Statistical control of sparse models in high dimension

Jerôme-Alexis Chevalier

université
PARIS-SACLAY

Statistical control of sparse model in high dimension

Thèse de doctorat de l'Université Paris-Saclay

École doctorale n° 580, Sciences et technologies de l'information et de la communication (ED STIC)
Spécialité de doctorat: Mathématiques et Informatique
Unité de recherche: Université Paris-Saclay, Inria, Inria Saclay-Île-de-France, 91120, Palaiseau, France
Référent: Faculté des sciences d'Orsay

Thèse présentée et soutenue en visioconférence totale, le 11 décembre 2020, par

Jérôme-Alexis CHEVALIER

Composition du jury:

Christophe Ambroise Professeur, Université d'Évry Val d'Essonne	Président
Chloé-Agathe Azencott Maîtresse de Conférence, HDR, Mines ParisTech	Rapportrice et Examinatrice
Thomas Nichols Professeur, University of Oxford	Rapporteur et Examinateur
Pierre Neuvial Chargé de Recherche, HDR, Université de Toulouse	Examinateur
Bertrand Thirion Directeur de Recherche, INRIA Paris Saclay	Directeur de thèse
Joseph Salmon Professeur, Université de Montpellier	Co-encadrant et Examinateur

Thèse de doctorat

NNT: 2020UPASG051

STATISTICAL CONTROL OF SPARSE MODELS IN HIGH DIMENSION

JÉRÔME-ALEXIS CHEVALIER

ABSTRACT

In this thesis, we focus on the multivariate inference problem in the context of high-dimensional structured data. More precisely, given a set of explanatory variables (features) and a target, we aim at recovering the features that are predictive conditionally to others, *i.e.,* recovering the support of a linear predictive model. We concentrate on methods that come with statistical guarantees since we want to have a control on the occurrence of false discoveries. This is relevant to inference problems on high-resolution images, where one aims at pixel- or voxel-level analysis, *e.g.,* in neuroimaging, astronomy, but also in other settings where features have a spatial structure, *e.g.,* in genomics. In such settings, existing procedures are not helpful for support recovery since they lack power and are generally not tractable. The problem is then hard both from the statistical modeling point of view, and from a computation perspective. In these settings, feature values typically reflect the underlying spatial structure, which can thus be leveraged for inference. For example, in neuroimaging, a brain image has a 3D representation and a given voxel is highly correlated with its neighbors. In the present work, we notably propose the ensemble of clustered desparsified Lasso (ecd-Lasso) estimator that combines three steps: *i)* a spatially constrained clustering procedure that reduces the problem dimension while taking into account data structure, *ii)* the desparsified Lasso (d-Lasso) statistical inference procedure that is tractable on reduced versions of the original problem, and *iii)* an ensembling method that aggregates the solutions of different compressed versions of the problem to avoid relying on only one arbitrary data clustering choice. Additionally, we extend this procedure to handle temporal data corrupted with autocorrelated noise. We consider new ways to control the occurrence of false discoveries with a given spatial tolerance. This control is well adapted to spatially structured data. We study the behavior of the procedures that we propose, by establishing their theoretical properties and conducting thorough empirical validations. In this work, we focus on neuroimaging datasets but the methods that we present can be adapted to other fields which share similar setups.

ACKNOWLEDGMENTS

I deeply thank my PhD advisors Bertrand Thirion and Joseph Salmon for their help and support during these 3 years preparing my PhD project. I also thank the Parietal team assistant, Corinne Petitot, and to the ED STIC assistant, Stéphanie Druetta, for their generous help with administrative matters. I would also like to thank the rest of the Parietal team —past and present— for ensuring such a nice work environment. I also want to thank PhD students and professors of Telecom ParisTech as I enjoyed working with them during the first year of my thesis. Finally, I am thankful to all the members of the jury: Christophe Ambroise, Chloé-Agathe Azencott, Thomas Nichols and Pierre Neuvial, for kindly accepting to evaluate my PhD project.

CONTENTS

ACRONYMS

BOLD Blood Oxygenation Level Dependent
d-Lasso desparsified Lasso
cd-Lasso clustered desparsified Lasso
ecd-Lasso ensemble of clustered desparsified Lasso
ECOG Electrocorticography
EEG Electroencephalography
FDR False Discovery Rate
FPR False Positive Rate
fMRI functional Magnetic Resonance Imaging
FReM Fast Regularized Ensembles of Models
FWER Family-Wise Error Rate
GLM General Linear Model
HCP Human Connectome Project
HRF Haemodynamic Response Function
MEG Magnetoencephalography
MR Magnetic Resonance
MRI Magnetic Resonance Imaging
OLS Ordinary Least Squares
PET Positron Emission Tomography
ROI Region of Interests
RSVP Rapid-Serial-Visual-Presentation
SNR signal to noise ratio
SPM Statistical Parametric Mapping
SVD Singular Value Decomposition
SVM Support Vector Machines
SVR Support Vector Regression

NOTATION

We denote scalars by lower-case letters, vectors by bold lower-case letters, and matrices by bold upper-case letters.

Notation	Name	Definition
$[p]$	integers from 1 to p (inclusive)	$\{1, 2, \ldots, p\}$
$\mathbf{x}_i$	i-th element of $\mathbf{x}$	
$\mathbf{X}_{i,j}$	Element (i, j) of $\mathbf{X}$	
$\mathbf{X}_{i,.}$	i-th row of $\mathbf{X}$	
$\mathbf{X}_{.,j}$	j-th column of $\mathbf{X}$	
$\mathbf{X}^{(-j)}$	$\mathbf{X}$ without its j-th column	
$\mathbf{X}^\top$	Transpose of $\mathbf{X}$	
$\|\mathbf{x}\|$	Vector euclidean norm (ℓ_2 norm)	$\left(\sum_{i=1}^n x_i^2\right)^{1/2}$
$\|\mathbf{x}\|_p$	Holder norm (ℓ_p norm)	$(\sum_{i=1}^n \|x_i\|^p)^{\frac{1}{p}}$
$\mathrm{Tr}(\mathbf{X})$	Trace of $\mathbf{X}$	
$\mathrm{diag}(\mathbf{X})$	Diagonal of $\mathbf{X}$	
$\|\mathbf{a}\|^2_{\mathbf{M}^{-1}}$	Mahalanobis norm of $\mathbf{a}$ for $\mathbf{M}$	$\mathrm{Tr}(\mathbf{a}^\top \mathbf{M}^{-1} \mathbf{a})$
$\|\mathbf{A}\|$	Frobenius norm	$\sqrt{\mathrm{Tr}(\mathbf{A}\mathbf{A}^\top)}$
$\|\mathbf{B}\|_{2,1}$	(2, 1)-matrix norm	$\sum_{j=1}^p \|\mathbf{B}_{j,.}\|$

SYNTHÈSE EN FRANÇAIS

Cette thèse s'intéresse au problème de l'inférence statistique multivariée en grande dimension en présence de données structurées. Plus précisément, étant données une variable cible et un ensemble de variables explicatives, nous souhaitons déterminer les variables explicatives qui sont prédictives conditionnellement aux autres, *i.e.*, nous cherchons à identifier le support dans le modèle prédictif linéaire. Comme nous désirons avoir un contrôle sur l'occurrence de faux positifs, nous nous concentrons sur les méthodes donnant des garanties statistiques. Cette étude s'applique notamment aux problèmes d'inférence sur des images haute-résolution dans lesquels le signal de chaque pixel ou voxel est considéré comme une variable explicative, c'est par exemple le cas en neuro-imagerie ou en astronomie. Cela peut également s'appliquer à d'autres problèmes dans lesquels les variables explicatives sont spatialement structurées comme en génomique par exemple. Pour ce type de données, les méthodes existantes destinées à l'identification de support ne sont pas satisfaisantes car elles manquent de puissance et ont généralement un coût computationnel trop élevé. Par conséquent, le problème est difficile en terme de modélisation statistique mais aussi du point de vue computationnel. Cependant, dans ce type de problème, les variables explicatives détiennent une structure spatiale qui peut être exploitée. Par exemple, en neuro-imagerie, une image de cerveau possède une représentation 3D dans laquelle un voxel est très corrélé à ses voisins.

Pour répondre à la problématique que nous venons de présenter, la thèse écrite en anglais s'organise comme suit. Le premier chapitre est une vue d'ensemble de la thèse décrivant chaque chapitre et mettant en avant les publications de l'auteur. Dans le deuxième chapitre, après avoir présenté quelques notions utiles de neuro-imagerie, nous introduisons la modélisation mathématique du problème que nous souhaitons résoudre. Il s'agit du problème inverse en grande dimension. Dans le troisième chapitre, nous faisons une revue des méthodes statistiques qui s'appliquent au modèle linéaire en grande dimension. En particulier, nous montrons que la méthode appelée "desparsified Lasso" est compétitive en terme de puissance statistique et possède des propriétées statistiques désirables. Dans le quatrième chapitre, nous introduisons la méthode "ensemble of clustered desparsified Lasso" (ecd-Lasso) qui combine trois éléments : *i)* une procédure de clustering avec contraintes spatiales pour réduire la dimension du problème en tenant compte de la structure de la donnée ; *ii)* la méthode d'inférence statistique "desparsified Lasso" qui peut être déployée sur le problème réduit ; et *iii)* une méthode d'ensembling qui agrège les solutions obtenues

sur les différents problèmes réduits afin d'éviter de dépendre d'un choix de clustering nécessairement imparfait et arbitraire. Cette méthode est au coeur de la thèse et les chapitres suivants visent à évaluer dans quelle mesure elle répond correctement à notre problématique, à mieux comprendre son fonctionnement, ses atouts et ses défauts, ou à l'étendre pour résoudre des problèmes connexes. Dans le cinquième chapitre, nous établissons les propriétés statistiques de ecd-Lasso : il contrôle l'occurence de faux positifs qui sont spatialement éloignés du support. Pour cela nous présentons une nouvelle façon de contrôler l'occurrence de faux positifs qui intègre une tolérance spatiale qui est mesurée par un paramètre correspondant à une distance. Dans le sixième chapitre, nous analysons le comportement empirique de ecd-Lasso en conduisant des expériences variées à partir de différents jeux de données de neuro-imagerie. Dans le septième chapitre, nous prolongeons ecd-Lasso afin qu'il puisse gérer des données ayant une dimension temporelle qui nécessite la modélisation d'un bruit auto-corrélé. Enfin, dans le huitième et dernier chapitre, nous récapitulons les contributions et proposons des axes de développement. Tout au long de cette thèse, nous nous focalisons sur des jeux de données de neuro-imagerie, mais les méthodes que nous présentons sont applicables à d'autres domaines qui partagent une configuration semblable.

OVERVIEW

1 GENERAL OVERVIEW OF THE THESIS

Here, we present the context and motivations or our work, then we summarize the content of the thesis highlighting our contributions. We also mention some additional contributions that are not presented in this document as we focus on our core contribution.

1.1 CONTEXT

In many scientific fields, data-acquisition devices have benefited of hardware improvement to increase the resolution of the observed phenomena, leading to ever larger datasets. For example, recorded images in medical imaging, in seismology or in astronomy can contain hundreds of thousands of pixels or voxels. Also, in genomics, we are now able to analyse Single Nucleotide Polymorphisms (SNPs) of a population, that typicall reach several millions. However, while the number of explanatory variables has increased, the number of samples remains limited, due to time, physical or financial constraints. Additionally, signal-to-noise ratio is also often limited by the physics of the measurement process resulting in datasets with low contrasts and requiring advanced statistical modeling.

Multivariate statistical models are used to explain a response of interest through a combination of measurements (pixels, voxels, SNPs). For instance, in neuroimaging, one might predict the age of a subject from its gray matter density map. Such an analysis may reveal i) to which extent gray matter density maps predict age and ii) which regions of the brain carry useful information for the prediction. Such multivariate estimators are viewed as powerful tools because they leverage the distribution of information across all measurements. Yet, an unavoidable difficulty is that they suffer from the curse of dimensionality. Another key issue is to perform reliable inference on these data, *i.e.,* inference that comes with some statistical guarantees. Altogether the problem is hard both from the statistical modeling point of view, and from a computation perspective.

Yet it turns out that high-dimensional data often display some spatial structure that can be leveraged. For example, in imaging problems, neighboring voxels are generally very similar; in genomics, there exist blocs of SNPs that tend to always occur together. Another helpful feature regarding high-dimensional inference problems is that they often lead to sparse models

since only a small proportion of the explanatory variables are truly predictive. For example, in neuroimaging, only few regions of a brain activity maps are necessary to predict some behavioral conditions; in genomics, only few alleles may be responsible for a disease.

In this thesis, focusing on neuroimaging datasets, we propose to address the multivariate high-dimensional statistical problem leveraging data structure. As suggested above, the methods we present in this document adapt to other fields which share similar setups.

1.2 ORGANIZATION OF THE THESIS

The thesis is organized in two parts. In the first part we introduce the problem from the neuroimaging angle, and then we present existing statistical tools that we will leverage to address the problem. In the second part, we present our main contributions.

INTRODUCTION TO THE DECODING PROBLEM. In Chapter 2, we present the core concepts that are needed to describe and formalize the problem we are willing to solve. Mainly working with fMRI (functional Magnetic Resonance Imaging) data, we mostly focus on the decoding problem, *i.e.,* the problems that lead to inferring behavioral or phenotypical information from brain activity using brain images, generally from several subjects. We state the mathematical formulation of the problem that we are targeting: identify the contribution of brain regions in the prediction.

STATISTICAL INFERENCE IN HIGH DIMENSION. In Chapter 3, we explain why naive solutions to the conditional inference decoding problem are bound to fail. Indeed, we show that the number of samples is too low with regards to the number of features to perform the statistical inference we are aiming at with standard tools. Then, we review the methods available in the literature that are suited for a number of samples of the same order as the number of features and benchmark them on a simulation. In particular, we review the desparsified Lasso (d-Lasso) procedure that we will leverage in the next chapters.

ENSEMBLE OF CLUSTERED DESPARSIFIED LASSO. In Chapter 4, we introduce two algorithms for high-dimensional multivariate statistical inference on structured data. This chapter mainly revisits our first publication at the 2018 *MICCAI* conference:

CHEVALIER, Jérôme-Alexis, SALMON, Joseph, et THIRION, Bertrand. Statistical inference with ensemble of clustered desparsified lasso. In : International

Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, 2018. p. 638-646.

The algorithms we present, that we called clustered desparsified Lasso (cd-Lasso) and ensemble of clustered desparsified Lasso (ecd-Lasso), are notably well suited for high-dimensional structured data such as neuroimaging data. We focus on explaining the rationale behind each step of the proposed algorithms along with experimental results to illustrate the potential of this approach.

STATISTICAL INFERENCE WITH SPATIAL TOLERANCE. In Chapter 5, we give the statistical guarantees provided by cd-Lasso and ecd-Lasso. We show that ecd-Lasso controls a generalization of the Family-Wise Error Rate (FWER) called δ-FWER, that takes into account a spatial tolerance of radius δ for the occurrence of false discoveries. This result is true under realistic assumptions and for a predetermined spatial tolerance parameter δ.

EMPIRICAL VALIDATION. In Chapter 6, we evaluate the statistical properties of ecd-Lasso along with three alternative standard methods by performing a thorough empirical study using functional Magnetic Resonance Imaging (fMRI) datasets. We also study the impact of the choice of the main free parameter of ecd-Lasso: the number of clusters C. Finally, we show that ecd-Lasso exhibits the best recovery properties while ensuring the expected statistical control. Also note that the content of this chapter has been submitted to the *NeuroImage* journal and is undergoing some revisions.

EXTENSION TO TEMPORAL DATA WITH APPLICATIONS TO MEG. In Chapter 7, we extend our work to the magnetoencephalography (MEG) and electroencephalography (EEG) source localization setup. This chapter mainly present our work accepted at the 2020 *NeuRIPS* conference:

CHEVALIER, Jérôme-Alexis, GRAMFORT, Alexandre, SALMON, Joseph, et al. Statistical control for spatio-temporal MEG/EEG source imaging with desparsified multi-task Lasso. In: Advances in Neural Information Processing Systems, 2020.

M/EEG source imaging requires working with spatio-temporal data and autocorrelated noise. To deal with this, we adapt the d-Lasso estimator to temporal data corrupted with autocorrelated noise by leveraging the debiased group Lasso estimators and introducing the desparsified multi-task Lasso (d-MTLasso). We combine d-MTLasso with spatially constrained clustering to reduce data dimension and with ensembling to mitigate the arbitrary choice of clustering; the resulting estimator is called ensemble of clustered desparsified multi-task Lasso (ecd-MTLasso). With respect to the current procedures, the two advantages of ecd-MTLasso are that *i)* it offers statistical guarantees and *ii)* it trades spatial specificity for sensitivity,

leading to a powerful adaptive method. Extensive simulations on realistic head geometries, as well as empirical results on various MEG datasets, demonstrate the high recovery performance of ecd-MTLasso and its primary practical benefit: offer a statistically principled way to threshold MEG/EEG source maps.

1.3 OTHER CONTRIBUTIONS

COLLABORATIVE WORK ON FDR-CONTROLLING PROCEDURES. In this thesis, we do not present the contribution regarding the Knockoff filters which is a False Discovery Rate (FDR) controlling procedure for high-dimensional data. Firstly, we proposed an adaptation of this procedure to high-dimensional structured data with application on fMRI datasets. In that respect, we had an accepted paper at the 2019 *IPMI* conference (equal contribution with Tuan-Binh Nguyen):

NGUYEN, Tuan-Binh, CHEVALIER, Jérôme-Alexis, et THIRION, Bertrand. ECKO: Ensemble of Clustered Knockoffs for multivariate inference on fMRI data. In: International Conference on Information Processing in Medical Imaging. Springer, Cham, 2019. p. 454-466.

Secondly, we contributed as second author in another study presented at the *ICML 2020* conference, which aims at increasing the stability of the Knockoff filters procedure:

NGUYEN, Tuan-Binh, CHEVALIER, Jérôme-Alexis, THIRION, Bertrand, et ARLOT, Sylvain. Aggregation of Multiple Knockoffs. In: 37th International Conference on Machine Learning, PMLR 119, 2020, Vienne, Austria.

COLLABORATIVE WORK ON PYTHON IMPLEMENTATION. Regarding the implementation and testing of the procedures we designed, we propose a package called HiDimStat developed conjointly with Tuan-Binh Nguyen that is available at `https://github.com/ja-che/hidimstat`. Our algorithms are implemented with Python = 3.6 and need the following packages Numpy = 1.16.2 (Walt, Colbert, and Varoquaux, 2011), Scipy = 1.2.1 (Virtanen et al., 2020), Scikit-Learn = 0.21 (Pedregosa et al., 2011), Joblib = 0.11 and Nilearn = 0.6.0 (Abraham et al., 2014).

Part I

INTRODUCTION

2 INTRODUCTION TO THE DECODING PROBLEM

In this chapter, we present the core concepts that are needed to describe and formalize the problem we are willing to solve. Mainly working with fMRI (functional Magnetic Resonance Imaging) data, we mostly focus on the decoding problem, *i.e.,* fitting behavioral or phenotypical information from brain activity measurements using brain images, generally from several subjects. We state the mathematical formulation of the problem that we are targeting: identify the contribution of brain regions in the prediction.

2.1 NEUROIMAGING OUTLOOK

In this section, we present fundamental principles and the main modalities that are encountered in neuroimaging. For a general review of the neuroimaging principles and challenges, one can refer to Zimmerman, Gibby, and C. (2012).

2.1.1 Principle of neuroimaging

Neuroimaging consists in acquiring brain images from a human subject or an animal. In this thesis we focus on human brain imaging.

Neuroimaging falls into two broad categories: structural imaging that deals with the structure (matter) of the nervous system, and functional imaging, that aims at capturing brain functional activity (*i.e.,* brain activity related to a state or a cognitive process). Most of our work will deal with functional neuroimaging.

2.1.2 Neuroimaging modalities

The techniques (or modalities) used to acquire brain images are numerous. In both functional or structural imaging, the modalities differ by their level of invasivity (e.g. need of opening the skull or not), their spatial resolution, their temporal resolution and the physiological mechanism they measure.

Structural imaging typically offers high spatial resolution since tissue contrasts can be captured over long periods of time. Indeed, in general (for a given level of invasivity), increasing the spatial resolution requires lowering

the temporal resolution. Additionally, in structural imaging, modalities also differ by the types of tissue they can characterize.

In Fig. 2.1, we compare different modalities. Except Magnetic Resonance Imaging (MRI), which can be classified as structural or functional imaging depending on the specific MRI sequence, all the presented modalities are functional imaging.

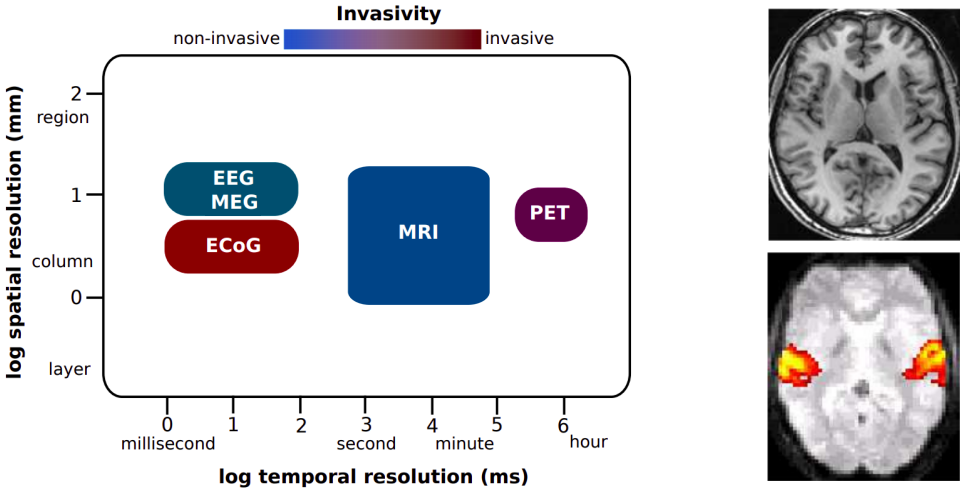


Figure 2.1: Comparison of neuroimaging modalities. Left: Spatial and temporal resolution of different neuroimaging modalities. Right: Examples of T1/anatomical MRI image (top) and fMRI image (bottom). The T1 MRI image has a high resolution and notably observes the gray matter, white matter and skull. The fMRI image series is used to yield the regions activated in an auditory task, represented in color. Its spatial resolution is a bit lower.

MRI uses magnetic fields and radio-frequency pulses to produce high quality brain images. There exist many MRI sequences —a sequence being a given setting of radiofrequency pulses and gradients, resulting in a particular signal, yielding specific images. We will mainly focus on functional Magnetic Resonance Imaging (fMRI) that measures brain activity by detecting changes in oxygen flow but also consider T1 MRI that notably recovers gray matter regions at mm spatial resolution. We will give a brief presentation of T1 MRI in Sec. 2.2. In turn, we provide a brief description of fMRI in Sec. 2.3.

Electroencephalography (EEG) and Magnetoencephalography (MEG) record the neural activity of the brain with a high temporal resolution by measuring the electric or magnetic field at the cortical surface. We will describe this technique in Sec. 2.5 and work with M/EEG datasets in Chapter 7.

Finally, Positron Emission Tomography (PET) that relies on a radioactive tracer to track glucose consumption is an invasive functional brain technique. Electrocorticography (ECOG) which measures electric field with

electrodes implanted at the surface of brain. We will not further describe these modalities as we do not deal with such datasets.

A deeper review of the different modalities in functional imaging can be found in Friston (2009).

2.2 MAGNETIC RESONANCE IMAGING

In the 70s, after the invention of the MRI technique by Lauterbur (1973), the first MRI image was acquired by Damadian, Goldsmith, and Minkoff (1977). This discovery is particularly important for neuroimaging since it contributed to the development of a very widely used technique called functional MRI (fMRI) developed initially by Ogawa et al. (1990).

In this section, we provide a brief review of the MRI acquisition principles. For a thorough review of MRI, the reader may refer to Vlaardingerbroek and Boer (2013) and Bushong and Clarke (2013).

2.2.1 MRI technique in a nutshell

During an MRI acquisition, a subject is placed in an MRI scanner. The MRI technique consists of three main phases that are repeated during all the acquisition process.

The first phase is the magnetization during which a constant magnetic field is applied to the atoms of the brain —MRI targets hydrogen nuclei. Thanks to the nuclear magnetic resonance of the atoms, it is possible to capture the *net magnetization* vector which is the sum of the *magnetic momenta* of all atoms —an atom magnetic momentum being induced by its energy level. The norm of the net magnetization directly influences the intensity of the Magnetic Resonance (MR) signal that will be recorded during the third phase.

The second phase is the excitation, during which the atoms are excited, *i.e.,* loaded in energy, using a radio-frequency pulse at the resonance frequency of magnetized atoms.

The third phase is the relaxation. During this phase, radio frequency pulse ceases and atoms come back to their original magnetized state, releasing the absorbed energy. This release is recorded as an MR signal.

They are two main components in the MR signal. The first is due to the longitudinal relaxation, *i.e.,* the recovery of atoms to their original magnetized state. The second is due to the transversal relaxation, *i.e.,* the phase decay that corresponds to a loss of atoms' spin phase coherence.

T1 recovery measures the duration of the longitudinal relaxation, while *T2 decay* characterized the duration of the transversal relaxation. The T1 is always longer than the T2, generally by a factor 10.

2.2.2 Echo-planar imaging

To map the MR signal across spatial locations, the magnetic fields are applied in three orthogonal directions. Then, choosing the intensity of the magnetic field properly, it is possible to recover the spatial origin of the MR signal.

In practice, we do not directly acquire a 3D image as 3D-MRI images are reconstructed by combining sequences of 2D-slices of the brain. *Echo-planar imaging* is one such 2D acquisition scheme.

2.2.3 Structural MRI

Structural imaging provides a static characterization of brain tissues. In this thesis, we worked with T1-weighted MRI images, thus we do not assess the whole range of techniques and applications of structural MRI and refer the reader to Bushong and Clarke (2013).

The T1-weighted MRI images are acquired using specific *echo time* and *repetition time*. The repetition time is the time that separates two consecutive excitation phases, *i.e.*, the total duration needed to record one 2D-slice. The echo time is a shorter duration related to the excitation phase. Compared to T2-weighted MRI images, both the echo time and the repetition time are longer. Consequently, the two techniques observe different tissues. For example, T1-weighted MRI images notably focus on fatty tissue while T2-weighted MRI images focus on high water content tissues.

2.3 FUNCTIONAL MRI

Functional MRI (fMRI) is a non-invasive imaging technique that is particularly adapted to the mapping of cognitive functions. In this thesis, several experiments use fMRI datasets, thus we summarize how fMRI datasets are obtained and analyzed. We only give the basic notions of fMRI data acquisition and primary analysis, for an extensive reference on this matter, see for example Poldrack (2011).

2.3.1 Heamodynamic response function and BOLD signal

When a neuron is activated —one says that the neuron spikes—, it requires oxygen, which is provided by hemoglobin. Consequently, in any volume

of the brain, after a neuronal activity increase, we observe an increase of oxygenated blood flow during 5 to 10 seconds and then an undershoot that lasts 10 to 20 seconds. This phenomenon is called the Blood Oxygenation Level Dependent (BOLD) response (or signal) and is detectable through MRI. Recording and analyzing this signal allows for the recovery of brain activity in any part of the brain.

As described above, the BOLD signal recorded by MRI has a specific temporal pattern which is called the Haemodynamic Response Function (HRF). We represent a canonical HRF pattern in Fig. 2.2.

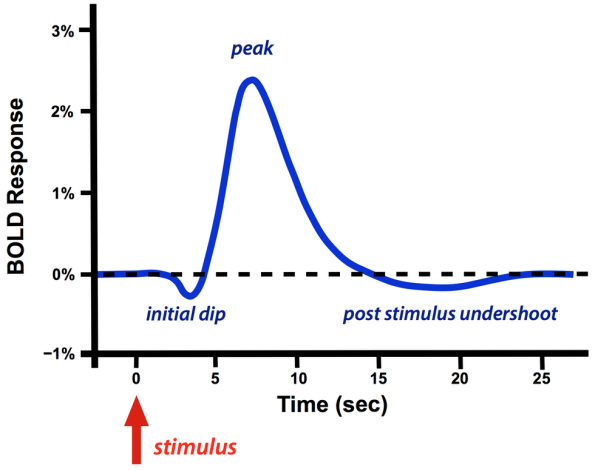


Figure 2.2: Heamodynamic response function. This represents the BOLD reponse following a brief neural activity event.

2.3.2 Data acquisition

During an fMRI acquisition, a subject is asked to lie in an MRI scanner. The scanner periodically records the BOLD signal. This process allows for the generation of a 3D image with a given periodicity that contains the BOLD signal measurement in each brain voxel, a voxel being a small cubic volume.

With the fMRI modality and a typical strength value of the magnetic field of 1.5 to 7 Tesla, one can expect to record an image with a spatial resolution of 1 to 27 mm^3. Thus, every volume is typically discretized into around 10^5 voxels. A typical value of the period, *i.e.*, temporal resolution, goes from 0.7 seconds to 2.5 seconds.

Finally, for each voxel, we obtain one measurement per time point forming *voxel time-series*. Then, the whole acquisition results in a 4D data array.

2.3.3 Preprocessing

Before analyzing fMRI data, several preprocessing steps are performed:

i) general quality check control;

ii) spatial distortion correction, to correct scanner-related artifacts;

iii) motion correction, to correct subject movements or breathing;

iv) slice timing correction, since slices are sequentially recorded;

v) spatial normalization, to align brains of different subjects in a common reference space called template (see, *e.g.,* Evans et al. (2012)) in order to allow inter-subject analysis;

vi) spatial smoothing, that can be applied to increase the signal to noise ratio (SNR), *i.e.,* to decrease noise, based on a spatial homogeneity assumption;

vii) temporal filtering, to reduce low-frequency noise.

2.3.4 Resting-state fMRI and task fMRI

Resting-state consists in acquiring fMRI images of a subject that has been asked to rest in the scanner, *i.e.,* to do nothing in particular. Task fMRI records brain activation of subjects who are asked to perform a task or are exposed to a stimulus. The acquisition process thus yields labeled data. By contrast resting-state fMRI leads to unlabeled data.

In this thesis both resting-state fMRI data and task fMRI data will be used but most of the experiments rely on task fMRI data.

2.3.5 Intra-subject and inter-subject analysis

In fMRI studies, the acquisition protocol is generally performed on many subjects and is often repeated several times by each subject. Then, fMRI data analysis might be performed at the subject-level (intra-subject), leading to so-called of *first-level analysis*, or at the group-level (inter-subject), leading to *second-level analysis*. Notably, in order to perform a second-level analysis, the results of first-level analyses of several subjects must be handled. Then, when performing first-level analyses, it is convenient to record single-subject brain images into a common space, *e.g.,* the MNI template (Fonov et al., 2009). This avoids the undesirable effects of the variability in brain shape during the second-level analysis. In this work, when running second-level analysis, we assume that subjects are aligned in some common spatial reference.

2.4 TASK FMRI ANALYSIS

Thanks to task fMRI data, it is possible to link brain activation and condition, *i.e.*, a behavior (task, stimuli, etc.) or a disease status. Either one may want to predict (or infer) brain activity maps from a condition —an *encoding* type of analysis—, or one may be willing to predict conditions from brain maps —performing a *decoding* analysis.

In this thesis we mainly focus on decoding models. In this section we describe the analysis of the raw preprocessed fMRI data which is required for such analysis.

2.4.1 First-level analysis

When conducting second-level analyses, encoding and decoding models are not directly constructed from the raw BOLD signal recorded by the MRI scanner. Indeed, the objective of first-level analysis is to produce individual statistical maps that will be subsequently used in second-level analyses. Second-level analyses consider multiple subjects and may aim at producing encoding or decoding models. We now explain how to construct the statistical maps (z-maps) from the fMRI recordings of one subject.

GENERAL LINEAR MODEL. A z-map corresponds to a statistical map of the brain (with one value per brain voxel) that represents brain activity in response to a given mental condition. To produce such maps we model the BOLD signal as the linear combination of weighted and convolved mental condition descriptors. A mental condition descriptor is simply a vector of the same size as the voxel time-series that describes a condition. For example, if we consider the condition "left-hand action", the entries of the descriptor are set to 1 whenever the participant actions his/her left hand, and 0 otherwise. Given that the MRI scanner does not measure directly brain activity but the BOLD signal which is close to a convolution of brain activity with the HRF, condition descriptors are convolved with the HRF pattern.

To model the BOLD signal, a common practice is to link linearly each voxel time-series independently with the convolved descriptors and confounding variables, this model is referred to as General Linear Model (GLM) in the literature (Friston et al., 1994). Note that the confounding variables are added to capture nuisance effects such as movements in the scanner.

Mathematically, we denote by $\mathbf{R} \in \mathbb{R}^{T \times k}$ the convolved descriptors and confounding variables, where T is the length of the experiment and k is the number of conditions and confounders. Also, $\mathbf{S} \in \mathbb{R}^{T \times p}$ stacks the voxel time-series, $\mathbf{B} \in \mathbb{R}^{k \times p}$ contains the β-maps (regression coefficients) for each condition and confounder, where p is the number of voxels used to represent

the brain, and $\mathbf{E}_{\text{flm}} \in \mathbb{R}^{T \times k}$ denotes noise of the first-level model (flm). Then, the GLM model yields

$$\mathbf{S} = \mathbf{R}\mathbf{B} + \mathbf{E}_{\text{flm}} \ . \tag{2.1}$$

ESTIMATING THE β-MAPS. To estimate the β-maps, *i.e.*, the columns of $\mathbf{B}$, the most standard method is to use the Ordinary Least Squares (OLS) estimator. Denoting by $\hat{\mathbf{B}}$ the estimator of $\mathbf{B}$, the OLS yields

$$\hat{\mathbf{B}} = (\mathbf{R}^\top \mathbf{R})^{-1} \mathbf{R}^\top \mathbf{S} \ . \tag{2.2}$$

Then to produce the noise normalized statistics we need to estimate the estimator variance for each voxel $j \in [p]$ and each condition $c \in [k]$, denoted by $\hat{\mathbf{V}}_{c,j}$:

$$\hat{\mathbf{V}}_{c,j} = \hat{\mathbf{h}}_j^2 (\mathbf{R}^\top \mathbf{R})^{-1}_{c,c} \ , \tag{2.3}$$

where

$$\hat{\mathbf{h}}_j^2 = \frac{\left\| \mathbf{S}_{\cdot,j} - \mathbf{R}\hat{\mathbf{B}}_{\cdot,j} \right\|_2^2}{p - k} \ . \tag{2.4}$$

Note that this previous solution is subject to adaptations, as the BOLD noise is autocorrelated. These adaptations are quite straightforward, and we refer to Monti (2011) for more details.

DERIVING THE Z-MAPS. Finally, to compute the z-maps we first derive the t-statistics for each voxel of each β-map. The t-statistics of the j-th voxel and the c-th condition denoted by $\mathbf{T}_{c,j}$ is given by

$$\mathbf{T}_{c,j} = \frac{\hat{\mathbf{B}}_{c,j}}{\sqrt{\hat{\mathbf{V}}_{c,j}}} \ . \tag{2.5}$$

Then, those t-statistics are converted into z-scores such that the implicit p-values remain the same. This last transformation is performed to facilitate statistical image interpretation, as the significance of z-scores is no longer bound to hyper-parameters, such as degrees of freedom.

2.4.2 Encoding

In second-level analyses, we aim at studying effects that statistically occur across a group of subjects. A first type of second-level analysis, called encoding, consists in inferring z-maps from conditions or combinations (contrasts) of conditions. To perform such an analysis we need the z-maps for several subjects. Say that we have m subjects available in our study.

First, we select two conditions c_1 and c_2 (in some cases we might choose more than two conditions as soon as they can be ranked on a given scale), and stack the z-maps $\mathbf{T}_{c_1,.}$ and $\mathbf{T}_{c_2,.}$ of all subjects by row in a matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ where $n = 2m$. We also construct a vector $\mathbf{y} \in \{-1, 1\}^n$ (or sometimes $\mathbf{y} \in \mathbb{R}^n$) that contains the condition label or value —keeping the correspondence between the rows of $\mathbf{X}$ and the rows of $\mathbf{y}$.

Then, to tackle the encoding problem one can consider the following linear model (see *e.g.*, Thirion (2016)):

$$\mathbf{X} = \mathbf{y}\mathbf{b}^\top + \mathbf{E}_{\text{enc}} \ , \tag{2.6}$$

where $\mathbf{b} \in \mathbb{R}^p$ is a parameter vector that corresponds to the typical z-map related to conditions c_1 or c_2 and $\mathbf{E}_{\text{enc}}$ is the noise in the encoding (enc) model. To address this problem, a common solution is to apply the same method as the one we used to estimate the β-maps. Since we independently fit one model per voxel, we also refer to this analysis as mass univariate analysis or marginal analysis.

2.4.3 Decoding

We now introduce a second type of second-level analysis, called decoding, in which we infer conditions from z-maps.

CONTEXT OF DECODING. Nowadays, predicting behavior or diseases status from brain images is an important analytical approach for imaging neurosciences, as it provides an effective evaluation of the information carried by brain images (Kriegeskorte, Goebel, and Bandettini, 2006). Indeed, supervised learning tools are often used on brain images to infer cognitive states (Cox and Savoy, 2003; Haynes and Rees, 2006; Norman et al., 2006) or to perform diagnosis or prognosis (Demirci et al., 2008; Fan et al., 2008). Brain images are obtained from MRI imaging, or even EEG- or MEG-based volume-based activity reconstruction (cf. Sec. 2.5). They are used to predict a *target* outcome: binary (*e.g.,* two-condition tasks), discrete (*e.g.,* multiple-condition tasks) or continuous (*e.g.,* age). The decoding models used for such predictions are most often linear models, characterized by a weight map that can be represented as a brain image (Mourao-Miranda et al., 2005; Varoquaux and Thirion, 2014).

Besides the prediction accuracy achieved, this estimated weight map is crucial to assess the information captured by the model. Indeed, it reflects the importance of brain structures in the predictive model. Unlike standard encoding analysis, this feature importance is tested *conditional on other brain features*, *i.e.*, it assesses whether each feature *adds* to information conveyed by other features. Weichwald et al. (2015) highlights the fact that decoding

and encoding are complementary, and that making the conditional analysis (or multivariate analysis) helps for causal interpretation regarding the implication of brain regions in the outcome of interest (see also Haufe et al. (2014)). Typically, the produced weight maps are used to identify discriminative patterns (Gramfort, Thirion, and Varoquaux, 2013; Haxby et al., 2001; Mourao-Miranda et al., 2005) and support reverse inferences (Poldrack, 2011; Schwartz, Thirion, and Varoquaux, 2013; Varoquaux et al., 2018), *i.e.*, conclude on the implication of brain regions in the studied process.

FORMALIZING THE DECODING PROBLEM. The target (condition to decode) is observed in n samples and still denoted by $\mathbf{y} \in \mathbb{R}^n$ ($\mathbf{y}$ can be binary, discrete or continuous). The brain volume is still discretized into p voxels. The corresponding p voxel signals are also referred to as explanatory variables, covariates or features. We denote by $\mathbf{X} \in \mathbb{R}^{n \times p}$ the matrix containing (column-wise) the p covariates $\{\mathbf{X}_1, \dots, \mathbf{X}_p\}$ of several subjects, it also still corresponds to the z-maps stacked by rows. Then, assuming a linear dependency between the covariates and the response, the decoding model is the following:

$$\mathbf{y} = \mathbf{X}\mathbf{w}^* + \boldsymbol{\varepsilon} \ , \tag{2.7}$$

where $\mathbf{w}^* \in \mathbb{R}^p$ is the true weight map and $\boldsymbol{\varepsilon}$ is the noise vector. In the present work, we assume that the noise is Gaussian, *i.e.*, $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma_\varepsilon^2 \mathbf{I}_n)$, but extension to sub-Gaussian noise is possible.

In this thesis, focusing on the inverse problem introduced by (2.7), we aim at estimating $\mathbf{w}^*$ with statistical guarantees. Ideally, we would like to *recover* the non-zero parameters of $\mathbf{w}^*$, also called support of $\mathbf{w}^*$, with a control on the false discovery.

PROBLEM SETTING. At first sight, the problem described by (2.7) may appear to be quite simple. Indeed, in small dimension ($n > p$), with moderate noise, a well-known solution is the OLS method. Also, we will see that there are several procedures that work when $n < p$, with p of the same order as n (see Chapter 3).

However, in the task fMRI setting, we are far from these settings, as n is in the order of 100 and p is in the order of 10^5. This corresponds to the case $n \ll p$ which is hard to solve in practice.

Another important feature is the spatial structure of the data. Indeed, the covariates in $\mathbf{X}$ exhibit short-and long-range correlations since neighboring voxels are highly correlated, yet remote voxels can also be correlated due to the connectivity of brain regions. Also, we assume that only few regions are involved in a cognitive task, then few voxels are predictive, *i.e.*, $\mathbf{w}^*$ is sparse, and those voxels are spatially concentrated. In general, the number of predictive voxels is expected to be less than 10% of total number of voxels.

2.5 MAGNETO/ELECTROENCEPHALOGRAPHY (M/EEG)

In this section, we briefly describe the EEG and MEG modalities. These also introduce an inverse problem that we aim at solving. For a complete review of those modalities, the reader may refer to to Niedermeyer and Silva (2005) for EEG and to Hämäläinen et al. (1993) for MEG.

2.5.1 MEG and EEG principles

EEG measures the electric field produced by brain activity, while MEG modality records the magnetic field. Both techniques are non-invasive as they capture the signal through sensors positioned at the surface of the scalp. Since sensors directly record the electric or magnetic field, the time resolution of M/EEG acquisitions is much better than in fMRI: between 1 and 5ms. However, the spatial resolution is poorer than in fMRI, it is around 1cm3.

2.5.2 The M/EEG inverse problem

In M/EEG, the source localization problem, a.k.a. M/EEG inverse problem, refers to the search of the location of the brain activity from the electric or magnetic signals recorded at sensor level. The source space corresponds to the discretization of brain cortical surface into multiple predefined prospective source locations, modeled as dipoles. Thanks to the Maxwell's equations, the electric or magnetic measurements made in each sensor are linear combinations of the emission arising from the dipoles.

We denote by $\mathbf{X} \in \mathbb{R}^{n \times p}$ the forward operator given by the Maxwell's equation, $\mathbf{Y} \in \mathbb{R}^{n \times T}$ the sensors' measurements and $\mathbf{B} \in \mathbb{R}^{p \times T}$ the (unknown) emissions in the source space, where n is the number of sensors, p is the number of dipoles and T is the number of recorded time points. Then the M/EEG inverse problem is defined by

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E} \ , \tag{2.8}$$

where $\mathbf{E} \in \mathbb{R}^{n \times T}$ is the noise of the model. In (2.8), $\mathbf{B}$ is also referred to as true parameter matrix. We aim at recovering its non-zero rows with a statistical control on the false discoveries.

The remarks made about high-dimensionality and structure of the data with respect to the fMRI decoding problem are also applicable to this problem; n being in the order of 100 and p in the order of 10^4.

3 STATISTICAL INFERENCE IN HIGH DIMENSION

In this chapter, we explain why naive solutions to the conditional inference decoding problem are bound to fail. Indeed, we show that the number of samples is too low with regards to the number of features to perform the statistical inference we are aiming at with standard tools. Then, we review the methods available in the literature that are suited for a number of samples of the same order as the number of features and benchmark them on a simulation. In particular, we review the d-Lasso procedure that we will leverage in the next chapters.

3.1 INTRODUCTION

In this section, after reviewing the linear model formulation, we introduce OLS (Ordinary Least Squares) and Lasso regression that are fundamental for our problem since most of the solutions proposed by the literature rely on these elementary procedures. Then, we show that there is no hope to perform a powerful statistical inference when the number of samples n is much lower than the number of features p, *i.e.*, $n \ll p$.

3.1.1 Linear Model

We consider the following model (2.7):

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon} \ ,$$

where $\mathbf{y} \in \mathbb{R}^n$ denotes the response vector, $\mathbf{X} \in \mathbb{R}^{n \times p}$ the design matrix, $\boldsymbol{\beta}^* \in \mathbb{R}^p$ the parameter vector and $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma_\varepsilon^2 \mathbf{I}_n)$ the random error vector where $\sigma_\varepsilon > 0$ is its unknown amplitude. Our aim is to recover the support, defined by $S(\boldsymbol{\beta}^*) = \{j \in [p] : \boldsymbol{\beta}^*_j \neq 0\}$, with a statistical control on the number of false discoveries. Additionally, we denote by $s(\boldsymbol{\beta}^*) = |S(\boldsymbol{\beta}^*)|$ the support size and assume that the true model is sparse, meaning that $\boldsymbol{\beta}^*$ has a small number of non-zero entries, *i.e.*, $s(\boldsymbol{\beta}^*) \ll p$.

3.1.2 Ordinary least square regression

Here, we consider the low-dimensional setting ($p < n$), and assume a fixed, full rank design matrix $\mathbf{X}$, *i.e.*, $\operatorname{rank}(\mathbf{X}) = p$. The (normalized) Gram

matrix $\hat{\mathbf{\Sigma}} = \mathbf{X}^\top \mathbf{X}/n$ is then invertible. Then, the OLS estimator defined by $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = \operatorname{argmin}_{\boldsymbol{\beta}\in\mathbb{R}^p} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$ is given by

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = (n\hat{\mathbf{\Sigma}})^{-1}\mathbf{X}^\top \mathbf{y} \tag{3.1}$$

$$= \boldsymbol{\beta}^* + \sigma_\varepsilon (n\hat{\mathbf{\Sigma}})^{-1}\mathbf{X}^\top \boldsymbol{\varepsilon} \ . \tag{3.2}$$

Then, since the noise is Gaussian, we obtain

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} \sim \mathcal{N}(\boldsymbol{\beta}^*, \sigma_\varepsilon^2 (n\hat{\mathbf{\Sigma}})^{-1}) \ . \tag{3.3}$$

When σ_ε is unknown, it can be estimated by

$$\hat{\sigma}^2_{\mathrm{OLS}} = \frac{\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}\|_2^2}{n-p} \ , \tag{3.4}$$

then, the normalized entries of $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}$ follow a Student law with $n-p$ degrees of freedom. With (3.1)-(3.4), one can compute confidence intervals on the entries of $\boldsymbol{\beta}^*$. For more details about the OLS regression, one can refer to Goldberger (1991).

3.1.3 Lasso

In this section, we study the properties of the Lasso support: we present the compatibility condition introduced in Bühlmann and van de Geer (2011) and the "beta-min" assumption that ensures the screening property of the Lasso.

For a given regularization parameter $\lambda > 0$, the Lasso estimator $\hat{\boldsymbol{\beta}}^{L(\lambda)}$ of $\boldsymbol{\beta}^*$, notably introduced by Chen and Donoho (1994) and Tibshirani (1996), is given by

$$\hat{\boldsymbol{\beta}}^{L(\lambda)} \in \operatorname*{argmin}_{\boldsymbol{\beta}\in\mathbb{R}^p} \left(\frac{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2}{n} + \lambda \|\boldsymbol{\beta}\|_1 \right) \ . \tag{3.5}$$

We say that the compatibility condition (Bühlmann and van de Geer, 2011) with constant $\phi_0 > 0$ holds if, for all $\boldsymbol{\beta}$ that verifies $\|\boldsymbol{\beta}_{S^c(\boldsymbol{\beta}^*)}\|_1 \leqslant 3\|\boldsymbol{\beta}_{S(\boldsymbol{\beta}^*)}\|_1$, we have

$$\left\|\boldsymbol{\beta}_{S(\boldsymbol{\beta}^*)}\right\|_1^2 \leqslant \frac{s(\boldsymbol{\beta}^*)}{\phi_0^2} \boldsymbol{\beta}^\top \hat{\mathbf{\Sigma}} \boldsymbol{\beta} \ . \tag{3.6}$$

This assumption is purely technical but it can be seen as a combination of a sparsity assumption and a moderate feature correlation assumption.

