel to the moisturizer. Each time, 7 Raman spectra are represented, one per depth of the skin analyzed.

5.3.2 Context of the study

This study aims at analyzing the efficiency of a moisturizer thanks to Raman microscopy.

The study was conducted on 13 volunteers that had received on one arm the moisturizer of interest and on the other arm, a placebo (with random assignment). Both of their arms were analyzed through Raman microscopy in order to understand if the moisturizer leads to changes in the constitution of the skin in comparison to the placebo. The skin of the volunteers was analyzed through a 28 μm thick portion from the skin surface. This corresponds mainly to the stratum corneum, the outermost layer of the epidermis. With Raman microscopy, this skin portion was divided in 7 layers of different depths, which led to 7 different Raman spectra per arm and volunteer, ranging from 2500 to 4000 cm^{-1}. Finally, the analysis was repeated at different time points: at the time of the moisturizer administration (T1), 2 weeks later (T2), 4 (T3), 8 (T4) and 12 (T5) weeks later.

The resulting data set is composed of 10 tensors (one per visit and arm) of dimensions 13 subjects $\times$ 751 wavenumbers (in cm^{-1}) of diffusion $\times$ 7 depths of the skin. The Raman spectrum associated with the T3 visit is plotted for both arms of an individual and all layers analyzed in Figure 5.3-6. The x-axis is labelled as «Raman shift» as the diffusion wavenumbers are measured in comparison to the wavenumber of the source of excitation. All these spectra were normalized using dedicated procedure for Raman microscopy data. Moreover, the data is centered and if a cross-validation procedure is

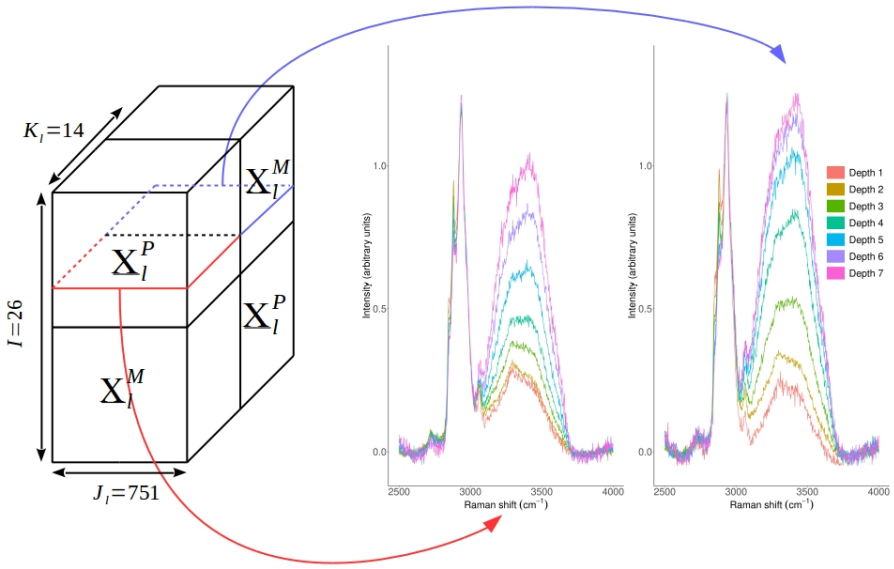


Figure 5.3-7 – Tensor construction for one visit.

used, this centering is nested inside the cross-validation.

5.3.3 Tensor construction

Our goal is to understand if the moisturizer indeed changed the moisturization of the skin in comparison to the placebo and if this effect is persistent over time.

For this purpose, MGCCA is used on the 10 tensors mentioned earlier. However, the tensor are modified in order to account for the pairwise design of the study. Indeed, for each subject, two measurements were collected, one per arm. If we take the example of the first visit T1, the two tensors are $\underline{\mathbf{X}}_1^P$ for the placebo and $\underline{\mathbf{X}}_1^M$ for the moisturizer. These tensors are of dimensions $13 \times 751 \times 7$. A first intermediary tensor $\underline{\mathbf{X}}_1^{P-M}$ is created by stacking $\underline{\mathbf{X}}_1^P$ and $\underline{\mathbf{X}}_1^M$ along their third mode and a second intermediary tensor $\underline{\mathbf{X}}_1^{M-P}$ is created by stacking $\underline{\mathbf{X}}_1^M$ and $\underline{\mathbf{X}}_1^P$ along their third mode. These intermediary tensors are both of dimensions $13 \times 751 \times 14$. Finally, $\underline{\mathbf{X}}_1^{P-M}$ and $\underline{\mathbf{X}}_1^{M-P}$ are stacked along their first mode leading to $\underline{\mathbf{X}}_1$ of dimensions $26 \times 751 \times 14$. This procedure is depicted in Figure 5.3-7. It is applied to every visit, hence, 5 tensors of size $26 \times 751 \times 14$ are constructed.

This particular tensor construction is meant to perform a differential analysis. Indeed, we do not want to determine if an arm of a subject was, in absolute, treated or not. However, having the Raman spectrum of both arms, we want to assign a class (treated or not) on both of them at the same time. This is particularly interesting to handle individual variations. Indeed, the arms of a subject might be in general more moisturized than the other subjects.

However, with this construction, we cannot learn a treated/non-treated classification model. On the contrary, as an observation is the concatenation of two arms, what we are going to predict now is if an observation is composed of first a spectrum associated with a treated arm and then a spectrum associated with a non-treated arm or the other way round.

In the end, we have 6 blocks: 5 tensors (one per visit) $\underline{\mathbf{X}}_l \in \mathbb{R}^{26\times751\times14}$ and one vector $\mathbb{R}^{26}$ composed of 2 classes: «P-M » (first a non-treated then a treated arm) or «M-P » (first a treated then a non-treated arm).

5.3.4 Methods

MGCCA is applied on this 6-block dataset with either a hierarchical (H; all blocks connected to the block corresponding to $\mathbf{y}$) or a complete (C; all blocks connected together) design. RGCCA is also applied with a hierarchical design on the mode-1 matricized tensors. For these 3 methods, the function g is set to the square function, only one component is extracted and all $\mathbf{M}_l$ matrices are set to the identity except for the $\mathbf{y}$ block where $\mathbf{M}_6 = \frac{1}{I}\mathbf{y}^\top\mathbf{y}$ (the covariance of $\mathbf{y}$ as $\mathbf{y}$ is centered).

The different methods are evaluated through a 10-fold Monte-Carlo Cross-Validation (MCCV) framework. Ten folds are created, where each time 8 individuals (so 16 pairwised observations) are randomly assigned (without replacement) to the train set and 5 to the test set.

MGCCA and RGCCA weights are learned on the train set. Then a Linear Discriminant Analysis (LDA) is applied on the components extracted from the 5 tensors in order to predict $\mathbf{y}$. In the end, the weights learnt for MGCCA/RGCCA and the LDA are applied on the test set in order to classify the spectra. The results are presented in the next section.

5.3.5 Results and Weights interpretation

All three methods perform the same on the train set with the median of the accuracy equal to 1 over the 10 folds. However on the test set, the two MGCCA models obtain a median of accuracy of 0.8 (one mistake) against 0.7 (two mistakes) for RGCCA.

The great advantage of MGCCA over RGCCA is the possibility to interpret separately the weight vectors depending on their mode. Figure 5.3-8 shows for each tensor block both mode-2 and mode-3 weight vectors for MGCCA with either a hierarchical or a complete design. First, concerning the mode-2 weight vectors, for any block, the Raman shift mainly involved the bandwidth from 3050 to 3700 cm^{-1}, which correspond to the water molecule. Even the shape of these weights is similar to the diffusion spectrum of the water molecule. Thus, MGCCA found that the best part of the Raman spectrum that helps distinguishing between a treated and a non-treated arm is the one associated with the water bandwidth. So, the moisturizer indeed led to a change in the constitution of the skin and this change concerns water, which is convenient for a moisturizer. Then, for the mode-3 weight vectors, for any block (except for the fourth visit in the hierarchical model), the main layers involved in the discrimination between treated and non-treated arms are the deepest ones (the four last ones). This means that the change in the composition of the skin mentioned earlier is located in these deepest layers. Moreover, through visits, the magnitude of the weights seems to be constant, and even slightly increasing if we look at $D4$ or $D5$. It means that the effect of the moisturizer is persistent over the 12 weeks of the study.

Furthermore, if we compare the hierarchical and complete models, the complete one, by linking all blocks together, offers smoother weights. This is particularly interesting here as all blocks have the same nature. We thus realize that, for the mode-3 weight vectors, T1 and T2 seems to isolate a rather strong effect at $D4$ and $D5$. But then, from $T3$ to $T5$, this effect seems to spread homogeneously (and thus decrease in intensity) among layers between $D4$ to $D7$.

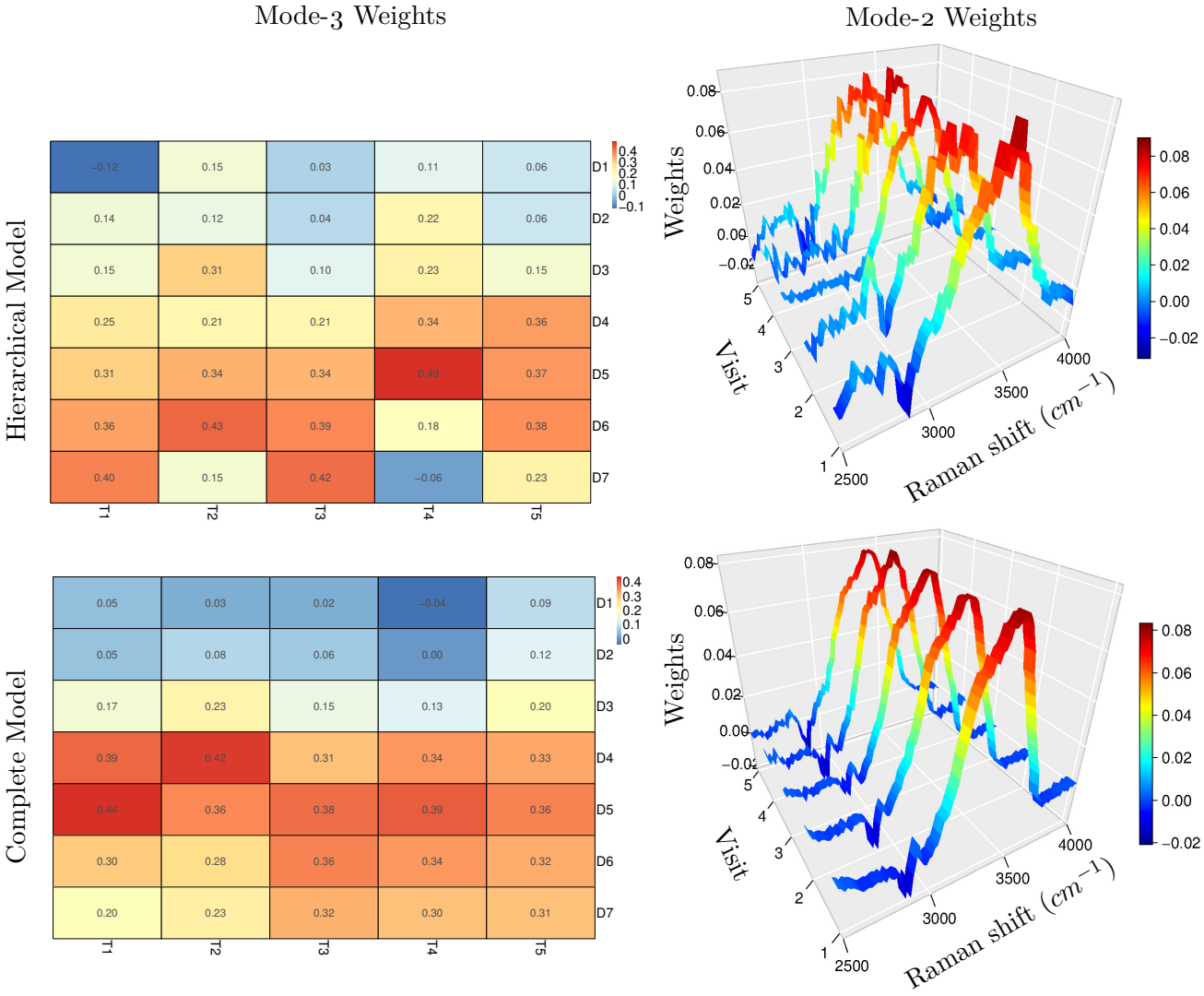


Figure 5.3-8 – MGCCA weight vectors with a hierarchical connection (first row) and a complete connection (second row). The first column corresponds to the mode-3 weight vectors and the second to the mode-2. For the mode-2 weight vectors, blocks are represented in the second dimension (Visit). For the mode-3 weight vectors, the rows are associated with the depth and the columns with the visits (one per block).

5.3.6 Conclusion

In this study, we have shown the usefulness of the MGCCA method for the analysis of a Raman microscopy dataset. In comparison to RGCCA, MGCCA performs not so better, however, it offers a much richer interpretability thanks to the mode weight vectors.

The discrimination between the placebo and the moisturizer seems to involve only a specific bandwidth and particular depths of the skin. It might be interesting to introduce sparsity into MGCCA in order to extract only these weights. It has already been done in the context of multiway LDA with this same dataset [Le Brusquet et al., 2015] and indeed led to non-null weights only around 3500 cm^{-1} for the mode-2 weights and for the last three depths for the mode-3 weights.

Looking at the mode weight vectors of the complete design, they seem quite similar over the visits. This suggests to handle the data as a fourth order tensor (visits are the fourth mode) rather than a collection of three-way tensors. In this perspective, we can use PARAFAC to analyze the 4-way tensor. Thus, the algorithm is going to extract effects that are parallel in every modes. However, for the new «visit mode», these effects are not parallel. When we get a close look at the mode-3 weights of the complete design, the magnitude of $D4$ and $D5$ decreases after the second visit when the one of $D6$ and $D7$ raises. This could be caught with a PARAFAC model but with two components at least. PARAFAC offers a way to extract phenomena that are parallel in every mode. MGCCA offers the possibility to relax this constraint on one mode. This allows to study the specificity of a parallel effect for every element of the chosen mode.

5.4 The BABABAGA experiment, an ElectroEncephaloGraphy (EEG) study

5.4.1 ElectroEncephaloGraphy (EEG)

The ElectroEncephaloGraphy (EEG) is a non-invasive technique allowing to measure the electrical activity of the brain through electrodes placed on the scalp. The electrical signal measured is the summation of the post-synaptic synchronous action potentials (AP) from a large number of neurons. The EEG is gifted with a good temporal resolution because its frequency of acquisition is in general around 250 Hz, so one sample every 4 ms. This is approximately at the same scale of the APs that last between 1 and 2 ms. However, the spatial resolution is poorer as the classical EEG-headsets comprise between 64 and 128 channels. Moreover, as the decay of the electric waves is in $1/r^2$, where r is the distance from the electric dipole generating the wave, the measured signals mainly correspond to neurons located in the cortex of the brain.

The main phenomenon studied through EEG in this section and the following one is the Evoked Potential (EP) also called Event-Related Potential (ERP). It refers to a change in the electrical potential generated by the nervous system in response to an external stimulus, mainly sensitive (image or sound), but also to an internal event, mainly cognitive activity (attention, motor attention). As the ERP are usually weak compared to the ambient noise, they are recorded along multiple trials in order to average all these trials to raise the SNR.

As mentioned in [Acar and Yener, 2009], multiway methods are particularly suited for EEG

study. Indeed, for example, when an ERP is generated at a specific location, the time that it takes to propagate towards the electrodes is so much higher than the time of acquisition that it seems that the different electrodes capture it simultaneously. However, as explained earlier, the magnitude of this signal decays with the distance so the electrodes are going to register an effect at the same time but with different magnitude depending on their distance from the source. This effect can indeed be decomposed as a Kronecker product.

In this section and the following one, EEG was used in order to collect the data.

5.4.2 Description of the Study

The objective of this study is to identify brain areas implicated in the process of discrimination between two close syllables /ba/ and /ga/ in two- to three-month-old human infants using high-density electroencephalography (EEG). For this purpose, EEG acquisitions were performed on 53 infants while they were listening to two stimuli of four syllables. The syllables were separated by a silence of 430 ms and the stimuli by a silence of 4 s. In each stimulus trial, the first three syllables were always repeated whereas the fourth was either similar to the previous ones (standard trials: «BA-BA-BA-BA» or «GA-GA-GA-GA») or different (deviant trials: «BA-BA-BA-GA» or «GA-GA-GA-BA»). Thus in total, four different stimuli are possible. In such a paradigm, we expect that the discriminative times between standard and deviant stimuli are located after the presentation of the fourth syllable.

A 128-channels recording device with a time resolution of 4 ms (250 Hz) was used. The raw signals were pre-processed as described in [Dehaene-Lambertz and Dehaene, 1994]. In a nutshell, the entire recording was band-pass filtered between [0.5-20] Hz, then an $5.8s$ epoch is defined around the fourth syllable onset $[-4.496\text{ s} ; 1.3\text{ s}]$. Channels contaminated by eye-motion or muscles artefacts were automatically rejected and trials with more than 50 bad channels were excluded. An average reference transformation was applied on the artefact-free trials to obtain reference-independent potentials. In the end, epochs were averaged per subject/individual/stimulus, leading to two tensors of size 53 subjects $\times$ 1450 time samples $\times$ 124 channels.

5.4.3 Normalization

This normalization procedure was applied per subject and channel and was carried out after averaging over trials.

It consists in a sliding window procedure. The normalization of a given time sample of an EEG signal was performed by estimating the mean and standard deviation of this EEG signal on the window containing this sample and the previous 144 samples (or less depending on the position of this sample). This procedure was repeated for each time sample. The window size is 580 ms. It was chosen because it is slightly higher than the time laps of the phenomena we wish to capture. In the end, this normalization procedure lowered down the size of the second mode of the tensor by one sample.

5.4.4 Tensor Construction

This construction is very similar to the description in section 5.3.3, except that a difference between the conditions is performed.

After pre-processing, a simple derivation of this EEG experiment yields to 2 tensors of size 53 subjects $\times$ 1450 time samples $\times$ 124 channels. The first tensor refers to the standard condition ($\underline{\mathbf{X}}_S$) and the second one to the deviant condition ($\underline{\mathbf{X}}_D$). Neuroscience studies are mostly interested by analyzing within-subject differences between the two conditions. This question can be tackled using the following protocol: a single EEG tensor ($\underline{\mathbf{X}}$) of dimension 106 samples $\times$ 1450 time samples $\times$ 124 channels is built as the first mode concatenation of $\underline{\mathbf{X}}_S - \underline{\mathbf{X}}_D$ and $\underline{\mathbf{X}}_D - \underline{\mathbf{X}}_S$. Additionally, a 106-vector $\mathbf{y}$ encoding the class membership: "Std - Dev" or "Dev - Std" is considered as second block.

5.4.5 Method

MGCCA was applied on ($\underline{\mathbf{X}}$, $\mathbf{y}$) in order to extract one component, with the constraint matrix set to the identity and $g(x) = x^2$. The tolerance of the stopping criteria ε was set to 10^{-8}. A bootstrap procedure [Efron, 1979, 1987] was performed to assess the reliability of parameter estimates. Two thousand bootstrap samples of the same size as the original data were repeatedly sampled with replacement from the original data. When a subject was sampled, it was for both conditions. MGCCA was then applied to each bootstrap sample to obtain estimates $\mathbf{w}^{K,b}$ (channel weights) and $\mathbf{w}^{J,b}$ (time weights) for $b = 1, \ldots, 2000$. We then calculated the mean and variance of the estimates over the bootstrap samples, from which we derived confidence interval with confidence level of 95% (under the assumption that the parameter estimates exhibited asymptotic normality). The resulting confidence intervals were not corrected.

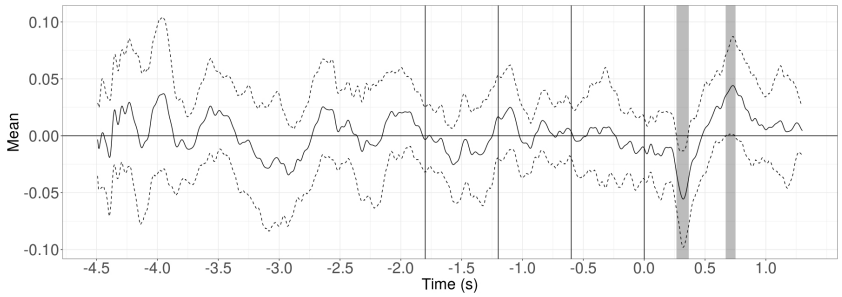
5.4.6 Results

The results are presented in Figure 5.4-9. On this figure, grey boxes for the time dimension (left panel) and black dots for the channel dimension (right panel) mark the significant weights. It appears that for both dimensions, MGCCA yields significant weights.

The significant time weights (Figure 5.4-9.a) identified after the presentation of the fourth syllable are located at [0.268; 0.364] ms and [0.672; 0.748] ms (corresponding respectively to 25 and 7 samples). These two responses were already reported in [Dehaene-Lambertz and Dehaene, 1994]. The first response corresponds to an early automatic response called mismatch response whose latency is usually around 300 ms at this age and the second one to a late response between 600 and 1000 ms when attention is attracted by the change of syllable.

Concerning the significant channel weights (Figure 5.4-9.b), their topography consists in two clusters of opposite sign, one over the frontal channels and the other over the posterior channels. In regard of the time weights, these results suggest that the polarity is inverted between the first response (negative time weights around 300 ms) and the second response (positive time weights around 700 ms). This polarity reversal has already been reported in [Basirat et al., 2014].

