Sparse high dimensional regression in the presence of colored heteroscedastic noise: application to M/EEG source imaging

Mathurin Massias

▶ **To cite this version:**

Mathurin Massias. Sparse high dimensional regression in the presence of colored heteroscedastic noise: application to M/EEG source imaging. Machine Learning [stat.ML]. Telecom Paristech, 2019. English. NNT : . tel-02401628v2

HAL Id: tel-02401628
https://theses.hal.science/tel-02401628v2

Submitted on 20 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

université PARIS-SACLAY

informatics mathematics

TELECOM Paris

IP PARIS

NNT : 2019SACLT053

Thèse de doctorat

Sparse high dimensional regression in the presence of colored heteroscedastic noise: application to M/EEG source imaging

Thèse de doctorat de l'Université Paris-Saclay
préparée à Inria et Télécom Paris

Ecole doctorale n°580 Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Mathématiques et informatique

Thèse présentée et soutenue à Palaiseau, le 04/12/2019, par

MATHURIN MASSIAS

Composition du Jury :

Gabriel Peyré Directeur de Recherche, Ecole Normale Supérieure	Président, Rapporteur
Mark Schmidt Associate Professor, University of British Columbia	Rapporteur
Nelly Pustelnik Chargée de Recherche, ENS de Lyon	Examinatrice
Olivier Fercoq Maître de Conférence, Télécom Paris	Examinateur
Julien Mairal Directeur de Recherche, Inria	Examinateur
Joseph Salmon Professeur, Université de Montpellier	Directeur de thèse
Alexandre Gramfort Directeur de Recherche, Inria	Co-directeur de thèse

Contents

1 Motivation and contributions **17**
- 1.1 Optimization for statistical learning . . . 17
- 1.2 The bio-magnetic inverse problem . . . 30
- 1.3 Contributions . . . 37
- 1.4 Publications . . . 38

I - Faster solvers for sparse Generalized Linear Models **41**

2 Faster solvers for the Lasso: screening, working sets and dual extrapolation **43**
- 2.1 Introduction . . . 44
- 2.2 Duality for the Lasso . . . 45
- 2.3 Gap Safe screening . . . 55
- 2.4 Working sets with aggressive gap screening . . . 55
- 2.5 Experiments . . . 57
- 2.6 Conclusion . . . 63

3 Duality improvements for sparse GLMs **65**
- 3.1 Introduction . . . 66
- 3.2 GLMs, Vector AutoRegressive sequences and sign identification . . . 67
- 3.3 Generalized linear models . . . 71
- 3.4 Working sets . . . 74
- 3.5 Experiments . . . 78
- 3.6 Conclusion . . . 82

II - Concomitant noise estimation for the M/EEG inverse problem **83**

4 Concomitant estimation **85**
- 4.1 Introduction . . . 86
- 4.2 Heteroscedastic concomitant estimation . . . 88
- 4.3 Optimization properties of CLaR and SGCL . . . 90
- 4.4 An analysis of CLaR through smoothing . . . 96
- 4.5 Conclusion . . . 102

5 Experimental validation **103**
- 5.1 Alternative estimators . . . 103
- 5.2 CLaR . . . 107
- 5.3 Preprocessing steps for realistic and real data . . . 114
- 5.4 Time comparison . . . 115

Conclusions and perspectives 117

Appendices 121

A Choice of parameters in Celer 121

B Concomitant estimation 125

Bibliography 135

Remerciements

Malheur à l'homme seul !

Je souhaite d'abord remercier mes directeurs pour m'avoir proposé ce sujet et m'avoir donné la liberté de l'explorer suivant mes idées. En particulier, merci Alexandre de m'avoir appris (par l'exemple !) l'efficacité et le principe de la trottinette, de m'avoir communiqué cette fameuse vitesse initiale et accordé en suivant une si grande confiance. Joseph, merci pour ta rigueur, ta disponibilité, nos séances à la craie à Montpellier, et l'attention constante que tu as portée à mon avenir dans la recherche académique. Selon l'expression consacrée : merci à tous les deux de m'avoir donné le luxe de me plaindre d'être trop encadré.

I thank Gabriel Peyré and Mark Schmidt for accepting to review this thesis. Je remercie également Nelly Pustelnik, Olivier Fercoq, et Julien Mairal d'avoir accepté de constituer mon jury. Je vous en suis très reconnaissant, et espère que la lecture de ce manuscrit vous sera agréable.

Parmi les camarades qui ont rendu ces trois années si plaisantes, Pierre "Forgui" Laforgue mériterait son propre chapitre : une passion commune du gibolin et des mostiquos non démentie depuis presque dix ans ! Merci pour ton inaltérable goût de ce qui est kalos kagathos, de la pédale, d'Aymé le mytho, de l'IØ, des capotos bombatos, des unes de l'Équipe sur les turpitudes du VAR, des recettes de Lil'B et des appels croisés relâchés. Tu n'es jamais le dernier quand il faut s'y filer comme un âne, et mort aux ratagasses. Il me faut aussi distinguer le gourmet Kévin Elgui – posso ? – qui a illuminé mes deux dernières années de thèse par son sens – posso ? – du style, de la formule, du triple quantième avec phase de lune et de la casquette Hermès. Quel dommage que tu fasses du 42. Vous êtes mes deux abominables viveurs, mon camaïeu de rouge, et je vous dis à bientôt sous le figuier aixois ou les oies sauvages. Espérons tout de même que Laf n'aura ni l'un des 19 signes de fracture du moral, ni autres chats à fouetter.

Ce fut un plaisir de partager un bureau, des moments de grâce entre RER et 91.06B, et des séjours en conférence avec Pierre Ablin, digne héritier de Jacques Mayol. "Where are you guys?", et puisqu'il a fallu vivre le traumatisme saclaysien, je suis content que ç'ait été avec toi. Devant témoin, je reconnais ta supériorité à Geoguessr : toi seul sais sentir la côte Ouest en quelques secondes. Toujours en E306, merci à mon prédécesseur dans les études du Lasso, le toujours calme et bienveillant Eugène Ndiaye, qui a essayé de m'apprendre la patience.

Une partie des travaux de cette thèse ont été réalisés conjointement avec Quentin Bertrand. Qobra, merci pour tous ces échanges toujours constructifs qui nous ont permis d'avancer autant ensemble. J'espère que nous continuerons à explorer ces idées et d'autres encore longtemps, bon courage pour le NeuroImage.

Merci à Lambert et Jache, mes petits jeunes préférés : bon courage pour la fin, il ne vous reste plus qu'à apprendre à perdre. Tragique ! Moins jeunes, Quentin et Charles, entre lancers de hache et canassons, auront été les véritables *impact players* de cette thèse : le sang frais à la soixantième qui fait la différence. Au sein de Télécom, merci à Simon, Adil l'homme par qui le scandale arrive, Anas, Sholom même s'il ne met pas son nom sur ses TP, Kiki "mi-figue mi-mangue", Lucho le bandido et Guillaume "Big Bob" Papa pour sa résilience inébranlable face aux piquettes à la coinche. Merci aux vieux du FIAP, Romain, Anna, Nicolas et Maël, pour leurs bons plans qui nous ont tant manqué par la suite. A warm thank you goes to the whole Parietal team for welcoming me after a few months of PhD, with special thoughts for Patricio, Antonia, Hamza, Thomas, Marine, Pierre, Jérôme and others with whom I got to spend a bit of time outside of work. Merci à Thomas Moreau de nous avoir rendu la vie facile en ouvrant la voie, et aux inséparables La Tour et Tom Dupré qui ont toujours su remettre les choses en perspective avec malice. J'ai profité de très bons moments à Seattle avec Evguenii Chzhen et Vincent Roulet ; j'ai hâte d'en passer d'autres ailleurs.

Merci à Matthieu Durut, Alain Durmus et Michal Valko pour leurs si précieux conseils de recherche qui m'ont énormément servi, et continueront de le faire pour longtemps.

I express my gratitude to Taiji Suzuki for hosting me during three months in his team in Tokyo, and to Akiko Takeda, Michael Metel et Pierre-Louis Poirion for welcoming me the way they did. It was a real pleasure to get to work with Boris Muzellec afterwards : I owe you a lot. I thank the Gdr ISIS, MOA and MIA, the STIC doctoral school, and the European Research Council for their financial support, which has enabled me to spend such a privileged PhD.

J'ai une petite pensée pour Gaïus et Raymond, pour le réconfort qu'ils peuvent m'apporter dans l'adversité. Merci à Orts ma petite craquotte, à Célius le Régis des hôtes de ces bois, et au Professeur Benhamou pour ses magistrales leçons.

Enfin, merski à Maud pour qui ça n'a pas dû être facile tous les jours, pour tes petites analyses décortiquantes, ton risotto et l'ouverture salvatrice à d'autres mondes. Buona fortuna.

Abstract

Understanding the functioning of the brain under normal and pathological conditions is one of the challenges of the 21$^{\text{st}}$ century. In the last decades, neuroimaging has radically affected clinical and cognitive neurosciences. Amongst neuroimaging techniques, magneto- and electroencephalography (M/EEG) stand out for two reasons: their non-invasiveness, and their excellent time resolution. Reconstructing the neural activity from the recordings of magnetic field and electric potentials is the so-called *bio-magnetic inverse problem*.

Because of the limited number of sensors, this inverse problem is severely ill-posed, and additional constraints must be imposed in order to solve it. A popular approach, considered in this manuscript, is to assume spatial *sparsity* of the solution: only a few brain regions are involved in a short and specific cognitive task. Solutions exhibiting such a neurophysiologically plausible sparsity pattern can be obtained through $\ell_{2,1}$-penalized regression approaches. However, this regularization requires to solve time-consuming high-dimensional and non-smooth optimization problems, with iterative (block) proximal gradients solvers.

Additionally, M/EEG recordings are usually corrupted by strong non-white noise, which breaks the classical statistical assumptions of inverse problems. To circumvent this, it is customary to whiten the data as a preprocessing step, and to average multiple repetitions of the same experiment to increase the signal-to-noise ratio. Averaging measurements has the drawback of removing brain responses which are not phase-locked, *i.e.,* do not happen at a fixed latency after the stimuli presentation onset.

In this work, we first propose speed improvements of iterative solvers used for the $\ell_{2,1}$-regularized bio-magnetic inverse problem. Typical improvements, screening and working sets, exploit the sparsity of the solution: by identifying inactive brain sources, they reduce the dimensionality of the optimization problem. We introduce a new working set policy, derived from the state-of-the-art Gap safe screening rules. In this framework, we also propose duality improvements, yielding a tighter control of optimality and improving feature identification techniques. This dual construction extrapolates on an asymptotic Vector AutoRegressive regularity of the dual iterates, which we connect to manifold identification of proximal algorithms. Beyond the $\ell_{2,1}$-regularized bio-magnetic inverse problem, the proposed methods apply to the whole class of sparse Generalized Linear Models.

Second, we introduce new concomitant estimators for multitask regression. Along with the neural sources estimation, concomitant estimators jointly estimate the noise covariance matrix. We design them to handle non-white Gaussian noise, and to exploit the multiple repetitions nature of M/EEG experiments. Instead of averaging the observations, our proposed method, CLaR, uses them all for a better estimation of the noise.

The underlying optimization problem is jointly convex in the regression coefficients and the noise variable, with a "smooth + proximable" composite structure. It is therefore solvable via standard alternate minimization, for which we apply the improvements detailed in the first part. We provide a theoretical analysis of our objective function, linking it to the smoothing of Schatten norms. We demonstrate the benefits of the proposed approach for source localization on real M/EEG datasets.

Our improved solvers and refined modeling of the noise pave the way for a faster and more statistically efficient processing of M/EEG recordings, allowing for interactive data analysis and scaling approaches to larger and larger M/EEG datasets.

Notation

$\triangleq$	Equal by definition	
$[d]$	Set of integers from 1 to d included	
$\mathcal{Y}^{\mathcal{X}}$	Set of functions from $\mathcal{X}$ to $\mathcal{Y}$	
$\mathbb{R}^{d_1 \times d_2}$	Set of real matrices of size d_1 by d_2	
Id_n	Identity matrix in $\mathbb{R}^{n \times n}$	
$A_{i:}$	i^{th} row of matrix A	
$A_{:j}$	j^{th} column of matrix A	
$\operatorname{Tr} A$	Trace of $A \in \mathbb{R}^{d \times d}$	$\operatorname{Tr} A = \sum_{i=1}^{d} A_{ii}$
$A^\top$	Transpose of matrix A	
$A^\dagger$	Moore-Penrose pseudo-inverse of matrix A	
$\operatorname{supp}(x)$	Support of $x \in \mathbb{R}^d$	$\Big\{ j \in [d] \,:\, x_j \neq 0 \Big\}$
$\|\cdot\|$	Euclidean norm on vectors and matrices	
$\|\cdot\|_0$	ℓ_0 pseudo-norm on vectors	$\|x\|_0 = \big\lvert \operatorname{supp} x \big\rvert$
$\|\cdot\|_p$	ℓ_p-norm on vectors for $p \in [1, +\infty]$	
$\mathcal{B}_p$	Unit ball of ℓ_p-norm	
$\mathcal{S}_{++}^n$	Positive definite matrices of size $n \times n$	
$\mathcal{S}_{+}^n$	Semipositive definite matrices of size $n \times n$	
$\|\cdot\|_{\mathscr{S},p}$	Schatten p-norm on matrices for $p \in [1, +\infty]$	
$\mathcal{B}_{\mathscr{S},p}$	Unit ball of Schatten p-norm	
$\|\cdot\|_{2,1}$	Row-wise $\ell_{2,1}$-mixed norm on matrices	$\|A\|_{2,1} = \sum_{j=1}^{p} \|A_{j:}\|$
$\|\cdot\|_{2,\infty}$	Row-wise $\ell_{2,\infty}$-mixed norm on matrices	$\|A\|_{2,\infty} = \max_{j \in [p]} \|A_{j:}\|$
$\langle \cdot, \cdot \rangle_S$	Vector scalar product weighted by $S \in \mathcal{S}_{++}^n$	$\langle x, y \rangle_S = x^\top S y$
$\|\cdot\|_S$	Mahalanobis matrix norm induced by $S \in \mathcal{S}_{++}^n$	$\|A\|_S = \sqrt{\operatorname{Tr}(A^\top S A)}$
$\|\cdot\|_2$	Spectral norm on matrices	

$a \vee b$	Maximum of real numbers a and b	
$a \wedge b$	Minimum of real numbers a and b	
$(a)_+$	Positive part of $a \in \mathbb{R}$	$a \vee 0$
$\mathrm{sign}(x)$	Sign of $x \in \mathbb{R}$	$\mathrm{sign}(x) = \frac{x}{\lvert x \rvert}$ and $\frac{0}{0} = 0$
$\odot$	Entrywise product between vectors	$(x \odot y)_j = x_j y_j$
$\mathbf{0}$	Vector or matrix of zeros	
$\mathbf{1}$	Vector or matrix of ones	
$\mathrm{ST}(x, \tau)$	Soft-thresholding of $x \in \mathbb{R}^d$ at level $\tau > 0$	$(\mathrm{sign}(x_j)(\lvert x_j \rvert - \tau)_+)_{j \in [d]}$
$\mathrm{BST}(A, \tau)$	Block soft-thresholding of $A \in \mathbb{R}^{d \times d'}$ at level $\tau > 0$	$(1 - \tau / \lVert A \rVert)_+ \cdot A$
$\Pi_{\mathcal{C}}$	Euclidean projection onto convex set $\mathcal{C}$	
$\iota_{\mathcal{C}}$	Indicator function of set $\mathcal{C}$	Definition 1.6
$f \,\square\, g$	Infimal convolution of f and g	Definition 1.7
f^*	Fenchel-Legendre transform of f	Definition 1.8
∂f	Subdifferential of f	Definition 1.12
$\mathrm{dom}\, f$	Domain of f	$\{x : f(x) < +\infty\}$

Model specific

$X \in \mathbb{R}^{n \times p}$	Design matrix
$\mathbf{x}_i \in \mathbb{R}^p$	i^{th} row of the design matrix
$y \in \mathbb{R}^n$	Observation vector
$Y \in \mathbb{R}^{n \times q}$	Observation matrix in multitask framework
$\Sigma \in \mathcal{S}^n_{++}$	Noise covariance matrix
$S \in \mathcal{S}^n_{++}$	Square root of noise covariance matrix

For two matrices S_1 and S_2 in $\mathbb{R}^{n \times n}$ we write $S_1 \succeq S_2$ (*resp.* $S_1 \succ S_2$) for $S_1 - S_2 \in \mathcal{S}^n_+$ (*resp.* $S_1 - S_2 \in \mathcal{S}^n_{++}$). When we write $S_1 \succeq S_2$ we implicitly assume that both matrices belong to $\mathcal{S}^n_+$.

There is an obvious notation clash: p refers to both the number of features and the index of ℓ_p or Schatten p-norms: we ask for the reader's forgiveness.

As much as possible, exponents between parenthesis (*e.g.,* $\beta^{(t)}$) denote iterates and subscripts (*e.g.,* β_j) denote vector entries.

We extend the small-o notation to vector valued functions in the following way: for $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^n$, $f = o(g)$ if and only if $\lVert f \rVert = o(\lVert g \rVert)$, *i.e.,* $\lVert f \rVert / \lVert g \rVert$ tends to 0 when $\lVert g \rVert$ tends to 0.

1 Motivation and contributions

– *"Andrea, com'era la mamma ?*
– *Non te la ricordi ?*
– *Prima si, adesso mica tanto."*

Contents

1.1 Optimization for statistical learning . . . 17
1.1.1 Statistical learning . . . 17
1.1.2 Regularization and sparsity . . . 21
1.1.3 Convex optimization tools . . . 26
1.2 The bio-magnetic inverse problem . . . 30
1.2.1 Basis of M/EEG . . . 30
1.2.2 Solving the inverse problem . . . 35
1.3 Contributions . . . 37
1.4 Publications . . . 38

1.1 Optimization for statistical learning

1.1.1 Statistical learning

Let Z be a random variable in a domain $\mathcal{Z}$. A statistical learning task is to find, in a certain set of models $\mathcal{H}$ called hypothesis class, the most suitable one. Formally, for a loss function $\ell : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}_+$, the best model minimizes the expected loss:

$$\arg\min_{\phi \in \mathcal{H}} \mathbb{E}[\ell(\phi, \mathsf{Z})] \ . \tag{1.1}$$

This framework encompasses tasks such as dimensionality reduction, classification, regression, clustering or feature selection (Shalev-Shwartz and Ben-David, 2014). In this thesis, we are interested in the *prediction* learning task: we wish to infer the relationship between a random variable X and a target random variable Y, taking values in sets $\mathcal{X}$ and $\mathcal{Y}$ respectively. In that case, $\mathsf{Z} = (\mathsf{X}, \mathsf{Y})$, $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ and the set of models $\mathcal{H}$ is a subset of $\mathcal{Y}^{\mathcal{X}}$, reflecting a priori knowledge about this dependency (Hastie et al., 2009). As a loss function, we use $\ell(\phi, (x, y)) = L(\phi(x), y)$ where $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ measures the discrepancy between two values in $\mathcal{Y}$. Problem (1.1) then becomes:

$$\arg\min_{\phi \in \mathcal{H}} \mathbb{E}[L(\phi(\mathsf{X}), \mathsf{Y})] \ . \tag{1.2}$$

Unfortunately, this expectation is generally impossible to compute since the joint distribution of X and Y is unknown. A most common setting is when a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i\in[n]} \in (\mathcal{X} \times \mathcal{Y})^n$ is available, comprising n independent samples drawn from the joint distribution of X and Y. The unavailable $\mathbb{E}[L(\phi(\mathsf{X}), \mathsf{Y})]$ can be approximated by the empirical mean $\frac{1}{n}\sum_{i=1}^n L(\phi(x_i), y_i)$, and a proxy for the best model can be obtained in the Empirical Risk Minimization (ERM) framework, by solving:

$$\arg\min_{\phi\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} L(\phi(x_i), y_i) \ . \tag{1.3}$$

We refer to the case where $\mathcal{H} = \{x \mapsto h(x;\beta) \ : \ \beta \in \mathbb{R}^d\}$, with $h : \mathcal{X} \times \mathbb{R}^d \to \mathcal{Y}$ fixed, as the (finite dimensional) parametric case. In this case, learning the optimal function ϕ reduces to learning the optimal parameter vector β.

We will focus on a particular class of parametric statistical models, called Generalized Linear Models (GLMs, introduced in McCullagh and Nelder 1989). For simplicity of the presentation, we now assume that $\mathcal{X} = \mathbb{R}^p$ and $\mathcal{Y} \subset \mathbb{R}$ (the later used *multitask* framework, $\mathcal{Y} = \mathbb{R}^q$, can be addressed easily at the price of heavier notation). First, let us introduce an *exponential family*, that is, a family of parametric probability densities (or mass functions), taking the form:

$$\{f(\upsilon;\theta) = c(\upsilon)\exp(\eta(\theta)^\top T(\upsilon) - \kappa(\theta)) \ : \ \theta \in \Theta\} \ , \tag{1.4}$$

and such that the support of $f(\cdot;\theta)$ does not depend on θ. The function η is called the *natural parameter* of the family, T is the *sufficient statistic*, κ the *cumulant function*, Θ the parameter space and c reflects the integrating measure. Exponential families provide a convenient unifying framework to analyze a variety of commonly used distributions: Gaussian, Poisson, Bernoulli, multinomial, exponential, etc.

Example 1.1 (Real Gaussian)**.** *If $\Upsilon \sim \mathcal{N}(\mu, \sigma^2)$, its density at $\upsilon \in \mathbb{R}$ is:*

$$\frac{1}{\sqrt{2\pi\sigma^2}}\exp\Big(-\frac{(\upsilon-\mu)^2}{2\sigma^2}\Big) = \frac{1}{\sqrt{2\pi}}\exp\Big(\frac{\mu}{\sigma^2}\upsilon - \frac{1}{2\sigma^2}\upsilon^2 - \frac{1}{2\sigma^2}\mu^2 - \log\sigma\Big) \ , \tag{1.5}$$

and it is easy to check that with:

$$\begin{cases} c(\upsilon) = 1/\sqrt{2\pi} \ , \\ \theta = (\theta_1, \theta_2) = \Big(\frac{\mu}{\sigma^2}, -\frac{1}{2\sigma^2}\Big) \ , \\ \eta(\theta) = \theta \ , \\ T(\upsilon) = (\upsilon, \upsilon^2) \ , \\ \kappa(\theta) = -\frac{\theta_1^2}{4\theta_2} - \frac{1}{2}\log(-2\theta_2) = \frac{1}{2\sigma^2}\mu^2 + \log\sigma \ , \\ \Theta = \mathbb{R} \times \left]-\infty, 0\right[\ , \end{cases} \tag{1.6}$$

Equation (1.5) fits the form of (1.4). Moreover, notice that:

$$\nabla\kappa(\theta) = \left(-\frac{\theta_1}{2\theta_2}, \frac{\theta_1^2}{4\theta_2^2} - \frac{1}{2\theta_2}\right) = \left(\mu, \mu^2 + \sigma^2\right) = \mathbb{E}[T(\Upsilon)] \ . \tag{1.7}$$

Now consider that σ is known; we can change the parametrization to:

$$\begin{cases} c(\upsilon) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}\upsilon^2\right) \ , \\ \theta = \frac{\mu}{\sigma^2} \ , \\ \eta(\theta) = \theta \ , \\ T(\upsilon) = \upsilon \ , \\ \kappa(\theta) = \frac{\sigma^2}{2}\theta^2 \ , \\ \Theta = \mathbb{R} \ . \end{cases} \tag{1.8}$$

In that case, T is the identity, i.e., *Υ itself is a sufficient statistic, and one can easily check that $\kappa'(\theta) = \mathbb{E}[\Upsilon]$.*

From now on, we restrict ourselves to distributions for which T is the identity. It turns out that we always have[1] $\nabla\kappa(\theta) = \kappa'(\theta) = \mathbb{E}[\Upsilon]$, and $\kappa''(\theta) = \mathrm{Var}[\Upsilon] > 0$. This means that the mapping $\theta \mapsto \mu \triangleq \mathbb{E}[\Upsilon]$ is one-one, which allows to parametrize the distribution not with θ, but with μ (*moment parametrization*). In this situation, to postulate that (X, Y) follows a GLM is to assume that for every $x \in \mathbb{R}^p$ the distributions of $\mathsf{Y}|\mathsf{X} = x$ (emphasis: not of Y) belong to a common exponential family, and that the parameter $\mu \triangleq \mathbb{E}[\mathsf{Y}|\mathsf{X} = x]$ is equal to $\psi^{-1}(\beta^\top x)$ (or $\theta = (\kappa')^{-1} \circ \psi^{-1}(x^\top\beta)$), for a fixed parameter β and a *response function* ψ.

The hypotheses of a GLM are summarized by:

$$f_{\mathsf{Y}|\mathsf{X}=x}(y;\beta) = c(y)\exp\left\{\eta\Big((\kappa')^{-1} \circ \psi^{-1}(x^\top\beta)\Big)y - \kappa\left((\kappa')^{-1} \circ \psi^{-1}(x^\top\beta)\right)\right\} \ . \tag{1.9}$$

Equation (1.9) evidences the two choices which characterize a GLM: the exponential family via c, η, and κ, and the response function ψ. To model the data, the choice of the exponential family usually depends on the nature of Y: continuous unbounded data can be modeled by a Gaussian, count data by a Poisson distribution, intervals by an exponential distribution, etc. The response function ψ is usually chosen so that it matches the constraints on the exponential family's mean, but different choices can be made for the same Y. Note that there exists a *canonical* choice of ψ: $\psi = (\kappa')^{-1}$, resulting in $\theta = x^\top\beta$.