Proposition 3.1.1 (Theorem 6.1 of Bühlmann and van de Geer (2011)). *Assume that the model is truly linear (2.7), the columns of* $\mathbf{X}$ *are standardized,* i.e., $\operatorname{diag}(\mathbf{\Sigma}) = 1$ *and the compatibility condition is verified, then taking* $\lambda \geqslant 4\sigma_\varepsilon \sqrt{\frac{t^2 + 2\log(p)}{n}}$, *with probability at least* $1 - 2\exp(-t^2/2)$, *we have*

$$\left\|\hat{\boldsymbol{\beta}}^{L(\lambda)} - \boldsymbol{\beta}^*\right\|_1 \leqslant 4\frac{\lambda s(\boldsymbol{\beta}^*)}{\phi_0^2} \ , \tag{3.7}$$

Adding up the "beta-min" assumption, we obtain the screening property of the Lasso. Let us denote by $\beta^*_{\min}$ the lowest non-zero entry of β^* in absolute value, *i.e.*, $\beta^*_{\min} = \min_{j \in S(\beta^*)} |\beta^*_j|$. Then, making the following beta-min assumption

$$\beta^*_{\min} > 4\lambda s(\beta^*)/\phi_0^2 \ , \tag{3.8}$$

we have, with high probability, the screening property:

$$S(\hat{\beta}^{L(\lambda)}) \supseteq S(\beta^*) \ . \tag{3.9}$$

This last result means that the estimated Lasso support contains all the predictive features. This is an interesting property for our problem. However, the compatibility assumption and the beta-min assumption are often unmet in practice and hard to check.

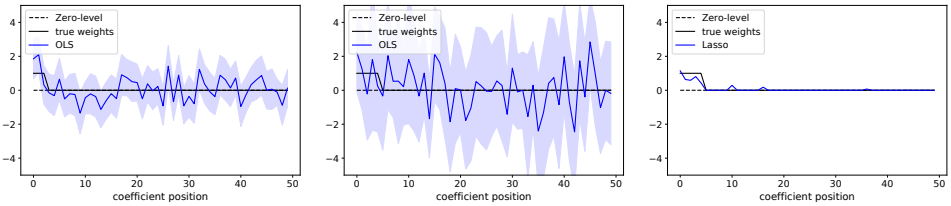
3.1.4 Curse of dimensionality

Now, we show that it is not possible to perform a powerful statistical inference when $n \ll p$.

PEDAGOGICAL EXAMPLES. To give intuition, we first present four different simulations with $n = 200$ and $p \in \{100; 180; 500; 5000\}$. In order to have a sparse reduced setting, we have taken $s(\beta^*) = \lfloor \frac{3p}{100} \rfloor \in \{3; 5; 15; 150\}$. Then, the true parameter vector β^* is defined by $\beta^*_j = 1$ for $1 \leqslant j \leqslant s(\beta^*)$ and $\beta^*_j = 0$ otherwise. The design matrix $\mathbf{X}$ contains normally distributed covariates, where every covariate is correlated at $\rho = 0.9$ with two other covariates randomly. We also set $\sigma_\varepsilon = 2$ to reflect noise regime observed in fMRI datasets. Defining the signal to noise ratio (SNR) by $\mathrm{SNR}_y = \|\mathbf{X}\beta^*\|_2^2 / \|\epsilon\|_2^2$, we have $\mathrm{SNR}_y = 1$ ($= 0\,\mathrm{dB}$) when $p = 100$, $\mathrm{SNR}_y = 1.5$ ($= 2\,\mathrm{dB}$) when $p = 180$, $\mathrm{SNR}_y = 5$ ($= 7\,\mathrm{dB}$) when $p = 500$ and $\mathrm{SNR}_y = 50$ ($= 17\,\mathrm{dB}$) when $p = 5000$.

Practically, to construct $\mathbf{X}$, we have drawn covariates as Gaussian vectors with a Toeplitz covariance matrix and then shuffle the covariates to avoid having a 1D spatial structure related to feature weights. The reason why we do not integrate a strong spatial structure in those scenarios is that in this chapter we mainly aim at studying the efficiency of statistical inference procedures when applied on compressed versions of the original problem. Then, if the dimension reduction is well realized, the original spatial structure is exploited and correlated features in the compressed problem have less chance to share the same effects. However, we propose different level of compression and when $p = 5000$ (low compression) there are still some blocks of correlated covariates that share the same effects. Note that in Chapter 4, we study the dimension reduction process; since we want to show the

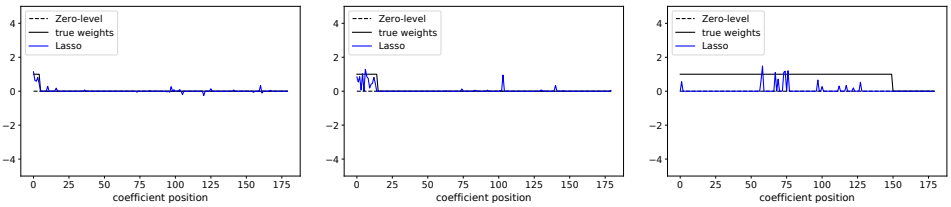
fact that we can leverage on data structure to ease inference, we propose similar scenarios without making the shuffling, *i.e.*, keeping a strong spatial structure.



(a) OLS: p=100 (b) OLS: p=180 (c) Lasso: p=180

Figure 3.1: Small dimension example. In (a), when p is significantly lower than n, the OLS 95% confidence intervals yield the support accurately. However, when p gets closer to n, the OLS solution may fail to identify the support (b). Lasso may yield the support in this setting (c) but does not provide confidence intervals.

In Fig. 3.1, we run the example with $p \in \{100; 180\}$. In Fig. 3.1-(a) and Fig. 3.1-(b), we can see that when p increases the problem gets harder and the OLS 95% confidence intervals do not identify the support when p is close to n. We recall that the OLS method is valid only if $n > p$. In Fig. 3.1-(c), we compare with the Lasso estimator. Even when $p = 180$, Lasso may recover the support however it does not provide confidence intervals.



(a) Lasso: p=180 (b) Lasso: p=500 (c) Lasso: p=5000

Figure 3.2: High dimension example. In (a) and (b), when p is lower than n or when p is slightly larger than n, the Lasso solution recovers the support decently. However, when p gets significantly larger than n, the Lasso solution is not satisfactory due to unfulfilled assumptions. In (c), p remains "only" 25 times larger than n, in fMRI datasets p can be around 1000 times larger than n.

In Fig. 3.2, we run the example with $p \in \{180; 500; 5000\}$; the number of features remains at least 20 times lower than in fMRI datasets. As in the previous example (see Fig. 3.1), one can observe that the problem becomes more difficult with a larger p. In Fig. 3.2-(a) and Fig. 3.2-(b), the Lasso which is almost always a fundamental element in most of the statistical inference procedures in high dimension (see Sec. 3.2) can decently recover the support when p is lower or slightly larger than n. However, when p

gets significantly larger than n, the estimator produced by the Lasso is not satisfactory anymore to recover the support due to unfulfilled assumptions (see Fig. 3.2-(c)). More precisely in that settings, to get the screening property (3.9), noticing that we always have $\phi_0^2 \geqslant s(\boldsymbol{\beta}^*)$, we would need to have at least $\boldsymbol{\beta}^*_{\min} \geqslant 10$.

These two examples illustrate the fact that it is highly over-optimistic to search for a solution of the original problem with $n \approx 100$ and $p \approx 10^5$ without preliminary dimension reduction.

THEORETICAL ARGUMENTS. In the original case where $n \ll p$, if we assume that the size of the support is between 1% and 10% of p, we have $n < s(\boldsymbol{\beta}^*)$. Then, without making any additional assumptions, the parameter vector is not identifiable. Indeed, since $\text{rank}(\mathbf{X}) \leqslant n < s(\boldsymbol{\beta}^*)$, it is clear that there exist an infinity of vectors $\boldsymbol{\theta} \neq \boldsymbol{\beta}^*$ such that $\mathbf{X}\boldsymbol{\theta} = \mathbf{X}\boldsymbol{\beta}^*$. Then, in this case, there is no hope to recover the support.

Besides, most of the statistical inference procedures for the high-dimensional setting are based on the Lasso screening property. From (3.6) and (3.8), it is clear that at least we need that $s(\boldsymbol{\beta}^*)$ remains "not too large" in front of $\sqrt{n/\log(p)}$. In the neuroimaging setting in which $\sqrt{n/\log(p)} \approx 3$, this assumption is problematic.

Furthermore, the study of Wainwright (2009) gives an interesting impossibility result on the screening properties of the Lasso. More precisely, it provides a threshold on the sample size to ensure possibility or impossibility to recover the support from the Lasso with high probability. In classic experimental neuroimaging settings, for p in the order of 10^5 and $s(\boldsymbol{\beta}^*) = \lfloor 0.03p \rfloor$, the threshold would be at least $n \geqslant 10^4$; since we only have n in the order of 100, it is very unlikely that the Lasso recovers the support.

OVERVIEW. We have shown that while the OLS solution produces confidence intervals, it fails when p becomes close to n. Also, the Lasso can handle the $n < p$ regime but does not produce confidence intervals (or p-values) and cannot handle the $n \ll p$ regime. Then, knowing that almost all the statistical inference procedures that work in high dimension leverage the Lasso, it clearly appears that the original problem cannot be solved without preliminarily reducing the dimension of the feature space. Solutions to operate dimension reduction will be discussed in Chapter 4. In the following sections, we consider statistical inference procedures producing p-values that can solve compressed versions of the original problem leading to $n < p$ regime but avoiding the $n \ll p$ regime.

3.2 LITERATURE REVIEW

The topic of high-dimensional statistical inference has been addressed in many recent works. In this section, we try to briefly review most of the popular procedures available in the literature before going into a more detailed description of the most promising ones.

3.2.1 Resampling methods

A first class of methods, probably the oldest one, is based on resampling. Among those, the classic bootstrap procedures are generally based on Lasso-type estimators (Bach, 2008; Chatterjee and Lahiri, 2011; Chatterjee, Lahiri, et al., 2013; Liu, Yu, et al., 2013), but also on more refined estimators (Dezeure, Bühlmann, and Zhang, 2017). El Karoui and Purdom (2018) provides an interesting study about the validity of the bootstrap procedure in high dimension, they conclude that the method tends to be anti-conservative. In the same spirit, Minnier, Tian, and Cai (2011) proposes a perturbation resampling-based procedure to approximate the distribution of a an estimator and produce confidence intervals. Finally Meinshausen and Bühlmann (2010) proposed the stability selection procedure that is based on the combination of subsampling with a selection algorithm and derives the probability that a covariate is selected by the selection algorithm. This method is known to be conservative.

A computationally efficient alternative is the single-split procedure introduced by Wasserman and Roeder (2009) that combines a screening step using the Lasso with an inference step using the OLS. It has been improved with randomization and ensembling by Meinshausen, Meier, and Bühlmann (2009): they propose to repeat several screening/inference operations and refer to it as multi-sample split. Sample splitting however results in power loss in a regime where $n \ll p$. In Sec. 3.3 and Sec. 3.4, we give more details for two popular procedures, namely the residual bootstrap Lasso and the multi-sample split.

3.2.2 Post selection inference procedures

In the single-split procedure, one half of the samples is used to make the screening and the other one is used for the inference; this leads to a loss of power. Post-selection inference procedures aim at merging the screening and inference steps into one and then use all the samples in a screening/inference solution. Adjustments must be made when performing the inference with the same samples that were used to do the screening. Several solutions have been proposed. Lockhart et al. (2014) test the significance of the predictor

variables that enter in the Lasso estimator along the Lasso solution path. Berk et al. (2013) propose to produce valid post-selection inference by suitably widening conventional confidence. This results in conservative estimates. Lee et al. (2016) and Tibshirani et al. (2016) characterize the distribution of a post-selection estimator conditioned on the selection event. All these methods notably apply to the Lasso.

However, they scale poorly when p becomes large. Another drawback of such methods is that they produce confidence intervals or p-values only for the selected variables. We aim at deriving p-values for each covariate, hence we did not further consider such methods.

3.2.3 Debiasing procedures

Another class of procedures tries to address the projection bias of classic high dimensional estimators for linear models. Indeed, solutions of the Lasso or Ridge can be seen as projections on a subspace of the original feature space. Under appropriate assumptions, debiased estimators asymptotically follow Gaussian laws. As with the OLS procedure, it is then possible to compute confidence intervals and p-values for all the model parameters. Bühlmann (2013) proposes the corrected Ridge procedure which aims at debiasing the Ridge estimator. Another procedure called "desparsified" (or "debiased") Lasso has recently been investigated by several authors (Javanmard and Montanari, 2014; Zhang and Zhang, 2014; van de Geer et al., 2014), and is still actively developed (Bellec and Zhang, 2019; Celentano, Montanari, and Wei, 2020; Javanmard, Montanari, et al., 2018). As one could expect, this procedure computes an estimator derived from the Lasso but having different nature and properties, *e.g.*, it is not sparse. In Sec. 3.5 and Sec. 3.6, we present in detail the corrected Ridge and the desparsified Lasso (d-Lasso).

3.2.4 Procedures testing groups of covariates

Another class of methods tries to untangle the problem by testing groups of covariates. Meinshausen (2015) provides "group bound" confidence interval, *i.e.*, confidence intervals on the ℓ_1-norm of several parameters, without making any assumptions on the design matrix. This method is known to be conservative in practice (Javanmard, Montanari, et al., 2018; Mitra and Zhang, 2016). Another procedure, referred to as hierarchical testing, developed successively by Blanchard, Geman, et al. (2005), Mandozzi and Bühlmann (2016), and Meinshausen (2008), makes significance tests along the tree of a hierarchical clustering algorithm starting from the root node and descending subsequently into children of rejected nodes. This procedure is constrained by the clustering tree and leverages a plug-in inference procedure, then it

does not provide a new way to compute p-values. We do not go into further detail concerning these methods since, at this stage, we aim at building p-values for each covariate.

3.2.5 Knockoff procedure

A recent method proposed by Barber and Candès (2015) and further developed by Candès et al. (2018) proposes to create "knockoff" variables that mimic the original variables checking whether original variables are selected at random or not. This procedure is quite powerful and suited to control the False Discovery Rate (FDR), which is the number of false discoveries divided by the total number of discover. The FDR is notably different from the FWER (Family-Wise Error Rate) which corresponds to the probability of making at least one false discovery. Then, it is easy to show that controlling the FWER is more restrictive (harder) than controlling the FDR. Extension of the Knockoff to FWER-type control was proposed (Janson and Su, 2016) but it is not very natural and turns out to be very conservative.

Based on the knockoff technique, we have proposed two contributions Nguyen, Chevalier, and Thirion (2019) (equal contribution with Tuan-Binh Nguyen) and Nguyen et al. (2020) (second author). We do not present our work on the FDR controlling procedure in this thesis since we have decided to focus on our core contributions: procedures that yield an FWER-type of control.

3.3 RESIDUAL BOOTSTRAP LASSO

We now detail the residual bootstrap Lasso method by Chatterjee and Lahiri (2011). First, compute the Lasso estimator $\hat{\beta}^{L(\lambda)}$ defined by (3.5), where $\lambda \in \mathbb{R}$ is set by cross validation (see Kohavi et al. (1995) for a reference about cross validation). Then, compute $\tilde{\beta} = \hat{\beta}^{L(\lambda)} \mathbb{1}_{|\hat{\beta}^{L(\lambda)}| > a_n}$ where a_n verifies $a_n + (n^{-1/2} \log n) a_n^{-1} \to 0$ when $n \to +\infty$. Estimated residuals $\hat{\varepsilon}$ are defined by $\hat{\varepsilon} = \mathbf{y} - \mathbf{X}\tilde{\beta}$. The centered estimated residuals are defined by $\tilde{\varepsilon} = \hat{\varepsilon} - \frac{1}{n} \sum_{i=1}^{n} \hat{\varepsilon}_i$. Then, repeat B times, the following procedure:

- Draw n elements from $\tilde{\varepsilon}$, denoted as $\tilde{\varepsilon}^{(b)}$ where $b \in [B]$
- Construct $\mathbf{y}^{(b)} = \mathbf{X}\tilde{\beta} + \tilde{\varepsilon}^{(b)}$
- Solve the Lasso for $\mathbf{y}^{(b)}$ and $\mathbf{X}$ denoting the solution by $\hat{\beta}^{L(\lambda),(b)}$

Then the distribution of $\beta^* - \hat{\beta}^{L(\lambda)}$ can be approximated by $\tilde{\beta} - \hat{\beta}^{L(\lambda),(b)}$. For a more formal formulation of this property, one can refer to Chatterjee and Lahiri (2011). This last result allows for the computation of confidence

intervals and p-values. In the simulations proposed in Sec. 3.7, we have taken $B = 200$.

3.4 MULTI-SAMPLE SPLIT

In this section, we introduce the multi-sample split technique, the two ingredients of this procedure are single-split and ensembling.

3.4.1 Single-split

The single-split procedure was introduced by Wasserman and Roeder (2009) and works as follows. The full sample is divided into two subsamples denoted by $I_{in} = (\mathbf{X}_{in}, \mathbf{y}_{in})$ and $I_{out} = (\mathbf{X}_{out}, \mathbf{y}_{out})$, with $|I_{in}| = \lfloor n/2 \rfloor$. Samples of I_{in} are used to run a Lasso screening step that selects at most $\lfloor n/2 \rfloor$ covariates (see Tibshirani et al. (2013)). Samples of I_{out} are used to compute an OLS regression keeping only the selected covariates. Then, one can derive p-values with respect to the hypothesis tests $H_{0,j} : \beta_j^* = 0$ for $j \in S(\hat{\boldsymbol{\beta}}^{L(\lambda)}(I_{in}))$, where $S(\hat{\boldsymbol{\beta}}^{L(\lambda)}(I_{in}))$ is the estimated Lasso support using I_{in}. Finally, one can define a generalized p-value for each entry of $\boldsymbol{\beta}^*$:

$$\hat{p}_j^{\text{single}} = \begin{cases} \hat{p}_j^{\text{OLS}}, & \text{if } j \in S(\hat{\boldsymbol{\beta}}^{L(\lambda)}(I_{in})) \\ 1, & \text{if } j \notin S(\hat{\boldsymbol{\beta}}^{L(\lambda)}(I_{in})) \end{cases}, \tag{3.10}$$

where $\hat{p}_j^{\text{OLS}}$ is the p-value obtained through the OLS procedure.

3.4.2 Multi sample-splitting

The problem with the single-split is that the solution highly depends on the initial splitting choice. A solution that tries to address this default has been proposed by Meinshausen, Meier, and Bühlmann (2009), the global procedure is called the multi sample-splitting. The idea is to repeat B times the single-split procedure using different random splits. Thanks to this process we collect B p-values for each entry of $\boldsymbol{\beta}^*$. Then, the aim is to aggregate for every $j \in [p]$, the B single-split p-values, denoted by $\hat{p}_j^{\text{single},(b)}$ for $b \in [B]$, into a final multi-split p-value $\hat{p}_j^{\text{multi}}$. The quantile aggregation method proposed by Meinshausen, Meier, and Bühlmann (2009) is defined by

$$\hat{p}_j^{\text{multi}}(\gamma) = \min\left(\gamma\text{-quantile}\left\{\frac{\hat{p}_j^{\text{single},(b)}}{\gamma}; b \in [B]\right\}, 1\right), \tag{3.11}$$

where $\gamma \in (0,1)$ is an arbitrary choice of γ-quantile. Another aggregation procedure was proposed by Meinshausen, Meier, and Bühlmann (2009). We refer to it as adaptive quantile aggregation procedure since it tests for several choice of γ-quantiles. The adaptive quantile aggregation p-values denoted by $\hat{p}_j^{\text{ada, multi}}$ for $j \in [p]$ are defined by

$$\hat{p}_j^{\text{ada, multi}} = \min\left((1-\log(\gamma_{\min})) \inf_{\gamma \in (\gamma_{\min},1)} \hat{p}_j^{\text{multi}}(\gamma), 1\right) , \tag{3.12}$$

where $\gamma_{\min} \in (0,1)$. In the simulations presented in Sec. 3.7, we have taken the adaptive quantile aggregation with $\gamma_{\min} = 0.1$ and $B = 25$.

3.5 CORRECTED RIDGE

In this section, we introduce the corrected Ridge that was developed by Bühlmann (2013).

3.5.1 Singular value decomposition

Before turning to the theory related to the corrected Ridge method, let us consider the Singular Value Decomposition (SVD) of $\mathbf{X} \in \mathbb{R}^{n\times p}$ assuming $n < p$:

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^\top , \tag{3.13}$$

where $\mathbf{U} \in \mathbb{R}^{n\times n}$, $\mathbf{S} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{p\times n}$ have the following properties:

$$\begin{aligned} &\mathbf{U}^\top\mathbf{U} = \mathbf{I}_n , \\ &\mathbf{V}^\top\mathbf{V} = \mathbf{I}_n , \\ &\mathbf{S} = \operatorname{diag}(s_1, s_2, \dots, s_n) . \end{aligned} \tag{3.14}$$

where $s_1 \geqslant s_2 \geqslant \cdots \geqslant s_n > 0$ are the singular values. The projection onto the linear space generated by the rows of $\mathbf{X}$ is defined by $\mathbf{P}_\mathbf{X} = \mathbf{V}\mathbf{V}^\top$. Then, θ^* defined by $\theta^* = \mathbf{P}_\mathbf{X}\beta^*$ verifies $\mathbf{X}\theta^* = \mathbf{X}\beta^*$. Additionally, θ^* is the only element of $\{\theta \in \mathbb{R}^p; \mathbf{X}\theta = \mathbf{X}\beta^*\} \cap \{\theta \in \mathbb{R}^p; \theta = \mathbf{P}_\mathbf{X}\theta\}$. Then, instead of estimating the non-identifiable β^*, one proceeds by estimating θ^* and to bound the quantity $\|\theta^* - \beta^*\|_1$.

3.5.2 Corrected Ridge

For a given regularization parameter $\lambda > 0$, the Ridge estimator $\hat{\beta}^{R(\lambda)}$ is defined by

$$\hat{\beta}^{R(\lambda)} = \operatorname*{argmin}_{\beta\in\mathbb{R}^p} \left(\frac{\|\mathbf{y} - \mathbf{X}\beta\|_2^2}{n} + \lambda \|\beta\|_2^2 \right) . \tag{3.15}$$

The closed solution of this equation is the following:

$$\hat{\boldsymbol{\beta}}^{R(\lambda)} = (\hat{\boldsymbol{\Sigma}} + \lambda \mathbf{I}_p)^{-1} n^{-1} \mathbf{X}^\top \mathbf{y} \ . \tag{3.16}$$

One can check that $\hat{\boldsymbol{\beta}}^{R(\lambda)} = \mathbf{P}_{\mathbf{X}} \hat{\boldsymbol{\beta}}^{R(\lambda)}$. Then, Bühlmann (2013) introduce $\boldsymbol{\Omega}_{R(\lambda)}$ the covariance matrix of the Ridge estimator divided by σ_ε^2 defined by:

$$\boldsymbol{\Omega}_{R(\lambda)} = n^{-1} (\hat{\boldsymbol{\Sigma}} + \lambda \mathbf{I}_p)^{-1} \hat{\boldsymbol{\Sigma}} (\hat{\boldsymbol{\Sigma}} + \lambda \mathbf{I}_p)^{-1} \ , \tag{3.17}$$

and show that:

$$\sigma_\varepsilon^{-1} (\hat{\boldsymbol{\beta}}^{R(\lambda)} - \theta^*) = \mathbf{w} \ , \quad \mathbf{w} \xrightarrow[\lambda \to 0^+]{\mathcal{L}} \mathcal{N}_p(0, \boldsymbol{\Omega}_{R(0^+)}) \ . \tag{3.18}$$

Thanks to (3.18), we have a control on $|\hat{\boldsymbol{\beta}}_j^{R(\lambda)} - \theta_j^*|$ for all $j \in [p]$. Then, the authors link $\boldsymbol{\beta}^*$ and θ^*. From the definition of θ^*, we have

$$\theta_j^* = (\mathbf{P}_{\mathbf{X}})_{j,j} \boldsymbol{\beta}_j^* + \sum_{k \neq j} (\mathbf{P}_{\mathbf{X}})_{j,k} \boldsymbol{\beta}_k^* \ . \tag{3.19}$$

Thus, we have

$$\frac{\theta_j^*}{(\mathbf{P}_{\mathbf{X}})_{j,j}} = \boldsymbol{\beta}_j^* + \sum_{k \neq j} \frac{(\mathbf{P}_{\mathbf{X}})_{j,k}}{(\mathbf{P}_{\mathbf{X}})_{j,j}} \boldsymbol{\beta}_k^* \ . \tag{3.20}$$

Then, (3.20) shows that one can estimate $\boldsymbol{\beta}_j^*$ from θ_j^* with an error equal to $\sum_{k \neq j} [(\mathbf{P}_{\mathbf{X}})_{j,k} \boldsymbol{\beta}_k^* / (\mathbf{P}_{\mathbf{X}})_{j,j}]$. Due to the estimation properties of the Lasso (see Sec. 3.1.3), the authors proposed to estimate the error term by replacing $\boldsymbol{\beta}_k^*$ with $\hat{\boldsymbol{\beta}}_k^{L(\lambda_0)}$ where $\lambda_0 = 4\sigma_\varepsilon \sqrt{2 \log(p)/n}$. Finally, we have all the ingredients to construct a bias-corrected Ridge estimator denoted $\hat{\boldsymbol{\beta}}^{CR(\lambda)}$ and defined by

$$\hat{\boldsymbol{\beta}}_j^{CR(\lambda)} = \frac{\hat{\boldsymbol{\beta}}_j^{R(\lambda)}}{(\mathbf{P}_{\mathbf{X}})_{j,j}} - \sum_{k \neq j} \frac{(\mathbf{P}_{\mathbf{X}})_{j,k}}{(\mathbf{P}_{\mathbf{X}})_{j,j}} \hat{\boldsymbol{\beta}}_k^{L(\lambda_0)} \ . \tag{3.21}$$

Then, with the additional sparsity assumptions that $s(\boldsymbol{\beta}^*) = O((n/\log(p))^\xi)$ for $0 \leqslant \xi < 1/2$, the following property can be shown:

$$\sigma_\varepsilon^{-1} (\hat{\boldsymbol{\beta}}_j^{CR(\lambda)} - \boldsymbol{\beta}_j^*) = \frac{\mathbf{w}_j}{(\mathbf{P}_{\mathbf{X}})_{j,j}} + \sigma_\varepsilon^{-1} \boldsymbol{\Delta}_j \ , \quad \mathbf{w} \xrightarrow[\lambda \to 0^+]{\mathcal{L}} \mathcal{N}_p(0, \boldsymbol{\Omega}_{R(0^+)}) \ , \tag{3.22}$$

where $\boldsymbol{\Delta}_j$ verifies:

$$|\boldsymbol{\Delta}_j| \leqslant \max_{k \neq j} \left| \frac{(\mathbf{P}_{\mathbf{X}})_{jk}}{(\mathbf{P}_{\mathbf{X}})_{j,j}} \right| (\log(p)/n)^{1/2 - \xi} \ . \tag{3.23}$$

A detailed proof of this result is given in Bühlmann (2013). Then, with (3.22) and (3.23) we can derive two-sided confidence intervals with respect to the entry of $\boldsymbol{\beta}^*$.

3.6 DESPARSIFIED LASSO

In this section, we present the d-Lasso procedure that has been proposed in parallel by several authors (Javanmard and Montanari, 2014; Zhang and Zhang, 2014; van de Geer et al., 2014).

3.6.1 Insights from low dimension

First, we give insight about the OLS estimator properties, this will next exhibit how d-Lasso extends the OLS procedure in Sec. 3.6.2. We assume that $p < n$. Starting from model (2.7), let us define $\mathbf{z}_j \in \mathbb{R}^n$ the residual of the OLS regression of $\mathbf{X}_{\cdot,j}$ versus $\mathbf{X}^{(-j)}$ given by:

$$\mathbf{z}_j = \mathbf{X}_{\cdot,j} - \mathbf{X}^{(-j)}\hat{\boldsymbol{\beta}}^{(-j)} \ , \tag{3.24}$$

where $\hat{\boldsymbol{\beta}}^{(-j)}$ refers to the estimator of the OLS regression of $\mathbf{X}_{\cdot,j}$ versus $\mathbf{X}^{(-j)}$. In particular, $\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} = 0$ for all $k \in [p] \setminus \{j\}$. Additionally, we have the following result:

Proposition 3.6.1. *If* $n > p$ *and* $\operatorname{rank}(\mathbf{X}) = p$*, then, for all* $j \in [p]$*:*

$$\hat{\boldsymbol{\beta}}_j^{\mathrm{OLS}} = \frac{\mathbf{z}_j^\top \mathbf{y}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} \ , \tag{3.25}$$

where $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}}$ *is the parameter vector estimates obtained from the* OLS regression of $\mathbf{y}$ against $\mathbf{X}$.

3.6.2 Desparsified Lasso

Now, we consider the high dimensional setting $n < p$. In this setting, it is not possible to construct a non-zero vector family $\{\mathbf{z}_j, j \in [p]\}$ (*i.e.*, a family verifying $\mathbf{z}_j \neq 0$ for all $j \in [p]$), such that $\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} = 0$ for all $k \neq j$. The idea proposed by Zhang and Zhang (2014) is to construct a family of vectors $\{\mathbf{z}_j, j \in [p]\}$, called score vectors, which would play the same role as the residual of the OLS regression of $\mathbf{X}_{\cdot,j}$ versus $\mathbf{X}^{(-j)}$ in (3.24) but relaxing (slightly) the constraint $\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} = 0$. To do so, instead of computing $\{\mathbf{z}_j, j \in [p]\}$ by OLS regression, they proposed to take the residual of the Lasso regressions[1] of $\mathbf{X}_{\cdot,j}$ against $\mathbf{X}^{(-j)}$. Then, from (2.7), one can derive the following:

$$\frac{\mathbf{z}_j^\top \mathbf{y}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} = \boldsymbol{\beta}_j^* + \frac{\mathbf{z}_j^\top \boldsymbol{\varepsilon}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} + \sum_{k \neq j} \frac{\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} \boldsymbol{\beta}_k^*}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} \ . \tag{3.26}$$

1 From our analysis, taking λ_j, the regularization parameter used in the Lasso regression of $\mathbf{X}_{\cdot,j}$ against $\mathbf{X}^{(-j)}$, equal to $0.01 \times \max_{k \in [p] \setminus \{j\}} |\mathbf{X}_{\cdot,j}^\top \mathbf{X}_{\cdot,k}|/n$ is appropriate to compute $\mathbf{z}_j$. Empirically, it results in a more conservative solution than the one proposed by Zhang and Zhang (2014) but it avoids doing computationally expensive grid-search.

Then, noticing that the second term in the right-hand side of (3.26) is a noise term and plugging in $\hat{\beta}^{L(\lambda_0)}$ instead of β^* as done in (3.21), they propose the desparsified Lasso (d-Lasso) estimator denoted by $\hat{\beta}^{DL}$ and defined by:

$$\hat{\beta}_j^{DL} = \frac{z_j^\top y}{z_j^\top X_{\cdot,j}} - \sum_{k \neq j} \frac{z_j^\top X_{\cdot,k} \hat{\beta}_k^{L(\lambda_0)}}{z_j^\top X_{\cdot,j}} \quad . \tag{3.27}$$

Here, one can notice that (3.27) generalizes (3.25) to $n < p$. Then, from (3.26) and (3.27) one can derive:

$$\sigma_\varepsilon^{-1}(\hat{\beta}_j^{DL} - \beta_j^*) = \underbrace{\sigma_\varepsilon^{-1} \frac{z_j^\top \varepsilon}{z_j^\top X_{\cdot,j}}}_{\eta_j} + \underbrace{\sigma_\varepsilon^{-1} \sum_{k \neq j} \frac{z_j^\top X_{\cdot,k}}{z_j^\top X_{\cdot,j}} (\beta_k^* - \hat{\beta}_k^{L(\lambda_0)})}_{\mu_j} \quad . \tag{3.28}$$

This yields:

$$\sigma_\varepsilon^{-1}(\hat{\beta}^{DL} - \beta^*) = \eta + \mu, \qquad \eta \sim \mathcal{N}_p(0, \Omega) \quad , \tag{3.29}$$

where:

$$\Omega_{jk} = \frac{z_j^\top z_k}{(z_j^\top X_{\cdot,j})(z_k^\top X_{\cdot,k})} \quad . \tag{3.30}$$

Asymptomatically and under sparsity assumptions (see van de Geer et al. (2014)), one can neglect the last term μ and obtain:

$$\sigma_\varepsilon^{-1}(\Omega_{jj})^{-1/2}(\hat{\beta}_j^{DL} - \beta_j^*) \sim \mathcal{N}(0, 1) \quad . \tag{3.31}$$

From (3.31), one can compute the confidence intervals and p-values of the coefficients of the estimated weight map.

3.6.3 Debiased Lasso approach

In Sec. 3.6.2, we took the approach proposed by Zhang and Zhang (2014) to construct the d-Lasso estimator. However, Javanmard and Montanari (2014) take a different approach to define the d-Lasso estimator (calling it debiased Lasso):

$$\hat{\beta}^{DL} = \hat{\beta}^{L(\lambda_0)} + \frac{1}{n} M X^\top (y - X \hat{\beta}^{L(\lambda_0)}) \quad , \tag{3.32}$$

where M is an estimate of the inverse of the covariance matrix $\hat{\Sigma}$. Then, the covariance of $\hat{\beta}^{DL}$ is given by:

$$\mathrm{Cov}\,(\hat{\beta}^{DL}) = \frac{1}{n^2} \sigma^2 M X^\top X M^\top \quad . \tag{3.33}$$

Now, taking the approach of Zhang and Zhang (2014), from (3.27) we can derive:

$$\hat{\boldsymbol{\beta}}^{\mathrm{DL}} = \mathbf{A}\,\mathbf{y} - \mathbf{P}\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)} \ , \tag{3.34}$$

where $\mathbf{A} \in \mathbb{R}^{p\times n}$ and $\mathbf{P} \in \mathbb{R}^{p\times p}$ are obtained by identification. Then, the covariance of $\hat{\boldsymbol{\beta}}^{\mathrm{DL}}$ is given by:

$$\mathrm{Cov}\,(\hat{\boldsymbol{\beta}}^{\mathrm{DL}}) = \sigma^2\mathbf{A}\mathbf{A}^\top \ . \tag{3.35}$$

Then, assuming that $\mathbf{M}$ verifies $\mathrm{diag}\,(\mathbf{M}\hat{\boldsymbol{\Sigma}}) = 1$, the two approaches are equivalent if:

$$\mathbf{A} = \frac{1}{n}\mathbf{M}\mathbf{X}^\top \ , \tag{3.36}$$

$$\mathbf{P} = \frac{1}{n}\mathbf{M}\mathbf{X}^\top\mathbf{X} - \mathbf{I} \ . \tag{3.37}$$

3.6.4 Degrees of freedom adjustment

d-Lasso being still actively developed, some recent works have proposed an additional degree-of-freedom adjustment (Bellec and Zhang, 2019; Celentano, Montanari, and Wei, 2020). Indeed, they have shown that if $s(\boldsymbol{\beta}^*)$ is not negligible in front of n, then it is necessary to adjust (3.32) as follows:

$$\hat{\boldsymbol{\beta}}^{\mathrm{DL}} = \hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)} + \frac{1}{n - s(\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)})}\mathbf{M}\mathbf{X}^\top(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)}) \ , \tag{3.38}$$

where $s(\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)}) = |S(\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)})|$. In the simulations proposed in Sec. 3.7, we test both original and degrees of freedom adjusted d-Lasso.

3.6.5 Noise estimation

In practice, the noise standard deviation σ_ε must be estimated to derive confidence intervals with the corrected Ridge or the d-Lasso. We use the method proposed by Reid, Tibshirani, and Friedman (2016) that we refer to as Reid procedure. Denoting the estimator by $\hat{\sigma}_\varepsilon$, they have proposed:

$$\hat{\sigma}_\varepsilon = \frac{\left\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)}\right\|_2^2}{n - s(\hat{\boldsymbol{\beta}}^{\mathrm{L}(\lambda_0)})} \ . \tag{3.39}$$

For studies dedicated to the estimation of the noise standard deviation in high dimensional linear model, one can refer to Ndiaye et al. (2017), Reid, Tibshirani, and Friedman (2016), and Yu and Bien (2019).

Algorithm 1: d-Lasso

input : $\mathbf{X} \in \mathbb{R}^{n \times p}, \mathbf{y} \in \mathbb{R}^n, \alpha \in (0,1)$

$q_{1-\frac{\alpha}{2}} \leftarrow$ Standard_Gaussian_Quantile$(1-\frac{\alpha}{2})$ // quantile

$\hat{\boldsymbol{\beta}}^{L(\lambda_0)} \leftarrow \text{Lasso}(\mathbf{X}, \mathbf{y})$ // Lasso estimator

$s(\hat{\boldsymbol{\beta}}^{L(\lambda_0)}) \leftarrow |S(\hat{\boldsymbol{\beta}}^{L(\lambda_0)})|$ // Lasso support

$\hat{\sigma}_\varepsilon \leftarrow \text{Reid}(\mathbf{X}, \mathbf{y})$ // noise estimator

for $j \in [p]$ **do**

$\mathbf{z}_j \leftarrow \text{Lasso}(\mathbf{X}^{(-j)}, \mathbf{X}_{\cdot,j})$ // score vectors

$\hat{\boldsymbol{\Omega}}_{j,j} \leftarrow \frac{n \mathbf{z}_j^\top \mathbf{z}_j}{|\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}||\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}|}$

$\hat{\boldsymbol{\beta}}_j^{(DL)} \leftarrow \frac{\mathbf{z}_j^\top \mathbf{y}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} - \sum_{k \neq j} \frac{\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} \hat{\boldsymbol{\beta}}_k^{L(\lambda_0)}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}}$ // d-Lasso estimator

$\hat{\boldsymbol{\beta}}_j^{(DL),\text{lower}} \leftarrow \hat{\boldsymbol{\beta}}_j^{(DL)} - q_{1-\frac{\alpha}{2}} n^{-1/2} \hat{\sigma}_\varepsilon \hat{\boldsymbol{\Omega}}_{j,j}^{1/2}$ // lower bound

$\hat{\boldsymbol{\beta}}_j^{(DL),\text{upper}} \leftarrow \hat{\boldsymbol{\beta}}_j^{(DL)} + q_{1-\frac{\alpha}{2}} n^{-1/2} \hat{\sigma}_\varepsilon \hat{\boldsymbol{\Omega}}_{j,j}^{1/2}$ // upper bound

end

return $\hat{\boldsymbol{\beta}}^{(DL)}, \hat{\boldsymbol{\beta}}^{(DL),lower}, \hat{\boldsymbol{\beta}}^{(DL),upper}$

3.6.6 Algorithm

The original d-Lasso algorithm given in Algo. 1 computes the d-Lasso estimator and its associated confidence intervals at $1-\alpha$ for $\alpha \in (0,1)$, *e.g.*, for $\alpha = 0.05$, we obtain the 95% confidence intervals.

3.7 EMPIRICAL COMPARISON

In this section, we benchmark the presented methods in various settings that correspond to compressed versions of the original neuroimaging setting.

3.7.1 Experimental setting

The simulations we propose here are built similarly as the one described in Sec. 3.1.4: covariates are Gaussian with Toeplitz covariance matrix. We also shuffle the covariates to reproduce a compressed version of the original problem into 500 synthetic features. In our experiments, we take $n \in \{100, 200, 400\}$, $p = 500$, $s(\boldsymbol{\beta}^*) \in \{5, 15, 50\}$, $\sigma_\varepsilon = 2$ and $\rho = 0.9$. Also, similarly as in Sec. 3.1.4, the true parameter vector $\boldsymbol{\beta}^*$ is defined by $\boldsymbol{\beta}_j^* = 1$ for $1 \leqslant j \leqslant s(\boldsymbol{\beta}^*)$ and $\boldsymbol{\beta}_j^* = 0$ otherwise.

We organize the simulations in three types of settings: a setting with small support where $s(\boldsymbol{\beta}^*) = 0.01p = 5$, a setting with medium support where $s(\boldsymbol{\beta}^*) = 0.03p = 15$ and a setting with large support where $s(\boldsymbol{\beta}^*) = 0.1p =$

50. For each type of setting, we vary the number samples. Then, for the 9 settings, we run 100 simulations drawing the covariate with a different seed to obtain meaningful results.

The SNR is the same for each type of settings: we have $\mathrm{SNR}_y = 1.5$ ($= 2\,\mathrm{dB}$) in the small support settings, $\mathrm{SNR}_y = 5$ ($= 7\,\mathrm{dB}$) in the medium support settings, $\mathrm{SNR}_y = 28$ ($= 14\,\mathrm{dB}$) in the large support settings. However, since $\sigma_\varepsilon = 2$ is fixed, the noise regime does not vary and the inference should become harder when the support size increases. This might be counter intuitive since the SNR_y also increases with the support size.

3.7.2 Results

In Fig. 3.3, we give the results obtained for the first seed in the central scenario with a medium support size. It is difficult to get definitive conclusion with only one seed, and all the methods seem to be approximately equally powerful in this setting. However, we can notice two main things: the multi-sample split produces p-values only for few coefficients due to its screening/inference strategy and the degrees of freedom adjusted d-Lasso is very close to the original d-Lasso.

In Fig. 3.4, we show that all the methods control the false positive rate (FPR), *i.e.*, the ratio between the number of false positives and the total number of actual negative, at the expected 5% nominal rate in all the settings, except the residual bootstrap Lasso when $s(\beta^*) = 50$ and $n = 100$. In Fig. 3.5, Fig. 3.6 and Fig. 3.7, we give the true positive rates for the small support size, the medium support size and the large support size. The true positive rate (or recall) is the number of discoveries divided by the size of the true support. In terms of power, for a small support size, we can notice that the residual bootstrap Lasso is the less powerful and the multi-sample split is the most powerful, the other methods remain competitive except the corrected Ridge when $n = 400$. For medium support size, original d-Lasso exhibits slightly better results, corrected Ridge is also powerful except when $n = 400$, multi-sample split is powerful except when $n = 100$, the residual bootstrap is less powerful and the degrees of freedom adjusted d-Lasso ($\mathrm{DL}_{\mathrm{ajd}}$) exhibits results close to the one of original d-Lasso but slightly worse for $n = 100$. For a large support size, the residual bootstrap Lasso is the most powerful when $n = 100$, however the FPR equals 14% in this case which is above the expected 5% nominal rate (see Fig. 3.4). Concerning the other methods, the original d-Lasso exhibits good recovery properties, the multi-sample split is not powerful and the other methods remain competitive except the corrected ridge when $n = 400$.

We can retain several main facts from these experiments. The residual bootstrap Lasso method is powerful when the support size is large, otherwise it is significantly less powerful than the other methods. The multi-sample split

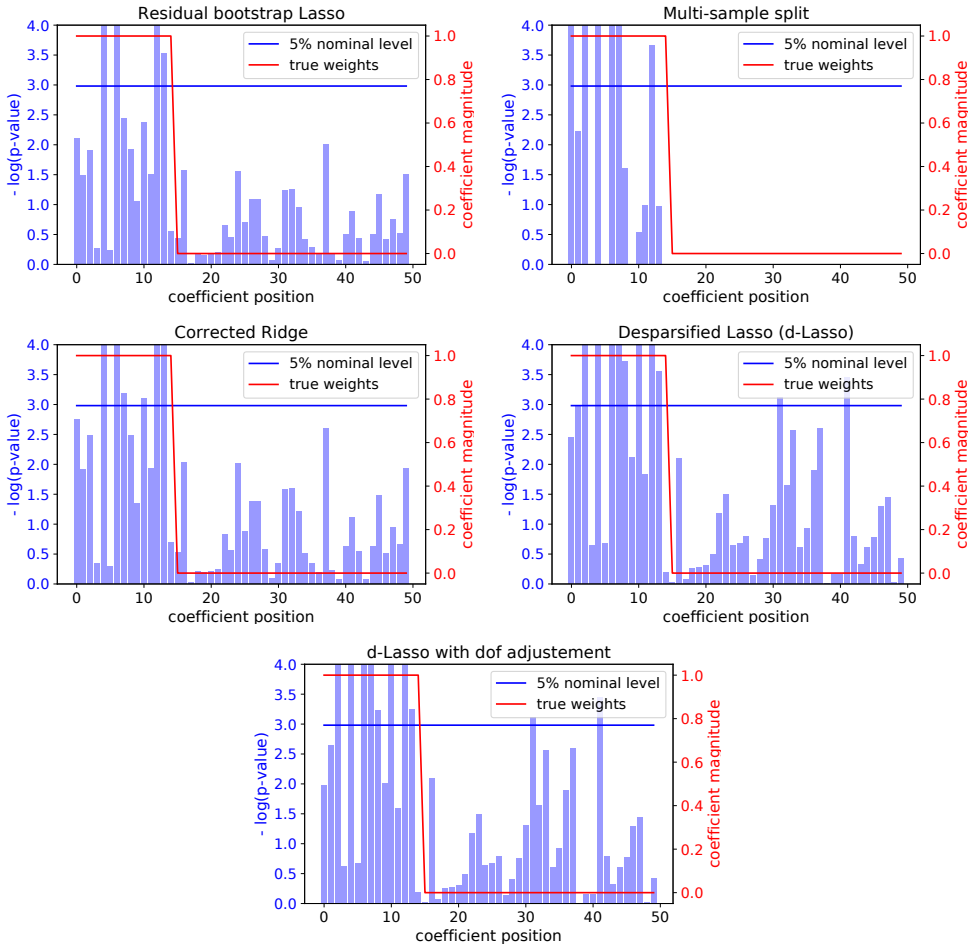


Figure 3.3: Qualitative results for the medium support size setting. We give here the p-values obtained by the different methods for the first seed in the medium support size scenario with $n = 200$. When a negative logarithm p-value is greater than 3, it corresponds to a p-value lower than 5% and the covariate is retained in the estimated support. It is difficult to get definitive conclusion with only one seed, and all the methods seem to be approximately equally powerful in this setting. These results show that the multi-sample split produces p-values only for few coefficients due to its screening/inference strategy and that the degrees of freedom adjusted d-Lasso is very close to the original d-Lasso.

procedure is competitive when $s(\beta^*) \ll n$ otherwise the method becomes significantly less powerful, this can be explained by the screening/inference strategy of the method. The corrected Ridge behave similarly as the d-Lasso, excepted for large sample size for which it is less powerful than the other methods. This effect is hard to understand but we noticed that the ridge estimator, which is the main component of the corrected ridge, does not give a better estimate of β^* when n goes from 200 to 400. Degrees of freedom adjusted d-Lasso does not improve over the original d-Lasso giving very

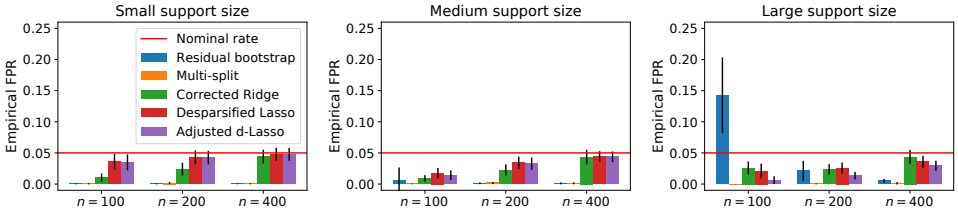


Figure 3.4: False positive rate control. Here we plot the empirical False Positive Rate (FPR) for all methods and all scenarios given a nominal level of 5%. Except for the residual bootstrap Lasso, for a large support size and a small sample size, the empirical FPR stays below the expected 5% nominal rate meaning that the occurrence of false positives is controlled accurately.

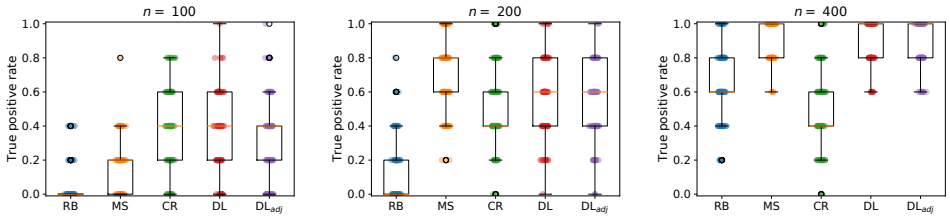


Figure 3.5: Power in small support size scenarios. Here we give the true positive rates for a support size of 1% of p. Note that since $s(\beta^*) = 5$, the true positive rate takes value in $\{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$. We can notice that the residual bootstrap Lasso (RB) is the less powerful, the multi-sample split (MS) is the most powerful when $n \geqslant 200$ and the original d-Lasso (DL) is the most powerful for $n = 100$. The corrected Ridge (CR), original d-Lasso and degrees of freedom adjusted d-Lasso (DL_{ajd}) are almost as powerful as multi-sample split when $n \geqslant 200$ except the corrected Ridge which fails to improve when $n = 400$. Note that in this scenario the empirical FPR stays below the expected 5% nominal rate for all the methods (see Fig. 3.4).