(a) Time dimension



(b) Channels dimension

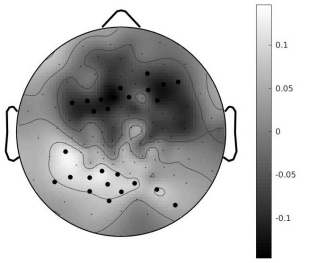


Figure 5.4-9 – MGCCA mode weights. (a): mode-2 weight vector (time dimension). On this left panel, vertical black lines indicate the onset of each syllable (-1.8, -1.2, -0.6 and 0 s). Moreover, continuous black line correspond to the mean of the weights across all bootstrap samples and dashed lines to the limits of the confidence interval. Grey boxes are associated with weights significantly different from 0. (b): mode-3 weight vector (channel dimension). On this right panel, black dots are associated with weights significantly different from 0.

5.4.7 Conclusion

In this EEG study, MGCCA managed to locate the relevant information in time and space in order to discriminate between the deviant and standard stimuli. Again, the Kronecker constraint added in order to deal with multiway data allowed to interpret the mode weight vectors. Moreover, here, the number of weights to estimate is only $751 + 124$ versus 751×124 with a non-multiway method, which is tremendously lower in this case, by a factor of 100.

The use of EEG for infants is particularly tricky as signals are much noisier than when they are recorded on adults. In particular, it is really hard to place the EEG-headset at the same position for every baby. In the case of a multiway analysis, this can be troublesome as the effect might lose its «parallel nature» across subjects due to this misplacement. In order to strengthen the multiway effect detected, a realignment procedure might be used upstream of the multiway analysis.

Here, a two-block scenario was conducted. In this setting, MGCCA is equivalent to another well known multiway method, N-way PLS (see appendix A). In the next section, that presents another EEG study, the potential of MGCCA with more than two blocks will be explored.

5.5 The Phoneme Encoding data, an EEG study

5.5.1 Description of the Study

The objective of this study was to identify whether the infant's brain encodes the phonetic features used by linguists to describe speech. Twenty four different consonant-vowel syllables were presented in a randomized order every 1000 ms during experimental sessions of 1 hour approximately. Brain responses were recorded at 500 Hz with a high-density electro-encephalographic net comprising 256 channels. In this study, two distinct phonetic features were considered for the consonant: the Manner Of Articulation (MOA) and the Place Of Articulation (POA). Consonants are divided into two classes for the MOA: the obstruent (/b/,/d/,/g/) vs. the sonorant (/m/,/n/,/ɲ/). The obstruent consonants are formed by obstructing airflow in contrast with the sonorants which have no such obstruction and

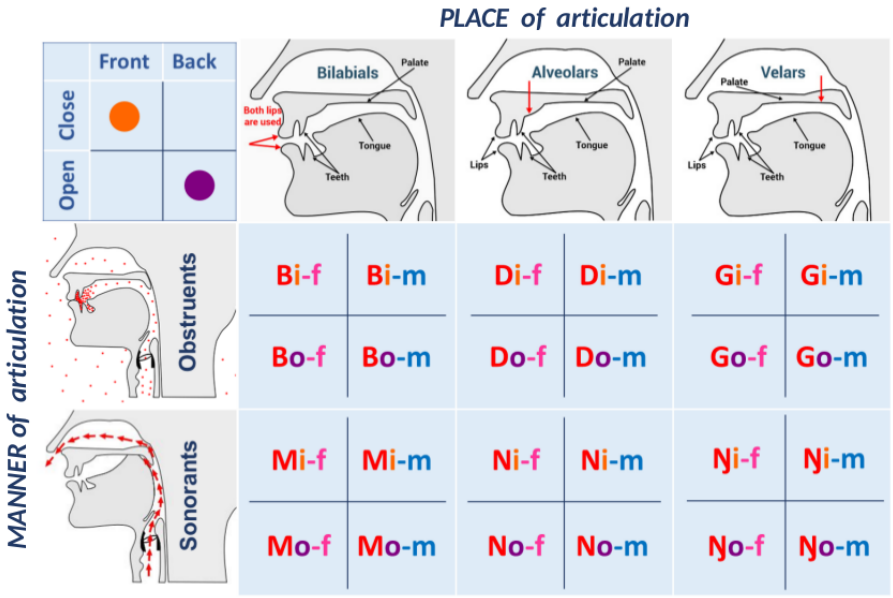


Figure 5.5-10 – Figure taken from [Gennari and Dehaene-Lambertz, 2019]. Rows correspond to the phonetic feature called the Manner Of Articulation (MOA) and columns to the Place Of Articulation (POA). For the MOA, two classes of consonants are considered, either obstruents or sonorants. For the POA, three classes: bilabials, alveolars and velars. The 6 different consonants studied are followed by either a /i/ or a /o/ and pronounced by either a female (f) or a male (m) voice.

thus resonate. For the POA, three classes: bilabials (/b/, /m/; articulated with both lips), alveolars (/d/, /n/; articulated with the tongue against the superior alveolar ridge) and velars (/g/, /ŋ/; articulated with the back of the tongue). Each consonant is followed by two possible vowels (/i/ and /o/) and pronounced by two voices (male or female) creating the 24 different syllables. The 24 stimuli are summarized in Figure 5.5-10.

5.5.2 Data Preprocessing

The neural signals were pre-processed in a similar way as described in [Dehaene-Lambertz and Dehaene, 1994]. The correct signal was then divided in epochs $[-0.2\text{ s} ; 1.4\text{ s}]$ around syllable onsets (i.e. 800 samples). An average reference transformation was applied on the artifact-free trials to obtain reference-independent potentials. Then, epochs were averaged per subject/individual/stimulus. For each subject, channel and syllable, averages were normalized thanks to a sliding window procedure, with a window of 290 samples (time range of 580 ms), as described in section 5.4.3. After pre-processing, a simple derivation of this EEG experiment yields 24 tensors of size 25 subjects $\times$ 799 time samples $\times$ 252 channels.

5.5.3 Analysis with MGCCA

MGCCA was applied on all 24 tensors $\underline{\mathbf{X}}_l \in \mathbb{R}^{799\times25\times252}$, $l \in [\![1;24]\!]$. The design matrix $\mathbf{C} \in \mathbb{R}^{24\times24}$ was constructed such that all tensors were connected to each other: $(\mathbf{C})_{ij} = 1$, if $i \neq j$. Constraint matrices were all set to the identity, $\mathrm{g}(x) = x^2$ (a.k.a. factorial scheme) and 5 components were

extracted for each block, with orthogonality at the level of the components. Tolerance of the stopping criteria ε was set to 10^{-8}.

All phonemes are characterized by similar time profiles. Thus MGCCA will be carried out by considering the time mode as common between tensors and we expect MGCCA to extract specific channel and subject weights for each phoneme.

A bootstrap procedure [Efron, 1979, 1987] was performed to assess the reliability of parameter estimates. Two thousand bootstrap samples of the same size as the original data were repeatedly sampled with replacement from the original data. MGCCA was applied to each bootstrap sample to obtain estimates $\mathbf{w}_l^{K,(r),b}$ (channel weights) and $\mathbf{y}_l^{(r),b}$ (time block components) for $r = 1, \ldots, 5$, $b = 1, \ldots, 2000$ and $i = 1, \ldots, 24$. We then calculated the mean and variance of the estimates over the bootstrap samples, from which we derived confidence interval with confidence level of $1 - \alpha/n_t$ (under the assumption that the parameter estimates exhibited asymptotic normality), where $n_t = (799 \text{ time samples} + 252 \text{ channels}) \times 5 \text{ components} \times 24 \text{ phonemes}$ is the number of tests undertaken. The procedure is similar to a Bonferroni correction. A coefficient is declared robust when zero is not included in its confidence interval.

5.5.4 Results

Only the components exhibiting at least one robust coefficient in its channel mode and time component were considered. This reduces the analysis to the first three components. Moreover, the third component exhibited robust time samples only in the pre-stimulus time range and thus was not considered either.

For the first component, (Figure 5.5-11.a) represents the mean of the time component across all syllables (solid line), and subdivided between obstruent (/b/,/d/,/g/: dot and dash) and sonorant (m/,/n/,/ɲ/: dash) phonemes. Grey areas indicate robust time component elements across all syllables simultaneously. These robust time ranges ([0.190; 0.502], [0.736; 1.188] s) correspond to two responses already reported in [Dehaene-Lambertz and Dehaene, 1994]: first an early auditory response originating from the associative auditory areas and second a late response between 600 and 1000 ms probably involving amodal frontal and top-down re-entrant activation of the auditory cortices, as explained in section 5.4.6. Similarly, main head on (Figure 5.5-11.b) represents the mapping of the mean of the channel weights across all syllables, with black dots corresponding to robust channel weights across all syllables simultaneously. This map consists in two clusters of opposite sign, one over the frontal channels and the other over the posterior channels. In regard of the time component elements, these results suggest that the polarity is reversed between the first response (positive time component elements for [0.190; 0.502]s) and the second response (negative component elements for [0.736; 1.188] s). This polarity reversal has already been reported in [Basirat et al., 2014].

Principal Component Analysis (PCA) was performed on each matrix $\mathbf{R}^{(b)} \in \mathbb{R}^{24\times(799+252)}$, $b = 1, \ldots, 2000$ where each line is composed by the row vector $\left[\mathbf{y}_l^{(1),b^\top}/\|\mathbf{y}_l^{(1),b}\|_2, \mathbf{w}_l^{K,(1),b^\top}\right]$ associated with phoneme $l = 1, \ldots, 24$. In order to compare the results provided by each PCA, it is possible to resort to Procrustes rotation [Kabsch, 1976] to fit the PCA configurations obtained from each $\mathbf{R}^{(b)}$ toward the fixed reference configuration obtained from the original data set. Results are presented in Figure (5.5-11.c). It appears that the first component explains the phonetic feature called the manner of

articulation (obstruent/sonorant) and the second component explains the voice (female/male).

Then, matrices $\mathbf{R}^{(b)},\ b = 1, \ldots, 2000$ were averaged element-wise and a t-test was performed, column-wise, considering the groups composed of manners (obstruent vs. sonorant). P-values were adjusted with the false discovery rate method [Benjamini and Hochberg, 1995]. It led to three significant time intervals ([0.106, 0.200], [0.242, 0.458] and [0.758, 0.926]s) as shown by stars on Figure (5.5-11.a). Twenty four channels were also declared as significant and are represented on the bottom left corner of Figure (5.5-11.b). It indicates that the differences between sonorant and obstruent phonemes appears around 300 ms and at 800 ms and mainly involved the left hemisphere where language is processed.

As previously, PCA was carried out on the second set of MGCCA components. It appears that there were still robust time and channel elements for each phoneme (results not shown). However, it was less obvious to understand which phenomena were captured.

5.5.5 Conclusion

In this EEG study, MGCCA managed to extract, in a unsupervised way, a component bearing a phonetic feature called the Manner Of Articulation, along with the gender of the voice heard by the infants. This study was particularly challenging as 24 blocks were involved. By linking them all together, MGCCA allowed to explore and found relevant information to characterize the different phonemes. Moreover, the flexibility of MGCCA was also shown as this time, the dimension used in order to join the tensors was not the subjects dimension but the time dimension.

As mentioned in section 5.3.6, instead of dealing with 24 tensors, only one tensor of size 25 subjects $\times$ 799 time samples $\times$ 252 channels $\times$ 24 phonemes could have be analyzed. But again, MGCCA allows to extract effects that are both common to all blocks and specific to each one of them. If we had worked with a four order tensor, it would have been impossible to locate in time and space the specificity of each phoneme. Nonetheless, in the study presented, an *ad hoc* procedure had to be used after MGCCA in order to analyze the phonemes specificity. Future works comprise including this procedure into MGCCA. A way is to introduce a super-block [Garali et al., 2017, Tenenhaus et al., 2017] which is a concatenation of all the blocks considered. This allows to create a space common to all variables. In this study, it would have allowed to represent all the phonemes in a common space.

More recently [Girka et al., 2020] analyzed this data set in the framework of sparse Rank-R multiway logistic regression. This study focus on predicting the MOA and in this context raised interesting results. However, the POA seems still out of reach. One way to investigate further this issue is to get back to the trials. Maybe the specificity of the POA is so thin and present so much individual variances that it is impossible to grasp by averaging across trials. In the case of a trial analysis, it is impossible to create a four order tensor as trials cannot be a mode. Indeed, for two different subjects, the same trial cannot be considered common as it was observed in different conditions. Then MGCCA is of hands in this condition but definitely, before that, a sparse extension has to be developed because trials would considerably increase the tensor sizes.

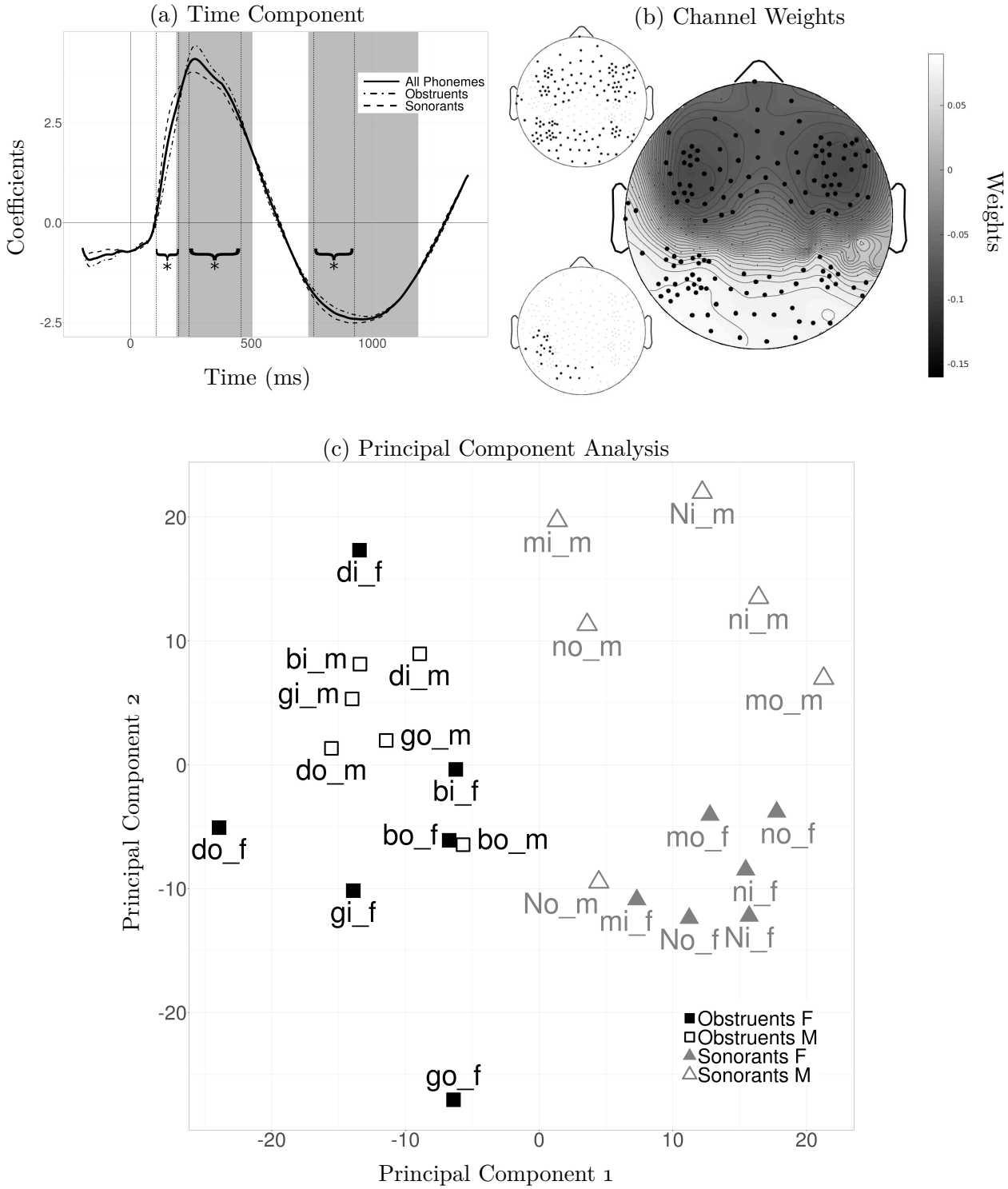


Figure 5.5-11 – Results of the first component of MGCCA on the EEG data. (a): Mean of the time component across all syllables (solid line), obstruent (/b/,/d/,/g/: dotdash) and sonorant (m/,/n/,/ɲ/: dash) phonemes. Grey areas indicate robust time component elements for all syllables simultaneously. Dashed boxes with brackets and stars indicate significant difference in mean between obstruent and sonorant phonemes. (b): Main head figure is associated with the mean of the channel weights across all phonemes where black dots refer to robust channel coefficients for all syllables simultaneously. Top left-hand corner head shows robust channel coefficients again. Bottom left-hand corner head presents significant channel coefficients difference in mean between obstruent and sonorant phonemes. (c): PCA was applied on matrices $\mathbf{R}^{(b)}$, $b = 1, \ldots, 2000$. Squares and triangles represent the mean of the first two PCs for all syllables across all bootstrap samples. The name of the phoneme come along with first the consonant (/b/,/d/,/g/, m/,/n/,/ɲ/; the last one is referred as N on the figure), then the vowel (/i/ or /o/) and then the voices (male or female). Squares/triangles refer to obstruent/sonorant phonemes and filled/empty shapes refer to female and male voices.

5.6 Conclusion

In this Chapter, the versatility and usefulness of RGCCA and MGCCA was investigated on five multiblock and/or multiway datasets. Both methods have presented interesting results in various fields such as Imaging genetic, Raman Microscopy, Alzheimer's Disease and EEG data. In all these analyses, a focus is made on the interpretation of the results by visualizing per block either the weight vectors, the mode weight vectors or the mode components. Mode weights/components are only available in the framework of MGCCA thanks to its Kronecker constraint which enables to interpret the effects of each mode separately. Moreover, this Kronecker constraint drops down, sometimes tremendously, the number of coefficients to estimate.

In almost all applications, we investigate the simplest situation where $\mathbf{M}_l$, $\mathbf{M}_l^K$ and $\mathbf{M}_l$ are identity matrices, leading to unit-norm weight/mode-weight vectors. In the case of EEG data for example, it would have been interesting to add specific proximity constraints at the level, for instance, of the electrodes by considering $\mathbf{M}_l^K = \frac{\mathbf{L}^K}{\lambda}$ and $\mathbf{M}_l^J = \mathbf{I}_J$, where $\mathbf{L}^K$ is a Laplacian matrix based on a specific definition of electrodes adjacency and $\lambda \in \mathbb{R}^+$ is a tuning parameter which modulates the importance of this constraint. This type of penalty has been investigated in the context of multiway Fisher Discriminant Analysis [Le Brusquet et al., 2015].

Furthermore, in order to highlight specific information in the mode weights, a possible strategy is to integrate within the MGCCA optimization problem (structured) sparsity constraint to any or all dimensions as explained in [Kanatsoulis et al., 2019, Löfstedt et al., 2016] or in Chapter 4.

* * *
* *
*

General Conclusions and Perspectives

Multiblock methods have encountered a renewed interest in the past few years as a result of the emergence of new technologies allowing the collection of various measurements on the same set of individuals. Each type of measurement is, in itself, difficult to analyze and dedicated algorithms are required to capture its overall complexity. Hence, multiblock data analysis methods have to evolve to handle both each source in the suitable way and find interactions between them. In this context, we have enhanced current data integration methods with various improvements and extensions: sequential to global, matrix to higher order tensors, variable selection to structured variable selection. We used a systematic approach that allows designing globally convergent algorithms for the methods proposed in this manuscript.

Contributions

The RGCCA framework is the foundation of our developments and efforts have been made to extend this framework in several directions:

- **From sequential to global.** Global RGCCA has been proposed as an alternative to sequential RGCCA. Global RGCCA allows extracting all the block components simultaneously through a single and very simple optimization problem. The global RGCCA algorithm is globally convergent under mild conditions.

- **From matrix to higher order tensors.** Multiway Generalized Canonical Correlation Analysis (MGCCA) has been proposed as an extension of RGCCA to higher order tensors. Sequential and global strategies have been designed for extracting several components per block. The different variants of the MGCCA algorithm are globally convergent under mild conditions.

- **From sparsity to structured sparsity.** The core of the SGCCA algorithm (initially proposed in [Tenenhaus et al., 2014]) has been improved. It provides a much faster globally convergent algorithm. The SGCCA algorithm has been extended to handle structured sparse penalties.

All these developments have been evaluated on simulation experiments and/or real studies and were (or will be) included in the RGCCA package (freely available at CRAN : cran.r-project.org).

Perspectives

Global procedures. In Chapters 2 and 3, sequential and global RGCCA/MGCCA have been benchmarked on simulation experiments and led to very similar results. Work in progress includes to compare and assess the efficiency of these methods on real multiblock/multiway datasets. This benchmark will include state-of-the-art competitors as JIVE, SCA (see section 1.2.3) and Generalized Structured Component Analysis (GSCA) [Hwang and Takane, 2004].

Regardless of this comparison results, the true asset of the global approaches is that only one optimization problem needs to be solved to extract all components simultaneously. Especially, it unleashes the possibility of adding more constraints across the different component levels. For example, when sparse constraints are added to the global RGCCA and MGCCA criteria, they can take the form of an ℓ_1 penalty on the whole block matrix $\mathbf{V}_l$ for RGCCA and $\mathbf{V}_l = \mathbf{V}_l^K \odot \mathbf{V}_l^J$ for MGCCA. This would allow sparsity to spread among variables and component levels (and modes for MGCCA). Another possibility is to apply a group-LASSO penalty, where each variable, over each component level, form a group. This would result in the selection of entire rows of the block weight matrices. Such ideas were presented in the context of CCA in [Kanatsoulis et al., 2019].

Convergence study. The optimization framework used to maximize a multi-convex function, described in Chapter 1 and used all along this document, is very simple and provides a systematic strategy to design globally convergent algorithms. However, several variations of this algorithm are possible. Within the BCA framework, the natural update is to move through the blocks in a cyclic way. But it is also possible to select the block that seems most in need of improvement or even choose the blocks in a random order.