Performing parameter inference in this setting leads to a variety of popular losses for ERM, with the key property that the Maximum Likelihood Estimator (MLE) problem is a convex one (Pitman, 1936), and therefore the corresponding ERM also is. For the previously introduced dataset $\mathcal{D}$, denoting by $\theta^{(i)}$ the parameters of the distributions of $\mathsf{Y}|\mathsf{X} = x_i$, independence of the samples leads to a log-likelihood equal to:

$$\ell(\theta^{(1)}, \dots, \theta^{(n)}|\mathcal{D}) = \sum_{i=1}^{n} \log c(y_i) + \sum_{i=1}^{n} \eta(\theta^{(i)})y_i - \kappa(\theta^{(i)}) \ . \tag{1.10}$$

We can write this as a function of β only, and the parameter $\hat{\beta}$ leading to the MLE $(\hat{\theta}^{(1)}, \dots, \hat{\theta}^{(n)})$, is:

$$\operatorname*{arg\,min}_{\beta\in\mathbb{R}^p} \sum_{i=1}^{n} \kappa \circ (\kappa')^{-1} \circ \psi^{-1}(x_i^\top\beta) - \eta \circ (\kappa')^{-1} \circ \psi^{-1}(x_i^\top\beta)y_i \ , \tag{1.11}$$

which is a finite dimensional, parametric instance of the ERM Problem (1.3) for $L(\hat{y}, y) = \kappa \circ (\kappa')^{-1}(\hat{y}) - \eta \circ (\kappa')^{-1}(\hat{y})y$, and $\mathcal{H} = \{x \mapsto \psi^{-1}(\beta^\top x) \ : \ \beta \in \mathbb{R}^p\}$.

[1] the general formula is $\nabla\kappa(\theta) = \mathrm{Jac}_\eta(\theta)^\top\,\mathbb{E}[T(\Upsilon)]$ where $\mathrm{Jac}_\eta(\theta)$ is the Jacobian matrix of η at θ

Example 1.2 (Bernoulli variable)**.** *In this example,* $\mathcal{X} = \mathbb{R}^p$ *and* $\mathcal{Y} = \{0, 1\}$*. The probability mass function of a Bernoulli variable of mean* $\mu \in]0, 1[$ *is:*

$$p(y; \mu) = \mu^y + (1 - \mu)^{1-y} = \exp\left(y \log\left(\frac{\mu}{1-\mu}\right) + \log(1 - \mu)\right) \ , \tag{1.12}$$

which belongs to an exponential family, with:

$$\begin{cases} c(y) = 1 \ , \\ \theta = \log\left(\frac{\mu}{1-\mu}\right) \ , \\ \eta(\theta) = \theta \ , \\ T(y) = y \ , \\ \kappa(\theta) = \log(1 + e^\theta) = -\log(1 - \mu) \ , \\ \Theta =]0, +\infty[\ . \end{cases} \tag{1.13}$$

So if we postulate that $\mathsf{Y}|\mathsf{X} = x$ *is a Bernoulli random variable of mean* $\mu = \kappa'(x^\top \beta) = 1/(1 - e^{-x^\top \beta})$*, we have a GLM with canonical response function, and Problem (1.11) can be written as:*

$$\arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \kappa(\beta^\top x_i) - \eta(\beta^\top x_i) y_i = \arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \log(1 + \exp(x_i^\top \beta)) - y_i x_i^\top \beta \ , \tag{1.14}$$

which is the logistic regression *ERM. Alternatively, we could postulate that* $\mathsf{Y}|\mathsf{X} = x$ *is a Bernoulli random variable of mean* $\mu = \Phi(x^\top \beta)$*, with* Φ *the cumulative distribution function of a standard Gaussian. This is the* probit *model, with MLE accessible via:*

$$\begin{aligned} &\arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \log\left(1 + \frac{\Phi(x_i^\top \beta)}{1 - \Phi(x_i^\top \beta)}\right) - \log\left(\frac{\Phi(x_i^\top \beta)}{1 - \Phi(x_i^\top \beta)}\right) y_i \\ &= \arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n -y_i \log(\Phi(x_i^\top \beta)) - (1 - y_i) \log(1 - \Phi(x_i^\top \beta)) \ . \end{aligned} \tag{1.15}$$

The examples derived above explain the ubiquity of problems of the form:

$$\arg\min \sum_{i=1}^n f_i(x_i^\top \beta) \ , \tag{1.16}$$

that we consider in Part I. Finally, the following example is of primary importance for the multitask case which arises in our application.

Example 1.3 (Multivariate Gaussian)**.** *Let* $\mathcal{X} = \mathbb{R}^p$*, let* $\mathcal{Y} = \mathbb{R}^q$*. Consider the density of a multivariate Gaussian of mean* $\mu \in \mathbb{R}^q$ *and covariance* $\Sigma \in \mathcal{S}_{++}^q$*, evaluated at* $z \in \mathbb{R}^q$*:*

$$\begin{aligned} &\frac{1}{(2\pi)^{q/2} (\det \Sigma)^{1/2}} \exp\left(-\frac{1}{2}(z - \mu)^\top \Sigma^{-1} (z - \mu)\right) \\ &= \frac{1}{(2\pi)^{p/2}} \exp\left(z^\top \Sigma^{-1} \mu - \frac{1}{2} z^\top \Sigma^{-1} z - \frac{1}{2} \mu^\top \Sigma^{-1} \mu - \frac{1}{2} \log\det \Sigma\right) \\ &= \frac{1}{(2\pi)^{p/2}} \exp\left(z^\top \Sigma^{-1} \mu - \frac{1}{2} \operatorname{Tr} \Sigma^{-1} z z^\top - \frac{1}{2} \mu^\top \Sigma^{-1} \mu - \frac{1}{2} \log\det \Sigma\right) \ . \end{aligned} \tag{1.17}$$

It belongs to an exponential family, with $\theta = (\Sigma^{-1}\mu, -\frac{1}{2} \operatorname{Vec} \Sigma^{-1})$*,* $T(z) = (z, \operatorname{Vec} z z^\top)$ *where* Vec *is the column-wise vectorization operator.*

Finally, going back to Example 1.1 leads to the most well-known instance of MLE: Ordinary Least Squares (OLS). For $\mathcal{X} = \mathbb{R}^p$ and $\mathcal{Y} = \mathbb{R}$, let us postulate that:

$$\mathsf{Y} = \mathsf{X}^\top \beta^* + \mathsf{E} \ , \tag{1.18}$$

where E is a real-valued Gaussian of law $\mathcal{N}(0, \sigma^2)$, independent from X, and $\beta^* \in \mathbb{R}^p$ is the true parameter vector. Then following Example 1.1, we have a GLM with:

$$\begin{cases} \theta = x^\top \beta \ , \\ \eta(\theta) = \frac{\theta}{\sigma^2} \ , \\ \kappa(\theta) = \frac{\theta^2}{2\sigma^2} \ , \\ \psi(u) = u \ , \end{cases} \tag{1.19}$$

and the MLE derived in Problem (1.11) reads:

$$\begin{aligned} \arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n \kappa(\beta^\top x_i) - \eta(\beta^\top x_i) y_i &= \arg\min_{\beta \in \mathbb{R}^p} \frac{1}{\sigma^2} \sum_{i=1}^n \frac{1}{2} (x_i^\top \beta)^2 - x_i^\top \beta y_i \\ &= \arg\min_{\beta \in \mathbb{R}^p} \sum_{i=1}^n (y_i - x_i^\top \beta)^2 \\ &= \arg\min_{\beta \in \mathbb{R}^p} \frac{1}{2} \left\| y - X\beta \right\|^2 \ , \end{aligned} \tag{1.20}$$

where we have introduced the *design matrix* $X \triangleq (x_1^\top, \dots, x_n^\top)^\top \in \mathbb{R}^{n \times p}$, and the *observation vector* $y \triangleq (y_1, \dots, y_n) \in \mathbb{R}^n$. Dating back to Legendre and Gauss (Legendre (1805); Gauss (1809), see also Plackett (1972) for a discussion on this discovery), least squares are extremely popular, and optimal amongst linear unbiased estimators (they have the lowest variance in this class[2]), but they also suffer from defects in certain settings which we detail in the following.

1.1.2 Regularization and sparsity

Consider n realizations of the linear model $\mathsf{Y} = \mathsf{X}\beta^* + \mathsf{E}$, written in vector form as above: $y = X\beta^* + \varepsilon$ ($\varepsilon \in \mathbb{R}^n$ with entries i.i.d. $\mathcal{N}(0, \sigma^2)$, and σ known). Assuming that $p \leq n$ and $\operatorname{rank} X = p$, the OLS estimator is uniquely defined and reads:

$$\hat{\beta}^{\mathrm{OLS}} = (X^\top X)^{-1} X^\top y = \beta^* + (X^\top X)^{-1} X^\top \varepsilon \ . \tag{1.21}$$

The expected (averaged) prediction error is:

$$\begin{aligned} \mathbb{E}[\tfrac{1}{n} \| X\hat{\beta}^{\mathrm{OLS}} - X\beta^* \|^2] &= \mathbb{E}[\tfrac{1}{n} \| X(X^\top X)^{-1} X^\top \varepsilon \|^2] \\ &= \sigma^2 \frac{p}{n} \ , \end{aligned} \tag{1.22}$$

which does not go to 0 when $n \to +\infty$, unless $p = o(n)$. This is problematic, as we may want to consider cases where p and n go to infinity together and at the same speed.

Other issues arise for MLE, when the number of parameters p outgrows the number of samples n: the solution of Problem (1.20) is not unique, meaning that multiple models exist, all of them perfectly fitting the training data when $\operatorname{rank} X = n$. In

[2] even when the noise is not Gaussian, provided the noise variance is constant across observations

that case, which model must be chosen? In addition, perfectly fitting the data is not necessarily a desirable property, as it may lead to poor generalization performance on new data – recent analysis (Hastie et al., 2019) may qualify this interpretation. When data is scarce, empirical risk minimization turns out to be insufficient, and we turn to *regularization*: instead of looking for the model minimizing the sole datafitting criterion, some constraint is added to the optimization problem, this constraint reflecting prior belief about the desired model.

In the case of least squares, with $\mathcal{R} : \mathbb{R}^p \to \mathbb{R}$, this takes the form:

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{2}\|y - X\beta\|^2 \quad s.t. \quad \mathcal{R}(\beta) \leq 0 \ . \tag{1.23}$$

A seminal choice for $\mathcal{R}$ is $\|\cdot\| - \tau$, with $\tau > 0$. It is noteworthy that the three following problems are equivalent:

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{2}\|y - X\beta\|^2 + \frac{\lambda}{2}\|\beta\|^2 \ , \tag{1.24}$$

$$\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|^2 \quad s.t. \quad \|\beta\| \leq \tau \ , \tag{1.25}$$

$$\min_{\beta \in \mathbb{R}^p} \|\beta\|^2 \quad s.t. \quad \|y - X\beta\| \leq \epsilon \ . \tag{1.26}$$

Problems (1.24), (1.25) and (1.26) are respectively known as Tikhonov, Ivanov and Morozov regularization (Tikhonov, 1943; Ivanov, 1976; Morozov, 1984), and also as Ridge regression (Hoerl and Kennard, 1970). They are equivalent in the sense that for each (positive) value of one parameter amongst λ, τ or ϵ, there exist values of the remaining two such that the three problems share the same solution. Each of these provides a different view on ℓ_2 regularization: formulation (1.26) looks for the approximate solution to a linear system with the minimum ℓ_2 norm. Formulation (1.25) looks for least squares solutions, with constrained ℓ_2 norm. Formulation (1.24) is the most widely employed, but perhaps the less explicit one: while it is clear that in the other formulations, norms can be squared or not in the objective functions, or squared in the constraints provided τ or ϵ are squared, it is not easy to see that, up to another choice of λ, Problem (1.24) retains the same solution if one of the squared norms is replaced by a plain norm. In this sense, it is somehow misleading to talk about *squared* ℓ_2 regularization: up to a change of value for λ, plain ℓ_2 regularization has the same effect. It is only the geometry of the level lines of the regularizer which matter, and those are not affected by squaring. The square in Tikhonov regularization is only used for practical reasons, as it makes the regularizer smooth.

Tikhonov regularization limits the magnitude of the estimate. The type of regularization considered in this work is different: for reasons detailed in Section 1.2, we want to favor simple and interpretable solutions. Generally speaking, the idea that this kind of models should be preferred can be dated back to Ockham's Razor (Ockham, 1319), and in a modern paradigm to Wrinch and Jeffreys' simplicity principle:

> *"It is justifiable to prefer a simple law to a more complex one that fits our observations slightly better."* (Wrinch and Jeffreys (1921))

In the context of GLMs, an application of this principle is to perform variable (or feature) selection: search for models which do not include all the potential p variables, but only a small subset of them. In mathematical terms, the estimator $\hat{\beta}$ should be

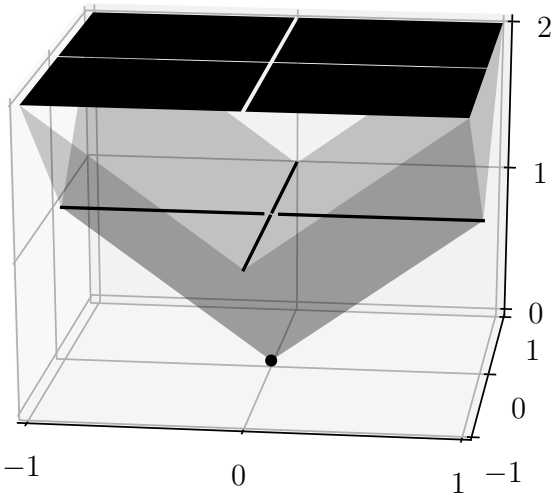


Figure 1.1 – Values of $\|\cdot\|_0$ (black) and $\|\cdot\|_1$ (grey) on $\mathcal{B}_\infty$ in dimension 2. The ℓ_1 norm is greater than any other convex minorant of $\|\cdot\|_0$ on this set.

sparse: $\|\hat{\beta}\|_0 \ll p$. Sparse solutions provide more interpretable models, since it is clear that variables whose coefficients are 0 have no effect on the target variable.

The idea to favor sparse models has been applied in various fields: finance (portfolio selection, Markowitz 1952), image processing (wavelet thresholding, Donoho and Johnstone 1994) or statistics (under the name best subset selection, see Miller (2002) for a review). In geophysics, the work of Santosa and Symes (1986) stands out as the first case of ℓ_1 penalized least-squares, central to this manuscript (earlier, Claerbout and Muir (1973); Taylor et al. (1979) had used the ℓ_1 norm both as datafitting term and regularizer).

However desirable sparse models may be, solving the underlying optimization problems is non trivial. For example, instead of looking for the minimum ℓ_2 norm approximate solution to a linear system as in Problem (1.26), one may seek the sparsest one by solving:

$$\min_{\beta \in \mathbb{R}^p} \|\beta\|_0 \quad s.t. \quad \|y - X\beta\| \leq \epsilon \ . \tag{1.27}$$

Unfortunately, Natarajan (1995) showed that this problem is NP-hard and hardly tractable when p is large, because the objective function $\|\cdot\|_0$ is not convex. Approximate solutions can nevertheless be computed. Forward and backward feature selection approaches exist (Efroymson, 1960), either starting from a null vector and iteratively adding the feature improving the datafit the most, or starting with an OLS solution and progressively removing the features less contributing to the model; bidirectional approaches can both add and remove features (Zhang, 2011). This is for example the spirit of the Matching Pursuit (Mallat and Zhang, 1993) and Orthogonal Matching Pursuit (Pati et al., 1993; Tropp, 2006) algorithms. Still, stepwise selection suffers from issues: small changes in the data can result in large differences in models and the estimator is not guaranteed to be sparse (Chen et al., 1998, Section 2.3.2).

Another massively followed route to sparsity has been the use of convex surrogates for the ℓ_0 pseudo-norm, amongst which the ℓ_1 norm holds a place of choice. Indeed, as Figure 1.1 illustrates in dimension 2, the ℓ_1 norm is the largest convex minorant of the ℓ_0 pseudo-norm on the unit ball of the ℓ_∞ norm. The seminal sparse convex estimator,

the Lasso (Tibshirani, 1996) (independently proposed by Chen and Donoho (1995) as Basis Pursuit Denoising), solves, for $\tau \geq 0$:

$$\arg\min_{\beta \in \mathbb{R}^p} \frac{1}{2}\|y - X\beta\|^2 \quad s.t. \quad \|\beta\|_1 \leq \tau \ . \tag{1.28}$$

A more employed equivalent form of Problem (1.28) is, for a regularization parameter $\lambda \geq 0$:

$$\arg\min_{\beta \in \mathbb{R}^p} \frac{1}{2}\|y - X\beta\|^2 + \lambda\|\beta\|_1 \ . \tag{1.29}$$

As for Tikhonov regularization, Problems (1.28) and (1.29) are equivalent[3] in the sense that for any value of λ, there exist a value of τ such that the two solutions coincide, and vice versa. This (data dependent) mapping is again not explicit in general, and we will therefore focus on the form (1.29), which is easier to solve in practice.

The respective impacts of ℓ_1 and (squared) ℓ_2 are well-illustrated on the so-called *orthogonal design* case, *i.e.,* when $X^\top X = \mathrm{Id}_p$. In that case, the OLS solution is $X^\top y$, the Lasso solution is $\mathrm{ST}(X^\top y, \lambda) \triangleq (\mathrm{sign}(X_{:j}^\top y) \cdot (|X_{:j}^\top y| - \lambda)_+)_{j \in [p]}$, and the Tikhonov solution is $\frac{1}{1+\lambda} X^\top y$. It is clear that Tikhonov regularization produces a downscaled version of the OLS estimate, but does not set coefficients to 0, while the Lasso sets OLS coefficients below λ in absolute value to 0, and shrinks others by λ.

The Lasso gave birth to numerous approaches, such as Elastic Net (Zou and Hastie, 2005), sparse logistic regression (Koh et al., 2007), group Lasso (Yuan and Lin, 2006), sparse-group Lasso (Simon et al., 2013), graphical Lasso (Friedman et al., 2008), multitask Lasso (Obozinski et al., 2010), square-root Lasso (Belloni et al., 2011) or nuclear norm penalization for matrices (Fazel, 2002; Argyriou et al., 2006). These *Lasso-type* problems all have a convex formulation, and can be solved via a multitude of well-studied optimization algorithms: primal-dual (Chambolle and Pock, 2011), forward-backward (Beck and Teboulle, 2010; Combettes and Pesquet, 2011), Alternating Direction Method of Multipliers (Boyd et al., 2011), accelerated proximal gradient descent (Nesterov, 1983; Beck and Teboulle, 2009), or proximal block coordinate descent (Wu and Lange, 2008; Tseng and Yun, 2009).

The convex approach has two benefits: it leads to fast algorithms with global convergence guarantees, and allows for an analysis of estimation consistency, prediction performance (Bickel et al., 2009; Negahban et al., 2010) and model consistency (Zhao and Yu, 2006). In Compressed Sensing, under some conditions, the ℓ_1 relaxation allows to recover perfectly the ℓ_0 solution (Candès et al., 2006; Donoho, 2006).

On the other hand, a notorious drawback is that the resulting estimates are biased in amplitude (Fan and Li, 2001), a bias which is easy to see on an orthogonal design. Alternative substitutes to the ℓ_0 penalty were proposed, for instance Smoothly Clipped Absolute Deviation (SCAD, Fan and Li 2001), Minimax Concave Penalty (MCP, Zhang 2010), ℓ_p pseudo-norms with $0 < p < 1$ (Frank and Friedman 1993, Chartrand 2007 in Compressed Sensing), log penalty (Candès et al., 2008) or CEL0 (Soubies et al., 2015). This type of penalties are usually called folded concave penalties, because coordinate-wise they are concave on $\mathbb{R}_+$ and symmetric *w.r.t.* origin. The interested reader may refer to Huang et al. (2012) for a review of convex and non-convex approaches for feature selection. An appealing property of SCAD and MCP is that, although not convex, their proximal operator can be computed in closed-form. Solving other non-convex penalties

[3] note that equivalence does not hold for the ℓ_0 penalty (Nikolova, 2016)

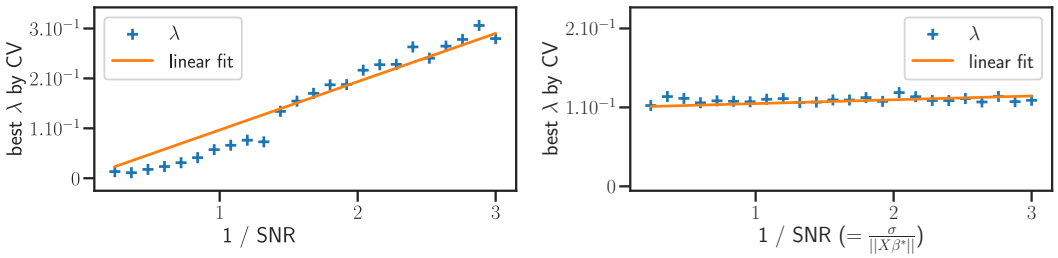


Figure 1.2 – Optimal value of the Lasso (left) and Concomitant Lasso (right) regularization parameters λ determined by cross validation on prediction error (blue), for a logarithmic grid of 100 values of λ between $\lambda_{\max}$ and $\lambda_{\max}/100$, as a function of the noise level on simulated values of y. As indicated by theory, the Lasso's optimal λ grows linearly with the noise level, while it remains constant for the Concomitant Lasso.

(log, square root) can be done by iterative reweighted ℓ_1 approaches (Zou, 2006; Candès et al., 2008; Gasso et al., 2009; Ochs et al., 2015), hence it remains of high interest, even in the non-convex setting, to have fast solvers for ℓ_1-type regularized problems. Finally, although they perform well theoretically and in practice, the non-convexity of these approaches often makes it difficult or impossible to find the exact solution in practice: algorithms are sensitive to initialization, multiple local minima exist, and a global convergence criterion is lacking.

In addition to interpretability, sparsity comes with statistical benefits. In the OLS example, let us assume that β^* is sparse, that its support $\mathcal{S}^*$ of size s is known, and that a sparse model is obtained by setting entries of $\hat{\beta}^{\text{OLS}}$ outside $\mathcal{S}^*$ to 0. Then, Equation (1.22) can be greatly improved:

$$\mathbb{E}[\tfrac{1}{n}\|X_{:\mathcal{S}^*}\hat{\beta}^{\text{OLS}}_{\mathcal{S}^*} - X_{:\mathcal{S}^*}\beta^*_{\mathcal{S}^*}\|^2] = \sigma^2\frac{s}{n} \ , \tag{1.30}$$

which goes to 0 if $s = o(n)$, without constraint on p. Of course, in practice $\mathcal{S}^*$ is not known, but Lounici (2009) showed that under sufficient conditions and for $\lambda = A\sigma\sqrt{(\log p)/n}$ with $A > 2\sqrt{2}$, the Lasso estimator satisfies:

$$\mathbb{E}[\tfrac{1}{n}\|X\hat{\beta} - X\beta^*\|^2] = \sigma^2\frac{s}{n}\log s \ , \tag{1.31}$$

i.e., that it only suffers a factor $\log s$ from the non-knowledge of $\mathcal{S}$, which is a very appealing statistical guarantee.

Yet this approach requires σ to be known and the noise to be homoscedastic, a situation seldom happening in practice. Since this bound is valid for $\lambda = A\sigma\sqrt{(\log p)/n}$, it also suggests that the optimal λ depends linearly on the unknown noise level. This is visible on Figure 1.2, where for a fixed design X (10 000 first columns of the *climate* dataset), we simulate $y = X\beta^* + \sigma\varepsilon$ for various σ, and for each σ we compute the optimal λ by cross-validation. The dependency indeed appears to be linear in practice. It is worth mentioning that Meinshausen and Bühlmann (2006) showed that prediction and variable selection conflict for the Lasso: the statistically optimal λ for prediction gives inconsistent variable selection results (see also Leng et al. (2006) in the orthogonal design).

The Concomitant Lasso, the square-root or the Scaled Lasso estimators (Owen, 2007; Belloni et al., 2011; Sun and Zhang, 2012) achieve the same bound as Equation (1.31), with a regularization parameter independent of σ: in this thesis, we aim at generalizing

this approach to correlated Gaussian noise in the multitask framework.

The statistical and practical benefits of sparsity have led to it being used in many applications: audio processing (Zibulevsky and Pearlmutter, 2001), astrophysics, sparse coding (Olshausen and Field, 1997), medical imaging through compressed sensing (Donoho, 2006; Candès et al., 2006), genomics (Bleakley and Vert, 2011), time series analysis (Nardi and Rinaldo, 2011), etc.

We now introduce optimization tools involved in the study of Lasso-type problems.

1.1.3 Convex optimization tools

Throughout this manuscript, we will make extensive use of a convenient class of functions, based on the framework of Bauschke and Combettes (2011).

Definition 1.4 (Proper, lower semicontinuous convex functions)**.** *We denote by* $\Gamma_0(\mathbb{R}^d)$ *the set of functions* $f : \mathbb{R}^d \to]-\infty, +\infty]$ *which are:*

- *proper:* $\operatorname{dom} f \triangleq \{x \in \mathbb{R}^d \,:\, f(x) < +\infty\} \neq \varnothing$,
- *lower semicontinuous:* $\forall x \in \mathbb{R}^d, \lim_{y \to x} f(y) \geq f(x)$,
- *convex:* $\forall x, y \in \mathbb{R}^d, \forall \alpha \in [0,1], f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y)$.

Since non proper functions are of limited interest, in the sequel functions are assumed to be proper even if not explicitly stated.