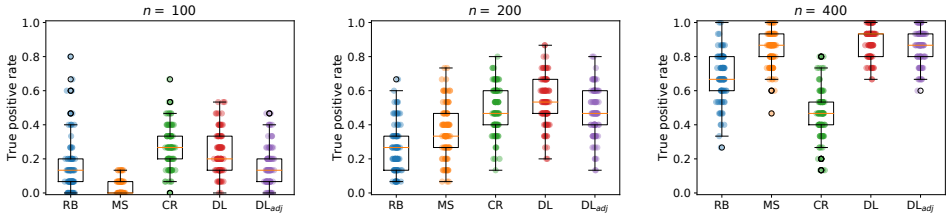


Figure 3.6: Power in medium support size scenarios. Here we give the true positive rates for a support size of 3% of p. The original d-Lasso (DL) exhibits slightly better results, corrected Ridge (CR) is also powerful except when $n = 400$, multi-sample split (MS) is powerful except when $n = 100$ and the residual bootstrap Lasso (RB) is less powerful. The degrees of freedom adjusted d-Lasso (DL_{ajd}) exhibits results close to the one of original d-Lasso but slightly worse for $n = 100$. Note that in this scenario the empirical FPR stays below the expected 5% nominal rate for all the methods (see Fig. 3.4).

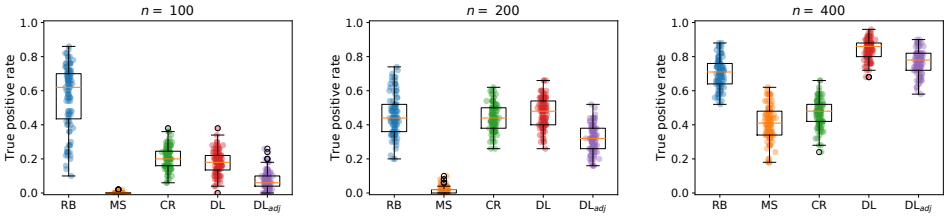


Figure 3.7: Power in large support size scenarios. Here we give the true positive rates for a support size of 10% of p. The residual bootstrap Lasso (RB) is the most powerful when $n = 100$, however the FPR equals 14% in this case which is above the expected 5% nominal rate (see Fig. 3.4). Concerning the other methods, original d-Lasso (DL) exhibits good recovery properties, corrected Ridge (CR) is also powerful except when $n = 400$, the multi-sample split (MS) is not powerful and degrees of freedom adjusted d-Lasso (DL_{ajd}) exhibits results close to the one of original d-Lasso but slightly worse. Note that, except for the residual bootstrap Lasso all the methods control the FPR at the expected level.

close results. Overall, d-Lasso is competitive in terms of power in every setting and controls the FPR as expected.

3.8 CONCLUSION

In this chapter, we have seen that several procedures can be used to address the statistical inference problem in high dimension. Among all the methods that we have presented and benchmarked, the d-Lasso offers several advantages. First, it is still actively developed and the scientific community seems to acknowledge its good behavior theoretically and empirically. Second, in our experiments, d-Lasso controlled the false positives as expected and exhibited competitive recovery properties with respect to the other methods in every setting. Then during the thesis we have decided to leverage the d-Lasso procedure.

Part II

MAIN CONTRIBUTIONS

4 ENSEMBLE OF CLUSTERED DESPARSIFIED LASSO

In this chapter, we introduce two algorithms for high-dimensional multivariate statistical inference on structured data. This chapter mainly revisit our first publication at the 2018 *MICCAI* conference:

CHEVALIER, Jérôme-Alexis, SALMON, Joseph, et THIRION, Bertrand. Statistical inference with ensemble of clustered desparsified lasso. In : International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, 2018. p. 638-646.

The algorithms we present, that we called cd-Lasso and ecd-Lasso, are notably well suited for high dimensional structured data such as neuroimaging data. We focus on explaining the rationale behind each step of the proposed algorithms along with experimental results to illustrate the potential of this approach.

4.1 INTRODUCTION

We recall that the model that describes the neuroimaging problem we are dealing with is given by (2.7):

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon} \ ,$$

where the response vector is denoted by $\mathbf{y} \in \mathbb{R}^n$, the design matrix by $\mathbf{X} \in \mathbb{R}^{n \times p}$, the parameter vector by $\boldsymbol{\beta}^* \in \mathbb{R}^p$ and the random error vector by $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma_\varepsilon^2 \mathbf{I}_n)$ where $\sigma_\varepsilon > 0$ is its unknown amplitude. In neuroimaging contexts, n the number of samples is the number of brain images available and p the number of covariates represents the number of voxels in each scan —a covariate being given by the level of activation of a voxel or some other imaging contrast. Our aim is to infer the weight map $\boldsymbol{\beta}^*$ that links the activation maps $\mathbf{X}$ to the conditions $\mathbf{y}$ with statistical guarantees on the proposed estimator. Ideally, we would like to recover all the covariates that are predictive with a control on the non-predictive variables selected.

For the current chapter, we only assume sample independence, sparsity and spatial structure of the weight map; these assumptions are discussed more in detail when we study the properties of the proposed estimators (see Chapter 5).

4.2 DIMENSION REDUCTION

Here, we show that, in the high-dimensional setting in which $n \ll p$, statistical inference is impossible without dimension reduction. We propose a way to compress the feature set that preserves data structure and keeps the problem as close as possible to the original one. Finally, we describe the clustered desparsified Lasso (cd-Lasso) algorithm for high-dimensional statistical inference.

4.2.1 Spatially-Constrained Clustering to Preserve Data Structure

In high dimension, several methods have been proposed to obtain confidence intervals or p-values (see Chapter 3). In particular, the d-Lasso estimator have recently been coined by several authors Javanmard and Montanari, 2014; Zhang and Zhang, 2014; van de Geer et al., 2014 and is one of the most promising methods to solve our problem (see Chapter 3). However, when $p \gg n$, we see in Fig. 4.1 that the method dramatically lacks power. Even more problematic is the fact that the number of predictive parameters (support size) denoted $s(\beta^*)$ is often greater than the number of samples even if we consider the sparse setting in which $s(\beta^*) \ll p$. This leads to an identifiability problem, hence one cannot retrieve the true parameter without further assumptions. Note that the impossibility of performing the statistical inference in the original setting was further discussed in Chapter 3.

Additionally, in high-dimensional inference, variables are often highly correlated. Specifically, a brain image has a 3D representation and a given voxel is highly correlated with its neighbors; β^* is very likely to carry the same structure. Then, in order to accurately compress the data and preserve the spatial structure, we want to avoid mixing voxels "far" from each other. A computationally attractive solution to alleviate high dimensionality, leveraging data structure, is to group adjacent voxels, producing a closely related, yet compressed version of the original problem. In decoding, the grouping of voxels via spatially-constrained clustering algorithms has already been used to reduce the problem dimension (Gramfort, Varoquaux, and Thirion, 2012; Varoquaux, Gramfort, and Thirion, 2012; Wang et al., 2015) in the prediction context. It is worth noting that this idea has also been successful in other domains, such as in genomics (Dehman, Ambroise, and Neuvial, 2015). Here, we consider a data-driven and spatially-constrained hierarchical clustering algorithm that uses Ward criterion following the conclusions by Varoquaux, Gramfort, and Thirion (2012) and Thirion et al. (2014). Specifically, groups of contiguous voxels can be replaced by the average signal they carry, reducing the dimensionality while improving the conditioning of the estimation problem. Also, an interesting aspect of this dimension reduction

method is its denoising property (Hoyos-Idrobo et al., 2018) since it averages signal from groups of noisy voxels.

The clustering solution highly depends on the choice of the number of groups (or clusters) which is denoted by C. Taking a small C leads to a more aggressive compression of the data and larger bias. However, the dimension of the problem is significantly reduced and the conditioning greatly improved, which leads to a much easier statistical inference problem. On the opposite, taking C large may not be sufficient to solve the issues encountered in the original uncompressed problem (see also Chapter 3).

The combination of the clustering algorithm and d-Lasso inference procedure, *i.e.*, the application of the d-Lasso onto the clustered problem, is referred to as the cd-Lasso algorithm.

4.2.2 Clustered Desparsified Lasso

Here, we present in Algo. 2 the cd-Lasso algorithm that produces p-values on the parameters of the model (2.7). In this algorithm, the function `Ward` corresponds to the clustering algorithm that takes in inputs the data $\mathbf{X}$ and a number of clusters and outputs a transformation matrix $\mathbf{A}$ to go from the original feature space to the compressed feature space. Then, the function `d-Lasso` corresponds to the d-Lasso inference that takes in inputs the clustered data $\mathbf{Z}$ and the target $\mathbf{y}$.

Algorithm 2: cd-Lasso algorithm

input : $\mathbf{X} \in \mathbb{R}^{n \times p}, \mathbf{y} \in \mathbb{R}^n$

param : C

$\mathbf{A} = \texttt{Ward}(C, \mathbf{X})$ `// transformation matrix`

$\mathbf{Z} = \mathbf{X}\mathbf{A}$ `// compressed design matrix`

$\hat{\mathbf{p}} = \texttt{d-Lasso}(\mathbf{Z}, \mathbf{y})$ `// uncorrected cluster-wise p-values`

$\hat{\mathbf{q}} = \min(1, C \times \hat{\mathbf{p}})$ `// corrected cluster-wise p-values`

for $j = 1, \ldots, p$ **do**

  $\hat{\mathbf{p}}_j = \hat{\mathbf{p}}^{(c)}$ if j in cluster c `// uncorr. feature-wise p-values`

  $\hat{\mathbf{q}}_j = \hat{\mathbf{q}}^{(c)}$ if j in cluster c `// corrected feature-wise p-values`

end

return $\hat{\mathbf{p}}_j, \hat{\mathbf{q}}_j$ for $j \in [p]$

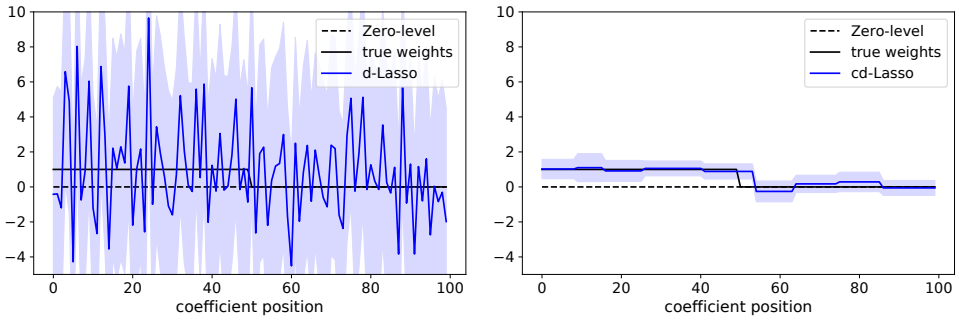
One can run the cd-Lasso algorithm on standard desktop stations — without using parallelization— with $n = 400$, $C = 500$ and $B = 25$ in less than 1 minute. In the cd-Lasso algorithm, the most expensive step is the d-Lasso inference, then $p \approx 10^5$ has a very limited impact on the computation time. The complexity for solving the Lasso depends on the

solver we choose, we then give the complexity in numbers of Lasso. The complexity for solving cd-Lasso is given by the complexity of the resolution of $\mathcal{O}(C)$ Lasso problems with n samples and C features. Note that, the complexity for solving the d-Lasso on the original problem is given by the complexity of the resolution of $\mathcal{O}(p)$ Lasso problems with n samples and p features. Then, as in Ganjgahi et al. (2018), the dimension reduction is not only allowing an increase of power but it is an essential feature for computational feasibility.

4.2.3 A 1D High Dimensional Simulation

Here, we introduce a 1D simulation to show the effect of compressing by spatially constrained clustering.

SIMULATION. Contrarily to the simulations of Chapter 3, this simulation has a 1D structure and we set $n = 100$ and $p = 2\,000$. We construct the design matrix $\mathbf{X}$ such that covariates are normally distributed and two consecutive covariates have a fixed correlation $\rho = 0.95$. The parameter vector $\boldsymbol{\beta}^*$ is such that $\boldsymbol{\beta}^*_j = 1$ for $1 \leqslant j \leqslant 50$ and $\boldsymbol{\beta}^*_j = 0$ otherwise, then $s(\boldsymbol{\beta}^*) = 50$. We also set $\sigma_\varepsilon = 10$ giving approximately $\mathrm{SNR}_y = 9$ ($= 10\,\mathrm{dB}$) where the SNR is defined by $\mathrm{SNR}_y = \|\mathbf{X}\boldsymbol{\beta}^*\|_2^2 / \|\boldsymbol{\epsilon}\|_2^2$ and describes the noise regime in any given experiment; this value is close to the estimated SNR in real MRI datasets. To compute the results given by the cd-Lasso, we took 200 clusters, reducing the dimension from $p = 2\,000$ to $C = 200$ before performing the inference.



(a) (b)

Figure 4.1: (a) 95% coefficient intervals given by the raw d-Lasso fail to retrieve the true support. (b) 95% coefficient intervals given by the cd-Lasso are much narrower, and yield a good support recovery.

RESULTS. In Fig. 4.1, we compare the results of the raw d-Lasso procedure with the one of the cd-Lasso algorithm. In Fig. 4.1-(a), we notice that the

raw d-Lasso fails to retrieve the true support since the confidence intervals are too wide. In the neuroimaging setting n is of the order of a hundred and p is around 10^5; here the d-Lasso is failing even with much lower p. In Fig. 4.1-(b), we can see that the cd-Lasso improves a lot over d-Lasso since it retrieves all the non-zero parameters with a limited number of false discoveries. Indeed, thanks to the clustering, the estimator variance is reduced. Then, cd-Lasso yields useful confidence intervals that could not be reached by standard d-Lasso. The impact of the choice of the hyperparameter C is further discussed in Chapter 6.

4.3 CLUSTERING RANDOMIZATION AND ENSEMBLING

In this section, we propose to randomize with regard to the clustering choice and aggregate several solutions. Then, the beneficial aspects of this randomization/aggregation step is then exhibited through some experimental results. Finally, we give another algorithm for high-dimensional statistical inference.

4.3.1 Randomization

The compression proposed in Sec. 4.2 introduces a bias, as the patterns are constrained by the clusters shape. It can be observed in Fig. 4.1-(b); for example, some non-zero coefficients are mixed some zero coefficients. This bias is problematic as there is no unique grouping (or clustering) of the voxels (Thirion et al., 2014): many different choices of clustering capture the signal accurately. Additionally, it is preferable not to rely on a particular clustering as small perturbations of the input data have a dramatic impact on the final solution. The approach presented in Varoquaux, Gramfort, and Thirion (2012) argues in favor of the randomization over the clustering step: to build B clusterings of the covariates, they use the same clustering method but with B different random subsamples of size $\lfloor 0.7n \rfloor$ from the full sample. The subsampling is only used for computing the clusters; but these grouping choices can be applied to the full data sample and statistical inference can be performed on each of the B compressed versions of the problem. Each compression reduces the problem into C clusters and running the statistical inference yields a p-value for each cluster. For all $b \in [B]$ and all $c \in [C]$, we denote by $\hat{\mathbf{p}}^{(b,c)}$ the p-value for the c-th cluster in the b-th fold. The p-value of the j-th voxel in the b-th repetition is denoted by $\hat{\mathbf{p}}_j^{(b)}$ and is taken equal to $\hat{\mathbf{p}}^{(b,c)}$ whenever j belongs to cluster c, *i.e.*, we attribute the same p-value to all the covariates in a given cluster. This yields B p-values for every coefficient of the parameter vector. In Fig. 4.5, we show that depending

on the random subsample, the choice of clustering —and consequently the solution given by the cd-Lasso— varies significantly, suggesting intrinsic instability of the cd-Lasso solution.

4.3.2 Aggregation

The benefits of model aggregation are well-known (Breiman, 1996; Zhou, 2012). In neuroimaging, in the prediction context, Varoquaux, Gramfort, and Thirion (2012) have shown empirically the beneficial aspect of randomization followed by aggregation; notably, it improves the prediction accuracy.

Similarly, random subspace methods (Ho, 1998; Kuncheva and Rodríguez, 2010; Kuncheva et al., 2010) also improve the prediction accuracy with more stable solutions —but in this case, subsampling is performed on the raw features. However, such a subsampling would make aggregation of statistical maps less straight forward and the combination with spatial clustering is also less obvious. The popular dropout method in deep learning actually borrows from the same idea Aydöre, Thirion, and Varoquaux, 2019.

More recently, another procedure, proposed by (Hoyos-Idrobo et al., 2018) and called Fast Regularized Ensembles of Models (FReM), has combined clustering and ensembling to reduce the variance of the weight map, while ensuring high prediction accuracy. Yet, FReM weight maps do not enjoy statistical guarantees.

Following these ideas, to mitigate the clustering bias, gain in stability and improve the overall solution, we are willing to aggregate the B statistical maps into one. However, in the estimation context, aggregating statistical maps requires more involved procedures than just averaging or taking the median of the solutions since we need to preserve the statistical properties on the p-value maps. The ensembling procedure we have considered is extracted from Meinshausen, Meier, and Bühlmann (2009). It consists in looking at a particular quantile of the p-value sets of size B for each voxel. The quantile aggregation procedure (Meinshausen, Meier, and Bühlmann, 2009) that yields the p-value $\hat{\mathbf{p}}_j$ of the j^{th} voxel, for any $j \in [p]$, is given by the following formula:

$$\hat{\mathbf{p}}_j = \min \left\{ 1, \gamma\text{-quantile} \left(\left\{ \frac{\hat{\mathbf{p}}_j^{(b)}}{\gamma} : b \in [B] \right\} \right) \right\} \quad , \tag{4.1}$$

where $\gamma \in (0, 1)$. A classic choice is to take $\gamma = 0.5$ which gives $\hat{\mathbf{p}}_j$ equals to twice the median of the set $\{\hat{\mathbf{p}}_j^{(b)} : b \in [B]\}$.

The combination of the randomization and the ensembling over the cd-Lasso algorithm is referred to as ecd-Lasso. To summarize, ecd-Lasso (Chevalier, Salmon, and Thirion, 2018) relies on three steps: a spatially

constrained clustering algorithm for reducing data dimension, a statistical inference procedure for deriving statistical maps, and an ensembling method for aggregating the statistical maps. A diagram summarizing ecd-Lasso is given in Fig. 4.2.

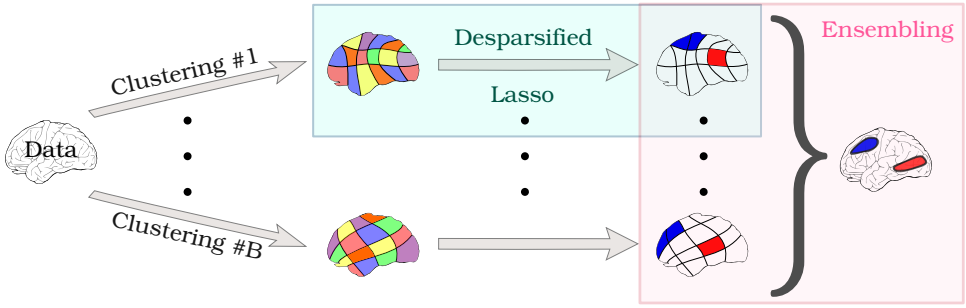


Figure 4.2: ecd-Lasso combines three algorithmic steps: a spatially constrained clustering procedure applied to images, the d-Lasso procedure to derive statistical maps, and an ensembling method that synthesizes several statistical maps.

Note that ecd-Lasso follows a scheme similar to FReM but the inference and ensembling procedures are different since they aim at producing p-value maps with statistical properties. In that regard, the aim of Chapter 5 is to treat the theoretical aspects and to show the statistical properties of cd-Lasso and ecd-Lasso.

4.3.3 Ensemble of Clustered Desparsified Lasso Algorithm

Now, we give in Algo. 3 the ecd-Lasso algorithm that produces p-values on the parameters of the model (2.7). In Algo. 3, the function `sample` corresponds to a subsampling of the data without replacement. Similarly as in Algo. 2, the function `Ward` derives the choice of clustering. But this time, the choice of clustering characterized by a transformation matrix $\mathbf{A}^{(b)}$ varies since the subsampled data $\mathbf{X}^{(b)}$ varies for each bootstrap $b \in [B]$. The function `d-Lasso` corresponds to the d-Lasso inference that takes in inputs the clustered data $\mathbf{Z}^{(b)}$ which varies for each $b \in [B]$ and the target $\mathbf{y}$. Once the clustering/inference steps are completed, the function `aggregation` makes the aggregation as presented in (4.1).

Computationally, to derive the ecd-Lasso solution we must solve B independent cd-Lasso problems, making the global problem embarrassingly parallel; nevertheless, we could run the ecd-Lasso algorithm on standard desktop stations —without using parallelization— with $n = 400$, $C = 500$ and $B = 25$ in less than 10 minutes. The complexity for solving ecd-Lasso is given by the complexity of the resolution of $\mathcal{O}(BC)$ Lasso problems with n samples and C features. Note that the clustering step being much quicker

Algorithm 3: ecd-Lasso algorithm

input : $\mathbf{X} \in \mathbb{R}^{n \times p}, \mathbf{y} \in \mathbb{R}^n$

param : C, B

for $b = 1, \ldots, B$ **do**

  $\mathbf{X}^{(b)} = \texttt{sample}(\mathbf{X})$ // sampling rows of $\mathbf{X}$

  $\mathbf{A}^{(b)} = \texttt{Ward}(C, \mathbf{X}^{(b)})$ // transformation matrix

  $\mathbf{Z}^{(b)} = \mathbf{X}\mathbf{A}^{(b)}$ // compressed design matrix

  $\hat{\mathbf{p}}^{(b)} = \texttt{d-Lasso}(\mathbf{Z}^{(b)}, \mathbf{y})$ // uncorrected cluster-wise p-values

  $\hat{\mathbf{q}}^{(b)} = \min(1, C \times \hat{\mathbf{p}}^{(b)})$ // corrected cluster-wise p-values

  for $j = 1, \ldots, p$ **do**

    $\hat{\mathbf{p}}_j^{(b)} = \hat{\mathbf{p}}^{(b,c)}$ if j in cluster c // uncorr. feature-wise p-values

    $\hat{\mathbf{q}}_j^{(b)} = \hat{\mathbf{q}}^{(b,c)}$ if j in cluster c // corrected feature-wise p-values

  end

end

for $j = 1, \ldots, p$ **do**

  $\hat{\mathbf{p}}_j = \texttt{aggregation}(\hat{\mathbf{p}}_j^{(b)}, b \in [B])$ // agg. uncorr. feature-wise p-val

  $\hat{\mathbf{q}}_j = \texttt{aggregation}(\hat{\mathbf{q}}_j^{(b)}, b \in [B])$ // agg. corr. feature-wise p-val

end

return $\hat{\mathbf{p}}_j, \hat{\mathbf{q}}_j$ for $j \in [p]$

than the inference step, p has a very limited impact on the total computation time; typically $p \approx 10^5$.

4.3.4 A 3D Simulation with Realistic Dimension

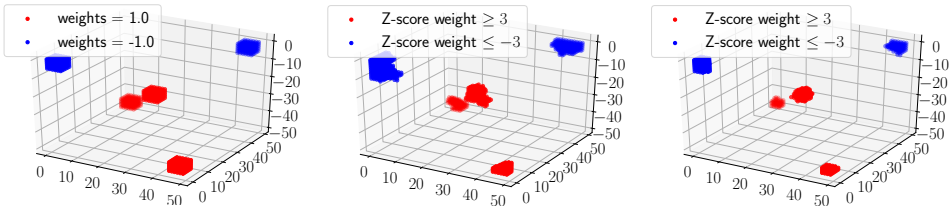
In this simulation we exhibit the improvements of ecd-Lasso over cd-Lasso in terms of recovery properties and control of FWER —the probability of making at least one false discovery.

SIMULATION. The simulation we consider has a 3D structure and tries to reproduce the setting of a standard MRI experiment. The feature space considered is a 3D cube with edge length $H = 50$, then $p = H^3 = 125k$ covariates (voxels) and we took $n = 400$ samples. To construct $\boldsymbol{\beta}^*$, we define a 3D weight vector $\tilde{\boldsymbol{\beta}}^*$ with five Region of Interests (ROI) represented in Fig. 4.3-(a) and then flatten $\tilde{\boldsymbol{\beta}}^*$ in a vector $\boldsymbol{\beta}^*$ of size p. Each ROI is a cube of width $h = 6$, leading to a size of support $s(\boldsymbol{\beta}^*) = 5h^3 = 1\,080$. Four ROIs are situated in corners of the cubic map and the last ROI is situated in the center of the cube. Finally, to construct $\mathbf{X}$, we first construct a 3D

design matrix $\tilde{\mathbf{X}}$ by drawing p random normal vectors of size n that are spatially smoothed with a 3D Gaussian filter (the smoothing is only made in the feature space for each sample independently, the samples are not mixed and remain independent), then we perform the same transformation (flattening) to go from $\tilde{\mathbf{X}}$ to $\mathbf{X}$ the $n \times p$ design matrix. The intensity of the spatial smoothing is designed to achieve similar feature correlation as for the Oasis experiment. We also set $\sigma_\varepsilon = 8$, to approximately get $\mathrm{SNR}_y = 9$ ($= 10\,\mathrm{dB}$).

RESULTS. To derive the ecd-Lasso solutions we aggregated $B = 25$ different cd-Lasso solutions during the ensembling step.

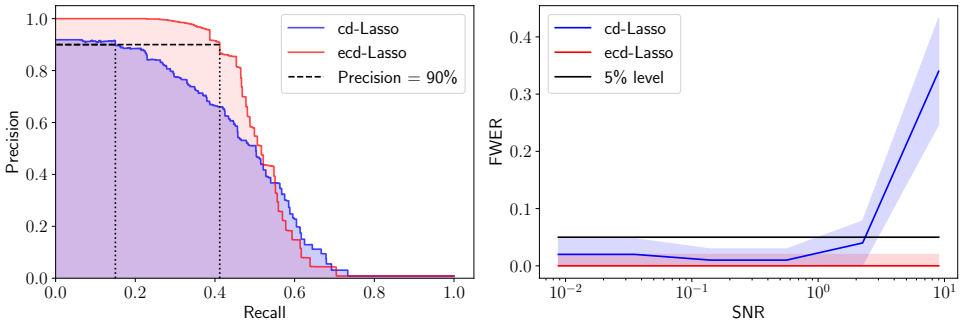
In Fig. 4.3, we give qualitative results for one simulation, we observe that the shape of the ecd-Lasso solution is more accurate than the one of cd-Lasso.



(a) 3D weight vector: $\tilde{\beta}^*$ (b) cd-Lasso (c) ecd-Lasso

Figure 4.3: For the 3D simulation proposed, the shape of the ecd-Lasso solution is more accurate than the one of cd-Lasso.

To confirm this qualitative observation, we run 100 simulations, and draw quantitative results in Fig. 4.4. In Fig. 4.4-(a), we display the precision-



(a) (b)

Figure 4.4: (a) The precision-recall curve for the recovery of β^* is much better adding an ensembling step over cd-Lasso. (b) The empirical-FWER (for a nominal rate at 5%) is controlled by the ecd-Lasso algorithm while for high level of SNR it is not controlled by the cd-Lasso algorithm.

recall curve[1] —the recall being the rate of recovery and the precision being the rate of true discoveries over the total number of discoveries— of the solutions obtained by each algorithm with $C = 500$ clusters. ecd-Lasso strongly outperforms cd-Lasso: for precision of at least 90%, the ecd-Lasso recall is 42% while the cd-Lasso recall is only 16%.

In order to check the FWER control, we define a neutral region that separates ROIs from the non-active region. Indeed, since the covariates are highly correlated, the detection of a null feature in the vicinity of an active one is not problematic from an application point of view. This spatial tolerance is made more rigorous in Chapter 5 and further justified in Chapter 6. Here, neutral regions enfold ROIs with a margin of 5 voxels. In Fig. 4.4-(b), the study of the FWER control is run for several values of SNR_y. One can observe that the FWER is always controlled using ecd-Lasso; the later is even conservative since the empirical FWER stays at 0% for a 5% nominal level. On the opposite, the FWER is not well controlled by cd-Lasso: its empirical value goes far above the 5% nominal rate for high SNR. This is due to the shape of the discovered regions that do not always correspond to the exact shape and location of ROIs. This effect is also observable considering thresholded z-score maps yielded by cd-Lasso and ecd-Lasso in Fig. 4.3. By increasing the number of clusters, we would obtain fewer discovered regions outside of the true ROIs, yet the statistical significance of the discovered regions would drop and the power would collapse.

4.3.5 Experiments on MRI Datasets

In this section, working on two real MRI datasets, we show the gain of stability produced by ecd-Lasso over cd-Lasso.

HAXBY DATASET. Haxby is a functional MRI dataset that maps the brain responses of subjects watching images of different objects (see Haxby et al., 2001). In this study, we only consider the responses related to images of faces and houses for the first subject, to identify brain regions that discriminate between these two stimuli, assuming that this problem can be modeled as a regression problem. Here $n = 200$, $p = 24k$, we estimated $\mathrm{SNR}_y = 1.0$ ($= 0\,\mathrm{dB}$) and we took $C = 500$ and $B = 25$.

OASIS DATASET. The Oasis MRI dataset (see Marcus et al., 2007) provides anatomical brain images of several subjects together with their age. The Statistical Parametric Mapping (SPM) voxel-based morphometry pipeline (Ashburner and Friston, 2000) was used to obtain individual gray matter

1 *cf.* Scikit-learn `precision_recall_curve` function

density maps. We aim at identifying which regions are informative to predict the age of a given subject. Here $n = 400$, $p = 125k$ and we estimated $SNR_y = 9.0$ ($= 10\,dB$); we also took $C = 500$ and $B = 25$ as in Sec. 4.3.4.

RESULTS. First, following established practice, we plot the results of these experiments in Fig. 4.5 displaying z-transform of the p-values. For clarity, we thresholded the z-score maps at 3 (and -3) keeping only the regions that have a high probability of being discriminative. The solutions given by the cd-Lasso algorithm with three different choices of clustering look noisy and unstable while the ecd-Lasso solution defines a synthesis of the cd-Lasso results and exhibits a nice symmetry in the case of Haxby. Thus, these results clearly illustrate that the ensembling step removes a significant part of the arbitrariness due to the clustering.

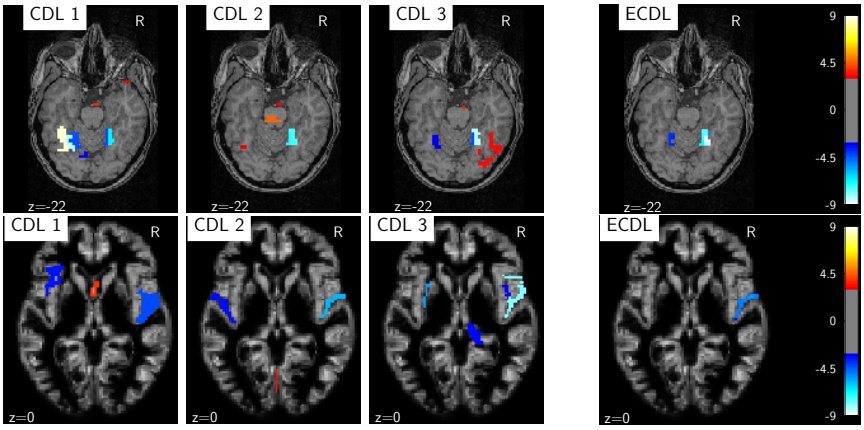


Figure 4.5: Results of the cd-Lasso (CDL) and ecd-Lasso (ECDL) algorithms on Haxby (top) and Oasis (bottom) experiments. cd-Lasso algorithm outcomes are highly dependent on the clustering, which creates a jitter in the solution. Drawing consensus among many cd-Lasso results, ecd-Lasso removes the arbitrariness related to the clustering scheme.

Again, to confirm this qualitative observation, we run 25 times the cd-Lasso and ecd-Lasso algorithms seeding differently the clustering, and draw quantitative results in Fig. 4.6 looking at the correlation and the Jaccard index of the solutions. Correlation between the full maps and Jaccard index of the detected areas (here, voxels with an absolute z-score greater than 3) show that ecd-Lasso is substantially more stable than cd-Lasso.

4.4 DISCUSSION

RECAPITULATION. We have introduced ecd-Lasso, an algorithm for high-dimensional multivariate inference on structured data which scales even

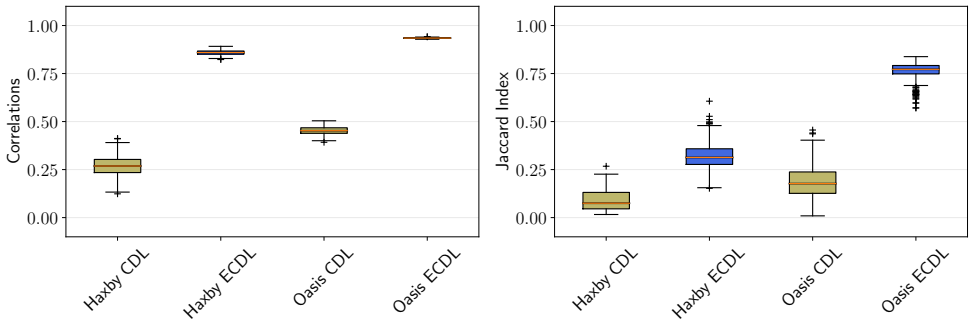


Figure 4.6: Correlation (left) and Jaccard index (right) are much higher with the ecd-Lasso (ECDL) algorithm than with cd-Lasso (CDL) across 25 replications of the analysis of the imaging datasets.

when the number of covariates $p \geqslant 10^5$ is much higher than the number of samples $n \leqslant 10^3$. It can be summarized as follows: i) perform B repetitions of the cd-Lasso algorithm, that runs d-Lasso inference on a compressed version of the problem obtained by spatial clustering, yielding several p-values for each predictor; ii) use an ensemble method aggregating all p-value maps to derive a single p-value map. In Sec. 4.2.3, we have shown that the clustering step, justified by specific data structures and locally high inter-predictor correlation, was necessary to yield an informative inference solution when $n \ll p$. Then, we have shown in Sec. 4.3.4, that randomizing and ensembling the cd-Lasso solutions improves both FWER control and precision-recall values. While the ensembling step obviously removes the arbitrariness of the clustering choice, in Sec. 4.3.5, we showed that it also increases stability. Also, the ensembling step helps to improve shape accuracy without loss in sensitivity, as the combination of multiple cd-Lasso solutions recovers finer spatial information.

OPEN QUESTIONS. The number of clusters C is the main free parameter, and an optimal value depends on characteristics of the data (inter-predictor correlation, SNR). It seems credible that the optimal choice for C results from a bias/variance trade-off: a small number of clusters reduces variance and enhances statistical power, while a greater number yields refined solutions. This central question around the optimal choice for C is further discussed in Chapter 6.

Another matter is the comparison with bootstrap and permutation-based approaches *e.g.*, Gaonkar and Davatzikos, 2012. This is also studied in Chapter 6.

A more theoretical aspect concerns the statistical guarantees on the p-values produced by ecd-Lasso. This question is addressed in Chapter 5.

5 STATISTICAL INFERENCE WITH SPATIAL TOLERANCE

In this chapter, we state the statistical guarantees provided by cd-Lasso and ecd-Lasso. We show that ecd-Lasso controls a generalization of the FWER called δ-FWER, that takes into account a spatial tolerance of radius δ for the occurrence of false discoveries. This result holds under realistic assumptions, for a predetermined spatial tolerance parameter δ.

5.1 INTRODUCTION

HIGH-DIMENSIONAL SETTING. High-dimensional settings correspond to the one where the number of covariates (or features) p exceeds the number of samples n. This type of setting occurs in many domains nowadays, typically to discover associations among some observations and outcomes of interest (target). Typical examples concern inference problems on high-resolution images, where one aims at pixel- or voxel-level analysis, *e.g.*, in neuroimaging (Button et al., 2013), astronomy (Richards et al., 2009), but also in other fields where features encompass a spatial structure *e.g.*, in genomics (Balding, 2006).

MULTIVARIATE MODEL. When fitting a target, one may want to assess whether each feature adds information to what is conveyed by the other features. In other words, one might try to recover features that are predictive conditionally to the others. When $n \ll p$, the most popular approach is to consider the multivariate linear model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon} \ ,$$

where the response vector is denoted by $\mathbf{y} \in \mathbb{R}^n$, the random Gaussian design matrix by $\mathbf{X} \in \mathbb{R}^{n \times p}$, the parameter vector by $\boldsymbol{\beta}^* \in \mathbb{R}^p$ and the random error vector by $\boldsymbol{\varepsilon} \in \mathbb{R}^n$. The aim is to infer $\boldsymbol{\beta}^*$, with statistical guarantees on the support estimate. Ideally, we would like recover all the covariates that are conditionally predictive without selecting non-predictive variables.

STATISTICAL INFERENCE ON INDIVIDUAL PARAMETERS. In the high-dimensional setting, the classic statistical approach does not apply but numerous methods have recently been proposed to recover the non-zero

parameters of β^* with statistical guarantees. Some methods rely on resampling: bootstrap procedures (Bach, 2008; Chatterjee and Lahiri, 2011; Liu, Yu, et al., 2013), perturbation resampling-based procedures (Minnier, Tian, and Cai, 2011), stability selection procedures (Meinshausen and Bühlmann, 2010) and randomized screening/inference procedures (Meinshausen, Meier, and Bühlmann, 2009; Wasserman and Roeder, 2009). Contrarily to the screening/inference procedure, post-selection inference procedures generally merge the screening and inference steps into one and then use all the samples in a screening/inference solution (Berk et al., 2013; Lee et al., 2016; Lockhart et al., 2014; Tibshirani et al., 2016). Another family of methods rely on debiasing procedures: the most prominent examples are corrected Ridge (Bühlmann, 2013) and desparsified Lasso (Javanmard and Montanari, 2014; Zhang and Zhang, 2014; van de Geer et al., 2014) that is still actively developed (Bellec and Zhang, 2019; Celentano, Montanari, and Wei, 2020; Javanmard, Montanari, et al., 2018). Additionally, the Knockoff filters (Barber and Candès, 2015; Candès et al., 2018) consist in creating mimicking variables checking whether original variables are selected at random or not. Finally, a general framework for statistical inference for sparse high-dimensional models has been proposed recently (Ning, Liu, et al., 2017).

FAILURE OF EXISTING STATISTICAL INFERENCE METHODS. In the high-dimensional setting that we are targeting, none of these methods are helpful for recovering the support. In particular, the number of predictive parameters (support) denoted $s(\beta^*)$ is often greater than the number of samples even in the sparse setting, where $s(\beta^*) \ll p$. This leads to an identifiability problem: in general, one cannot retrieve all the predictive parameters. Some studies (Wainwright, 2009) have highlighted the impossibility of recovering the support in such a setting. Beyond the fact that statistical inference is hard when $p \gg n$, two other reasons make it even more difficult. Especially in high-dimensional settings, feature correlation is high, challenging the conditions for recovery given in the above publications. Second, when testing for several multiple hypothesis, the correction cost is heavy (Benjamini and Hochberg, 1995; Dunn, 1961; Westfall and Young, 1993); for example with Bonferroni correction (Dunn, 1961), p-values are corrected by a factor of p when testing every feature. This may make this type of inference method powerless.

COMBINING CLUSTERING AND INFERENCE. In the setting we are given, variables often depict a strong spatial structure. For example, in neuroimaging, a brain image has a 3D representation and a given voxel is highly correlated with its neighbors; in genomics, there exists blocs of Single Nucleotide Polymorphisms (SNPs) that tend to occur together. Additionally, the true parameter vector generally displays the same structure. A computa-

tionally attractive solution to alleviate high dimensionality, leveraging data structure, is to group correlated neighboring features, producing a closely related, yet compressed version of the original problem.

Inference combined with a fixed clustering has notably been proposed by Bühlmann et al. (2013) and seems to be a promising technique to overcome the dimensionality issue, however this study does not provide procedures that derive cluster-wise confidence intervals or p-values. Inspired by this idea, we have proposed in Chevalier, Salmon, and Thirion (2018) the ensemble of clustered desparsified Lasso (ecd-Lasso) procedure that exhibits strong empirical performances in terms of support recovery even when $p \gg n$. The ecd-Lasso estimator combines three steps: *i)*a spatially constrained clustering procedure that reduces the problem dimension but preserves data structure; *ii)*the desparsified Lasso procedure that is tractable on clustered versions of the problem; and *iii)*an ensembling method that aggregates the solutions of different compressed versions of the problem to avoid relying on a single clustering choice. Concerning the high-dimensional statistical inference method, this uses the desparsified Lasso (Javanmard and Montanari, 2014; Zhang and Zhang, 2014; van de Geer et al., 2014) following the comparative study of Dezeure et al. (2015) and noting that this procedure is actively developed by several groups (Bellec and Zhang, 2019; Celentano, Montanari, and Wei, 2020; Dezeure, Bühlmann, and Zhang, 2017; Javanmard, Montanari, et al., 2018). By contrast, we do not further consider the popular Knockoff procedure (Barber and Candès, 2015; Candès et al., 2018), that appears to be powerful inference method, since it does not produce p-values and to control the family-wise error rate (FWER) but only the false discovery rate (FDR).

Additionally, Meinshausen (2015) provides "group bound" confidence intervals, corresponding to confidence intervals on the ℓ_1-norm of several parameters, without making any assumptions on the design matrix. However, this method is known to be conservative Javanmard, Montanari, et al. (2018) and Mitra and Zhang (2016) in practice.

Finally, hierarchical testing (Blanchard, Geman, et al., 2005; Mandozzi and Bühlmann, 2016; Meinshausen, 2008) also leverages this clustering/inference combination but in a different way. It consists in performing significance tests along the tree of a hierarchical clustering algorithm starting from the root node and descending subsequently into children of rejected nodes. This procedure has the drawback of being constrained by the clustering tree. The problem with a fixed clustering choice is that in practice there does not exist a "true" clustering of the data. Also, a small variation of the clustering choice may significantly change the solution, leading to instability in the results.

STATISTICAL INFERENCE WITH SPATIAL TOLERANCE. Looking for groups of covariates instead of covariates means that one is willing to accept a loss

of accuracy on the support estimate as long as this improves statistical power. Additionally, given the spatial structure of data, *i.e.*, 1/the spatial structure of the covariates —neighboring features are highly correlated— and 2/the assumed shape of the support of the true parameter vector —predictive features are spatially localized—, we argue that all false discoveries do not have the same severity: a false discovery made at a small distance from the support is more acceptable. Also, we advocate that producing a cluster-wise inference is not completely satisfactory as it assumes implicitly that there exists a true clustering choice. Then, to untangle the problem, we propose instead to introduce a spatial tolerance —distance being defined relatively to the feature space geometry— in false discovery control.

CONTRIBUTIONS. In this study, our first contribution is to introduce a generalization of the FWER called δ-FWER, that takes into account a spatial tolerance of length δ for the false discoveries. Consequently, in the high-dimensional setting we are given, we aim at controlling the δ-FWER which notably coincides with the usual FWER if $\delta = 0$. Then, our main contribution is to prove that ecd-Lasso controls the δ-FWER under reasonable assumptions for a predetermined tolerance parameter δ. Finally, we provide an empirical study, showing that ecd-Lasso exhibits good recovery properties and displays the expected δ-FWER control empirically in realistic simulations.

ORGANIZATION. In Sec. 5.2, we bring useful model and data structure assumptions that naturally arise in the proposed setting. In Sec. 5.3, we introduce the δ-FWER that generalizes the FWER by incorporating a spatial tolerance and is well suited for the recovery problem we are addressing. In Sec. 5.4, we bring all the material to prove that ecd-Lasso controls the δ-FWER under realistic assumptions for a predetermined tolerance parameter δ. In Sec. 5.5, we perform numerical simulations to validate the previous theoretical results and benchmark ecd-Lasso against two other procedures.

5.2 MODEL AND DATA ASSUMPTIONS

5.2.1 Generative models of high-dimensional data: random fields

In the settings that we consider, we assume that covariates (or features) come with a natural representation in a discretized metric space, generally the discretized 2D or 3D Euclidean space. In such settings, discrete random fields are convenient to model the random variables representing the covariates. Indeed, denoting by $\mathbf{X} = (\mathbf{X}_{i,j})_{i\in[n],j\in[p]}$ the random design matrix, where n is the number of samples and p the number of features, covariate samples

$(\mathbf{X}_{i,.})_{i\in[n]}$ can be modeled as random fields operating on a common discrete domain.

5.2.2 Gaussian random design model and high dimensional settings

We further assume that the covariates follow a centered Gaussian distribution, *i.e.*, for all $i \in [n]$, $\mathbf{X}_{i,.} \sim \mathcal{N}_p(0, \mathbf{\Sigma})$ where $\mathbf{\Sigma}$ is the covariance matrix of the covariates. Our aim is to derive confidence bounds or p-values on the coefficients of the parameter vector denoted by $\boldsymbol{\beta}^*$, under the Gaussian linear model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon} \ , \tag{5.1}$$

where $\mathbf{y} \in \mathbb{R}^n, \mathbf{X} \in \mathbb{R}^{n\times p}, \boldsymbol{\beta}^* \in \mathbb{R}^p$, $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma_\varepsilon^2 \mathbf{I}_n)$ and $\sigma_\varepsilon > 0$. $\mathbf{X}$ is the random design matrix, $\mathbf{y}$ is the response vector (or target), $\boldsymbol{\varepsilon}$ is the error vector and σ_ε is the noise standard deviation (generally unknown). For all $i \in [n]$, the $\mathbf{X}_{i,.}$'s are assumed independent and identically distributed. We also assume that $\boldsymbol{\varepsilon}$ is independent of $\mathbf{X}$.