The global convergence of an algorithm relies on the existence and uniqueness of the update. Nonetheless, when the uniqueness is not satisfied (point-to-set maps), the global convergence can still be studied using the Zangwill's theory [Zangwill, 1969].

This general optimization framework cannot be used as such to demonstrate the global convergence of the Structured SGCCA algorithm. Indeed, the structure of this algorithm is different from the others as it is composed of an "inner loop", where the parameter μ that regulates the amount of the penalty associated with the feasible solutions is fixed, and an "outer loop", where μ is gradually increased to enforce the solution to be in the feasible set. The convergence of the "inner loop" can still be studied with the Meyer's theory. However, it needs to be shown that the sequences generated by this "inner loop" lie in a compact set, which is verified when the objective function is coercive. This approach is similar to the work undertaken in [Chi et al., 2013] to prove the global convergence of the distance majorization algorithm. In this article, convergence of the "outer loop" is also studied. Work in progress includes adapting this convergence study to our Structured SGCCA algorithm.

Constraints & Parameters. In the RGCCA framework, an ℓ_2-norm constraint is imposed and involves a positive definite matrix $\mathbf{M}_l$. As explained in Chapter 1, this positive definite matrix usually

equals to $\tau_l \mathbf{I} + (1-\tau_l) I^{-1} \mathbf{X}_l^\top \mathbf{X}_l$, where the shrinkage parameter τ_l interpolates smoothly between maximizing the covariance and maximizing the correlation. MGCCA offers the possibility to define two positive definite matrices: $\mathbf{M}_l^J$ and $\mathbf{M}_l^K$, one per mode, which allows to impose dedicated constraint to each mode depending on their nature. This flexibility is rather unexplored in this document. In the case of EEG data for example, it would have been interesting to add specific proximity constraints at the level, for instance, of the electrodes by considering $\mathbf{M}_l^K = \frac{\mathbf{L}^K}{\lambda}$ and $\mathbf{M}_l^J = \mathbf{I}_J$, where $\mathbf{L}^K$ is a Laplacian matrix based on a specific definition of electrodes adjacency and $\lambda \in \mathbb{R}^+$ is a tuning parameter which modulates the importance of this constraint. This type of penalty has been already investigated in the context of multiway Fisher Discriminant Analysis [Le Brusquet et al., 2015].

Along with such constraints, parameters have to be tuned. When the analysis is oriented towards the prediction of a specific block, which is the case of almost all the analyses presented in Chapter 5, parameters can be set based on the cross-validated prediction error. However, for unsupervised multiblock analysis, permutation based strategies [Witten et al., 2009] can be used.

Another parameter to tune for all multiblock component methods is the number R of components to extract. As mentioned above, depending on the situation, either a cross-validation or a permutation based procedure can be used to set this number. It can also be fixed arbitrarily in order to explore the multi-dimensionality of a dataset. A last possibility is to estimate R before applying a multiblock method. In [Bro and Kiers, 2003], the core consistency diagnosis (CONCORDIA) is proposed to estimate the number of components to extract in a CP model. In the context of MGCCA, CONCORDIA can be applied to every higher-order cross-covariance matrices $\mathcal{P}_{lk} = \underline{\mathbf{P}}_l \times_1^1 \underline{\mathbf{P}}_k$, for $1 \leq l < k \leq L$ (see section 1.3.4.3 for more details about the operator $\times_1^1$) in order to estimate its number of components to extract R_{lk} and define $R = \min_{1 \leq l < k \leq L} R_{lk}$.

Missing values. In the context of multi-source datasets, some measurements can be missing for some sources. A first possibility to handle such an issue is to work only with observations without any missing values across the sources. However, this could tremendously decrease the number of available observations and lead to a poorly informative model. Another strategy is to impute these missing data based on the sources that are complete as in [Zhu et al., 2018]. Moreover, these missing values can appear in different form. For example, in the ADNI dataset, see section 5.2, some individuals were not present at every visit, which leads to entire mode-2 fibers missing in the tensor. Sometimes it can be a whole slice of a tensor that is missing, or a row of a matrix or only isolated variables for some subjects. Taking into account the nature of each block and the global structure of multi-source dataset is mandatory to properly handle missing values.

Link between blocks. In RGCCA, the definition of the design matrix $\mathbf{C}$ specifying which links are taken into account is left to the user. A perspective is to iteratively estimate them inside the algorithm. Moreover, the nature of the links can be changed from a linear correlation to a partial correlation, a non-linear link or a link of causality like in the framework of GSCA [Hwang and Takane, 2004]. The estimation and the nature of the interactions between blocks in RGCCA and in general in multiblock methods is a relatively unexplored field that calls for new developments.

Appendices

Appendix **A**

MGCCA as a generalization of several methods

It has been shown that the RGCCA framework allows to recover several important multivariate analysis methods [Tenenhaus et al., 2017]. In the same vein, the MGCCA optimization criterion (3.1) allows to recover some well known multiway methods. Let $\underline{\mathbf{X}}$ be a tensor of dimension $I \times J \times K$ and let $\mathbf{X} = [\mathbf{X}_{..1}, \ldots, \mathbf{X}_{..K}]$ be its first mode matricization of dimension $I \times JK$.

A.1 Normalized PARAFAC

PARAFAC is an acronym for PARAllel FACtor Analysis and has been designed at the same time by [Harshman, 1970] and [Carroll and Chang, 1970]. In this section, a rank-one normalized PARAFAC model is considered [ten Berge, 1993]:

$$\begin{aligned} &\min_{\mathbf{y},\mathbf{w}} \ \|\mathbf{X} - \mathbf{y}\mathbf{w}^\top\|_F^2 \\ &\text{s.t.} \quad \mathbf{w}^\top\mathbf{w} = 1 \text{ and } \mathbf{w} = \mathbf{w}^K \otimes \mathbf{w}^J \end{aligned} \tag{A.1}$$

For a fixed vector $\mathbf{w}$, the solution of optimization problem (A.1) is obtained for $\mathbf{y} = \mathbf{X}\mathbf{w}$. Therefore, from the following identities: $\|\mathbf{X} - \mathbf{y}\mathbf{w}^\top\|_F^2 = \|\mathbf{X} - \mathbf{X}\mathbf{w}\mathbf{w}^\top\|_F^2 = \mathrm{Tr}\left(\mathbf{X}^\top\mathbf{X}\right) - \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w}$, optimization problem (A.1) is equivalent to:

$$\begin{aligned} &\max_{\mathbf{w}} \ \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w} \\ &\text{s.t.} \quad \mathbf{w}^\top\mathbf{w} = 1 \text{ and } \mathbf{w} = \mathbf{w}^K \otimes \mathbf{w}^J, \end{aligned} \tag{A.2}$$

which is a special case of MGCCA ($\mathrm{g}(x) = x$ and $\mathbf{X}$ links to itself). Hence, it can be solved using Algorithm 4.

A.2 Multilinear Partial Least Squares Regression

In the framework of N-way Partial Least Squares 2 (NPLS2) [Bro, 1996], we consider the tensor $\underline{\mathbf{X}}$ and a response matrix $\mathbf{Y}$ of dimension $I \times L$. NPLS2 is defined as the following optimization problem:

$$\begin{aligned} &\max_{\mathbf{a},\mathbf{q}} \ \mathbf{a}^\top\mathbf{X}^\top\mathbf{Y}\mathbf{q} \\ &\text{s.t.} \quad \mathbf{a}^\top\mathbf{a} = \mathbf{q}^\top\mathbf{q} = 1 \text{ and } \mathbf{a} = \mathbf{c} \otimes \mathbf{b} \end{aligned} \tag{A.3}$$

which is special case of (3.1), with function g set to the identity function, $\mathbf{M}_1$ and $\mathbf{M}_2$ are defined as the identity matrix, $c_{11} = c_{22} = 0$, $c_{12} = c_{21} = 1$ and the second block is a matrix and not a tensor. Monotone convergence properties of NPLS2 are discussed in [Hanafi et al., 2015] and [Ouertani et al., 2014]. In the case of a univariate response, NPLS1 is recovered.

A.3 Link between PARAllel FACtor analysis (PARAFAC) and MGCCA

The goal of this section is to show that optimization problem (3.24), at the heart of the global MGCCA algorithm, is in fact a PARAFAC problem with specific constraints.

As $\mathbf{P}_l = \left[\mathbf{P}^l_{..1}, \ldots, \mathbf{P}^l_{..K_l}\right]$, we can fold this matrix by stacking all the frontal slices in a third mode of dimension K_l and we get a tensor $\underline{\mathbf{P}}_l$ such that $\underline{\mathbf{P}}_l = I^{-1}\underline{\mathbf{X}}_l \times_2 \mathbf{M}_l^{J^{-1/2}} \times_3 \mathbf{M}_l^{K^{-1/2}}$, where $\times_2$ and $\times_3$ are respectively the second and third mode product, which is an extension of the inner product for tensors (see section 1.3.4.2 for more details). That being said, optimization problem (3.24) is equivalent to a really well-known optimization problem:

$$\left(\hat{\mathbf{V}}_l^J, \hat{\mathbf{V}}_l^K\right) = \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmin}} \left\| \underline{\mathbf{P}}_l - [\![\mathbf{Z}_l, \tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K]\!] \right\|_F^2 \tag{A.4}$$

where $\|.\|_F$ is the equivalent of the Frobenius norm for matrices (cf. section 1.3.4.3). For more explanation about the notation $[\![\mathbf{Z}_l, \tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K]\!]$, see section 1.4.1.

So optimization problem (A.4) is a rank-R PARAFAC model of $\underline{\mathbf{P}}_l$ where the factor matrix associated to the first mode is fixed and with orthogonality constraints on the two other factor matrices.

To show the equivalence between (3.24) and (A.4), the above optimization problem (A.4) can be rewritten by unfolding each tensor in the first mode as explained in (1.27) :

$$\left(\hat{\mathbf{V}}_l^J, \hat{\mathbf{V}}_l^K\right) = \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmin}} \left\| \mathbf{P}_{l(1)} - \mathbf{Z}_l \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)^\top \right\|_F^2 \tag{A.5}$$

Using properties of the Trace operator, the fact that $\mathbf{P}_{l(1)} = \mathbf{P}_l$ and the property (1.17) of the Khatri-Rao product combined with the constraints, we have that:

$$\begin{aligned}
\left(\hat{\mathbf{V}}_l^J, \hat{\mathbf{V}}_l^K\right) &= \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmin}} \operatorname{Tr}\left(\mathbf{P}_l^\top \mathbf{P}_l\right) - 2\operatorname{Tr}\left(\mathbf{P}_l^\top \mathbf{Z}_l \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)^\top\right) + \operatorname{Tr}\left(\left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right) \mathbf{Z}_l^\top \mathbf{Z}_l \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)^\top\right) \\
&= \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmin}} - 2\operatorname{Tr}\left(\left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right) \mathbf{Z}_l^\top \mathbf{P}_l\right) + \operatorname{Tr}\left(\mathbf{Z}_l^\top \mathbf{Z}_l \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)^\top \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)\right) \\
&= \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmin}} - 2\operatorname{Tr}\left(\mathbf{Z}_l^\top \mathbf{P}_l \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)\right) + \operatorname{Tr}\left(\mathbf{Z}_l^\top \mathbf{Z}_l \left(\mathrm{I}_R \star \mathrm{I}_R\right)\right) \\
&= \underset{(\tilde{\mathbf{V}}_l^J, \tilde{\mathbf{V}}_l^K) \in \Omega_l^J \times \Omega_l^K}{\operatorname{argmax}} \operatorname{Tr}\left(\mathbf{Z}_l^\top \mathbf{P}_l \left(\tilde{\mathbf{V}}_l^K \odot \tilde{\mathbf{V}}_l^J\right)\right),
\end{aligned} \tag{A.6}$$

which is exactly optimization problem (3.24). The classical algorithm used to solve (A.4) is an Alternative Least Squares (ALS) algorithm. The PARAFAC-ALS algorithm alternates on each factor matrices by updating them in turn with each update being the solution of a Least Squares (LS) problem such as (A.5). Here, a same alternate procedure was proposed for global MGCCA.

A.4 Link between Coupled Matrix Tensor Factorization (CMTF) and MGCCA

Let us introduce the following fourth order tensors $\mathcal{P}_{lk} = \underline{\mathbf{P}}_l \times_1^1 \underline{\mathbf{P}}_k$, for $1 \leq l < k \leq L$ of dimension $J_l \times K_l \times J_k \times K_k$. $\underline{\mathbf{P}}_l$ was introduced in equation (A.4). So $\mathcal{P}_{lk}$ contains all 2-by-2 inner products between every mode-1 fibers of $\underline{\mathbf{P}}_l$ and $\underline{\mathbf{P}}_k$. We want to show that solving the global MGCCA optimization criterion (3.19) (in the case of function g being the identity) under both constraints (3.20) and (3.21) is equivalent to solving a Coupled Matrix Tensor Factorization (CMTF) problem based on coupling all $\mathcal{P}_{lk}$ tensors for $1 \leq l < k \leq L$ along all 4 modes. This CMTF problem can be formulated as follows:

$$\underset{\mathbf{V}_1^J, \mathbf{V}_1^K \ldots, \mathbf{V}_L^J, \mathbf{V}_L^K}{\operatorname{argmin}} \sum_{k,l=1}^{L} c_{lk} \|\mathcal{P}_{lk} - [\![\mathbf{V}_l^J, \mathbf{V}_l^K, \mathbf{V}_k^J, \mathbf{V}_k^K]\!]\|_F^2 \tag{A.7}$$

$$\text{s.t. } {\mathbf{V}_l^J}^\top \mathbf{V}_l^J = {\mathbf{V}_l^K}^\top \mathbf{V}_l^K = \mathrm{I}_R, l = 1, \ldots, L \quad . \tag{A.8}$$

First, let us notice that working on the Frobenius norm of the tensors or on the Frobenius norm of their mode-1 matricization is the same:

$$\|\mathcal{P}_{lk} - [\![\mathbf{V}_l^J, \mathbf{V}_l^K, \mathbf{V}_k^J, \mathbf{V}_k^K]\!]\|_F^2 = \| \left(\mathcal{P}_{lk}\right)_{(1)} - \left([\![\mathbf{V}_l^J, \mathbf{V}_l^K, \mathbf{V}_k^J, \mathbf{V}_k^K]\!]\right)_{(1)} \|_F^2 \tag{A.9}$$

Moreover, it is possible to show that (cf. [Kolda and Bader, 2009]):

$$\left([\![\mathbf{V}_l^J, \mathbf{V}_l^K, \mathbf{V}_k^J, \mathbf{V}_k^K]\!]\right)_{(1)} = \mathbf{V}_l^J \left(\mathbf{V}_k^K \odot \mathbf{V}_k^J \odot \mathbf{V}_l^K\right)^\top . \tag{A.10}$$

Based on (A.9) and (A.10) and thanks to similar development as in (A.6), we can show that (A.7) is equivalent to:

$$\underset{\mathbf{V}_1^J, \mathbf{V}_1^K, \ldots, \mathbf{V}_L^J, \mathbf{V}_L^K}{\operatorname{argmax}} \sum_{k,l=1}^{L} c_{lk} \operatorname{Tr}\left({\mathbf{V}_l^J}^\top \left(\mathcal{P}_{lk}\right)_{(1)} \left(\mathbf{V}_k^K \odot \mathbf{V}_k^J \odot \mathbf{V}_l^K\right)\right) \tag{A.11}$$

$$\text{s.t. } {\mathbf{V}_l^J}^\top \mathbf{V}_l^J = {\mathbf{V}_l^K}^\top \mathbf{V}_l^K = \mathrm{I}_R, l = 1, \ldots, L \quad . \tag{A.12}$$

Moreover, we have the following equality:

$$\begin{aligned} \operatorname{Tr}\left({\mathbf{V}_l^J}^\top \left(\mathcal{P}_{lk}\right)_{(1)} \left(\mathbf{V}_k^K \odot \mathbf{V}_k^J \odot \mathbf{V}_l^K\right).\right) &= \sum_{r=1}^{R} {\mathbf{v}_l^{J,(r)}}^\top \left(\mathcal{P}_{lk}\right)_{(1)} \left(\mathbf{v}_k^{K,(r)} \otimes \mathbf{v}_k^{J,(r)} \otimes \mathbf{v}_l^{K,(r)}\right) \\ &= \sum_{r=1}^{R} \mathcal{P}_{lk} \times_1 {\mathbf{v}_l^{J,(r)}}^\top \times_2 {\mathbf{v}_l^{K,(r)}}^\top \times_3 {\mathbf{v}_k^{J,(r)}}^\top \times_4 {\mathbf{v}_k^{K,(r)}}^\top , \end{aligned} \tag{A.13}$$

where the last equality comes from [Kolda, 2006], Proposition 4.3.b.

So, for now we have shown that optimization problem (A.7) can be formulated as:

$$\underset{\mathbf{V}_1^J, \mathbf{V}_1^K, \ldots, \mathbf{V}_L^J, \mathbf{V}_L^K}{\operatorname{argmax}} \quad \sum_{k,l=1}^{L} c_{lk} \sum_{r=1}^{R} \mathcal{P}_{lk} \times_1 {\mathbf{v}_l^{J,(r)}}^\top \times_2 {\mathbf{v}_l^{K,(r)}}^\top \times_3 {\mathbf{v}_k^{J,(r)}}^\top \times_4 {\mathbf{v}_k^{K,(r)}}^\top \tag{A.14}$$

$$\text{s.t. } {\mathbf{V}_l^J}^\top \mathbf{V}_l^J = {\mathbf{V}_l^K}^\top \mathbf{V}_l^K = 1, l = 1, \ldots, L. \tag{A.15}$$

As explained in [Lathauwer et al., 2000], this new formulation can be interpreted as a sum of multilinear singular value decomposition of fourth order tensors $\mathcal{P}_{lk}$, for $1 \leq l < k \leq L$. Moreover, $\mathcal{P}_{lk}$ can be interpreted as a cross-covariance matrix between $\underline{\mathbf{P}}_l$ and $\underline{\mathbf{P}}_k$.

Now, lets us focus on reformulating only the term $\mathcal{P}_{lk} \times_1 \mathbf{v}_l^{J,(r)^\top} \times_2 \mathbf{v}_l^{K,(r)^\top} \times_3 \mathbf{v}_k^{J,(r)^\top} \times_4 \mathbf{v}_k^{K,(r)^\top}$ and let us drop the superscript (r) for the sake of clarity. As explained earlier, this term can be expressed as:

$$\mathcal{P}_{lk} \times_1 \mathbf{v}_l^{J^\top} \times_2 \mathbf{v}_l^{K^\top} \times_3 \mathbf{v}_k^{J^\top} \times_4 \mathbf{v}_k^{K^\top} = \mathbf{v}_l^{J^\top} \left(\mathcal{P}_{lk}\right)_{(1)} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J \otimes \mathbf{v}_l^K\right) \tag{A.16}$$

If we recall that $\mathcal{P}_{lk} = \underline{\mathbf{P}}_l \times_1^1 \underline{\mathbf{P}}_k$ and that mode-1 unfolding consists in concatenating mode-1 fibers in the same order as their modes, then we can see that:

$$\left(\mathcal{P}_{lk}\right)_{(1)} = \left[\mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.11}^k | \mathbf{P}_{..2}^{l^\top}\mathbf{p}_{.11}^k | \dots | \mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.11}^k | \mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.21}^k | \dots | \mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.21}^k | \mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.31}^k | \dots | \mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.J_l1}^k | \mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.12}^k | \dots\right] \tag{A.17}$$

So, $\mathbf{v}_l^{J^\top} \left(\mathcal{P}_{lk}\right)_{(1)}$ is a row vector and it can be folded into a tensor of size $K_l \times J_k \times K_k$. This tensor is noted $\underline{\mathbf{P}}_{lk}$. Then:

$$\begin{aligned} \mathcal{P}_{lk} \times_1 \mathbf{v}_l^{J^\top} \times_2 \mathbf{v}_l^{K^\top} \times_3 \mathbf{v}_k^{J^\top} \times_4 \mathbf{v}_k^{K^\top} &= \underline{\mathbf{P}}_{lk} \times_1 \mathbf{v}_l^{K^\top} \times_2 \mathbf{v}_k^{J^\top} \times_3 \mathbf{v}_k^{K^\top} \\ &= \mathbf{v}_l^{K^\top} \left(\underline{\mathbf{P}}_{lk}\right)_{(1)} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \end{aligned} \tag{A.18}$$

And finally, by noticing that $\left(\underline{\mathbf{P}}_{lk}\right)_{ijk} = \mathbf{v}_l^{J^\top} \mathbf{P}_{..i}^{l^\top} \mathbf{p}_{.jk}^k$, we have

$$\begin{aligned} \mathbf{v}_l^{K^\top} \left(\underline{\mathbf{P}}_{lk}\right)_{(1)} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) &= \mathbf{v}_l^{K^\top} \begin{bmatrix} \mathbf{v}_l^{J^\top}\mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.11}^k & \mathbf{v}_l^{J^\top}\mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.21}^k & \dots & \mathbf{v}_l^{J^\top}\mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.J_k1}^k & \mathbf{v}_l^{J^\top}\mathbf{P}_{..1}^{l^\top}\mathbf{p}_{.12}^k & \dots \\ \vdots & \vdots & & \vdots & \vdots & \\ \mathbf{v}_l^{J^\top}\mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.11}^k & \mathbf{v}_l^{J^\top}\mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.21}^k & \dots & \mathbf{v}_l^{J^\top}\mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.J_k1}^k & \mathbf{v}_l^{J^\top}\mathbf{P}_{..K_l}^{l^\top}\mathbf{p}_{.12}^k & \dots \end{bmatrix} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \\ &= \begin{bmatrix} v_{l1}^K \left(\mathbf{P}_{..1}^l \mathbf{v}_l^J\right)^\top & \dots & v_{lK_l}^K \left(\mathbf{P}_{..K_l}^l \mathbf{v}_l^J\right)^\top \end{bmatrix} \begin{bmatrix} \mathbf{p}_{.11}^k & \mathbf{p}_{.21}^k & \dots & \mathbf{p}_{.J_k1}^k & \mathbf{p}_{.12}^k & \dots \\ \vdots & \vdots & & \vdots & \vdots & \\ \mathbf{p}_{.11}^k & \mathbf{p}_{.21}^k & \dots & \mathbf{p}_{.J_k1}^k & \mathbf{p}_{.12}^k & \dots \end{bmatrix} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \\ &= \left[\sum_{k=1}^{K_l} v_{lk}^K \left(\mathbf{P}_{..k}^l \mathbf{v}_l^J\right)^\top\right] \begin{bmatrix} \mathbf{p}_{.11}^k & \mathbf{p}_{.21}^k & \dots & \mathbf{p}_{.J_k1}^k & \mathbf{p}_{.12}^k & \dots \end{bmatrix} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \\ &= \left[\sum_{k=1}^{K_l} v_{lk}^K \left(\mathbf{P}_{..k}^l \mathbf{v}_l^J\right)^\top\right] \begin{bmatrix} \mathbf{P}_{..1}^k & \dots & \mathbf{P}_{..K_k}^k \end{bmatrix} \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \\ &= \left[\sum_{k=1}^{K_l} v_{lk}^K \left(\mathbf{P}_{..k}^l \mathbf{v}_l^J\right)^\top\right] \mathbf{P}_k \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \\ &= \left[\mathbf{P}_l \left(\mathbf{v}_l^K \otimes \mathbf{v}_l^J\right)\right]^\top \mathbf{P}_k \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) \\ &= \left(\mathbf{v}_l^K \otimes \mathbf{v}_l^J\right)^\top \mathbf{P}_l^\top \mathbf{P}_k \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right), \end{aligned} \tag{A.19}$$

which is exactly the one-component MGCCA criterion, so we have what we wanted to show.