Strong convexity and smoothness are two function properties, used to derive convergence guarantees and rates for algorithms minimizing functions:

Definition 1.5 (Strong convexity and smoothness)**.** *Let* f *be a differentiable function*[4] *from* $\mathbb{R}^d$ *to* $]-\infty, +\infty]$*. For* $\mu, M > 0$*, we say that* f *is* μ-strongly convex *if:*

$$\forall x, y \in \mathbb{R}^d, f(x) \geq f(y) + \nabla f(y)^\top (x - y) + \frac{\mu}{2} \|x - y\|^2 \ , \tag{1.32}$$

and that f *is* M-smooth *if:*

$$\forall x, y \in \mathbb{R}^d, f(x) \leq f(y) + \nabla f(y)^\top (x - y) + \frac{M}{2} \|x - y\|^2 \ . \tag{1.33}$$

If f *is both* μ*-strongly convex and* M*-smooth, we have* $\mu \leq M$ *and the* condition number $\frac{M}{\mu} \geq 1$ *is a useful quantity, appearing in the convergence rate of many algorithms.*

Convex indicators, infimal convolution, Fenchel-Legendre transform and proximal operators are the workhorses of continuous convex optimization.

Definition 1.6 (Indicator function)**.** *Let* $\mathcal{C}$ *be a subset of* $\mathbb{R}^d$*. The* indicator function *of* $\mathcal{C}$ *is:*

$$\begin{aligned} \iota_{\mathcal{C}} : \mathbb{R}^d &\to]-\infty, +\infty] \\ x &\mapsto \begin{cases} 0 \ , & \text{if } x \in \mathcal{C} \ , \\ +\infty \ , & \text{otherwise} \ . \end{cases} \end{aligned} \tag{1.34}$$

[4] a more general definition exists for strong convexity, not requiring differentiability

Table 1.1 – Useful Fenchel transforms

Function	Fenchel transform	
h^*	$h, \quad \forall h \in \Gamma_0(\mathbb{R}^d)$	(1.37)
$g \,\square\, h$	$g^* + h^*$	(1.38)
ah	$ah^*\left(\frac{\cdot}{a}\right), \quad \forall a > 0$	(1.39)
$\|\cdot\|_p$	$\iota_{\mathcal{B}_{p^*}}$, where $\frac{1}{p} + \frac{1}{p^*} = 1$	(1.40)
$(h + \delta)$	$h^* - \delta, \quad \forall \delta \in \mathbb{R}$	(1.41)
$\frac{1}{2}\|\cdot\|^2$	$\frac{1}{2}\|\cdot\|^2$	(1.42)

We have that $\iota_{\mathcal{C}} \in \Gamma_0(\mathbb{R}^d)$ if and only if $\mathcal{C}$ is non empty, closed and convex (Bauschke and Combettes, 2011, Examples 1.25 and 8.3).

Definition 1.7 (Infimal convolution)**.** *Let f and g be two functions from $\mathbb{R}^d$ to $]-\infty, +\infty]$. The* infimal convolution *of f and g is:*

$$\begin{aligned} f \,\square\, g : \mathbb{R}^d &\to [-\infty, +\infty] \\ x &\mapsto \inf_{u \in \mathbb{R}^d} f(x - u) + g(u) \ . \end{aligned} \tag{1.35}$$

Definition 1.8 (Fenchel-Legendre transform)**.** *Let $f : \mathbb{R}^d \to [-\infty, +\infty]$. Its* Fenchel-Legendre transform *or* conjugate*, f^* is defined as:*

$$\begin{aligned} f^* : \mathbb{R}^d &\to]-\infty, +\infty] \\ u &\mapsto \sup_{x \in \mathbb{R}^d} u^\top x - f(x) \ . \end{aligned} \tag{1.36}$$

Note that f needs not be convex, but f^* always is. Frequently used Fenchel-Legendre transforms are reminded in Table 1.1 (see Bauschke and Combettes 2011, Propositions 13.16, 13.20 and 13.21 and Example 13.24 (iv) for proofs).

Proposition 1.9 (Smoothness and strong convexity linked by Fenchel transform, Hiriart-Urruty and Lemaréchal 1993, Thm 4.2.1)**.** *Let $f \in \Gamma_0(\mathbb{R}^d)$. Then, for $\gamma > 0$, f is γ-smooth if and only if f^* is $1/\gamma$-strongly convex.*

Proposition 1.9 provides a way to transform a function into a smooth one, that we will use in Chapter 4. Given a non-smooth function $f \in \Gamma_0(\mathbb{R}^d)$, we can add a strongly convex function ω to f^*, thus making it strongly convex, then take the Fenchel transform again to obtain a smooth function. Formally, the smooth approximation of f is $(f^* + \omega)^*$, which is also equal to $f \,\square\, \omega^*$ by Equation (1.38). As illustrated on Figure 1.3, this technique is a possible construction for the famous Huber function, a smooth approximation to the absolute value function.

Definition 1.10 (Proximal operator)**.** *Let $f \in \Gamma_0(\mathbb{R}^d)$. The* proximal operator *of f, introduced in the seminal work of Moreau (1965), is:*

$$\begin{aligned} \operatorname{prox}_f : \mathbb{R}^d &\to \mathbb{R}^d \\ x &\mapsto \arg\min_{y \in \mathbb{R}^d} \frac{1}{2}\|x - y\|^2 + f(y) \ . \end{aligned} \tag{1.43}$$

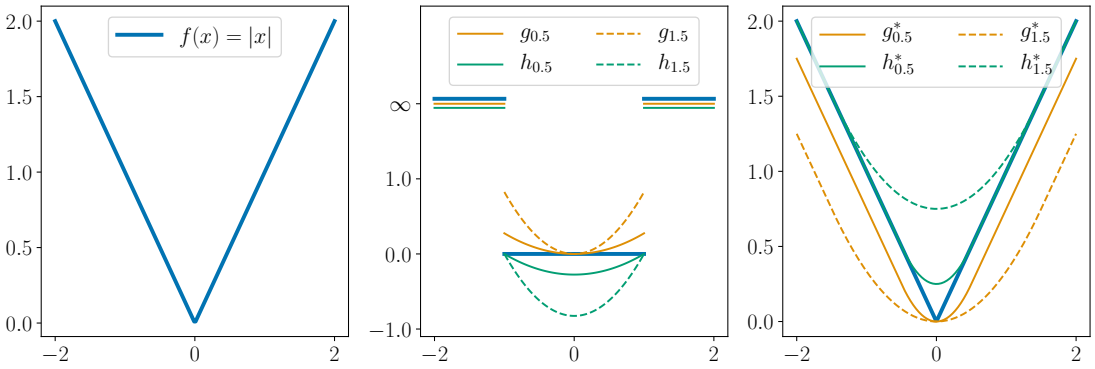


Figure 1.3 – Various ways to smooth the absolute value function f, by adding a strongly convex term to $f^* = \iota_{[-1,1]}$. Taking the Fenchel transform of the strongly convex functions $g_\rho : u \mapsto f^*(u) + \rho u^2/2$ and $h_\rho : u \mapsto f^*(u) + \rho(u^2/2 - 1/2)$ yields smooth approximations of f. As ρ increases, the approximations get smoother, but further away from f.

The two following proximal operators are extensively used in our work.

Proposition 1.11 (Proximal operators of ℓ_1 and Euclidean norm, Bach et al. 2012, Section 3.3, p. 45)**.** *Let $x \in \mathbb{R}^d$, $A \in \mathbb{R}^{d \times d'}$ and $\tau > 0$. The proximal operators of $\tau\|\cdot\|_1$ and $\tau\|\cdot\|$ are respectively the* soft-thresholding *and* block soft-thresholding *operators:*

$$\operatorname{prox}_{\tau\|\cdot\|_1}(x) = \operatorname{ST}(x, \tau) \triangleq \Big(\operatorname{sign}(x_j)(|x_j| - \tau)_+\Big)_{j \in [d]} \quad , \tag{1.44}$$

$$\operatorname{prox}_{\tau\|\cdot\|}(A) = \operatorname{BST}(x, \tau) \triangleq (1 - \tau/\|A\|)_+ \cdot A \quad . \tag{1.45}$$

Definition 1.12 (Subdifferential)**.** *Let $f : \mathbb{R}^d \to]-\infty, +\infty]$. The* subdifferential *of f at $x \in \operatorname{dom}(f)$ is:*

$$\partial f(x) \triangleq \{u \in \mathbb{R}^d \ : \ \forall y \in \mathbb{R}^d, f(y) \geq f(x) + u^\top (y - x)\} \quad , \tag{1.46}$$

i.e., *the set of slopes of all affine minorants of f which are exact at x.*

Elements of the subdifferential are called *subgradients*, and for a convex function f, its subdifferential is non empty at every point of the relative interior of $\operatorname{dom} f$. In some sense, subgradients are a generalization of gradients: for a convex differentiable function, the subdifferential at $x \in \operatorname{dom} f$ has only one element: $\nabla f(x)$. Subdifferentiability allows to generalize first order optimality conditions to non differentiable convex functions.

Proposition 1.13 (Fermat's rule)**.** *Let f be a proper convex function. Then, for all $\hat{x} \in \mathbb{R}^d$:*

$$\hat{x} \in \operatorname*{arg\,min}_{x \in \mathbb{R}^d} f(x) \Leftrightarrow \mathbf{0}_d \in \partial f(\hat{x}) \quad . \tag{1.47}$$

The notion of *strong duality* is extensively used in both Part I and Part II.

Proposition 1.14 (Fenchel duality, Rockafellar 1997, Thm. 31.3)**.** *Let $f \in \Gamma_0(\mathbb{R}^n)$ and $g \in \Gamma_0(\mathbb{R}^p)$. Let $X \in \mathbb{R}^{n \times p}$ and $\lambda > 0$. The following problems are called respectively* primal *and* dual *problems:*

$$\hat{\beta} \in \operatorname*{arg\,min}_{\beta \in \mathbb{R}^p} \underbrace{f(X\beta) + \lambda g(\beta)}_{\mathcal{P}(\beta)} \quad , \tag{1.48}$$

$$\hat{\theta} \in \arg\max_{\theta \in \mathbb{R}^n} \underbrace{-f^*(-\lambda\theta) - \lambda g^*(X^\top \theta)}_{\mathcal{D}(\theta)} \ . \tag{1.49}$$

Given $\beta \in \mathbb{R}^p$ *and* $\theta \in \mathbb{R}^n$*, the* duality gap *is* $\mathcal{P}(\beta) - \mathcal{D}(\theta) \geq 0$.

Strong duality, i.e., $\mathcal{P}(\hat{\beta}) = \mathcal{D}(\hat{\theta})$ *holds if and only if:*

$$-\lambda\hat{\theta} \in \partial f(X\hat{\beta}) \ , \tag{1.50}$$

$$X^\top \hat{\theta} \in \partial g(\hat{\beta}) \ , \tag{1.51}$$

or equivalently:

$$-X\hat{\beta} \in \partial f^*(-\lambda\hat{\theta}) \ , \tag{1.52}$$

$$\hat{\beta} \in \partial g^*(X^\top \hat{\theta}) \ . \tag{1.53}$$

These conditions are called Kuhn-Tucker conditions. It is worth mentioning that a sufficient condition for strong duality to hold is that the relative interiors of the domains of f *and* g *intersect, which happens for example if neither* f *nor* g *take the value* $+\infty$.

Definition 1.15 (Block indexing)**.** *Let* $\beta \in \mathbb{R}^p$, $\mathrm{B} \in \mathbb{R}^{p \times q}$ *and* $X \in \mathbb{R}^{n \times p}$*. Let* $\mathcal{I}$ *denote a partition of* $[p]$ *and let* $I \in \mathcal{I}$.

The vector $\beta_I \in \mathbb{R}^{|I|}$ *is obtained by keeping only entries of* β *whose indices are in* I*. In the multitask setting,* $B_{I:} \in \mathbb{R}^{|I| \times q}$ (resp. $X_{:I} \in \mathbb{R}^{n \times |I|}$) *is the matrix obtained by keeping only rows of* B (resp. *columns of* X) *whose indices are in* I.

For a L*-smooth function* $f \in \Gamma_0(\mathbb{R}^d)$, $\nabla_I f \in \mathbb{R}^{|I|}$ *is the gradient of* f *when only coordinates in* I *vary, and* L_I *is the Lipschitz constant of this gradient (which exists because* f *is itself* L*-smooth).*

Proposition 1.16 (Proximal operator of separable function, Parikh et al. 2013, Sec. 2.1)**.** *Let* $\mathcal{I}$ *be a partition of* $[d]$ *and* $g \in \Gamma_0(\mathbb{R}^d)$ *be a function admitting a block decomposable structure:* $g(x) = \sum_{I \in \mathcal{I}} g_I(x_I)$*. Then,* $\mathrm{prox}_g(x)$ *is equal to the vector obtained by concatenation of the vectors* $\mathrm{prox}_{g_I}(x_I) \in \mathbb{R}^{|I|}$*, for* $I \in \mathcal{I}$.

Definition 1.17 ("Smooth + proximable" composite problem)**.** *Let* $f \in \Gamma_0(\mathbb{R}^d)$ *be* L*-smooth, and* $g \in \Gamma_0(\mathbb{R}^d)$ *be such that* prox_g *can be computed exactly. We call the optimization problem:*

$$\min f(x) + g(x) \ , \tag{1.54}$$

a "smooth + proximable" composite problem.

In general, non-smooth convex optimization is harder than smooth convex optimization, in the sense that the worst case convergence rate for first order (subgradient) methods is $\mathcal{O}(k^{-1/2})$ (Goffin, 1977), in opposition to $\mathcal{O}(k^{-1})$ and $\mathcal{O}(k^{-2})$ for (eventually accelerated) first order methods in the smooth case (Nesterov, 1983). But for problems presenting a smooth + proximable structure, which are legion in Machine Learning, one needs not worry: they can be solved with proximal gradient methods, with same optimal rates as gradient methods – up to linear when f is strongly convex (Beck and Teboulle, 2009). In this thesis, we will have a particular interest in solving instances of Problem (1.54), using two algorithms: proximal gradient descent and proximal block coordinate descent (Combettes and Pesquet, 2011). We recall them in Algorithms 1.1 and 1.2. Although both algorithms have worst case convergence rates of $\mathcal{O}(1/k)$, Figure 1.4 illustrates that practical results can be very different. Amongst the reasons

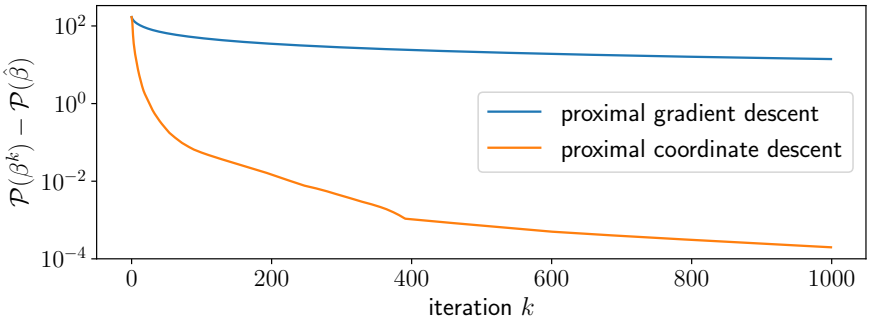


Figure 1.4 – Convergence speed of proximal gradient descent and proximal coordinate descent on a Lasso problem (subsampled *climate* dataset ($n = 857, p = 10\,000$), $\lambda = \lambda_{\max}/20$, resulting in $\|\hat{\beta}\|_0 = 220$). Although both algorithm have the same convergence rates, proximal coordinate descent outperforms gradient descent by several order of magnitude.

Algorithm 1.1 PROXIMAL GRADIENT DESCENT FOR PROBLEM (1.54)

input : L, T
init : $\beta^{(0)}$
for $t = 1, \dots, T$ **do**
  $\beta^{(t)} = \operatorname{prox}_{\frac{\lambda}{L} g}\left(\beta^{(t-1)} - \frac{1}{L}\nabla f(\beta^{(t-1)})\right)$
return $\beta^{(T)}$

Algorithm 1.2 CYCLIC PROXIMAL BLOCK COORDINATE DESCENT FOR PROBLEM (1.54)

input : $\{L_I\}_{I \in \mathcal{I}}, T$
init : $\beta^{(0)}$
for $t = 1, \dots, T$ **do**
  $\beta^{(t)} = \beta^{(t-1)}$
  for $I \in \mathcal{I}$ **do**
    $\beta_I^{(t)} = \operatorname{prox}_{\frac{\lambda}{L_I} g_I}\left(\beta_I^{(t)} - \frac{1}{L_I}\nabla_I f(\beta^{(t)})\right)$
return $\beta^{(T)}$

explaining this disparity, the algorithms may take different times to identify the support (and they enjoy a linear convergence rate after support identification), or different constant values in the $\mathcal{O}$. This motivates our preference for coordinate descent in the whole manuscript.

Equipped with this mathematical background, we now move to our application focus: the bio-magnetic inverse problem.

1.2 The bio-magnetic inverse problem

1.2.1 Basis of M/EEG

Brain imaging modalities can be divided into two categories: indirect and direct approaches. Indirect approaches, such as near infrared spectroscopy or positron emission tomography, detect brain activity by measuring a correlated physical quantity,

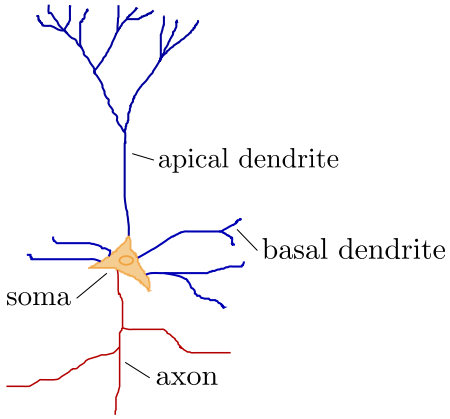


Figure 1.5 – Schematic view of a pyramidal neuron by the author. The name *pyramidal* comes from the shape of the soma.

e.g., metabolic activity. The best-known indirect brain imaging modality is functional magnetic resonance imaging (fMRI), which measures the hemodynamic response, *i.e.,* the delivery of blood to active neuronal tissues. The main feature of fMRI analysis is its excellent spatial resolution, ranging from 4 mm to 0.5 mm for the most recent MRI scanners (Duyn, 2012; Huber et al., 2017). However, because the hemodynamic response is slow and lagging in time, the time resolution of fMRI is only around 1 s, making it impractical for the study of dynamic brain processes.

On the contrary, electroencephalography (EEG), magnetoencephalography (MEG), intracranial electroencephalography (iEEG, also known as electrocortigraphy) and stereotaxic electroencephalography (sEEG) are direct approaches, which record electrical potentials or magnetic fields generated by the activity of the brain. Because they directly measure the quantity of interest, their temporal resolution is excellent: around 1 ms. For this reason, they are widely used to localize foci of epilepsy, or to map brain areas to be excluded from surgical removal, *e.g.,* associated to speech or movement. Very accurate techniques, iEEG and sEEG are also highly invasive: iEEG requires craniotomy in order to insert a grid of electrodes inside the brain, and small openings must be drilled in the skull to insert sEEG electrodes in the brain. In this manuscript, we focus on magneto- and electroencephalography, which in contrast stand out by their low invasiveness.

What do M/EEG measure? A neuron consists of a cell (the soma), and dendrites; neurons can be connected to other neurons via axons. As visible on Figure 1.5, a pyramidal neuron possesses an *apical dendrite*. Each neuron maintains a varying electrical potential at its soma's membrane, due to ionic concentration differences within the cell. This potential can trigger an action potential, traveling in the axon to connected neighbors, with excitatory or inhibitory effect on the receiving cell.

When a neuron receives such a pulse, Excitatory Post-Synaptic Potentials (EPSPs) are generated at its apical dendritic tree (Gloor, 1985). The resulting potential difference between the apical dendritic tree on one side, and the membrane of the soma and basal dendrites on the other causes primary electrical currents to travel intracellularly in the dendrite, from the former to the latter. As a consequence, secondary (or volume) currents travel extracellularly in the head tissue, closing the current loop. These currents can be modeled by a current dipole, oriented along the dendrite. The typical moment of such a dipole is very small: 20 fA.m, and neural activity is only made measurable

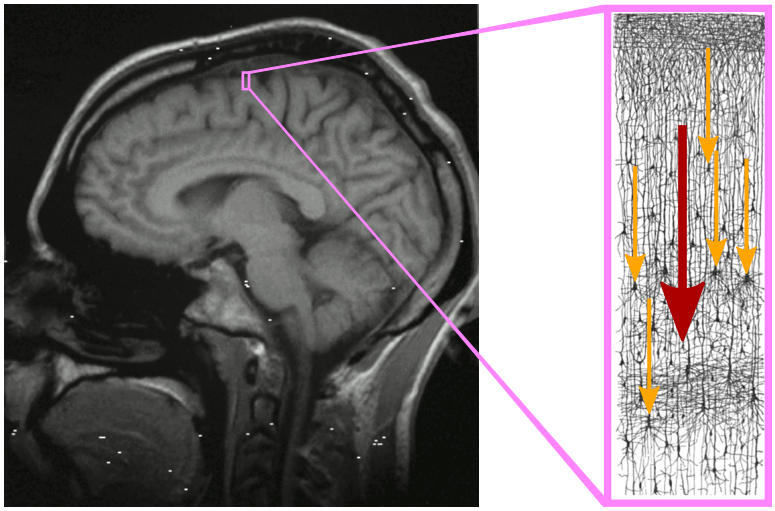


Figure 1.6 – Left: sagittal MRI view of the brain. Right: pyramidal cells as drawn by Ramon y Cajal (1899), with dipoles modeling currents from apical dendrites to somas (orange). The parallel alignment of the dipoles results in constructive interference, modeled as an Equivalent Current Dipole of larger moment (red).

because of two phenomena. First, as shown on Figure 1.6, the columnar functional organization of the cortex causes large groups of pyramidal neurons to have parallel alignment of apical dendrites, and thus EPSPs associated currents traveling in the same direction. Second, the long duration of the post-synaptic potentials makes them likely to overlap over a synchronized group of neurons. The axonal action potentials, on the other hand, are not detected by M/EEG: the current flows are in opposite directions and are too brief to interfere constructively (Nunez and Srinivasan, 2006). The spatio-temporal superposition allow primary and secondary currents to interact constructively, adding up to 50 nA.m, a threshold high enough to be measured extracranially by M/EEG (Murakami and Okada (2015) estimate that the critical population comprises at least 10 000 neurons, corresponding to a cortical patch of 25 mm^2). The currents in such a neuron population can be modeled at the macroscopic scale by an Equivalent Current Dipole (ECD), which is the sum of all the current dipoles in the patch of synchronized neurons.

EEG measures the difference of potentials between electrodes and a reference: around 60 electrodes are used, positioned at standard locations on the scalp to allow for reproducibility of recordings.

The acquired potentials are of the order of 10 μV. MEG is a somehow more refined technique than EEG: the measured magnetic fields are of the order of 10 fT, seven or eight orders of magnitude smaller than the Earth's magnetic field. Recording such small values is only made possible by the use of magnetic shielding and superconductivity-exploiting magnetometers (superconductivity quantum interference device, SQUIDs). Along with magnetometers, gradiometers measuring the spatial gradient of the magnetic field are used to reduce the sensibility to interferences. There usually are around 200 magnetometers and 100 gradiometers, isolated in a liquid helium cooled vacuum flask, which makes MEG sensors further away from the neural sources than EEG sensors.

Figure 1.7 – Patient undergoing a cognitive experiment in a MEG scanner. Courtesy of National Institute of Mental Health.

These technical differences are reflected in historical landmarks: while the first EEG was recorded in 1924 by Hans Berger, the first MEG recording was performed by David Cohen in 1968 (Cohen, 1968) and it is only in the nineties that the first full head MEG devices were used for the first time. To this day, MEG is still more expensive to operate than EEG, because of magnetic shielding and the liquid helium needed for superconductivity in the sensors. The two techniques are complementary: EEG is sensitive to radial and tangential dipoles, whereas MEG is insensitive to radial sources, but has a higher signal-to-noise ratio, and can use more sensors. Instead of using only MEG or EEG, pooling electrodes, magnetometers and gradiometers allows to locate more accurately the origin of brain activity, for example in the case of epilepsy (Aydin et al., 2015).

In our experimental setup, the patient undergoes repetitions of the same simple stimulation (sensory, cognitive or motor) for a short period of time. Neuronal activity can then be divided in two categories: spontaneous and event-related activity. Event-related activity is triggered by the stimuli, and is either *evoked* if the response is phase-locked with respect to the stimuli, or *induced* otherwise. Despite the sophisticated sensors and shielding, M/EEG suffers from a poor signal-to-noise ratio (SNR); among corrupting factors are eye movements, heartbeats and other muscle activity, movement, sensor drift and ambient electromagnetic noise (Gross et al., 2013). Various signal preprocessing techniques are used to increase the SNR (Parkonnen, 2010; Gross et al., 2013): spectral filtering, signal decomposition via Independent Component Analysis (Makeig et al., 1996; Ablin et al., 2018) or Signal Space Separation. Another mandatory step to increase the SNR is to average several repetitions (called *trials*) of the experiment with the same patient. As shown in Figure 1.8, as more and more trials are averaged, the signal becomes smoother and the brain response to the stimuli at $t = 0.1$ s, appears once the SNR is high enough. The averaging procedure preserves phase locked responses, but removes induced response, hence the need for more refined solvers taking into account all the trials and not only their average.