5.2.3 Data structure

Let us first describe the covariates spatial structure. Since the covariates have a natural representation in a metric space, we assume the spatial distances between covariates to be known. With a slight abuse of notation, the distance between covariates j and k is denoted by $d(j,k)$ for $(j,k) \in [p]\times[p]$ and the correlation between covariates j and k is given by $\mathrm{Cor}(\mathbf{X}_{.,j}, \mathbf{X}_{.,k}) \stackrel{\text{def}}{=} \mathbf{\Sigma}_{j,k}/\sqrt{\mathbf{\Sigma}_{j,j}\mathbf{\Sigma}_{k,k}}$. We now introduce a key assumption: two covariates at a spatial distance smaller than δ are positively correlated. Formally, the covariates verify the *spatial homogeneity assumption* with parameter $\delta > 0$, if:

$$\forall (j,k) \in [p]^2, \quad d(j,k) \leqslant \delta \implies \mathbf{\Sigma}_{j,k} \geqslant 0 \ . \tag{AH_δ}$$

Now, we introduce an assumption to model the parameter spatial structure. Under model (5.1), each coordinate of the parameter vector $\boldsymbol{\beta}^*$ links one covariate to the response vector. Then, $\boldsymbol{\beta}^*$ has the same underlying organization as the covariates and is also called *weight map* in these settings. Defining the true *support* as $S(\boldsymbol{\beta}^*) = \{j \in [p] : \boldsymbol{\beta}^*_j \neq 0\}$ and its cardinal as $s(\boldsymbol{\beta}^*) = |S(\boldsymbol{\beta}^*)|$, we assume that the true model is sparse, meaning that $\boldsymbol{\beta}^*$ has a small number of non-zero entries, *i.e.*, $s(\boldsymbol{\beta}^*) \ll p$. The complementary of $S(\boldsymbol{\beta}^*)$ in $[p]$ is called the *null region* and is denoted by $N(\boldsymbol{\beta}^*)$, *i.e.*, $N(\boldsymbol{\beta}^*) = \{j \in [p] : \boldsymbol{\beta}^*_j = 0\}$. Additionally to the sparse assumption, we assume that $\boldsymbol{\beta}^*$ is (spatially) smooth. More precisely, to reflect sparsity and smoothness, we introduce AS_δ which states that weights associated to

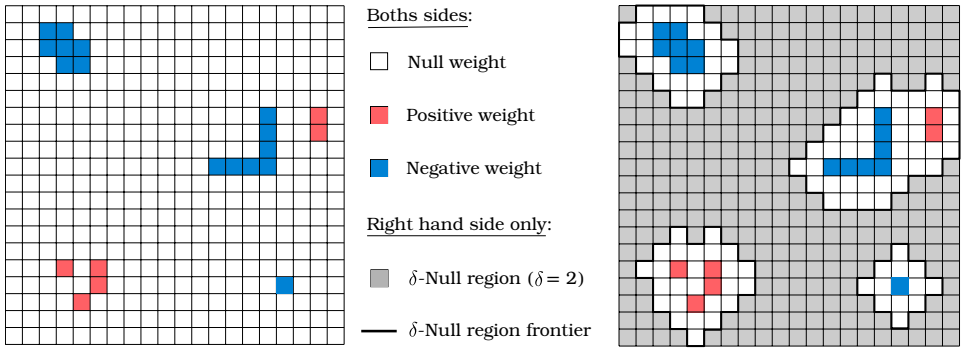


Figure 5.1: Left: Example of weight map having a 2D-spatial structure. Assuming that all pixels have a unit side length, this map notably verifies the sparse-smooth assumption defined in (AS_δ) with $\delta = 2$ with respect to the L1-distance. Right: For the weight map proposed on the left, the gray squares represent the δ-null region for $\delta = 2$ with respect to the L1-distance.

covariates with small spatial distances have the same sign (zero being booth positive and negative). Formally, we say that β^* verifies the *sparse-smooth assumption* with parameter $\delta > 0$ if:

$$\forall (j,k) \in [p]^2, \quad d(j,k) \leqslant \delta \implies \operatorname{sign}(\beta^*_j) = \operatorname{sign}(\beta^*_k) \ . \tag{AS_δ}$$

Equivalently, the sparse-smooth assumption with parameter δ holds if the distance between the two closest weights of opposite sign is larger than δ. In Fig. 5.1, we give an example of a weight map verifying the sparse-smooth assumption with $\delta = 2$.

5.3 STATISTICAL CONTROL WITH SPATIAL TOLERANCE

Discoveries that are not in the support but closer than δ to the support are not considered as false discoveries, because there is no point in making inference at a resolution finer than δ. This means δ can be interpreted as a tolerance parameter on the shape of the regions that we aim at recovering. In this section, we introduce a new metric closely related to the Family-Wise Error Rate (FWER) that takes into account spatial tolerance and we call it δ-Family Wise Error Rate (δ-FWER). To do so, we assume that we consider a general estimator $\hat{\beta}$ that comes with p-values denoted by $\hat{p} = (\hat{p}_j)_{j \in [p]}$, and we also denote by $S(\hat{\beta}) \subset [p]$ a general support estimate.

Definition 5.3.1 (δ-null hypothesis). *For all $j \in [p]$, the j-th δ-null hypothesis $H_0^\delta(j)$ states that every covariate at a distance less than δ from the j-th covariate*

has an associated weight equals to zero in the true model (5.1) and the alternative hypothesis is denoted $H_1^\delta(j)$*:*

$$\begin{aligned} H_0^\delta(j) &: \text{"}\forall k \in [p] \text{ s.t. } d(j,k) \leqslant \delta,\ \beta_k^* = 0\text{"} \ , \\ H_1^\delta(j) &: \text{"}\exists k \in [p] \text{ s.t. } d(j,k) \leqslant \delta \text{ and } \beta_k^* \neq 0\text{"} \ . \end{aligned} \tag{5.2}$$

Consequently, we say that a δ*-type 1 error* is made if a covariate $j \in [p]$ is selected, *i.e.*, $j \in S(\hat{\beta})$, while $H_0^\delta(j)$ holds true. Note that, taking $\delta = 0$ recovers the usual null-hypothesis $H_0(j) : \text{"}\beta_j^* = 0\text{"}$ and usual type 1 error.

Definition 5.3.2 (Control of the δ-type 1 error)**.** *The p-value related to the* j*-th covariate denoted by* $\hat{p}_j$ *controls the* δ*-type 1 error if, under* $H_0^\delta(j)$*, for all* $\alpha \in (0,1)$*, we have:*

$$\mathbb{P}(\hat{p}_j \leqslant \alpha) \leqslant \alpha \ , \tag{5.3}$$

where $\mathbb{P}$ *is the probability distribution* w.r.t. *the covariates and the error.*

Definition 5.3.3 (δ-null region)**.** *The set of indexes of covariates verifying the* δ*-null hypothesis is called the* δ*-null region and is denoted by* $N^\delta(\beta^*)$ *(or simply* N^δ*):*

$$N^\delta(\beta^*) = \{j \in [p] : \forall k \in [p], d(j,k) \leqslant \delta \implies \beta_k^* = 0\} \ . \tag{5.4}$$

When $\delta = 0$ the δ-null region is simply the null region and $N^0(\beta^*) = N(\beta^*)$. We also point out the nested property of δ-null regions *w.r.t.* δ: for $0 \leqslant \delta_1 \leqslant \delta_2$ we have $N^{\delta_2}(\beta^*) \subseteq N^{\delta_1}(\beta^*) \subseteq N(\beta^*)$. In Fig. 5.1, we draw the δ-null region, taking $\delta = 2$ and using the L1-distance (or Manhattan distance), for an arbitrary initial weight map.

Definition 5.3.4 (Rejection region)**.** *Given a family of p-values* $\hat{p} = (\hat{p}_j)_{j\in[p]}$ *and a threshold* $\alpha \in (0,1)$*, the rejection region is the set of indexes of covariates having a p-value lower than* α*, we denote it by* $R_\alpha(\hat{p})$*:*

$$R_\alpha(\hat{p}) = \left\{j \in [p] : \hat{p}_j \leqslant \alpha\right\} \ . \tag{5.5}$$

Definition 5.3.5 (δ-type 1 error region)**.** *Given a family of p-values* $\hat{p} = (\hat{p}_j)_{j\in[p]}$ *and a threshold* $\alpha \in (0,1)$*, the* δ*-type 1 error region at level* α *is* $\mathscr{E}_\alpha^\delta$*, the set of indexes of covariates belonging both to the* δ*-null region and to the rejection region at level* α*. We also refer to this region as the erroneous rejection region at level* α *with tolerance* δ*:*

$$\mathscr{E}_\alpha^\delta(\hat{p}) = N^\delta \cap R_\alpha(\hat{p}) \ . \tag{5.6}$$

When $\delta = 0$ the δ-type 1 error region is simply the type 1 error region which is denoted by $\mathscr{E}_\alpha(\hat{p})$. From Def. 5.3.3, Def. 5.3.4 and Def. 5.3.5, one can verify that for $0 \leqslant \delta_1 \leqslant \delta_2$ we have $\mathscr{E}_\alpha^{\delta_2}(\hat{p}) \subseteq \mathscr{E}_\alpha^{\delta_1}(\hat{p}) \subseteq \mathscr{E}_\alpha(\hat{p})$.

Definition 5.3.6 (δ-family wise error rate). *Given a family of p-values* $\hat{p} = (\hat{p}_j)_{j\in[p]}$ *and a threshold* $\alpha \in (0,1)$*, the* δ*-FWER at level* α w.r.t. *the family* $\hat{p}$*, denoted* $\delta\text{-FWER}_\alpha(\hat{p})$*, is the probability that the* δ*-type 1 error region at level* α *is not empty:*

$$\delta\text{-FWER}_\alpha(\hat{p}) = \mathbb{P}(|\mathscr{E}^\delta_\alpha(\hat{p})| \geqslant 1) = \mathbb{P}(\min_{j\in N^\delta} \hat{p}_j \leqslant \alpha) \ . \tag{5.7}$$

Definition 5.3.7 (δ-FWER control). *We say that the family of p-values* $\hat{p} = (\hat{p}_j)_{j\in[p]}$ *controls the* δ*-FWER if, for all* $\alpha \in (0,1)$*:*

$$\delta\text{-FWER}_\alpha(\hat{p}) \leqslant \alpha \ . \tag{5.8}$$

When $\delta = 0$ the δ-FWER is the usual FWER. Additionally, for $0 \leqslant \delta_1 \leqslant \delta_2$, one can verify that $\delta_2\text{-FWER}_\alpha(\hat{p}) \leqslant \delta_1\text{-FWER}_\alpha(\hat{p}) \leqslant \text{FWER}_\alpha(\hat{p})$. Thus, δ-FWER control is a weaker property than usual FWER control.

5.4 δ–FWER CONTROL WITH CLUSTERED DESPARSIFIED LASSO

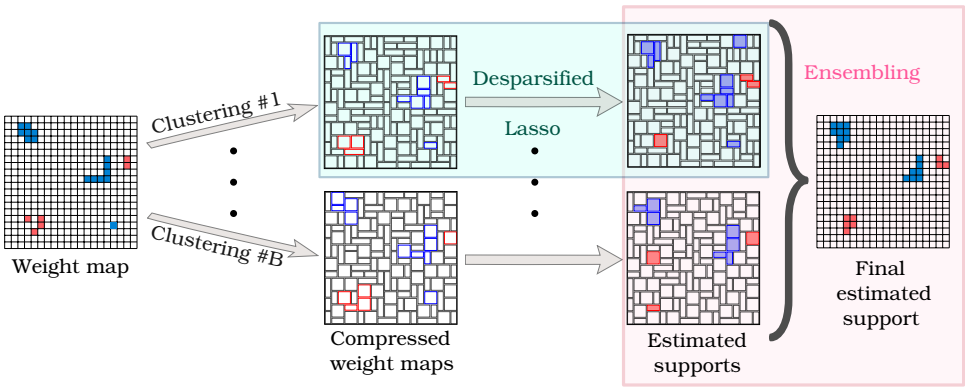


Figure 5.2: This diagram summarizes the ensemble of clustered desparsified Lasso (ecd-Lasso), which is a statistical inference procedure suited for high dimensional structured data. ecd-Lasso combines three algorithmic steps: a clustering procedure, the desparsified Lasso statistical inference procedure to derive p-value maps, and an ensembling method that synthesizes several p-value maps into one.

In this section, we focus on establishing the δ-FWER control property of the Clustered desparsified Lasso (cd-Lasso) and ensemble of clustered desparsified Lasso (ecd-Lasso) algorithms. The cd-Lasso algorithm, which was introduced in Chevalier, Salmon, and Thirion (2018), consists in partitioning the covariates into groups (or clusters) **with diameter smaller than** δ before applying the desparsified Lasso statistical inference procedure

introduced by Zhang and Zhang (2014). The ecd-Lasso algorithm, summarized in Fig. 5.2 and also introduced in Chevalier, Salmon, and Thirion (2018), is the ensembling of several cd-Lasso solutions —obtained by using different choice of clustering— using the p-value aggregation proposed by Meinshausen, Meier, and Bühlmann (2009). The detailed algorithm of the ecd-Lasso procedure is described in Algo. 3 in Chapter 4.

5.4.1 Compressed representation

The benefit of using groups of covariates that are spatially concentrated is to reduce the dimension while preserving the spatial structure of the data. The number of groups is denoted by $q < p$ and, for $r \in [q]$, we denote by G_r the r-th group. The collection of all the groups is denoted by $\mathcal{G} = \{G_1, G_2, \dots, G_q\}$ and forms a partition of $[p]$. Every group representative variable is defined by the average of the covariates it contains. Then, denoting by $\mathbf{Z} \in \mathbb{R}^{n \times q}$ the compressed random design matrix that contains the group representative variables in columns and, without loss of generality, enforcing the suitable ordering of the original covariates, *i.e.*, of the columns of $\mathbf{X}$, dimension reduction can be written :

$$\mathbf{Z} = \mathbf{X}\mathbf{A} \ , \tag{5.9}$$

where $\mathbf{A} \in \mathbb{R}^{p \times q}$ is the transformation matrix defined by:

$$\mathbf{A} = \begin{bmatrix} \alpha_1 \text{—} \alpha_1 & 0 \text{—} 0 & \dots & 0 \text{—} 0 \\ 0 \text{—} 0 & \alpha_2 \text{—} \alpha_2 & \dots & 0 \text{—} 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 \text{—} 0 & 0 \text{—} 0 & \dots & \alpha_q \text{—} \alpha_q \end{bmatrix} \ ,$$

where $\alpha_r = 1/|G_r|$ for all $r \in [q]$. Consequently, the distribution of the i-th row of $\mathbf{Z}$ is given by $\mathbf{Z}_{i,\cdot} \sim \mathcal{N}_q(0, \mathbf{T})$, where $\mathbf{T} = \mathbf{A}^\top \mathbf{\Sigma} \mathbf{A}$. The correlation between the groups $r \in [q]$ and $l \in [q]$ is given by $\mathrm{Cor}(\mathbf{Z}_{\cdot,r}, \mathbf{Z}_{\cdot,l}) \stackrel{\text{def}}{=} \mathbf{T}_{r,l} / \sqrt{\mathbf{T}_{r,r}\mathbf{T}_{l,l}}$. As mentioned in Bühlmann et al. (2013), due to the Gaussian assumption in (5.1), we have the following compressed representation:

$$\mathbf{y} = \mathbf{Z}\theta^* + \boldsymbol{\eta} \ , \tag{5.10}$$

where $\theta^* \in \mathbb{R}^q$, $\boldsymbol{\eta} \sim \mathcal{N}(0, \sigma_\eta^2 \mathbf{I}_n)$, $\sigma_\eta \geqslant \sigma_\varepsilon > 0$ and $\boldsymbol{\eta}$ is independent of $\mathbf{Z}$.

Remark 5.4.1. *Dimension reduction is not the only desirable effect produced by clustering with regards to statistical inference. Indeed, this compression also generally improves the conditioning of the problem and eases the correlation structure in the (compressed) feature space. Then, assumptions needed for valid statistical inference are more likely to be met. For, more details about this conditioning enhancement, the reader may refer to Bühlmann et al. (2013).*

5.4.2 Properties of the compressed model weights

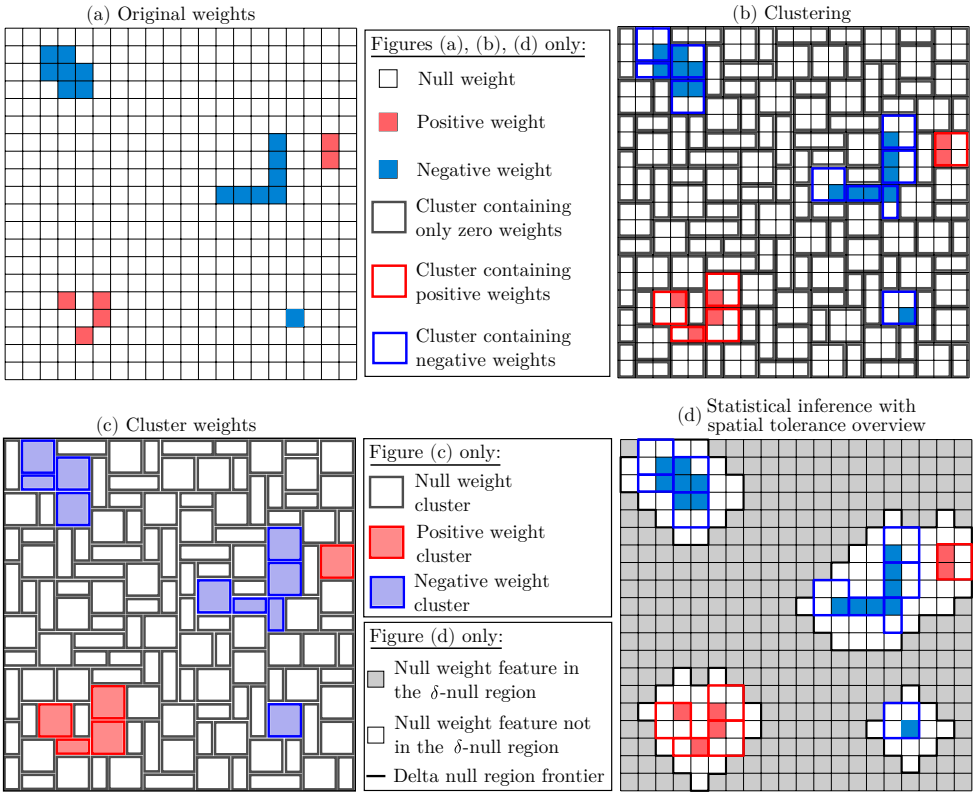


Figure 5.3: Item a: Representation of a weight map having a 2D-spatial structure. This is the same map as in Fig. 5.1. Item b: Arbitrary choice of spatially constrained clustering with a diameter of 2 units with respect to the L1-distance. Also this clustering choice verifies assumption (iii) of Prop. 5.4.1. Item c: Under the assumption of Prop. 5.4.1, the cluster weights have the same sign as the feature weights they contain. Item d: Under the assumption of Prop. 5.4.1, one can notice that all the non-zero weight groups have no intersection with the δ-null region for $\delta = 2$.

We now give a property of the weights of the compressed problem which is a consequence of Bühlmann et al. (2013, Proposition 4.4).

Proposition 5.4.1. *Considering the Gaussian linear model in (5.1) and assuming:*

$$\begin{aligned}
&\text{(i)}\ \forall r \in [q],\ \forall j,k \in G_r^2,\ \boldsymbol{\Sigma}_{j,k} \geqslant 0\ ,\\
&\text{(ii)}\ \forall r \in [q],\ \forall l \in [q]\setminus\{r\},\ \mathbf{T}_{r,l} = 0\ ,\\
&\text{(iii)}\ \forall r \in [q],\ \big(\forall j \in G_r, \boldsymbol{\beta}_j^* \geqslant 0\big) \text{ or } \big(\forall j \in G_r, \boldsymbol{\beta}_j^* \leqslant 0\big)\ ,
\end{aligned}$$

then, in the compressed representation (5.10), for $r \in [q]$, $\theta_r^* \neq 0$ *if and only if there exists* $j \in G_r$ *such that* $\boldsymbol{\beta}_j^* \neq 0$. *If such an index* j *exists then* $\operatorname{sign}(\theta_r^*) = \operatorname{sign}(\boldsymbol{\beta}_j^*)$.

Proof. With assumption (ii) we can use Bühlmann et al. (2013, Proposition 4.3). Then, with the assumptions (i) and (iii), we directly obtain the claimed property. □

Assumption (i) states that the covariates in a group are all positively correlated. Let us define the group diameter of G_r by the distance that separates its two most distant covariates, *i.e.*, $\mathrm{Diam}(G_r) \stackrel{\text{def}}{=} \max\{d(j,k) : (j,k) \in (G_r)^2\}$ and the clustering diameter of $\mathcal{G}$ by the largest diameter of its groups, *i.e.*, $\mathrm{Diam}(\mathcal{G}) \stackrel{\text{def}}{=} \max\{\mathrm{Diam}(G_r) : r \in [q]\}$. In Fig. 5.3-(b), we propose a choice of clustering of the initial weight map in Fig. 5.3-(a) for which the clustering diameter is equal to 2 with respect to the L1-distance. Then assumption (i) is notably established if the clustering diameter is smaller than δ and the spatial homogeneity assumption with parameter δ defined in (AH_δ) is verified. Assumption (ii) states that the groups are independent. A sufficient condition is to assume that the feature covariance matrix $\boldsymbol{\Sigma}$ is block diagonal, where the blocks coincide with the groups; *i.e.*, , assumption (ii) is verified if the features of two different groups are independent. In practice, this assumption may be unmet; then, in Sec. 5.4.6, we relax assumption (ii). Assumption (iii) states that all the weights in a group are of the same sign. This is notably the case when the clustering diameter is smaller than δ and the weight map satisfies the sparse-smooth assumption with parameter δ defined in (AS_δ). For instance, a clustering-based compressed representation of the weight map in Fig. 5.3-(a) is given in Fig. 5.3-(c).

5.4.3 Statistical inference on the compressed model

To perform statistical inference on the compressed problem (5.10), we consider the desparsified Lasso developed in Zhang and Zhang (2014) and in Javanmard and Montanari (2014) and thoroughly analyzed in van de Geer et al. (2014). With regard to (5.10), let us define the true support in the compressed model by $S(\theta^*) = \{r \in [q] : \theta^*_r \neq 0\}$ and its size is $s(\theta^*) = |S(\theta^*)|$. We also denote by $\boldsymbol{\Omega} \in \mathbb{R}^{q \times q}$ the inverse of the population covariance matrix (the precision matrix) of the groups, *i.e.*, $\boldsymbol{\Omega} = \mathbf{T}^{-1}$. Then, for $r \in [q]$, we denote the sparsity with respect to the r-th row of $\boldsymbol{\Omega}$ (or r-th column) by $s(\boldsymbol{\Omega}_{r,\cdot}) = |S(\boldsymbol{\Omega}_{r,\cdot})|$ where $S(\boldsymbol{\Omega}_{r,\cdot}) = \{l \in [q] : \boldsymbol{\Omega}_{r,l} \neq 0\}$. We also denote the smallest eigenvalue of $\mathbf{T}$ by $\phi_{\min} > 0$. We can now state the assumptions required for probabilistic inference with desparsified Lasso van de Geer et al. (2014):

Proposition 5.4.2 (Theorem 2.2 of van de Geer et al. (2014)). *Considering the compressed representation in (5.10) and assuming:*

$$
\begin{aligned}
&\text{(i)}\ 1/\phi_{\min} = \mathcal{O}(1)\ ,\\
&\text{(ii)}\ \max_{r\in[q]}(\mathbf{T}_{r,r}) = \mathcal{O}(1)\ ,\\
&\text{(iii)}\ s(\theta^*) = o(\sqrt{n}/\log(q))\ ,\\
&\text{(iv)}\ \max_{r\in[q]}(s(\mathbf{\Omega}_{r,\cdot})) = o(n/\log(q))\ ,
\end{aligned}
$$

then, denoting by $\hat{\theta}$ the desparsified Lasso estimator that can be derived from the inference procedure described in Zhang and Zhang (2014) and van de Geer et al. (2014), the following holds:

$$
\begin{aligned}
&\sqrt{n}(\hat{\theta} - \theta^*) = \mu + \tau\ ,\\
&\mu|\mathbf{Z} \sim \mathcal{N}_q(0, \sigma_\eta^2 \hat{\mathbf{\Omega}})\ ,\\
&\|\tau\|_\infty = o_{\mathbb{P}}(1)\ ,
\end{aligned}
$$

where $\hat{\mathbf{\Omega}}$ is an estimate of $\mathbf{\Omega}$ that verifies $\left\|\hat{\mathbf{\Omega}} - \mathbf{\Omega}\right\|_\infty = o_{\mathbb{P}}(1)$ under the previous assumptions.

Remark 5.4.2. *In Prop. 5.4.2, to compute confidence intervals, the noise standard deviation σ_η in the compressed problem must be estimated. We refer the reader to surveys (Ndiaye et al., 2017; Reid, Tibshirani, and Friedman, 2016; Yu and Bien, 2019) that are dedicated to this subject.*

As argued in van de Geer et al. (2014), from Prop. 5.4.2 we obtain asymptotic confidence intervals for the r-th element of θ^* from the following equations, for all $z_1 \in \mathbb{R}$ and $z_2 \in \mathbb{R}^+$:

$$
\begin{aligned}
&\mathbb{P}\left[\frac{\sqrt{n}(\hat{\theta}_r - \theta_r^*)}{\sigma_\eta\sqrt{\hat{\mathbf{\Omega}}_{rr}}} \leqslant z_1 \middle| \mathbf{Z}\right] - \Phi(z_1) = o_{\mathbb{P}}(1)\ ,\\
&\mathbb{P}\left[\frac{\sqrt{n}|\hat{\theta}_r - \theta_r^*|}{\sigma_\eta\sqrt{\hat{\mathbf{\Omega}}_{rr}}} \leqslant z_2 \middle| \mathbf{Z}\right] - (2\Phi(z_2) - 1) = o_{\mathbb{P}}(1)\ ,
\end{aligned}
\tag{5.11}
$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. Thus, for each $r \in [q]$ one can provide a p-value that assesses whether or not θ_r^* is equal to zero. In the case of a two-sided single test, for each $r \in [q]$, the p-value associated with this test is denoted by $\hat{p}_r^{\mathcal{G}}$ (since θ^* and $\hat{\theta}$ depends on the initial choice of clustering $\mathcal{G}$) and given by:

$$
\hat{p}_r^{\mathcal{G}} = 2\left(1 - \Phi\left(\frac{\sqrt{n}|\hat{\theta}_r|}{\sigma_\eta\sqrt{\hat{\mathbf{\Omega}}_{rr}}}\right)\right)\ . \tag{5.12}
$$

Under the above assumptions, the p-values $\hat{p}_r^{\mathcal{G}}$ control type 1 errors. Indeed, under $H_0(G_r)$ —the null hypothesis that states that θ_r^* is equal to zero in the true compressed model— and using (5.11), we have, for any $\alpha \in (0,1)$:

$$\begin{aligned}\mathbb{P}(\hat{p}_r^{\mathcal{G}} \leqslant \alpha | \mathbf{Z}) &= 1 - \mathbb{P}\left[\frac{\sqrt{n}|\hat{\theta}_r|}{\sigma_\eta \sqrt{\hat{\mathbf{\Omega}}_{rr}}} \leqslant \Phi^{-1}\left(1 - \frac{\alpha}{2}\right) \middle| \mathbf{Z}\right] \\ &= \alpha + o_{\mathbb{P}}(1) \ .\end{aligned} \tag{5.13}$$

To correct for multiple comparisons, we use the Bonferroni correction (Dunn, 1961) which is a conservative procedure but has the advantage of being valid without any additional assumptions. Note furthermore that the correction factor is only for the number of groups, not the number of voxels. We can thus ensure FWER control using the family of corrected p-values $\hat{q}^{\mathcal{G}} = (\hat{q}_r^{\mathcal{G}})_{r \in [q]}$ defined by:

$$\hat{q}_r^{\mathcal{G}} = \min\{1, q \times \hat{p}_r^{\mathcal{G}}\} \ . \tag{5.14}$$

Let us denote by $N_{\mathcal{G}}(\theta^*)$ (or simply $N_{\mathcal{G}}$) the null region in the compressed problem for a given choice of clustering $\mathcal{G}$, *i.e.*, $N_{\mathcal{G}}(\theta^*) = \{r \in [q] : \theta_r^* = 0\}$. Then, the family of corrected p-values $\hat{q}^{\mathcal{G}}$ defined in (5.14) verifies the following equation, for all $\alpha \in (0,1)$:

$$\text{FWER}_\alpha(\hat{q}^{\mathcal{G}}) = \mathbb{P}(\min_{r \in N_{\mathcal{G}}} \hat{q}_r^{\mathcal{G}} \leqslant \alpha | \mathbf{Z}) \leqslant \alpha + o_{\mathbb{P}}(1) \ . \tag{5.15}$$

In this section, we have shown that the desparsified Lasso applied to a compressed version of the original problem provides cluster-wise p-value families $\hat{p}^{\mathcal{G}}$ and $\hat{q}^{\mathcal{G}}$ that asymptotically control respectively the type 1 error and the FWER in the compressed model. In the following sections we will omit the term "conditionally to $\mathbf{Z}$" to ease the notation.

5.4.4 De-grouping

Given the families of cluster-wise p-values $\hat{p}^{\mathcal{G}}$ and corrected p-values $\hat{q}^{\mathcal{G}}$ as defined in (5.12) and (5.14), our next aim is to derive families of p-values and corrected p-values related to the covariates themselves. To construct these families, we simply set the (corrected) p-value of the j-th covariate to be equal to the (corrected) p-value of its corresponding group:

$$\begin{aligned}\forall j \in [p], \quad \hat{p}_j &= \sum_{r \in [q]} \mathbb{1}_{\{j \in G_r\}} \hat{p}_r^{\mathcal{G}} = \hat{p}_{g(j)}^{\mathcal{G}} \ , \\ \forall j \in [p], \quad \hat{q}_j &= \sum_{r \in [q]} \mathbb{1}_{\{j \in G_r\}} \hat{q}_r^{\mathcal{G}} = \hat{q}_{g(j)}^{\mathcal{G}} \ ,\end{aligned} \tag{5.16}$$

where the grouping function g matches the covariate index to its corresponding group index:

$$\begin{aligned} g : [p] &\to [q] \\ j &\mapsto r \text{ if } j \in G_r \ . \end{aligned}$$

Proposition 5.4.3. *Under the assumptions of Prop. 5.4.1 and Prop. 5.4.2 and assuming that the diameter of all clusters is smaller than δ, then:*

(i) elements of the family $\hat{p}$ defined as in (5.16) asymptotically control the δ-type 1 error:

$$\forall j \in N^{\delta},\ \forall \alpha \in (0,1),\ \mathbb{P}(\hat{p}_j \leqslant \alpha) \leqslant \alpha \ ,$$

(ii) the family $\hat{q}$ defined as in (5.16) asymptotically controls the δ-FWER:

$$\forall \alpha \in (0,1),\ \mathbb{P}(\min_{j \in N^{\delta}}(\hat{q}_j) \leqslant \alpha) \leqslant \alpha \ .$$

Proof. See Sec. 5.6.1 in Appendix for a formal proof. □

The previous de-grouping properties can be guessed from Fig. 5.3-(d).

5.4.5 Ensembling

Let us consider B families of corrected p-values that asymptotically control the δ-FWER, such as the corrected p-values provided by the cd-Lasso algorithm with B different choices of clustering. For any $b \in [B]$, we denote by $\hat{q}^{(b)}$ the b-th family of corrected p-values. Then, in this section, we show that the ensembling method proposed in Meinshausen, Meier, and Bühlmann (2009) yields a family that also enforces δ-FWER control.

Proposition 5.4.4. *Assume that we have B families $(\hat{q}_j^{(b)})_{j \in [p]}$, where $b \in [B]$ is the family index, that control the δ-FWER. For γ in $(0,1)$, let us define the family $(\tilde{q}_j(\gamma))_{j \in [p]}$ by the following equation:*

$$\forall j \in [p],\ \tilde{q}_j(\gamma) = \min\left\{1, \gamma\text{-quantile}\left(\left\{\frac{\hat{q}_j^{(b)}}{\gamma} : b \in [B]\right\}\right)\right\} \ . \tag{5.17}$$

Then $(\tilde{q}_j(\gamma))_{j \in [p]}$ also controls the δ-FWER.

Proof. See Sec. 5.6.2 in Appendix. □

With this last proposition, we have all the ingredients to state our main result: ecd-Lasso asymptotically controls the δ-FWER.

Proposition 5.4.5. *Assume the model given in (5.1) and that assumptions AH_δ and AS_δ are verified. Consider B clustering choices into q clusters such that the largest cluster diameter is always smaller than δ. Assume that for every choice of clustering the assumptions of Prop. 5.4.1 and Prop. 5.4.2 are verified. Then the p-value family constructed following the ensemble of clustered desparsified Lasso,* i.e., *using **1/** the inference on the compressed problem as presented in Sec. 5.4.1, Sec. 5.4.2 and Sec. 5.4.3, **2/** the de-grouping method proposed in Sec. 5.4.4 and **3/** the ensembling technique of Sec. 5.4.5, controls the δ-FWER asymptotically.*

Proof. Directly follows from Prop. 5.4.1, Prop. 5.4.2, Prop. 5.4.3 and Prop. 5.4.4. □

5.4.6 Relaxing the uncorrelated clusters assumption

In this section, we relax the assumption (ii) of Prop. 5.4.1 and show that we can compute an adjusted corrected p-value that asymptotically controls the δ-FWER. First, we replace Prop. 5.4.1 by the next proposition that is a consequence of Bühlmann et al. (2013, Proposition 4.4).

Proposition 5.4.6. *Considering the Gaussian linear model in (5.1) and assuming:*

$$
\begin{aligned}
&\text{(i)}\ \forall r \in [q],\ \forall j,k \in G_r^2,\ \mathrm{Cov}(\mathbf{X}_{\cdot,j},\mathbf{X}_{\cdot,k}|\{\mathbf{Z}_{\cdot,l} : l \neq r\}) \geqslant 0\ ,\\
&\text{(ii.a)}\ \forall r \in [q],\ \exists \nu_r \in \mathbb{R}^+,\ \forall j \in G_r,\ \forall k \notin G_r,\\
&\qquad |\mathrm{Cov}(\mathbf{X}_{\cdot,j},\mathbf{X}_{\cdot,k}|\{\mathbf{Z}_{\cdot,l} : l \neq r\})| \leqslant \nu_r\ ,\\
&\text{(ii.b)}\ \forall r \in [q],\ \exists C_r > 0, \mathrm{Var}(\mathbf{Z}_{\cdot,r}|\{\mathbf{Z}_{\cdot,l} : l \neq r\}) \geqslant C_r\ ,\\
&\text{(iii)}\ \forall r \in [q],\ \big(\forall j \in G_r, \boldsymbol{\beta}^*_j \geqslant 0\big) \text{ or } \big(\forall j \in G_r, \boldsymbol{\beta}^*_j \leqslant 0\big)\ ,
\end{aligned}
$$

then, in the compressed representation (5.10), θ^* admits the following decomposition:

$$\theta^* = \tilde{\theta} + \boldsymbol{\kappa}\ , \tag{5.18}$$

where, for $r \in [q]$, $|\boldsymbol{\kappa}_r| \leqslant (\nu_r \,/\, C_r)\|\boldsymbol{\beta}^*\|_1$ and $\tilde{\theta}_r \neq 0$ if and only if there exists $j \in G_r$ such that $\boldsymbol{\beta}^*_j \neq 0$. If such an index j exists then $\mathrm{sign}(\tilde{\theta}_r) = \mathrm{sign}(\boldsymbol{\beta}^*_j)$.

Proof. Use (ii.a) and (ii.b) to get the first statement of Bühlmann et al. (2013, Proposition 4.4). Then use (i) and (iii) to directly obtain the result. □

The assumptions (i) and (ii) in Prop. 5.4.1 are replaced by (i), (ii.a) and (ii.b) in Prop. 5.4.6. More precisely, instead of assuming that the covariates of a same group are positively correlated, we assume that they are positively correlated conditionally to all other groups. Also, we relax the more questionable assumption of groups independence; we assume instead that (ii.a) the conditional (same conditioning as in (i)) covariance of two covariates of different groups is bounded above and that (ii.b) the conditional variance

of the group representative variable is nonzero. In practice, except if groups are linearly dependent (very unlikely), we can always find values for which (ii.a) and (ii.b) are verified, but we would like the upper bound of (ii.a) as low as possible and the lower bound of (ii.b) as high as possible. Finally, assumption (iii) remains unchanged.

Then, similarly as done in Sec. 5.4.3, we can build $\hat{\theta}$. Under the same assumptions, Prop. 5.4.2 is still valid and $\hat{\theta}$ still verifies (5.11). However, here we want to estimate $\tilde{\theta}$, not θ^*. Combining Prop. 5.4.2 and Prop. 5.4.6, we can see $\hat{\theta}$ as a biased estimator of $\tilde{\theta}$. To take this bias into account, we need to adjust the definition of the p-values given by (5.12). Let us assume that:

$$\max_{r\in[q]}\left(\frac{\nu_r}{C_r\sqrt{\hat{\Omega}_{rr}}}\right) \leqslant \frac{a\,\sigma_\epsilon}{\|\beta^*\|_1}\ , \tag{5.19}$$

where $a \in \mathbb{R}^+$ is a constant that is discussed at the end of this section. Then, for all $r \in [q]$, the adjusted p-values are given by:

$$\hat{p}_r^{\mathcal{G}} = 2\left(1-\Phi\left(\sqrt{n}\left[\frac{|\hat{\theta}_r|}{\sigma_\eta\sqrt{\hat{\Omega}_{rr}}}-a\right]_+\right)\right)\ . \tag{5.20}$$

Let us denote by $q_{1-\frac{\alpha}{2}}$ the $1-\frac{\alpha}{2}$ quantile of the standard Gaussian distribution, *i.e.*, $q_{1-\frac{\alpha}{2}} = \Phi^{-1}(1-\frac{\alpha}{2})$. Then, under $H_0(G_r)$ —the hypothesis which states that $\beta_j^* = 0$ for $j \in G_r$ implying that $\tilde{\theta}_r = 0$—, we have, for any $\alpha \in (0,1)$:

$$\begin{aligned}
\mathbb{P}(\hat{p}_r^{\mathcal{G}} \leqslant \alpha|\mathbf{Z}) &= 1-\mathbb{P}\left[\sqrt{n}\left[\frac{|\hat{\theta}_r|}{\sigma_\eta\sqrt{\hat{\Omega}_{rr}}}-a\right]_+ \leqslant q_{1-\frac{\alpha}{2}}\middle|\mathbf{Z}\right]\\
&\leqslant 1-\mathbb{P}\left[\sqrt{n}\left[\frac{|\hat{\theta}_r|}{\sigma_\eta\sqrt{\hat{\Omega}_{rr}}}-\frac{\nu_r\,\|\beta^*\|_1}{\sigma_\epsilon C_r\sqrt{\hat{\Omega}_{rr}}}\right]_+ \leqslant q_{1-\frac{\alpha}{2}}\middle|\mathbf{Z}\right]\\
&\leqslant 1-\mathbb{P}\left[\sqrt{n}\left[\frac{|\hat{\theta}_r|-|\kappa_r|}{\sigma_\eta\sqrt{\hat{\Omega}_{rr}}}\right]_+ \leqslant q_{1-\frac{\alpha}{2}}\middle|\mathbf{Z}\right]\\
&= 1-\mathbb{P}\left[\sqrt{n}\left[\frac{|\hat{\theta}_r|-|\theta_r^*|}{\sigma_\eta\sqrt{\hat{\Omega}_{rr}}}\right]_+ \leqslant q_{1-\frac{\alpha}{2}}\middle|\mathbf{Z}\right]\\
&\leqslant 1-\mathbb{P}\left[\sqrt{n}\frac{|\hat{\theta}_r-\theta_r^*|}{\sigma_\eta\sqrt{\hat{\Omega}_{rr}}} \leqslant q_{1-\frac{\alpha}{2}}\middle|\mathbf{Z}\right]\\
&= \alpha+o_{\mathbb{P}}(1)\ .
\end{aligned} \tag{5.21}$$

Finally, we have built a cluster-wise adjusted p-value family which (asymptotically) exhibits, with low probability ($< \alpha$), low value ($< \alpha$) for the clusters which contain only zero weight features.

Now, let us come back to the interpretation and choice for the constant a. In Prop. 5.4.6, we have shown that, when groups are not independent, a group weight in the compressed model can be non-zero even if the group only contains zero weight features. However, the absolute value of the weight of a such group is upper bounded. Then, we introduced $a \in \mathbb{R}^+$ in (5.20) to increase the p-values by a relevant amount and keep statistical guarantees concerning the non-discovery of a such group. The value of a depends on the physics of the problem and on the choice of clustering. While the physics of the problem is fixed, the choice of clustering has a strong impact on the left term of (5.19) and a "good" choice of clustering results in a lower a (less correction). To estimate a, we need to find an upper bound of $\|\beta^*\|_1$, a lower bound of σ_ϵ and to estimate the left term of (5.19). In practice, to compute p-values, we took $a = 0$ since the formula in (5.12) was already conservative for all the problems we considered.

To complete the proof in the case of correlated clusters, one can proceed as in uncorrelated cluster case taking (5.20) instead of (5.12).

5.5 NUMERICAL SIMULATIONS

In this section, we run several simulations in order to give empirical evidence of the theoretical properties of ecd-Lasso and compare its recovering properties with two other procedures.

5.5.1 3D Simulations

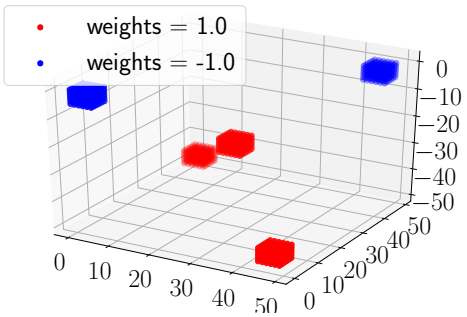


Figure 5.4: 3D representation of the true parameter vector β^*. The support is composed of five connected regions, four are situated in corners of the cubic map and the last region is situated in the center of the cube.

The proposed simulations have a 3D structure and aim at approximating the characteristics of image volumes dataset, such as MRI scans (see *e.g.,* (Marcus et al., 2007)). The feature space considered is a 3D cube with

edge length $H = 50$, then $p = H^3 = 125k$ covariates (voxels) and we took $n = 400$ samples. To construct $\boldsymbol{\beta}^*$, we define a 3D weight vector $\tilde{\boldsymbol{\beta}}^*$ with five non-zero regions represented in Fig. 5.4 and then flatten $\tilde{\boldsymbol{\beta}}^*$ in a vector $\boldsymbol{\beta}^*$ of size p. Each non-zero region is a cube of width $h = 6$, leading to a size of support $s_* = 5h^3 = 1\,080$ (around 1% of p). Four regions are situated in corners of the map and the last region is situated in the center of the cube. Then, to construct $\mathbf{X}$, we first construct a 3D design matrix $\tilde{\mathbf{X}}$ by drawing p random normal vectors of size n that are spatially smoothed with a 3D Gaussian filter to create a feature correlation structure related to the feature space geometry, then we reuse the same flattening to go from $\tilde{\mathbf{X}}$ to $\mathbf{X}$ the $n \times p$ design matrix. The intensity of the spatial smoothing is designed to achieve similar feature correlation as that measure from the Oasis dataset (Marcus et al., 2007); namely the correlation between two adjacent voxels is around 0.8. We also set $\sigma_\varepsilon = 8.5$, to approximately get $\mathrm{SNR}_y = 10$ ($= 10\,\mathrm{dB}$), where the signal to noise ratio (SNR) is defined by $\mathrm{SNR}_y = \|\mathbf{X}\boldsymbol{\beta}^*\|_2^2 / \|\boldsymbol{\epsilon}\|_2^2$. We also try two other scenarios with higher level of noise: $\sigma_\varepsilon = 12$ leading to $\mathrm{SNR}_y = 5$ ($= 7\,\mathrm{dB}$) and $\sigma_\varepsilon = 19$ leading to $\mathrm{SNR}_y = 2$ ($= 3\,\mathrm{dB}$). When running ecd-Lasso, we took $q = 500$, which is relevant for medical imaging contexts. Note that a small q generally improves the recovery properties of ecd-Lasso, it entails a high tolerance. Also, we took the number of bootstraps B equal to 100.

5.5.2 Alternative methods

In such settings, there are almost no existing powerful procedures that provide statistical guarantees. The only dedicated method we found was proposed by Gaonkar and Davatzikos (2012) and its assumptions could be problematic in practice. The second method we propose is a permutation test made over SVR weights. This procedure is theoretically valid assuming a pivotality property of estimated weights. Both method are related to the SVR formulation (Cortes and Vapnik, 1995; Smola and Schölkopf, 2004).

APPROXIMATIVE PERMUTATION THRESHOLD SVR. Here, we introduce Approximative Permutation Threshold SVR, a statistical inference procedure that produces a weight map and confidence intervals around it; it is also almost equivalent to thresholding the SVR weights non-uniformly. This procedure was first presented by Gaonkar and Davatzikos (2012). First, the authors derived an estimated weight $\hat{\mathbf{w}}^{\mathrm{APT}}$ linearly related to the target by approximating the hard margin SVM formulation, their estimator is given by the following equation:

$$\hat{\mathbf{w}}^{\mathrm{APT}} = \mathbf{L}\,\mathbf{y} \ , \tag{5.22}$$

where $\mathbf{y}$ is the target variable and $\mathbf{L} \in \mathbb{R}^{p \times n}$ only depends on the design matrix $\mathbf{X}$:

$$\mathbf{L} = \mathbf{X}^\top \left[(\mathbf{X}\mathbf{X}^\top)^{-1} - (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{1}(\mathbf{1}^\top(\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{1})^{-1}\mathbf{1}^\top(\mathbf{X}\mathbf{X}^\top)^{-1} \right] \quad , \quad (5.23)$$

where $\mathbf{1} \in \mathbb{R}^n$ is a vector of ones. The approximation made by (5.22) is notably valid under the assumption that all the data samples are support vectors, which might hold at least if $n \ll p$. Then, if $\mathbf{y}$ is standardized and if n is large enough (so that the central limit theorem holds), one expects that under the null hypothesis for the j-th covariate:

$$\hat{\mathbf{w}}_j^{\mathrm{APT}} \sim \mathcal{N}(0, \sum_{k=1}^{n} \mathbf{L}_{j,k}^2) \quad . \quad (5.24)$$

From (5.24), p-values can be computed and corrected by applying a Bonferroni correction (multiplying the raw p-values by a factor p).

PERMUTATION TEST SVR. Now, we introduce a uniform thresholding strategy of SVR weights based upon the Westfall-Young permutation test procedure. To derive corrected p-values from a permutation test, we first regress the design matrix against the response vector using a linear SVR to obtain an estimate $\hat{\mathbf{w}}^{\mathrm{SVR}}$ of the weights map similarly as made in the Thr-SVR procedure. Then, permuting randomly R times the response vector and regressing the design matrix against the permuted response by a linear SVR, we obtain R maps $(\hat{\mathbf{w}}^{\mathrm{SVR},(r)})_{r \in [R]}$. We can now apply the Westfall-Young step-down maxT adjusted p-values algorithm (Westfall and Young, 1993, p. 116-117) taking the raw SVR weights instead of the usual t-statistics to derive the corrected p-values. A sufficient assumption to ensure the validity of the p-values is the pivotality of the SVR weights. Keeping the corrected p-values that are less than a given significance level —equal to 10% in this study— this procedure is equivalent to thresholding the SVR weight map. We call this procedure Permutation Test SVR. To perform the permutation test procedure, we use $R = 1000$ permutations.

5.5.3 Results

In Fig. 5.5, we study the FWER and δ-FWER control for all the methods presented, for $\delta = 5$ voxels. Under appropriate assumptions, ecd-Lasso theoretically controls the δ-FWER for δ equal to the largest cluster diameter which is around 15 voxels in average over the 100 simulations for each setting. However, by ensembling several solution (p-value maps), the probability of making false discoveries at this theoretical distance (upper-bound) is greatly reduced. Then, in practice the control is generally even effective for δ equal to the average cluster radius, here around $\delta = 5$.

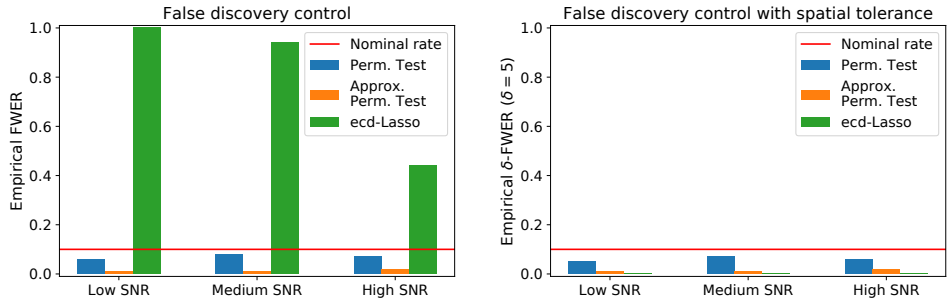


Figure 5.5: Empirical FWER control. Left: Excepted ecd-Lasso, all methods should control the FWER. For those methods, the empirical FWER remains below the 10% nominal rate as expected. Right: Under appropriate assumptions, ecd-Lasso theoretically controls the δ-FWER for δ equal to the largest cluster diameter. In practice the control is generally even effective for δ equal to the average cluster radius, here around $\delta = 5$ voxels. Concerning ecd-Lasso, while the FWER was not controlled, the empirical δ-FWER remains below the nominal rate as expected; this means that mistakes are made close to the support which is more acceptable in many experimental settings.