We can also notice that $\left(\mathbf{v}_l^K \otimes \mathbf{v}_l^J\right)^\top \mathbf{P}_l^\top \mathbf{P}_k \left(\mathbf{v}_k^K \otimes \mathbf{v}_k^J\right) = \left(\underline{\mathbf{P}}_l \times_2 \mathbf{v}_l^{J^\top} \times_3 \mathbf{v}_l^{K^\top}\right) \times_1 \left(\underline{\mathbf{P}}_k \times_2 \mathbf{v}_k^{J^\top} \times_3 \mathbf{v}_k^{K^\top}\right) = \left(\underline{\mathbf{P}}_l \times_2 \mathbf{v}_l^{J^\top} \times_3 \mathbf{v}_l^{K^\top}\right)^\top \left(\underline{\mathbf{P}}_k \times_2 \mathbf{v}_k^{J^\top} \times_3 \mathbf{v}_k^{K^\top}\right)$.

Appendix B

Demonstration for the scalar product maximization under ℓ_1 and ℓ_2-norm constraints

This appendix undertakes the explanation of assumption (4.11) and demonstration of proposition 4.3.1 and 4.3.2, all presented in Chapter 4. For the sake of clarity, they are all recalled here.

The optimization problem of interest in this appendix is restated below:

$$\underset{\mathbf{x}\in\Omega}{\operatorname{argmax}}\ \mathbf{a}^\top\mathbf{x},$$

where $\mathbf{a}\in\mathbb{R}^J$ and $\Omega=\left\{\mathbf{x}\in\mathbb{R}^J \mid \|\mathbf{x}\|_2\leq 1 \text{ and } \|\mathbf{x}\|_1\leq s\right\}$ with $s\in\mathbb{R}_+^\star$. As shown in [Witten et al., 2009], solution of (4.8) satisfies $\mathbf{u}=\mathcal{S}(\mathbf{a},\lambda)/\|\mathcal{S}(\mathbf{a},\lambda)\|_2$, where $\lambda=0$ if $\|\mathbf{u}\|_1\leq s$ and λ is chosen such that $\|\mathbf{u}\|_1=s$ otherwise.

B.1 Assumption (4.11)

In section 4.3, the following assumption was made:

$$\operatorname{card}\left(\underset{i\in[\![1,J]\!]}{\operatorname{argmax}}\ |a_i|\right)=1.$$

This assumption is equivalent to say that the maximum value of the vector $|\mathbf{a}|$ is reached for only one element.

In order to understand this condition, an example where all the elements of $\mathbf{a}$ are equals is considered: $\mathbf{a}=(a,\ldots,a)^\top\in\mathbb{R}^J$, where $a\in\mathbb{R}_+^\star$. The fact that a is positive is not necessary but it is going to ease the discussion. Then, $\forall\lambda\in[0;a[$, the solution of (4.8) can be written as:

$$\mathbf{u}_1=\frac{\mathcal{S}(\tilde{\mathbf{a}},\lambda)}{\|\mathcal{S}(\tilde{\mathbf{a}},\lambda)\|_2}=\frac{(a-\lambda,\ldots,a-\lambda)^\top}{\sqrt{J\times(a-\lambda)^2}}=(1/\sqrt{J},\ldots,1/\sqrt{J})^\top, \tag{B.1}$$

thus, $\|\mathbf{u}_1\|_1=\sqrt{J}$. In this case, if $s\neq\sqrt{J}$, no solution of the form $\mathbf{u}=\mathcal{S}(\mathbf{a},\lambda)/\|\mathcal{S}(\mathbf{a},\lambda)\|_2$ can be found.

Another example can be studied, where the maximum element of $\mathbf{a}$ appears twice: $\mathbf{a} = (a, a, b, \ldots, b)^\top \in \mathbb{R}^J$, where $a, b \in \mathbb{R}_+^\star$ and $b < a$. This time, $\forall \lambda \in [b; a[$, the solution of (4.8) can be written as:

$$\mathbf{u}_2 = \frac{\mathcal{S}(\tilde{\mathbf{a}}, \lambda)}{\|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_2} = \frac{(a-\lambda, a-\lambda, 0, \ldots, 0)^\top}{\sqrt{2 \times (a-\lambda)^2}} = (1/\sqrt{2}, 1/\sqrt{2}, 0, \ldots, 0)^\top, \tag{B.2}$$

thus, $\|\mathbf{u}_2\|_1 = \sqrt{2}$. In this case, if $s \in \left[1; \sqrt{2}\right[$, no solution of the form $\mathbf{u} = \mathcal{S}(\mathbf{a}, \lambda)/\|\mathcal{S}(\mathbf{a}, \lambda)\|_2$ can be found.

Under the assumption (4.11), these examples never happens. Ongoing work tries to show that in the demonstration of the solutions of (4.8), these cases are linked to a null Lagrange multiplier associated to the ℓ_2-norm constraint. Hence, when assumption (4.11) is not verified, the point on $\Omega = \left\{\mathbf{x} \in \mathbb{R}^J \mid \|\mathbf{x}\|_2 \leq 1 \text{ and } \|\mathbf{x}\|_1 \leq s\right\}$ that leads to the highest covariance with $\mathbf{a}$ is its projection onto the ℓ_1-norm ball of radius s.

B.2 Proof of Proposition 4.3.1

Proposition B.2.1. *The following function, defined on* $[0; \tilde{a}_2] \mapsto \mathbb{R}^+$*, is strictly decreasing:*

$$\psi(\lambda) = \frac{\|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_1}{\|\mathcal{S}(\tilde{\mathbf{a}}, \lambda)\|_2}, \tag{B.3}$$

with $\psi(0) = \|\mathbf{a}\|_1/\|\mathbf{a}\|_2$ *and* $\psi(\tilde{a}_2) = 1$.

Proof of proposition 4.3.1. The numerator and denominator of ψ are continuous as composition of continuous functions. Moreover, for $\lambda \in [0; \tilde{a}_2]$, $\|S(\tilde{\mathbf{a}}, \lambda)\|_2 \neq 0$. Therefore, ψ is continuous as quotient of 2 non-null continuous functions.
Assuming $\tilde{a}_{p+1} = 0$, for $\lambda \in [0; \tilde{a}_2]$ it exists $k \in [\![1; J]\!]$ such that $\tilde{a}_{k+1} \leq \lambda < \tilde{a}_k$. For this specific λ, we have:

$$\|S(\tilde{\mathbf{a}}, \lambda)\|_1 = \left[\sum_{j=1}^{k} \tilde{a}_j\right] - k\lambda \tag{B.4}$$

$$\|S(\tilde{\mathbf{a}}, \lambda)\|_2^2 = \sum_{j=1}^{k} (\tilde{a}_j - \lambda)^2 = \left[\sum_{j=1}^{k} \tilde{a}_j^2\right] - 2\lambda \left[\sum_{j=1}^{k} \tilde{a}_j\right] + k\lambda^2 \tag{B.5}$$

From equations (B.4) and (B.5), the derivate of ψ is :

$$\psi'(\lambda) = \frac{1}{\|S(\tilde{\mathbf{a}}, \lambda)\|_2^2} \left(\frac{\|S(\tilde{\mathbf{a}}, \lambda)\|_1^2}{\|S(\tilde{\mathbf{a}}, \lambda)\|_2} - k\|S(\tilde{\mathbf{a}}, \lambda)\|_2\right) = \frac{1}{\|S(\tilde{\mathbf{a}}, \lambda)\|_2}(\psi(\lambda)^2 - k) \tag{B.6}$$

Moreover, the number of non-null elements of $S(\tilde{\mathbf{a}}, \lambda)$ is equal to k. We introduce $\mathbb{1}_k$, the vector of size J such that $(\mathbb{1}_k)_i = 1$ if $(S(\tilde{\mathbf{a}}, \lambda))_i \neq 0$ and $(\mathbb{1}_k)_i = 0$ otherwise. Therefore, from Cauchy-Schwarz, the inequality $\|S(\tilde{\mathbf{a}}, \lambda)\|_1 =< \mathbb{1}_k, S(\tilde{\mathbf{a}}, \lambda) >\leq \sqrt{k}\|S(\tilde{\mathbf{a}}, \lambda)\|_2 = \sqrt{(< \mathbb{1}_k, \mathbb{1}_k >}\|S(\tilde{\mathbf{a}}, \lambda)\|_2$ holds, implying that $\psi(\lambda) \leq \sqrt{k}$ and so that $\psi'(\lambda) \leq 0$. Moreover $\psi'(\lambda) = 0$ when $\mathbb{1}_k$ and $S(\tilde{\mathbf{a}}, \lambda)$ are colinear, meaning when all the k non-null elements of $S(\tilde{\mathbf{a}}, \lambda)$ are equals. Based on assumption (4.11), on $[0; \tilde{a}_2]$, this is only possible when $\lambda = \tilde{a}_2$.

So, $\forall \lambda \in [0; \tilde{a}_2[$, $\psi'(\lambda) < 0$ and $\psi'(\tilde{a}_2) = 0$, which means that ψ is strictly decreasing on $[0; \tilde{a}_2]$
Furthermore, $\psi(0) = \|\mathbf{a}\|_1/\|\mathbf{a}\|_2$, which is direct. $\psi(\tilde{a}_2) = \frac{\tilde{a}_1 - \tilde{a}_2}{\sqrt{(\tilde{a}_1 - \tilde{a}_2)^2}} = 1$.

□

B.3 Proof of Proposition 4.3.2

Proposition B.3.1. *Giving $s \in [1; \sqrt{J}]$, the assumption that $\|\mathbf{a}\|_1/\|\mathbf{a}\|_2 > s$ is made. Then, there exists a unique $i \in [\![2; J]\!]$ and a unique $\delta \in [0; \tilde{a}_i - \tilde{a}_{i+1}[$ such that $\psi(\tilde{a}_i - \delta) = s$, then δ is a root of a second degree polynomial equation.*

Proof of proposition 4.3.2. We assume that $\tilde{a}_{p+1} = 0$. Thus, ψ is strictly decreasing from $\psi(\tilde{a}_{p+1} = 0) = \|\mathbf{a}\|_1/\|\mathbf{a}\|_2 > s$ to $\psi(\tilde{a}_2) = 1$.

Moreover, $\tilde{a}_2 \neq 0$. Otherwise, $\psi(\tilde{a}_2) = \|\mathbf{a}\|_1/\|\mathbf{a}\|_2 = 1 \leq s$ as $s \in [1; \sqrt{J}]$.

It implies that for $s \in [1; \sqrt{J}]$, it exists a unique $i \in [\![2; p]\!]$ such that $\psi(\tilde{a}_i) \leq s < \psi(\tilde{a}_{i+1})$. Finally, as ψ is continuous and strictly decreasing, it exists a unique $\delta \in [0; \tilde{a}_i - \tilde{a}_{i+1}[$ such that $\psi(\tilde{a}_i - \delta) = s$.

Using the notations $l_1 = \|S(\tilde{\mathbf{a}}, \tilde{a}_i)\|_1$ and $l_2 = \|S(\tilde{\mathbf{a}}, \tilde{a}_i)\|_2$:

$$\|S(\tilde{\mathbf{a}}, \tilde{a}_i - \delta)\|_1 = \sum_{j=1}^{i} [\tilde{a}_j - (\tilde{a}_i - \delta)] = \sum_{j=1}^{i} [\tilde{a}_j - \tilde{a}_i] + i\delta = \|S(\tilde{\mathbf{a}}, \tilde{a}_i)\|_1 + i\delta = l_1 + i\delta \tag{B.7}$$

$$\|S(\tilde{\mathbf{a}}, \tilde{a}_i - \delta)\|_2^2 = \sum_{j=1}^{i} [\tilde{a}_j - (\tilde{a}_i - \delta)]^2 = \sum_{j=1}^{i} [(\tilde{a}_j - \tilde{a}_i)^2 + 2\delta(\tilde{a}_j - \tilde{a}_i) + \delta^2] = l_2^2 + 2\delta l_1 + i\delta^2 \tag{B.8}$$

Moreover, as $\psi(\tilde{a}_i - \delta) = s = \|S(\tilde{\mathbf{a}}, \tilde{a}_i - \delta)\|_1/\|S(\tilde{\mathbf{a}}, \tilde{a}_i - \delta)\|_2$, the following equality holds:

$$\|S(\tilde{\mathbf{a}}, \tilde{a}_i - \delta)\|_1^2 = s^2 \|S(\tilde{\mathbf{a}}, \tilde{a}_i - \delta)\|_2^2 \tag{B.9}$$

Incorporating (B.7) and (B.8) in (B.9) gives:

$$\delta^2[i^2 - is^2] + 2\delta l_1[i - s^2] + l_1^2 - s^2 l_2^2 = 0 \tag{B.10}$$

The goal is now to find the positive root of this second degree polynomial equation. The discriminant Δ is equal to $4s^2[s^2 - i][l_1^2 - il_2^2]$. It remains to show that Δ is positive.
First, the number of non-null elements of $S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})$ is equal to i and the Cauchy-Schwarz inequality yields $\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_1 \leq \sqrt{i}\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_2$. Second, $\psi(\tilde{a}_{i+1}) = \dfrac{\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_1}{\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_2} > s$ so $\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_1 > s\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_2$. Combining the two previous inequalities yields $(i - s^2)\|S(\tilde{\mathbf{a}}, \tilde{a}_{i+1})\|_1 > 0$ which implies that $i - s^2 > 0$. Third, from $\psi(\tilde{a}_i) = l_1/l_2 \leq s < \sqrt{i}$, we deduce that $l_1^2 - il_2^2 \leq 0$ which ensures that Δ is positive.
To conclude, the sign of $\dfrac{l_1^2 - s^2 l_2^2}{i^2 - is^2}$ corresponds to the sign of the product of the 2 roots. As this term is negative, the 2 roots have opposite signs. The single solution of $\psi(\tilde{a}_i - \delta) = s$ is:

$$\delta = \frac{-2l_1(i - s^2) + \sqrt{\Delta}}{2i(i - s^2)} = \frac{-2l_1(i - s^2) + 2s\sqrt{[s^2 - i][l_1^2 - il_2^2]}}{2i(i - s^2)} = -\frac{l_1}{i} + \frac{s}{i}\sqrt{\frac{il_2^2 - l_1^2}{i - s^2}}.$$

Using the fact that $\psi(\tilde{a}_i) = l_1/l_2$, the previous equation can be simplified as

$$\delta = \frac{\|S(\tilde{\mathbf{a}}, \tilde{a}_i)\|_2}{i} \left(s\sqrt{\frac{i - \psi(\tilde{a}_i)^2}{i - s^2}} - \psi(\tilde{a}_i) \right). \tag{B.11}$$

□

Remark. $s < \sqrt{i}$ implies that if you know the number of non-null elements you want to keep, then s is in $[1; \sqrt{i}]$.

Appendix C

Surrogate functions of structured sparse penalties and extended results

In this appendix, the concept of Sharp Quadratic Majorization is introduced, followed by the presentation of quadratic majorizing surrogate functions of several penalties. Extended results are also presented on the comparison between RGCCA, SGCCA, structured SGCCA with either the smoothing framework of [Nesterov, 2004] and proximity operators [Löfstedt et al., 2016] (PROX_SGCCA) or with the distance majorization algorithm [Chi et al., 2013] which combines two key ingredients: quadratic penalty method and MM principle (MM_SGCCA) (see section 4.4.2).

C.1 Sharp Quadratic Majorization

The goal here is to define the notion of Sharp Quadratic Majorization. Let us consider a function f defined from $\mathbb{R}$ to $\mathbb{R}$. Then, as explained in [de Leeuw and Lange, 2009], if f is differentiable at $w^0 \in \mathbb{R}$ and $a > 0$, we have the following inequality $\forall w \in \mathbb{R}$:

$$f(w) \leq f(w^0) + f'(w^0)(w - w^0) + \frac{1}{2}a(w - w^0)^2, \tag{C.1}$$

which is true if and only if $\forall w \neq w^0$:

$$a \geq \frac{f(w) - f(w^0) - f'(w^0)(w - w^0)}{\frac{1}{2}(w - w^0)^2}. \tag{C.2}$$

This condition is not needed when $w = w^0$ as $f(w^0) \leq f(w^0)$ is already true. The left part of inequality (C.1) is a surrogate function of f anchored at w^0 as it satisfies both the tangent and domination condition (see section 4.4.1.4).

Let us define the function:

$$A(w^0) = \sup_{w \neq w^0} \frac{f(w) - f(w^0) - f'(w^0)(w - w^0)}{\frac{1}{2}(w - w^0)^2}. \tag{C.3}$$

Whenever $A(w^0) < \infty$, a quadratic majorization is considered as sharp if $a = A(w^0)$. This notion comes from the fact that if we want to minimize f in a MM framework, we would repeatedly minimize the majorization function of f defined in (C.1). This leads to the update $w = w^0 - \frac{1}{a}f'(w^0)$. We realize that if we set a to its minimal value possible, the descent is optimal.

C.2 Quadratic majorizing surrogate functions of several structured sparse penalties

In this section, penalties are referred as $p(\mathbf{w})$, where $\mathbf{w} = [w_1, \dots, w_J] \in \mathbb{R}^J$, and their surrogate as $\tilde{p}(\mathbf{w}|\mathbf{w}^0)$, where $\mathbf{w}^0 = [w_1^0, \dots, w_J^0] \in \mathbb{R}^J$ is the supporting point of the surrogate function. For all the penalties considered (except SparseStep and Smoothed ℓ_q-regularization), when at the supporting point $\mathbf{w}^0$, the penalty is not differentiable, it is not possible to find a quadratic majorizing surrogate. As these points can be identified quite easily, they are not mentioned again in the rest of this appendix. This is troublesome as these supporting points correspond to sparse vectors. This issue is evoked in the discussion of Chapter 4.

C.2.1 Least Absolute Shrinkage and Selection Operator (LASSO)

In order to find a sharp quadratic majorization for the LASSO penalty, $A(w^0)$ has to be computed. As presented in [de Leeuw and Lange, 2009], in the case where $w^0 > 0$, $f'(w^0) = +1$ and (C.3) becomes:

$$\sup_{w \neq w^0} \frac{|w| - w}{\frac{1}{2}(w - w^0)^2} = \frac{1}{|w^0|}. \tag{C.4}$$

Similarly, it is possible to show that in the case where $w^0 < 0$:

$$\sup_{w \neq w^0} \frac{|w| + w}{\frac{1}{2}(w - w^0)^2} = \frac{1}{|w^0|}. \tag{C.5}$$

So in the end, $\forall w^0 \in \mathbb{R}^\star$, $A(w^0) = 1/|w^0|$. This lead to the following sharp quadratic majorization $\forall w^0 \in \mathbb{R}_+^\star$, $\forall w \in \mathbb{R}_+$:

$$|w| \leq |w^0| + \operatorname{sign}(w^0)(w - w^0) + \frac{1}{2|w^0|}(w - w^0)^2 = \frac{1}{2}\frac{w^2}{|w^0|} + \frac{1}{2}|w^0|. \tag{C.6}$$

When $w^0 = 0$, no quadratic majorization exists [de Leeuw and Lange, 2009]. This inequality can be also found thanks to the inequality of arithmetic and geometric means (see [de Leeuw and Lange, 2009] for more details) or through the concavity of the square root function (see [Lange, 2016, O'Connell et al., 2006, Van Deun et al., 2011]).

Finally, the following surrogate function can be defined for the LASSO penalty:

$$p(\mathbf{w}) = \sum_{i=1}^{J} |w_i| = \|\mathbf{w}\|_1 \leq \frac{1}{2}\mathbf{w}^\top \Delta \mathbf{w} + \frac{1}{2}\|\mathbf{w}^0\|_1 := \widetilde{p}(\mathbf{w}|\mathbf{w}^0), \tag{C.7}$$

where Δ is a diagonal matrix of size J such that $(\Delta)_{jj} = \frac{1}{|w_j^0|}$.