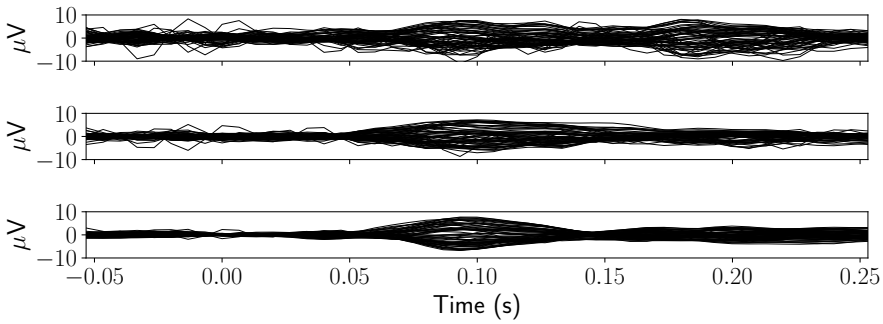


Figure 1.8 – Amplitude of 59 EEG signals, averaged across 5 (top), 10 (middle), and 50 (bottom) trials. As the number of averaged repetitions increases, the noise is reduced and the measurements become smoother, revealing the brain response to the stimuli around 0.1 s after it occurred. The stimuli is an auditory stimulation in the left ear.

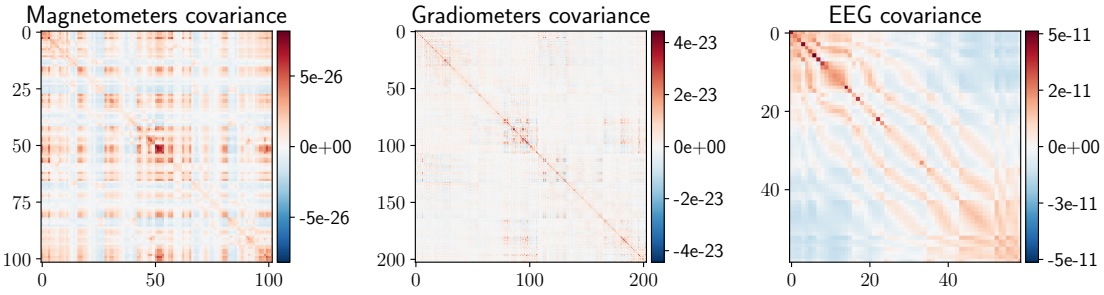


Figure 1.9 – Covariance of the three types of sensors (left: magnetometers, middle: gradiometers, right: electrodes). The covariance matrices are clearly not scalar: EEG covariance has a band diagonal structure, and magnetometers covariance has a block structure.

Apart from averaging data, another critical preprocessing step is spatial noise whitening (Engemann et al., 2015). For the raw measurements, the noise is far from being white, for example because there exist brain noise correlation between neighboring sensors, as shown in Figure 1.9. To decorrelate the noise, a spatial whitening step is applied during the preprocessing, based on an estimate of the noise covariance matrix. This covariance can be estimated in multiple ways: empty room measurements if only MEG is used or empirically using pre-stimulus data, considered as raw noise. When analysis is performed over both EEG and MEG, spatial whitening also allows to harmonize the different units (μV, fT and fT.m^{-1}). Empirical estimation being imperfect, various regularization techniques such as shrinkage (Ledoit and Wolf, 2004) have been proposed. In their extensive review, Engemann and Gramfort (2015b) showed that there is no single best approach, and devised an automatic way to select the best method on a case-by-case basis.

The non-invasiveness of M/EEG comes at a price: the electrical activity is not measured directly at its location in the brain, but outside the scalp, and is thus transformed by the head tissues. Determining the causal factors (brain activity) from a set of observations they produced (the electromagnetic measurements outside the head) is an inverse problem, which can be solved in many ways.

1.2.2 Solving the inverse problem

Historically, three kind of methods have emerged: parametric, scanning and imaging ones. They share the same goal: to determine which areas of the brain are involved in a cognitive task, and how these areas interact together.

Parametric methods *Dipole fitting* (Scherg and von Cramon, 1985) models the brain activity by a *fixed* small number of ECD, whose varying locations and amplitudes are estimated via gradient descent or simulated annealing (Uutela et al., 1998). Sequential dipole fitting estimates the dipoles parameters one by one; for more than one dipole the optimization process – non-linear least squares – is generally non-convex and thus sensitive to initialization. It may also be difficult to correctly estimate the number of dipoles a priori (possible approaches are based on ICA, PCA or SVD as in Kobayashi et al. (2002); Koles and Soong (1998); Huang et al. (1998)), and sequential approaches may fail in the presence of correlated or overlapped source activity.

Scanning methods Beamforming techniques (Van Veen et al., 1997) and signal classification ones (MUSIC, RAP-MUSIC, Mosher et al. 1999; Mosher and Leahy 1999) use a predefined grid of potential locations. They apply spatial filters to evaluate the contribution of each source. As dipole fitting techniques, beamforming fails when sources are correlated (Robinson and Vrba, 1999) and requires a covariance to be estimated from short signals. MUSIC and its derivative are greedy approaches and as such suffer from a high sensitivity to the data.

Imaging methods In distributed source imaging, dipoles are fitted *simultaneously* at a set of locations defined a priori. First, they require to solve the bio-electromagnetic *forward* problem: determining the sensor measurements given a distribution of internal currents. By Maxwell equations, the measurements are a linear function of the dipoles activities. In an ideal noiseless setting, if we postulate a discrete grid of ECD locations in the brain (the *source model*), then the *noiseless* measurements $Y^* \in \mathbb{R}^{n \times q}$ and the true parameter matrix $\mathrm{B}^* \in \mathbb{R}^{p \times q}$ are linked by:

$$Y^* = X\mathrm{B}^* \ . \tag{1.55}$$

Each row of Y^* is the activity of a sensor – a times series of length q – while B^* contains p time signals, each one corresponding to the activity of one neural source. Given the source model and a realistic geometrical model of the patient's head and conductivities of the tissues involved (the *head model*), solving the forward problem (computing X) is achieved with a numerical solver based on finite element or boundary element methods.

Given muscle activity, spontaneous brain and sensor noise, a realistic model is the multitask regression one:

$$Y = Y^* + \mathrm{E} = X\mathrm{B}^* + \mathrm{E} \ . \tag{1.56}$$

The typical orders of magnitude for Model (1.56) are $n \approx 100$ sensors, $q \approx 100$ time instants, $p \approx 10\,000$ neural sources. This makes the problem ill-posed in the sense of Hadamard: it cannot be solved directly without more assumptions. For example, Ordinary Least Squares yield an infinity of solutions; it is still possible to use the one with minimal Frobenius norm, $(X^\top X)^\dagger X^\top Y$, but it is highly sensitive to noise. Using Tikhonov regularization leads to a unique and more stable solution, the Minimum Norm Estimate (Hämäläinen and Ilmoniemi, 1994), which is very fast to compute. Alas, it

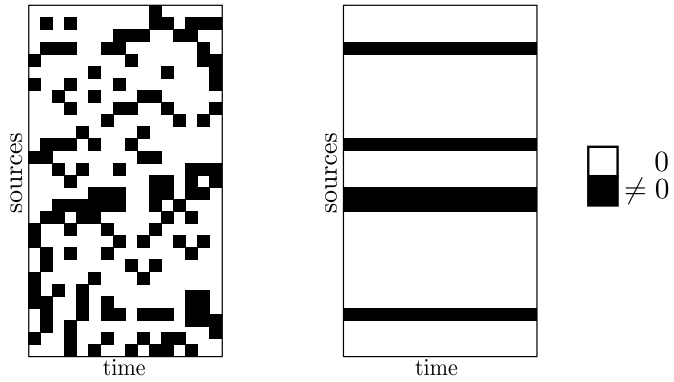


Figure 1.10 – Sparsity patterns obtained by MCE/Lasso (left) and $\ell_{2,1}$/MxNE/Multitask Lasso (right). MCE does not yield a consistent set of active sources over time.

produces dense neural estimates, with activity smeared over all sources, making it unfit to identify clearly localized brain activity. Other notable dense methods are dSPM (Dale et al., 2000) and sLORETA (Pascual-Marqui et al., 2002). All of these are linear: in various ways, each computes a kernel $K \in \mathbb{R}^{p \times q}$, and the estimate is KY.

On the contrary, sparse methods are usually non linear. Sparse Bayesian Learning is a notable sparse approach (Wipf et al., 2008; Haufe et al., 2008), with algorithms such as γ-MAP (Wipf and Nagarajan, 2009) and full MAP (Lucka et al., 2012). They rely heavily on covariance estimation and are therefore not really "parameter free" as one could think.

Straightforwardly applying the Lasso to the multitask framework (*i.e.,* penalizing the sum of the absolute values of the coefficients of B) yields a sparse estimate, known in neuroscience as Selective Minimum Norm or Minimum Current Estimate (Matsuura and Okabe, 1995; Uutela et al., 1999). It is easy to show that this approach amounts to solving q Lasso problems independently; as visible on Figure 1.10, the support of the estimate varies from one time instant to the next, which is not plausible. To produce a consistent set of active sources over time, it has been proposed to use group penalties (Ou et al., 2009) imposing joint sparsity over time, yielding the Mixed Norm Estimate (MxNE) (Gramfort et al., 2012):

$$\hat{\mathrm{B}} \in \underset{\mathrm{B} \in \mathbb{R}^{p \times q}}{\arg\min} \frac{1}{2} \|Y - X\mathrm{B}\|^2 + \lambda \|\mathrm{B}\|_{2,1} \ , \tag{1.57}$$

a problem known in the optimization community as Multitask Lasso (Obozinski et al., 2010), also an instance of group Lasso (Yuan and Lin, 2006). More refined formulations based on MxNE have been proposed, for example using non-convex penalties (ir-MxNE, Strohmeier et al. 2016) or sparse group Lasso formulation in the time frequency domain to produce non stationary activations (TF-MxNE, Gramfort et al. 2013); we refer the reader to Strohmeier (2016) for a very clear presentation of the topic. In this family of estimators, MxNE remains the building block of stable spatio-temporal source reconstruction. The setting of Problem (1.57) assumes a *fixed orientation*: the orientation of each dipole is fixed across time (with a direction usually chosen normal to the cortical mantle). The only quantity to estimate for a dipole is then its magnitude. We may also consider *free orientation*, where the dipoles are allowed to rotate in time: at each time instant, a dipole is represented by its coordinates in a basis of 3 orthogonal

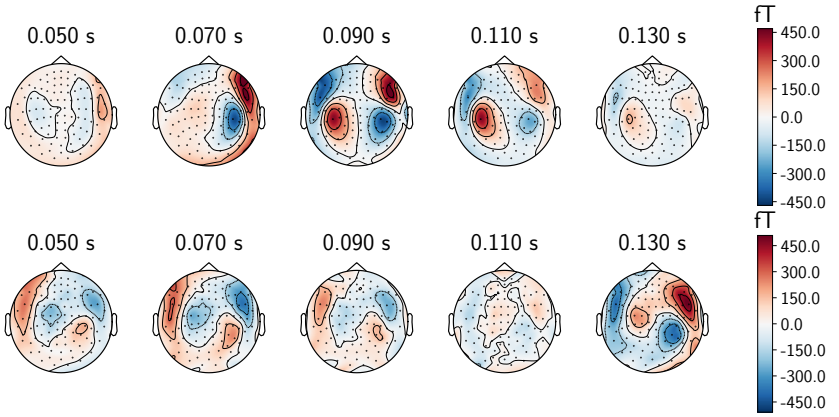


Figure 1.11 – Real (top) and simulated (bottom) magnetometers topographic maps. We simulate the activity of two dipoles in the left and right auditory cortex; the real topographic maps exhibits dipolar patterns similar to the simulated one, justifying the dipolar assumption.

vectors. In this setting, $X \in \mathbb{R}^{n \times 3p}$, and $\mathrm{B} \in \mathbb{R}^{3p \times q}$. Using a mixed $\ell_{2,1}$ penalty would bias the estimates towards the axes of the orthogonal basis used, which is arbitrary. A formulation leading to an orientation-unbiased solution is:

$$\hat{\mathrm{B}} \in \underset{\mathrm{B} \in \mathbb{R}^{3p \times q}}{\arg\min} \frac{1}{2} \|Y - X\mathrm{B}\|^2 + \lambda \sum_{g=1}^{p} \|\mathrm{B}_{\mathcal{G}_g:}\| \ , \tag{1.58}$$

where $\mathcal{G}_g = \{3g, 3g+1, 3g+2\}$, and thus $\mathrm{B}_{\mathcal{G}_g:} \in \mathbb{R}^{3 \times q}$ contains the $3g^{th}$, $3g+1^{th}$ and $3g+2^{th}$ lines of B. This penalty encourages $\mathrm{B}_{\mathcal{G}_g:}$ to be zero, but is isotropic as a change of orthonormal basis does not affect $\|\mathrm{B}_{\mathcal{G}_g:}\|$.

1.3 Contributions

The organization of this manuscript is as follows. Each chapter can be read independently, and thus features a small introduction which can be redundant with this introductory chapter: the reader may feel free to skip them.

Part I is devoted to the design of faster solvers for ℓ_1-type regularized ERM:

- In Chapter 2, we design an efficient solver for the seminal Lasso estimator. We first describe the two main techniques used to speed-up proximal gradient and coordinate descent solvers for sparse GLMs: screening rules and working set policies. Both ignore non-significant variables from the optimization problem, making it smaller, hence faster to solve. In a backward fashion, screening rules prune the set of features, progressively reducing the number of variables. On the contrary, working sets are forward techniques which solve a sequence of growing subproblems, including more and more variables. We show that screening rules can be used aggressively to design working set policies, and introduce *aggressive screening*, a working set policy based on the state-of-the-art Gap Safe screening rules. Screening and working sets both rely on duality. We exhibit the Vector AutoRegressive (VAR) behavior of the Lasso dual iterates, when the primal problem is solved with proximal coordinate descent or proximal gradient descent. We exploit

this structure to construct a dual point with extrapolation, which improves the efficiency of Gap Safe screening rules and working sets. The combination of aggressive screening and dual extrapolation is coined Celer (Constraint Elimination for the Lasso with Extrapolated Residuals).

- In Chapter 3, we generalize the approach of the previous chapter to other datafitting terms and penalties, making it applicable to sparse logistic regression or Multitask Lasso. We adapt the previously introduced dual extrapolation procedure to asymptotic Vector AutoRegressive sequences of dual iterates. We extensively benchmark our approach against state-of-the-art solvers, and highlight the improvements that dual extrapolation brings to Gap Safe screening and working sets. For non quadratic datafitting terms, taking into account the second order information is known to provide a great speedup: we show how to adapt dual extrapolation to the popular prox-Newton solver. We contribute to the reproducibility of our findings by presenting a detailed explanation of the algorithms used. We release Celer as a high-level open source Python package, with a detailed documentation and examples to reproduce the benchmarks presented.

In Part II, we focus on Concomitant noise structure estimation for the bio-magnetic inverse problem.

- In Chapter 4, we introduce new concomitant estimators for the bio-magnetic inverse problem. Along with the optimal regression parameters, concomitant estimators estimate noise variables. The proposed estimators, the Smoothed Generalized Concomitant Lasso (SGCL) and Concomitant Lasso with Repetitions (CLaR) jointly compute the square-root of the noise covariance matrix. They are designed to handle non-white Gaussian noise, with correlation and varying noise levels in the model. The main estimator, CLaR, takes advantage of the multiple repetitions which compose a M/EEG experiment to build a better estimate of the noise covariance. The connections between CLaR's optimization problem and the smoothing of Schatten norms is highlighted: we show how our formulation amounts to solving the previously introduced Multivariate square root Lasso. This result paves the way for an easier use of non-smooth Schatten norms as datafitting terms.
- In Chapter 5, we detail several alternative estimators for multitask regression problems with non-white Gaussian noise. We benchmark the proposed approaches against Concomitant and non Concomitant multi-task estimators. The source recovery performance is evaluated on real M/EEG measurements. The benchmarks are made available in a public, open source implementation.

1.4 Publications

The works presented in this document resulted in the following peer-reviewed publications and preprints (star indicates equal contribution):

- **M. Massias**, A. Gramfort, and J. Salmon. From safe screening rules to working sets for faster lasso-type solvers. In *NIPS-OPT workshop*, 2017

- **M. Massias**, O. Fercoq, A. Gramfort, and J. Salmon. Generalized concomitant multi-task Lasso for sparse multimodal regression. In *AISTATS*, pages 998–1007, 2018a
- **M. Massias**, A. Gramfort, and J. Salmon. Celer: a fast solver for the Lasso with dual extrapolation. In *ICML*, pages 3321–3330, 2018b
- Q. Bertrand*, **M. Massias***, A. Gramfort, and J. Salmon. Handling correlated and repeated measurements with the smoothed Multivariate square-root Lasso. In *NeurIPS*, 2019
- **M. Massias**, S. Vaiter, A. Gramfort, and J. Salmon. Dual extrapolation for sparse Generalized Linear Models. *arXiv preprint arXiv:1907.05830*, 2019

Other articles were published during this PhD, which are not included in the manuscript:

- P. Ablin, T. Moreau, **M. Massias**, and A. Gramfort. Learning stepsizes for unfolded sparse coding. In *NeurIPS*, 2019

Part I

Faster solvers for sparse Generalized Linear Models

2

Faster solvers for the Lasso: screening, working sets and dual extrapolation

"Par dessus l'étang, soudain j'ai vu passer les oies sauvages"

Contents

2.1 Introduction . . . 44
2.2 Duality for the Lasso . . . 45
2.2.1 Stopping iterative solvers . . . 45
2.2.2 Dual extrapolation . . . 47
2.2.3 Dual perspective on coordinate descent . . . 53
2.3 Gap Safe screening . . . 55
2.4 Working sets with aggressive gap screening . . . 55
2.5 Experiments . . . 57
2.5.1 Practical implementation . . . 57
2.5.2 Higher dual objective . . . 58
2.5.3 Better Gap Safe screening performance . . . 59
2.5.4 Working sets application to Lasso path . . . 60
2.6 Conclusion . . . 63

In this chapter, we address the need for faster Lasso solvers to handle high dimensional modern datasets. To accelerate solvers, state-of-the-art approaches consist in reducing the size of the optimization problem. In a regression context, this is achieved either by discarding irrelevant features (screening techniques) or by prioritizing features likely to be included in the support of the solution (working set techniques). The performances of both of these approaches critically depends on the construction of a dual point, as close as possible to optimum. To construct a dual point tighter than the one classically used, we use an extrapolation procedure, which exploits the Vector AutoRegressive structure of the sequence of residuals. We also use Gap Safe rules in an aggressive fashion, to design a working set policy. The resulting method is coined Celer: Constraint Elimination for the Lasso with Extrapolated Residuals. Thanks to our new dual point construction, we show significant computational speedups on multiple real-world Lasso problems.

This chapter covers the following publications:

- **M. Massias**, A. Gramfort, and J. Salmon. From safe screening rules to working sets for faster lasso-type solvers. In *NIPS-OPT workshop*, 2017
- **M. Massias**, A. Gramfort, and J. Salmon. Celer: a fast solver for the Lasso with dual extrapolation. In *ICML*, pages 3321–3330, 2018b

2.1 Introduction

Following the seminal work on the Lasso (Tibshirani, 1996) and Basis Pursuit (Chen and Donoho, 1995), convex sparsity-inducing regularizations have had a major impact on machine learning (see Bach et al. (2012) for a review of practical applications). Now thoroughly analyzed in terms of statistical efficiency (Bickel et al., 2009), the Lasso yields a sparse solution, hence both a more interpretable model and reduced time for prediction. In machine learning applications, the default algorithm to solve the Lasso is (proximal) coordinate descent (Fu, 1998; Tseng, 2001; Friedman et al., 2010).

Since by design only a fraction of features are included in the optimal solution (what we refer to as the solution's *support*), state-of-the-art solver speed-ups rely on limiting the size of the problems to consider. To do so, various approaches can be distinguished: *screening* techniques (Wang et al., 2013; Ogawa et al., 2013; Fercoq et al., 2015), following the seminal work of El Ghaoui et al. (2012), strong rules (Tibshirani et al., 2012), or correlation screening (Xiang and Ramadge, 2012). Similar techniques have also been considered to discard samples in stochastic gradient descent or Support Vector Machines applications (Vainsencher et al., 2015; Shibagaki et al., 2016; Hong et al., 2019). When a screening rule guarantees that all discarded features cannot be in the solution, it is called *safe*. The current state-of-the-art safe screening rules are the so-called Gap Safe rules (Ndiaye et al., 2017b), relying on duality gap evaluation and the knowledge of a suitable dual point.

Alternatively, *working sets* (WS) techniques (Fan et al., 2008; Boisbunon et al., 2014; Johnson and Guestrin, 2015) select a subset of important features according to a particular criterion, and approximately solve the subproblem restricted to these features. A new subset is then defined, and the procedure is repeated. While screening techniques start from full problems and prune the feature set, working set techniques rather start with small problems and include more and more features if needed. Working sets are also called *active sets* in the literature, for example in the domain of Linear Programming where they originated (Thompson et al., 1966; Palacios-Gomez et al., 1982; Myers and Shih, 1988); we choose *working* because in the screening literature, "active set" refers to the non discarded features. For these techniques, duality can also come into play, both in the stopping criterion of the subproblem solver, as well as in the working set definition.

The organization of the chapter is as follows: in Section 2.2, we remind the practical importance of duality for Lasso solvers and present a technique coined dual extrapolation to obtain better dual points. We also shed some light on the success of our approach when combined with cyclic coordinate descent by interpreting the latter as Dykstra's algorithm in the Lasso dual. In Section 2.3, we show how dual extrapolation is well-suited to improve Gap Safe screening. We present in Section 2.4 a working set strategy based on an aggressive relaxation of the Gap Safe rules. Experiments in Section 2.5 show significant computational speedups on multiple real-world Lasso problems.

2.2 Duality for the Lasso

The Lasso estimator is defined as a solution of:

$$\hat{\beta} \in \arg\min_{\beta \in \mathbb{R}^p} \underbrace{\frac{1}{2} \left\| y - X\beta \right\|^2 + \lambda \left\| \beta \right\|_1}_{\mathcal{P}(\beta)} \ , \tag{2.1}$$

where λ is a positive scalar parameter controlling the trade-off between data-fitting and regularization.

A dual formulation of the Lasso reads (see Proposition 1.14 or Kim et al. (2007) for a precise derivation):

$$\hat{\theta} = \arg\max_{\theta \in \Delta_X} \underbrace{\frac{1}{2} \left\| y \right\|^2 - \frac{\lambda^2}{2} \left\| \theta - \frac{y}{\lambda} \right\|^2}_{\mathcal{D}(\theta)} \ , \tag{2.2}$$

where $\Delta_X \triangleq \{\theta \in \mathbb{R}^n \ : \ \|X^\top \theta\|_\infty \leq 1\}$ is the (rescaled) dual feasible set. The associated duality gap is defined by $\mathcal{G}(\beta, \theta) \triangleq \mathcal{P}(\beta) - \mathcal{D}(\theta)$, for any primal-dual pair $(\beta, \theta) \in \mathbb{R}^p \times \Delta_X$. In particular, as illustrated in Figure 2.2a, the dual problem is equivalent to computing $\Pi_{\Delta_X}(y/\lambda)$, and $\hat{\theta}$ is unique even if the primal has more than one solution.

Remark 2.1. *To fit the framework of Proposition 1.14 rigorously, the dual problem should be unconstrained, and $\mathcal{D}(\theta)$ should include a $-\iota_{\Delta_X}(\theta)$ term, so that non-feasible points have an objective value of $-\infty$. We sacrifice this rigor to the benefit of lighter notation, and only apply $\mathcal{D}$ to feasible dual points.*

Proposition 2.2. *Strong duality holds for Problems (2.1) and (2.2):*

$$\mathcal{G}(\hat{\beta}, \hat{\theta}) = 0 \ , \tag{2.3}$$

and primal and dual solutions are linked by:

$$\hat{\theta} = \frac{1}{\lambda}(y - X\hat{\beta}) \ . \tag{2.4}$$

Proof The primal problem is unconstrained and convex. The results follow from Proposition 1.14. ∎

2.2.1 Stopping iterative solvers

In general, Problem (2.1) does not admit a closed-form solution. Iterative optimization procedures such as (block) coordinate descent (BCD/CD, Tseng 2001; Friedman et al. 2007) (*resp.* FISTA, Beck and Teboulle 2009) are amongst the most popular algorithms when dealing with high dimensional applications in machine learning (*resp.* in image processing). A key practical question for iterative algorithms is the stopping criterion: when should the algorithm be stopped? For any pair $(\beta, \theta) \in \mathbb{R}^p \times \Delta_X$, we have $\mathcal{P}(\beta) - \mathcal{P}(\hat{\beta}) \leq \mathcal{G}(\beta, \theta)$, which means that the duality gap provides an upper bound for the suboptimality gap. Therefore, given a tolerance $\epsilon > 0$, if at iteration t of the algorithm we can construct $\theta \in \Delta_X$ such that $\mathcal{G}(\beta^{(t)}, \theta) \leq \epsilon$, then the current primal iterate $\beta^{(t)}$ is guaranteed to be an ϵ-optimal solution of Problem (2.1) – meaning that it is at a distance less than ϵ of the optimum, in terms of objective function value. For this reason, the corresponding θ is sometimes called a *dual certificate*.