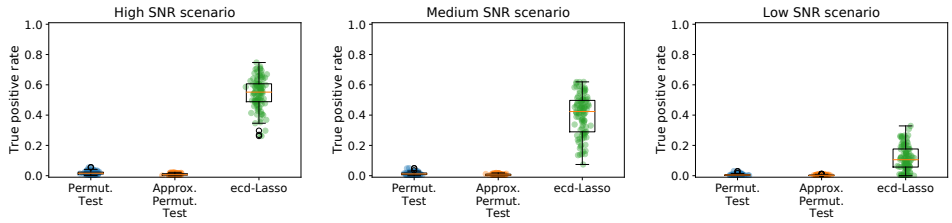


Figure 5.6: True positive rate. Here, we exhibit the recovery rate when controlling the FWER (or δ-FWER in the case of ecd-Lasso) at 10%. One can notice that ecd-Lasso is clearly improving over all other methods for all noise regimes.

In Fig. 5.6, we plot the recovery rate when controlling the FWER (or δ-FWER in the case of ecd-Lasso) at 10%. In every setting ecd-Lasso clearly improves over all other methods. Additionally, for all methods, increasing the SNR makes the inference problem easier and the true discovery rate increases; this effect was expected.

5.5.4 Discussion

In this chapter, we have established the theoretical properties of ecd-Lasso and exhibited that they hold on a given simulation.

In theory, the required spatial tolerance is equal to the largest cluster diameter. In practice, since we compress the problem with different clustering choices that contain clusters of different sizes and aggregate the solutions, the required spatial tolerance is generally lower than the average cluster

radius. Also, we have observed that ecd-Lasso is conservative even with this reduced spatial tolerance. This is mainly due the fact that the Bonferroni correction and the aggregation procedure are conservative.

In Chapter 6, we conduct a thorough empirical validation to support the theoretical claims and show the sustainability of the assumptions.

5.6 SUPPLEMENTARY MATERIAL

5.6.1 Proof of Prop. 5.4.3

Proof. *(i)* Suppose that $H_0^\delta(j)$ is true. Since the diameters of the groups are all smaller than δ, all the covariates in $G_{g(j)}$ have a corresponding weight equal to zero. Thus, using Prop. 5.4.1, we have $\theta^*_{g(j)} = 0$, *i.e.*, we are under $H_0(G_{g(j)})$. Under this last null-hypothesis, using (5.13) and (5.16), we have asymptotically:

$$\forall \alpha \in (0,1),\ \mathbb{P}(\hat{p}^{\mathcal{G}}_{g(j)} \leqslant \alpha) = \mathbb{P}(\hat{p}_j \leqslant \alpha) = \alpha \ .$$

This last result being true for any $j \in N^\delta$, we have shown that the elements of the family $\hat{p}$ asymptotically control the δ-type 1 error.

(ii) As mentioned in Sec. 5.4.3, we know that, the family $\hat{q}^{\mathcal{G}}$ asymptotically controls the FWER, *i.e.*, for $\alpha \in (0,1)$ we have $\mathbb{P}(\min_{r \in N_{\mathcal{G}}} \hat{q}^{\mathcal{G}}_r \leqslant \alpha) \leqslant \alpha$. Let us denote by $g^{-1}(N_{\mathcal{G}})$ the set of indexes of covariates that belong to the groups of $N_{\mathcal{G}}$, *i.e.*, $g^{-1}(N_{\mathcal{G}}) = \{j \in [p] : g(j) \in N_{\mathcal{G}}\}$. Again, given that all the diameters of the groups are smaller than δ and using Prop. 5.4.1, if $j \in N^\delta$ then $g(j) \in N_{\mathcal{G}}$. That is to say $N^\delta \subset g^{-1}(N_{\mathcal{G}})$. Then, we have the following result:

$$\min_{j \in N^\delta}(\hat{q}_j) \geqslant \min_{j \in g^{-1}(N_{\mathcal{G}})}(\hat{q}_j) \ .$$

We can also notice that:

$$\begin{aligned}\min_{j \in g^{-1}(N_{\mathcal{G}})}(\hat{q}_j) &= \min_{j \in g^{-1}(N_{\mathcal{G}})}(\hat{q}^{\mathcal{G}}_{g(j)}) \\ &= \min_{g(j) \in N_{\mathcal{G}}}(\hat{q}^{\mathcal{G}}_{g(j)}) \ .\end{aligned}$$

Replacing $g(j) \in [q]$ by $r \in [q]$, and using (5.15), we obtain the following asymptotic result:

$$\forall \alpha \in (0,1),\ \mathbb{P}(\min_{j \in N^\delta}(\hat{q}_j) \leqslant \alpha) \leqslant \mathbb{P}(\min_{r \in N_{\mathcal{G}}} \hat{q}^{\mathcal{G}}_r \leqslant \alpha) \leqslant \alpha \ .$$

This establishes that the family $(\hat{q}_j)_{j \in [p]}$ asymptotically controls the δ-FWER. □

5.6.2 Proof of Prop. 5.4.4

The proof of Prop. 5.4.4 is inspired by the one proposed by Meinshausen, Meier, and Bühlmann (2009). However, it is quite different since we can not remove the term $\min_{j\in N^\delta}$ and have to work with it to obtain the desired inequality; this makes the proof a bit more difficult. First, we start by making a short remark about the γ-quantile quantity.

Definition 5.6.1 (empirical γ-quantile)**.** *For a set V of real numbers and $\gamma \in (0,1)$, let*

$$\gamma\text{-quantile}(V) = \min\left\{ v \in V : \frac{1}{|V|}\sum_{w\in V} \mathbb{1}_{\{w\leqslant v\}} \geqslant \gamma \right\} \quad . \tag{5.25}$$

Remark 5.6.1. *For a set of real number V and for $a \in \mathbb{R}$, let us define the quantity $\pi(a, V)$ by the following:*

$$\pi(a, V) = \frac{1}{|V|}\sum_{v\in V} \mathbb{1}_{\{v\leqslant a\}} \quad . \tag{5.26}$$

Then, for $\gamma \in (0,1)$, the two events $E_1 = \{\pi(a, V) \geqslant \gamma\}$ and $E_2 = \{\gamma\text{-quantile}(V) \leqslant a\}$ are identical.

Now, we give the proof of Prop. 5.4.4.

Proof. First, one can notice that, from (5.17), we have:

$$\min_{j\in N^\delta}(\tilde{q}_j(\gamma)) \geqslant \min\left\{ 1, \gamma\text{-quantile}\left(\left\{ \min_{j\in N^\delta}\left(\frac{\hat{q}_j^{(b)}}{\gamma}\right) : b \in [B]\right\}\right)\right\} \quad .$$

Then, for $\alpha \in (0,1)$:

$$\begin{aligned}
&\mathbb{P}\left(\min_{j\in N^\delta}(\tilde{q}_j(\gamma)) \leqslant \alpha\right) \\
&\quad\leqslant \mathbb{P}\left(\min\left\{ 1, \gamma\text{-quantile}\left(\left\{ \min_{j\in N^\delta}\left(\frac{\hat{q}_j^{(b)}}{\gamma}\right) : b \in [B]\right\}\right)\right\} \leqslant \alpha\right) \\
&\quad= \mathbb{P}\left(\gamma\text{-quantile}\left(\left\{ \min_{j\in N^\delta}\left(\frac{\hat{q}_j^{(b)}}{\gamma}\right) : b \in [B]\right\}\right) \leqslant \alpha\right) \quad .
\end{aligned}$$

Using Rem. 5.6.1, for $\gamma \in (0,1)$, with:

$$V = \left\{ \min_{j\in N^\delta}\left(\frac{\hat{q}_j^{(b)}}{\gamma}\right) : b \in [B]\right\} \quad \text{and} \quad a = \alpha \ ,$$

and noticing that:

$$\pi\left(\alpha, \left\{\min_{j\in N^\delta}\left(\frac{\hat{q}_j^{(b)}}{\gamma}\right) : b \in [B]\right\}\right) = \frac{1}{B}\sum_{b=1}^{B} \mathbb{1}_{\left\{\min_{j\in N^\delta}(\hat{q}_j^{(b)})\leqslant \alpha\gamma\right\}},$$

then, we have:

$$\begin{aligned}\mathbb{P}\left(\gamma\text{-quantile}\left(\left\{\min_{j\in N^\delta}\left(\frac{\hat{q}_j^{(b)}}{\gamma}\right) : b \in [B]\right\}\right) \leqslant \alpha\right) \\ = \mathbb{P}\left(\frac{1}{B}\sum_{b=1}^{B} \mathbb{1}_{\left\{\min_{j\in N^\delta}(\hat{q}_j^{(b)})\leqslant \alpha\gamma\right\}} \geqslant \gamma\right).\end{aligned}$$

Then, the Markov inequality gives:

$$\mathbb{P}\left(\frac{1}{B}\sum_{b=1}^{B} \mathbb{1}_{\left\{\min_{j\in N^\delta}(\hat{q}_j^{(b)})\leqslant \alpha\gamma\right\}} \geqslant \gamma\right) \leqslant \frac{1}{\gamma}\mathbb{E}\left[\frac{1}{B}\sum_{b=1}^{B} \mathbb{1}_{\left\{\min_{j\in N^\delta}(\hat{q}_j^{(b)})\leqslant \alpha\gamma\right\}}\right].$$

Then, using the assumption that the B families $(\hat{q}_j^{(b)})_{j\in[p]}$ control of the δ-FWER (last inequality), we have:

$$\frac{1}{\gamma}\mathbb{E}\left[\frac{1}{B}\sum_{b=1}^{B} \mathbb{1}_{\left\{\min_{j\in N^\delta}(\hat{q}_j^{(b)})\leqslant \alpha\gamma\right\}}\right] = \frac{1}{\gamma}\frac{1}{B}\sum_{b=1}^{B} \mathbb{P}\left(\min_{j\in N^\delta}(\hat{q}_j^{(b)}) \leqslant \alpha\gamma\right) \leqslant \alpha\,.$$

Finally, we have shown that, for $\alpha \in (0,1)$:

$$\mathbb{P}\left(\min_{j\in N^\delta}(\tilde{q}_j(\gamma)) \leqslant \alpha\right) \leqslant \alpha\ .$$

This establishes that the family $(\tilde{q}_j(\gamma))_{j\in[p]}$ controls the δ-FWER. □

Remark 5.6.2. *Using the aggregation formula of (5.17), we could also construct a family that controls the δ-type 1 error from B families controlling the δ-type 1 error.*

6 EMPIRICAL VALIDATION

In Chapter 4, we presented ecd-Lasso, a multivariate statistical inference procedure well suited for very high dimensional structured data such as neuroimaging data. In Chapter 5, we give the statistical guarantees provided by ecd-Lasso.

In this chapter, we evaluate the statistical properties of ecd-Lasso along with three alternative standard methods by performing a thorough empirical study using fMRI datasets. We also study the impact of the choice of the main free parameter of ecd-Lasso: the number of clusters C. Finally, we show that ecd-Lasso exhibits the best recovery properties while ensuring the expected statistical control. Also note that, the content of this chapter has been submitted to the *NeuroImage* journal and has received a very positive first feedback.

6.1 INTRODUCTION

In Chapter 2, we introduced decoding and explained that, while it produces weight maps used to predict diseases or behaviors from brain activation maps, little or nothing can be concluded about the significant features of these weight maps. Indeed, those maps do not come with well-controlled statistical properties, making decoding models hard to interpret. For instance, considering linear Support Vector Machines (SVM) (Cortes and Vapnik, 1995) or linear Support Vector Regression (SVR) (Smola and Schölkopf, 2004), that are very popular in neuroimaging (Pereira, Mitchell, and Botvinick, 2009; Rizk-Jackson et al., 2011), a natural way to recover predictive regions is to threshold the estimated weight maps. However, this approach is problematic for two reasons: there exists no clear way to choose the threshold as a function of a desired significance, and it is unclear whether such a thresholded map is still an accurate predictor of the outcome.

The main goal of the present work is to conduct an empirical study in order to benchmark standard procedures that are used for source localization in the neuroimaging context and more specifically when dealing with fMRI data. Then, our first contribution is to show that the natural procedure which consists in thresholding standard decoders, such as SVR, is not a relevant solution. In this respect, we consider two thresholding strategies: one that computes a threshold through a parametric approach, and another

one that derives the threshold by performing a permutation test. These two thresholding strategies can be derived from statistical testing considerations —yet, these statistical properties are not assumption free.

Another procedure, developed by Gaonkar and Davatzikos (2012) and referred to as Ada-SVR, derives decoder maps that come with confidence intervals around the estimated weights. Ada-SVR was specifically designed for this neuroimaging source localization problem then we also consider this procedure in this chapter. Similarly to the thresholding procedures, Ada-SVR relies on algorithmic shortcuts, approximations and assumptions that are more or less problematic in practice.

Finally, the ecd-Lasso algorithm that was presented in Chapter 4 produces maps which verify interesting statistical properties derived in Chapter 5; then we benchmark the procedure and validate empirically its theoretical properties. Consequently, our second contribution is to compare and validate the theoretical results obtained in Chapter 5 in several real fMRI scenarios.

For all methods considered, the control of false detections is only achieved within a certain theoretical framework, and given a series of assumptions that are not always checked. It is thus fundamental to analyze their statistical behavior with an extensive empirical study. We present here a set of experiments assessing the accuracy of the error rate control and support recovery on real and semi-synthetic brain-imaging data. In particular, we consider standard experiments adapted from the seminal work of Eklund, Nichols, and Knutsson (2016) to highlight the properties and limitations of these statistical models.

The present chapter is organized as follows: in Sec. 6.2, we briefly recall the underlying model and useful quantities for statistical inference with spatial tolerance, that we introduced in Chapter 5, taking a more practical approach; in Sec. 6.3, we describe thoroughly the methods that be tested to address the source localization problem; Sec. 6.4 and Sec. 6.5 follow with extensive experiments on simulations and large-scale fMRI datasets that study the behavior of the benchmarked solutions regarding false positive control and recovery.

6.2 MODEL FORMULATION AND STATISTICAL TOOLS

In this section, we restate the noise model and introduce the δ-FWER taking a more practical approach than in Chapter 5.

6.2.1 Formal problem setting

We recall that the model that describes the neuroimaging problem we are dealing with is given by (2.7):

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon} \ ,$$

where the response vector is denoted by $\mathbf{y} \in \mathbb{R}^n$, the design matrix by $\mathbf{X} \in \mathbb{R}^{n \times p}$, the parameter vector by $\boldsymbol{\beta}^* \in \mathbb{R}^p$ and the random error vector by $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma_\varepsilon^2 \mathbf{I}_n)$ where $\sigma_\varepsilon > 0$ is the unknown standard deviation. We also recall that in the neuroimaging context, n the number of samples is the number of brain images available and p, the number of covariates, represents the number of voxels in each scan —a covariate being given by the level of activation of a voxel. Our aim is to infer the weight map $\boldsymbol{\beta}^*$ that links the activation maps $\mathbf{X}$ to the conditions $\mathbf{y}$ with statistical guarantees on the proposed estimator. We also assume sample independence, sparsity and spatial structure of the weight map; for further details we refer the reader to Chapter 5.

6.2.2 δ-Family Wise Error Rate (δ-FWER)

In Chapter 2, we have explained that it was quite natural to incorporate a spatial tolerance in the sought statistical control in order to untangle this hard source localization problem. Then, to control the occurrence of false discoveries, we use the generalization of the FWER (Hochberg and Tamhane, 1987a) proposed in Chapter 5 called δ-FWER. This control is particularly well adapted for the problem we are given since it does not take into account the false discoveries made at a distance larger than δ from the support. In this section, we take another approach to define the δ-FWER, it is more practical and complementary to the approach taken in Chapter 5 since we do not consider p-value families but simply a general estimated support.

TRUE SUPPORT UNDER LINEAR MODEL ASSUMPTION. For the current chapter only, to ease notation, we use simply S to denote the true *support*. In the setting we are given, S is the subset of voxels that explains the outcome $\mathbf{y}$, while the *null region* N is its complementary $N = [p] \setminus S$. More precisely, S is the set of voxels that are associated with $\mathbf{y}$, *given* all the other voxels:

$$\begin{aligned} &\forall j \in S, \qquad \mathbf{X}_j \not\perp\!\!\!\perp \mathbf{y} \mid \{\mathbf{X}_k, k \in [p] \setminus \{j\}\} \ , \\ &\forall j \in N, \qquad \mathbf{X}_j \perp\!\!\!\perp \mathbf{y} \mid \{\mathbf{X}_k, k \in S\} \ , \end{aligned} \tag{6.1}$$

where the sign $\perp\!\!\!\perp$ denotes the (conditional) independence. Then, under the linear assumption made in (2.7), $\boldsymbol{\beta}_j^* = 0$ holds for any $j \in N$ and $\boldsymbol{\beta}_j^* \neq 0$ for any $j \in S$.

δ-NEIGHBORHOOD. The variables $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_p$ can also be characterized by the spatial proximity of their underlying voxels in brain space: given $\delta \geqslant 0$, a voxel $k \in [p]$ is in the δ-neighborhood of a voxel (or a set of voxels) if their distance is less than δ.

δ-NULL REGION. For $\delta \geqslant 0$, we denote by $S^{(\delta)}$ the δ-dilation of the support S, *i.e.,* the set of voxels in S or in its δ-neighborhood. By definition, $S \subset S^{(\delta)}$. We denote by N^{δ} the δ-erosion (inverse operation of a δ-dilation) of the null region N, implying that $N^{\delta} \subset N$. From the definition of N we have immediately:

$$N^{\delta} = [p] \setminus S^{(\delta)} \ , \tag{6.2}$$

We refer to N^{δ} as the δ-null region. As shown in Fig. 6.1, we interpret the δ-null region as the subset of the covariates which are at a distance less than δ from the support covariates.

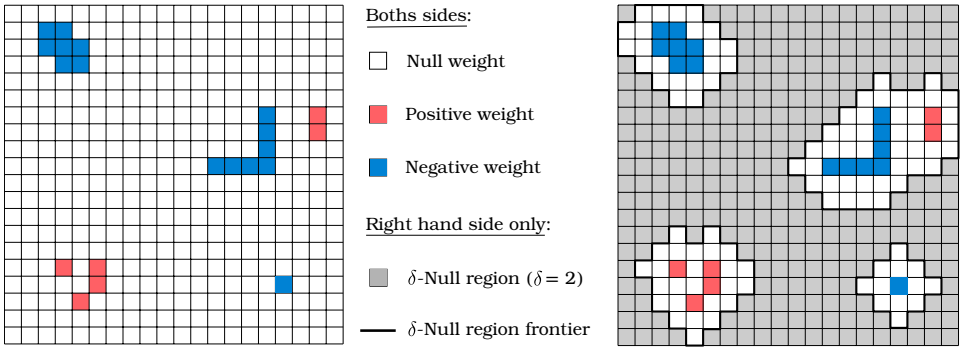


Figure 6.1: Spatial tolerance to false discoveries. Left: example of 2D-weight map, small squares represent voxels. The map is sparse. Right: representation of the δ-null region for the associated map with $\delta = 2$. The covariates in the δ-null region are "far" from non-null covariates, discoveries in this area are highly undesired. Discovering a null covariate "close" to a non-null covariate is tolerated.

δ-FAMILY WISE ERROR RATE (δ-FWER). If we have an estimate of the support $\hat{S} \subset [p]$, we recall that the FWER is defined as the probability of making a false detection (Hochberg and Tamhane, 1987a):

$$\text{FWER}(\hat{S}) = \mathbb{P}(\hat{S} \cap N \neq \varnothing) \ . \tag{6.3}$$

Similarly, given $\delta \geqslant 0$, we defined the δ-FWER to be

$$\delta\text{-FWER}(\hat{S}) = \mathbb{P}(\hat{S} \cap N^{\delta} \neq \varnothing) \ , \tag{6.4}$$

i.e., the probability of making a detection at distance more than δ from the true support. The δ-FWER control is thus weaker than the FWER control, except when $\delta = 0$ and when the true support is empty (*i.e.,* $N = [p]$), in which case the δ-FWER coincides with the classical FWER.

δ-PRECISION. Given an estimate of the support $\hat{S} \subset [p]$, its precision is defined as the proportion of true discoveries within $\hat{S}$:

$$\text{precision}(\hat{S}) = \frac{|\hat{S} \cap S|}{|\hat{S}|} \ . \tag{6.5}$$

Similarly, given $\delta \geqslant 0$, we defined the δ-precision as

$$\delta\text{-precision}(\hat{S}) = \frac{|\hat{S} \cap S^{(\delta)}|}{|\hat{S}|} \ , \tag{6.6}$$

i.e., the proportion of discoveries at distance less than δ from the true support. Notice that $\delta\text{-precision}(\hat{S}) \geqslant \text{precision}(\hat{S})$.

6.3 METHODS

In this section, we provide a bit of context regarding the current practices for solving the source localization problem, we then describe thoroughly the benchmarked procedures with their assumptions and statistical guarantees.

6.3.1 Current practices

NAKED EYE CRITERIA. Probably the most natural procedure used to recover discriminative patterns is to threshold decoders with high prediction performance —a popular choice is the linear SVM/SVR decoder (Pereira, Mitchell, and Botvinick, 2009; Rizk-Jackson et al., 2011). Thresholding decoder maps at a uniform value —*i.e.*, the threshold is the same for all weights— is probably the most common practice in neuroimaging; threshold value being generally arbitrary: "naked-eye criteria". It is not thought of as a statistical operation, and is sometimes left to the reader, who is presented unthresholded maps and yet told to interpret only the salient features of these maps.

PERMUTATION TEST. Permutation testing can also be used to derive a uniform threshold with explicit guarantees. The classical Westfall-Young permutation test procedure (Westfall and Young, 1993) is well-known in the univariate context to control the FWER (Anderson, 2001), but its application to multivariate testing is not as straightforward. Then, instead of considering the usual t-statistics, a permutation test can use the linear SVR weights. An estimated weight map must be computed for the original problem and for several permuted problems before performing the Westfall-Young procedure; this method is detailed in Sec. 6.3.3.

Under some assumptions (see Sec. 6.3.2 and Sec. 6.3.3) that are more or less problematic in practice, these two previous uniform thresholding strategies might recover the predictive patterns with FWER control.

APPROXIMATED PERMUTATION TEST. The issues with the previous permutation testing procedure is i/the pivotality assumption —which is not verified for SVR weights— and ii/the computational cost. A method proposed by Gaonkar and Davatzikos (2012), specifically designed for neuroimaging settings, relies on the analytic approximation of a permutation test performed over a linear SVM/SVR estimator. This method produces confidence intervals around the weights of the proposed estimator. Then, under some assumptions (see Sec. 6.3.4) that are not always met in practice, this procedure controls the FWER. It is almost equivalent to thresholding the SVR weights with a non-uniform threshold —*i.e.*, the threshold is specific to each weight. We refer to it as Adaptive Permutation Threshold SVR (Ada-SVR) from now on.

6.3.2 Thresholded SVR (Thr-SVR)

Thresholded SVR (Thr-SVR) is a procedure that thresholds uniformly the estimated SVR weight map, keeping extreme weights; this method corresponds to the most standard and simple approach to recover predictive patterns. The first step is to derive the SVR weights $\hat{\mathbf{w}}^{\text{SVR}}$. Then, assuming that the estimated weights of the null region are sampled from a given distribution centered on 0, the corresponding standard deviation σ_{SVR} can be approximated with the following estimator:

$$\hat{\sigma}_{\text{SVR}} = \sqrt{\frac{1}{p}\sum_{j=1}^{p}(\hat{\mathbf{w}}_j^{\text{SVR}})^2} \quad . \tag{6.7}$$

We could also consider other estimators to approximate this quantity (*e.g.*, Schwartzman et al., 2009) but the former is simple and at worst biased upward when the support is not empty. Now, assuming a Gaussian distribution for the SVR weights in the null region, *i.e.*, for $j \in N$:

$$\hat{\mathbf{w}}_j^{\text{SVR}} \sim \mathcal{N}\left(0, \sigma^2_{\text{SVR}}\right) \quad , \tag{6.8}$$

we can produce (corrected) p-values by applying a Bonferroni correction. The produced p-values are at worst conservative under the two assumptions discussed in Sec. 6.6. In this procedure, the regression method considered is a linear SVR but similar results were obtained with other procedures (*e.g.*, Ridge regression).

6.3.3 Permutation Test SVR (Perm-SVR)

The other uniform thresholding strategy of SVR weights we consider is the method described in Chapter 5. It is based upon the Westfall-Young permutation test procedure and we refer to this procedure as Permutation Test SVR (Perm-SVR). The only difference between Perm-SVR and the Thr-SVR procedure is the way of computing the threshold. Please refer to Sec. 5.5.2 for a detailed presentation of the method.

6.3.4 Adaptive Permutation Threshold SVR (Ada-SVR)

Adaptive Permutation Threshold SVR (Ada-SVR) is a statistical inference procedure that produces a weight map and confidence intervals around it; it is also almost equivalent to thresholding the SVR weights non-uniformly. This procedure was first presented by Gaonkar and Davatzikos (2012). Please refer to Sec. 5.5.2 for a detailed presentation of the method.

6.3.5 Ensemble of Clustered Desparsified Lasso Algorithm (ecd-Lasso)

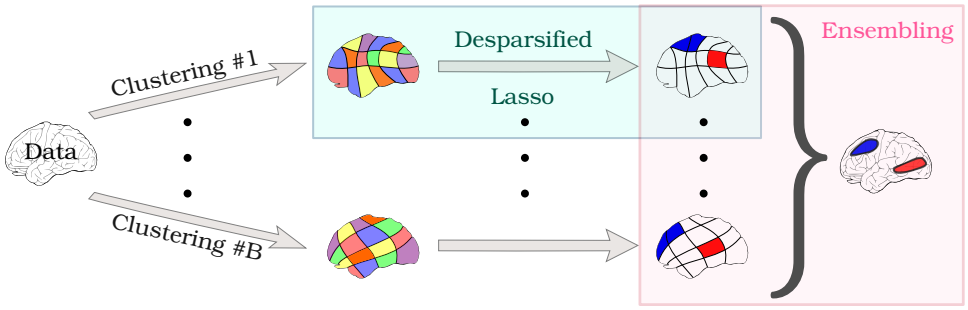


Figure 6.2: Ensemble of Clustered Desparsified Lasso (ecd-Lasso) algorithm. ecd-Lasso combines three algorithmic steps: a clustering (or parcellation) procedure applied to images, the Desparsified Lasso procedure (statistical inference) to derive statistical maps, and an ensembling method that synthesizes several statistical maps. In the first step, B clusterings of voxels are generated using B random subsamples of the original sample. Then, for each grouping-based data reduction, a statistical inference procedure is run resulting in B z-score maps (or p-value maps). Finally, these maps are ensembled into a final z-score map using an aggregation method that preserves statistical properties.

ECD-LASSO IN A NUTSHELL We recall that ecd-Lasso is a multivariate statistical inference procedure designed for high dimensional spatial data introduced in Chevalier, Salmon, and Thirion (2018). As illustrated in Fig. 6.2, ecd-Lasso relies on three steps: a spatially-constrained clustering algorithm for reducing the problem dimension, a statistical inference procedure for

deriving statistical maps, and an ensembling method for aggregating the statistical maps. The reader can refer to Chapter 4 for a comprehensive description of the method.

CHOOSING δ FOR δ-FWER CONTROL Under assumptions given in Chapter 5 —notably sparsity and smoothness of the true weight map and i.i.d. data samples— ecd-Lasso gives statistical guarantees, namely it controls the δ-FWER. Theoretically, the minimal spatial tolerance δ that guarantees a control of the δ-FWER with ecd-Lasso is the largest parcel diameter. However, in practice, we aggregate many statistical maps obtained from different choices of voxel grouping, reducing this tolerance to the average radius. Then, the value of δ for which we observe the δ-FWER control varies approximately linearly with the cubic root of the average number of voxels per cluster. In standard fMRI settings, we propose the following formula for δ:

$$\delta_0 = \left(\frac{p}{2C}\right)^{1/3} , \tag{6.9}$$

the ratio p/C being the average number of voxels per cluster, δ_0 is a distance in voxel size unit. However, when the setting is particularly favorable for inference, *e.g.*, if $\log(n)/C$ is large, or σ_ε is small, the choice of δ given by (6.9) might be slightly too liberal. We have found empirically that a suitable multiplicative factor, denoted by $\tau > 0$, that could be used to correct δ_0 is given by:

$$\tau = -45 \log\left(\frac{\sigma_\varepsilon}{\mathrm{std}(\mathbf{y})}\right) \frac{\log(n)}{C} , \tag{6.10}$$

where σ_ε is the standard deviation of the noise ε. In practice σ_ε has to be estimated; in the fMRI datasets we studied, estimates of $\frac{\sigma_\varepsilon}{\mathrm{std}(\mathbf{y})}$ were close to 0.1. However, given the heuristic derivation of this quantity and the uncertainty about the value of τ, we do not recommend correcting δ_0 with a factor lower than 1 as it could lead to a dramatic under estimation of the valid δ. Then, the final formula to compute the δ such that δ-FWER control is ensured, is:

$$\delta^* = \max(1, \tau)\ \delta_0 \ . \tag{6.11}$$

Note that the formula given by (6.9) and even (6.11) are not bullet proof but rather give reasonable estimates of practical value to take for δ.

ECD-LASSO HYPER PARAMETERS. The number of clusters C is a crucial hyperparameter of ecd-Lasso. Generally, a suitable C depends on intrinsic physical properties of the problem and on the targeted spatial tolerance δ. Decreasing C increases the statistical power while reducing the spatial precision. When working with fMRI data, taking $C = 500$ is a fair default

value achieving a suitable trade-off between spatial precision and statistical power when the number of samples is a few hundreds. With this choice, the spatial tolerance should be close to $\delta = 10\,\text{mm}$ when working with masked fMRI data.

As a more adaptive approach, we recommend to tune C according to n *e.g.,* $C \in [n/2, n]$. This choice should still ensure the δ-FWER control with δ given by (6.9) and is justified by the results given in Sec. 6.5.6 (experiment being described in Sec. 6.4.6).

The parameter B, the number of cd-Lasso solutions to be aggregated, is discussed in Sec. 6.3.5. The larger B the more stable the solution, yet the heavier the computational cost. In our experiments, we have set $B = 25$ (see Hoyos-Idrobo et al. (2018) for a more complete discussion on this parameter).

6.3.6 Implementation

The Python code that implements Thr-SVR, Perm-SVR, Ada-SVR and ecd-Lasso can be found on `https://github.com/ja-che/hidimstat`. Our algorithms are implemented with Python = 3.6.8 and need the following packages Numpy = 1.16.2 (Walt, Colbert, and Varoquaux, 2011), Scipy = 1.2.1 (Virtanen et al., 2020), Scikit-Learn = 0.21 (Pedregosa et al., 2011), Joblib = 0.11 and Nilearn = 0.6.0 (Abraham et al., 2014).

6.4 EXPERIMENTAL PROCEDURES

In this section, we describe datasets and experiments used to benchmark the methods described in Sec. 6.3.

6.4.1 Data

To validate empirically the statistical guarantees of the four algorithms — Thr-SVR, Perm-SVR, Ada-SVR and ecd-Lasso— described in Sec. 6.3, we perform several experiments on resting-state fMRI and task fMRI data. We focus on three datasets: Human Connectome Project (HCP) S900 (900 subjects) resting-state fMRI, HCP S900 task fMRI and Rapid-Serial-Visual-Presentation (RSVP) task fMRI.

HCP RESTING-STATE FMRI DATA. HCP S900 resting-state fMRI dataset (Van Essen et al., 2012) contains 4 runs of 15 minutes resting-state recordings with a 0.76s-repetition time (corresponding to 1200 frames per run) for 796 subjects. We use the MNI-resampled images provided in the HCP S900 release. For this dataset the number of samples is equal to 1200 (only one

run is used) and the number of voxels is 156k after gray-matter masking (the spatial resolution being 2 mm isotropic).

HCP TASK FMRI DATA. We also use the HCP S900 task-evoked fMRI dataset (Van Essen et al., 2012), in which we take the masked 2 mm z-maps of the 796 subjects from 6 tasks to solve 7 binary classification problems: emotion (*emotional face* vs *shape outline*), gambling (*reward* vs *loss*), language (*story* vs *math*), motor hand (*left* vs *right* hand), motor foot (*left* vs *right* foot), relational (*relational* vs *match*) and social (*mental interaction* vs *random interaction*). We consider the fixed-effect maps for each condition, yielding one image per subject per condition (which corresponds to two images per subject for each classification problem). Then, for each problem, the number of samples available is 1592 ($= 2 \times 796$) and the number of voxels is 156k after gray-matter masking.

UNMASKED RSVP TASK FMRI DATA. We also use activation maps obtained from a RSVP task of the the individual brain charting dataset (Pinho et al., 2018), augmented with 9 additional subjects performing the same task, under the same experimental procedures and scanning parameters. No masking is used for this dataset, so that out-of-brain voxels are not withdrawn from preprocessing. We consider the unmasked 3 mm-resolution statistical z-maps of the 6 sessions of the 21 subjects for a reading task with 6 different contrasts that have been grouped into 2 classes: language (words, simple sentences, complex sentences) vs pseudo-language (consonant strings, pseudo-word lists, jabberwocky). The images are all registered to MNI space and per-condition effects are estimated with Nistats v0.0.1 library (Abraham et al., 2014). For this dataset the number of samples available is equal to 756 (21 subjects $\times$ 6 runs $\times$ 6 images per run) and the number of voxels is 173k (unmasked images resampled at 3-mm resolution). We run the inter-subject experiment described in Sec. 6.4.5 with this dataset.

6.4.2 Statistical control on semi-simulated data

A first series of experiments study whether the four different methods exhibit the expected δ-FWER control and are competitive in terms of support recovery, as measured with the δ-precision-recall curve. To do so, we have to construct the true weight map β^*. We generate "semi-simulated" data: generating signals from estimates on real data. To avoid circularity in the definition of the ground truth, we used two different tasks: one to build β^* and another one to define $\mathbf{X}$.

BUILDING A REFERENCE WEIGHT MAP FROM HCP MOTOR HAND. To construct an underlying weight map, we use the motor hand (MH) task of the

HCP S900 task fMRI dataset described in Sec. 6.4.1. Specifically, we build a design matrix $\mathbf{X}_{MH} \in \mathbb{R}^{n \times p}$ from the motor hand task z-maps of all subjects associated with a binary condition index $\mathbf{y}_{MH}$. To obtain an initial weight map β^{SVC}_{MH} we regress $\mathbf{X}_{MH}$ against $\mathbf{y}_{MH}$ by fitting a linear Support Vector Classifier (SVC) (Cortes and Vapnik, 1995). From β^{SVC}_{MH} we only kept the 10% most extreme values ensuring that the connected groups of non zero-weight voxels have a minimal size of 1 cm^3 by removing small clusters. We chose this map (represented in Fig. 6.3 and Fig. 6.4) to be the true weight map $\beta^* \in \mathbb{R}^p$ for the whole simulated experiments.

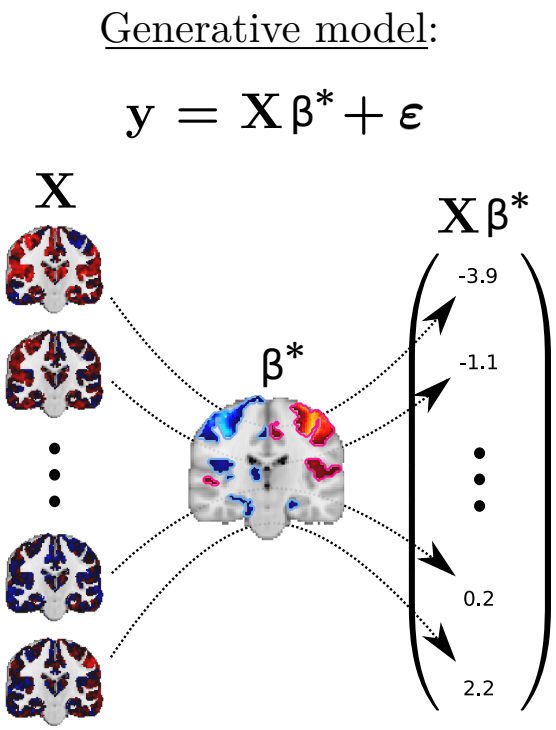


Figure 6.3: Generating a hybrid dataset with known ground truth and actual fMRI data. To generate the response for a given sample we multiply the corresponding brain activation map by the true weight map and add a Gaussian noise with fixed variance. To highlight the predictive regions, we circle them in pink for positive coefficients and in light blue for negative coefficients. As an illustration, we take four different data samples with negative or positive output value.

SIMULATING RESPONSES WITH HCP MOTION DATASET. We then take $\mathbf{X}$ to be the set of z-maps from the emotion task of the HCP S900 task fMRI dataset described in Sec. 6.4.1. To generate a continuous response vector $\mathbf{y}$, we draw a Gaussian random noise vector $\varepsilon \sim \mathcal{N}(0, \sigma^2_\varepsilon \mathbf{I}_n)$ and use the linear

model introduced in (2.7), where $\sigma_\varepsilon = 0.2$ to reach $\mathrm{SNR}_y = 10$, where SNR_y is given by:

$$\mathrm{SNR}_y = \frac{\|\mathbf{X}\,\boldsymbol{\beta}^*\|^2}{n\sigma_\varepsilon^2} \ . \tag{6.12}$$

The way we simulate $\mathbf{y}$ is summarized in Fig. 6.3.

QUANTIFICATION OF ERROR CONTROL AND DETECTION ACCURACY. To obtain representative results, we then run the procedures described in Sec. 6.3 for 100 different response vectors $\mathbf{y}$ generated from different random samples of subjects and different draws of ε. We let the number of samples vary from $n = 100$ (50 random subjects taken among the 796) to $n = 1200$ (600 subjects), the exact number of voxels being $p = 156\,374$. For each simulation, we record the empirical δ-FWER and the δ-precision-recall curves.

BINARY VERSION OF THE SEMI-SIMULATED EXPERIMENT. In the above experiment the response vector $\mathbf{y}$ is continuous, hence we also benchmark the inference procedures for a binary response. For that, we simply take as response vector the signs of the continuous $\mathbf{y}$ generated as in the previous paragraph.

6.4.3 Statistical control under global null with i.i.d. data

In this experiment, we test whether the procedures control the FWER under a global null model. ecd-Lasso only controls the δ-FWER theoretically but, when the true weight vector $\boldsymbol{\beta}^*$ is null, the δ-FWER and the classical FWER are identical. Then, all procedures should control the FWER. Here, we considered the tasks of the HCP S900 task fMRI dataset described in Sec. 6.4.1 keeping all the subjects ($n = 1592$). Then, to get a noise-only response, we (uniformly) randomly permute the original response vector. Similarly as in Sec. 6.4.2, the i.i.d. assumption is legitimate, since the data correspond to z-maps of different subjects. For each task, we draw 100 different permutations of the response and check if the different methods enforce the chosen nominal FWER of 10%.

6.4.4 Statistical control under global null with autocorrelated data

In this experiment, we study how the different procedures control the FWER when the data are temporally autocorrelated; hence violating the i.i.d. assumption. Notably, this is the case if the data correspond to fMRI signal recordings of one given subject during an acquisition. We consider

data from the HCP S900 resting-state fMRI dataset described in Sec. 6.4.1 with full samples ($n = 1200$), the design matrix $\mathbf{X}$ containing the 15-minutes fMRI signal registration. As in Eklund, Nichols, and Knutsson (2016), we construct $\mathbf{y}$ such that it corresponds to two activity paradigms: block or event responses, with several frequencies: 10s on/off, 20s on/off, 30s on/off, 2s-activation/6s-rest, 4s-activation/8s-rest. Thus, $\mathbf{y}$ is temporally autocorrelated. In these simulations $\beta^* = 0$ so the δ-FWER and the classical FWER are identical. To better assess the impact of correlation, we also generate $\mathbf{y}$ as an i.i.d. —uncorrelated— Bernoulli or standard Gaussian random variable (here again $\beta^* = 0$), breaking spurious correlations between $\mathbf{X}$ and $\mathbf{y}$. These two cases enable to check if the procedures still control the FWER at the targeted nominal level on this dataset under the i.i.d. assumption. For each kind of response, we repeat the experiment 100 times, using data from 100 different subjects.

6.4.5 Statistical control of out-of-brain detections

In this experiment we test the four procedures on an unmasked task fMRI dataset to verify that no spurious detection is made outside of the brain —up to the allowed error rate. Indeed, the non-null coefficients of the weight vector β^* should all be contained in the brain since there is no informative signal in out-of-brain voxels. To do so, we take the unmasked RSVP task fMRI dataset, described in Sec. 6.4.1 (with design matrix $\mathbf{X}$ containing $n = 756$ unmasked z-maps). Then, we report how frequently some voxels are detected outside the brain volume.

6.4.6 Insights on the choice of number of clusters

In this experiment, we assess empirically the impact of C, the number of clusters used in the ecd-Lasso algorithm. We use the same generative method as in Sec. 6.4.2 to produce an experiment with known ground truth. Then, we run the ecd-Lasso algorithm varying the numbers of clusters C from $C = 200$ to $C = 1000$. We also vary the number of samples n from 100 to 1200. As in Sec. 6.4.2, we run the experiment for 100 different response vectors and report aggregated results. We report two statistics: the empirical δ-FWER and the AUC of the δ-precision-recall curve for every value of C and n.

6.4.7 Face validity on HCP

In this experiment, we consider the output of the procedures in terms of brain regions that are conditionally associated with the task performed by

the subjects. Similarly as in Sec. 6.4.3, we consider the tasks of the HCP S900 task fMRI dataset described in Sec. 6.4.1, keeping this time the true response vector. We run all the procedures on every task and report the statistical maps thresholded such that the FWER $< 10\%$ or the δ-FWER $< 10\%$ (for ecd-Lasso). For this, we use all the available samples ($n = 1592$).

6.4.8 Prediction performance

Even if it is not the purpose of this study, we also checked the prediction performance of the decoders produced by each method. Since Thr-SVR and Perm-SVR rely on the same predictive function, there are three different decoders: SVR, Ada-SVR and ecd-Lasso. To perform this experiment, we consider the tasks of the HCP S900 task fMRI dataset described in Sec. 6.4.1. We run all the procedures on every task using a sample size $n = 400$, keeping the rest of the samples to test the trained model. For each task and each method, we take 100 different random subsamples to produce the results.

6.5 RESULTS

In this section, after setting the value of the tolerance parameter δ in the different datasets, we present the experimental results.

6.5.1 Estimating δ in HCP and RSVP datasets

In all the experiments, unless specified otherwise, we run ecd-Lasso with the default choice $C = 500$. Reversing (6.9), we obtain a tolerance parameter of $\delta_{\text{HCP}} = 5.4$ voxels for HCP S900 and $\delta_{\text{RSVP}} = 5.6$ voxels for RSVP, corresponding to $\delta_{\text{HCP}} = 12\,\text{mm}$ and $\delta_{\text{RSVP}} = 18\,\text{mm}$ respectively after rounding up.

6.5.2 Statistical control with known ground truth

Here, we describe the results obtained from the experiment described in Sec. 6.4.2.

QUALITATIVE COMPARISON OF THE MODEL SOLUTIONS. In Fig. 6.4, we present a qualitative comparison of the model solutions when $n = 400$. None of the methods yields false discoveries for the chosen threshold —taken such that δ-FWER $< 10\%$. ecd-Lasso recovers more active regions than the other procedures, which makes it the most powerful procedure, followed by Ada-SVR. The other two procedures do not discover the expected patterns.

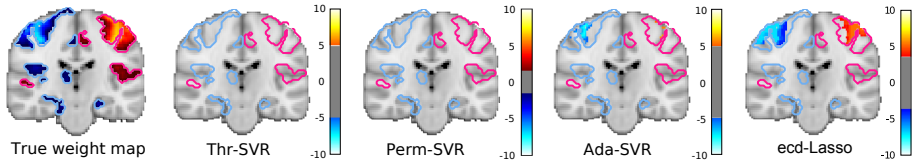


Figure 6.4: Qualitative comparison of the model solutions. Here, we show the solutions (z-maps) given by the four inference procedures, for a single random draw of the noise vector in the experiment described in Sec. 6.4.2. The weight maps are thresholded such that δ-FWER $< 10\%$ theoretically. We can observe that none of the methods yield false discoveries but the Ensemble of Clustered Desparsified Lasso (ecd-Lasso) procedure is the most powerful followed by Adaptive Permutation Threshold SVR (Ada-SVR).

These results displayed are obtained for a single random draw of the noise vector, but similar results holds for different draws.

δ-FWER CONTROL. In this experiment, we check if Thr-SVR, Perm-SVR, Ada-SVR and ecd-Lasso control the δ-FWER at the targeted nominal level (here being 10%). Fig. 6.5 shows that Perm-SVR and ecd-Lasso procedures control the δ-FWER for all sample sizes since their empirical δ-FWER remain below the targeted nominal level, whereas Thr-SVR and Ada-SVR fail to control the δ-FWER in every setting. In particular, the empirical δ-FWER for Ada-SVR is above the targeted nominal level for $n \geqslant 800$. This might occur since the approximation made by **??** is valid only if n remains "sufficiently low" (Gaonkar and Davatzikos, 2012). Also, Thr-SVR fails to control empirically the δ-FWER for any value of n. This might be due to the two assumptions made in Sec. 6.3.2 not being satisfied —it is indeed unlikely that the SVR weights of the null region follow the same distribution. We further discuss this point in Sec. 6.6. We obtain similar results for the binary version of the experiment (see Fig. 6.13).

δ-PRECISION-RECALL. In this experiment, we also evaluate the recovery properties of the four methods by comparing the δ-precision-recall curve for different value of n. Fig. 6.5 shows that ecd-Lasso has the best δ-precision-recall curve for $n = 400$. We recall that the perfect precision-recall curve is reached if the precision is equal to 1 for any value of recall between 0 and 1. Similar results were obtained for the other sample sizes tested (see Fig. 6.12). Indeed, when $n = 400$, for a 90% δ-precision, ecd-Lasso gives a recall of 23% and Ada-SVR a recall of 16%. Thr-SVR and Perm-SVR share the same δ-precison-recall curve since they both produce p-values arranged in the reverse order of the absolute SVR weights. These thresholding methods were not able to reach the 90% δ-precision; their recovery properties are much

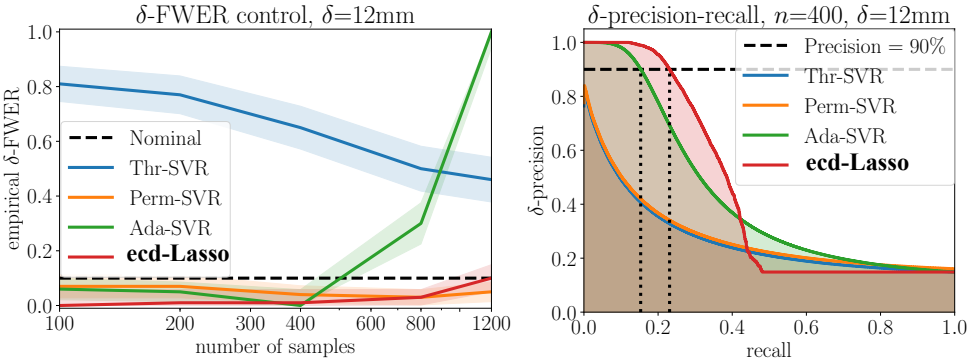


Figure 6.5: δ-FWER control and δ-precision-recall curve on semi-simulated data (known ground truth). Left: The results of the experiment described in Sec. 6.4.2 show that the permutation test (Perm-SVR) and Ensemble of Clustered Desparsified Lasso (ecd-Lasso) are the only procedures that correctly control the δ-FWER at the nominal level (10%). This is not the case for Adaptive Permutation Threshold SVR (Ada-SVR) and Thresholded SVR (Thr-SVR) procedures. Right: For the same experiment, ecd-Lasso has the best performance in terms of δ-precision-recall curve. For $n = 400$, and ensuring 90% δ-precision, ecd-Lasso obtains a recall of 23% and Ada-SVR a recall of 16%. Thr-SVR and Perm-SVR share the same δ-precison-recall curve and were not able to reach 90% δ-precision.

weaker. The binary version of the experiment yields similar conclusions (see Fig. 6.13).

6.5.3 Statistical control under global null with i.i.d. data

FWER CONTROL UNDER GLOBAL NULL (PERMUTED RESPONSE). Here, we summarize the results of the experiment testing control of the FWER in a global null setting (Sec. 6.4.3). Fig. 6.6 shows that, when samples are i.i.d., all the procedures control the FWER, except Thr-SVR. ecd-Lasso is even conservative since the empirical FWER remains at 0 for all the different tasks tested. This result is not surprising since at least two steps of the ecd-Lasso procedure are conservative: the Bonferroni correction and the ensembling of the p-values maps.