C.2.2 The group-LASSO (GL) penalty

As presented in section 4.4.2.2, the non-overlapping group-LASSO, first introduced in [Yuan and Lin, 2006], is the $\ell_{1,2}$-mixed norm. By introducing a partition $\mathcal{G}$ of $[\![1; J]\!]$, the group-LASSO penalty is defined as:

$$p(\mathbf{w}) = \sum_{g \in \mathcal{G}} \|\mathbf{w}_{i_g}\|_2, \tag{C.8}$$

where $\mathbf{w}_{i_g}$ is a subvector of $\mathbf{w}$ containing only the elements of the g^{th} group of $\mathcal{G}$. The group-LASSO penalty acts like the LASSO at the group level and an entire group of variables may drop out of jointly.

By introducing $\mathbf{w}_\mathcal{G} = (\|\mathbf{w}_{i_1}\|_2, \ldots, \|\mathbf{x}_{w_G}\|_2)$, it appears that $\|\mathbf{w}_\mathcal{G}\|_1 = p(\mathbf{w})$. Hence, the surrogate function derived for the ℓ_1-norm can be used again for the group-LASSO penalty, applied on $\mathbf{w}_\mathcal{G}$:

$$\tilde{p}(\mathbf{w}|\mathbf{w}^0) := \frac{1}{2}p(\mathbf{w}^0) + \frac{1}{2}\mathbf{w}^\top \Delta \mathbf{w}, \tag{C.9}$$

where Δ is a diagonal matrix of size J such that $(\Delta)_{jj} = \frac{1}{\|\mathbf{w}^0_{j_g}\|_2}$ if variable j is in group g. This surrogate function for the group-LASSO penalty can also be found in [Van Deun et al., 2011].

C.2.3 Total Variation (TV)

As explained in section 4.4.2.2, the TV penalty, first introduced in [Rudin et al., 1992], is widely used as a tool in image denoising and restoration. It accounts for the spatial structure of images by encoding piecewise smoothness and enabling the recovery of homogeneous regions separated by sharp boundaries [Pierrefeu, 2018]. The TV penalty can be formulated as follows:

$$p(\mathbf{w}) = \sum_{j=1}^{J-1} |w_{j+1} - w_j| = \|\mathbf{D}\mathbf{w}\|_1, \tag{C.10}$$

where $\mathbf{D} \in \mathbb{R}^{J-1 \times J}$ is defined such that $(\mathbf{D})_{jj} = -1$, $(\mathbf{D})_{jj+1} = 1$ and 0 elsewhere.

In order to find a surrogate function, equation (C.6) is applied to $w_{j+1} - w_j$:

$$|w_{j+1} - w_j| \leq \frac{1}{2}|w^0_{j+1} - w^0_j| + \frac{1}{2|w^0_{j+1} - w^0_j|}\left(w_{j+1} - w_j\right)^2. \tag{C.11}$$

Then we can sum the previous expression on j from 1 to $J-1$ and get the following surrogate function:

$$\widetilde{p}(\mathbf{w}|\mathbf{w}^0) := \frac{1}{2}\sum_{j=1}^{J-1}|w^0_{j+1} - w^0_j| + \frac{1}{2}\frac{1}{|w^0_{j+1} - w^0_j|}\left(w_{j+1} - w_j\right)^2 = \frac{1}{2}p(\mathbf{w}^0) + \frac{1}{2}\mathbf{w}^\top \mathbf{D}^\top \boldsymbol{\Delta} \mathbf{D}\mathbf{w}, \tag{C.12}$$

where Δ is a diagonal matrix of size $J-1$ such that $(\Delta)_{jj} = \frac{1}{|w^0_{j+1} - w^0_j|}$.

C.2.4 Elitist LASSO

The Elitist LASSO (e-LASSO) is the $\ell_{2,1}$-mixed norm. Similarly to the group-LASSO penalty, a partition $\mathcal{G}$ of $[\![1; J]\!]$ is introduced. However, the e-LASSO penalty, instead of promoting sparsity between groups, enforces sparsity within groups. The e-LASSO penalty is defined as:

$$\begin{aligned} p(\mathbf{w}) = \sum_{g\in\mathcal{G}} \|\mathbf{w}_{i_g}\|_1^2 = \sum_{g\in\mathcal{G}} (|w_{i_1,1}| + \ldots + |w_{i_1,J_g}|)^2 &= \sum_{g\in\mathcal{G}} \left(\sum_{i=1}^{J_g} w^2_{i_g,i} + 2 \sum_{1\leq i<j\leq J_g} |w_{i_g,i}||w_{i_g,j}| \right) \\ &= \sum_{g\in\mathcal{G}} \left(\|\mathbf{w}_{i_g}\|_2^2 + 2 \sum_{1\leq i<j\leq J_g} |w_{i_g,i}||w_{i_g,j}| \right) \end{aligned} \tag{C.13}$$

where J_g is the cardinal of the g^{th} group of $\mathcal{G}$, $\mathbf{w}_{i_g}$ is a subvector of $\mathbf{w}$ containing only the elements of this group and $w_{i_g,j}$ is its j^{th} element.

To begin with, the groups are discarded to clarify the notations. The following inequality can be written:

$$\begin{aligned} 0 \leq \left(\sqrt{\frac{|w_j^0|}{|w_i^0|}}|w_i| - \sqrt{\frac{|w_i^0|}{|w_j^0|}}|w_j| \right)^2 &= \frac{|w_j^0|}{|w_i^0|}|w_i|^2 + \frac{|w_i^0|}{|w_j^0|}|w_j|^2 - 2|w_i||w_j| \\ 2|w_i||w_j| &\leq \frac{|w_j^0|}{|w_i^0|}|w_i|^2 + \frac{|w_i^0|}{|w_j^0|}|w_j|^2 \end{aligned} \tag{C.14}$$

Then, if this inequality is summed on i and j, a surrogate for the square of the ℓ_1-norm can be found:

$$\begin{aligned} 2 \sum_{1 \leq i < j \leq p} |w_i||w_j| &\leq \sum_{i=1}^{J_g} \frac{|w_i|^2}{|w_i^0|} \left(\|\mathbf{w}^0\|_1 - |w_i^0| \right) = -\|\mathbf{w}\|_2^2 + \|\mathbf{w}^0\|_1 \sum_{i=1}^{J_g} \frac{|w_i|^2}{|w_i^0|} \\ \|\mathbf{w}\|_2^2 + 2 \sum_{1 \leq i < j \leq p} |w_i||w_j| &\leq \|\mathbf{w}^0\|_1 \sum_{i=1}^{J_g} \frac{|w_i|^2}{|w_i^0|} \\ \|\mathbf{w}\|_1^2 &\leq \|\mathbf{w}^0\|_1 \sum_{i=1}^{J_g} \frac{|w_i|^2}{|w_i^0|} \end{aligned} \tag{C.15}$$

Thus, if the previous inequality is summed on groups, a surrogate function of the e-LASSO can be defined:

$$p(\mathbf{w}) = \sum_{g \in \mathcal{G}} \|\mathbf{w}_{i_g}\|_1^2 \leq \sum_{g \in \mathcal{G}} \left(\|\mathbf{w}_{i_g}^0\|_1 \sum_{i=1}^{J_g} \frac{|w_{i_g,i}|^2}{|w_{i_g,i}^0|} \right) := \widetilde{p}(\mathbf{w}|\mathbf{w}^0). \tag{C.16}$$

Such surrogate function can also be found in [Van Deun et al., 2011].

C.2.5 Octagonal Shrinkage and Clustering Algorithm for Regression (OSCAR)

The Octagonal Shrinkage and Clustering Algorithm for Regression (OSCAR) regularization [Bondell and Reich, 2008] can be seen as a trade-off between an ℓ_1-norm, promoting sparsity, and a pairwise ℓ_∞-norm, encouraging the equality of each pair of entries in $\mathbf{w}$ [El Gueddari, 2019]. This penalty function can be written as:

$$\begin{aligned} p^{\lambda,\gamma}(\mathbf{w}) = \lambda\|\mathbf{w}\|_1 + \gamma \sum_{1 \leq j < i \leq J} \max(|w_j|, |w_i|) &= \lambda\|\mathbf{w}\|_1 + \frac{\gamma}{2} \sum_{1 \leq j < i \leq J} (|w_j - w_i| + |w_j + w_i|) \\ &= \lambda\|\mathbf{w}\|_1 + \frac{\gamma}{2} \left(\|\mathbf{D}_-\mathbf{w}\|_1 + \|\mathbf{D}_+\mathbf{w}\|_1 \right), \end{aligned} \tag{C.17}$$

where $\mathbf{D}_-$ and $\mathbf{D}_+$ are two matrices of size $J(J-1)/2 \times J$ that allow to compute all pairwise difference and sum respectively. Thus $\mathbf{w}^\top \mathbf{D}_-^\top \mathbf{D}_- \mathbf{w} = \sum_{1 \leq j < i \leq J} (w_j - w_i)^2$ and $\mathbf{w}^\top \mathbf{D}_+^\top \mathbf{D}_+ \mathbf{w} = \sum_{1 \leq j < i \leq J} (w_j + w_i)^2$.

Then, using the surrogate function of the LASSO and the TV penalty defined in equation (C.7) and (C.12) respectively, the following surrogate function can be stated for the OSCAR penalty:

$$p^{\lambda,\gamma}(\mathbf{w}) \leq \frac{1}{2}\mathbf{w}^\top \left(\lambda\Delta + \frac{\gamma}{2} \left(\mathbf{D}_-^\top \Delta_- \mathbf{D}_- + \mathbf{D}_+^\top \Delta_+ \mathbf{D}_+ \right) \right) \mathbf{w} + \frac{1}{2} p^{\lambda,\gamma}(\mathbf{w}^0) := \tilde{p}^{\lambda,\gamma}(\mathbf{w}|\mathbf{w}^0) \tag{C.18}$$

where $\Delta, \Delta_-, \Delta_+$ are diagonal matrices of size $J(J-1)/2$ such that $(\Delta)_{ii} = \frac{1}{|w_i^0|}$, $(\Delta_-)_{ii} = \frac{1}{|w_j^0 - w_i^0|}$ and $(\Delta_+)_{ii} = \frac{1}{|w_j^0 + w_i^0|}$. This surrogate function has already been presented in [Sharma et al., 2013].

C.2.6 Pairwise Absolute Clustering and Sparsity (PACS)

In [Sharma et al., 2013], a weighted version of the OSCAR penalty is proposed. This new penalty is called Pairwise Absolute Clustering and Sparsity (PACS) and can be stated as follow:

$$p^{\lambda,\gamma}(\mathbf{w}) = \lambda \sum_{j=1}^{J} \beta_j |w_j| + \frac{\gamma}{2} \sum_{1\leq j<i\leq J} \beta_{ji(-)} |w_j - w_i| + \frac{\gamma}{2} \sum_{1\leq j<i\leq J} \beta_{ji(+)} |w_j + w_i| \tag{C.19}$$

where $\beta_j, \beta_{ji(-)}, \beta_{ji(+)}$ are non-negative weights defined by the user.

Following the previous section, a surrogate function for the PACS penalty can be defined as:

$$p^{\lambda,\gamma}(\mathbf{w}) \leq \frac{1}{2}\mathbf{w}^\top \left(\lambda\Delta + \frac{\gamma}{2}\left(\mathbf{D}_-^\top \Delta_- \mathbf{D}_- + \mathbf{D}_+^\top \Delta_+ \mathbf{D}_+\right)\right)\mathbf{w} + \frac{1}{2}p^{\lambda,c}(\mathbf{w}^0) := \tilde{p}^{\lambda,\gamma}(\mathbf{w}|\mathbf{w}^0), \tag{C.20}$$

where $\Delta, \Delta_-, \Delta_+$ are diagonal matrices of size $J(J-1)/2$ such that $(\Delta)_{ii} = \frac{\beta_i}{|w_i^0|}$, $(\Delta_-)_{ii} = \frac{\beta_{ji(-)}}{|w_j^0 - w_i^0|}$ and $(\Delta_+)_{ii} = \frac{\beta_{ji(+)}}{|w_j^0 + w_i^0|}$. This surrogate function has already been presented in [Sharma et al., 2013].

C.2.7 Truncated ℓ_1-norm or Capped ℓ_1-norm penalty

The truncated ℓ_1-norm penalty (TLP) [Zhang, 2010] is a non-convex approximation of the ℓ_0-pseudo-norm. It can be formulated as follows:

$$p^\tau(\mathbf{w}) = \sum_{i=1}^{J} \min\left(\frac{|w_i|}{\tau}, 1\right) = \sum_{i=1}^{J} \frac{1}{2\tau}\left[|w_i| + \tau - ||w_i| - \tau|\right], \tag{C.21}$$

where $\tau \in \mathbb{R}_+^\star$. Given an appropriate τ, TLP balances between the ℓ_1-norm and the ℓ_0-pseudo-norm. Indeed, for one term of the sum, it is equivalent to the ℓ_1-norm if $|w_i| \leq \tau$, while it becomes the ℓ_0-pseudo-norm as the coefficient goes beyond the threshold τ.

Using (C.6), it is possible to show that:

$$\begin{aligned} |w_i| + \tau &\leq \frac{w_i^2 + {w_i^0}^2}{2|w_i^0|} + \tau \\ ||w_i| - \tau| &\leq \text{sign}\left(|w_i^0| - \tau\right)\left(\frac{w_i^2 + {w_i^0}^2}{2|w_i^0|} - \tau\right) \end{aligned} \tag{C.22}$$

Thus, the following surrogate function for the TLP, already presented in [Du et al., 2017], can be expressed:

$$\begin{aligned} p^\tau(\mathbf{w}) &\leq \frac{1}{2\tau}\sum_{i=1}^{J}\left[\frac{w_i^2 + {w_i^0}^2}{2|w_i^0|}\left(1 - \text{sign}\left(|w_i^0| - \tau\right)\right) + \tau\left(1 + \text{sign}\left(|w_i^0| - \tau\right)\right)\right] \\ &\leq \frac{1}{2\tau}\mathbf{w}^\top \Delta \mathbf{w} + Cste\left(\mathbf{w}^0, \tau\right) := \tilde{p}^\tau(\mathbf{w}|\mathbf{w}^0) \end{aligned} \tag{C.23}$$

where $\Delta \in \mathbb{R}^{J\times J}$ is a diagonal matrix such that $(\Delta)_{ii} = \frac{\left(1 - \text{sign}\left(|w_i^0| - \tau\right)\right)}{2|w_i^0|}$.

C.2.8 Truncated Group-LASSO or Capped Group-LASSO

As it is possible to define an evolution of the LASSO that promote sparsity at the group level, it is also possible to introduce a group-TLP [Du et al., 2017]. By defining a partition $\mathcal{G}$ of $[\![1; J]\!]$, the

group-TLP can be written as:

$$p^{\tau}(\mathbf{w}) = \sum_{g \in \mathcal{G}} \min\left(\frac{\|\mathbf{w}_{i_g}\|_2}{\tau}, 1\right), \tag{C.24}$$

where $\tau \in \mathbb{R}_+^{\star}$ and $\mathbf{w}_{i_g}$ is a subvector of $\mathbf{w}$ containing only the elements of the g^{th} group of $\mathcal{G}$.

Based on the surrogate function of the TLP (see previous section):

$$p^{\tau}(\mathbf{w}) \leq \frac{1}{2\tau}\mathbf{w}^{\top}\Delta\mathbf{w} + Cste\left(\mathbf{w}^0, \tau\right) := \tilde{p}^{\tau}(\mathbf{w}|\mathbf{w}^0),$$

where $\Delta \in \mathbb{R}^{J\times J}$ is a diagonal matrix such that $(\Delta)_{ii} = \frac{\left(1-\text{sign}\left(\|\mathbf{w}_{i_g}\|_2-\tau\right)\right)}{2\|\mathbf{w}_{i_g}\|_2}$.

C.2.9 SparseStep penalty

Another approximation of the ℓ_0-pseudo-norm was proposed in [de Rooi and Eilers, 2011] and is called the SparseStep penalty, which can be formulated as:

$$p^{\gamma}(\mathbf{w}) = \sum_{i=1}^{J} f(w_i) = \sum_{i=1}^{J} \frac{w_i^2}{w_i^2 + \gamma^2}, \tag{C.25}$$

where $\gamma \in \mathbb{R}^{\star}$. Indeed, $\lim_{\gamma \to 0^+} p^{\gamma}(\mathbf{w}) = \text{card}\left(\{i \in [\![1; J]\!]; w_i \neq 0\}\right)$, the ℓ_0-pseudo-norm.

As explained in [van den Burg, 2018], since the function f, defined in equation (C.25) is differentiable, an even function and $f^{'}(w)/w$ is decreasing on $[0, +\infty[$, Theorem 4.5 from [de Leeuw and Lange, 2009] applies and a sharp quadratic majorization function of f is given as:

$$\tilde{f}(w_i|w_i^0) := \frac{f^{'}(w_i^0)}{2w_i^0}(w_i^2 - w_i^{0^2}) + f(w_i^0) = \frac{\gamma^2}{\left(w_i^{0^2} + \gamma^2\right)^2}(w_i^2 - w_i^{0^2}) + \frac{w_i^{0^2}}{w_i^{0^2} + \gamma^2} = \frac{\gamma^2 w_i^2 + w_i^{0^4}}{\left(w_i^{0^2} + \gamma^2\right)^2}, \tag{C.26}$$

which leads to the following surrogate function for the Sparstep penalty:

$$\tilde{p}^{\gamma}(\mathbf{w}|\mathbf{w}^0) := \sum_{i=1}^{J} \frac{\gamma^2 w_i^2 + w_i^{0^4}}{\left(w_i^{0^2} + \gamma^2\right)^2} = \mathbf{w}^{\top}\mathbf{\Delta}\mathbf{w} + \sum_{i=1}^{J} f(w_i^0)^2, \tag{C.27}$$

where $\Delta \in \mathbb{R}^{J\times J}$ is a diagonal matrix such that $(\Delta)_{ii} = \frac{\gamma^2}{\left(w_i^{0^2}+\gamma^2\right)^2}$.

C.2.10 Smoothed ℓ_q-regularization

For $q \in [0, 2]$, $p \geq q$, and $\gamma \geq 0$, the Smoothed ℓ_q-regularization [van den Burg, 2018] can be defined as:

$$p^{\gamma}(\mathbf{w}) = \sum_{i=1}^{J} f(w_i) = \sum_{i=1}^{J} \frac{|w_i|^p}{|w_i|^{p-q} + \gamma^{p-q}} \tag{C.28}$$

The limit case for $\gamma \to 0$ is one of the reasons this penalty is interesting:

$$\lim_{\gamma \to 0} p^{\gamma}(\mathbf{w}) = \sum_{i=1}^{J} |w_i|^q,$$

which is the ℓ_q-norm. The Smoothed ℓ_q-regularization can be seen as a generalization of the SparseStep penalty. Indeed, when $q = 0$ and $p = 2$, equation (C.25) is recovered.

As explained in [van den Burg, 2018], since the function f, defined in equation (C.28) is differentiable, an even function and $f^{'}(w)/w$ is decreasing on $[0, +\infty[$ (see [van den Burg, 2018], equation (5.13) for more details which explains also why $q \in [0, 2]$), Theorem 4.5 from [de Leeuw and Lange, 2009] applies and a sharp quadratic majorization function of f is given as:

$$\tilde{f}(w_i|w_i^0) := \frac{f^{'}(w_i^0)}{2w_i^0}\left(w_i^2 - w_i^{0^2}\right) + f(w_i^0) \tag{C.29}$$

which leads to the following surrogate function for the Smoothed ℓ_q-regularization:

$$\tilde{p}^{\gamma}(\mathbf{w}|\mathbf{w}^0) := \sum_{i=1}^{J} \frac{f^{'}(w_i^0)}{2w_i^0}\left(w_i^2 - w_i^{0^2}\right) + f(w_i^0) = \left(\mathbf{w} - \mathbf{w}^0\right)^{\top} \Delta \left(\mathbf{w} - \mathbf{w}^0\right) + p^{\gamma}(\mathbf{w}^0),$$

where Δ is a diagonal matrix such that $(\Delta)_{ii} = \frac{f^{'}(w_i^0)}{2w_i^0}$.

C.3 Extended results on Structured SGCCA

This section presents extended results for the experiment of Chapter 4 section 4.4.2. This study compares RGCCA, SGCCA and MM_SGCCA on simulated data sets. Here, the comparison with PROX_SGCCA, the structured SGCCA method with the smoothing framework of [Nesterov, 2004] and proximity operators [Löfstedt et al., 2016], is added. For further information about how the data are generated, which structured sparse penalties are used, which constraints are imposed and how parameters are tuned, the reader is referred to sections 4.4.2.1, 4.4.2.2 and 4.4.2.3.

Here, PROX_SGCCA is used in the exact same configuration as MM_SGCCA (same design matrix $\mathbf{C}$, same structured penalties, same constraints, same parameters to tune) except for the function g which is set to the identity function (versus the square function) for PROX_SGCCA as it is the only function handled. Moreover, for PROX_SGCCA, this configuration is evaluated under two different values of the smoothing parameter γ (see the preamble of section 4.4 in Chapter 4 for more details about γ): $\gamma = 5 \times 10^{-3}$ for both penalties or $\gamma = 5 \times 10^{-4}$ for both penalties.

C.3.1 Results

For each value of $\eta \in \{0.5, 1, 2\}$, 40 datasets were generated according to section 4.4.2.1. For each dataset, four methods are compared: RGCCA, SGCCA, MM_SGCCA and PROX_SGCCA. For PROX_SGCCA, 2 values of the smoothing parameter are considered ($\gamma = 5\times10^{-3}$ and $\gamma = 5\times10^{-4}$). For each algorithm and value of η, mean and standard deviation of the ACC (defined in (4.31)) and of $\kappa_l,\ l = 1, 2$ (defined in (4.32)) are computed through datasets and reported in table C.1. The median of the number of iterations of each algorithm, their interquartile range and the mean and standard deviation of the execution time are presented in table C.2. On Figure C.3-1 and C.3-2, the weight vectors for the first and second block respectively are shown for each method.