Algorithm 2.1 CYCLIC COORDINATE DESCENT FOR THE LASSO, WITH DUAL EXTRAPOLATION

input : $X = [X_{:1}|\dots|X_{:p}], y, \lambda, \beta^{(0)}, \epsilon$
param: $T, K = 5, f^{\text{dual}} = 10$
init : $r = r^{(0)} = y - X\beta^{(0)}, \theta^{(0)} = r/\max(\lambda, \|X^\top r\|_\infty)$

for $t = 1, \dots, T$ **do**
  if $t = 0 \mod f^{\text{dual}}$ **then** // θ every f^{dual} epoch only
    $s = t/f^{\text{dual}}$ // dual point indexing
    $r^{(s)} = r$
    compute $\theta^{(s)}_{\text{res}}$ and $\theta^{(s)}_{\text{accel}}$ with eqs. (2.5), (2.25) and (2.26)
    $\theta^{(s)} = \underset{\theta \in \{\theta^{(s-1)}, \theta^{(s)}_{\text{accel}}, \theta^{(s)}_{\text{res}}\}}{\arg\max} \mathcal{D}(\theta)$ // Equation (2.38)
    if $\mathcal{G}(\beta^{(t)}, \theta^{(s)}) < \epsilon$ **then**
      break
  for $j = 1, \dots, p$ **do**
    $\beta_j^{(t+1)} = \text{ST}\left(\beta_j^{(t)} + \frac{X_{:j}^\top r}{\|X_{:j}\|^2}, \frac{\lambda}{\|X_{:j}\|^2}\right)$
    if $\beta_j^{(t+1)} \neq \beta_j^{(t)}$ **then**
      $r \mathrel{+}= (\beta_j^{(t)} - \beta_j^{(t+1)}) X_{:j}$
return $\beta^{(t)}$, $\theta^{(s)}$

Since Equation (2.4) holds at optimality, a canonical choice of dual point is called *residuals rescaling* (Mairal, 2010). It consists in choosing, at iteration t where the gap is to be computed, a dual feasible point proportional to the residuals $r^{(t)} \triangleq y - X\beta^{(t)}$:

$$\theta^{(t)}_{\text{res}} \triangleq r^{(t)}/\max(\lambda, \|X^\top r^{(t)}\|_\infty) \ . \tag{2.5}$$

It is clear that if $\beta^{(t)}$ converges to $\hat{\beta}$, $\theta^{(t)}_{\text{res}}$ converges to $\hat{\theta}$, hence the duality gap for the pair $(\beta^{(t)}, \theta^{(t)}_{\text{res}})$ goes to 0. Additionally, the cost of computing $\theta^{(t)}_{\text{res}}$ is moderate: $\mathcal{O}(np)$, the same as a single proximal gradient descent step or an epoch of coordinate descent.

However, using rescaled residuals has two noticeable drawbacks: it ignores information from previous iterates, and rescaling the residuals $r^{(t)}$ makes an "unbalanced" use of computations in the sense that most of the burden is spent on improving β while θ is obtained by solving a crude 1D optimization problem, *i.e.,* $\min\{\alpha \in [\lambda, +\infty] : r^{(t)}/\alpha \in \Delta_X\}$. In practice (see Section 2.5), it turns out that, while safe and simple, such a construction massively overestimates the suboptimality gap, leading to slow safe feature identification and to numerical solvers running for more steps than actually needed. The new dual point construction we propose aims at improving upon this default strategy.

Before we focus on duality, let us mention that other criteria than suboptimality are also often considered. For instance, the solver can be stopped when the ℓ_2 or ℓ_∞ norm of $\beta^{(t)} - \beta^{(t-1)}$ goes below a threshold ϵ, or when the objective function stops decreasing fast enough ($\mathcal{P}(\beta^{(t-1)}) - \mathcal{P}(\beta^{(t)}) < \epsilon$). However, contrary to duality gap based stopping criteria, such heuristic rules do not offer a control on suboptimality. They are also tightly coupled with the value of the step size, making the use of a general ϵ difficult.

2.2.2 Dual extrapolation

Building on the work on nonlinear regularized acceleration by Scieur et al. (2016), we propose a new construction to obtain a better dual point. Instead of relying only on the last residuals $r^{(t)}$, the dual point is improved by extrapolating previous residuals, *i.e.,* using $r^{(t)}, r^{(t-1)}, r^{(t-2)}$, etc. We explain what is meant by "structure", and how to exploit it, in the following.

Definition 2.3 (Vector AutoRegressive sequence)**.** *We say that $(r^{(t)})_{t\in\mathbb{N}} \in (\mathbb{R}^n)^{\mathbb{N}}$ is a Vector AutoRegressive (VAR) sequence if there exists $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ such that for $t \in \mathbb{N}$:*

$$r^{(t+1)} = Ar^{(t)} + b \ . \tag{2.6}$$

Proposition 2.4 (Extrapolation for VAR sequences (Scieur, 2018, Thm 3.2.2))**.** *Let $(r^{(t)})_{t\in\mathbb{N}}$ be a VAR sequence in $\mathbb{R}^n$, satisfying $r^{(t+1)} = Ar^{(t)} + b$ with $A \in \mathbb{R}^{n\times n}$ a symmetric positive definite matrix such that $\|A\|_2 < 1$, $b \in \mathbb{R}^n$ and $K < n$. Assume that for $t \geq K$, the family $\{r^{(t-K)} - r^{(t-K+1)}, \dots, r^{(t-1)} - r^{(t)}\}$ is linearly independent and define:*

$$U^{(t)} \triangleq [r^{(t-K)} - r^{(t-K+1)}, \dots, r^{(t-1)} - r^{(t)}] \in \mathbb{R}^{n\times K} \ , \tag{2.7}$$

$$(c_1, \dots, c_K) \triangleq \frac{(U^{(t)\top}U^{(t)})^{-1}\mathbf{1}_K}{\mathbf{1}_K^\top (U^{(t)\top}U^{(t)})^{-1}\mathbf{1}_K} \in \mathbb{R}^K \ , \tag{2.8}$$

$$r_{\text{extr}} \triangleq \sum_{k=1}^{K} c_k r^{(t-K-1+k)} \in \mathbb{R}^n \ . \tag{2.9}$$

Then, r_{extr} satisfies:

$$\left\|Ar_{\text{extr}} - b - r_{\text{extr}}\right\| \leq \mathcal{O}(\rho^K) \ , \tag{2.10}$$

where $\rho \triangleq \frac{1-\sqrt{1-\|A\|_2}}{1+\sqrt{1-\|A\|_2}} < 1$.

The justification for this extrapolation procedure is the following: since $\|A\|_2 < 1$, $(r^{(t)})_{t\in\mathbb{N}}$ converges: let us call its limit $\hat{r}$. For $t \in \mathbb{N}$, we have $r^{(t+1)} - \hat{r} = A(r^{(t)} - \hat{r})$. Let $(a_0, \dots, a_n) \in \mathbb{R}^{n+1}$ be the coefficients of A's characteristic polynomial. By Cayley-Hamilton's theorem, $\sum_{k=0}^n a_k A^k = 0$. Given that $\|A\|_2 < 1$, 1 is not an eigenvalue of A and $\sum_{k=0}^n a_k \neq 0$, so we can normalize these coefficients to have $\sum_{k=0}^n a_k = 1$. For $t \geq n$, we have:

$$\sum_{k=0}^{n} a_k \Big(r^{(t-n+k)} - \hat{r}\Big) = \Big(\sum_{k=0}^{n} a_k A^k\Big)(r^{(t-n)} - \hat{r}) = 0 \ , \tag{2.11}$$

$$\text{and so} \qquad \sum_{k=0}^{n} a_k r^{(t-n+k)} = \sum_{k=0}^{n} a_k \hat{r} = \hat{r} \ . \tag{2.12}$$

Hence, $\hat{r} \in \text{Span}(r^{(t-n)}, \dots, r^{(t)})$.

Therefore, it is natural to seek to approximate $\hat{r}$ as an affine combination of the $(n+1)$ last iterates $(r^{(t-n)}, \dots, r^{(t)})$. Using $(n+1)$ iterates might be costly for large values of n, so one might rather consider only a smaller number K, *i.e.,* find $(c_1, \dots, c_K) \in \mathbb{R}^K$ such that $\sum_{k=1}^K c_k r^{(t-K-1+k)}$ approximates $\hat{r}$. Since $\hat{r}$ is a fixed point of $r \mapsto Ar + b$,

$\sum_{k=1}^K c_k r^{(t-K-1+k)}$ should be one too. Under the normalizing condition $\sum_{k=1}^K c_k = 1$, this means that the quantity

$$\sum_{k=1}^K c_k r^{(t-K-1+k)} - A\sum_{k=1}^K c_k r^{(t-K-1+k)} - b = \sum_{k=1}^K c_k r^{(t-K-1+k)} - \sum_{k=1}^K c_k \left(r^{(t-K+k)} - b\right) - b$$
$$= \sum_{k=1}^K c_k \left(r^{(t-K-1+k)} - r^{(t-K+k)}\right) \qquad (2.13)$$

should be as close to $\mathbf{0}_n$ as possible; this leads to solving:

$$\hat{c} = \underset{\substack{c\in\mathbb{R}^K \\ c^\top \mathbf{1}_K = 1}}{\arg\min} \left\| \sum_{k=1}^K c_k \left(r^{(t-K+k)} - r^{(t-K-1+k)}\right) \right\| , \qquad (2.14)$$

which admits a closed-form solution:

$$\hat{c} = \frac{(U^{(t)\top}U^{(t)})^{-1}\mathbf{1}_K}{\mathbf{1}_K^\top (U^{(t)\top}U^{(t)})^{-1}\mathbf{1}_K} , \qquad (2.15)$$

where $U^{(t)} = [r^{(t-K+1)} - r^{(t-K)}, \dots, r^{(t)} - r^{(t-1)}] \in \mathbb{R}^{n\times K}$. In practice, the next proposition shows that when $U^{(t)}$ does not have full column rank, it is theoretically sound to use a lower value for the number of extrapolation coefficients K.

Proposition 2.5. *If $U^{(t)\top}U^{(t)}$ is not invertible, then $\hat{r} \in \mathrm{Span}(r^{(t-1)}, \dots, r^{(t-K)})$.*

Proof Let $x \in \mathbb{R}^K \setminus \{\mathbf{0}_K\}$ be such that $U^{(t)\top}U^{(t)}x = \mathbf{0}_K$, with $x_K \neq 0$ (the proof is similar if $x_K = 0, x_{K-1} \neq 0$, etc.). Then $U^{(t)}x = \sum_{k=1}^K x_k(r^{(t-K+k)} - r^{(t-K+k-1)}) = \mathbf{0}_n$ and, setting $x_0 \triangleq 0$, $r^{(t)} = \frac{1}{x_K}\sum_{k=1}^K (x_k - x_{k-1}) r^{(t-K+k-1)} \in \mathrm{Span}(r^{(t-1)}, \dots, r^{(t-K)})$. Since $\frac{1}{x_K}\sum_{k=1}^K (x_k - x_{k-1}) = 1$, it follows that:

$$\begin{aligned} r^{(t+1)} &= Ar^{(t)} + b \\ &= \tfrac{1}{x_K}\sum_{k=1}^K (x_k - x_{k-1})(Ar^{(t-K+k-1)} + b) \\ &= \tfrac{1}{x_K}\sum_{k=1}^K (x_k - x_{k+1}) r^{(t-K+k)} \in \mathrm{Span}(r^{(t-1)}, \dots, r^{(t-K)}) , \end{aligned} \qquad (2.16)$$

and subsequently $r^{(s)} \in \mathrm{Span}(r^{(t-1)}, \dots, r^{(t-K)})$ for all $s \geq t$. By going to the limit, $\hat{r} \in \mathrm{Span}(r^{(t-1)}, \dots, r^{(t-K)})$. ∎

Using the identification property of coordinate descent and proximal gradient descent (proved in Theorem 3.7 for a wider class of problems), we can formalize the VAR behavior of dual iterates:

Theorem 2.6. *When $(\beta^{(t)})_{t\in\mathbb{N}}$ is obtained by a cyclic coordinate descent or proximal gradient descent applied to the Lasso problem, $(X\beta^{(t)})_{t\in\mathbb{N}}$ is a VAR sequence after sign identification.*

Proof Let $t \in \mathbb{N}$ denote an epoch after sign identification.

Coordinate descent: Let $j_1, \dots, j_S$ be the indices of the support of $\hat{\beta}$, in increasing order. As the sign is identified, coefficients outside the support are 0 and remain 0. We decompose the t-th epoch of coordinate descent into individual coordinate updates.

Let $\tilde{\beta}^{(0)} \in \mathbb{R}^p$ denote the initialization (*i.e.,* the beginning of the epoch, $\tilde{\beta}^{(0)} = \beta^{(t)}$), $\tilde{\beta}^{(1)} \in \mathbb{R}^p$ the iterate after coordinate j_1 has been updated, etc., up to $\tilde{\beta}^{(S)}$ after coordinate j_S has been updated, *i.e.,* at the end of the epoch ($\tilde{\beta}^{(S)} = \beta^{(t+1)}$).

Let $s \in [S]$, then $\tilde{\beta}^{(s)}$ and $\tilde{\beta}^{(s-1)}$ are equal everywhere, except at coordinate j_s:

$$\begin{aligned}\tilde{\beta}^{(s)}_{j_s} &= \mathrm{ST}\left(\tilde{\beta}^{(s-1)}_{j_s} + \frac{1}{\|X_{:j_s}\|^2} X_{:j_s}^\top \left(y - X\tilde{\beta}^{(s-1)}\right), \frac{\lambda}{\|X_{:j_s}\|^2}\right) \\ &= \tilde{\beta}^{(s-1)}_{j_s} + \frac{1}{\|X_{:j_s}\|^2} X_{:j_s}^\top \left(y - X\tilde{\beta}^{(s-1)}\right) - \frac{\lambda \operatorname{sign}(\hat{\beta}_{j_s})}{\|X_{:j_s}\|^2} \ , \end{aligned} \tag{2.17}$$

where we have used sign identification: $\operatorname{sign}(\tilde{\beta}^{(s)}_{j_s}) = \operatorname{sign}(\hat{\beta}_{j_s})$. Therefore:

$$\begin{aligned} X\tilde{\beta}^{(s)} - X\tilde{\beta}^{(s-1)} &= X_{:j_s}\left(\tilde{\beta}^{(s)}_{j_s} - \tilde{\beta}^{(s-1)}_{j_s}\right) \\ &= X_{:j_s}\left(\frac{X_{:j_s}^\top (y - X\tilde{\beta}^{(s-1)}) - \lambda \operatorname{sign}(\hat{\beta}_{j_s})}{\|X_{:j_s}\|^2}\right) \\ &= \frac{1}{\|X_{:j_s}\|^2} X_{:j_s} X_{:j_s}^\top \left(y - X\tilde{\beta}^{(s-1)}\right) - \frac{\lambda \operatorname{sign}(\hat{\beta}_{j_s})}{\|X_{:j_s}\|^2} X_{:j_s} \ . \end{aligned} \tag{2.18}$$

This leads to the following linear recurrent equation:

$$X\tilde{\beta}^{(s)} = \underbrace{\left(\mathrm{Id}_n - \frac{1}{\|X_{:j_s}\|^2} X_{:j_s} X_{:j_s}^\top\right)}_{A_s \in \mathbb{R}^{n\times n}} X\tilde{\beta}^{(s-1)} + \underbrace{\frac{X_{:j_s}^\top y - \lambda \operatorname{sign}(\hat{\beta}_{j_s})}{\|X_{:j_s}\|^2} X_{:j_s}}_{b_s \in \mathbb{R}^n} \ . \tag{2.19}$$

Hence, one gets recursively:

$$\begin{aligned} X\tilde{\beta}^{(S)} &= A_S X\tilde{\beta}^{(S-1)} + b_S \\ &= A_S A_{S-1} X\tilde{\beta}^{(S-2)} + A_S b_{S-1} + b_S \\ &= \underbrace{A_S \dots A_1}_{A} X\tilde{\beta}^{(0)} + \underbrace{A_S \dots A_2 b_1 + \dots + A_S b_{S-1} + b_S}_{b} \ . \end{aligned} \tag{2.20}$$

We can thus write the following VAR equations for $X\beta$ at the end of each coordinate descent epoch:

$$X\beta^{(t+1)} = AX\beta^{(t)} + b \ , \tag{2.21}$$

$$X\beta^{(t+1)} - X\hat{\beta} = A(X\beta^{(t)} - X\hat{\beta}) \ . \tag{2.22}$$

Proximal gradient: Recall that $\beta^{(t)}_{\mathcal{S}}$, $\hat{\beta}_{\mathcal{S}}$ and $X_{:\mathcal{S}}$ denote respectively $\beta^{(t)}$, $\hat{\beta}$ and X restricted to features in the support $\mathcal{S} = \operatorname{supp}(\hat{\beta})$. Notice that since we are in the identified sign regime, $X\beta^{(t)} = X_{:\mathcal{S}}\beta^{(t)}_{\mathcal{S}}$. With $L = \|X^\top X\|_2$, a proximal gradient descent update reads:

$$\begin{aligned} \beta^{(t+1)}_{\mathcal{S}} &= \mathrm{ST}\left(\beta^{(t)}_{\mathcal{S}} - \tfrac{1}{L} X_{:\mathcal{S}}^\top (X_{:\mathcal{S}}\beta^{(t)}_{\mathcal{S}} - y), \tfrac{\lambda}{L}\right) \\ &= \beta^{(t)}_{\mathcal{S}} - \tfrac{1}{L} X_{:\mathcal{S}}^\top \left(X_{:\mathcal{S}}\beta^{(t)}_{\mathcal{S}} - y\right) - \tfrac{\lambda}{L} \operatorname{sign}(\hat{\beta}_{\mathcal{S}}) \\ &= \left(\mathrm{Id}_S - \tfrac{1}{L} X_{:\mathcal{S}}^\top X_{:\mathcal{S}}\right) \beta^{(t)}_{\mathcal{S}} + \tfrac{1}{L} X_{:\mathcal{S}}^\top y - \tfrac{\lambda}{L} \operatorname{sign}(\hat{\beta}_{\mathcal{S}}) \ . \end{aligned} \tag{2.23}$$

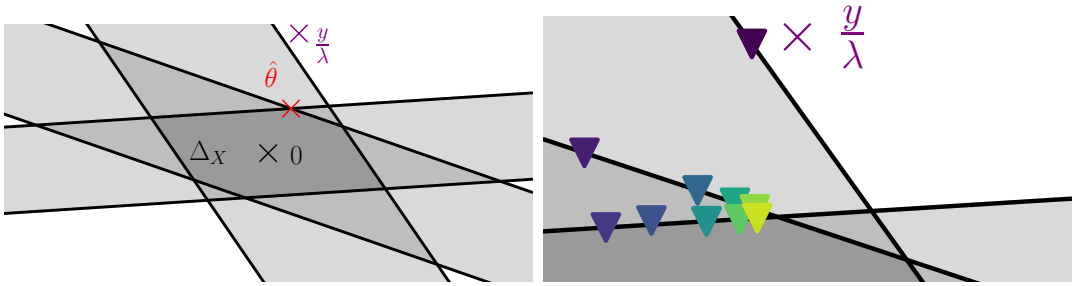


Figure 2.1 – Illustration of the VAR nature of the dual iterates of the Lasso, on a toy dataset with $n = 2$ and $p = 3$. Left: dual of the Lasso problem; the dual optimum $\hat{\theta}$ is the projection of y/λ onto Δ_X. Right: sequence of residuals after each update of coordinate descent (first iterates in blue, last in yellow). After four updates, the iterates alternate geometrically between the same two constraint hyperplanes.

Hence the equivalent of Equation (2.21) for proximal gradient descent is:

$$X\beta^{(t+1)} = \left(\mathrm{Id}_n - \tfrac{1}{L} X_{:\mathcal{S}} X_{:\mathcal{S}}^\top\right) X\beta^{(t)} + \tfrac{1}{L} X_{:\mathcal{S}} X_{:\mathcal{S}}^\top y - \tfrac{\lambda}{L} X_{:\mathcal{S}} \operatorname{sign}(\hat{\beta}_{\mathcal{S}}) \ . \tag{2.24}$$

■

Figure 2.1 represents the Lasso dual for a toy problem and illustrates the VAR nature of $r^{(t)}/\lambda$. As recently highlighted again by Tibshirani (2017) (see Section 2.2.3), the iterates $r^{(t)}/\lambda$ correspond to the iterates of Dykstra's algorithm to project y/λ onto Δ_X. During the first updates, the dual iterates do not have a regular trajectory. However, after a certain number of updates (after the sign identification epoch is reached), they alternate in a geometric fashion between the same two hyperplanes. In this regime, it becomes beneficial to use extrapolation to obtain a point closer to $\hat{\theta}$. We recall the correspondence between cyclic coordinate descent for the Lasso and Dykstra's algorithm more extensively in Section 2.2.3.

Remark 2.7. *Equation (2.20) shows why we combine extrapolation with* cyclic *coordinate descent: if the coefficients are not always updated in the same order (see Figures 2.2c and 2.2d), the matrix A depends on the epoch, and the VAR structure may no longer hold.*

Having highlighted the VAR behavior of $(X\beta^{(t)})_{t\in\mathbb{N}}$, we can introduce our proposed dual extrapolation.

Definition 2.8 (Extrapolated dual point for the Lasso)**.** *For a fixed number K of proximal gradient descent or coordinate descent epochs, let $r^{(t)}$ denote the residuals $y - X\beta^{(t)}$ at epoch t of the algorithm. We define the extrapolated residuals:*

$$r_{\text{accel}}^{(t)} = \begin{cases} r^{(t)}, & \text{if } t \leq K \ , \\ \displaystyle\sum_{k=1}^{K} c_k r^{(t+1-k)}, & \text{if } t > K \ . \end{cases} \tag{2.25}$$

where $c = (c_1, \dots, c_K)^\top \in \mathbb{R}^K$ is defined as in (2.15) with $U^{(t)} = [r^{(t+1-K)} - r^{(t-K)}, \dots, r^{(t)} - r^{(t-1)}] \in \mathbb{R}^{n\times K}$. Then, we define the extrapolated dual point as:

$$\theta_{\text{accel}}^{(t)} \triangleq r_{\text{accel}}^{(t)} / \max(\lambda, \|X^\top r_{\text{accel}}^{(t)}\|_\infty) \ . \tag{2.26}$$

In practice, we use $K = 5$ and do not compute $\theta^{(t)}_{\text{accel}}$ if $U^{(t)\top}U^{(t)}$ cannot be inverted. Additionally, to impose monotonicity of the dual objective, and guarantee an objective function at least as high as with rescaled residuals, we use as dual point at iteration t:

$$\theta^{(t)} = \underset{\theta \in \{\theta^{(t-1)}, \theta^{(t)}_{\text{accel}}, \theta^{(t)}_{\text{res}}\}}{\arg\max} \mathcal{D}(\theta) \ . \tag{2.27}$$

There are two reasons why the results of Proposition 2.4 cannot be straightforwardly applied to Equation (2.26):

1. the theoretical analysis by Scieur et al. (2016) requires A to be symmetrical, which is the case for proximal gradient descent but not for cyclic coordinate descent (as $\mathrm{Id}_n - X_{:j_s}X_{:j_s}^\top/\|X_{:j_s}\|^2$ and $\mathrm{Id}_n - X_{:j_{s'}}X_{:j_{s'}}^\top/\|X_{:j_{s'}}\|^2$ only commute if $X_{:j_s}$ and $X_{:j_{s'}}$ are collinear). To circumvent this issue, we can make A symmetrical: instead of considering cyclic updates, we could consider that iterates $\beta^{(t)}$ are produced by a cyclic pass over the coordinates, *followed by a cyclic pass over the coordinates in reverse order*. The matrix of the VAR in this case is no longer $A = A_S \dots A_1$, but $A_1 \dots A_S A_S \dots A_1 = A_1^\top \dots A_S^\top A_S \dots A_1 = A^\top A$ (the A_s's are symmetrical). A result from Bollapragada et al. (2018) indicates that the bound still holds for a non-symmetric A (coherent with the practical results from Section 3.5), at the price of a much more complex analysis. Therefore, we still use regular cyclic passes over the features.

2. for both proximal gradient and coordinate descent we have $\|A\|_2 = 1$ instead of $\|A\|_2 < 1$ as soon as $S < n$: if the support of $\hat{\beta}$ is of size smaller than n ($S < n$), 1 is an eigenvalue of A. Indeed, for coordinate descent, if $S < n$, there exists a vector $u \in \mathbb{R}^n$, orthogonal to the S vectors $X_{:j_1}, \dots, X_{:j_s}$. The matrix $A_s = \mathrm{Id}_n - \frac{1}{\|X_{:j_s}\|^2} X_{:j_s}X_{:j_s}^\top$ corresponding the orthogonal projection onto $\mathrm{Span}(X_{:j_s})^\perp$, we therefore have $A_s u = u$ for every $s \in [S]$, hence $Au = u$. For proximal gradient descent, $\frac{1}{L}X_{:\mathcal{S}}X_{:\mathcal{S}}^\top$ is not invertible when $S < n$, hence 1 is an eigenvalue of $\mathrm{Id}_n - \frac{1}{L}X_{:\mathcal{S}}X_{:\mathcal{S}}^\top$. This seems to contradict the convergence of the VAR sequence, but is addressed in Lemmas 2.9 and 2.10.