FACE VALIDITY (ORIGINAL RESPONSE). Additionally, we run the procedures with the original (not permuted) response vector to check whether the methods can recover predictive patterns; this corresponds to the experiment described Sec. 6.4.7. We plot the results for the two first tasks (emotion and gambling) in Fig. 6.7; see Fig. 6.14 for the five other tasks. Qualitatively, ecd-Lasso recovers the most plausible predictive patterns, Ada-SVR sometimes makes dubious discoveries: patterns are too wide and implausible. The two other methods exhibit a very weak statistical power.

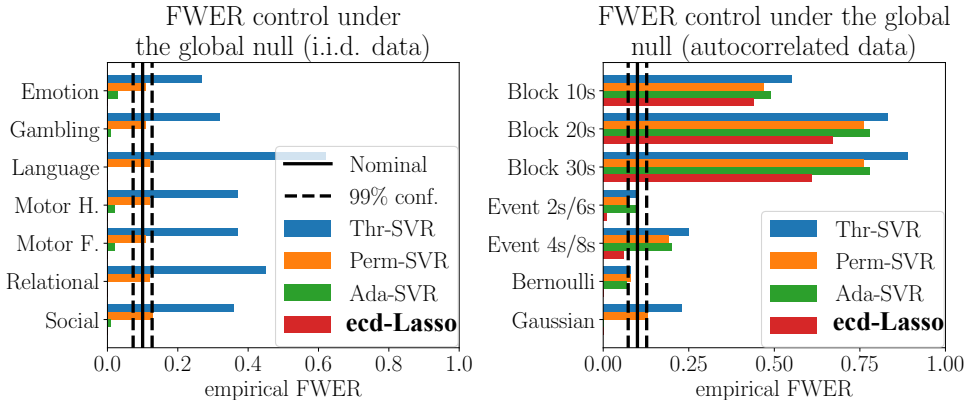
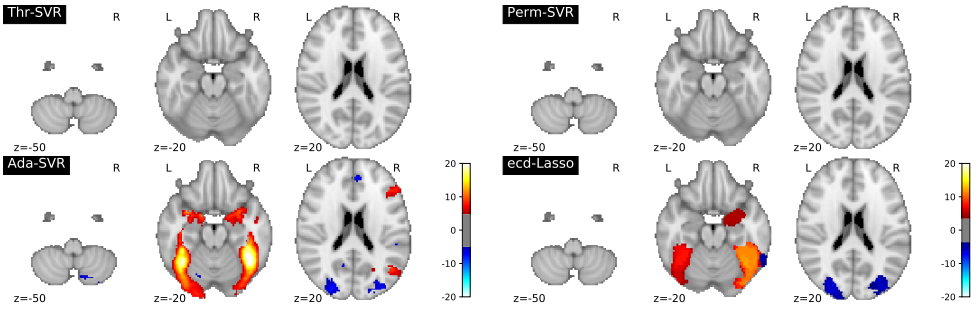


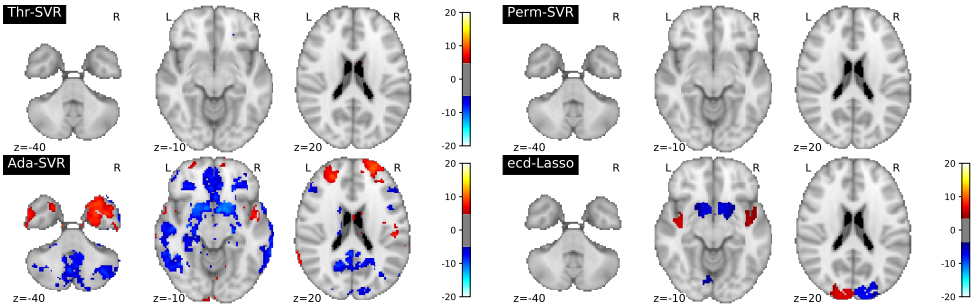
Figure 6.6: FWER control under global null with i.i.d. data vs with autocorrelated data. Left: The results of the experiment with i.i.d. data under global null, described in Sec. 6.4.3, show that, only the Thresholded SVR (Thr-SVR) fails to control the FWER empirically in this context. ecd-Lasso makes no detection: it is a conservative approach, as one could expect from theory. Right: The results of the experiment with correlated data under global null, described in Sec. 6.4.4, show that, when the data are temporally autocorrelated, all the procedures fail to control the FWER. Indeed, for all the fictitious block response paradigms, the empirical FWER exceeds the targeted nominal level of 10% for every procedure. This result is not surprising as the procedures control the δ-FWER under the assumption that the samples are i.i.d.; this is not the case for the block or event response paradigms. However, when the fictitious response breaks the temporal dependency (binary or Gaussian random responses), the i.i.d. assumption is met and the FWER is empirically well controlled except for the Thr-SVR procedure.

6.5.4 Statistical control under global null with autocorrelated data

Here, we report the results of the experiment testing the statistical control properties of the procedures with data correlated across samples, hence violating the i.i.d. assumption (Sec. 6.4.4). In Fig. 6.6 right, we observe that for all the fictitious block response paradigms, for every procedure, the empirical FWER exceeds the targeted nominal level (10%), as one would expect. This result is not surprising since independence across samples is a key assumption for a valid statistical inference with any of the four procedures. Notably, concerning ecd-Lasso, Desparsified Lasso needs the i.i.d. assumption (Zhang and Zhang, 2014; van de Geer et al., 2014) to produce valid confidence intervals or p-values. This assumption is not verified for the block or event response paradigms due to the temporal dependency in the data. However, when the target $\mathbf{y}$ is i.i.d. —*i.e.*, without temporal dependency (Bernoulli or Gaussian random responses)— the FWER is controlled (except for Thr-SVR). Indeed, the model is no longer confounded by the correlation structure underlying the data.



(a) Emotion



(b) Gambling

Figure 6.7: Estimated predictive patterns on standard task fMRI dataset. Here, we plot the results for the emotion and gambling tasks of the experiment described in Sec. 6.4.7 thresholding the statistical maps such that the δ-FWER stays lower than 10% for $\delta = 12$ mm. Qualitatively, ecd-Lasso discovers the most plausible patterns, Ada-SVR sometimes makes dubious discoveries, patterns are too wide and implausible, while the two other methods exhibit a very weak statistical power. The results of the five other tasks are available in Fig. 6.14.

6.5.5 Statistical control of out-of-brain discoveries

We now report the results from the unmasked RSVP task data experiment (Sec. 6.4.5). Here, we check whether out-of-brain detections are made. In Fig. 6.8, the z-score maps are thresholded such that the FWER (for Perm-SVR, Thr-SVR, and Ada-SVR) or the δ-FWER (for ecd-Lasso) are at most 10% for $\delta = 6$ voxels (or 18 mm). We observe that Ada-SVR makes some out-of-brain discoveries, and it does not control the FWER empirically. Thr-SVR and Perm-SVR do not yield spurious detections but very few detections are made, hence these methods have low statistical power. ecd-Lasso does not make any out-of-brain detections and it outlines predictive regions in the temporal lobe and Broca's area, expected for a reading task.

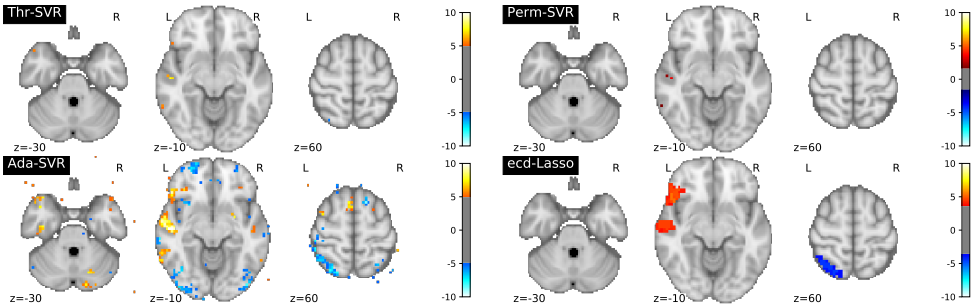


Figure 6.8: Statistical maps for unmasked RVSP data. The results of the unmasked task-fMRI experiment, described in Sec. 6.4.5, show that ecd-Lasso, Thresholded SVR (Thr-SVR) and the permutation test (Perm-SVR) do not return out-of-brain discoveries, while the Adaptive Permutation Threshold SVR (Ada-SVR) does. Here z-score maps are thresholded such that the δ-FWER is at most 10% for $\delta = 6$ voxels (or 18 mm). Thr-SVR and the Perm-SVR do not yield spurious detections but very few detections are made, hence these method have low statistical power. ecd-Lasso does not make any spurious detection; rather it makes detections in the temporal lobe and Broca's area, which are expected for a reading task.

6.5.6 Insights on choosing the number of clusters

Here, we report the results obtained of the experiment task-fMRI data (Sec. 6.4.6) studying the impact of C (number of clusters) on the δ-FWER control and the recovery properties of ecd-Lasso for various sample sizes. These results are obtained with 100 repetitions for every sample and cluster sizes. In Fig. 6.9, we notice that a lower C leads to improved recovery, according to the area under the δ-precision-recall curves. However, when the number of clusters is lower, the average cluster radius increases and overcomes the spatial tolerance of δ, leading to inflated error rates. More precisely, for $\delta = 6$ voxels (or 12 mm), the δ-FWER is controlled only when $C \geqslant 500$. However, for $C < 500$, it is possible to control the δ-FWER, provided a larger spatial tolerance $\delta > 6$ voxels. To compute the requested δ, one can use (6.9). Besides, we observe that the recovery score of ecd-Lasso improves when n increases, as expected. We also notice that the empirical δ-FWER increases with n. To explain this effect, we first recall that theoretically the δ-FWER is controlled for δ equal to the largest cluster diameter, likely to be too large in practice. In this study, we have taken δ equal to the average radius of the clusters, since in practice this choice ensures the δ-FWER control. However, when the setting is particularly favorable for inference (*e.g.*, if $\log(n)/C > 1.5 \times 10^{-2}$), some false discoveries can be made at a distance greater than the average radius from the support. The choice of δ is further discussed in Sec. 6.3.5.

Additionally, we can notice from Fig. 6.9 and Fig. 6.10 that for a fixed C/n ratio the recovery capability is almost stable. We proposed $C = 500$ as a

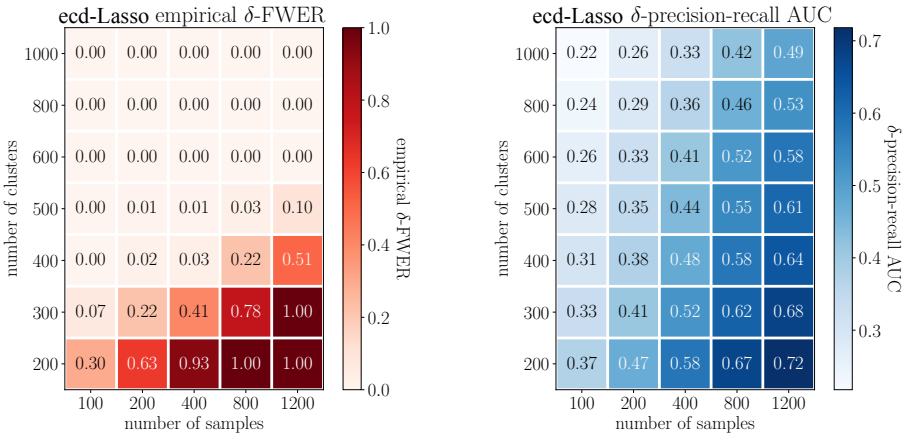


Figure 6.9: Influence of the number C of clusters on δ-FWER control and the recovery properties of ecd-Lasso. The results of the experiment described in Sec. 6.4.6 show the impact of C on the δ-FWER control and the recovery score of ecd-Lasso. When $C \geqslant 500$, clusters are smaller, hence the δ-FWER is controlled for $\delta = 12\,\text{mm}$ (and potentially lower values of δ) since all the empirical δ-FWER's are lower than the 10% nominal rate. Conversely, when $C < 500$, clusters are wider and the spatial tolerance is overcome by the model inaccuracy, hence the δ-FWER is not controlled for $\delta = 12\,\text{mm}$. However, it remains controlled for higher values of δ. Concerning the recovery properties we see that reducing the number of clusters improves the δ-precision-recall curves. Thus, the more spatial uncertainty is tolerated, the best recovery properties ecd-Lasso offers.

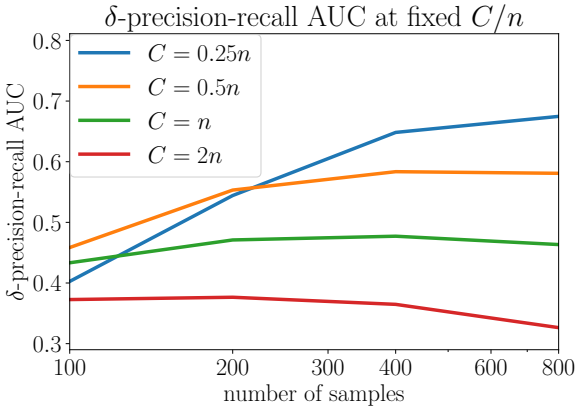


Figure 6.10: Influence of the C/n ratio on the δ-precision-recall AUC. The results of the experiment described in Sec. 6.4.6 show that the δ-precision-recall AUC depends almost linearly on $\log(n/C)$ except when C is critically low creating very wide clusters and deteriorating the precision-recall curve. This limit depends on the physical properties of the problem; here, C should not be lower than 50. Keeping this limit in mind, we advise to take $C \in [n/2, n]$ to recover most of the predictive regions.

default choice in Sec. 6.3.5, yet intuitively, C should adapt to the the amount of data available: larger samples size lead to better estimation, allowing refined localization, hence higher C. Then, as discussed in Sec. 6.3.5, we advise to take C of the same order as p (*e.g.*, $C \in [p/2, p]$) when the goal is to recover most of the predictive regions without strong requirements on the accuracy of their shapes —since the value of δ given by (6.9) might be not small with regards to the predictive region itself.

6.5.7 Prediction performance

In this section, we give results with respect to the prediction performance of the methods. In Fig. 6.11, we plot the results of the experiment described in Sec. 6.4.8. We notice that the classification error rate is almost the same for SVR (the weight map of Thr-SVR and Perm-SVR) and Ada-SVR, their prediction performance is slightly better than the one of ecd-Lasso. Hence we do not recommend to use ecd-Lasso to achieve state-of-the art prediction accuracy, but only for statistical inference purpose.

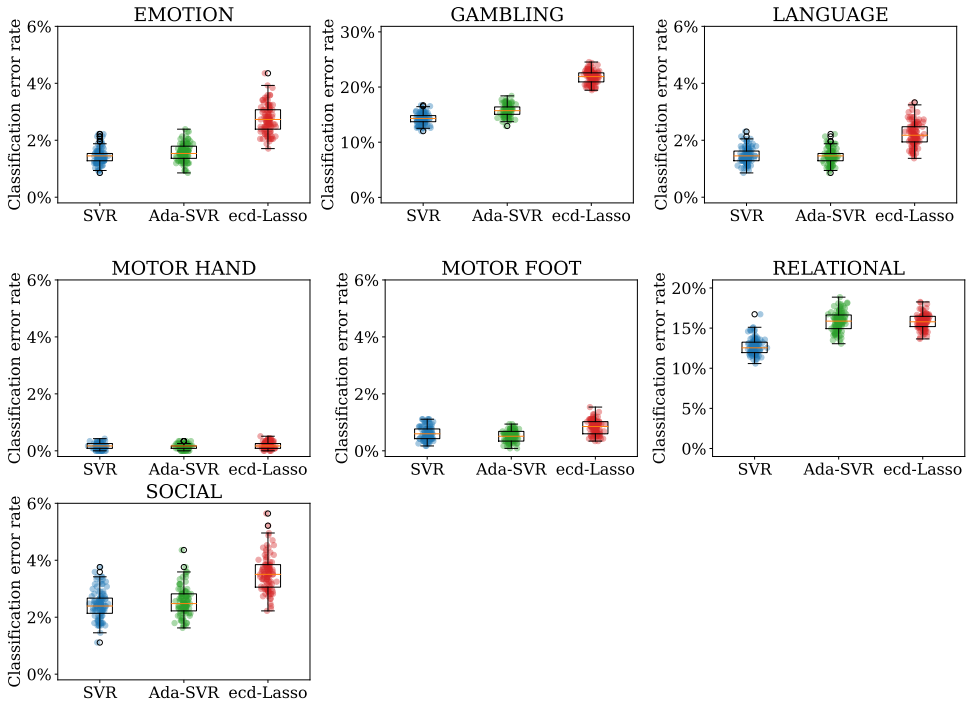


Figure 6.11: Prediction performance. Here we plot the results for the experiment described in Sec. 6.4.8. The classification error rate is almost the same for SVR and Ada-SVR. Their prediction performance is slightly better than the one of ecd-Lasso. Hence we do not recommend to use ecd-Lasso to achieve state-of-the art prediction accuracy, but only for statistical inference purpose. For all the task, "chance" classification error rate is 50%.

6.6 DISCUSSION

Decoding models are fundamental for causal interpretation of the implication of brain regions for an outcome of interest, mental process or disease status (Weichwald et al., 2015). They produce weight maps that are needed to support this type of inference (Poldrack, 2011; Varoquaux et al., 2018). These weight maps capture how brain regions relate to the outcome, *conditional on* the other regions, which is a key difference with respect to standard brain mapping based on mass univariate models. However, the weight maps produced by the common decoders come without good statistical properties. Indeed, decoders optimize the quality of their prediction, but give no control on conditional feature importance. This is difficult due to the large number of covariates —voxels— as well as the severe multi-collinearity: voxel-level inference is untenable. On the other hand, given the spatial structure of the data, a spatial tolerance in the statistical control is natural, as in Gaussian random field theory used in standard analysis (Nichols, 2012).

In this chapter, we leveraged on the spatial statistical control introduced in Chapter 5 called δ-FWER control, a control of false discoveries up to a spatial slack δ. This definition uncovers a fundamental trade-off between accuracy in the localization of the brain structures involved and statistical power: here we deliberately degrade spatial accuracy, acknowledging current concerns on statistical power in neuroimaging studies (Button et al., 2013; Noble, Scheinost, and Constable, 2019).

Thanks to this, we performed an empirical study of the statistical control of four procedures computing decoding maps, ranging from thresholding procedures applied to SVR weights, to a dedicated decoding procedure, ecd-Lasso. Experiments show that the Thr-SVR procedure, thresholding SVR weights, fails to achieve useful statistical control. Exact permutation testing yields the expected statistical control but with very poor statistical power for all experimental settings we have studied. On the other hand, Adaptive Permutation Threshold SVR (Ada-SVR) (Gaonkar and Davatzikos, 2012), does not control the FWER as it should, though it exhibits a fair δ-precision-recall curve in our semi-simulated experiments. This shows how difficult it is to identify a statistically valid threshold for SVR weight maps. This is due to the fact that under the null hypothesis, estimated weights are not distributed according to a fixed distribution —notably because of the dependency structure of the data— and more precisely, the variance of these distributions differs. Then, thresholding linear decoders (SVR, logistic regression) based on their estimated weights amplitudes is not a principled approach to control false discoveries.

ecd-Lasso uses a different decoding procedure to estimate the weight maps (Chevalier, Salmon, and Thirion, 2018), and as a result comes with theoretical statistical guarantees: it controls the δ-FWER for a predetermined

tolerance parameter δ equal to the largest diameter of the clusters, assuming that the observed samples are i.i.d. and that the weight maps are homogeneous and sparse. The experiments show that, indeed, for i.i.d. scenarios, ecd-Lasso controls the δ-FWER for δ equal to the average radius of the clusters. Though, in some very high SNR or high sample size regimes, it might be necessary to take δ larger than the average radius (see Sec. 6.3.5). In practice, our choice of δ is conservative, and with current fMRI datasets, δ-FWER control holds for smaller δ, even in relatively large cohorts ($n = 1200$). In addition, ecd-Lasso exhibits the best support recovery performance in the proposed semi-simulated experiments with fMRI data but also finds patterns with good face validity in more qualitative experiments plotted in Fig. 6.7. On the other hand, we also notice that ecd-Lasso tends to be over-conservative.

Although it is not the main purpose of this study, we also checked the prediction performance of the decoders produced by each method. It is important to note that ecd-Lasso has been designed for the recovery of conditional statistical associations, not for prediction. In practice, the prediction performance is almost the same for SVR and Ada-SVR, and is slightly better than the one of ecd-Lasso (see Fig. 6.11). For prediction purpose, we recommend using *Fast Regularized Ensembles of Models* (FReM) (Hoyos-Idrobo et al., 2018), which is a stable and computationally efficient decoder with state of the art prediction performance.

For pedagogical purpose, we have also considered a dataset where cross-sample independence is violated due to serial correlation, reproducing an experiment of Eklund, Nichols, and Knutsson (2016). The ensuing loss of statistical control underlines the importance of the i.i.d. assumption. Hence, ecd-Lasso should not be used to make inference from intra-subject dataset recorded over one session. With these warnings in mind, we think that ecd-Lasso can be used safely in neuroimaging context. Our code, implemented with Python 3, can be found on `https://github.com/ja-che/hidimstat` along with some examples.

We have not considered the method proposed by Nguyen, Chevalier, and Thirion (2019) based on the Knockoff filters (Barber and Candès, 2015; Candès et al., 2018) that yet appear to be an appealing procedure, as it can only control the FDR. In this study we have focused on δ-FWER control, and hence defer the analysis of FDR-controlling procedures to future work. Also, we have not benchmarked post-selection inference procedures (Berk et al., 2013; Lee et al., 2016), as we found them challenging to run in high dimensional settings and prone to numerical underflows.

Our empirical results clearly show that standard thresholding procedures, including classical permutation tests, are not reliable to infer regions importance on decoder maps, due to the high number of covariates. Since, in

neuroimaging studies, these maps are used to give evidence on the brain regions that supports an outcome, it is crucial to use a procedure with statistical control on the brain maps. Our study shows that ecd-Lasso provides such a control.

6.7 SUPPLEMENTARY MATERIAL

6.7.1 Statistical control with known ground truth: additional plots

In this section, we provide additional experimental results to assess the detection accuracy of the multivariate estimators, to complement the results in Sec. 6.4.2. Fig. 6.12 shows additional $\delta-$precision-recall curves, obtained for different values of n: these different settings preserve the relative performance of the methods, while larger n results in better curves. Fig. 6.13

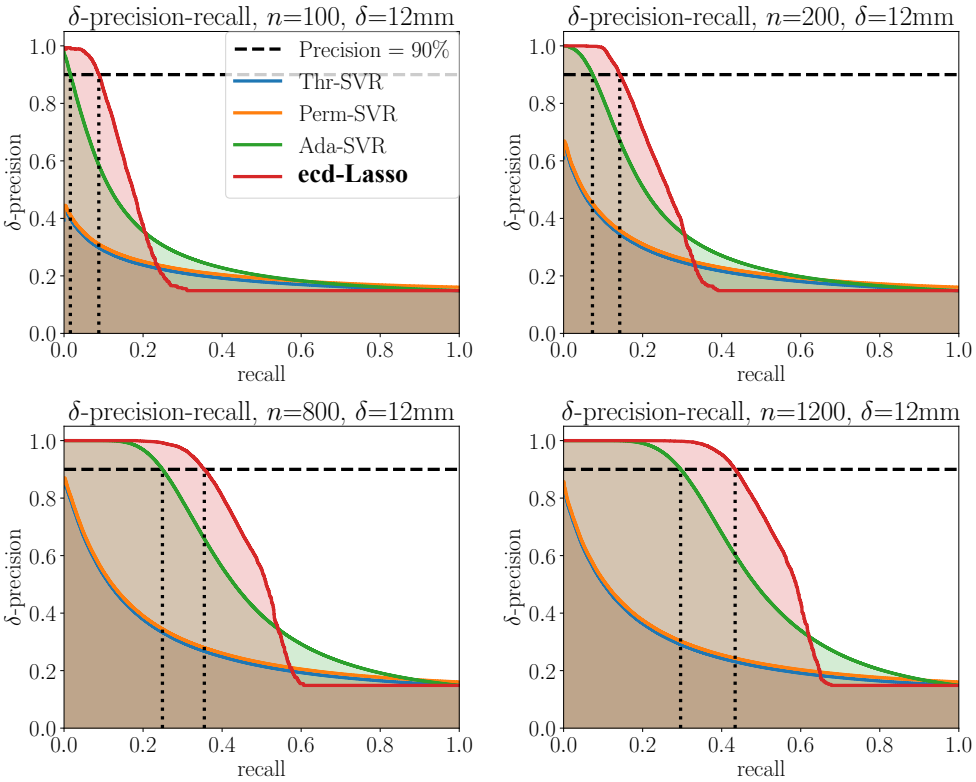


Figure 6.12: δ-precision-recall curves on semi-simulated data with continuous response vector. The results of the experiment described in Sec. 6.4.2 show that ecd-Lasso has the best performance in terms of δ-precision-recall curve; these results are similar to the one observed Fig. 6.5.

displays the performance of the methods in terms of δ-FWER control and

δ-precision-recall curves on semi-simulated data where **y** is binary. This induces a violation of the ecd-Lasso model that reduces its performance in terms of δ precision-recall. Yet, unlike Ada-SVR, it still controls the δ-FWER accurately.

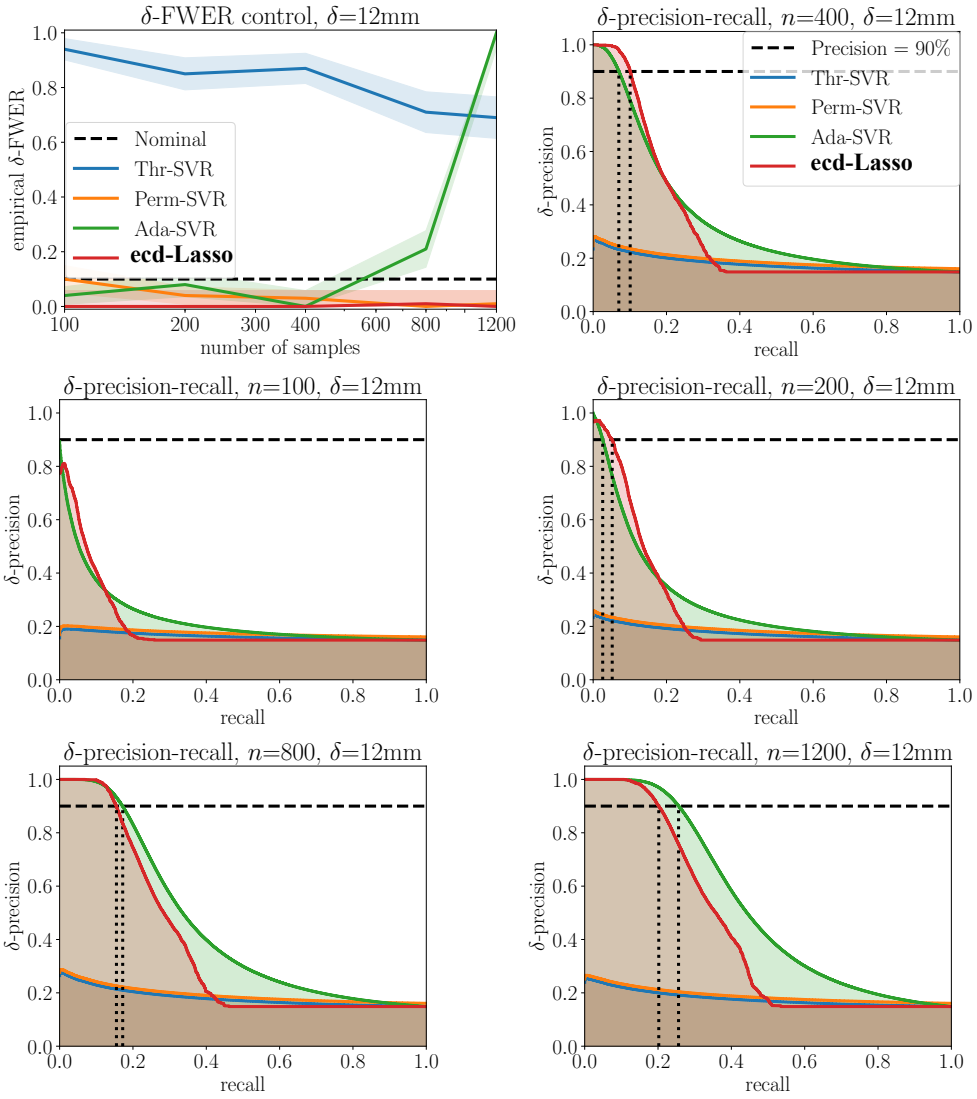
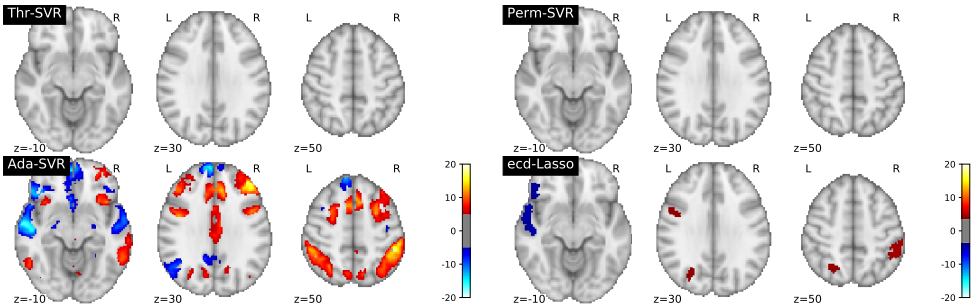


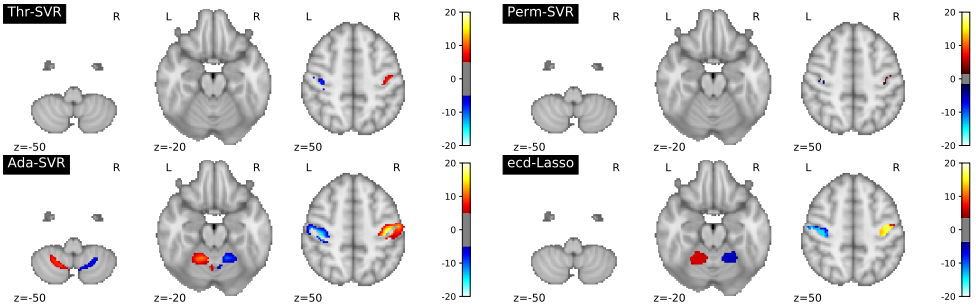
Figure 6.13: δ-FWER control and δ-precision-recall curves on semi-simulated data with binary response vector. The results of the experiment described in Sec. 6.4.2 with binary response show that the permutation test (Perm-SVR) and ecd-Lasso are the only procedures that empirically control the δ-FWER. In terms of statistical power, Ada-SVR and ecd-Lasso have the best δ-precision-recall curve. These results are quite similar to the one presented in Fig. 6.6.

6.7.2 Face validity on HCP dataset: additional plots

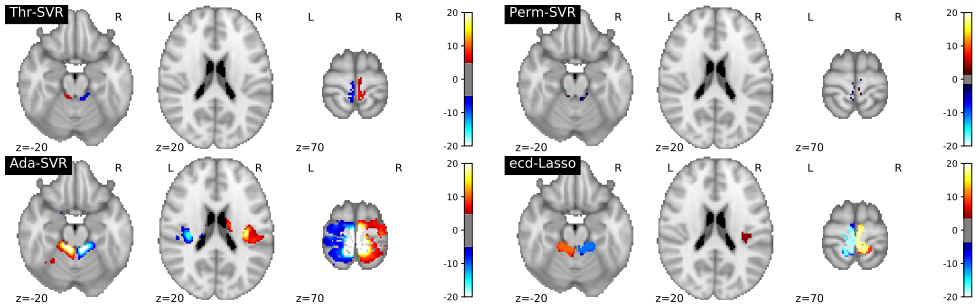
In Fig. 6.14, we plot the results for five tasks taken from the HCP dataset, besides of the two described in Sec. 6.4.7. For all methods, the statistical maps are thresholded such that the δ-FWER stays lower than 10% for $\delta = 12\,\text{mm}$. Qualitatively, ecd-Lasso discovers the most plausible patterns, Ada-SVR often makes dubious discoveries, patterns are too wide and implausible, while the two other methods exhibit a very weak statistical power.



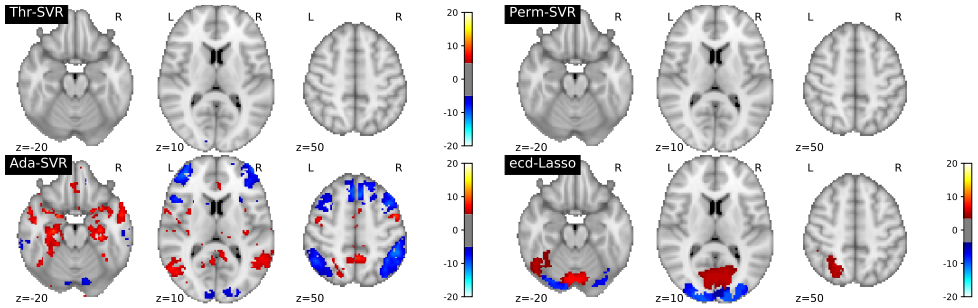
(c) Language



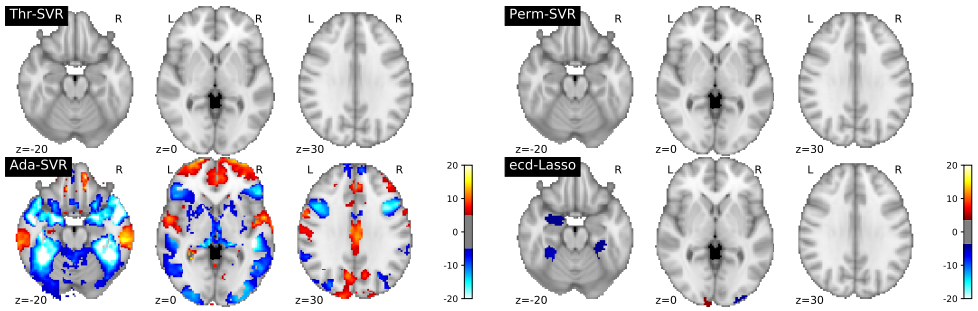
(d) Motor Hand



(e) Motor Foot



(f) Relational



(g) Social

Figure 6.14: Estimated predictive patterns on standard task fMRI dataset. Here, we plot the results for five tasks of the experiment described in Sec. 6.4.7 thresholding the statistical maps such that the δ-FWER stays lower than 10% for $\delta = 12\,\text{mm}$. Qualitatively, ecd-Lasso discovers the most plausible patterns, Ada-SVR often makes dubious discoveries, patterns are too wide and implausible, while the two other methods exhibit a very weak statistical power. The results of emotion and gambling tasks are available in Fig. 6.7.

7 EXTENSION TO TEMPORAL DATA WITH APPLICATIONS TO MEG

In this chapter, we extend our work to the magnetoencephalography (MEG) and electroencephalography (EEG) source localization setup. This chapter mainly present our work (Chevalier et al., 2020) accepted at the 2020 *NeuRIPS* conference:

CHEVALIER, Jérôme-Alexis, GRAMFORT, Alexandre, SALMON, Joseph, et al. Statistical control for spatio-temporal MEG/EEG source imaging with desparsified multi-task Lasso. In: Advances in Neural Information Processing Systems, 2020.

M/EEG source imaging requires to work with spatio-temporal data and autocorrelated noise. To deal with this, we adapt the d-Lasso estimator to temporal data corrupted with autocorrelated noise by leveraging on debiased group Lasso estimators and introducing the desparsified multi-task Lasso (d-MTLasso). We combine d-MTLasso with spatially constrained clustering to reduce data dimension and with ensembling to mitigate the arbitrary choice of clustering; the resulting estimator is called ensemble of clustered desparsified multi-task Lasso (ecd-MTLasso). With respect to the current procedures, the two advantages of ecd-MTLasso are that *i)*it offers statistical guarantees and *ii)* it trades spatial specificity for sensitivity, leading to a powerful adaptive method. Extensive simulations on realistic head geometries, as well as empirical results on various MEG datasets, demonstrate the high recovery performance of ecd-MTLasso and its primary practical benefit: offer a statistically principled way to threshold MEG/EEG source maps.

7.1 INTRODUCTION

Source imaging with magnetoencephalography (MEG) and electroencephalography (EEG) delivers insights into brain activity with high temporal and good spatial resolution in a non-invasive way (Baillet, Mosher, and Leahy, 2001). It however requires to solve the bioelectromagnetic inverse problem, which is a high-dimensional ill-posed regression problem. Various approaches have been proposed to regularize the estimation of the regression coefficients that map activity to brain locations. Historically, ℓ_2 regularization was considered first (Hämäläinen and Ilmoniemi, 1994), with successive improvements known as dSPM (Dale et al., 2000) and sLORETA (Pascual-

Marqui, 2002) that are referred to as "noise normalized" solutions. The reason is that the coefficients are standardized with an estimate of the noise standard deviation, producing outputs that are comparable to T or F statistics, yet not statistically calibrated. These latter techniques have since become standard when using ℓ_2 approaches.

More recently, alternative approaches based on sparsity assumptions have been proposed with the ambition to improve the spatial specificity of M/EEG source imaging (Gramfort, Kowalski, and Hämäläinen, 2012; Haufe et al., 2009; Lucka et al., 2012; Matsuura and Okabe, 1995; Wipf and Nagarajan, 2009). The output of such methods consists of focal sources as opposed to blurred images obtained with ℓ_2 regularization. However, obtaining statistics ("noise normalized") from sparse or non-linear estimators seems challenging, especially since M/EEG data are spatio-temporal data with complex noise structure. A natural way to deal with the temporal dimension is to consider a multi-task estimator and structured sparse priors based on ℓ_1/ℓ_2 mixed norms (Gramfort, Kowalski, and Hämäläinen, 2012; Ou, Hämaläinen, and Golland, 2009).

In the statistical literature, some attempts to obtain an estimate of both regression coefficients and their variance have been proposed for linear models in high dimension (Bühlmann, 2013; Meinshausen, Meier, and Bühlmann, 2009; Wasserman and Roeder, 2009). These estimates can then be translated to p-value maps, *i.e.*, maps of p-values associated with each covariate. Some methods adapted for sparse scenarios have then proposed to debias the Lasso to obtain p-values or confidence intervals (Javanmard and Montanari, 2014; Zhang and Zhang, 2014; van de Geer et al., 2014). We refer to such variants as desparsified Lasso. Recently, desparsified extensions of group Lasso have also been considered (Mitra and Zhang, 2016; Stucky and van de Geer, 2018). However, all these previous methods generally lack of power when $p \gg n$. Here, we propose to address a multi-task setting in the presence of correlated noise, and to deal with high-dimensional when $p \gg n$ leveraging on data structure as done by Chevalier, Salmon, and Thirion (2018). All these challenges need to be considered for M/EEG source imaging.

Our first contribution is to propose the desparsified multi-task Lasso (d-MTLasso), an adaptation of the desparsified Lasso (d-Lasso) (Zhang and Zhang, 2014; van de Geer et al., 2014) to multi-task setting (Obozinski, Taskar, and Jordan, 2010). This contribution is complementary to Mitra and Zhang (2016), since taking the multi-task approach allows for *i)*a simple statistic test formula with *ii)*a natural integration of auto-correlated noise and *iii)*a simplification of mathematical assumptions since they reduce mainly to the Restricted Eigenvalue (RE) assumption (cf. Sec. 7.2.4). Our second contribution is to introduce ensemble of clustered desparsified multi-task Lasso (ecd-MTLasso), which has two advantages compared to current methods: *i)*it offers statistical guarantees and *ii)*it trades spatial specificity for sensi-

tivity, leading to a powerful adaptive method. Our third contribution is an empirical validation of the theoretical claims. In particular, we run extensive simulations on realistic head geometries, as well as empirical results on various MEG datasets to demonstrate the high recovery performance of ecd-MTLasso and its primary practical benefit: offer a statistically principled way to threshold MEG/EEG source maps.

7.2 THEORETICAL BACKGROUND

In this section, we give the noise model in the multi-task setting, we provide standard tools for solving the source localization problem and, mainly, we present three new methods with their assumptions and statistical guarantees.

7.2.1 Model and notation

Similarly as in the previous chapters (*e.g.*, Chapter 4), we assume that the underlying model is linear. However, in M/EEG the treatment of the time dimension leads to a multi-task setting:

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{E} \ , \tag{7.1}$$

where $\mathbf{Y} \in \mathbb{R}^{n \times T}$ is the signal observed on M/EEG sensors, $\mathbf{X} \in \mathbb{R}^{n \times p}$ the design matrix representing the M/EEG forward model, $\mathbf{B} \in \mathbb{R}^{p \times T}$ the underlying signal in source space and $\mathbf{E} \in \mathbb{R}^{n \times T}$ the noise. We assume that there exist $\rho \in [0, 1)$ and $\sigma > 0$ such that all $t \in [T]$, $\mathbf{E}_{\cdot,t} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_n)$ and that for all $i \in [n]$ and all $t \in [T-1]$, $\mathrm{Cor}(\mathbf{E}_{i,t}, \mathbf{E}_{i,t+1}) = \rho$. For all $i \in [n]$, $\mathbf{E}_{i,\cdot}$ is Gaussian with Toeplitz covariance, *i.e.*, defining $\mathbf{M} \in \mathbb{R}^{T \times T}$ by $\mathbf{M}_{t,u} = \sigma^2 \rho^{|t-u|}$ for all $(t, u) \in [T]^2$, we have:

$$\mathbf{E}_{i,\cdot} \sim \mathcal{N}(0, \mathbf{M}) \ . \tag{7.2}$$

We further assume that $\mathbf{X}$ has been column-wise standardized and denote by $\hat{\mathbf{\Sigma}} \in \mathbb{R}^{p \times p}$ the empirical covariance matrix of $\mathbf{X}$, *i.e.*, $\hat{\mathbf{\Sigma}} = \mathbf{X}^\top \mathbf{X}/n$ with $\hat{\mathbf{\Sigma}}_{j,j} = 1$. Finally, the (row) support of $\mathbf{B}$ is defined by $\mathrm{Supp}(\mathbf{B}) = \{j \in [p] : \mathbf{B}_{j,\cdot} \neq 0\}$. All proofs are given in Sec. 7.5.

7.2.2 Metrics for statistical inference in M/EEG

To quantify the ability of a M/EEG source imaging technique to obtain a good estimated $\hat{\mathbf{B}}$, a commonly reported quantity is the Peak Localization Error (PLE) (Hauk, Wakeman, and Henson, 2011). It consists in measuring the distance (in mm) along the cortical surface between the true simulated source and the location with maximum amplitude in the estimator. By

contrast, spatial dispersion (SD) measures how much the activity is spread out by the inverse method (Molins et al., 2008).

To quantify the control of statistical errors, we consider a generalization of the Family Wise Error Rate (FWER) (Hochberg and Tamhane, 1987b): the δ-FWER. As illustrated in Fig. 7.1, it is the FWER taken with respect to a ground truth dilated spatially by an amount δ —in the present study a distance in mm. A rigorous definition of δ-FWER is given in Chapter 5 and a more practical approach is proposed in Chapter 6. The rationale is that detections made outside of the support, but less than δ away from the support should count as slight inaccuracies of the methods, not as false positives. In an analogous manner, δ-FDR $=$ $(1 -$ δ-precision$)$ has been proposed recently as an extension of the False Discovery Rate (FDR) (Benjamini and Hochberg, 1995) to include a spatial tolerance (Gimenez and Zou, 2019; Nguyen, Chevalier, and Thirion, 2019). We thus characterize the selection capabilities of the methods through a δ-precision/recall curve.

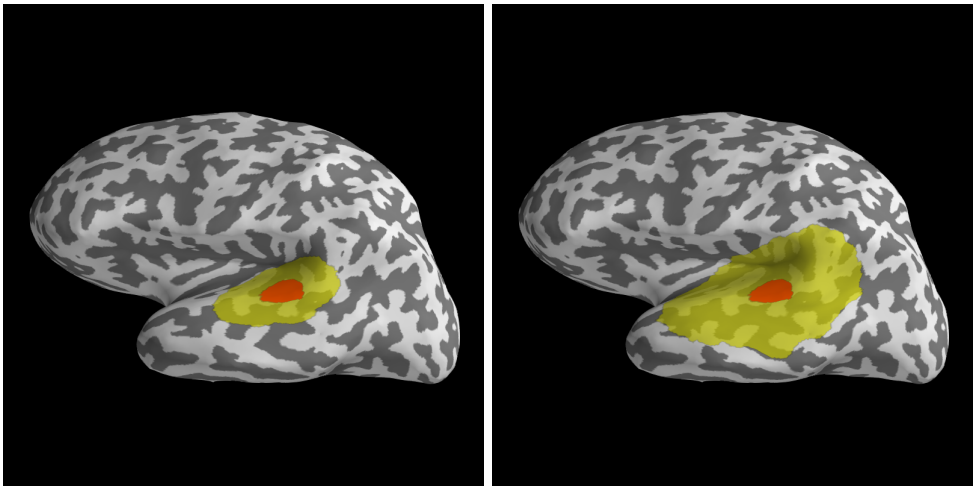


Figure 7.1: Illustrating spatial tolerance of size δ = 20 mm and δ = 40 mm. The true source in red has a 10 mm radius (distance measured on the cortical surface) and the spatial tolerance extend this region by 20 mm on the left side and 40 mm on the right side in yellow. The δ-FWER is the probability of making false discoveries outside of the extended region. Then, a false discovery made in the yellow region is not counted neither as an error nor a true positive.

7.2.3 Classical Solutions

The sLORETA and dSPM estimators are derived from the ridge estimator (Hoerl and Kennard, 1970):

$$\hat{\mathbf{B}}^{\text{Ridge}} = \mathbf{KY} \quad \text{where} \quad \mathbf{K} = \mathbf{X}^\top (\mathbf{X}\mathbf{X}^\top + \lambda \mathbf{I})^{-1} \ . \tag{7.3}$$

They are obtained by scaling each row j in $\hat{\mathbf{B}}^{\text{Ridge}}$ by an estimate of the noise level at location j. It reads (Lin et al., 2006) $\hat{\mathbf{B}}^{\text{dSPM}}_{j,t} = \hat{\mathbf{B}}^{\text{Ridge}}_{j,t} / \sigma^{\text{dSPM}}_j$ and $\hat{\mathbf{B}}^{\text{sLORETA}}_{j,t} = \hat{\mathbf{B}}^{\text{Ridge}}_{j,t} / \sigma^{\text{sLORETA}}_j$, where $\sigma^{\text{dSPM}}_j = \sqrt{\sigma^2 [\mathbf{K}\mathbf{K}^\top]_{j,j}}$ and $\sigma^{\text{sLORETA}}_j = \sqrt{[\mathbf{K}(\sigma^2 \mathbf{I} + \mathbf{X}\mathbf{X}^\top)\mathbf{K}^\top]_{j,j}}$. Interestingly, it can be proved that in the absence of noise and when only a single coefficient is non-zero, the sLORETA estimate has its maximum at the correct location (Pascual-Marqui, 2002). Assuming $\mathbf{B}_{\cdot,t} \sim \mathcal{N}(0, \mathbf{I})$, the covariance of $\mathbf{Y}$ reads $\sigma^2 \mathbf{I} + \mathbf{X}\mathbf{X}^\top$. Hence, one can consider that sLORETA adds to dSPM an extra term in the sensor covariance matrix that comes from the sources. Note that these methods treat each time instant independently, hence ignoring source and noise temporal autocorrelations.