In table C.1, PROX_SGCCA κ_1 results are as good as MM_SGCCA or even better. However, κ_2 is always equal to zeros, meaning no selection at the group level is performed. This must be explained

by a wrong setting of the parameters (s_1, ψ_1 and ψ_2) or of the smoothing parameters γ. In comparison to MM_SGCCA two additional parameters have to be tuned.

In table C.2, it appears that PROX_SGCCA always reach the maximum number of iterations authorized. This should be dealt with in future works as it makes the result not really comparable to the other methods.

In figures C.3-1 and C.3-2, the rows are associated with a specific value of η and the columns with a specific method. It is interesting to visualize how the estimations evolve with SNR and methods. For example, only MM_SGCCA manages to catch almost all the right null elements for $\eta = 0.5$ for the two blocks. SGCCA only reaches this goal when $\eta = 2$. As mentioned earlier, RGCCA cannot perform this estimation as no sparse constraint is imposed. Furthermore, especially for $\eta = 1$ and the first block weight vector, PROX_SGCCA ($\gamma = 5 \times 10^{-3}$) presents smoother estimates than PROX_SGCCA ($\gamma = 5 \times 10^{-4}$) as its smoothing parameter is higher.

C.3.2 Conclusion

Cautions must be taken with these results as PROX_SGCCA always reached the maximum number of iterations allowed. Work in progress aims at improving the results obtain with PROX_SGCCA and adding the comparison with other structured sparse CCA algorithms as [Chen and Liu, 2012, Chen et al., 2012a].

Table C.1 – For each value of $\eta \in \{0.5, 1, 2\}$, 40 datasets were generated. For each dataset, four methods are compared: RGCCA, SGCCA, MM_SGCCA and PROX_SGCCA. For PROX_SGCCA 2 values of γ (5×10^{-3} or 5×10^{-4}) are considered. For each algorithm, the mean and standard deviation (std) of the ACC (defined in (4.31)) and of κ_l, $l = 1, 2$ (defined in (4.32)) are reported.

SNR	Algorithm	ACC (mean ± std)	κ_1 (mean ± std)	κ_2 (mean ± std)
$\eta = 0.5$	RGCCA	0.952 ± 0.004	\	\
	SGCCA	0.935 ± 0.005	0.77 ± 0.04	0.82 ± 0.05
	MM_SGCCA	0.973 ± 0.006	0.9 ± 0.1	1 ± 0
	PROX_SGCCA $\gamma = 5e-3$	0.969 ± 0.004	0.94 ± 0.03	0 ± 0
	PROX_SGCCA $\gamma = 5e-4$	0.966 ± 0.004	0.91 ± 0.06	0 - 0
$\eta = 1$	RGCCA	0.984 ± 0.002	\	\
	SGCCA	0.981 ± 0.001	0.90 ± 0.03	0.97 ± 0.03
	MM_SGCCA	0.9928 ± 0.0007	0.99 ± 0.01	1 ± 0
	PROX_SGCCA $\gamma = 5e-3$	0.989 ± 0.001	0.99 ± 0.01	0 - 0
	PROX_SGCCA $\gamma = 5e-4$	0.988 ± 0.001	0.98 ± 0.02	0 - 0
$\eta = 2$	RGCCA	0.992 ± 0.002	\	\
	SGCCA	0.9944 ± 0.0002	0.99 ± 0.01	1 ± 0
	MM_SGCCA	0.9977 ± 0.0002	1 ± 0	1 ± 0
	PROX_SGCCA $\gamma = 5e-3$	0.9959 ± 0.0002	0.999 ± 0.004	0 - 0
	PROX_SGCCA $\gamma = 5e-4$	0.9961 ± 0.0002	1 ± 0.002	0 - 0

Table C.2 – For each value of $\eta \in \{0.5, 1, 2\}$, 40 datasets were generated. For each dataset, four methods are compared: RGCCA, SGCCA, MM_SGCCA and PROX_SGCCA. For PROX_SGCCA, 2 values of γ (5×10^{-3} or 5×10^{-4}) are considered. For each algorithm, the median (MD) of the number of iterations (Iter), their interquartile range (IQR) and the mean and standard deviation of the execution time (Time) are reported.

SNR	Algorithm	Iter (MD - IQR)	Time (s) (mean ± std)
$\eta = 0.5$	RGCCA	6 - 0	0.03 ± 0.01
	SGCCA	6 - 0	0.042 ± 0.007
	MM_SGCCA	2180 - 1055	127 ± 35
	PROX_SGCCA $\gamma = 5e-3$	19980 - 0	74 ± 17
	PROX_SGCCA $\gamma = 5e-4$	19980 - 0	63 ± 23
$\eta = 1$	RGCCA	4 - 0	0.03 ± 0.01
	SGCCA	4 - 0	0.033 ± 0.006
	MM_SGCCA	740 - 127	46 ± 6
	PROX_SGCCA $\gamma = 5e-3$	19980 - 0	84 ± 13
	PROX_SGCCA $\gamma = 5e-4$	19980 - 0	80 ± 26
$\eta = 2$	RGCCA	4 - 0	0.04 ± 0.02
	SGCCA	3 - 1	0.029 ± 0.006
	MM_SGCCA	426 - 52	28 ± 3
	PROX_SGCCA $\gamma = 5e-3$	19980 - 0	64 ± 20
	PROX_SGCCA $\gamma = 5e-4$	19980 - 0	93 ± 31

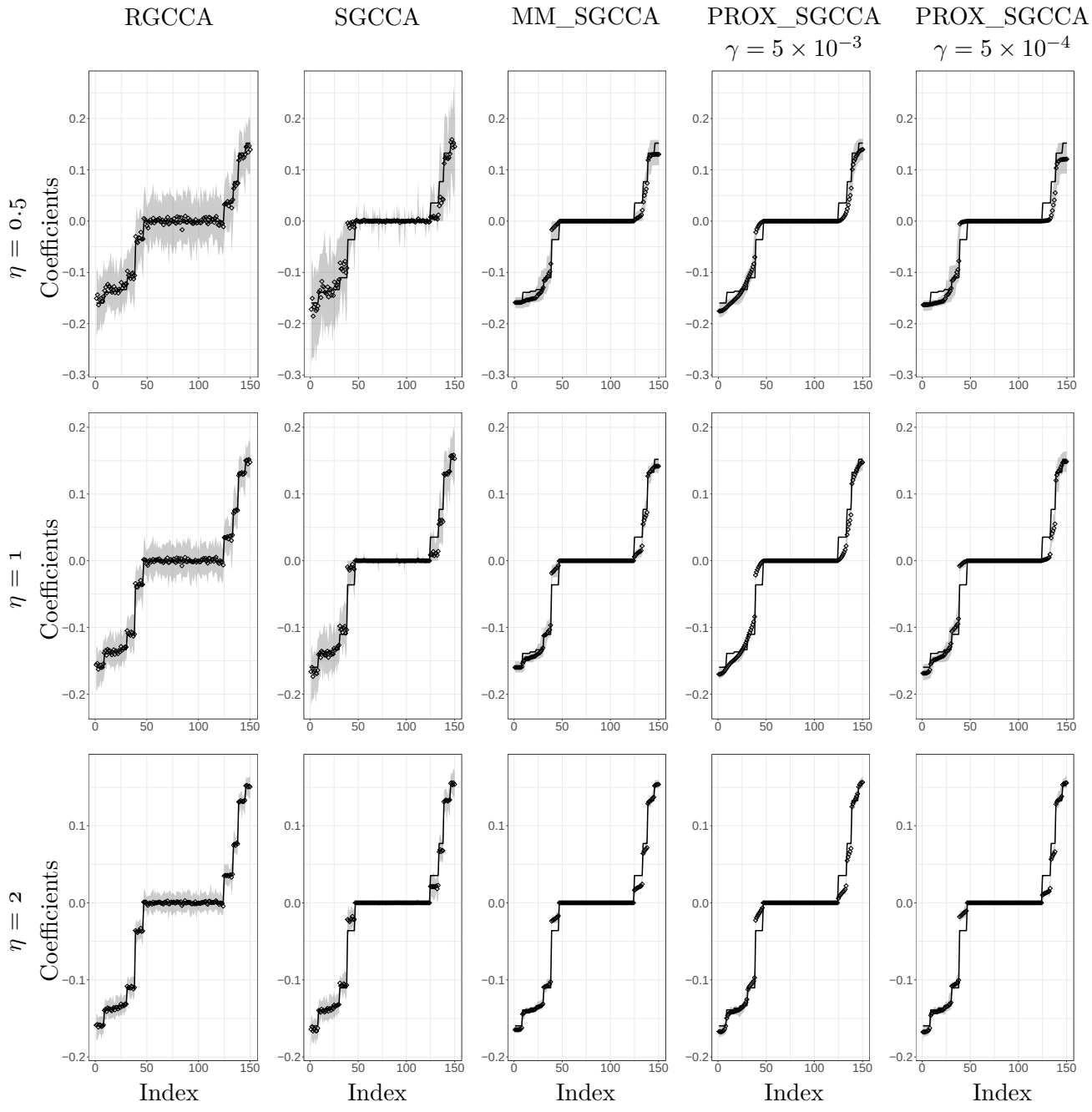


Figure C.3-1 – Continuous lines correspond to the first block weight vector and points to its estimates obtained with RGCCA, SGCCA, MM_SGCCA and PROX_SGCCA. For PROX_SGCCA, 2 values of γ (5×10^{-3} or 5×10^{-4}) are considered. Each row is associated with a specific value of η (0.5, 1 and 2) arranged in increasing order and each column with a method. 4 worst runs for each method according to a weighted sum of ACC, κ_1 and κ_2 were removed. For each element of this estimated vector, grey areas stand for the *min* and *max* of its distribution based on the 36 remaining runs.

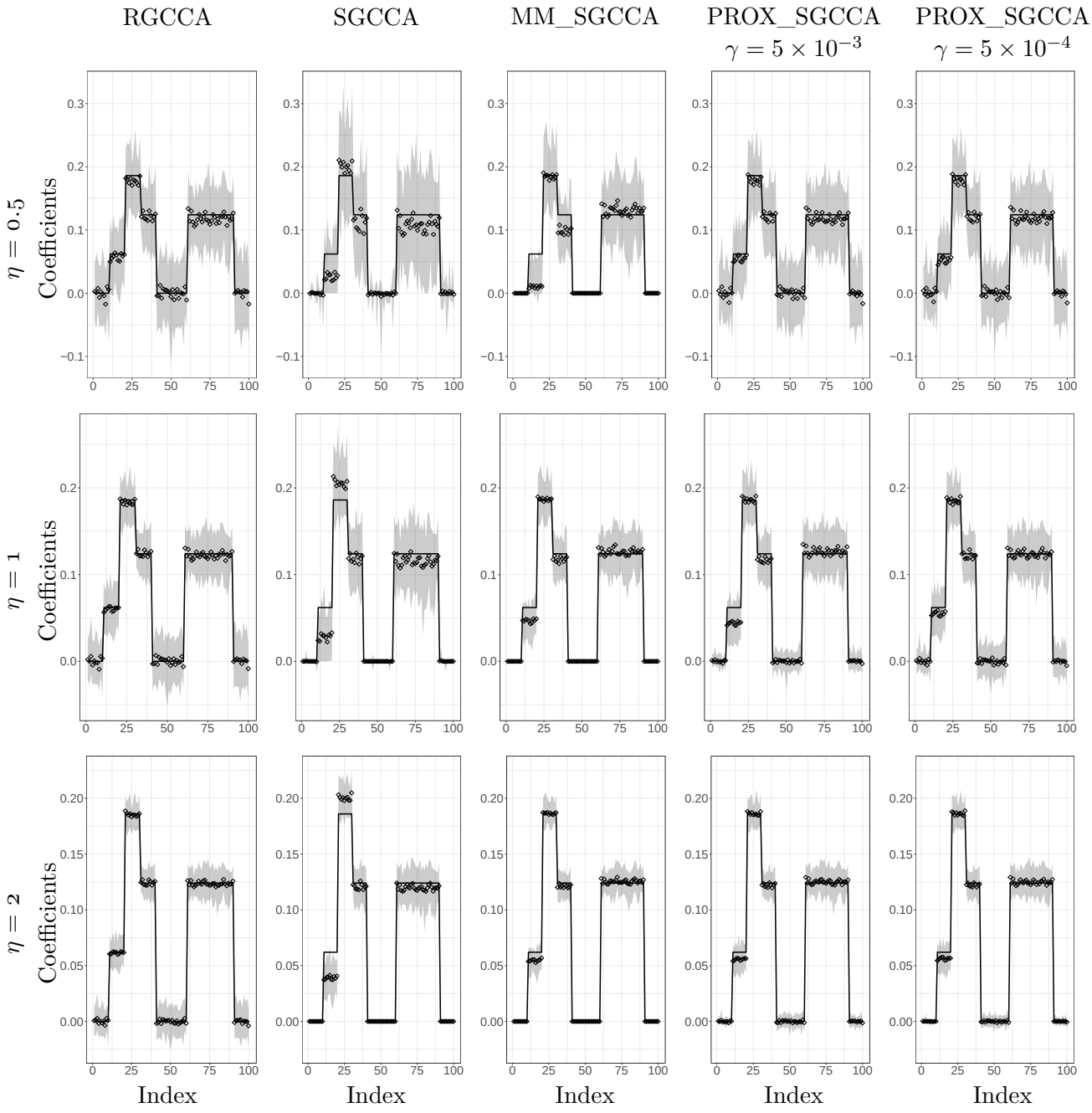


Figure C.3-2 – Continuous lines correspond to the second block weight vector and points to its estimates obtained with RGCCA, SGCCA, MM_SGCCA and PROX_SGCCA. For PROX_SGCCA, 2 values of γ (5×10^{-3} or 5×10^{-4}) are considered. Each row is associated with a specific value of η (0.5, 1 and 2) arranged in increasing order and each column with a method. 4 worst runs for each method according to a weighted sum of ACC, κ_1 and κ_2 were removed. For each element of this estimated vector, grey areas stand for the min and max of its distribution based on the 36 remaining runs.

Appendix D

Résumé en français

Abstract in French

Sujet : Un cadre statistique et algorithmique pour l'analyse de données multibloc et multivoie.

Nous résumons ici les différents aspects abordés au travers de cette thèse. Après avoir décrit les enjeux et motivations qui nous ont poussé au développement des méthodes abordées dans ce travail, nous résumerons chacune des contributions.

Motivations et contextes

L'étude des relations entre plusieurs ensembles de variables mesurées sur un même groupe d'individus est un défi majeur en statistique. La littérature fait référence à ce paradigme sous plusieurs termes : «analyse de données multimodales», «intégration de données», «fusion de données» ou «analyse de données multibloc». Ce type de problématique se retrouve dans des domaines aussi variés que la biologie, la chimie, l'analyse multi-capteurs, le marketing, la recherche agro-alimentaire, où l'objectif commun est d'identifier les variables de chaque bloc intervenant dans les intéractions entre blocs. Par exemple, afin d'expliquer le lien complexe entre un phénotype comportemental ou clinique et la génétique, la neuroimagerie est souvent utilisée comme phénotype intermédiaire (ou endophénotype). Ainsi, en imagerie-génétique, le but est d'identifier un ensemble de biomarqueurs génétiques expliquant une certaine variabilité de la neuroimagerie, elle-même engendrant des variations du comportement. Par ailleurs, chaque bloc est composé d'un très grand nombre de variables ($\sim$ 1M), nécessitant le calcul de milliards d'associations. L'élaboration d'un cadre statistique adapté à la structure particulière des données ainsi qu'à leur hétérogénéité est donc primordial pour étudier ce type de données.

En plus de cette structure globale multi-source, chaque source peut être représentée sous la forme d'un tenseur ou d'une matrice d'ordre supérieur. Par exemple, l'Imagerie par Résonance Magnétique (IRM) permet d'obtenir une image tridimensionnelle du cerveau, donc par nature, un tenseur. De même, l'ÉlectroEncéphaloGraphie (EEG) ou la MagnétoEncéphaloGraphie (MEG) donnent accès respectivement aux ondes cérébrales électriques ou magnétiques. Ces ondes sont mesurées par plusieurs

capteurs simultanément, ce qui permet d'obtenir des données spatio-temporelles. Lorsque ces données spatio-temporelles bidimensionnelles sont mesurées sur différents individus, un tenseur peut alors être construit. Le respect de cette structure tensorielle éventuelle est indispensable pour analyser les données sans risque de perte d'information.

Le principe de parcimonie est au cœur de nombreux domaines scientifiques ; en effet, une explication plus simple d'un phénomène donné doit être préférée aux explications plus complexes. En statistique, cette parcimonie peut se traduire par de la sélection de variables. Dans le cadre de l'étude d'une maladie neurodégénérative impliquant à la fois des mesures en génétique et en imagerie, cette sélection de variables permet d'identifier un sous-ensemble de variants génétiques impliqués dans la neurodégénérescence de certaines régions du cerveau.

Des algorithmes de modélisation spécialisés, capables de prendre en compte les propriétés structurelles inhérentes à ces ensembles de données multi-sources, sont donc indispensables pour exploiter pleinement leur complexité et fournir des informations pertinentes et robustes.

Déroulement du manuscrit de thèse

Le développement de méthodes d'analyse de données hétérogènes, potentiellement de grande dimension, est au coeur de ce travail. Ces développements se basent sur l'Analyse Canonique Généralisée Régularisée (RGCCA), un cadre général pour l'analyse de données multiblocs. Le coeur algorithmique de RGCCA se résume à une unique «update», répétée jusqu'à convergence. Si cette update possède certaines «bonnes» propriétés, la convergence globale de l'algorithme est garantie. Tout au long de ce manuscrit, nous avons essayé de préserver à la fois la flexibilité et la simplicité du cadre d'optimisation algorithmique proposé par RGCCA.

Dans une seconde partie, l'analyse de plusieurs jeux de données est menée à l'aide de ces nouvelles méthodes. La polyvalence des ces outils est démontrée sur (i) deux études en imagerie-génétique, (ii) deux études en électroencéphalographie ainsi (iii) qu'une étude en microscopie Raman. L'accent est mis sur l'interprétation des résultats facilitée par la prise en compte des structures multiblocs, tensorielles et/ou parcimonieuses.

Ce manuscrit de thèse est organisé comme suit :

Chapitre 1 : *Contexte des méthodes multiblocs et multivoies*

Ce chapitre commence par une description de l'Analyse Canonique Généralisée Régularisée (RGCCA), suivie d'un aperçu d'autres méthodes multiblocs - cas particulier ou non de RGCCA. Dans une deuxième partie, les notations et opérateurs classiques de l'analyse tensorielles sont rappelées. Les modèles multivoie les plus courants sont également présentés. Ce chapitre se conclut par la présentation d'un cadre d'optimisation algorithmiques simple et très général. Ce cadre va nous servir de base pour tous les développements algorithmiques abordés dans le cadre de ce travail. Comme nous le verrons, il offre une approche systématique pour construire des algorithmes globalement convergents.

Chapitre 2 : *Une version gloable de l'Analyse Canonique Généralisée et Régularisée*

L'objectif de RGCCA est de construire un ensemble de composantes pour chaque bloc permettant de décrire les blocs et les relations entre blocs. RGCCA appartient à la famille des méthodes multiblocs séquentielles. Cela signifie que les composantes de chaque bloc sont déterminées séquentiellement (R

problèmes d'optimisation successifs doivent être résolus afin d'extraire R composants de chaque bloc). D'un point de vue algorithmique, cette stratégie semble être sous-optimale. Aussi, nous présentons dans le chapitre 2 une version globale de RGCCA permettant d'extraire simultanément les composantes de chaque bloc à l'aide d'un unique problème d'optimisation. Ce problème d'optimisation globale de RGCCA est présenté et nous montrons que l'algorithme correspondant est globalement convergent. Les approches séquentielle et globale de RGCCA sont enfin comparées sur données simulées.

Chapitre 3 : *L'Analyse Canonique Généralisée et Multivoie (MGCCA)*

L'Analyse Canonique Généralisée Multivoie (MGCCA) étend RGCCA à l'analyse conjointe d'un ensemble de tenseurs ou de matrices d'ordre supérieur. Des versions séquentielle et globale de MGCCA sont proposées. Pour l'approche séquentielle, deux stratégies différentes permettent d'obtenir des composantes de niveau supérieur. Les deux algorithmes proposés pour MGCCA (global ou séquentiel) sont globalement convergents. Ces deux approches sont comparées sur données simulées.

Chapitre 4 : *Parcimonie Structurée dans l'Analyse Canonique Généralisée Parcimonieuse (SGCCA)*

Un des défis majeurs de l'analyse de données multi-source est d'identifier les variables de chaque source forcent de liaison entre blocs, notamment lorsque les données sont de grande dimension. L'Analyse Canonique Généralisée Parcimonieuse (SGCCA) est une version de RGCCA permettant de sélectionner les variables qui interagissent le plus entre les blocs. Un nouvel algorithme est présenté pour résoudre plus rapidement le problème d'optimisation de SGCCA. Nous démontrons que ce nouvel algorithme est globalement convergent. La sélection de variables dans SGCCA repose sur la norme ℓ_1 qui ne prend pas en compte les intéractions possibles entre les variables à l'intérieur d'un bloc. SGCCA a donc été améliorée en introduisant de la parcimonie structurée (LASSO, groupe LASSO, élitiste LASSO) dans son critère d'optimisation.