Lemma 2.9. *For coordinate descent, if an eigenvalue of $A = A_S \dots A_1$ has modulus 1, it is equal to 1.*

Proof The matrix $A_s = \mathrm{Id}_n - \frac{1}{\|X_{:j_s}\|^2} X_{:j_s}X_{:j_s}^\top$ corresponds to the orthogonal projection onto $\mathrm{Span}(X_{:j_s})^\perp$. Hence,

$$\forall x \in \mathbb{R}^n, \|A_s x\| = \|x\| \implies A_s x = x \ . \tag{2.28}$$

Let $(\mu, x) \in \mathbb{C} \times \mathbb{R}^n$ s.t. $|\mu| = 1$, $\|x\| = 1$ and $Ax = \mu x$. This means $\|Ax\| = 1$. Because $\|A_1 x\| < 1 \implies \|A_S \dots A_1 x\| \leq \|A_S \dots A_2\| \|A_1 x\| < 1 \implies \|Ax\| < 1$, we must have $\|A_1 x\| \geq 1$. Since it holds that $\|A_1 x\| \leq \|x\| = 1$, we have $\|A_1 x\| = \|x\|$, thus $A_1 x = x$ because A_1 is an orthogonal projection. By a similar reasoning, $A_2 x = x$, etc. up to $A_S x = x$, hence $Ax = x$ and $\mu = 1$. ∎

Lemma 2.10. *For coordinate descent (*resp. *proximal gradient descent) applied to solve the Lasso, the VAR parameters $A \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$ defined in (2.20) (*resp. *(2.24)) satisfy $b \in \mathrm{Ker}(\mathrm{Id}_n - A)^\perp$.*

Proof *Coordinate descent case:* Let us remind that $b = A_S \dots A_2 b_1 + \dots + A_S b_{S-1} + b_S$ in this case, with $b_s = X_{:j_s}^\top y - \lambda \operatorname{sign}(\hat{\beta}_{j_s}) X_{:j_s} / \|X_{:j_s}\|^2$. Let $v \in \operatorname{Ker}(\mathrm{Id}_n - A)$. Following the proof of Lemma 2.9, we have $A_1 v = \dots = A_S v = v$. For $s \in [S]$, since A_s is the projection on $\operatorname{Span}(X_{:j_s})^\perp$, this means that v is orthogonal to $X_{:j_s}$. Additionally, $v^\top A_S \dots A_{s+1} b_s = (A_{s+1} \dots A_S v)^\top b_s = v^\top b_s = 0$ since b_s is collinear to $X_{:j_s}$. Thus, v is orthogonal to the S terms which compose b, and $b \perp \operatorname{Ker}(\mathrm{Id}_n - A)$.

Proximal gradient descent case: Let $v \in \operatorname{Ker}(\mathrm{Id}_n - A) = \operatorname{Ker}(X_{:\mathcal{S}} X_{:\mathcal{S}}^\top)$. We have $v^\top X_{:\mathcal{S}} X_{:\mathcal{S}}^\top v = 0 = \|X_{:\mathcal{S}}^\top v\|^2$, hence $X_{:\mathcal{S}}^\top v = 0$. It is now clear that $v^\top b = v^\top (-X_{:\mathcal{S}} X_{:\mathcal{S}}^\top y + \lambda X_{:\mathcal{S}} \operatorname{sign} \hat{\beta})/L = 0$, hence $b \perp \operatorname{Ker}(\mathrm{Id}_n - A)$. ■

Proposition 2.11. *Proposition 2.4 holds for the residuals $r^{(t)}$ (produced either by proximal gradient descent or coordinate descent) even though $\|A\|_2 = 1$ in both cases.*

Proof Let us write $A = \bar{A} + \underline{A}$ with $\bar{A}$ the orthogonal projection on $\operatorname{Ker}(\mathrm{Id}_n - A)$. By Lemma 2.9, $\|\underline{A}\|_2 < 1$.

Then, one can check that $A\underline{A} = \underline{A}^2$ and $A\bar{A} = \bar{A}^2 = \bar{A}$ and $Ab = \underline{A}b$.

Let T be the epoch when support identification is achieved. For $t \geq T$, we have:

$$r^{(t+1)} = \underline{A} r^{(t)} + b + \bar{A} r^{(T)} \ . \tag{2.29}$$

Indeed, it is trivially true for $t = T$ and if it holds for t,

$$\begin{aligned} r^{(t+2)} &= A r^{(t+1)} + b \\ &= A(\underline{A} r^{(t)} + b + \bar{A} r^{(T)}) + b \\ &= \underline{A}^2 r^{(t)} + \underline{A} b + \bar{A} r^{(T)} + b \\ &= \underline{A}(\underline{A} r^{(t)} + b) + \bar{A} r^{(T)} + b \\ &= \underline{A} r^{(t+1)} + \bar{A} r^{(T)} + b \ . \end{aligned} \tag{2.30}$$

Therefore, on $\operatorname{Ker}(\mathrm{Id}_n - A)$, the sequence $(r^{(t)})_{t \in \mathbb{N}}$ is constant, and on the orthogonal of $\operatorname{Ker}(\mathrm{Id}_n - A)$ it is a VAR sequence with associated matrix $\underline{A}$, whose spectral norm is strictly less than 1. Therefore, the results of Proposition 2.4 still hold. ■

Although until now we have proven results for both coordinate descent and proximal gradient descent for the sake of generality, we observed that coordinate descent consistently converges faster (see Figure 1.4, page 30). Hence from now on, we will only consider coordinate descent.

2.2.3 Dual perspective on coordinate descent

Algorithm 2.2 DYKSTRA'S ALTERNATING PROJECTIONS

input : $\Pi_{C_1}, \dots, \Pi_{C_p}, z$
init : $\theta = z, q_1 = 0, \dots, q_p = 0$
for $t = 1, \dots$ **do**
  for $j = 1, \dots, p$ **do**
    $\tilde{\theta} \leftarrow \theta + q_j$
    $\theta \leftarrow \Pi_{C_j}(\tilde{\theta})$
    $q_j \leftarrow \tilde{\theta} - \theta$
return θ

Algorithm 2.3 DYKSTRA FOR THE LASSO DUAL

input : $X = [X_{:1}| \dots |X_{:p}], y, \lambda$
init : $r = y, \tilde{\beta}_1 = 0, \dots, \tilde{\beta}_p = 0$
for $t = 1, \dots$ **do**
  for $j = 1, \dots, p$ **do**
    $\tilde{r} \leftarrow r + X_{:j}\tilde{\beta}_j$
    $r \leftarrow \tilde{r} - \mathrm{ST}\left(\frac{X_{:j}^\top \tilde{r}}{\|X_{:j}\|^2}, \frac{1}{\|X_{:j}\|^2}\right) \cdot X_{:j}$
    $\tilde{\beta}_j \leftarrow \mathrm{ST}\left(\frac{X_{:j}^\top \tilde{r}}{\|X_{:j}\|^2}, \frac{1}{\|X_{:j}\|^2}\right)$
return r/λ

In this section, we provide some insights on the efficiency of extrapolation for cyclic coordinate descent, described in Algorithm 2.1, by studying its connections with Dykstra's algorithm (Dykstra, 1983). As a reminder, note that the points extrapolated are only the residuals $r^{(t)}$ obtained every f^{dual} epochs: performing extrapolation and gap computation at every coordinate descent update would be time consuming.

Dykstra's algorithm aims at solving problems of the form:

$$\hat{\theta} = \underset{\theta \in \cap_{j=1}^p C_j}{\arg\min} \left\| z - \theta \right\|^2 \quad , \tag{2.31}$$

where $C_1, \dots, C_p$ are p closed convex sets, with associated projections $\Pi_{C_1}, \dots, \Pi_{C_p}$. The iterates of the (cyclic[1]) Dykstra algorithm are defined in Algorithm 2.2 (see Bauschke and Combettes (2011, Th. 29.2) for a convergence proof in the cyclic case).

The connection with coordinate descent for the Lasso has already been noticed (Tibshirani, 2017). In the Lasso dual, the closed convex sets are the p slabs $C_j = \{\theta \in \mathbb{R}^n : -1 \leq X_{:j}^\top \theta \leq 1\}$, and the point to be projected is $z = y/\lambda$. In this context, Dykstra's algorithm produces (non-necessarily feasible) iterates converging to $\hat{\theta}$.

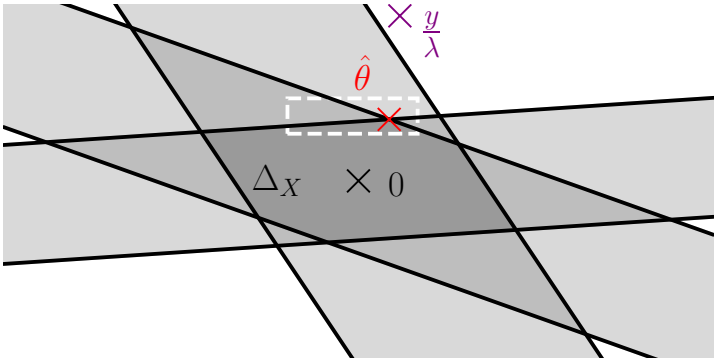
The connection with coordinate descent can be made noticing that:

$$(\mathrm{Id}_n - \Pi_{C_j})(\theta) = \mathrm{ST}(X_{:j}^\top \theta / \|X_{:j}\|^2, 1/\|X_{:j}\|^2) \cdot X_{:j} \quad . \tag{2.32}$$

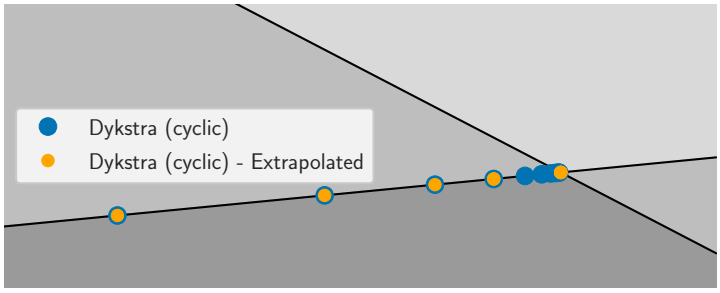
Using the change of variable $r = \lambda\theta$, $\tilde{r} = \lambda\tilde{\theta}$ and $q_j = X_{:j}\beta_j/\lambda$ and the previous expression, Algorithm 2.2 is equivalent to Algorithm 2.3. It is to be noted that this is exactly cyclic coordinate descent for the Lasso, where the output r of the algorithm corresponds to the residuals (and not the primal estimate β).

On Figure 2.2, we illustrate Problem (2.2) for $n = 2$, $p = 3$ (Figure 2.2a). Figure 2.2b (*resp.* Figure 2.2c) shows the iterates at the end of each epoch, produced by the cyclic (*resp.* shuffle) Dykstra algorithm, and their extrapolated version for $K = 4$. This corresponds to Algorithm 2.1 with $f^{\text{dual}} = 1$. On Figure 2.2b, the iterates eventually always lie on the same hyperplane, and exhibit a VAR structure while converging to $\hat{\theta}$: using only the last $K = 4$ points, extrapolation finds the true solution up to machine precision at the 5th iteration (Figure 2.2d). On the contrary, when the projection order on C_1 and C_2 is shuffled (Figure 2.2c), the iterates might not lie on the same hyperplane,

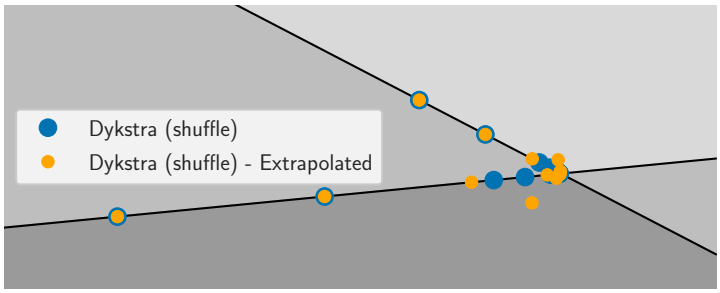
[1] in the *shuffle* variant, the order is shuffled after each epoch



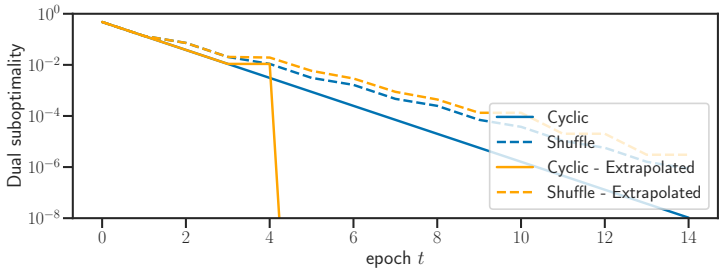
(a) Lasso dual problem with $X \in \mathbb{R}^{2\times 3}$. A close-up on the dashed rectangle around $\hat{\theta}$ is given in Figure 2.2b and Figure 2.2c.



(b) Close up for cyclic Dykstra in the dual (end of each epoch)



(c) Close up for shuffle Dykstra in the dual (end of each epoch)



(d) Dual suboptimality with and without extrapolation

Figure 2.2 – In the Lasso case, the dual solution $\hat{\theta}$ is the projection of y/λ onto the convex set Δ_X (the intersection of the three slabs).

and the trajectory tends to be less regular and harder to extrapolate. In what follows, we only consider cyclic orders due to their appealing interplay with extrapolation.

2.3 Gap Safe screening

The Lasso gives sparse solutions, meaning that $\|\hat{\beta}\|_0 \ll p$. Hence, if it were possible to discard features whose associated final coefficients vanish, the problem would become much smaller while having the same solutions. Discarding such features is called *screening*, and a key proposition for screening rules is the following:

$$\forall j \in [p], |X_{:j}^\top \hat{\theta}| < 1 \Rightarrow \hat{\beta}_j = 0 \ . \tag{2.33}$$

Hence, the knowledge of $\hat{\theta}$ allows to identify the *equicorrelation set* $\{j \in [p] : |X_{:j}^\top \hat{\theta}| = 1\}$. The problem restricted to the equicorrelation set has the same solutions as Problem (2.1), while being simpler: it typically has far less features. However, $\hat{\theta}$ is unknown so Equation (2.33) is not practical. To address this issue, Fercoq et al. (2015) have introduced the Gap Safe rules to remove the j-th feature:

$$|X_{:j}^\top \theta| < 1 - \|X_{:j}\| \sqrt{\frac{2}{\lambda^2} \mathcal{G}(\beta, \theta)} \Rightarrow \hat{\beta}_j = 0 \ , \tag{2.34}$$

which for any primal-dual feasible pair $(\beta, \theta) \in \mathbb{R}^p \times \Delta_X$, is *safe*, meaning that it will not wrongly discard a feature.

The Gap Safe rules have the appealing property of being convergent: at optimality, features not in the equicorrelation set have all been discarded. Additionally, they can be applied in a safe way in a sequential setting (Ndiaye et al., 2017b) when only an approximate solution of Problem (2.1) is available for a λ' close to λ (*e.g.,* when testing multiple regularization parameters in a cross-validation setting). Dynamic screening (Bonnefoy et al., 2014, 2015) is also possible using an iterate $\beta^{(t)}$ for a solver converging to $\hat{\beta}$: more and more features can be discarded along iterations.

Gap Safe rules performance depends strongly on how well θ approximates $\hat{\theta}$. Hence, θ acts as a certificate to discard irrelevant features: if the duality gap is large, the upper bound in Equation (2.34) is crude, resulting in fewer (possibly) discarded features. Section 2.5.3 shows that θ_{accel} helps discarding more features than θ_{res}, thus accelerating coordinate descent solvers and achieving safe feature identification in fewer epochs.

A potential drawback of screening rules is that, if the first duality gaps are large, coordinate descent computations are wasted on useless features during the first iterations (note that this is not the case when an approximation is available, *e.g.,* when performing cross-validation; see Section 2.5.4). In the next section, we design a WS strategy to address this issue.

2.4 Working sets with aggressive gap screening

Working set (WS) approaches involve two nested iteration loops: in the outer one, a set of features $\mathcal{W}^{(t)} \subset [p]$ is defined. In the inner one, an iterative algorithm is launched to solve the problem restricted to $X_{:\mathcal{W}^{(t)}}$ (*i.e.,* considering only the features in $\mathcal{W}^{(t)}$). In this section, we propose a working set construction based on an aggressive use of Gap Safe rules.

As it appears in Equation (2.34), the critical quantity measuring the importance of the j-th feature is:

$$d_j(\theta) \triangleq \frac{1 - |X_{:j}^\top \theta|}{\|X_{:j}\|} \ , \tag{2.35}$$

because:

$$d_j(\theta) > \sqrt{\frac{2}{\lambda^2}\mathcal{G}(\beta, \theta)} \Rightarrow \hat{\beta}_j = 0 \ . \tag{2.36}$$

Rather than discarding feature j from the problem if $d_j(\theta)$ is too large, the working set is composed of the coordinates achieving the lowest $d_j(\theta)$'s values. To do so, a first approach would consist in introducing a parameter $\rho \in]0, 1[$ and creating a working set with features such that $d_j(\theta) < \rho\sqrt{2\mathcal{G}(\beta, \theta)/\lambda^2}$. However, a pitfall for this strategy is that the working set size is not explicitly under control: an inaccurate choice of ρ could lead to extremely large working sets, and would limit their benefits. Instead, to achieve a good control on the working set growth, we reorder the $d_j(\theta)$'s in a non-decreasing way: $d_{j_p}(\theta) \geq \cdots \geq d_{j_1}(\theta)$. Then, for a given working set size $p^{(t)}$, we choose:

$$\mathcal{W}^{(t)} = \{j_1, \ldots, j_{p^{(t)}}\} \ . \tag{2.37}$$

This notation is clearly ambiguous, as j_1 also depends on t. To avoid the heavy notation $j_1^{(t)}$, we warn the reader at this point: the value of j_1 changes from one working set definition to the next.

When $\theta = \theta_{\text{res}}^{(t)}$ and the features are normalized (a common, but not systematic preprocessing step), this working set construction simply consists in finding the $X_{:j}$'s achieving the largest correlation with the residual, *i.e.,* finding the features leading to the largest $|X_{:j}^\top r^{(t)}|$. Writing the data-fitting term $F(\beta) = \left\|y - X\beta\right\|^2/2$, and checking that $\nabla_j F(\beta^{(t)}) = -X_{:j}^\top r^{(t)}$, then the previous rule coincides with gradient-based and correlation-based ones (Stich et al., 2017; Perekrestenko et al., 2017):

$$\begin{aligned} 1 - d_j(\theta_{\text{res}}^{(t)}) &= |X_{:j}^\top r^{(t)}| / \max(\lambda, \|X^\top r^{(t)}\|_\infty) \\ &\propto |X_{:j}^\top r^{(t)}| = |\nabla_j F(\beta^{(t)})| \ . \end{aligned}$$

However, the advantage of Equation (2.35) is that there is no restriction on the choice of the dual feasible point $\theta \in \Delta_X$. If a better candidate than rescaled residuals is available, it should be used instead. Considering the (ideal) case where the dual point constructed is $\hat{\theta}$, then the working set rule (2.37) yields the equicorrelation set (if $p^{(t)}$ has the proper value), which is the best performance to expect in general for a working set construction.

When subproblems are solved with the same precision ϵ as considered for stopping the outer-loop and if the working set $\mathcal{W}^{(t)}$ grows geometrically (*e.g.,* $p^{(t+1)} = 2p^{(t)}$) and monotonically (*i.e.,* $\mathcal{W}^{(t)} \subset \mathcal{W}^{(t+1)}$), then convergence is guaranteed provided the inner solver converges. Indeed, this growth strategy guarantees that as long as the problem has not been solved up to precision ϵ, more features are added, eventually starting the inner solver on the full problem until it reaches an ϵ-solution. The initial working set size is set to $p^{(1)} = 100$, except when an initialization $\beta^{(0)} \neq \mathbf{0}_p$ is provided (*e.g.,* for path or sequential computations, see Section 2.5.4), in which case we set $p^{(1)} = \|\beta^{(0)}\|_0$. This working set construction has many advantages: as it only requires a dual point, it is flexible and can be adapted to other objective functions (contrary to approaches such as Kim and Park (2010) which need to rewrite the Lasso as a Quadratic Program). Moreover, exact resolution of the subproblems is not required for convergence. Our policy to choose $p^{(t)}$ avoids two common working set drawbacks: working sets growing one feature at a time, and cyclic behaviors, *i.e.,* features entering and leaving the working set repeatedly.

Algorithm 2.4 Celer

input : $X, y, \lambda, \beta^{(0)}$
param: $p_{\text{init}} = 100, \epsilon, \underline{\epsilon} = 0.3, T, \text{prune} = \text{True}$
init : $\theta^{(0)} = \theta^{(0)}_{\text{inner}} = y/\|X^\top y\|_\infty$

if $\beta^{(0)} \neq \mathbf{0}_p$ **then** // warm start
  $p^{(1)} = \|\beta^{(0)}\|_0$
else
  $p^{(1)} = p_{\text{init}}$
for $t = 1, \dots, T$ **do**
  compute $\theta^{(t)}_{\text{res}}$
  $\theta^{(t)} = \arg\max_{\theta \in \{\theta^{(t-1)}, \theta^{(t-1)}_{\text{inner}}, \theta^{(t)}_{\text{res}}\}} \mathcal{D}(\theta)$ // Equation (2.5)
  $g^{(t)} = \mathcal{G}(\beta^{(t-1)}, \theta^{(t)})$ // global gap
  if $g^{(t)} \leq \epsilon$ **then**
    break
  for $j = 1, \dots, p$ **do**
    compute $d^{(t)}_j = (1 - |X_{:j}^\top \theta^{(t)}|)/\|X_{:j}\|$
  if prune **then**
    $\epsilon_t = \underline{\epsilon} g^{(t)}$
    set $(d^{(t)})_{\text{supp}(\beta^{(t-1)})} = -1$ // monotonicity
    if $t \geq 2$ **then**
      $p^{(t)} = \min(2\|\beta^{(t-1)}\|_0, p)$ // Equation (2.39)
  else
    $\epsilon_t = \epsilon$
    set $(d^{(t)})_{\mathcal{W}^{(t-1)}} = -1$ // monotonicity
    if $t \geq 2$ **then**
      $p^{(t)} = \min(2p^{(t-1)}, p)$ // doubling size
  $\mathcal{W}^{(t)} = \{j \in [p] : d^{(t)}_j$ among $p^{(t)}$ smallest values of $d^{(t)}\}$
  // Approximately solve sub-problem :
  get $\tilde{\beta}^{(t)}, \theta^{(t)}_{\text{inner}}$ with Algorithm 2.1 applied to $(y, X_{:\mathcal{W}^{(t)}}, \lambda, (\beta^{(t-1)})_{\mathcal{W}^{(t)}}, \epsilon_t)$
  set $\beta^{(t)} = \mathbf{0}_p$ and $(\beta^{(t)})_{\mathcal{W}^{(t)}} = \tilde{\beta}^{(t)}$
  $\theta^{(t)}_{\text{inner}} = \theta^{(t)}_{\text{inner}} / \max(\lambda, \|X^\top \theta^{(t)}_{\text{inner}}\|_\infty)$
return $\beta^{(t)}, \theta^{(t)}$

We have coined our proposed algorithm implementing this working set strategy with dual extrapolation Celer (Constraint Elimination for the Lasso with Extrapolated Residuals).

2.5 Experiments

2.5.1 Practical implementation

The implementation is done in Python and Cython (Behnel et al., 2011). It is available at https://github.com/mathurinm/celer, and made available as a pip-installable Python package which also contains the further developments of Chapter 3.

Linear system If the linear system $(U^{(t)})^\top U^{(t)} z = \mathbf{1}_K$ is ill-conditioned, rather than using Tikhonov regularization and solve $((U^{(t)})^\top U^{(t)} + \gamma I)z = \mathbf{1}_K$ as proposed in Scieur et al. (2016), we stop the computation for θ_{accel} and set $\mathcal{D}(\theta_{\text{accel}}) = -\infty$ for this iteration. In practice, this does not prevent the proposed methodology from computing significantly lower gaps than the standard approach.

Practical cost of dual extrapolation The storage cost of dual extrapolation is $\mathcal{O}(nK)$ (storing $r^{(t)}, \dots, r^{(t-K)}$). The main computation cost lies in the dual rescaling of the extrapolated residuals, which is $\mathcal{O}(np)$, and corresponds to the same cost as an epoch of CD/ISTA. The cost of computing c is small, since the matrix $(U^{(t)})^\top U^{(t)}$ is only $K \times K$. One should notice that there is no additional cost to compute the residuals: in reasonable coordinate descent implementations, they have to be maintained at all iterations to avoid costly partial gradients computation (see Algorithm 2.1); for ISTA their computation is also required at each epoch to evaluate the gradients $X^\top r^{(t)}$. As usual for iterative algorithms, we do not compute the duality gap (nor the dual points) at every update of β, but rather after every $f^{\text{dual}} = 10$ CD/ISTA epochs[2]. This makes the cost of dual extrapolation small compared to the iterations in the primal. The influence of f^{dual} and K in practice is illustrated in additional experiments in Appendix A.1.1.

Robustifying dual extrapolation Even if in practice we have observed fast convergence of $\theta^{(t)}_{\text{accel}}$ towards $\hat{\theta}$, we cannot provide guarantees about the behavior of $\theta^{(t)}_{\text{accel}}$ when the residuals are constructed from iterates of coordinate descent or other algorithms. Hence, for a cost of $\mathcal{O}(np)$, in Algorithm 2.1 we also compute $\theta^{(t)}_{\text{res}}$ and use as dual point:

$$\theta^{(t)} = \underset{\theta \in \{\theta^{(t-1)}, \theta^{(t)}_{\text{accel}}, \theta^{(t)}_{\text{res}}\}}{\arg\max} \mathcal{D}(\theta) \ . \tag{2.38}$$

Taking into account the previous dual point $\theta^{(t-1)}$ ensures monotonic improvements on the dual objective. The total computation cost of the dual is only doubled, which remains small compared to the cost of f^{dual} epochs of ISTA/CD, while guaranteeing monotonicity of the dual objective, and a behavior at least as good as $\theta^{(t)}_{\text{res}}$.

Pruning While the monotonic geometric growth detailed in Section 2.4 guarantees convergence, if $p^{(1)}$ is chosen too large, the working sets will never decrease. To remediate this, we introduce a variant called *pruning*:

$$p^{(t)} = \min(2\|\beta^{(t-1)}\|_0, p) \ , \tag{2.39}$$

in which $\mathcal{W}^{(t)}$ approximately doubles its size at each iteration. This still guarantees that, even if $p^{(1)}$ was set too small, $p^{(t)}$ will grow quickly to reach the correct value. On the other hand, if $p^{(1)}$ is too big, many useless features are included at the first iteration, but is is likely that their coefficients will be 0, and hence $\|\beta^{(1)}\|_0$ will be small, making $p^{(2)}$ small. This is illustrated by an experiment in Appendix A.1.2.