7.2.4 Desparsified multi-task Lasso (d-MTLasso)

Let us first recall the definition of the multi-task Lasso (MTLasso) estimator (Obozinski, Taskar, and Jordan, 2010) in our setting. For a tuning parameter[1] $\lambda > 0$, it is defined as

$$\hat{\mathbf{B}}^{\text{MTL}} \in \underset{\mathbf{B} \in \mathbb{R}^{p \times T}}{\operatorname{argmin}} \left\{ \frac{1}{2n} \|\mathbf{Y} - \mathbf{X}\mathbf{B}\|^2 + \lambda \|\mathbf{B}\|_{2,1} \right\} \quad . \tag{7.4}$$

It is well known that similarly to the Lasso, MTLasso is biased: it tends to shrink rows with large amplitude towards zero. Below, we provide an adaptation of the Desparsified Lasso following the approach by Zhang and Zhang (2014), see also Mitra and Zhang, 2016, to ensure statistical control. The approach relies on the introduction of score vectors $\mathbf{z}_1, \dots, \mathbf{z}_p$ in $\mathbb{R}^n$ defined by

$$\mathbf{z}_j = \mathbf{X}_{\cdot,j} - \mathbf{X}^{(-j)} \hat{\boldsymbol{\beta}}^{(-j)}_{\alpha_j} \quad , \tag{7.5}$$

where, for $j \in [p]$, $\hat{\boldsymbol{\beta}}^{(-j)}_{\alpha_j}$ is the Lasso solution (Chen and Donoho (1994) and Tibshirani (1996)) of the regression of $\mathbf{X}_{\cdot,j}$ against $\mathbf{X}^{(-j)}$ with regularization parameter[2] α_j. Note that these score vectors are independent of $\mathbf{Y}$ and their computation is then equivalent to solving the node-wise Lasso (Meinshausen and Bühlmann, 2006). For such vectors, the noise model in (7.1) yields

$$\frac{\mathbf{z}_j^\top \mathbf{Y}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} = \mathbf{B}_{j,\cdot} + \frac{\mathbf{z}_j^\top \mathbf{E}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} + \sum_{k \neq j} \frac{\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} \mathbf{B}_{k,\cdot}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} \quad . \tag{7.6}$$

1 λ is set by cross-validation on a logarithmic grid going from $\frac{\lambda_{\max}}{100}$ to $\lambda_{\max}$, where $\lambda_{\max} = \|\mathbf{X}^\top \mathbf{Y}\|_{2,\infty}$.

2 In (Zhang and Zhang, 2014, Table 1) an algorithm for choosing α_j is proposed. We noticed that taking for all $j \in [p]$, $\alpha_j = c\alpha_{\max,j} := c\|\mathbf{X}^{(-j)}\mathbf{X}_{\cdot,j}\|_\infty / n$ with $c = 0.5\%$ for M/EEG data leads to a significant computation gain and yields adequate residuals for $C = 1000$ (see Sec. 7.2.6).

Discarding the noise term and plugging $\hat{\mathbf{B}}^{\text{MTL}}_{k,\cdot}$ as a preliminary estimator of $\mathbf{B}_{k,\cdot}$ in (7.6), we coin the desparsified multi-task Lasso (d-MTLasso), a debiased estimator of $\hat{\mathbf{B}}^{\text{MTL}}$ defined for all $j \in [p]$ by

$$\hat{\mathbf{B}}^{(\text{d-MTLasso})}_{j,\cdot} = \frac{\mathbf{z}_j^\top \mathbf{Y}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} - \sum_{k \neq j} \frac{\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} \hat{\mathbf{B}}^{\text{MTL}}_{k,\cdot}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} \quad . \tag{7.7}$$

To derive d-MTLasso statistical properties, we need the extended Restricted Eigenvalue (RE) property (Lounici et al., 2011, Assumption 3.1), detailed in Sec. 7.5.2. More precisely, we assume that

(A1) RE($\mathbf{X}, s$) is verified on $\mathbf{X}$ for a sparsity parameter $s \geqslant |\text{Supp}(\mathbf{B})|$ and a constant $\kappa = \kappa(s) > 0$.

Roughly, A1 can be seen as a combination of sparsity and "moderate" feature correlation assumptions.

Proposition 7.2.1. *Considering the model in (7.1), assuming A1 and for a choice of λ large enough[3] in (7.4), then with high probability:*

$$\sqrt{n}(\hat{\mathbf{B}}^{(\textit{d-MTLasso})} - \mathbf{B}) = \mathbf{\Lambda} + \mathbf{\Delta} \ , \tag{7.8}$$

$$\mathbf{\Lambda}_{j,\cdot} \sim \mathcal{N}_p(0, \hat{\mathbf{\Omega}}_{j,j}\mathbf{M}), \textit{ for all } j \in [p], \textit{ where } \hat{\mathbf{\Omega}}_{j,k} = \frac{n\mathbf{z}_j^\top \mathbf{z}_k}{|\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}||\mathbf{z}_k^\top \mathbf{X}_{\cdot,k}|}$$

$$\|\mathbf{\Delta}\|_{2,1} = O\left(\frac{s\lambda}{\sqrt{n}\kappa^2}\right) \tag{7.9}$$

Then, under the j-th null hypothesis $H_0^{(j)}$: "$\mathbf{B}_{j,\cdot} = 0$" and neglecting the term $\mathbf{\Delta}$ (see Sec. 7.5.4 for more details) in (7.8) as done by van de Geer et al. (2014), $\hat{\mathbf{B}}^{(\text{d-MTLasso})}_{j,\cdot}$ is Gaussian with zero-mean. Finally, using standard results on χ^2 distributions (see Sec. 7.5.3), we obtain

$$n \left\| \hat{\mathbf{B}}^{(\text{d-MTLasso})}_{j,\cdot} \right\|^2_{\mathbf{M}^{-1}} \sim \hat{\mathbf{\Omega}}_{j,j} \chi^2_T \ .$$

If $\mathbf{M}$ is known, the quantity $n\|\hat{\mathbf{B}}^{(\text{d-MTLasso})}_{j,\cdot}\|^2_{\mathbf{M}^{-1}}/\hat{\mathbf{\Omega}}_{j,j}$ can be used as a decision statistic to obtain a p-value testing the importance of source j by comparison with the χ^2_T distribution. In practice we need to estimate $\mathbf{M}$ by $\hat{\mathbf{M}}$. Notably, assuming that we have an estimator $\hat{\sigma}$ of σ that verifies approximately $(n - \hat{s})\hat{\sigma}^2/\sigma^2 \sim \chi^2_{n-\hat{s}}$, where $\hat{s} = |\text{Supp}(\hat{\mathbf{B}}^{\text{MTL}})|$ (see Sec. 7.2.5), we take

$$\hat{f}_j := \frac{n\|\hat{\mathbf{B}}^{(\text{d-MTLasso})}_{j,\cdot}\|^2_{\hat{\mathbf{M}}^{-1}}}{T\,\hat{\mathbf{\Omega}}_{j,j}} \ , \tag{7.10}$$

as statistic to compare with a Fisher distribution with parameters T and $n - \hat{s}$, to compute the p-values. The full d-MTLasso algorithm is given in Algo. 4.

3 See the proof of (Lounici et al., 2011, Theorem3.1).

Algorithm 4: d-MTLasso

input : $\mathbf{X} \in \mathbb{R}^{n \times p}, \mathbf{Y}$
$\hat{\mathbf{B}}^{\text{MTL}} \leftarrow \text{MTL}(\mathbf{X}, \mathbf{Y})$ // cross-validated multi-task Lasso
$\hat{\mathbf{E}} \leftarrow \mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}^{\text{MTL}}$ // Residuals
$\hat{s} \leftarrow |\text{Supp}(\hat{\mathbf{B}}^{\text{MTL}})|$
for $t \in [T]$ **do** // Noise level estimation
  $\hat{\sigma}_t^2 = \|\hat{\mathbf{E}}_{\cdot,t}\|^2/(n - \hat{s})$
end
$\hat{\sigma}^2 = \text{median}(\{\hat{\sigma}_t^2, t \in [T]\})$
Get $\hat{\mathbf{M}}$ thanks to Sec. 7.2.5
for $j \in [p]$ **do**
  $\mathbf{z}_j \leftarrow \text{Lasso}(\mathbf{X}^{(-j)}, \mathbf{X}_{\cdot,j})$ // cross-validated Lasso
  $\hat{\mathbf{\Omega}}_{j,j} \leftarrow \frac{n\mathbf{z}_j^\top \mathbf{z}_j}{|\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}||\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}|}$
  $\hat{\mathbf{B}}_{j,\cdot}^{(\text{d-MTLasso})} \leftarrow \frac{\mathbf{z}_j^\top \mathbf{Y}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} - \sum_{k \neq j} \frac{\mathbf{z}_j^\top \mathbf{X}_{\cdot,k} \hat{\mathbf{B}}_{k,\cdot}^{\text{MTL}}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}}$ // Desparsified multi-task Lasso
  $\hat{f}_j \leftarrow \frac{n\|\hat{\mathbf{B}}_{j,\cdot}^{(\text{d-MTLasso})}\|_{\hat{\mathbf{M}}-1}^2}{T\hat{\mathbf{\Omega}}_{j,j}}$ // Inference statistics
end
return $\hat{f}_1, \ldots, \hat{f}_p$

7.2.5 Noise parameters estimation

In Sec. 7.2.1 noise is assumed homogeneous across sensors, this helps to obtain a robust estimator. Extending Reid, Tibshirani, and Friedman (2016) to multi-task regression, we consider the residuals $\hat{\mathbf{E}} = \mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}^{\text{MTL}}$, and the estimated support size $\hat{s}$. Defining, for $t \in [T]$, $\hat{\sigma}_t^2 = \|\hat{\mathbf{E}}_{\cdot,t}\|^2/(n - \hat{s})$, an estimate of σ^2 is:

$$\hat{\sigma}^2 = \text{median}(\{\hat{\sigma}_t^2, t \in [T]\}) \ .$$

Taking the median instead of the mean avoids depending on prospective under-fitted time steps and turns out to be more robust empirically. Similarly, defining for all $t \in [T-1]$, $\hat{\rho}_t = \text{cor}_n(\hat{\mathbf{E}}_{\cdot,t}, \hat{\mathbf{E}}_{\cdot,t+1})$ (where $\text{cor}_n(.,.)$ is the empirical correlation), ρ is estimated by taking $\hat{\rho} = \text{median}(\{\hat{\rho}_t, t \in [T-1]\})$. Then, an estimator $\hat{\mathbf{M}}$ of $\mathbf{M}$ is given by $\hat{\mathbf{M}}_{t,u} = \hat{\sigma}^2 \hat{\rho}^{|t-u|}$.

7.2.6 Clustering to handle spatially structured high-dimensional data

Similarly as in the fMRI setting (see Chapter 2), in the M/EEG setting, the number of sensors is more than one order of magnitude smaller than the number of sources: $n \ll p$. Therefore, estimators of conditional association between sources and observations struggle to identify the solution. The

setting is even more difficult due to the presence of very high correlation between sources as illustrated in Fig. 7.2.

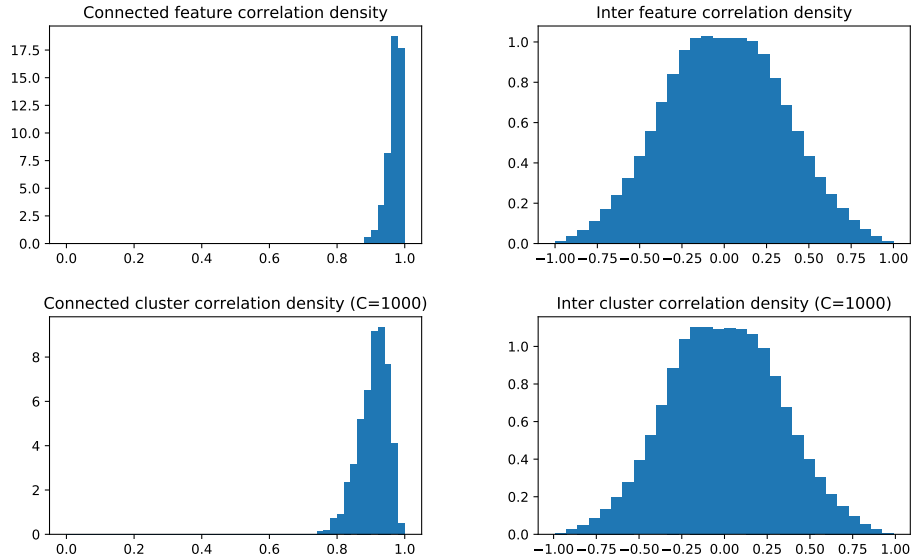


Figure 7.2: Illustrating correlation in MNE sample MEG data. (left): Distribution of the maximum correlation between a feature (resp. cluster) and another connected feature (resp. cluster). (Top) the maximum connected feature correlation is close to 0.98 in average. (Bottom) the maximum connected cluster correlation is lower, close to 0.9 on average. Clustering improves conditioning significantly. (right): The density of the inter feature correlation (top) looks similar to the density of the inter cluster correlation (bottom). By focusing the extreme values of correlation, we see a little decrease of extreme values for the clustered data.

As argued in Chapter 4, further gains can however come from a compression of the design matrix (Bühlmann, 2013; Mandozzi and Bühlmann, 2016). Again, we propose to perform a spatially-constrained clustering to reduce data dimensionality while leveraging spatial structure. More precisely, we consider the hierarchical clustering algorithm described by Varoquaux, Gramfort, and Thirion (2012) that uses Ward criterion[4]. Other clustering schemes might be considered, as long as they yield spatially contiguous regions of the cortical surface. The combination of this clustering algorithm with the d-Lasso or d-MTLasso algorithms will be respectively referred to as clustered desparsified Lasso (cd-Lasso) and clustered desparsified multi-task Lasso (cd-MTLasso).

The number of clusters is denoted by C and, for $r \in [C]$, we denote by G_r the r-th group. Every cluster representative variable is given by the average

4 A typical choice is $C = 1000$ clusters for M/EEG data.

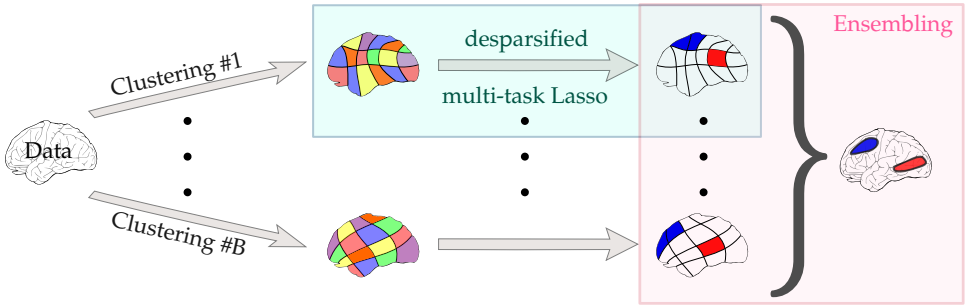


Figure 7.3: ecd-MTL overview diagram. While cd-MTLasso applies d-MTLasso to clustered data, ecd-MTLasso aggregates several cd-MTLasso solutions.

of the covariates it contains. Then, reordering conveniently the columns of $\mathbf{X}$, the compressed design matrix $\mathbf{Z} \in \mathbb{R}^{n \times C}$ is given by:

$$\mathbf{Z} = \mathbf{X}\mathbf{A} \ , \quad \mathbf{A} = \begin{bmatrix} \frac{1}{|G_1|} & — & \frac{1}{|G_1|} & 0 & — & 0 & \dots & 0 & — & 0 \\ 0 & — & 0 & \frac{1}{|G_2|} & — & \frac{1}{|G_2|} & \dots & 0 & — & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & — & 0 & 0 & — & 0 & \dots & \frac{1}{|G_r|} & — & \frac{1}{|G_r|} \end{bmatrix} , \tag{7.11}$$

where $\mathbf{A} \in \mathbb{R}^{p \times C}$.

We say that the compression of $\mathbf{X}$ is of good quality if:

(A2) there exists $\mathbf{\Gamma} \in \mathbb{R}^{C \times T}$ such that $\mathbf{\Gamma}_{r,\cdot} = \sum_{j \in G_r} w_j \mathbf{B}_{j,\cdot}$ with $w_j \geqslant 0$ for all $j \in [p]$, and the associated compression loss $\mathbf{X}\mathbf{B} - \mathbf{Z}\mathbf{\Gamma}$ is "small enough" with respect to the model noise (see Sec. 7.5.5 for more details).

(A3)[5] $\mathrm{RE}(\mathbf{Z}, s')$ is verified on $\mathbf{Z}$ for sparsity parameter $s' \geqslant |\mathrm{Supp}(\mathbf{\Gamma})|$ and constant $\kappa' = \kappa'(s') > 0$.

Proposition 7.2.2. *Assume (7.1), A2, A3, a choice of regularization parameter in the MTLasso regression of $\mathbf{Z}$ against $\mathbf{Y}$ that is large enough, and that the largest cluster of the compression is of size δ, then cd-MTLasso controls the δ-FWER.*

7.2.7 Ensemble of clustered desparsified multi-task Lasso (ecd-MTLasso)

Similarly as in Chapter 4, to reduce the sensitivity of cd-MTLasso to small data perturbations, we propose to randomize over the clustering. We build several clustering solution, considering $B = 100$ different random subsamples of size 10% of the full sample; then we aggregate the p-value maps output by cd-MTLasso. To aggregate the B cd-MTLasso solutions, we use

5 $|\mathrm{Supp}(\mathbf{\Gamma})| \leqslant |\mathrm{Supp}(\mathbf{B})|$ and $\mathbf{Z}$ is generally better conditioned than $\mathbf{X}$ making A3 more plausible than A1.

the adaptive quantile aggregation proposed by Meinshausen, Meier, and Bühlmann (2009) detailed in Chapter 5. The full procedure of ensembling B cd-MTLasso (resp. cd-Lasso), solutions is called ecd-MTLasso for ensemble of clustered desparsified multi-task Lasso (resp. ecd-Lasso).

Proposition 7.2.3. *Assume that for each of the* B *compressions the assumptions of Prop. 7.2.2 are verified, then ecd-MTLasso controls the* δ*-FWER.*

This result is conservative and mixing several cd-MTLasso usually reduces the spatial tolerance δ.

We give an overview diagram to clarify the nesting structure of the proposed solutions in Fig. 7.3 and we give the full algorithm of ecd-MTLasso in Algo. 5.

Algorithm 5: ecd-MTLasso

input : $\mathbf{X} \in \mathbb{R}^{n \times p}, \mathbf{Y}$

param : $C = 1000, B = 100$

for $b = 1, \dots, B$ **do**

  $\mathbf{X}^{(b)} = \texttt{sample}(\mathbf{X})$

  $\mathbf{A}^{(b)} = \texttt{Ward}(C, \mathbf{X}^{(b)})$

  $\mathbf{Z}^{(b)} = \mathbf{X}\mathbf{A}^{(b)}$

  $q^{(b)} = \min(1, C \times \texttt{d-MTLasso}(\mathbf{Z}^{(b)}, \mathbf{Y}))$ `// corr. cluster-wise p-val at bootstrap b`

  for $j = 1, \dots, p$ **do**

    $p_j^{(b)} = q_r^{(b)}$ if $j \in G_r$ `// corrected feature-wise p-values at bootstrap b`

  end

end

for $j = 1, \dots, p$ **do**

  $p_j = \texttt{aggregation}(p_j^{(b)}, b \in [B])$ `// aggregated corrected feature-wise p-values`

end

return p_j for $j \in [p]$

7.2.8 Computational aspects

For solving Lasso or multi-task Lasso problems, we rely for additional speed-up on `celer`[6] (Massias, Gramfort, and Salmon, 2018; Massias et al., 2019), a

6 https://github.com/mathurinm/CELER

solver which is much more efficient than the standard coordinate descent (speed up by more than 10x on our experiments).

To compute d-MTLasso, we must solve p Lasso of size $(n, (p-1))$, and 1 multi-task Lasso with cross-validation on a dataset of size (n, p, T). For $n = 200$, $p = 7500$ and $T = 10$, the algorithms can be run on a standard laptop in around 10 hours (using only 1 CPU). However, the algorithm is embarrassingly parallel and requires around 15 minutes if run on a machine with 50 CPUs. To compute cd-MTLasso, we must solve C Lasso of size $(n, (C-1))$. and 1 multi-task Lasso with cross-validation on a dataset of size (n, C, T). For $n = 200$, $C = 1000$ and $T = 10$, it can be run on a standard local device in less than 1 minute (using only 1 CPU). Finally, to compute ecd-MTLasso, we must solve B cd-MTLasso. For $B = 100$ (25 is already a good value to get most of the advantages of ensembling), $n = 200$, $C = 1000$ and $T = 10$, it can be run on a standard laptop in around 1 hour (using only 1 CPU) and around 1 minute on a machine with 50 CPUs.

Although, when using coordinate-descent-like algorithms, the complexity depends on solver parameters such as tolerance on stopping criteria, the complexity in C (or p) appears empirically to be cubic, while it is linear in n and T. It is also linear in B.

The code for running the different methods, implemented with Python 3, will be released on `https://github.com/ja-che/hidimstat` along with some examples.

7.3 EXPERIMENTS

In this section, after describing the M/EEG datasets, we evaluate the presented methods for source localization. First, in a typical point source simulation, we compare the methods with respect to the standard PLE metric; notably, we study the effect of i/clustering and ii/integrating time dimension. In a second simulation with more realistic features, we examine the δ-FWER control property and compare the support recovery properties of all methods. Lastly, working on real MEG data, we show that, contrary to sLORETA, ecd-MTLasso retrieves expected patterns using a universal threshold.

7.3.1 Data description

In our experiments we use two different datasets: the *sample* dataset and the *somatosensory* dataset that are publicly available from the MNE software (Gramfort et al., 2014).

In the sample dataset, checkerboard patterns were presented to the subject the left and right visual field, interspersed by tones to the left or right ear. Then the sample data is divided in two datasets: an auditory evoked fields (AEF) corresponding to the stimuli in the left ear and a visual evoked field (VEF) corresponding to the stimuli in the left visual hemifield. In the somatosensory dataset, the somatosensory evoked fields (SEF) are obtained following electrical stimulation of the left median nerve on the wrist.

The design matrix $\mathbf{X}$ is computed with a three-shell boundary element model with $p = 7498$ candidate cortical locations with fixed orientation (normal to the cortical surface). For the AEF and VEF datasets, data contained one artifactual channel leading to $n = 203$, while for the somatosensory evoked fields (SEF) data were preprocessed for removal of environmental noise leading to an effective number of samples of $n = 64$ (Taulu, 2006).

For the AEF and VEF datasets, the analysis window for source estimation was chosen from 50 to 100ms based on visual inspection of the evoked data to capture the dominant component, leading to $T = 6$. For the SEF dataset, we analyzed SEFs evoked by bipolar electrical stimulation (0.2ms in duration) of the left median nerve. Then, to capture the main peaks of the evoked response and exclude the strong stimulus artifact, the analysis window was chosen from 30 to 40ms based on visual inspection of the sensor signal.

Preprocessing was done following the standard pipeline from the MNE software (Gramfort et al., 2014). Baseline correction using pre-stimulus data was used. Epochs with peak-to-peak amplitudes exceeding predefined rejection parameters (3 pT for magnetometers and 400 pT/m for gradiometers, and 150 µV for EOG on AEF and VEF and 350 µV for SEF) were assumed to be affected by artifacts and discarded. This resulted in 55 (AEF), 67 (SEF) and 111 (SEF) artifact-free measurements which were average to produce the target matrix $\mathbf{Y}$.

7.3.2 Simulation study

In a first simulation we aim to study how well the proposed estimators perform compared to standard ℓ_2 regularized approaches in terms of localization accuracy, and assess whether time-aware statistical analysis improves upon static d-Lasso, as it is essential for M/EEG source imaging results.

We use the head anatomy and the recording setup from the *sample* dataset. We consider here only gradiometers and remove one defective sensor leading to $n = 203$. Also $p = 7498$. When considering multiple consecutive time instants to demonstrate the ability of the solver to leverage spatio-temporal data, the source is fixed and the temporal noise autocorrelation is set to $\rho = 0.3$.

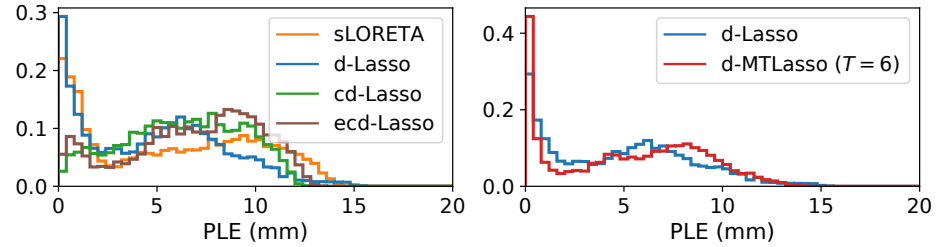


Figure 7.4: Peak Localization Error (PLE) histograms. (left): PLE on a fixed time point (T = 1), sLORETA is outperformed by desparsified Lasso; cd-Lasso and ecd-Lasso are more concentrated and exhibit a smaller number of very low PLE but also a smaller number of extreme PLE values. (right): PLE for desparsified multi-task Lasso (d-MTLasso) with T = 6 compared to d-Lasso (T = 1). More time points improve the results by reducing the PLE.

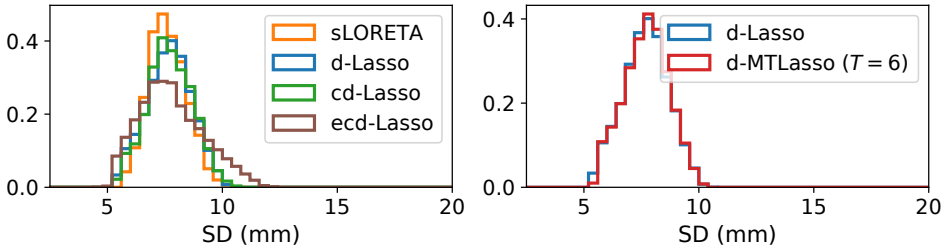


Figure 7.5: Spatial Dispersion (SD) histograms. (left): SD on a fixed time point (Hauk, Wakeman, and Henson, 2011). All methods lead to comparable spatial dispersion. (right): SD for desparsified multi-task Lasso (d-MTLasso) with increasing time points. See Fig. 7.4 for PLE histograms on the same experiments.

Fig. 7.4 reports the normalized histograms of PLE for the 7498 locations for the different methods investigated. While it might seem simplistic to consider a single source, this experiment demonstrates that d-Lasso improves over sLORETA in the presence of noise (see Fig. 7.4, left). In the same figure, one can observe that clustering degrades this performance, as it carries an intrinsic spatial blur. However, even in this adversarial scenario (Dirac-like source location), cd-Lasso and ecd-Lasso remain competitive *w.r.t.* sLORETA, avoiding extreme PLE values. Note that, here, a single time point was used (T = 1).

The right panel in Fig. 7.4 shows that d-MTLasso (T = 6) significantly outperforms d-Lasso (T = 1) in terms of PLE. Leveraging spatio-temporal data indeed increases the signal-to-noise ratio, which enhances spatial specificity.

Effects in terms of spatial dispersion (SD) are minor as exhibited Fig. 7.5.

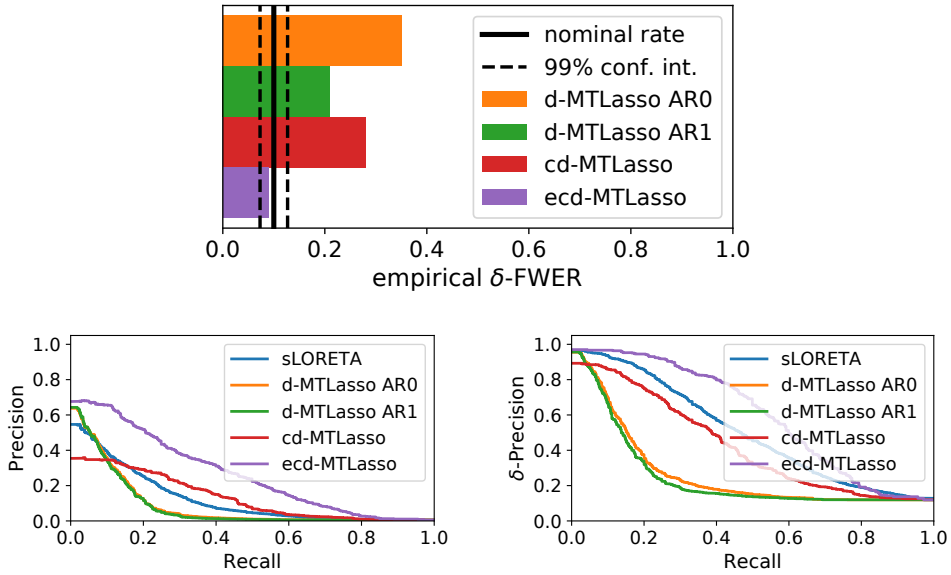


Figure 7.6: δ-FWER, Precision-Recall. (top): δ-FWER control of the different d-MTLasso methods. δ-FWER control is hard for d-MTLasso and cd-MTLasso, as some detections are made far from the true sources, due to remote correlations. Ensembles of clusters mitigates these false detections. (bottom): Precision-recall and δ-precision-recall curves: sLORETA outperforms d-MTLasso AR0 and AR1, because the problem is too high dimensional for the d-MTLasso to work properly. Clustering improves the outcome, and ensembling brings further benefits: ecd-MTLasso outperforms sLORETA.

7.3.3 Experiments on FWER control

We now investigate whether the different versions of d-MTLasso control the δ-FWER on a realistic simulation, and compare their support recovery properties. The data are the same as in Sec. 7.3.2. To simulate the sources, we randomly draw 3 active regions by selecting parcels from a subdivided cortical Freesurfer parcellation with 448 parcels (Khan et al., 2018). For each selected parcel we take as sources all the dipoles at a 10-mm geodesic distance from the center of the parcel (around 10 dipoles per region), fixing the amplitude at 10 nAm. To evaluate how the methods control the δ-FWER, we perform 100 simulations and count how often active sources are found outside the δ-dilated ground truth.

At the top of Fig. 7.6, we see that d-MTLasso does not control the δ-FWER, due to the violation of some assumptions of proposition 1, in particular those regarding source correlation. However, we notice that handling noise autocorrelation reduces the empirical δ-FWER. Using clustering, assumptions of Prop. 7.2.2 are more easily met, in particular the conditioning of the problem is improved (Mattout et al., 2005). Yet cd-MTLasso does not

control the δ-FWER for $\delta = 40\,\text{mm}$, because the δ-FWER is controlled if δ is smaller than the largest cluster diameter, which may not hold. Finally, randomization via ecd-MTLasso further improves FWER control. Empirically, we observe that the δ-FWER is controlled for δ around twice the average cluster diameter. Then, with the limitation of having a compressed design matrix well conditioned (C not too large), we can reduce the tolerance δ by increasing C as shown in Fig. 7.7. We have excluded sLORETA from this study since it does not provide guarantees on the false discoveries.

At the bottom of Fig. 7.6 we show the δ-precision recall curve of the different methods. We first notice that d-MTLasso cannot compete with sLORETA, because the high dimensionality of the problem makes the computation of the source importance overly ill-posed. cd-MTLasso improves detection accuracy, but still does not perform as well as sLORETA when looking at the δ-precision-recall curve. However, adding the ensembling step, the δ-precision improves strongly, making ecd-MTLasso much better than sLORETA.

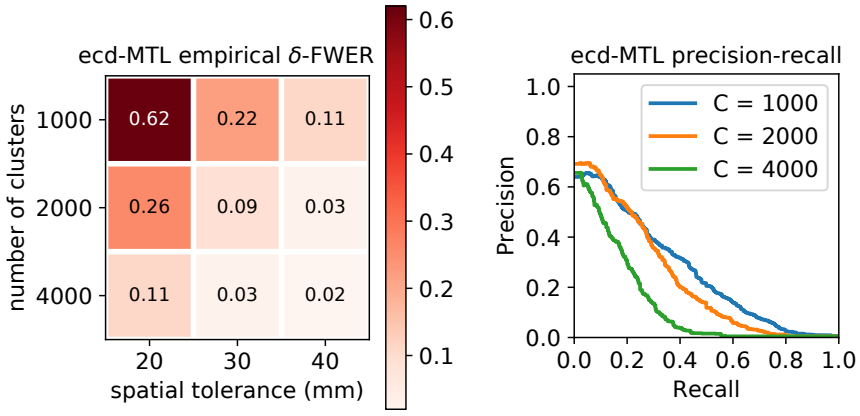


Figure 7.7: ecd-MTLasso empirical δ-FWER and precision recall for different choice of cluster sizes. (left): Running the same simulation as in Sec. 7.3.3, we observe that the spatial tolerance δ can be reduced to 20 mm by increasing the number of clusters up to 4000. With $C = 1000$ clusters (resp. $C = 2000$, $C = 4000$), the average cluster diameter is around 18 mm (resp. 13 mm and 9 mm). It turns out that the δ-FWER is controlled for around twice the diameter (if the compressed design matrix verifies assumption A1). (right): We see that this decrease in spatial tolerance comes with a price regarding support recovery: the precision-recall curve declines with when C is increased. (both): Note that we need to set the hyperparameter c that is used to compute the regularization parameters α (see note coming with (7.5)). We found empirically that it should be inversely proportional to C: for $C = 1000$, $c = 0.5\%$; for $C = 2000$, $c = 0.25\%$; for $C = 4000$, $c = 0.15\%$.

7.3.4 Results on three MEG datasets

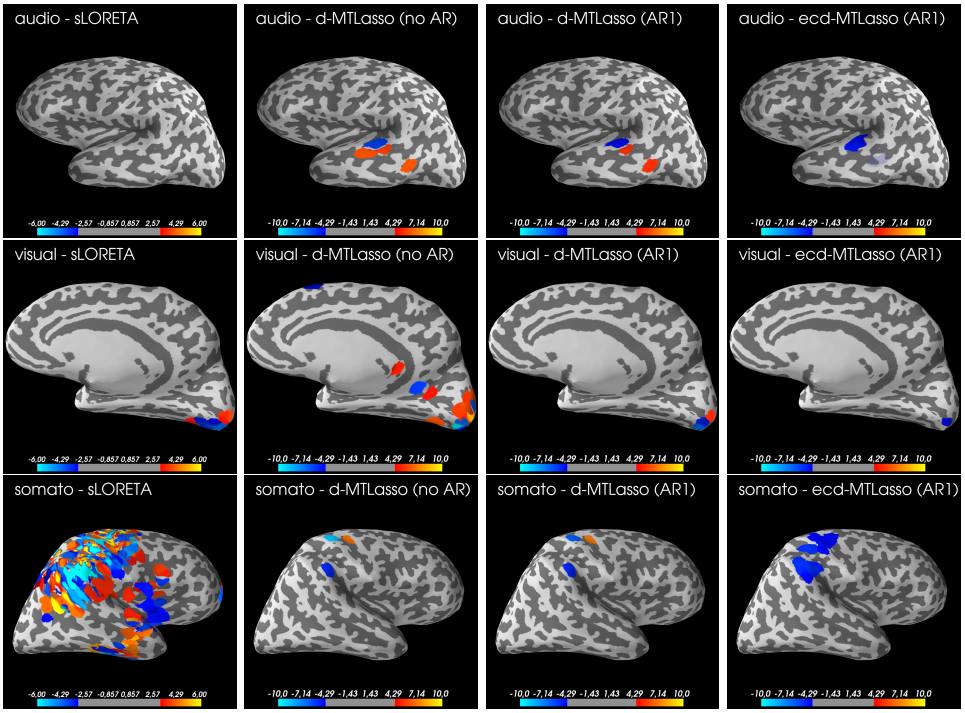


Figure 7.8: Empirical comparison on 3 MEG datasets. From left to right one can see sLORETA, d-MTLasso without AR modeling (assuming non-autocorrelated noise), d-MTLasso with an AR1 noise model and the ecd-MTLasso using also an AR1. Results correspond to auditory (top), visual (middle) and somatosensory (bottom) evoked fields. Colormaps are fixed across datasets and adjusted based on meaningful statistical thresholds in order to qualitatively illustrate FWER control issues.

We now report results on three MEG datasets spanning three types of sensory stimuli: auditory, visual and somatosensory (cf. Sec. 7.3.1). The main results are presented in Fig. 7.8. Additional results, notably concerning EEG data, are presented in Sec. 7.5.7.

Among the many methods for M/EEG source imaging present in the literature, the methods that are compared here have in common to output a statistical map. The ℓ_2 regularized sLORETA method is compared to the debiased sparse estimators presented and evaluated above. The input for all solvers is a time window of data: from $t = 50$ to $t = 100$ ms for AEF and VEF, and from $t = 30$ to $t = 40$ ms for SEF. During such time intervals one can expect the sources to originate primarily from the early sensory cortices whose locations are anatomically known for normal subjects.

Analyzing Fig. 7.8, one can see that all methods manage to highlight the proper functional sensory units (planum temporale for AEF, calcarine region

for VEF and central sulcus for SEF). However, considering sLORETA results, one can observe that at a common threshold of 3.0 on the Student statistic, the estimator is quite spatially specific for VEF, but is overly conservative for AEF and clearly leading to many false positives for SEF. By inspection of the d-MTLasso solution, one can observe that taking into account the autocorrelation of the noise leads to a better calibrated noise variance, and therefore fewer dubious detection. Considering ecd-MTLasso results, while all maps are also thresholded with a single level, one can see that it retrieves expected patterns without making dubious discoveries.

7.3.5 Summary, guidelines and limitations

SUMMARY OF EXPERIMENTS. In Sec. 7.3.2, we have shown that taking into account the time dimension improve the results in terms of PLE. Also, we have seen that even in this adversarial point source scenario (cf. Sec. 7.3.2), clustered methods remain competitive. In Sec. 7.3.3, while no control of false discoveries is proposed by sLORETA, ecd-MTL is the only method that offers statistical control in practice. Namely, it controls the δ-FWER for δ equals to twice the average cluster diameter. Additionally, in this realistic simulation, ecd-MTL exhibits the best support recovery properties. In Sec. 7.3.4, working on real MEG data, we show that, contrary to sLORETA, ecd-MTLasso produces calibrated statistics with universal threshold and retrieves expected patterns without making dubious discoveries. Overall, ecd-MTL offers statistical guarantees and is our privileged method.

GUIDELINES FOR RUNNING ECD-MTLASSO ON M/EEG DATA. First, we try to give guidelines concerning the number of clusters C. Hoyos-Idrobo et al. (2015) exhibit that clustering improves problem conditioning, this means that the Restricted Eigenvalue (RE) property (see assumptions A1 and A3) is more likely to be verified. Complementary, we argue that, keeping C over a hundred (limiting compression loss), the fewer clusters, the more A3 is likely to be verified for Prop. 7.2.2 and Prop. 7.2.3 to hold but also the better the sensitivity of ecd-MTL. However, small C also requires a higher spatial tolerance. We then hit a fundamental trade-off for statistical inference between sensitivity and spatial specificity. Then, C can be chosen depending on the problem setting: if it is difficult (noisy), it seems natural to lower spatial tolerance expectations (diminish C); in that sense ecd-MTL is an adaptive method (cf. Fig. 7.7). For the present use case, taking $C = 1000$ seems an adequate trade-off to ensure δ-FWER control with reasonable spatial tolerance.

Now, we give recommendation concerning the window size and the time sampling to use. Choosing too short windows complicate AR model estimation due to the lack of data while choosing too large windows may

lead to non stationary support. Then, we recommend taking windows of 20 to 50ms with a time sampling at 5 or 10ms since keeping $T < 10$ reduces computation time and should not decrease sensitivity significantly.

Finally, when working with M/EEG data, we recommend to use only 10% of the full data to compute several clustering solutions with spatial constraint and Ward criteria to ensure enough diversity.

LIMITATIONS. The main limitation is the fact that mixing different types of sensors violates modeling assumptions both on temporal correlations and on spatial correlations, that is why we had to treat MEG and EEG sensors separately. A possibility to handle heterogeneous sensors is to follow Massias et al. (2018), but for the temporal part further developments are required and left for future work.

Also left for future work, is the possibility of studying windows larger than 50ms. A simple solution is to slide a window of 20 to 50ms over the considered period of time.

Finally, a more common limitation is the fact that assumptions are hard to test in practice.

7.4 CONCLUSION

The MEG source imaging problem poses a hard statistical inference challenge: namely that of high-dimensional statistical analysis, furthermore with high correlations in the design. We have proposed an estimator that calibrates correctly the effects size and variance, up to a number of assumptions, that are not easily met: some level of sparsity, mild correlation across sensors, homogeneity and heteroscedasticity of the noise. Up to these assumptions, and up to a spatial tolerance on the exact location of the sources, we provide the first method with statistical guarantees for source imaging. This is made possible by bringing several improvements to the original desparsified lasso solution: a multi-task formulation that increases power by basing inference on multiple time steps, a clustering step that renders the design less ill-posed and an ensembling step that mitigates the (hard) choice of clusters. Finally, our privileged method, ecd-MTLasso, runs in less than 10 mn on a real dataset on non-specialized hardware, making it usable by practitioners.

7.5 SUPPLEMENTARY MATERIAL

7.5.1 Statement of broader impact

Magnetoencephalography (MEG) and electroencephalography (EEG) offer a unique opportunity to image brain activity non-invasively with a temporal resolution in the order of milliseconds. This is relevant for cognitive neuroscience to describe the sequence of active areas during certain cognitive tasks, but also for clinical neuroscience, where electrophysiology is used for diagnosis (*e.g.*, sleep medicine, epilepsy presurgical mapping). Yet, doing brain imaging with M/EEG requires to solve a challenging high-dimensional inverse problem for which statistical guarantees are crucially important. In this work, we address this statistical challenge when using sparsity promoting regularization and when considering the specificity of M/EEG signals: data are spatio-temporal and the noise is temporally autocorrelated. The proposed algorithm is built on very recent work in optimization to speed up Lasso-type solvers, as well as work in mathematical statistics on desparsified Lasso estimators. We believe that this work, whose contribution is both on the modeling side and on the inference aspects, brings sparse estimators close to a wide adoption in the neuroscience community.

We also would like to emphasize that the inference framework can be adapted to many other high-dimensional problems where data structure can be leveraged: biomedical data and physical observations (cardiac or brain monitoring, genomics, seismology, etc.), especially those that involve severely ill-posed inverse problems.

7.5.2 Extended Restricted Eigenvalue assumption

Here, we rewrite (Lounici et al., 2011, Assumption 3.1), adjusting it for the multi-task lasso case (particular case of the more general group Lasso). Notice that for a given value of T, the assumption is equivalent to (Lounici et al., 2011, Assumption 4.1). Let $1 \leqslant s \leqslant p$ be an integer that gives an upper bound on the sparsity $|\mathrm{Supp}(\mathbf{B})|$. The extended Restricted Eigenvalue assumption $\mathrm{RE}(\mathbf{X}, s)$ is verified on $\mathbf{X}$ for sparsity parameter s and constant $\kappa = \kappa(s) > 0$, if:

$$\min \left\{ \frac{\|\mathbf{X}\boldsymbol{\Theta}\|}{\sqrt{nT}\,\|\boldsymbol{\Theta}_J\|} : |J| \leqslant s, \boldsymbol{\Theta} \in \mathbb{R}^{p\times T} \setminus \{0\}, \left\|\boldsymbol{\Theta}_{J^C}\right\|_{2,1} \leqslant 3\,\|\boldsymbol{\Theta}\|_{2,1} \right\} \geqslant \kappa \ , \tag{7.12}$$

where $J \subset [p]$ and J^C denotes its complementary *i.e.*, $J^C = [p] \setminus J$, and $\boldsymbol{\Theta}_J$ refers to the matrix $\boldsymbol{\Theta}$ without the rows J^C.

7.5.3 Probability lemma

Lemma 7.5.1. *Let $\varepsilon \in \mathbb{R}^T$ be a centered Gaussian random vector with (symmetric positive definite) covariance $\mathbf{M} \in \mathbb{R}^{T\times T}$. Then, the random variable $\varepsilon^\top \mathbf{M}^{-1}\varepsilon$ follows a χ^2_T distribution.*

Proof. Note first that since $\mathbf{M}$ is symmetric positive definite, its square-root $\mathbf{N} \in \mathbb{R}^{T\times T}$ exists and is a symmetric positive definite matrix satisfying $\mathbf{N}^2 = \mathbf{M}$. Hence, this leads to the following displays

$$\varepsilon^\top \mathbf{M}^{-1}\varepsilon = (\mathbf{N}^{-1}\varepsilon)^\top(\mathbf{N}^{-1}\varepsilon).$$

We have that $\mathbf{N}^{-1}\varepsilon$ is a centered Gaussian random vector, and its covariance matrix reads:

$$\begin{aligned}
\mathbb{E}\left[(\mathbf{N}^{-1}\varepsilon)(\mathbf{N}^{-1}\varepsilon)^\top\right] &= \mathbb{E}\left[\mathbf{N}^{-1}\varepsilon\varepsilon^\top\mathbf{N}^{-1}\right] \\
&= \mathbb{E}\left[\mathbf{N}^{-1}\varepsilon\varepsilon^\top\mathbf{N}^{-1}\right] \\
&= \mathbf{N}^{-1}\mathbb{E}\left[\varepsilon\varepsilon^\top\right]\mathbf{N}^{-1} \\
&= \mathbf{N}^{-1}\mathbf{M}\mathbf{N}^{-1} \\
&= \mathbf{N}^{-1}\mathbf{N}^2\mathbf{N}^{-1} \\
&= \mathrm{Id}_T \quad .
\end{aligned}$$

To conclude $\mathbf{N}^{-1}\varepsilon \in \mathbb{R}^T$ is a centered Gaussian vector with covariance Id_T, hence its squared Euclidean norm $\left\|\mathbf{N}^{-1}\varepsilon\right\|^2 = (\mathbf{N}^{-1}\varepsilon)^\top(\mathbf{N}^{-1}\varepsilon)$ follows a χ^2_T distribution. □

7.5.4 Proof of Prop. 7.2.1

Now, we give a proof of Prop. 7.2.1:

Proof. First, let us fix an index $j \in [p]$. Then, using (7.7) we have:

$$\begin{aligned}
\sqrt{n}(\hat{\mathbf{B}}^{(\text{d-MTLasso})}_{j,\cdot} - \mathbf{B}_{j,\cdot}) &= \sqrt{n}\frac{\mathbf{z}_j^\top \mathbf{E}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} - \sum_{k\neq j}\frac{\sqrt{n}\,\mathbf{z}_j^\top \mathbf{X}_{\cdot,k}(\hat{\mathbf{B}}^{\text{MTL}}_{k,\cdot} - \mathbf{B}_{k,\cdot})}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} \\
&= \boldsymbol{\Lambda}_{j,\cdot} + \boldsymbol{\Delta}_{j,\cdot} \quad ,
\end{aligned} \tag{7.13}$$

where $\boldsymbol{\Lambda}_{j,\cdot} = \sqrt{n}\frac{\mathbf{z}_j^\top \mathbf{E}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}}$ and $\boldsymbol{\Delta}_{j,\cdot} = \sqrt{n}\sum_{k\neq j}\mathbf{P}_{j,k}(\mathbf{B}_{k,\cdot} - \hat{\mathbf{B}}^{\text{MTL}}_{k,\cdot})$ with

$$\mathbf{P}_{j,k} = \frac{\mathbf{z}_j^\top \mathbf{X}_{\cdot,k}}{\mathbf{z}_j^\top \mathbf{X}_{\cdot,j}} \quad .$$

Now, we show that $\boldsymbol{\Lambda}_{j,.} \sim \mathcal{N}_p(0, \hat{\boldsymbol{\Omega}}_{j,j}\mathbf{M})$, or equivalently we show that $\mathbf{E}^\top \mathbf{z}_j \sim \mathcal{N}(0, n\left\|\mathbf{z}_j\right\|^2 \mathbf{M})$. It is clear that $\mathbf{E}^\top \mathbf{z}_j$ is a centered Gaussian vector. Then, its covariance denoted by $\mathbf{V}^{(j)}$, can be computed as follows:

$$\mathbf{V}^{(j)} = \mathbb{E}(\mathbf{E}^\top \mathbf{z}_j \mathbf{z}_j^\top \mathbf{E}) \in \mathbb{R}^{T\times T} \quad ,$$

whose general term is given for $t, t' \in [T]$ by

$$\begin{aligned}
\mathbf{V}^{(j)}_{t,t'} &= \mathbb{E}(\mathbf{E}_{.,t}^\top \mathbf{z}_j \mathbf{z}_j^\top \mathbf{E}_{.,t'}) \\
&= \mathbb{E}(\mathbf{z}_j^\top \mathbf{E}_{.,t'} \mathbf{E}_{.,t}^\top \mathbf{z}_j) \qquad \text{(scalar values commute)} \\
&= \mathbf{z}_j^\top \mathbb{E}(\mathbf{E}_{.,t'} \mathbf{E}_{.,t}^\top) \mathbf{z}_j \\
&= \mathbf{z}_j^\top \mathbb{E}(\sum_{i=1}^{n} \mathbf{E}_{i,t'} \mathbf{E}_{i,t}^\top) \mathbf{z}_j \\
&= \mathbf{z}_j^\top \sum_{i=1}^{n} \mathbb{E}(\mathbf{E}_{i,t'} \mathbf{E}_{i,t}^\top) \mathbf{z}_j \quad .
\end{aligned}$$

Then, the noise structure in (7.2) yields $\mathbf{V}^{(j)}_{t,t'} = \mathbf{z}_j^\top n\mathbf{M}_{t,t'}\mathbf{z}_j = n\left\|\mathbf{z}_j\right\|^2 \mathbf{M}_{t,t'}$.