Chapitre 5 : *Analyse de données multiblocs multivoies*

Dans ce dernier chapitre, la polyvalence de RGCCA et/ou MGCCA est évaluée sur cinq jeu de données de nature multibloc et/ou multivoie. Dans une première étude, on cherche à questionner l'influence de la génétique sur le vieillissement normal du cerveau au sein de la cohorte United Kingdom Biobank (UKB). La seconde est une étude d'imagerie-génétique sur la base de données «Alzheimer's disease Neuroimaging Initiative»(ADNI) qui vise à comprendre certains mécanismes de la maladie grâce à une approche multimodale (génétique, transcriptomique, IRM longitudinale, facteurs cliniques). La troisième étude vise à analyser, à partir de la microscopie Raman, l'efficacité d'une crème hydratante. Les deux dernières études cherchent à identifier chez le nourrisson humain, à partir de l'ÉlectroEncéphaloGraphie (EEG), des zones du cerveau impliquées dans le processus de discrimination entre des syllabes proches.

* * *
* *
*

Publications

Articles in Peer-Reviewed Journals

Vincent Guillemot, Derek Beaton, **Arnaud Gloaguen**, Tommy Löfstedt, Brian Levine, Nicolas Raymond, Arthur Tenenhaus and Hervé Abdi. *A constrained singular value decomposition method that integrates sparsity and orthogonality.* PLoS ONE, Public Library of Science, 2019, 14 (3), pp.e0211463.

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Giulia Gennari, Ghislaine Dehaene-Lambertz, Laurent Le Brusquet and Arthur Tenenhaus. *Multiway Generalized Canonical Correlation Analysis.* Biostatistics, 2020, Accepted.

Angeline Mihailov, Cathy Philippe, **Arnaud Gloaguen**, Antoine Grigis, Charles Laidi, Camille Piguet, Josselin Houenou and Vincent Frouin. *Cortical Signatures in Behaviorally Clustered Autistic Traits Subgroups.* Translational Psychiatry, 2020. Accepted.

International Conferences Paper Presented with Reading Committee and Proceedings

Slim Karkar, **Arnaud Gloaguen**, Yann Le Guen, Morgane Pierre-Jean, Claire Dandine-Roulland, Edith Le Floch, Cathy Philippe, Arthur Tenenhaus and Vincent Frouin. *Multivariate Haplotype Analysis Of 96 Sulci Opening For 15,612 UK-Biobank Sujects.* Proceedings of the IEEE International Symposium of Biomedical Imaging, Venice, Italy, 2019. Poster.

Nicolas Guigui, Cathy Philippe, **Arnaud Gloaguen**, Slim Karkar, Vincent Guillemot, Tommy Löfstedt and Vincent Frouin. *Network Regularization in Imaging Genetics Improves Prediction Performances and Model Interpretability on Alzheimer's Disease.* Proceedings of the IEEE International Symposium of Biomedical Imaging, Venice, Italy, 2019. Oral.

Abstracts Presented at National Conferences with Reading Committee

Arnaud Gloaguen, Vincent Guillemot, and Arthur Tenenhaus. *An efficient algorithm to satisfy l_1 and l_2 constraints.*49ème Journées de Statistique (JDS), Avignon, France, 2017. Oral.

Arnaud Gloaguen, Cathy Philippe, Vincent Frouin, Laurent Le Brusquet and Arthur Tenenhaus. *Multiway Generalized Canonical Correlation Analysis.*Chimiométrie XIX, Paris, France, 2018. Oral.

Etienne Camenen, **Arnaud Gloaguen**, François-Xavier Lejeune, Ivan Moszer and Arthur Tenenhaus. *A Shiny and Galaxy interactive software for multi-source data analysis.* Journées Ouvertes Biologie, Informatique et Mathématiques (JOBIM), Nantes, France, 2019. Oral.

Vincent Guillemot, Julie Le Borgne, **Arnaud Gloaguen**, Arthur Tenenhaus, Gilbert Saporta, Sylvie Chollet, Derek Beaton and Hervé Abdi. *Sparse Multiple Correspondence Analysis.* 52ème Journées de Statistique (JDS), France, 2020. Submitted.

Fabien Girka, Pierrick Chevalier, **Arnaud Gloaguen**, Laurent Le Brusquet and Arthur Tenenhaus. *Rank-R Multiway Logistic Regression.* 52ème Journées de Statistique (JDS), France, 2020. Accepted.

Bibliography

Evrim Acar and Bulent Yener. Unsupervised Multiway Data Analysis: A Literature Survey. *IEEE Transactions on Knowledge and Data Engineering*, 21(1):6–20, 2009. (Cited on page 100.)

Evrim Acar, Tamara G. Kolda, and Daniel M. Dunlavy. All-at-once optimization for coupled matrix and tensor factorizations. *In Proc. KDD Workshop Mining and Learning with Graphs (MLG), San Diego, California*, available online: http://arxiv.org/abs/1105.3422, 2011. (Cited on pages 19, 33, 36, and 51.)

Evrim Acar, Morten Arendt Rasmussen, Francesco Savorani, Tormod Næs, and Rasmus Bro. Understanding data fusion within the framework of coupled matrix and tensor factorizations. *Chemometrics and Intelligent Laboratory Systems*, 129:53–63, 2013. (Cited on pages 33, 36, 44, and 51.)

Evrim Acar, Evangelos E Papalexakis, Gözde Gürdeniz, Morten A Rasmussen, Anders J Lawaetz, Mathias Nilsson, and Rasmus Bro. Structure-revealing data fusion. *BMC Bioinformatics*, 15(1), July 2014. (Cited on pages 19 and 36.)

Kohei Adachi. *Matrix-Based Introduction to Multivariate Data Analysis*. Springer Singapore, 2016. (Cited on page 31.)

Fidel Alfaro-Almagro, Mark Jenkinson, Neal K. Bangerter, Jesper L.R. Andersson, Ludovica Griffanti, Gwenaëlle Douaud, Stamatios N. Sotiropoulos, Saad Jbabdi, Moises Hernandez-Fernandez, Emmanuel Vallee, Diego Vidaurre, Matthew Webster, Paul McCarthy, Christopher Rorden, Alessandro Daducci, Daniel C. Alexander, Hui Zhang, Iulius Dragonu, Paul M. Matthews, Karla L. Miller, and Stephen M. Smith. Image processing and quality control for the first 10, 000 brain imaging datasets from UK biobank. *NeuroImage*, 166:400–424, February 2018. (Cited on page 83.)

Brett W. Bader and Tamara G. Kolda. Algorithm 862. *ACM Transactions on Mathematical Software*, 32(4):635–653, December 2006. (Cited on page 15.)

Anahita Basirat, Stanislas Dehaene, and Ghislaine Dehaene-Lambertz. A hierarchy of cortical responses to sequence violations in three-month-old infants. *Cognition*, 132(2):137–150, August 2014. (Cited on pages 102 and 105.)

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57 (1):289–300, 1995. ISSN 00359246. (Cited on page 106.)

Claude Berge. *Espaces topologiques*. Dunod Paris, 1966. (Cited on page 22.)

Dimitri P. Bertsekas, Angelia Nedić, and Asuman E. Ozdaglar. *Convex Analysis and Optimization*. 07 2020. ISBN 9781886529458. (Cited on page 66.)

Howard D. Bondell and Brian J. Reich. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with OSCAR. *Biometrics*, 64(1):115–123, Mar 2008. (Cited on page 126.)

Leonie Borne, Jean-François Mangin, and Denis Riviere. A patch-based segmentation approach with high level representation of the data for cortical sulci recognition. In W. Bai, G. Sanroma, G. Wu, B. Munsell, Y. Zhan, and P. Coupe, editors, *Patch-Based Techniques in Medical Imaging*, volume 11075 of *Lecture Notes in Computer Science*. Springer, Cham, 2018. ISBN 978-3-030-00499-6. (Cited on page 85.)

Stephen Boyd and Jon Dattorro. Alternating Projections. Ee392o, Stanford University, 2003. (Cited on page 61.)

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004. (Cited on page 58.)

Rasmus Bro. Multiway calidration. Multilinear PLS. *Journal of Chemometrics*, 10:47–61, 1996. (Cited on pages 37, 38, 39, 42, and 115.)

Rasmus Bro and Henk A.L. Kiers. A new efficient method for determining the number of components in parafac models. *Journal of Chemometrics*, 17(5):274–286, 2003. (Cited on page 111.)

Clare Bycroft, Colin Freeman, Desislava Petkova, Gavin Band, Lloyd T. Elliott, Kevin Sharp, Allan Motyer, Damjan Vukcevic, Olivier Delaneau, Jared O'Connell, Adrian Cortes, Samantha Welsh, Gil McVean, Stephen Leslie, Peter Donnelly, and Jonathan Marchini. Genome-wide genetic data on ~500, 000 UK biobank participants. July 2017. (Cited on page 83.)

Emmanuel J. Candes and Justin K. Romberg. Signal recovery from random projections, 2005. (Cited on page 62.)

J. Douglas Carroll. A generalization of canonical correlation analysis to three or more sets of variables. In *Proceeding 76th Conv. Am. Psych. Assoc.*, pages 227–228, 1968. (Cited on page 9.)

J. Douglas Carroll and Jih-Jie Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. *Psychometrika*, 35(3):283–319, 1970. (Cited on pages 16, 36, 38, and 115.)

Xi Chen and Han Liu. An efficient optimization algorithm for structured sparse CCA, with applications to eQTL mapping. *Statistics in Biosciences*, 4(1):3–26, December 2012. (Cited on pages 65, 76, and 130.)

Xi Chen, Liu Han, and Jaime Carbonell. Structured sparse canonical correlation analysis. In Neil D. Lawrence and Mark Girolami, editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 199–207, La Palma, Canary Islands, 21–23 Apr 2012a. PMLR. (Cited on pages 36, 65, 76, and 130.)

Xi Chen, Qihang Lin, Seyoung Kim, Jaime G. Carbonell, and Eric P. Xing. Smoothing proximal gradient method for general structured sparse regression. *The Annals of Applied Statistics*, 6(2): 719–752, 2012b. ISSN 19326157. (Cited on page 65.)

Daniel Chessel and Mohamed Hanafi. Analyse de la co-inertie de K nuages de points. *Revue de Statistique Appliquée*, 44:35–60, 1996. (Cited on page 9.)

Eric C. Chi, Hua Zhou, and Kenneth Lange. Distance majorization and its applications. *Mathematical Programming*, 146(1-2):409–436, jun 2013. (Cited on pages 66, 67, 68, 69, 76, 110, and 123.)

Ji Yeh Choi, Seungmi Yang, Arthur Tenenhaus, and Heungsun Hwang. Three-way generalized structured component analysis. In Marie Wiberg, Steven Culpepper, Rianne Janssen, Jorge González, and Dylan Molenaar, editors, *Quantitative Psychology*, pages 195–209, Cham, 2018. Springer International Publishing. ISBN 978-3-319-77249-3. (Cited on page 36.)

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, April 1960. (Cited on page 72.)

Ingrid Daubechies, Massimo Fornasier, and Ignace Loris. Accelerated projected gradient method for linear inverse problems with sparsity constraints. *Journal of Fourier Analysis and Applications*, 14 (5-6):764–792, 2008. (Cited on page 62.)

Jan De Leeuw. Block-relaxation algorithms in statistics. In Hans-Hermann Bock, Wolfgang Lenski, and Michael M. Richter, editors, *Information Systems and Data Analysis*, pages 308–324, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg. ISBN 978-3-642-46808-7. (Cited on pages 20, 48, 66, and 67.)

Jan De Leeuw. Mm algorithms for smoothed absolute values. 2018. (Cited on page 76.)

Jan de Leeuw and Kenneth Lange. Sharp quadratic majorization in one dimension. *Computational Statistics & Data Analysis*, 53(7):2471 – 2484, 2009. ISSN 0167-9473. (Cited on pages 76, 123, 124, 128, and 129.)

Johan de Rooi and Paul Eilers. Deconvolution of pulse trains with the l0 penalty. *Analytica Chimica Acta*, 705(1):218 – 226, 2011. ISSN 0003-2670. A selection of papers presented at the 12th International Conference on Chemometrics in Analytical Chemistry. (Cited on page 128.)

Ghislaine Dehaene-Lambertz and Stanislas Dehaene. Speed and cerebral correlates of syllable discrimination in infants. *Nature*, 370(6487):292–295, jul 1994. (Cited on pages 101, 102, 104, and 105.)

Katrijn Van Deun, Age K. Smilde, Mariët J. van der Werf, Henk A.L. Kiers, and Iven Van Mechelen. A structured overview of simultaneous component based data integration. *BMC Bioinformatics*, 10 (1), August 2009. (Cited on pages 10 and 36.)

Lei Du, Jingwen Yan, Sungeun Kim, Shannon L. Risacher, Heng Huang, Mark Inlow, Jason H. Moore, Andrew J. Saykin, Li Shen, and [Authorinst]for the Alzheimer's Dis Initiative. GN-SCCA: GraphNet based sparse canonical correlation analysis for brain imaging genetics. In *Brain Informatics and Health*, pages 275–284. Springer International Publishing, 2015. (Cited on page 66.)

Lei Du, , Heng Huang, Jingwen Yan, Sungeun Kim, Shannon Risacher, Mark Inlow, Jason Moore, Andrew Saykin, and Li Shen. Structured sparse CCA for brain imaging genetics via graph OSCAR. *BMC Systems Biology*, 10(S3), August 2016a. (Cited on page 66.)

Lei Du, Heng Huang, Jingwen Yan, Sungeun Kim, Shannon L. Risacher, Mark Inlow, Jason H. Moore, Andrew J. Saykin, and Li Shen and. Structured sparse canonical correlation analysis for brain imaging genetics: an improved GraphNet method. *Bioinformatics*, 32(10):1544–1551, January 2016b. (Cited on page 66.)

Lei Du, Kefei Liu, Tuo Zhang, Xiaohui Yao, Jingwen Yan, Shannon L Risacher, Junwei Han, Lei Guo, Andrew J Saykin, and Li Shen and. A novel SCCA approach via truncated ℓ_1-norm and truncated group lasso for brain imaging genetics. *Bioinformatics*, 34(2):278–285, September 2017. (Cited on pages 66, 76, and 127.)

John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the L1 -ball for learning in high dimensions. *Proceedings of the 25th international conference on Machine learning - ICML*, pages 272–279, 2008. (Cited on page 62.)

Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, sep 1936. (Cited on page 17.)

Bradley Efron. Bootstrap methods: another look at the jackknife. *Annals of Statistics*, 7:1–26, 1979. (Cited on pages 102 and 105.)

Bradley Efron. Better Bootstrap Confidence Intervals. *Journal of the American Statistical Association*, 82:171–185, 1987. (Cited on pages 102 and 105.)

Loubna El Gueddari. *Proximal structured sparsity regularization for online reconstruction in high-resolution accelerated Magnetic Resonance imaging*. PhD thesis, 2019. Thèse de doctorat dirigée par Boulant, Nicolas et Ciuciu, Philippe Imagerie et physique médicale Université Paris-Saclay (ComUE) 2019. (Cited on page 126.)

Jeffrey Fessler. Monotone convergence. Lecture notes, 2004. https://web.eecs.umich.edu/ fessler/course/600/l/lmono.pdf. (Cited on page 23.)

Mário A. T. Figueiredo, Jose B. Dias, João P. Oliveira, and Robert D. Nowak. On total variation denoising: A new majorization-minimization algorithm and an experimental comparisonwith wavalet denoising. In *2006 International Conference on Image Processing*. IEEE, October 2006. (Cited on page 72.)

Clara Fischer, Grégory Operto, S. Laguitton, Matthieu Perrot, Isabelle Denghien, Denis Rivière, and Jean-François Mangin. Morphologist 2012: the new morphological pipeline of brainvisa. In *Proc. HBM*, 2012. (Cited on page 85.)

Bruce Fischl, André Van Der Kouwe, Christophe Destrieux, Eric Halgren, Florent Ségonne, David H Salat, Evelina Busa, Larry J Seidman, Jill Goldstein, David Kennedy, et al. Automatically parcellating the human cerebral cortex. *Cerebral cortex*, 14(1):11–22, 2004. (Cited on page 89.)

Anders M. Fjell and Kristine B. Walhovd. Structural brain changes in aging: Courses, causes and cognitive consequences. *Reviews in the Neurosciences*, 21(3), jan 2010. (Cited on page 84.)

Jean-Philippe Fortin, Nicholas Cullen, Yvette I. Sheline, Warren D. Taylor, Irem Aselcioglu, Philip A. Cook, Phil Adams, Crystal Cooper, Maurizio Fava, Patrick J. McGrath, Melvin McInnis, Mary L. Phillips, Madhukar H. Trivedi, Myrna M. Weissman, and Russell T. Shinohara. Harmonization of cortical thickness measurements across scanners and sites. *NeuroImage*, 167:104–120, February 2018. (Cited on page 90.)

Yifan Fu, Junbin Gao, Xia Hong, and David Tien. Tensor regression based on linked multiway parameter analysis. In *2014 IEEE International Conference on Data Mining*, pages 821–826, 2014. (Cited on page 36.)

Imene Garali, Isaac M Adanyeguh, Farid Ichou, Vincent Perlbarg, Alexandre Seyer, Benoit Colsch, Ivan Moszer, Vincent Guillemot, Alexandra Durr, Fanny Mochel, and Arthur Tenenhaus. A strategy for multimodal data integration: application to biomarkers identification in spinocerebellar ataxia. *Briefings in Bioinformatics*, 19(6):1356–1369, July 2017. (Cited on pages 7 and 106.)

Yulin Ge, Robert I. Grossman, James S. Babb, Marcie L. Rabin, Lois J. Mannon, and Dennis L. Kolson. Age-related total gray matter and white matter changes in normal adult brain. part ii: Quantitative magnetization transfer ratio histogram analysis. *American Journal of Neuroradiology*, 23(8):1334–1341, 2002. ISSN 0195-6108. (Cited on page 84.)

Giulia Gennari and Ghislaine Dehaene-Lambertz. EEG Multivariate Pattern Analysis reveals higher-order linguistic extraction of phonetic features at 3 months of age. In *NeuroFrance 2019, Poster*, Marseille, France, May 2019. (Cited on pages x and 104.)

Fabien Girka, Pierrick Chevalier, Arnaud Gloaguen, Laurent Le Brusquet, and Arthur Tenenhaus. Rank-R Multiway Logistic Regression. In *52èmes Journées de Statistique*, volume Submitted, France, 2020. (Cited on page 106.)

John R. Gleason. Algorithms for balanced bootstrap simulations. *The American Statistician*, 42(4): 263–266, November 1988. (Cited on page 86.)

Arnaud Gloaguen, Vincent Guillemot, and Arthur Tenenhaus. An efficient algorithm to satisfy l1 and l2 constraints. In *49èmes Journées de Statistique*, Avignon, France, May 2017. (Cited on page 62.)

Logan Grosenick, Brad Klingenberg, Stephanie Greer, Jonathan Taylor, and Brian Knutson. Whole-brain sparse penalized discriminant analysis for predicting choice. *NeuroImage*, 47:S58, July 2009. (Cited on page 65.)

Nicolas Guigui, Cathy Philippe, Arnaud Gloaguen, Slim Karkar, Vincent Guillemot, Tommy Löfstedt, and Vincent Frouin. Network regularization in imaging genetics improves prediction performances

and model interpretability on alzheimer's disease. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 1403–1406, 2019. (Cited on pages 77, 91, 92, and 95.)

Fouad Hadj-Selem, Tommy Lofstedt, Elvis Dohmatob, Vincent Frouin, Mathieu Dubois, Vincent Guillemot, and Edouard Duchesnay. Continuation of nesterov's smoothing for regression with structured sparsity in high-dimensional neuroimaging. *IEEE Transactions on Medical Imaging*, 37(11): 2403–2413, November 2018. (Cited on page 65.)

M. Hanafi and Henk A.L. Kiers. Analysis of K sets of data, with differential emphasis on agreement between and within sets. *Computational Statistics and Data Analysis*, 51:1491–1508, 2006. (Cited on page 9.)

Mohamed Hanafi. PLS Path modelling: computation of latent variables with the estimation mode B. *Computational Statistics*, 22:275–292, 2007. (Cited on page 9.)

Mohamed Hanafi, Samia Samar Ouertani, Julien Boccard, Gérard Mazerolles, and Serge Rudaz. Multi-way pls regression: Monotony convergence of tri-linear pls2 and optimality of parameters. *Computational Statistics & Data Analysis*, 83:129–139, 2015. (Cited on pages 42 and 116.)

Richard A Harshman. *Foundations of the parafac procedure: models and conditions for an "explanatory" multimodal factor analysis*. University of California at Los Angeles, 1970. (Cited on pages 16, 36, 38, and 115.)

Nathaniel E. Helwig. multiway: Component Models for Multi-Way Data. *https://CRAN.R-project.org/package=multiway*, R package version 1.0-5, 2018. (Cited on page 44.)

Frank L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1-4):164–189, April 1927. (Cited on page 16.)

Frank L Hitchcock. Multiple invariants and generalized rank of a p-way matrix or tensor. *Journal of Mathematics and Physics*, 7(1-4):39–79, April 1928. (Cited on page 16.)

Paul Horst. Relations among m sets of variables. *Psychometrika*, 26:126–149, 1961. (Cited on page 9.)

H. Hotelling. Relation Between Two Sets of Variates. *Biometrika*, 28:321–377, 1936. (Cited on pages 6, 8, 9, and 36.)

David R. Hunter and Kenneth Lange. A tutorial on mm algorithms. *The American Statistician*, 58 (1):30–37, 2004. (Cited on page 67.)

Heungsun Hwang and Yoshio Takane. Generalized structured component analysis. *Psychometrika*, 69 (1):81–99, March 2004. (Cited on pages 36, 110, and 111.)

W. Evan Johnson, Cheng Li, and Ariel Rabinovic. Adjusting batch effects in microarray expression data using empirical Bayes methods. *Biostatistics*, 8(1):118–127, Jan 2007. (Cited on page 90.)

Michel Journée, Yurii Nesterov, Peter Richtárik, and Rodolphe Sepulchre. Generalized power method for sparse principal component analysis. *J. Mach. Learn. Res.*, 11:517–553, March 2010. ISSN 1532-4435. (Cited on page 27.)

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923, 1976. (Cited on page 105.)