2.5.2 Higher dual objective

We start by investigating the efficiency of our dual point in a case where λ is fixed. Figure 2.3 shows, for the coordinate descent solver given in Algorithm 2.1, the duality gaps evaluated with the standard approach $\mathcal{P}(\beta^{(t)}) - \mathcal{D}(\theta^{(t)}_{\text{res}})$ and our proposed dual

[2]This explains why the indices for β and θ differ in Algorithm 2.1

extrapolation $\mathcal{P}(\beta^{(t)}) - \mathcal{D}(\theta^{(t)}_{\text{accel}})$, as well as the exact suboptimality $\mathcal{P}(\beta^{(t)}) - \mathcal{P}(\hat{\beta})$ (note that the latter is not available to the practitioner before convergence). The experiment is performed on the *leukemia* dataset ($n = 72, p = 7129$), with the design matrix columns set to unit ℓ_2-norm, and y centered and set to unit ℓ_2-norm so that the first primal objective is $\mathcal{P}(\mathbf{0}_p) = 0.5$. The algorithm is run without warm start ($\beta^{(0)} = \mathbf{0}_p$) for $\lambda = \lambda_{\max}/20$, and the values of $\theta^{(t)}_{\text{accel}}$ and $\theta^{(t)}_{\text{res}}$ are monitored[3]. $\mathcal{P}(\hat{\beta})$ is obtained by running the solver up to machine precision.

For a better analysis of the impact of dual extrapolation, in this experiment (and here only), we have not imposed monotonicity of the various dual objectives, nor have we used the best of both points as proposed in Equation (2.38).

As claimed in Section 2.2, we can observe that θ_{res} massively overestimates the suboptimality gap: while a true suboptimality gap of 10^{-6} is reached around epoch 200, the classical upper bound achieves this value at epoch 400 only. This means that if the duality gap were used as stopping criterion, the solver would run for twice too long. On the contrary, after a number of iterations where it behaves like the canonical approach, the proposed choice θ_{accel} accelerates and provides a duality gap much closer to the true suboptimality. After a sufficient number of epochs, the two are even almost equal, meaning that $\theta^{(t)}_{\text{accel}}$ is extremely close to $\hat{\theta}$. The difference between the two approaches is particularly striking for low values of ϵ. We also see, that, although more bumpy than the standard approach, our proposed duality gap does not behave erratically. Hence, stabilizing it as stated in Equation (2.38) does not seem mandatory (but since it is cheap, we still do it for other experiments). Practical choices of f and K are discussed in Appendix A.1.1. The same behavior is visible in the following chapter (Figure 3.1a, 80).

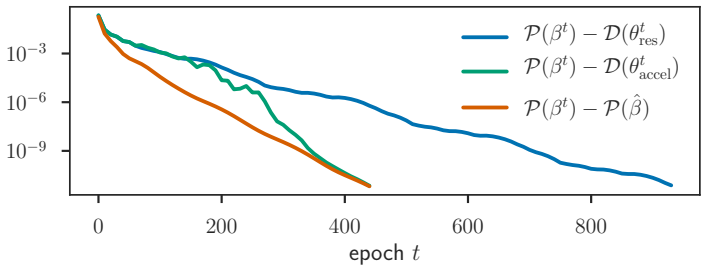


Figure 2.3 – Duality gaps evaluated with the canonical dual point θ_{res} and the proposed construction θ_{accel}, along with the true suboptimality gap. Performance is measured for Algorithm 2.1 on the *leukemia* dataset, for $\lambda = \lambda_{\max}/20$. Our duality gap quickly gets close to the true suboptimality, while the canonical approach constantly overestimates it.

2.5.3 Better Gap Safe screening performance

Figure 2.3 shows that extrapolated residuals yield tighter estimates of the suboptimality gap than rescaled residuals. However, one may argue either that using the duality gap as stopping criterion is infrequent (let us nevertheless mention that this criterion is for example the one implemented in the popular package `scikit-learn` (Pedregosa et al.,

[3] $\lambda_{\max} \triangleq \|X^\top y\|_\infty$ is the smallest λ s.t. $\hat{\beta} = \mathbf{0}_p$

2011)), or that vanilla coordinate descent is seldom implemented alone, but rather combined with screening or working set techniques. Here we demonstrate the benefit of the proposed extrapolation when combined with screening: the number of screened features grows more quickly when our new dual construction is used. This leads to faster coordinate descent solvers, and quicker safe feature identification.

The dataset for this experiment is the Finance/E2006-log1p dataset (publicly available from LIBSVM[4]), preprocessed as follows: features with strictly less than 3 non-zero entries are removed, features are set to unit ℓ_2-norm, y is centred and set to unit ℓ_2-norm, and an unregularized intercept feature is added. After preprocessing, $n = 16\,087$ and $p = 1\,668\,738$.

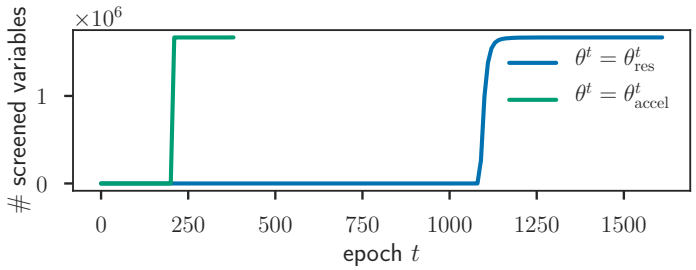


Figure 2.4 – Number of variables discarded by the (dynamic) Gap Safe rule as a function of epochs of Algorithm 2.1, depending on the dual point used, for $\lambda = \lambda_{\max}/5$ (Finance dataset).

Figure 2.4 shows the number of screened variables as a function of the number of epochs in Algorithm 2.1, when using either standard residual rescaling or dual extrapolation to get the dual point $\theta^{(t)}$ in Equation (2.34). The solver stops once a duality gap of 10^{-6} is reached. We can see that the faster convergence of $\theta^{(t)}_{\text{accel}}$ towards $\hat{\theta}$ observed in Figure 2.3 translates into a better Gap Safe screening: features are discarded in fewer epochs than when $\theta^{(t)}_{\text{res}}$ is used. The gain in number of screened variables is directly reflected in terms of computation time: 70 s for the proposed approach, compared to 290 s for Gap Safe rule with rescaled residuals.

2.5.4 Working sets application to Lasso path

In practice, it rarely happens that the solution of Problem (2.1) must be computed for a single λ: the ideal value of the regularization parameter is not known, and $\hat{\beta}$ is computed for several λ's, before the best is selected (*e.g.,* by cross-validation). The values of λ are commonly[5] chosen on a logarithmic grid of 100 values between $\lambda_{\max}$ and $\lambda_{\max}/10^2$ or $\lambda_{\max}/10^3$. For the Finance dataset, we considered $\lambda_{\max}/10^2$, leading to a support of size 15 000. In such sequential context, warm start is standard and we implement it for all algorithms. It means that all solvers computing $\hat{\beta}$ are initialized with the approximate solution obtained for the previous λ on the grid (starting from $\lambda_{\max}$).

We refer to the PhD work Johnson (2018, Section 3.7) for a very extensive comparison which shows that Blitz outperforms Lasso solvers such as L1_LS (Kim et al., 2007), APPROX (Fercoq and Richtárik, 2015) or GLMNET (Friedman et al., 2010) on a large

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.
[5] this is the default grid in GLMNET or scikit-learn

collections of datasets and settings. In our experiments, we use Blitz's C++ open source implementation, available at https://github.com/tbjohns/BlitzL1/.

Path computation For a fine (*resp.* coarse) grid of 100 (*resp.* 10) values of λ geometrically distributed between $\lambda_{\max}$ and $\lambda_{\max}/100$, the competing algorithms solve the Lasso on various real world datasets from LIBLINEAR (Fan et al., 2008). Warm start is used for all algorithms: except for the first value of λ, the algorithms are initialized with the solution obtained for the previous value of λ on the path. Note that paths are computed following a decreasing sequence of λ (from high value to low). Computing Lasso solutions for various values of λ is a classical task, in cross-validation for example. The values we choose for the grid are the default ones in `scikit-learn` or GLMNET. For Gap Safe Rules (GSR), we use the strong warm start variant which was shown by Ndiaye et al. (2017b, Section 4.6.4) to have the best performance. We refer to "GSR + extr." when, on top of this, our proposed dual extrapolation technique is used to create the dual points for screening. To evaluate separately the performance of working sets and extrapolation, we also implement "Celer w/o extr.", *i.e.,* Algorithm 2.4 without using an extrapolated dual point. Doing this, GSR can be compared to GSR + extrapolation, and Celer without extrapolation to Celer.

On Figures 2.5 to 2.7, one can see that using acceleration systematically improves the performance of Gap Safe rules, up to a factor 3. Similarly, dual extrapolation makes Celer more efficient than a working set approach without extrapolation (Blitz or Celer w/o extr.) This improvement is more visible for low values of stopping criterion ϵ, as dual extrapolation is beneficial once the support is identified. Generally, working set approaches tend to perform better on coarse grids, while screening is beneficial on fine grids – a finding corroborating Lasso experiments in Ndiaye et al. (2017b).

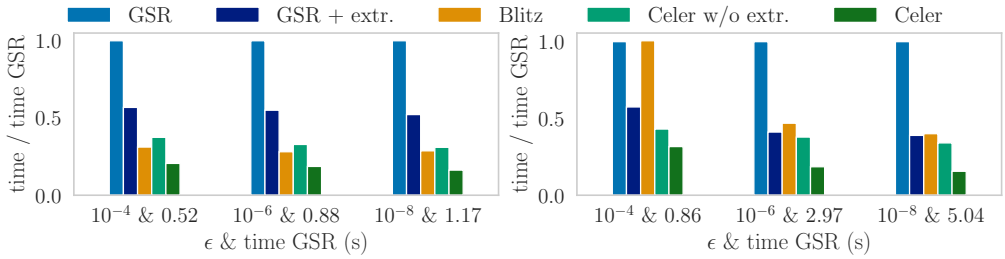


Figure 2.5 – Time to compute a Lasso path from $\lambda_{\max}$ to $\lambda_{\max}/100$ on the *leukemia* dataset (left: coarse grid of 10 values, right: fine grid of 100 values). $\lambda_{\max}/100$ gives a solution with 60 nonzero coefficients.

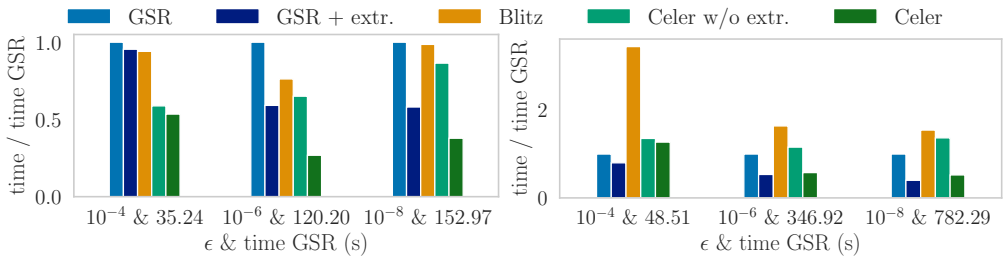


Figure 2.6 – Time to compute a Lasso path from $\lambda_{\max}$ to $\lambda_{\max}/100$ on the *news20* dataset (left: coarse grid of 10 values, right: fine grid of 100 values). $\lambda_{\max}/100$ gives a solution with 14 817 nonzero coefficients.

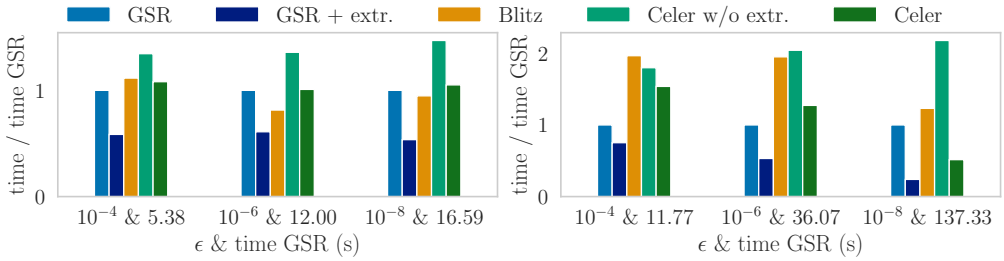


Figure 2.7 – Time to compute a Lasso path from $\lambda_{\max}$ to $\lambda_{\max}/100$ on the *rcv1* dataset (left: coarse grid of 10 values, right: fine grid of 100 values). $\lambda_{\max}/100$ gives a solution with 4610 nonzero coefficients.

Table 2.1 – Computation time (in seconds) for Celer, Blitz and scikit-learn to reach a given precision ϵ on the Finance dataset with $\lambda = \lambda_{\max}/20$ (without warm start: $\beta^{(0)} = \mathbf{0}_p$).

ϵ	10^{-2}	10^{-3}	10^{-4}	10^{-6}
Celer	5	7	8	10
Blitz	25	26	27	30
scikit-learn	470	1350	2390	-

GLMNET comparison Another popular solver for the Lasso is GLMNET, which uses working sets heuristics based on KKT conditions. However, the resulting solutions are not safe in terms of feature identification. Figure 2.8 shows that for the same value of stopping criterion[6], the supports identified by GLMNET contain much more features outside of the equicorrelation set (determined with Gap Safe rules after running Celer with $\epsilon = 10^{-14}$).

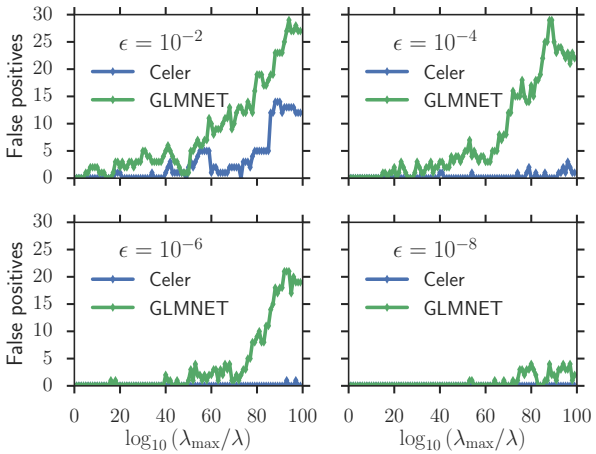


Figure 2.8 – Number of false positives for GLMNET and Celer on a Lasso path on the *leukemia* dataset, depending on the stopping criterion ϵ.

[6] on primal decrease for GLMNET, on duality gap for Celer

Single λ To demonstrate that the performance observed in Figures 2.5 to 2.7 is not only due to the sequential setting, we also perform an experiment for a single value of $\lambda = \lambda_{\max}/20$. The Lasso estimator is computed up to a desired precision ϵ which is varied between 10^{-2} and 10^{-6} (all solvers use duality gap). Celer is orders of magnitude faster than scikit-learn, which uses vanilla coordinate descent. The working set approach of Blitz is also outperformed, especially for low ϵ values.

2.6 Conclusion

The working set approach of Blitz, analogous to that of Celer, is based on a geometric interpretation of the dual. The criterion to build $\mathcal{W}^{(t)}$ can be reformulated to match (2.37) (with the notable difference that $p^{(t)}$ is determined at runtime by solving an auxiliary optimization problem). However, for the analysis to hold, the dual point $\theta^{(t)}$ used in the outer loop must be a barycenter of the previous dual point $\theta^{(t-1)}$ and the current residuals, rescaled on the subproblem $r^{(t-1)}/\max(\lambda, \|X_{:\mathcal{W}_{(t-1)}}^\top r^{(t-1)}\|_\infty)$. This prevents Blitz from using extrapolation. The flexibility of Celer *w.r.t.* the choice of dual point enables it to benefit from the extrapolated dual point returned by the inner solver. The experiments confirm that this dual point is key to outperform Blitz.

In this chapter, we have illustrated the importance of improving duality gap computations for practical Lasso solvers. Using an extrapolation technique to create more accurate dual candidates, we were able to accelerate standard solvers relying on screening and working set techniques. Our experiments on popular (sparse or dense) datasets showed the importance of dedicating some effort to the improvement of dual solutions: the combined benefits obtained both from improved stopping time and from screening accuracy has led to improved state-of-the-art solvers at little coding effort. The goal of the next chapter is to generalize the proposed approach to other datafitting terms and penalties.

3

Duality improvements for sparse GLMs

Il est de retour. Il - est - de - retour.

Contents

3.1 Introduction . . . 66
3.2 GLMs, Vector AutoRegressive sequences and sign identification . . . 67
3.3 Generalized linear models . . . 71
3.3.1 Coordinate descent for ℓ_1 regularization . . . 71
3.3.2 Multitask Lasso . . . 73
3.4 Working sets . . . 74
3.4.1 Improved working sets policy . . . 74
3.4.2 Newton-Celer . . . 75
3.5 Experiments . . . 78
3.5.1 Illustration of dual extrapolation . . . 79
3.5.2 Improved screening and working set policy . . . 80
3.6 Conclusion . . . 82

In this chapter, we generalize the dual extrapolation procedure for the Lasso (Celer) of Chapter 2 to any ℓ_1-regularized GLM, in particular sparse Logistic regression. Theoretical guarantees based on *sign identification* of coordinate descent are provided. Experiments show that dual extrapolation yields more efficient Gap Safe screening rules and working sets solvers. Finally, we adapt Celer to make it compatible with prox-Newton solvers, and empirically demonstrate its applicability to the Multi-task Lasso, for which the proof is left to future work.

This chapter is based on the following work, currently under review for the Journal of Machine Learning Research:

- **M. Massias**, S. Vaiter, A. Gramfort, and J. Salmon. Dual extrapolation for sparse Generalized Linear Models. *arXiv preprint arXiv:1907.05830*, 2019

Generalized Linear Models (GLM) form a wide class of regression and classification models, where prediction is a function of a linear combination of the input variables. For statistical inference in high dimension, sparsity inducing regularizations have proven to be useful while offering statistical guarantees. However, solving the resulting optimization problems can be challenging: even for popular iterative algorithms such as

coordinate descent, one needs to loop over a large number of variables. To mitigate this, techniques known as *screening rules* and *working sets* diminish the size of the optimization problem at hand, either by progressively removing variables, or by solving a growing sequence of smaller problems. For both techniques, significant variables are identified thanks to convex duality arguments. In this paper, we show that the dual iterates of a GLM exhibit an *asymptotic* Vector AutoRegressive (VAR) behavior after sign identification, when the primal problem is solved with proximal gradient descent or cyclic coordinate descent. Exploiting this regularity, one can construct dual points that offer tighter certificates of optimality, enhancing the performance of screening rules and helping to design competitive working set algorithms.

3.1 Introduction

Sparsity inducing penalties have been used in a variety of statistical estimators, both for regression and classification tasks: sparse logistic regression (Koh et al., 2007), Group Lasso (Yuan and Lin, 2006), Sparse Group Lasso (Simon et al., 2013), multitask Lasso (Obozinski et al., 2010), Square-Root Lasso (Belloni et al., 2011). All of these estimators fall under the framework of Generalized Linear Models (McCullagh and Nelder, 1989), where the output is assumed to follow an exponential family distribution whose mean depends on a linear combination of the input variables (see Section 1.1.1). The key property of ℓ_1-type regularization is that it allows to jointly perform feature selection and prediction, which is particularly useful in high dimensional settings. Indeed, it can drastically reduce the number of variables needed for prediction, thus improving model interpretability and computation time for prediction. Amongst the algorithms proposed to solve these, coordinate descent[1] (Tseng, 2001; Friedman et al., 2007) is the most popular in machine learning scenarios (Fan et al., 2008; Friedman et al., 2010; Richtárik and Takáč, 2014; Fercoq and Richtárik, 2015; Perekrestenko et al., 2017; Karimireddy et al., 2018). It consists in updating the vector of parameters one coefficient at a time, looping over all the predictors until convergence.

Since only a fraction of the coefficients are non-zero in the optimal parameter vector, a recurring idea to speed up solvers is to limit the size of the optimization problem by ignoring features which are not included in the solution. To do so, two approaches can be distinguished:

- *screening rules*, introduced by El Ghaoui et al. (2012) and later developed by Ogawa et al. (2013); Wang et al. (2013); Xiang et al. (2016); Bonnefoy et al. (2014); Fercoq et al. (2015); Ndiaye et al. (2017b), progressively remove features from the problems in a backward approach,
- *working sets* techniques (Fan and Lv, 2008; Roth and Fischer, 2008; Kowalski et al., 2011; Tibshirani et al., 2012; Johnson and Guestrin, 2015) solve a sequence of smaller problems restricted to a growing number of features.

One common idea between the current state-of-art methods for screening (Gap Safe rules, Fercoq et al. 2015; Ndiaye et al. 2017b) and working sets (Blitz, Johnson and Guestrin 2015, 2018) is to use a dual point to identify useful features. The quality of such a dual point is critical here as it has a direct impact on performance. However,

[1] throughout the chapter, this means *cyclic and proximal* coordinate descent unless specified otherwise

although a lot of attention has been devoted to creating a sequence of primal iterates that converges fast to the optimum (Fercoq and Richtárik, 2015), the construction of dual iterates has not been scrutinized, and the standard approach to obtain dual iterates from primal ones (Mairal, 2010), although converging, is crude.

In this chapter, we propose a principled way to construct a sequence of dual points that converges faster than the standard approach proposed by Mairal (2010). Based on an extrapolation procedure inspired by Scieur et al. (2016), it comes with no significant extra computational costs, while retaining convergence guarantees of the standard approach. We first introduced this construction for non-smooth optimization in Chapter 2 for the Lasso case only: here, we generalize it here to any Generalized Linear Model (GLM). We properly define, quantify and prove the asymptotic Vector AutoRegressive (VAR) behavior of dual iterates for sparse GLMs solved with proximal gradient descent or cyclic coordinate descent. As for the Lasso, the resulting new construction:

- provides a tighter control of optimality through duality gap evaluation,
- improves the performance of Gap Safe rules,
- improves the aggressive use of Gap Safe screening rules proposed in Chapter 2, thanks to better feature identification,
- is easy to implement and combine with other solvers.

The chapter proceeds as follows. We introduce the framework of ℓ_1-regularized GLMs and duality in Section 3.2. We generalize the techniques of the previous chatper to a variety of problems in Sections 3.3 and 3.4. Results of Section 3.5 demonstrate a systematic improvement in computing time when dual extrapolation is used together with Gap Safe rules or working set policies.

Notation The design matrix $X \in \mathbb{R}^{n \times p}$ is composed of observations $\mathbf{x}_i \in \mathbb{R}^p$ stored row-wise, and whose j-th column is $X_{:j} \in \mathbb{R}^n$; the vector $y \in \mathbb{R}^n$ (*resp.* $\{-1, 1\}^n$) is the response vector for regression (*resp.* binary classification).

The sigmoid function is $\sigma : x \mapsto 1/(1 + e^{-x})$. Applied to vectors, sign, σ and $\mathrm{ST}(\cdot, \nu)$ (for $\nu \in \mathbb{R}_+$) act element-wise.

3.2 GLMs, Vector AutoRegressive sequences and sign identification

We consider the following optimization problem:

Definition 3.1 (Sparse Generalized Linear Model)**.**

$$\hat{\beta} \in \arg\min_{\beta \in \mathbb{R}^p} \underbrace{\sum_{i=1}^{n} f_i(\beta^\top \mathbf{x}_i) + \lambda \|\beta\|_1}_{\mathcal{P}(\beta)} \ , \tag{3.1}$$

where all f_i belong to $\Gamma_0(\mathbb{R})$ (see Definition 1.4), and are differentiable with $1/\gamma$-Lipschitz gradients. The parameter λ is a non-negative scalar, controlling the trade-off between data fidelity and regularization.

Two popular instances of Problem (3.1) are the Lasso ($f_i(t) = \frac{1}{2}(y_i - t)^2$, $\gamma = 1$) and Sparse Logistic regression ($f_i(t) = \log(1 + \exp(-y_i t))$, $\gamma = 4$).

Note that Problem (3.1) could be called "ℓ_1-regularized ERM" rather than GLM, since all instances of ERM do not come from the Maximum Likelihood Estimator of a GLM (see Problem (1.11) in Section 1.1.1). This is a misuse of language, originating from our focus on Lasso and Sparse Logistic regression.

We could use a more complex regularizer in Problem (3.1), to handle group penalties for example. For the sake of clarity we rather remain specific, and generalize to other penalties when needed in Section 3.3.2.