Now, we show that with high probability $\|\boldsymbol{\Delta}\|_{2,1} = O\left(\frac{s\lambda}{\sqrt{n}\kappa^2}\right)$. First, notice that:

$$\|\boldsymbol{\Delta}\|_{2,1} \leqslant \sqrt{n} \max_{k\neq j} |\mathbf{P}_{j,k}| \left\|\hat{\mathbf{B}}^{\text{MTL}} - \mathbf{B}\right\|_{2,1} \quad .$$

For a convenient choice of the regularization parameters $\boldsymbol{\alpha}$, using Bühlmann and van de Geer (2011, Lemma 2.1) and following the same approach as Dezeure et al. (2015, Appendix A.1), we obtain, with high probability:

$$\sqrt{n} \max_{k\neq j} |\mathbf{P}_{j,k}| = O\left(\frac{1}{\sqrt{n}}\right) \quad .$$

Bounds on $\|\hat{\mathbf{B}}^{\text{MTL}} - \mathbf{B}\|_{2,1}$ are also available in the literature (Lounici et al., 2011) for $\rho = 0$ and can be extended to $\rho > 0$ similarly. Notably, provided $\rho = 0$, assuming A1 for a sparsity parameter $|\text{Supp}(\mathbf{B}^*)| \leqslant s$, a given constant $\kappa = \kappa(s) > 0$, and a choice of λ large enough in (7.4), (Lounici et al., 2011, Theorem 3.1) gives directly the following bound, with high probability:

$$\left\|\hat{\mathbf{B}}^{\text{MTL}} - \mathbf{B}\right\|_{2,1} = O\left(\frac{s\lambda}{\kappa^2}\right) \quad .$$

□

Remark 7.5.1. *Following van de Geer et al. (2014), to neglect $\boldsymbol{\Delta}$ we need to have $\|\boldsymbol{\Delta}\|_\infty = o(1)$. This condition is verified if $s = o(\frac{\sqrt{n}\kappa^2}{\lambda})$.*

7.5.5 Proof of Prop. 7.2.2

Before starting the proof, let us give more precision on assumption A2, the complete assumption is the following:

(A2) there exists $\mathbf{\Gamma} \in \mathbb{R}^{C \times T}$ such that $\mathbf{\Gamma}_{r,\cdot} = \sum_{j \in G_r} w_j \mathbf{B}_{j,\cdot}$ with $w_j \geqslant 0$ for all $j \in [p]$, so that the associated compression loss $\mathbf{XB} - \mathbf{Z\Gamma}$ is bounded as follows:

$$\|\mathbf{XB} - \mathbf{Z\Gamma}\|_{2,2}^2 \leqslant \xi \frac{T\phi_{\min}^2(\mathbf{M})}{n} = \xi \frac{T\phi_{\min}^2(\mathbf{R})\sigma^2}{n} \quad , \tag{7.14}$$

where $\xi > 0$ is an arbitrary small constant, $\phi_{\min}^2(\mathbf{M}) > 0$ is the smallest eigenvalue of $\mathbf{M}$ and $\phi_{\min}^2(\mathbf{R}) > 0$ is the smallest eigenvalue of $\mathbf{R}$, the temporal correlation matrix of the noise defined by $\mathbf{R} = \mathbf{M}/\sigma^2$. The assumption plainly means that the noise induced by design matrix compression is small enough with respect to the model noise.

Now we give a proof of Prop. 7.2.2:

Proof. First, we derive the d-MTLasso for the compressed problem, for $r \in [C]$:

$$\hat{\mathbf{\Gamma}}_{r,\cdot}^{(\text{d-MTLasso})} = \frac{\mathbf{a}_r^\top \mathbf{Y}}{\mathbf{a}_r^\top \mathbf{Z}_{\cdot,r}} - \sum_{l \neq r} \frac{\mathbf{a}_r^\top \mathbf{Z}_{\cdot,l} \hat{\mathbf{\Gamma}}_{r,\cdot}^{\text{MTL}}}{\mathbf{a}_r^\top \mathbf{Z}_{\cdot,r}} \quad , \tag{7.15}$$

where a_r's are the residuals obtained by nodewise Lasso on $\mathbf{Z}$ playing the same role as the z_j's in (7.7). Then, as done in Sec. 7.5.4, we derive:

$$\begin{aligned}
\sqrt{n}(\hat{\mathbf{\Gamma}}_{r,\cdot}^{(\text{d-MTLasso})} - \mathbf{\Gamma}_{r,\cdot}) &= \sqrt{n} \frac{\mathbf{a}_r^\top \mathbf{E}}{\mathbf{a}_r^\top \mathbf{Z}_{\cdot,r}} - \sum_{l \neq r} \frac{\sqrt{n}\, \mathbf{a}_r^\top \mathbf{Z}_{\cdot,l} (\hat{\mathbf{\Gamma}}_{l,\cdot}^{\text{MTL}} - \mathbf{\Gamma}_{l,\cdot})}{\mathbf{a}_r^\top \mathbf{Z}_{\cdot,r}} \\
&\quad + \frac{\sqrt{n}\, \mathbf{a}_r^\top (\mathbf{XB} - \mathbf{Z\Gamma})}{\mathbf{a}_r^\top \mathbf{Z}_{\cdot,r}} \\
&= \mathbf{\Lambda}'_{r,\cdot} + \mathbf{\Delta}'_{r,\cdot} + \mathbf{\Pi}_{r,\cdot} \quad ,
\end{aligned} \tag{7.16}$$

We treat $\mathbf{\Lambda}'$ and $\mathbf{\Delta}'$ as in Sec. 7.5.4, assuming that the assumptions that are used to bound (hence, neglect) $\mathbf{\Delta}'$ are verified (notably A3).

Next, for $r \in [C]$, we want to establish that $\frac{n\|\mathbf{\Pi}_{r,\cdot}\|_{\mathbf{M}^{-1}}^2}{T\hat{\Omega}'_{r,r}}$ is negligible, *i.e.*, that $\mathbf{\Pi}$ has a negligible effect on all decision statistics, where the covariance $\hat{\mathbf{\Omega}}'$ has the following generic diagonal term:

$$\hat{\Omega}'_{r,r} = \frac{n\,\|\mathbf{a}_r\|^2}{|\mathbf{a}_r^\top \mathbf{Z}_{\cdot,r}|^2} \quad .$$

Given that

$$\|\mathbf{\Pi}_{r,\cdot}\|_{\mathbf{M}^{-1}}^2 = \frac{n\left\|\mathbf{a}_r^\top(\mathbf{XB}-\mathbf{Z\Gamma})\right\|_{\mathbf{M}^{-1}}^2}{|\mathbf{a}_r^\top\mathbf{Z}_{\cdot,r}|^2} \tag{7.17}$$

$$\leqslant n\frac{\left\|\mathbf{a}_r^\top\right\|^2}{|\mathbf{a}_r^\top\mathbf{Z}_{\cdot,r}|^2}\frac{\|\mathbf{XB}-\mathbf{Z\Gamma}\|_{2,2}^2}{\phi_{\min}^2(\mathbf{M})}\ , \tag{7.18}$$

where $\|\cdot\|_{2,2}$ denotes the spectral norm. Then, we obtain that

$$\frac{n\,\|\mathbf{\Pi}_{r,\cdot}\|_{\mathbf{M}^{-1}}^2}{T\hat{\mathbf{\Omega}}'_{r,r}} \leqslant \frac{n}{T}\frac{\|\mathbf{XB}-\mathbf{Z\Gamma}\|_{2,2}^2}{\phi_{\min}^2(\mathbf{M})} \leqslant \xi\ . \tag{7.19}$$

Then, if A2 is verified for ξ small enough, we can also neglect $\mathbf{\Pi}$ in front of $\mathbf{\Lambda}'$.

Then, by neglecting $\mathbf{\Pi}$ and $\mathbf{\Delta}'$, we have:

$$\sqrt{n}(\hat{\mathbf{\Gamma}}^{(\text{d-MTLasso})} - \mathbf{\Gamma}) \sim \mathcal{N}_C(0,\ \hat{\mathbf{\Omega}}'_{r,r}\mathbf{M})\ . \tag{7.20}$$

Then we can construct p-values that test the r-th null hypothesis $H_0^{(r)}$: "$\mathbf{\Gamma}_{j,\cdot} = 0$", applying the same technique as in Sec. 7.2.4. By correcting these p-values —*e.g.*, using the Bonferroni correction (Dunn, 1961), we multiply by C the initial p-values—, we obtain cluster-wise corrected p-values that control the FWER.

Since, for all $r \in [C]$, $\mathbf{\Gamma}_{r,\cdot}$ is a linear combination of $\mathbf{B}_{j,\cdot}$ for $j \in G_r$, then $\mathbf{\Gamma}_{r,\cdot} \neq 0$ if at least there exist $j \in G_r$ such that $\mathbf{B}_{j,\cdot} \neq 0$.

Then, defining the feature-wise corrected p-values by the corrected p-values of the corresponding cluster, and assuming that clusters are at most of size δ, such corrected p-values control the δ-FWER.

□

Remark 7.5.2. In assumption A2, having a positive linear combination is not necessary, a simple linear combination is sufficient. However, we assumed that $\mathbf{\Gamma}_{r,\cdot}$ was a positive linear combination of $\mathbf{B}_{j,\cdot}$ for $j \in G_r$, to get the following desired properties:

If additionally for $r \in [C]$, *for all* $(j,k) \in G_r^2$, *we have* $\text{sign}(\mathbf{B}_{j,\cdot}) = \text{sign}(\mathbf{B}_{k,\cdot})$, *then* $\text{sign}(\mathbf{\Gamma}_{r,\cdot}) = \text{sign}(\mathbf{B}_{j,\cdot})$.

This means that if all the features' weights in a cluster have the same sign, there exists a compression verifying A2 such that the cluster weight preserves the sign.

7.5.6 Proof of Prop. 7.2.3

Proof. Assuming the assumptions of Prop. 7.2.3 and applying Prop. 7.2.2, we can, for each of the B compression of the problem in (7.1), construct a

corrected p-value family that control the δ-FWER. Applying the quantile aggregate method of Meinshausen, Meier, and Bühlmann (2009), we derive a corrected p-value family taking into account for each compression choice; this aggregated corrected p-value family also controls the δ-FWER (cf. Chapter 5). □

7.5.7 Supplementary figures

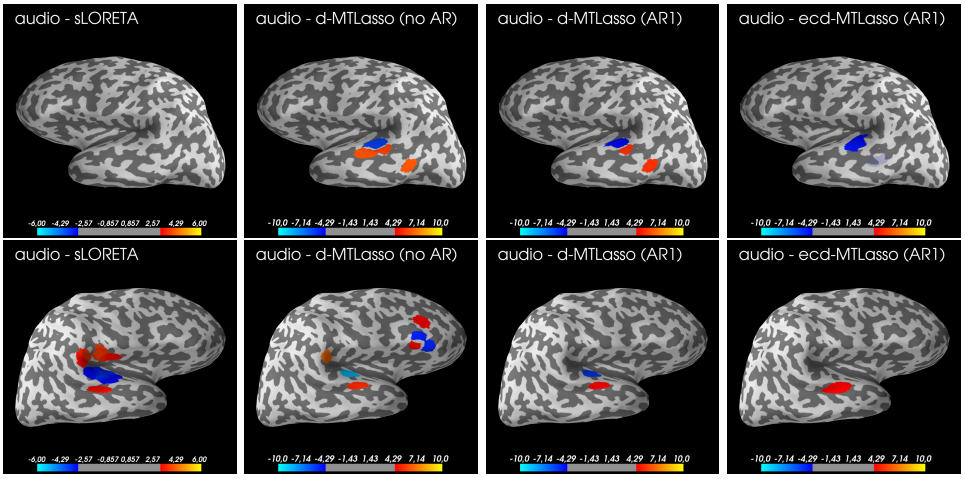


Figure 7.9: Comparison on audio dataset on both hemispheres. From left to right are compared sLORETA, d-MTLasso without AR modeling (noise is assumed non-autocorrelated), d-MTLasso with an AR1 noise model and the ecd-MTLasso using also an AR1. Results correspond to auditory evoked fields.

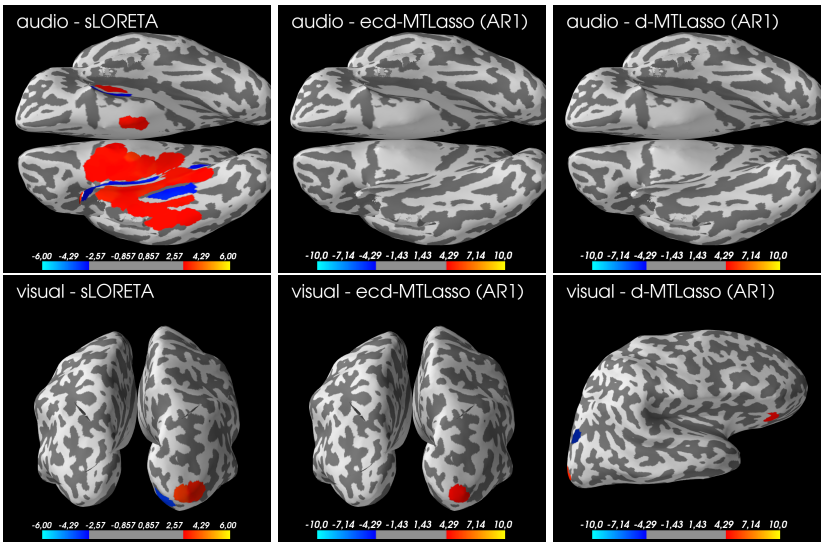


Figure 7.10: Results on real data keeping only EEG sensors. Auditory activations (top) have historically been hard to infer with EEG sensors: sLORETA produces only false discoveries while ecd-MTL and d-MTL make no discoveries. In the visual experiment (bottom): sLORETA and ecd-MTL produce expected patterns, d-MTL produces expected patterns plus one false discovery in the frontal lobe. In our work, we have emphasized MEG experiments: they offer more sensors compared to EEG leading to improved statistical power.

CONCLUSION

CONCLUSION

SUMMARY OF THE CONTRIBUTIONS. The aim of this thesis was to propose multivariate statistical inference procedures that can handle high-dimensional data with spatial structure such as task fMRI data. The procedures we have proposed leverage state-of-the-art high-dimensional inference procedures, on spatially constrained clustering algorithms, and on ensembling techniques. A critical point was to establish their theoretical properties and conduct a thorough empirical validation. We have notably revealed the existence of a trade-off between strong power and high spatial accuracy. Indeed, a key step was to propose to integrate a spatial tolerance in the statistical control of the false discoveries. This brings two benefits: an effective statistical control for spatially-structured data, and some power to detect true positives. Finally, we have been able to adapt the method initially calibrated for fMRI datasets to spatio-temporal data and autocorrelated noise, leading to an application to MEG source imaging.

FUTURE DIRECTIONS. We have several directions in mind to complete or improve the solutions we have proposed. First, we would like to extend our work to binary classification, to do so, it would require to investigate the existing adaptations of the d-Lasso procedure (or other procedures) to this binary setup. So far, we considered that the regression model could properly approximate this setup, and we preferred to focus on the extension to spatio-temporal data first. But this is clearly needed to make the method more useful for practitioners. Second, regarding computational aspects, we believe that the d-Lasso and the d-MTLasso can be accelerated. Indeed, a first lead to optimize the algorithm is to use a warm start technique for solving the Lasso problems necessary to compute the score vectors. Third, having in mind that we need homogeneous cluster diameters to minimize spatial tolerance, strong clustering variance to increase the effect of ensembling but also a decent data-fitting capacity to limit the compression loss, we think that it is possible to provide or improve the spatially constrained clustering algorithm. A first idea would be to add up a criterion to the clustering algorithm we are currently using: it would consist in ensuring that the ratio of the largest cluster diameter over the smallest cluster diameter is upper bounded by a given value. Finally, it seems natural to extend the scope of application of the algorithms we introduced to other fields, especially to genomics.

REFERENCES

REFERENCES

Abraham, A., F. Pedregosa, M. Eickenberg, P. Gervais, A. Mueller, J. Kossaifi, A. Gramfort, B. Thirion, and G. Varoquaux (2014). "Machine learning for neuroimaging with scikit-learn." In: *Frontiers in neuroinformatics* 8, p. 14.

Anderson, M. J. (2001). "Permutation tests for univariate or multivariate analysis of variance and regression." In: *Canadian journal of fisheries and aquatic sciences* 58.3, pp. 626–639.

Ashburner, J. and K. J. Friston (2000). "Voxel-based morphometry—the methods." In: *Neuroimage* 11.6, pp. 805–821.

Aydöre, S., B. Thirion, and G. Varoquaux (2019). "Feature Grouping as a Stochastic Regularizer for High-Dimensional Structured Data." In: *International Conference on Machine Learning*.

Bach, F. R. (2008). "Bolasso: model consistent lasso estimation through the bootstrap." In: *Proceedings of the 25th international conference on Machine learning*, pp. 33–40.

Baillet, S., J. C. Mosher, and R. M. Leahy (2001). "Electromagnetic brain mapping." In: *IEEE Signal Proc. Mag.* 18.6, pp. 14–30.

Balding, D.J. (2006). "A tutorial on statistical methods for population association studies." In: *Nature reviews genetics* 7.10, pp. 781–791.

Barber, R. F. and E. Candès (Oct. 2015). "Controlling the false discovery rate via knockoffs." In: *Ann. Statist.* 43.5, pp. 2055–2085.

Bellec, Pierre C and Cun-Hui Zhang (2019). "De-biasing the lasso with degrees-of-freedom adjustment." In: *arXiv preprint arXiv:1902.08885*.

Benjamini, Y. and Y. Hochberg (1995). "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing." In: *J. R. Stat. Soc. Ser. B Stat. Methodol.* 57.1, pp. 289–300.

Berk, R., L. Brown, A. Buja, K. Zhang, and L. Zhao (2013). "Valid post-selection inference." In: *Ann. Statist.* 41.2, pp. 802–837.

Blanchard, G., D. Geman, et al. (2005). "Hierarchical testing designs for pattern recognition." In: *The Annals of Statistics* 33.3, pp. 1155–1202.

Breiman, L. (1996). "Bagging Predictors." In: *Machine Learning* 24.2, pp. 123–140.

Bühlmann, P. (Sept. 2013). "Statistical significance in high-dimensional linear models." In: *Bernoulli* 19.4, pp. 1212–1242.

Bühlmann, P. and S. van de Geer (2011). *Statistics for high-dimensional data*. Springer Series in Statistics. Methods, theory and applications. Heidelberg: Springer.

Bushong, S. C. and G. Clarke (2013). *Magnetic Resonance Imaging-E-Book: Physical and Biological Principles*. Elsevier Health Sciences.

Button, K. S., J. P. A. Ioannidis, C. Mokrysz, B. A. Nosek, J. Flint, E. S. J. Robinson, and M. R. Munafò (2013). "Power failure: why small sample size undermines the reliability of neuroscience." In: *Nature Reviews Neuroscience* 14, p. 365.

Bühlmann, P., P. Rütimann, S. van de Geer, and C.-H. Zhang (2013). "Correlated variables in regression: Clustering and sparse estimation." In: *Journal of Statistical Planning and Inference* 143.11, pp. 1835 –1858.

Candès, E., Y. Fan, L. Janson, and J. Lv (2018). "Panning for gold:'model-X' knockoffs for high dimensional controlled variable selection." In: *J. R. Stat. Soc. Ser. B Stat. Methodol.* 80.3, pp. 551–577.

Celentano, M., A. Montanari, and Y. Wei (2020). "The Lasso with general Gaussian designs with applications to hypothesis testing." In: *arXiv preprint arXiv:2007.13716*.

Chatterjee, A. and S. N. Lahiri (2011). "Bootstrapping lasso estimators." In: *J. Amer. Statist. Assoc.* 106.494, pp. 608–625.

Chatterjee, A., S. N. Lahiri, et al. (2013). "Rates of convergence of the adaptive LASSO estimators to the oracle distribution and higher order refinements by the bootstrap." In: *The Annals of Statistics* 41.3, pp. 1232–1259.

Chen, S. and D. L. Donoho (1994). "Basis pursuit." In: *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*. Vol. 1. IEEE, pp. 41–44.

Chevalier, J.-A., A. Gramfort, J. Salmon, and B. Thirion (2020). "Statistical control for spatio-temporal MEG/EEG source imaging with desparsified multi-task Lasso." In: *NeurIPS*.

Chevalier, J.-A., J. Salmon, and B. Thirion (2018). "Statistical Inference with Ensemble of Clustered Desparsified Lasso." In: *MICCAI*. Springer, pp. 638–646.

Cortes, C. and V. Vapnik (1995). "Support-vector networks." In: *Machine learning* 20.3, pp. 273–297.

Cox, D. D. and R. L. Savoy (2003). "Functional magnetic resonance imaging (fMRI)"brain reading": detecting and classifying distributed patterns of fMRI activity in human visual cortex." In: *Neuroimage* 19.2, pp. 261–270.

Dale, A. M., A. K. Liu, B. R. Fischl, R. L. Buckner, J. W. Belliveau, J. D. Lewine, and E. Halgren (2000). "Dynamic Statistical Parametric Mapping." In: *Neuron* 26.1, pp. 55–67.

Damadian, R., M. Goldsmith, and L. Minkoff (1977). "Nmr in cancer: Xvi. fonar image of the uve human body." In: *Physiological chemistry and physics*.

Dehman, A., C. Ambroise, and P. Neuvial (2015). "Performance of a blockwise approach in variable selection using linkage disequilibrium information." In: *BMC bioinformatics* 16.1, p. 148.

Demirci, O., V. P. Clark, V. A. Magnotta, N. C. Andreasen, J. Lauriello, K. A. Kiehl, G. D. Pearlson, and V. D. Calhoun (2008). "A review of challenges in the use of fMRI for disease classification/characterization and a projection

pursuit application from a multi-site fMRI schizophrenia study." In: *Brain imaging and behavior* 2.3, pp. 207–226.

Dezeure, R., P. Bühlmann, L. Meier, and N. Meinshausen (2015). "High-Dimensional Inference: Confidence Intervals, p-Values and R-Software hdi." In: *Statist. Sci.* 30.4, pp. 533–558.

Dezeure, R., P. Bühlmann, and C.-H. Zhang (2017). "High-dimensional simultaneous inference with the bootstrap." In: *Test* 26.4, pp. 685–719.

Dunn, O. J. (1961). "Multiple comparisons among means." In: *J. Amer. Statist. Assoc.* 56.293, pp. 52–64.

Eklund, A., T. Nichols, and H. Knutsson (2016). "Cluster failure: Why fMRI inferences for spatial extent have inflated false-positive rates." In: *Proc. Natl. Acad. Sci. U.S.A.* 113.28, pp. 7900–7905.

El Karoui, N. and E. Purdom (2018). "Can we trust the bootstrap in high-dimensions? the case of linear models." In: *The Journal of Machine Learning Research* 19.1, pp. 170–235.

Evans, A. C., A. L. Janke, D. L. Collins, and S. Baillet (2012). "Brain templates and atlases." In: *Neuroimage* 62.2, pp. 911–922.

Fan, Y., N. Batmanghelich, C. M. Clark, C. Davatzikos, and Alzheimer's Disease Neuroimaging Initiative (2008). "Spatial patterns of brain atrophy in MCI patients, identified via high-dimensional pattern classification, predict subsequent cognitive decline." In: *Neuroimage* 39.4, pp. 1731–1743.

Fonov, V. S., A. C. Evans, R. C. McKinstry, C. R. Almli, and D. L. Collins (2009). "Unbiased nonlinear average age-appropriate brain templates from birth to adulthood." In: *NeuroImage* 47, S102.

Friston, K. J. (2009). "Modalities, modes, and models in functional neuroimaging." In: *Science* 326.5951, pp. 399–403.

Friston, K. J., A. P. Holmes, K. J. Worsley, J.-P. Poline, C. D. Frith, and R. S. J. Frackowiak (1994). "Statistical parametric maps in functional imaging: a general linear approach." In: *Human brain mapping* 2.4, pp. 189–210.

Ganjgahi, H., A. M. Winkler, D. C. Glahn, J. Blangero, B. Donohue, P. Kochunov, and T. E. Nichols (2018). "Fast and powerful genome wide association of dense genetic data with high dimensional imaging phenotypes." In: *Nature communications* 9.1, pp. 1–13.

Gaonkar, B. and C. Davatzikos (2012). "Deriving statistical significance maps for SVM based image classification and group comparisons." In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 723–730.

Gimenez, J. R. and J. Zou (2019). "Discovering Conditionally Salient Features with Statistical Guarantees." In: *International Conference on Machine Learning*, pp. 2290–2298.

Goldberger, A. S. (1991). *A course in econometrics*. Harvard University Press.

Gramfort, A., M. Kowalski, and M. Hämäläinen (2012). "Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods." In: *Phys. Med. Biol.* 57.7, pp. 1937–1961.

Gramfort, A., M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen, and M. S. Hämäläinen (2014). "MNE software for processing MEG and EEG data." In: *NeuroImage* 86, pp. 446–460.

Gramfort, A., B. Thirion, and G. Varoquaux (2013). "Identifying predictive regions from fMRI with TV-L1 prior." In: *2013 International Workshop on Pattern Recognition in Neuroimaging*. IEEE, pp. 17–20.

Gramfort, A., G. Varoquaux, and B. Thirion (2012). "Beyond brain reading: randomized sparsity and clustering to simultaneously predict and identify." In: *Machine Learning and Interpretation in Neuroimaging*. Springer, pp. 9–16.

Hämäläinen, M. S. and R. J. Ilmoniemi (1994). "Interpreting magnetic fields of the brain: minimum norm estimates." In: *Medical & Biological Engineering & Computing* 32.1, pp. 35–42.

Hämäläinen, M., R. Hari, R. J. Ilmoniemi, J. Knuutila, and O. V. Lounasmaa (1993). "Magnetoencephalography—theory, instrumentation, and applications to noninvasive studies of the working human brain." In: *Reviews of modern Physics* 65.2, p. 413.

Haufe, S., Frank Meinecke, Kai G., S. Dähne, J.-D. Haynes, B. Blankertz, and F. Bießmann (2014). "On the interpretation of weight vectors of linear models in multivariate neuroimaging." In: *Neuroimage* 87, pp. 96–110.

Haufe, S., V. V. Nikulin, A. Ziehe, K.-R. Müller, and Guido Nolte (2009). "Estimating vector fields using sparse basis field expansions." In: *NeurIPS*. Ed. by D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou. Curran Associates, Inc., pp. 617–624.

Hauk, O., D. G. Wakeman, and R. Henson (2011). "Comparison of noise-normalized minimum norm estimates for MEG analysis using multiple resolution metrics." In: *NeuroImage* 54.3, pp. 1966 –1974.

Haxby, J. V., M. I. Gobbini, M. L. Furey, A. Ishai, J. L. Schouten, and P. Pietrini (2001). "Distributed and overlapping representations of faces and objects in ventral temporal cortex." In: *Science* 293.5539, pp. 2425–2430.

Haynes, J.-D. and G. Rees (2006). "Neuroimaging: decoding mental states from brain activity in humans." In: *Nature Reviews Neuroscience* 7.7, p. 523.

Ho, T. K. (1998). "The random subspace method for constructing decision forests." In: *IEEE Trans. Pattern Anal. Mach. Intell.* 20.8, pp. 832–844.

Hochberg, Y. and A. C. Tamhane (1987a). *Multiple comparison procedures*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc.

– (1987b). *Multiple comparison procedures*. Wiley Series in Probability and Statistics. John Wiley & Sons, Inc.

Hoerl, A. E. and R. W. Kennard (1970). "Ridge regression: Biased estimation for nonorthogonal problems." In: *Technometrics* 12.1, pp. 55–67.

Hoyos-Idrobo, A., G. Varoquaux, Y. Schwartz, and B. Thirion (2018). "FReM–scalable and stable decoding with fast regularized ensemble of models." In: *NeuroImage* 180, pp. 160–172.

Hoyos-Idrobo, Andrés, Yannick Schwartz, Gaël Varoquaux, and Bertrand Thirion (2015). "Improving sparse recovery on structured images with bagged clustering." In: *2015 International Workshop on Pattern Recognition in NeuroImaging*. IEEE, pp. 73–76.

Janson, L. and W. Su (2016). "Familywise error rate control via knockoffs." In: *Electron. J. Stat.* 10.1, pp. 960–975.

Javanmard, A. and A. Montanari (2014). "Confidence Intervals and Hypothesis Testing for High-Dimensional Regression." In: *J. Mach. Learn. Res.* 15, pp. 2869–2909.

Javanmard, Adel, Andrea Montanari, et al. (2018). "Debiasing the lasso: Optimal sample size for gaussian designs." In: *The Annals of Statistics* 46.6A, pp. 2593–2622.

Khan, S. et al. (2018). "Maturation trajectories of cortical resting-state networks depend on the mediating frequency band." In: *NeuroImage* 174, pp. 57 –68.

Kohavi, R. et al. (1995). "A study of cross-validation and bootstrap for accuracy estimation and model selection." In: *Ijcai*. Vol. 14. 2. Montreal, Canada, pp. 1137–1145.

Kriegeskorte, N., R. Goebel, and P. Bandettini (2006). "Information-based functional brain mapping." In: *Proceedings of the National Academy of Sciences* 103.10, pp. 3863–3868.

Kuncheva, L. I. and J. J. Rodríguez (2010). "Classifier ensembles for fMRI data analysis: an experiment." In: *Magnetic resonance imaging* 28.4, pp. 583–593.

Kuncheva, L. I., J. J. Rodríguez, C. O. Plumpton, D. E. J. Linden, and S. J. Johnston (2010). "Random subspace ensembles for fMRI classification." In: *IEEE Trans. Med. Imaging* 29.2, pp. 531–542.

Lauterbur, P. C. (1973). "Image formation by induced local interactions: examples employing nuclear magnetic resonance." In: *nature* 242.5394, pp. 190–191.

Lee, J., D. Sun, Y. Sun, and J. Taylor (2016). "Exact post-selection inference, with application to the lasso." In: *Ann. Statist.* 44.3, pp. 907–927.

Lin, F. H., T. Witzel, S. P. Ahlfors, S. M. Stufflebeam, J. W. Belliveau, and M. S. Hämäläinen (2006). "Assessing and improving the spatial accuracy in MEG source localization by depth-weighted minimum-norm estimates." In: *NeuroImage* 31.1, pp. 160–71.

Liu, H., B. Yu, et al. (2013). "Asymptotic properties of Lasso+ mLS and Lasso+ Ridge in sparse high-dimensional linear regression." In: *Electronic Journal of Statistics* 7, pp. 3124–3169.

Lockhart, R., J. Taylor, R. J. Tibshirani, and R. Tibshirani (2014). "A significance test for the lasso." In: *Annals of statistics* 42.2, p. 413.

Lounici, K., M. Pontil, S. van de Geer, and A. B. Tsybakov (2011). "Oracle inequalities and optimal inference under group sparsity." In: *Ann. Statist.* 39.4, pp. 2164–2204.

Lucka, F., S. Pursiainen, M. Burger, and C. Wolters (2012). "Hierarchical Bayesian inference for the EEG inverse problem using realistic FE head models: Depth localization and source separation for focal primary currents." In: *NeuroImage* 61.4, pp. 1364–1382.

Mandozzi, J. and P. Bühlmann (2016). "Hierarchical Testing in the High-Dimensional Setting With Correlated Variables." In: *J. Amer. Statist. Assoc.* 111.513, pp. 331–343.

Marcus, D., T. Wang, J. Parker, J. Csernansky, J. Morris, and R. Buckner (2007). "Open Access Series of Imaging Studies (OASIS): Cross-sectional MRI Data in Young, Middle Aged, Nondemented, and Demented Older Adults." In: *Journal of Cognitive Neuroscience* 19.9, pp. 1498–1507.

Massias, M., A. Gramfort, and J. Salmon (2018). "Celer: a Fast Solver for the Lasso with Dual Extrapolation." In: *ICML*. Vol. 80, pp. 3315–3324.

Massias, M., S. Vaiter, A. Gramfort, and J. Salmon (2019). "Dual Extrapolation for Sparse Generalized Linear Models." In: *arXiv preprint arXiv:1907.05830*.

Massias, Mathurin, Olivier Fercoq, Alexandre Gramfort, and Joseph Salmon (2018). "Generalized concomitant multi-task lasso for sparse multimodal regression." In: *ICML*, pp. 998–1007.

Matsuura, K. and Y. Okabe (1995). "Selective minimum-norm solution of the biomagnetic inverse problem." In: *IEEE Trans. Biomed. Eng.* 42.6, pp. 608–615. ISSN: 0018-9294.

Mattout, J., M. Pélégrini-Issac, L. Garnero, and H. Benali (2005). "Multivariate source prelocalization (MSP): Use of functionally informed basis functions for better conditioning the MEG inverse problem." In: *NeuroImage* 26.2, pp. 356 –373.

Meinshausen, N. (2008). "Hierarchical testing of variable importance." In: *Biometrika* 95.2, pp. 265–278.

– (2015). "Group bound: confidence intervals for groups of variables in sparse high dimensional regression without assumptions on the design." In: *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pp. 923–945.

Meinshausen, N. and P. Bühlmann (2006). "High-dimensional graphs and variable selection with the Lasso." In: *Ann. Statist.* 34.3, pp. 1436–1462.

– (2010). "Stability Selection." In: *J. R. Stat. Soc. Ser. B Stat. Methodol.* 72, pp. 417–473.

Meinshausen, N., L. Meier, and P. Bühlmann (2009). "P-values for high-dimensional regression." In: *J. Amer. Statist. Assoc.* 104.488, pp. 1671–1681.

Minnier, J., L. Tian, and T. Cai (2011). "A perturbation method for inference on regularized regression estimates." In: *J. Amer. Statist. Assoc.* 106.496, pp. 1371–1382.

Mitra, R. and C.-H. Zhang (2016). "The benefit of group sparsity in group inference with de-biased scaled group Lasso." In: *Electron. J. Stat.* 10.2, pp. 1829–1873.

Molins, A., S. M. Stufflebeam, E. N. Brown, and M. S. Hämäläinen (2008). "Quantification of the benefit from integrating MEG and EEG data in minimum L2-norm estimation." In: *NeuroImage* 42.3, pp. 1069 –1077.

Monti, M. M. (2011). "Statistical analysis of fMRI time-series: a critical review of the GLM approach." In: *Frontiers in human neuroscience* 5, p. 28.

Mourao-Miranda, J., A. L. W. Bokde, C. Born, H. Hampel, and M. Stetter (2005). "Classifying brain states and determining the discriminating activation patterns: support vector machine on functional MRI data." In: *NeuroImage* 28.4, pp. 980–995.

Ndiaye, E., O. Fercoq, A. Gramfort, V. Leclère, and J. Salmon (2017). "Efficient smoothed concomitant Lasso estimation for high dimensional regression." In: *Journal of Physics: Conference Series*. Vol. 904. 1. IOP Publishing, p. 012006.

Nguyen, T.-B., J.-A. Chevalier, and B. Thirion (2019). "ECKO: Ensemble of Clustered Knockoffs for Robust Multivariate Inference on fMRI Data." In: *International Conference on Information Processing in Medical Imaging*. Springer, pp. 454–466.

Nguyen, T.-B., J-A Chevalier, B. Thirion, and S. Arlot (2020). "Aggregation of Multiple Knockoffs." In: *ICML*.

Nichols, T. E. (2012). "Multiple testing corrections, nonparametric methods, and random field theory." In: *Neuroimage* 62, p. 811.

Niedermeyer, E. and F. H. L. da Silva (2005). *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins.

Ning, Y., H. Liu, et al. (2017). "A general theory of hypothesis tests and confidence regions for sparse high dimensional models." In: *The Annals of Statistics* 45.1, pp. 158–195.

Noble, S., D. Scheinost, and R. Constable (Dec. 2019). "Cluster failure or power failure? Evaluating sensitivity in cluster-level inference." In: *NeuroImage* 209, p. 116468.

Norman, K. A., S. M. Polyn, G. J. Detre, and J. V. Haxby (2006). "Beyond mind-reading: multi-voxel pattern analysis of fMRI data." In: *Trends in cognitive sciences* 10.9, pp. 424–430.

Obozinski, G., B. Taskar, and M. I. Jordan (2010). "Joint covariate selection and joint subspace selection for multiple classification problems." In: *Statistics and Computing* 20.2, pp. 231–252.

Ogawa, S., T.-M. Lee, A. R. Kay, and D. W. Tank (1990). "Brain magnetic resonance imaging with contrast dependent on blood oxygenation." In: *proceedings of the National Academy of Sciences* 87.24, pp. 9868–9872.

Ou, W., M. S. Hämaläinen, and P. Golland (2009). "A Distributed Spatio-Temporal EEG/MEG Inverse Solver." In: *NeuroImage* 44.3, pp. 932–946.

Pascual-Marqui, R. (2002). "Standardized low resolution brain electromagnetic tomography (sLORETA): technical details." In: *Methods Find. Exp. Clin. Pharmacology* 24.D, pp. 5–12.

Pedregosa, F. et al. (2011). "Scikit-learn: Machine Learning in Python." In: *J. Mach. Learn. Res.* 12, pp. 2825–2830.

Pereira, F., T. Mitchell, and M. Botvinick (2009). "Machine learning classifiers and fMRI: a tutorial overview." In: *Neuroimage* 45.1, S199–S209.

Pinho, A. L., A. Amadon, T. Ruest, M. Fabre, E. Dohmatob, I. Denghien, C. Ginisty, S. Becuwe-Desmidt, S. Roger, L. Laurier, et al. (2018). "Individual Brain Charting, a high-resolution fMRI dataset for cognitive mapping." In: *Scientific data* 5, p. 180105.

Poldrack, R. A. (2011). "Inferring mental states from neuroimaging data: from reverse inference to large-scale decoding." In: *Neuron* 72.5, pp. 692–697.

Reid, S., R. Tibshirani, and J. Friedman (2016). "A study of error variance estimation in lasso regression." In: *Statistica Sinica*, pp. 35–67.

Richards, J.W., P.E. Freeman, A.B. Lee, and C.M. Schafer (2009). "Exploiting low-dimensional structure in astronomical spectra." In: *The Astrophysical Journal* 691.1, p. 32.

Rizk-Jackson, A., D. Stoffers, S. Sheldon, J. Kuperman, A. Dale, J. Goldstein, J. Corey-Bloom, R. A. Poldrack, and A. R. Aron (2011). "Evaluating imaging biomarkers for neurodegeneration in pre-symptomatic Huntington's disease using machine learning techniques." In: *Neuroimage* 56.2, pp. 788–796.

Schwartz, Y., B. Thirion, and G. Varoquaux (2013). "Mapping coginitive ontologies to and from the brain." In: *Advances in neural information processing systems*, pp. 1673–1681.

Schwartzman, A., R. F. Dougherty, J. Lee, D. Ghahremani, and J. E. Taylor (2009). "Empirical null and false discovery rate analysis in neuroimaging." In: *Neuroimage* 44.1, pp. 71–82.

Smola, A. J. and B. Schölkopf (2004). "A tutorial on support vector regression." In: *Stat. Comput.* 14.3, pp. 199–222.

Stucky, B. and S. van de Geer (2018). "Asymptotic confidence regions for high-dimensional structured sparsity." In: 66.8, pp. 2178–2190.

Taulu, S. (2006). "Spatiotemporal Signal Space Separation method for rejecting nearby interference in MEG measurements." In: *Physics in Medicine and Biology* 51.7, pp. 1759–1769.

Thirion, B. (2016). "Functional neuroimaging group studies." In: *Handbook of Neuroimaging Data Analysis*, pp. 335–354.

Thirion, B., G. Varoquaux, E. Dohmatob, and J.-B. Poline (2014). "Which fMRI clustering gives good brain parcellations?" In: *Frontiers in Neuroscience* 8, p. 167.

Tibshirani, R. J., J. Taylor, R. Lockhart, and R. Tibshirani (2016). "Exact post-selection inference for sequential regression procedures." In: *Journal of the American Statistical Association* 111.514, pp. 600–620.

Tibshirani, R. J et al. (2013). "The lasso problem and uniqueness." In: *Electronic Journal of statistics* 7, pp. 1456–1490.

Tibshirani, R. (1996). "Regression shrinkage and selection via the lasso." In: *J. R. Stat. Soc. Ser. B Stat. Methodol.* 58.1, pp. 267–288.

Van Essen, D. C. et al. (2012). "The Human Connectome Project: a data acquisition perspective." In: *Neuroimage* 62.4, pp. 2222–2231.

Varoquaux, G., A. Gramfort, and B. Thirion (2012). "Small-sample brain mapping: sparse recovery on spatially correlated designs with randomization and clustering." In: *International Conference on Machine Learning*.

Varoquaux, G., Y. Schwartz, R. A. Poldrack, B. Gauthier, D. Bzdok, J.-B. Poline, and B. Thirion (2018). "Atlases of cognition with large-scale human brain mapping." In: *PLoS computational biology* 14.11, e1006565.

Varoquaux, G. and B. Thirion (2014). "How machine learning is shaping cognitive neuroimaging." In: *GigaScience* 3.1, p. 28.

Virtanen, P. et al. (2020). "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." In: *Nature Methods*.

Vlaardingerbroek, M. T. and J. A. Boer (2013). *Magnetic resonance imaging: theory and practice*. Springer Science & Business Media.

Wainwright, M. J. (2009). "Sharp thresholds for High-Dimensional and noisy sparsity recovery using ℓ_1-Constrained Quadratic Programming (Lasso)." In: *IEEE Trans. Image Process.* 55.5, pp. 2183–2202.

Walt, S. Van der, S. C. Colbert, and G. Varoquaux (2011). "The NumPy array: a structure for efficient numerical computation." In: *Computing in Science & Engineering* 13.2, pp. 22–30.

Wang, Y., J. Zheng, S. Zhang, X. Duan, and H. Chen (2015). "Randomized structural sparsity via constrained block subsampling for improved sensitivity of discriminative voxel identification." In: *Neuroimage* 117, pp. 170–183.

Wasserman, L. and K. Roeder (2009). "High-dimensional variable selection." In: *Ann. Statist.* 37.5A, pp. 2178–2201.

Weichwald, S., T. Meyer, O. Özdenizci, B. Schölkopf, T. Ball, and M. Grosse-Wentrup (2015). "Causal interpretation rules for encoding and decoding models in neuroimaging." In: *Neuroimage* 110, pp. 48–59.

Westfall, P. H. and S. S. Young (1993). *Resampling-based multiple testing: Examples and methods for p-value adjustment*. Vol. 279. John Wiley & Sons.

Wipf, D. and S. Nagarajan (2009). "A unified Bayesian framework for MEG/EEG source imaging." In: *NeuroImage* 44.3, pp. 947–966.

Yu, G. and J. Bien (2019). "Estimating the error variance in a high-dimensional linear model." In: *Biometrika* 106.3, pp. 533–546.

Zhang, C.-H. and S. S. Zhang (2014). "Confidence intervals for low dimensional parameters in high dimensional linear models." In: *J. R. Stat. Soc. Ser. B Stat. Methodol.* 76.1, pp. 217–242.

Zhou, Z.-H. (2012). *Ensemble methods: foundations and algorithms*. Chapman and Hall/CRC.

Zimmerman, R. A., W. A. Gibby, and R. F. C. (2012). *Neuroimaging: clinical and physical principles*. Springer Science & Business Media.

van de Geer, S., P. Bühlmann, Y. Ritov, and R. Dezeure (2014). "On asymptotically optimal confidence regions and tests for high-dimensional models." In: *Ann. Statist.* 42.3, pp. 1166–1202.

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Contrôle statistique de modèles parcimonieux en grande dimension

Mots clés : Inférence statistique, grande dimension, neuroimagerie

Résumé : Cette thèse s'intéresse au problème de l'inférence statistique multivariée en grande dimension en présence de données structurées. Plus précisément, étant données une variable cible et un ensemble de variables explicatives, nous souhaitons déterminer les variables explicatives qui sont prédictives conditionnellement aux autres, *i.e.,* nous cherchons à identifier le support dans le modèle prédictif linéaire. Comme nous désirons avoir un contrôle sur l'occurrence de faux positifs, nous nous concentrons sur les méthodes donnant des garanties statistiques. Cette étude s'applique notamment aux problèmes d'inférence sur des images haute-résolution dans lesquels le signal de chaque pixel ou voxel est considéré comme une variable explicative, c'est par exemple le cas en neuro-imagerie ou en astronomie. Cela peut également s'appliquer à d'autres problèmes dans lesquels les variables explicatives sont spatialement structurées comme en génomique par exemple. Pour ce type de données, les méthodes existantes destinées à l'identification de support ne sont pas satisfaisantes car elles manquent de puissance et ont généralement un coût computationnel trop élevé. Par conséquent, le problème est difficile en terme de modélisation statistique mais aussi du point de vue computationnel. Dans ce type de problème, les variables explicatives détiennent une structure spatiale qui peut être exploitée. Par exemple, en neuro-imagerie, une image de cerveau possède une représentation 3D dans laquelle un voxel est très corrélé à ses voisins. Nous proposons notamment la méthode "ensemble of clustered desparsified Lasso" qui combine trois éléments: *i)* une procédure de clustering avec contraintes spatiales pour réduire la dimension du problème en tenant compte de la structure de la donnée; *ii)* une méthode d'inférence statistique appelée "desparsified Lasso" qui peut être déployée sur le problème réduit; et *iii)* une méthode d'ensembling qui agrège les solutions obtenues sur les différents problèmes réduits afin d'éviter de dépendre d'un choix de clustering nécessairement imparfait et arbitraire. Nous proposons également une nouvelle façon de contrôler l'occurrence de faux positifs en intégrant une tolérance spatiale dans ce contrôle. Dans cette étude, nous nous focalisons sur des jeux de donnée de neuro-imagerie, mais les méthodes que nous présentons sont applicables à d'autres domaines qui partagent une configuration semblable.

Title : Statistical control of sparse models in high dimension

Keywords : Statistical inference, high dimension, neuroimaging

Abstract : In this thesis, we focus on the multivariate inference problem in the context of high-dimensional structured data. More precisely, given a set of explanatory variables (features) and a target, we aim at recovering the features that are predictive conditionally to others, *i.e.,* recovering the support of a linear predictive model. We concentrate on methods that come with statistical guarantees since we want to have a control on the occurrence of false discoveries. This is relevant to inference problems on high-resolution images, where one aims at pixel- or voxel-level analysis, *e.g.,* in neuroimaging, astronomy, but also in other settings where features have a spatial structure, *e.g.,* in genomics. In such settings, existing procedures are not helpful for support recovery since they lack power and are generally not tractable. The problem is then hard both from the statistical modeling point of view, and from a computation perspective. In these settings, feature values typically reflect the underlying spatial structure, which can thus be leveraged for inference. For example, in neuroimaging, a brain image has a 3D representation and a given voxel is highly correlated with its neighbors. We notably propose the ensemble of clustered desparsified Lasso (ecd-Lasso) estimator that combines three steps: *i)* a spatially constrained clustering procedure that reduces the problem dimension while taking into account data structure, *ii)* the desparsified Lasso (d-Lasso) statistical inference procedure that is tractable on reduced versions of the original problem, and *iii)* an ensembling method that aggregates the solutions of different compressed versions of the problem to avoid relying on only one arbitrary data clustering choice. We consider new ways to control the occurrence of false discoveries with a given spatial tolerance. This control is well adapted to spatially structured data. In this work, we focus on neuroimaging datasets but the methods that we present can be adapted to other fields which share similar setups.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France