Charilaos I. Kanatsoulis, Xiao Fu, Nicholas D. Sidiropoulos, and Mingyi Hong. Structured SUMCOR multiview canonical correlation analysis for large-scale data. *IEEE Transactions on Signal Processing*, 67(2):306–319, January 2019. (Cited on pages 55, 108, and 110.)

Jon Roberts Kettenring. Canonical analysis of several sets of variables. *Biometrika*, 58:433–451, 1971. (Cited on pages 9 and 36.)

Henk A.L. Kiers. *SCA: A Program for Simultaneous Components Analysis of Variables Measured in Two Or More Populations: user's Manual*. iec ProGamma, 1990. (Cited on page 8.)

Henk A.L. Kiers. Towards a standardized notation and terminology in multiway analysis. *Journal of chemometrics*, 14(3):105–122, 2000. (Cited on pages 11, 13, and 37.)

Henk A.L. Kiers and Jos M. F. Berge. Hierarchical relations between methods for simultaneous component analysis and a technique for rotation to a simple simultaneous structure. *British Journal of Mathematical and Statistical Psychology*, 47(1):109–126, May 1994. (Cited on page 36.)

Henk A.L. Kiers and Jos MF ten Berge. Alternating least squares algorithms for simultaneous components analysis with equal component weight matrices in two or more populations. *Psychometrika*, 54(3):467–473, 1989. (Cited on page 8.)

Tae-Kyun Kim and Roberto Cipolla. Canonical correlation analysis of video volume tensors for action categorization and detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31 (8):1415–1428, 2009. (Cited on pages 37 and 65.)

Peter Kochunov, Jean-François Mangin, Thomas Coyle, Jack Lancaster, Paul Thompson, Dennis Rivière, Yann Cointepas, Jean Régis, Anita Schlosser, Don R. Royall, Karl Zilles, John Mazziotta, Arthur Toga, and Peter T. Fox. Age-related morphology trends of cortical sulci. *Human Brain Mapping*, 26(3):210–220, 2005. (Cited on page 84.)

Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3): 455–500, 2009. (Cited on pages 12, 13, 14, 16, 37, and 117.)

Tamara Gibson Kolda. Multilinear operators for higher-order decompositions. Technical report, April 2006. (Cited on pages 12, 14, 18, and 117.)

Nicole Kramer. Analysis of high-dimensional data with partial least squares and boosting. In *Doctoral dissertation, Technischen Universitat Berlin*, 2007. (Cited on page 9.)

Joseph B. Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics. *Linear Algebra and its Applications*, 18(2):95–138, 1977. (Cited on page 17.)

Kenneth Lange. *MM Optimization Algorithms*. SIAM-Society for Industrial and Applied Mathematics, USA, 2016. ISBN 1611974399, 9781611974393. (Cited on pages 66 and 124.)

Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, January 2000. (Cited on pages 14 and 117.)

Laurent Le Brusquet, Arthur Tenenhaus, Gisela Lechuga, Vincent Perlbarg, Louis Puybasset, and Damien Galanaud. Une pénalité de groupe pour des données multivoie de grande dimension. In *47èmes Journée de Statistique de la SFdS (JdS 2015)*, Lille, France, June 2015. (Cited on pages 100, 108, and 111.)

Yann Le Guen, Guillaume Auzias, François Leroy, Marion Noulhiane, Ghislaine Dehaene-Lambertz, Edouard Duchesnay, Jean-François Mangin, Olivier Coulon, and Vincent Frouin. Genetic influence on the sulcal pits: On the origin of the first cortical folds. *Cerebral Cortex*, 28(6):1922–1933, apr 2017. (Cited on pages 84 and 85.)

Yann Le Guen, Cathy Philippe, Denis Riviere, Hervé Lemaitre, Antoine Grigis, Clara Fischer, Ghislaine Dehaene-Lambertz, Jean-François Mangin, and Vincent Frouin. eQTL of KCNK2 regionally influences the brain sulcal widening: evidence from 15, 597 UK biobank participants with neuroimaging data. *Brain Structure and Function*, 224(2):847–857, dec 2018. (Cited on pages ix, 84, and 85.)

Gisela Lechuga, Laurent Le Brusquet, Vincent Perlbarg, Louis Puybasset, Damien Galanaud, and Arthur Tenenhaus. Discriminant analysis for multiway data. In *Springer Proceedings in Mathematics & Statistics*, number 115-126 in The Multiple Facets of Partial Least Squares and Related Methods. 2016. (Cited on page 42.)

Olivier Ledoit and Michael Wolf. A well conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88:365–411, 2004. (Cited on pages 7 and 86.)

Sue E. Leurgans, Rana A. Moyeed, and Bernard W. Silverman. Canonical correlation analysis when the data are curves. *Journal of the Royal Statistical Society. Series B*, 55:725–740, 1993. (Cited on page 36.)

Charles F.Van Loan. The ubiquitous kronecker product. *Journal of Computational and Applied Mathematics*, 123(1-2):85–100, November 2000. (Cited on page 14.)

Eric F. Lock. Tensor-on-tensor regression. *Journal of Computational and Graphical Statistics*, 27(3): 638–647, 2018. (Cited on page 36.)

Eric F. Lock, Katherine A. Hoadley, J. S. Marron, and Andrew B. Nobel. Joint and individual variation explained (JIVE) for integrated analysis of multiple data types. *The Annals of Applied Statistics*, 7 (1):523–542, March 2013. (Cited on page 10.)

Samuel N. Lockhart and Charles DeCarli. Structural imaging measures of brain aging. *Neuropsychology Review*, 24(3):271–289, August 2014. (Cited on page 84.)

Tommy Löfstedt, Fouad Hadj-Selem, Vincent Guillemot, Cathy Philippe, Nicolas Raymond, Edouard Duchesney, Vincent Frouin, and Arthur Tenenhaus. A general multiblock method for structured variable selection. In *Proceedings of the 8th International Conference on Partial Least Squares and Related Methods, Paris, France*. 2016. (Cited on pages 65, 69, 76, 77, 88, 108, 123, and 129.)

Marco Lorenzi, Andre Altmann, Boris Gutman, Selina Wray, Charles Arber, Derrek P Hibar, Neda Jahanshad, Jonathan M Schott, Daniel C Alexander, Paul M Thompson, Sebastien Ourselin, and for the Alzheimer's Disease Neuroimaging Alzheimer's Disease Neuroimaging Initiative. Susceptibility of brain atrophy toTRIB3in Alzheimer's disease, evidence from functional prioritization in imaging genetics. *PNAS*, 115(12):3162–3167, 3 2018. ISSN 1091-6490. (Cited on page 92.)

Haiping Lu. Learning canonical correlations of paired tensor sets via tensor-to-vector projection. In *Twenty-Third International Joint Conference on Artificial Intelligence, Beijing, China*, 2013. (Cited on page 37.)

Robert R. Meyer. Sufficient conditions for the convergence of monotonic mathematicalprogramming algorithms. *Journal of computer and system sciences*, 12(1):108–121, 1976. (Cited on page 23.)

Eun Jeong Min, Eric C. Chi, and Hua Zhou. Tensor canonical correlation analysis. *Stat*, 8(1):e253, 2019. e253 sta4.253. (Cited on pages 37 and 54.)

Yurii Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1): 127–152, December 2004. (Cited on pages 65, 69, 123, and 129.)

Guillaume Obozinski, Laurent Jacob, and Jean-Philippe Vert. Group lasso with overlaps: the latent group lasso approach, 2011. (Cited on page 65.)

Ann A. O'Connell, Ingwer Borg, and Patrick Groenen. *Modern Multidimensional Scaling: Theory and Applications*, volume 94. 2006. ISBN 9780387251509. (Cited on page 124.)

Samia Samar Ouertani, Gérard Mazerolles, Julien Boccard, Serge Rudaz, and Mohamed Hanafi. Multiway pls for discrimination: Compact form equivalent to the tri-linear pls2 procedure and its monotony convergence. *Chemometrics and Intelligent Laboratory Systems*, 133:25–32, 2014. (Cited on page 116.)

Matthieu Perrot, Denis Rivière, and Jean-François Mangin. Cortical sulci recognition and spatial normalization. *Medical Image Analysis*, 15(4):529–550, aug 2011. (Cited on page 85.)

Amicie de Pierrefeu. *Machine Learning with Structured Sparsity : application to Neuroimaging-based Phenotyping in Autism Spectrum Disorder and Schizophrenia*. Theses, Université Paris-Saclay, October 2018. (Cited on pages 71 and 125.)

Randall J. Pruim, Ryan P. Welch, Serena Sanna, Tanya M. Teslovich, Peter S. Chines, Terry P. Gliedt, Michael Boehnke, Gonçalo R. Abecasis, and Cristen J. Willer. LocusZoom: regional visualization of genome-wide association scan results. *Bioinformatics*, 26(18):2336–2337, July 2010. (Cited on page 84.)

Denis Rivière, Dominique Geffroy, Isabelle Denghien, Nicolas Souedet, and Yann Cointepas. BrainVISA: an extensible software environment for sharing multimodal neuroimaging data and processing tools. *NeuroImage*, 47:S163, July 2009. (Cited on page 84.)

Blandine Roig. *Bimodal spectroscopy for in vivo skin characterization: Diffuse Reflectance Spectroscopy and Raman Spectroscopy.* Theses, Université de Reims Champagne Ardenne URCA, Nov 2015. (Cited on page 95.)

Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, November 1992. (Cited on pages 65, 71, and 125.)

Juliane Schäfer and Korbinian Strimmer. A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical applications in genetics and molecular biology*, 4 (1):Article 32, 2005. (Cited on pages 7 and 86.)

Ivan W. Selesnick. Total variation denoising (an mm algorithm). 2012. (Cited on page 72.)

Dhruv B. Sharma, Howard D. Bondell, and Hao Helen Zhang. Consistent group identification and variable selection in regression with correlated predictors. *Journal of Computational and Graphical Statistics*, 22(2):319–340, April 2013. (Cited on pages 126 and 127.)

Xinke Shen, Tao Liu, Dacheng Tao, Yubo Fan, Jicong Zhang, Shuyu Li, Jiyang Jiang, Wanlin Zhu, Yilong Wang, Yongjun Wang, Henry Brodaty, Perminder Sachdev, and Wei Wen. Variation in longitudinal trajectories of cortical sulci in normal elderly. *NeuroImage*, 166:1–9, February 2018. (Cited on page 84.)

Age K. Smilde, Johan A. Westerhuis, and Ricard Boqué. Multiway multiblock component and covariates regression models. *Journal of Chemometrics*, 14(3):301–331, 2000. (Cited on page 36.)

Age K. Smilde, Johan A. Westerhuis, and Sijmen de Jong. A framework for sequential multiblock component methods. *Journal of Chemometrics*, 17(6):323–337, 2003. (Cited on page 10.)

Age K. Smilde, Rasmus Bro, and Paul Geladi. Multi way analysis — applications in chemical sciences. 01 2004. (Cited on page 14.)

Jos MF ten Berge. *Least squares optimization in multivariate analysis.* DSWO Press, Leiden University Leiden, The Netherlands, 1993. (Cited on pages 8, 17, 33, 44, 51, and 115.)

Jos MF Ten Berge, Henk A.L. Kiers, and Véronique Van der Stel. Simultaneous components analysis. *Statistica Applicata*, 4(4):277–392, 1992. (Cited on page 8.)

Arthur Tenenhaus and Vincent Guillemot. RGCCA: Regularized and Sparse Generalized Canonical Correlation Analysis for Multi-Block Data. *https://CRAN.R-project.org/package=RGCCA*, R package version 2.1, 2017. (Cited on page 44.)

Arthur Tenenhaus and Michel Tenenhaus. Regularized Generalized Canonical Correlation Analysis. *Psychometrika*, 76:257–284, 2011. (Cited on pages 7, 9, and 36.)

Arthur Tenenhaus, Cathy Philippe, Vincent Guillemot, Kim-Anh Le Cao, Jacques Grill, and Vincent Frouin. Variable selection for generalized canonical correlation analysis. *Biostatistics (Oxford, England)*, 15(3):569–83, 2014. (Cited on pages 58, 59, 88, and 109.)

Arthur Tenenhaus, Laurent Le Brusquet, and Gisela Lechuga. Multiway Regularized Generalized Canonical Correlation Analysis. In *47èmes Journée de Statistique de la SFdS (JdS 2015)*, Lille, France, June 2015. (Cited on page 37.)

Michel Tenenhaus, Arthur Tenenhaus, and Patrick J. F. Groenen. Regularized Generalized Canonical Correlation Analysis: A Framework for Sequential Multiblock Component Methods. *Psychometrika*, May 2017. (Cited on pages 6, 8, 20, 26, 36, 87, 106, and 115.)

Markus Thom and Günther Palm. Efficient Sparseness-Enforcing Projections. pages 1–15, 2013. (Cited on page 63.)

Ledyard R. Tucker. An inter-battery method of factor analysis. *Psychometrika*, 23:111–136, 1958. (Cited on pages 8 and 9.)

Ledyard R. Tucker. Implications of factor analysis of three-way matrices for measurement of change. In C. W. Harris, editor, *Problems in measuring change.*, pages 122–137. University of Wisconsin Press, Madison WI, 1963. (Cited on pages 18 and 36.)

Ledyard R. Tucker. The extension of factor analysis to three-dimensional matrices. In H. Gulliksen and N. Frederiksen, editors, *Contributions to mathematical psychology.*, pages 110–127. Holt, Rinehart and Winston, New York, 1964. (Cited on pages 18 and 36.)

John P. Van de Geer. Linear relations among k sets of variables. *Psychometrika*, 49:70–94, 1984. (Cited on page 9.)

Ewout van den Berg, Mark Schmidt, M. P. Friedlander, and Kevin Murphy. Group Sparsity Via Linear-Time Projection. Technical report tr-2008-09, Department of Computer Science, University of British Columbia, 2008. (Cited on pages 61 and 63.)

Gertjan van den Burg. *Algorithms for Multiclass Classification and Regularized Regression.* PhD thesis, E, January 2018. (Cited on pages 128 and 129.)

Arnold L. Van den Wollenberg. Redudancy analysis: an alternative for canonical correlation analysis. *Psychometrika*, 42:207–219, 1977. (Cited on pages 8 and 9.)

Katrijn Van Deun, Tom F. Wilderjans, Robert A. van den Berg, Anestis Antoniadis, and Iven Van Mechelen. A flexible framework for sparse simultaneous component based data integration. *BMC Bioinformatics*, 12:448, Nov 2011. (Cited on pages 124, 125, and 126.)

Hrishikesh D. Vinod. Canonical ridge and econometrics of joint production. *Journal of Econometrics*, 4:147–166, 1976. (Cited on page 36.)

Eric Westman, J-Sebastian Muehlboeck, and Andrew Simmons. Combining mri and csf measures for classification of alzheimer's disease and prediction of mild cognitive impairment conversion. *NeuroImage*, 62(1):229–238, August 2012. ISSN 1053-8119. (Cited on pages 89 and 95.)

Daniela M. Witten, Robert Tibshirani, and Trevor Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3): 515–534, Jul 2009. (Cited on pages 36, 60, 111, and 119.)

Herman Wold. Soft Modeling: The Basic Design and Some Extensions. In *Systems under indirect observation, Part 2, K.G. Jöreskog and H. Wold (Eds), North-Holland, Amsterdam*, pages 1–54, 1982. (Cited on pages 9 and 36.)

Svante Wold, Nouna Kettaneh, and Kjell Tjessem. Hierarchical multiblock pls and pc models for easier model interpretation and as an alternative to variable selection. *Journal of Chemometrics*, 10(5-6):463–482. (Cited on page 9.)

Donghyeon Yu, Joong-Ho Won, Taehoon Lee, Johan Lim, and Sungroh Yoon. High-dimensional fused lasso regression using majorization–minimization and parallel processing. *Journal of Computational and Graphical Statistics*, 24(1):121–153, January 2015. (Cited on page 76.)

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, February 2006. (Cited on pages 65, 71, and 124.)

Willard I. Zangwill. *Nonlinear programming : a unified approach*. Englewood Cliffs, N.J. : Prentice-Hall, 1969. Bibliography: p. 332-345. (Cited on page 110.)

Tong Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11(35):1081–1107, 2010. (Cited on page 127.)

Hua Zhou, Lexin Li, and Hongtu Zhu. Tensor regression with applications in neuroimaging data analysis. *Journal of the American Statistical Association*, 108(502):540–552, 2013. (Cited on pages 36 and 37.)

Huichen Zhu, Gen Li, and Eric F Lock. Generalized integrative principal component analysis for multi-type data with block-wise missing structure. *Biostatistics*, 21(2):302–318, 09 2018. ISSN 1465-4644. (Cited on page 111.)

Yassine Zniyed, Sebastian Miron, Remy Boyer, and David Brie. Uniqueness of tensor train decomposition with linear dependencies. In *XXVIIème Colloque francophone de traitement du signal et des images, GRETSI 2019*, Lille, France, August 2019. (Cited on page 15.)

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Un cadre statistique et algorithmique pour l'analyse de données multibloc et multivoie

Mots clés : Statistique Multivarié, Analyse Canonique Généralisée, Analyse de données multivoie, Parcimonie, Neuroimagerie, Génétique

Résumé :
L'étude des relations entre plusieurs ensembles de variables mesurées sur un même groupe d'individus est un défi majeur en statistique. La littérature fait référence à ce paradigme sous plusieurs termes : "analyse de données multimodales", "intégration de données", "fusion de données" ou encore "analyse de données multibloc". Ce type de problématique se retrouve dans des domaines aussi variés que la biologie, la chimie, l'analyse multi-capteurs, le marketing, la recherche agro-alimentaire, où l'objectif commun est d'identifier les variables de chaque bloc intervenant dans les intéractions entre blocs. Par ailleurs, il est possible que chaque bloc soit composé d'un très grand nombre de variables ($\sim$1M), nécessitant le calcul de milliards d'associations. L'élaboration d'un cadre statistique épousant la complexité et l'hétérogénéité des données est donc primordial pour mener une analyse pertinente.
Le développement de méthodes d'analyse de données hétérogènes, potentiellement de grande dimension, est au coeur de ce travail. Ces développements se basent sur l'Analyse Canonique Généralisée Régularisée (RGCCA), un cadre général pour l'analyse de données multiblocs. Le coeur algorithmique de RGCCA se résume à un unique "update", répété jusqu'à convergence. Si cet update possède certaines "bonnes" propriétés, la convergence globale de l'algorithme est garantie. Au cours de ces travaux, le cadre algorithmique de RGCCA a été étendu dans plusieurs directions :
Du séquentiel au global. Plutôt que d'extraire de chaque bloc les composantes de manière séquentielle, un problème d'optimisation globale permettant de construire ces composantes simultanément a été proposé.
De la matrice au tenseur. L'Analyse Canonique Généralisée Multivoie (MGCCA) étend RGCCA à l'analyse conjointe d'un ensemble de tenseurs. Des versions séquentielle et globale de MGCCA ont été proposées. La convergence globale de ces algorithmes est montrée.
De la parcimonie à la parcimonie structurée. Le coeur de l'algorithme d'Analyse Canonique Généralisée Parcimonieuse (SGCCA) a été amélioré en fournissant un algorithme à convergence globale beaucoup plus rapide. Des contraintes de parcimonie structurée ont également été ajoutées à SGCCA.
Dans une seconde partie, l'analyse de plusieurs jeux de données est menée à l'aide de ces nouvelles méthodes. La polyvalence des ces outils est démontrée sur (i) deux études en imagerie-génétique, (ii) deux études en électroencéphalographie ainsi (iii) qu'une étude en microscopie Raman. L'accent est mis sur l'interprétation des résultats facilitée par la prise en compte des structures multiblocs, tensorielles et/ou parcimonieuses.

Title : A statistical and computational framework for multiblock and multiway data analysis

Keywords : Multivariate Statistics ; Canonical Correlation Analysis ; Multiway Analysis ; Sparsity ; Neuroimaging ; Genetics ;

Abstract :
A challenging problem in multivariate statistics is to study relationships between several sets of variables measured on the same set of individuals. In the literature, this paradigm can be stated under several names as "learning from multimodal data", "data integration", "data fusion" or "multiblock data analysis". Typical examples are found in a large variety of fields such as biology, chemistry, sensory analysis, marketing, food research, where the common general objective is to identify variables of each block that are active in the relationships with other blocks. Moreover, each block can be composed of a high number of measurements ($\sim$1M), which involves the computation of billion(s) of associations. A successful investigation of such a dataset requires developing a computational and statistical framework that fits both the peculiar structure of the data as well as its heterogeneous nature.
The development of multivariate statistical methods constitutes the core of this work. All these developments find their foundations on Regularized Generalized Canonical Correlation Analysis (RGCCA), a flexible framework for multiblock data analysis that grasps in a single optimization problem many well known multiblock methods. The RGCCA algorithm consists in a single yet very simple update repeated until convergence. If this update is gifted with certain conditions, the global convergence of the procedure is guaranteed. Throughout this work, the optimization framework of RGCCA has been extended in several directions :
From sequential to global. We extend RGCCA from a sequential procedure to a global one by extracting all the block components simultaneously with a single optimization problem.
From matrix to higher order tensors Multiway Generalized Canonical Correlation Analysis (MGCCA) has been proposed as an extension of RGCCA to higher order tensors. Sequential and global strategies have been designed for extracting several components per block. The different variants of the MGCCA algorithm are globally convergent under mild conditions.
From sparsity to structured sparsity The core of the Sparse Generalized Canonical Correlation Analysis (SGCCA) algorithm has been improved. It provides a much faster globally convergent algorithm. SGCCA has been extended to handle structured sparse penalties.
In the second part, the versatility and usefulness of the proposed methods have been investigated on various studies : (i) two imaging-genetic studies, (ii) two Electroencephalography studies and (iii) one Raman Microscopy study. For these analyses, the focus is made on the interpretation of the results eased by considering explicitly the multiblock, tensor and sparse structures.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France