Proposition 3.2 (Strong duality for sparse GLMs)**.** *A dual formulation of Problem (3.1) reads:*

$$\hat{\theta} = \underset{\theta \in \Delta_X}{\arg\max} \underbrace{\left(- \sum_{i=1}^{n} f_i^*(-\lambda \theta_i) \right)}_{\mathcal{D}(\theta)} , \tag{3.2}$$

where $\Delta_X = \{\theta \in \mathbb{R}^n : \|X^\top \theta\|_\infty \leq 1\}$*. The dual solution* $\hat{\theta}$ *is unique because the* f_i^**'s are* γ*-strongly convex (see Proposition 1.9) and the KKT conditions read:*

$$\forall i \in [n], \qquad \hat{\theta}_i = -f_i'(\hat{\beta}^\top \mathbf{x}_i)/\lambda \qquad \textbf{\textit{(link equation)}} \tag{3.3}$$

$$\forall j \in [p], \qquad X_{:j}^\top \hat{\theta} \in \partial |\cdot|(\hat{\beta}_j) \qquad \textbf{\textit{(subdifferential inclusion)}} \tag{3.4}$$

If for $u \in \mathbb{R}^n$ *we write* $F(u) \triangleq \sum_{i=1}^n f_i(u_i)$*, the link equation reads* $\hat{\theta} = -\nabla F(X\hat{\beta})/\lambda$*.*

For any $(\beta, \theta) \in \mathbb{R}^p \times \Delta_X$*, one has* $\mathcal{D}(\theta) \leq \mathcal{P}(\beta)$*, and* $\mathcal{D}(\hat{\theta}) = \mathcal{P}(\hat{\beta})$*. The duality gap* $\mathcal{P}(\beta) - \mathcal{D}(\theta)$ *can thus be used as an upper bound for the sub-optimality of a primal vector* β*: for any* $\epsilon > 0$*, any* $\beta \in \mathbb{R}^p$*, and any feasible* $\theta \in \Delta_X$*:*

$$\mathcal{P}(\beta) - \mathcal{D}(\theta) \leq \epsilon \Rightarrow \mathcal{P}(\beta) - \mathcal{P}(\hat{\beta}) \leq \epsilon \ . \tag{3.5}$$

These results hold because Problem (3.1) is unconstrained; and the convex objective function has domain $\mathbb{R}^p$ *(Boyd and Vandenberghe, 2004, §5.2.3); see Proposition 1.14 for the general framework of duality.*

Remark 3.3. *Equation (3.5) shows that even though* $\hat{\beta}$ *is unknown in practice and the sub-optimality gap cannot be evaluated, creating a dual feasible point* $\theta \in \Delta_X$ *allows to construct an upper bound which can be used as a tractable stopping criterion.*

In high dimension, solvers such as proximal gradient descent (PG) and coordinate descent (CD) are slowed down due to the large number of features. However, by design of the ℓ_1 penalty, $\hat{\beta}$ is expected to be sparse, especially for large values of λ. Thus, a key idea to speed up these solvers is to identify the support of $\hat{\beta}$ so that features outside of it can be safely ignored. Doing this leads to a smaller problem that is faster to solve. Removing features when it is guaranteed that they are not in the support of the solution is at the heart of the so-called *Gap Safe Screening rules* (Fercoq et al., 2015; Ndiaye et al., 2017b).

Proposition 3.4 (Gap Safe Screening rule, (Ndiaye et al., 2017b, Thm. 6))**.** *The Gap Safe screening rule for Problem (3.1) reads:*

$$\forall j \in [p], \forall (\beta, \theta) \in \mathbb{R}^p \times \Delta_X, \ |X_{:j}^\top \theta| < 1 - \|X_{:j}\| \sqrt{\tfrac{2}{\gamma \lambda^2}(\mathcal{P}(\beta) - \mathcal{D}(\theta))} \implies \hat{\beta}_j = 0 \ . \tag{3.6}$$

Therefore, while running an iterative solver and computing the duality gap at iteration t, the criterion (3.6) can be tested for all features j, and the features guaranteed to be inactive at optimum can be ignored in the subsequent iterations.

Equations (3.5) and (3.6) do not require a specific choice of θ, provided it is in Δ_X. It is up to the user and so far it has not attracted much attention in the literature. Thanks to the link equation $\hat{\theta} = -\nabla F(X\hat{\beta})/\lambda$, a natural way to construct a dual feasible point $\theta^{(t)} \in \Delta_X$ at iteration t, when only a primal vector $\beta^{(t)}$ is available, is:

$$\theta^{(t)}_{\text{res}} \triangleq -\nabla F(X\beta^{(t)}) / \max(\lambda, \|X^\top \nabla F(X\beta^{(t)})\|_\infty) \ . \tag{3.7}$$

As detailed in Chapter 2, this was coined *residuals rescaling* (Mairal, 2010) following the terminology used for the Lasso case where $-\nabla F(X\beta)$ is equal to the residuals, $y - X\beta$.

To improve the control of sub-optimality, and to better identify useful features, the aim of our proposed *dual extrapolation* is to obtain a better dual point (closer to the optimum $\hat{\theta}$). The idea is to do it at a low computational cost by exploiting the structure of the sequence of dual iterates $(X\beta^{(t)})_{t\in\mathbb{N}}$ (and not the residuals as for the Lasso). Since the gradient of F is not linear in general, the structure is not exactly a VAR as in Definition 2.3.

Definition 3.5 (Asymptotic Vector AutoRegressive sequence)**.** *We say that the sequence $(r^{(t)})_{t\in\mathbb{N}}$, converging to $\hat{r}$, is an* asymptotic *VAR sequence if there exist $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ such that for $t \in \mathbb{N}$:*

$$r^{(t+1)} - Ar^{(t)} - b = o(r^{(t)} - \hat{r}) \ . \tag{3.8}$$

Finally, we state the results on sign identification, which implies support identification. For these results, which connect sparse GLMs to VAR sequences and extrapolation, we need to make the following assumption.

Assumption 3.6. *The solution of Problem (3.1) is unique.*

Assumption 3.6 may seem stringent, as whenever $p > n$ the loss function $\mathcal{P}$ is not strictly convex and several global minima may exist. However, following earlier results by Rosset et al. (2004), Tibshirani (2013) showed that when the entries of X are sampled from a continuous distribution, Assumption 3.6 is satisfied almost surely. It is also worth noting that other works on support identification (Nutini et al., 2017; Sun et al., 2019; Poon et al., 2018) involve a non-degeneracy condition which boils down to Assumption 3.6 in our case. Motivated by practical applications on real-life datasets, we will therefore use Assumption 3.6.

In the following, we extend results by Hale et al. (2008) about sign identification from proximal gradient to coordinate descent.

Theorem 3.7 (Sign identification for proximal gradient and coordinate descent)**.** *Let Assumption 3.6 hold. Let $(\beta^{(t)})_{t\in\mathbb{N}}$ be the sequence of iterates converging to $\hat{\beta}$ and produced by proximal gradient descent or coordinate descent when solving Problem (3.1) (reminded in lines 10 and 13 of Algorithm 3.1).*

There exists $T \in \mathbb{N}$ such that: $\forall j \in [p], t \geq T \implies \text{sign}(\beta_j^{(t)}) = \text{sign}(\hat{\beta}_j)$. The smallest epoch T for which this holds is when sign identification *is achieved.*

Proof For lighter notation in this proof, we denote $l_j = \|X_{:j}\|^2/\gamma$ and $h_j(\beta) = \beta_j - \frac{1}{l_j} X_{:j}^\top \nabla F(X\beta)$. For $j \in [p]$, the subdifferential inclusion (3.4) reads:

$$-\frac{X_{:j}^\top \nabla F(X\hat{\beta})}{\lambda} \in \begin{cases} \{1\} \ , & \text{if } \hat{\beta}_j > 0 \ , \\ \{-1\} \ , & \text{if } \hat{\beta}_j < 0 \ , \\ [-1, 1] \ , & \text{if } \hat{\beta}_j = 0 \ . \end{cases} \tag{3.9}$$

Motivated by these conditions, the *equicorrelation set* introduced by Tibshirani (2013) is:

$$E \triangleq \{j \in [p] \ : \ |X_{:j}^\top \nabla F(X\hat{\beta})| = \lambda\} = \{j \in [p] \ : \ |X_{:j}^\top \hat{\theta}| = 1\} \ . \tag{3.10}$$

We introduce the *saturation gap* associated to Problem (3.1):

$$\hat{\delta} \triangleq \min \left\{ \frac{\lambda}{l_j} \left(1 - \frac{|X_{:j}^\top \nabla F(X\hat{\beta})|}{\lambda} \right) : j \notin E \right\} = \min \left\{ \frac{\lambda}{l_j} \left(1 - |X_{:j}^\top \hat{\theta}| \right) : j \notin E \right\} > 0 \ . \tag{3.11}$$

As $\hat{\theta}$ is unique, $\hat{\delta}$ is well-defined, and strictly positive by definition of E. By Equation (3.9), the support of any solution is included in the equicorrelation set and we have equality for almost every λ since we assumed that the solution is unique (Tibshirani, 2013, Lemma 13) – the non equality holding only at the kinks of the Lasso path, which we exclude from our analysis.

Because of Assumption 3.6, we only need to show that the coefficients outside the equicorrelation eventually vanish. The proof requires to study the primal iterates after each update (instead of after each epoch), hence we use the notation $\tilde{\beta}^{(s)}$ for the primal iterate after the s-th update of coordinate descent. This update only modifies the j-th coordinate, with $s \equiv j - 1 \mod p$:

$$\tilde{\beta}_j^{(s+1)} = \mathrm{ST}\left(h_j(\tilde{\beta}^{(s)}), \tfrac{\lambda}{l_j} \right) \ . \tag{3.12}$$

Note that at optimality, for every $j \in [p]$, one has:

$$\hat{\beta}_j = \mathrm{ST}\left(h_j(\hat{\beta}), \tfrac{\lambda}{l_j} \right) \ . \tag{3.13}$$

Let us consider an update $s \in \mathbb{N}$ of coordinate descent such that the updated coordinate j verifies $\tilde{\beta}_j^{(s+1)} \neq 0$ and $j \notin E$, hence, $\hat{\beta}_j = 0$. Then:

$$\begin{aligned} |\tilde{\beta}_j^{(s+1)} - \hat{\beta}_j| &= \left| \mathrm{ST}\left(h_j(\tilde{\beta}^{(s)}), \tfrac{\lambda}{l_j} \right) - \mathrm{ST}\left(h_j(\hat{\beta}), \tfrac{\lambda}{l_j} \right) \right| \\ &\leq \left| h_j(\tilde{\beta}^{(s)}) - h_j(\hat{\beta}) \right| - \left(\tfrac{\lambda}{l_j} - |h_j(\hat{\beta})| \right) \ , \end{aligned} \tag{3.14}$$

where we used the following inequality (Hale et al., 2008, Lemma 3.2):

$$\mathrm{ST}(x, \nu) \neq 0, \mathrm{ST}(y, \nu) = 0 \implies |\mathrm{ST}(x, \nu) - \mathrm{ST}(y, \nu)| \leq |x - y| - (\nu - |y|) \ . \tag{3.15}$$

Now notice that by definition of the saturation gap (3.11), and since $j \notin E$:

$$\begin{aligned} &\frac{\lambda}{l_j} \left(1 - \frac{|X_{:j}^\top \nabla F(X\hat{\beta})|}{\lambda} \right) \geq \hat{\delta} \ , \\ \text{that is,} \quad &\frac{\lambda}{l_j} - |h_j(\hat{\beta})| \geq \hat{\delta} \quad (\text{using } \hat{\beta}_j = 0) \ . \end{aligned} \tag{3.16}$$

Algorithm 3.1 PG/CYCLIC CD FOR PROBLEM (3.1) WITH DUAL EXTRAPOLATION

input : $X = [x_1|\dots|x_p], y, \lambda, \beta^{(0)}, \epsilon$
param: $T, K = 5, f^{\text{dual}} = 10$
init : $X\beta = X\beta^{(0)}, \theta^{(0)} = -\nabla F(X\beta^{(0)}) / \max(\lambda, \|X^\top \nabla F(X\beta^{(0)})\|_\infty)$
for $t = 1, \dots, T$ **do**
  if $t = 0 \mod f^{\text{dual}}$ **then** // compute θ and gap every f epoch only
    $t' = t / f^{\text{dual}}$ // dual point indexing
    $r^{(t')} = X\beta$
    compute $\theta^{(t')}_{\text{res}}$ and $\theta^{(t')}_{\text{accel}}$ with Equations (2.25), (2.26) and (3.7)
    $\theta^{(t')} = \arg\max \left\{ \mathcal{D}(\theta) \,:\, \theta \in \{\theta^{(t'-1)}, \theta^{(t')}_{\text{accel}}, \theta^{(t')}_{\text{res}}\} \right\}$ // robust dual extr. with (2.27)
    if $\mathcal{P}(\beta^{(t)}) - \mathcal{D}(\theta^{(t')}) < \epsilon$ **then** break
  if PG **then** // proximal gradient descent:
    $X\beta = X\beta^{(t)}$
    $\beta^{(t+1)} = \text{ST}\left(\beta^{(t)} - \frac{\gamma}{\|X^\top X\|_2} X^\top \nabla F(X\beta), \frac{\lambda\gamma}{\|X^\top X\|_2}\right)$
  else if CD **then** // cyclic coordinate descent:
    for $j = 1, \dots, p$ **do**
      $\beta_j^{(t+1)} = \text{ST}\left(\beta_j^{(t)} - \frac{\gamma X_{:j}^\top \nabla F(X\beta)}{\|X_{:j}\|^2}), \frac{\gamma\lambda}{\|X_{:j}\|^2}\right)$
      $X\beta \mathrel{+}= (\beta_j^{(t+1)} - \beta_j^{(t)}) X_{:j}$
return $\beta^{(t)}$, $\theta^{(t')}$

Combining Equations (3.14) and (3.16) yields

$$|\tilde{\beta}_j^{(s+1)} - \hat{\beta}_j| \leq |h_j(\tilde{\beta}^{(s)}) - h_j(\hat{\beta})| - \hat{\delta} \ . \tag{3.17}$$

This can only be true for a finite number of updates, since otherwise taking the limit would give $0 \leq -\hat{\delta}$, and $\hat{\delta} > 0$ (Equation (3.11)). Therefore, after a finite number of updates, $\tilde{\beta}_j^{(s)} = 0$ for $j \notin E$.

For $j \in E$, $\hat{\beta}_j \neq 0$ by Assumption 3.6, so $\beta_j^{(t)}$ has the same sign eventually since it converges to $\hat{\beta}_j$.

The proof for proximal gradient descent is a result of Hale et al. (2008, Theorem 4.5), who provide the bound $T \leq \|\tilde{\beta}^{(s)} - \hat{\beta}\|_2^2 / \hat{\delta}^2$. ■

Equipped with these definitions, we can highlight and exploit the regularity of the dual iterates.

3.3 Generalized linear models

3.3.1 Coordinate descent for ℓ_1 regularization

Theorem 3.8 (VAR for coordinate descent and Sparse GLM)**.** *When Problem (3.1) is solved by cyclic coordinate descent, the dual iterates $(X\beta^{(t)})_{t\in\mathbb{N}}$ form an asymptotical VAR sequence.*

Proof As in the proof of Theorem 2.6, we place ourselves in the identified sign regime, and consider only one epoch t of CD: let $\tilde{\beta}^{(0)}$ denote the value of the primal iterate at the beginning of the epoch ($\tilde{\beta}^{(0)} = \beta^{(t)}$), and for $s \in [S]$, $\tilde{\beta}^{(s)} \in \mathbb{R}^p$ denotes its value after the j_s coordinate has been updated ($\tilde{\beta}^{(S)} = \beta^{(t+1)}$). Recall that in the framework of Problem (3.1), the data-fitting functions f_i have $1/\gamma$-Lipschitz gradients, and $\nabla F(u) = (f_1'(u_1), \dots, f_n'(u_n))$.

For $s \in [S]$, $\tilde{\beta}^{(s)}$ and $\tilde{\beta}^{(s-1)}$ are equal everywhere except at entry j_s, for which the coordinate descent update with fixed step size $\frac{\gamma}{\|X_{:j_s}\|^2}$ is:

$$\begin{aligned}\tilde{\beta}^{(s)}_{j_s} &= \mathrm{ST}\left(\tilde{\beta}^{(s-1)}_{j_s} - \tfrac{\gamma}{\|X_{:j_s}\|^2} X_{:j_s}^\top \nabla F(X\tilde{\beta}^{(s-1)}), \tfrac{\gamma}{\|X_{:j_s}\|^2}\lambda\right) \\ &= \tilde{\beta}^{(s-1)}_{j_s} - \tfrac{\gamma}{\|X_{:j_s}\|^2} X_{:j_s}^\top \nabla F(X\tilde{\beta}^{(s-1)}) - \tfrac{\gamma}{\|X_{:j_s}\|^2}\lambda \operatorname{sign}(\hat{\beta}_{j_s}) \ . \end{aligned} \tag{3.18}$$

Therefore,

$$\begin{aligned} X\tilde{\beta}^{(s)} - X\tilde{\beta}^{(s-1)} &= X_{:j_s}\left(\tilde{\beta}^{(s)}_{j_s} - \tilde{\beta}^{(s-1)}_{j_s}\right) \\ &= X_{:j_s}\left(-\tfrac{\gamma}{\|X_{:j_s}\|^2} X_{:j_s}^\top \nabla F(X\tilde{\beta}^{(s-1)}) - \tfrac{\gamma}{\|X_{:j_s}\|^2}\lambda \operatorname{sign}(\hat{\beta}_{j_s})\right) \ . \end{aligned} \tag{3.19}$$

Using point-wise linearization of the function ∇F around $X\hat{\beta}$, we have:

$$\nabla F(X\beta) = \nabla F(X\hat{\beta}) + D(X\beta - X\hat{\beta}) + o(X\beta - X\hat{\beta}) \ , \tag{3.20}$$

where $D \triangleq \operatorname{diag}(f_1''(\hat{\beta}^\top \mathbf{x}_1), \dots, f_n''(\hat{\beta}^\top \mathbf{x}_n)) \in \mathbb{R}^{n\times n}$. Therefore:

$$\begin{aligned} X\tilde{\beta}^{(s)} &= \left(\mathrm{Id}_n - \tfrac{\gamma}{\|X_{:j_s}\|^2} X_{:j_s} X_{:j_s}^\top D\right) X\tilde{\beta}^{(s-1)} \\ &\quad + \tfrac{\gamma}{\|X_{:j_s}\|^2}\left(X_{:j_s}^\top (DX\hat{\beta} - \nabla F(X\hat{\beta})) - \lambda \operatorname{sign}(\hat{\beta}_{j_s})\right) X_{:j_s} + o(X\tilde{\beta}^{(s)} - X\hat{\beta}) \ . \\ D^{1/2}X\tilde{\beta}^{(s)} &= \underbrace{\left(\mathrm{Id}_n - \tfrac{\gamma}{\|X_{:j_s}\|^2} D^{1/2} X_{:j_s} X_{:j_s}^\top D^{1/2}\right)}_{A_s} D^{1/2} X\tilde{\beta}^{(s-1)} \\ &\quad + \underbrace{\tfrac{\gamma}{\|X_{:j_s}\|^2} X_{:j_s}^\top (DX\hat{\beta}) D^{1/2} X_{:j_s}}_{b_s} + o(X\tilde{\beta}^{(s)} - X\hat{\beta}) \ , \end{aligned} \tag{3.21}$$

since the subdifferential inclusion (3.4) gives $-X_{:j_s}^\top \nabla F(X\hat{\beta}) - \lambda \operatorname{sign}(\hat{\beta}_{j_s}) = 0$. Thus $(D^{1/2}X\beta^{(t)})_{t\in\mathbb{N}}$ is an asymptotical VAR sequence:

$$D^{1/2}X\beta^{(t+1)} = A_S \dots A_1 D^{1/2} X\beta^{(t)} + b_S + \dots + A_S \dots A_2 b_1 + o(X\beta^{(t)} - X\hat{\beta}) \ , \tag{3.22}$$

and so is $(X\beta^{(t)})_{t\in\mathbb{N}}$:

$$X\beta^{(t+1)} = \underbrace{D^{-\frac{1}{2}} A_S \dots A_1 D^{\frac{1}{2}}}_{A} X\beta^{(t)} + \underbrace{D^{-\frac{1}{2}}(b_S + \dots + A_S \dots A_2 b_1)}_{b} + o(X\beta^{(t)} - X\hat{\beta}) \ . \tag{3.23}$$

■

Proposition 3.9. *As in Lemmas 2.9 and 2.10, for the VAR parameters A and b defined in Equation (3.23), 1 is the only eigenvalue of A whose modulus is 1 and $b \perp \operatorname{Ker}(\mathrm{Id}_n - A)$.*

Proof First, notice that as in the Lasso case, we have $\mathrm{Id}_n \succeq A_s \succeq 0$. Indeed, because f_i'' takes values in $]0, 1/\gamma[$, $D^{1/2}$ exists and $\frac{1}{\sqrt{\gamma}}\mathrm{Id}_n \succeq D^{1/2} \succeq 0$. For any $u \in \mathbb{R}^n$,

$$u^\top D^{1/2} X_{:j_s} X_{:j_s}^\top D^{1/2} u = (X_{:j_s}^\top D^{1/2} u)^2 \geq 0, \tag{3.24}$$

$$\begin{aligned} \text{and} \qquad X_{:j_s}^\top D^{1/2} u &\leq \|X_{:j_s}\| \|D^{1/2} u\| \\ &\leq \|X_{:j_s}\| \|D^{1/2}\| \|u\| \\ &\leq \tfrac{1}{\sqrt{\gamma}} \|X_{:j_s}\| \|u\| \ , \end{aligned} \tag{3.25}$$

thus $\frac{\|X_{:j_s}\|^2}{\gamma}\mathrm{Id}_n \succeq D^{1/2} X_{:j_s} X_{:j_s}^\top D^{1/2} \succeq 0$ and $\mathrm{Id}_n \succeq A_s \succeq 0$.

However, contrary to the Lasso case, because $\|D^{1/2} X_{:j_s}\| \neq \sqrt{\gamma}\|X_{:j_s}\|$, A_s is not the orthogonal projection on $(\operatorname{Span} D^{1/2} X_{:j_s})^\perp$. Nevertheless, we still have $A_s = A_s^\top$, $\|A_s\| \leq 1$, and for $v \in \mathbb{R}^n$, $A_s v = v$ means that $v^\top D^{1/2} X_{:j_s} = 0$, so the proof of Lemma 2.9 can be applied to show that the only eigenvalue of $A_S \dots A_1$ which has modulus 1 is 1. Then, observing that $A = D^{-1/2} A_S \dots A_1 D^{1/2}$ has the same spectrum as $A_S \dots A_1$ concludes the first part of the proof.

For the second result, let $v \in \operatorname{Ker}(\mathrm{Id}_n - A)$, *i.e.*, $Av = v$, hence $A_S \dots A_1 D^{1/2} v = D^{1/2} A v = D^{1/2} v$. Therefore $D^{1/2} v$ is a fixed point of $A_S \dots A_1$, and as in the Lasso case this means that for all $s \in [S]$, $A_s D^{1/2} v = D^{1/2} v$ and $(D^{1/2} v)^\top D^{1/2} X_{:j_s} = 0$. Now recall that

$$b = D^{-1/2}(b_S + \dots + A_S \dots A_2 b_1) \ , \tag{3.26}$$

$$\begin{aligned} b_s &= \tfrac{\gamma}{\|X_{:j_s}\|^2} \left(X_{:j_s}^\top (DX\hat{\beta} - \nabla F(X\hat{\beta})) - \lambda \operatorname{sign}(\hat{\beta}_{j_s}) \right) D^{1/2} X_{:j_s} \\ &= \tfrac{\gamma}{\|X_{:j_s}\|^2} (X_{:j_s}^\top D X \hat{\beta}) D^{1/2} X_{:j_s} \ . \end{aligned} \tag{3.27}$$

Additionally, $v^\top D^{-1/2} A_S \dots A_{s+1} b_s = (A_{s+1} \dots A_S D^{-1/2} v)^\top b_s = (D^{-1/2} v)^\top b_s = 0$. Hence v is orthogonal to all the terms which compose b, hence $v^\top b = 0$. ∎

Theorem 3.8 and Proposition 3.9 show that we can construct an extrapolated dual point for any sparse GLM, by extrapolating the sequence $(r^{(t)} = X\beta^{(t)})_{t \in \mathbb{N}}$ with the construction of Equation (2.25), and creating a feasible point with:

$$\theta_{\text{accel}}^{(t)} \triangleq -\nabla F(r_{\text{accel}}^{(t)}) / \max(\lambda, \|X^\top \nabla F(r_{\text{accel}}^{(t)})\|_\infty) \ . \tag{3.28}$$

3.3.2 Multitask Lasso

Let $q \in \mathbb{N}$ be a number of tasks, and consider an observation matrix $Y \in \mathbb{R}^{n \times q}$, whose i-th row is the target in $\mathbb{R}^q$ for the i-th sample.

Definition 3.10. *The multitask Lasso estimator is defined as the solution of:*

$$\hat{\mathrm{B}} \in \operatorname*{arg\,min}_{\mathrm{B} \in \mathbb{R}^{n \times q}} \frac{1}{2} \left\| Y - X\mathrm{B} \right\|_F^2 + \lambda \left\| \mathrm{B} \right\|_{2,1} \ . \tag{3.29}$$

Let $j_1 < \cdots < j_S$ denote the (row-wise) support of $\hat{\mathrm{B}}$, and let t denote an iteration after support identification. Note that the guarantees of support identification for multitask Lasso requires more assumptions than the case of the standard Lasso. In particular it requires a source condition which depends on the design matrix X. This was investigated for instance by Vaiter et al. (2018) when considering a proximal gradient descent algorithm.

Let $\mathrm{B}^{(0)} = \mathrm{B}^{(t)}$, and for $s \in [S]$, let $\mathrm{B}^{(s)}$ denote the primal iterate after coordinate j_s has been updated. Let $s \in [S]$, with $\mathrm{B}^{(s)}$ and $\mathrm{B}^{(s-1)}$ being equal everywhere, except for their j_s row for which one iteration of proximal block coordinate descent gives $\phi(\mathrm{B}) \triangleq \mathrm{B}_{j_s} + \frac{1}{\|X_{:j_s}\|^2} X_{:j_s}^\top (Y - X\mathrm{B}) \in \mathbb{R}^{1\times q}$:

$$\mathrm{B}^{(s)}_{j_s:} = \left(1 - \frac{\lambda/\|X_{:j_s}\|^2}{\|\phi(\mathrm{B}^{(s-1)})\|}\right)\phi(\mathrm{B}^{(s-1)}) \ . \tag{3.30}$$

Hence

$$\begin{aligned} X\mathrm{B}^{(s)} - X\mathrm{B}^{(s-1)} &= X_{:j_s}(\mathrm{B}^{(s)}_{j_s:} - \mathrm{B}^{(s-1)}_{j_s:}) \\ &= X_{:j_s}\left(\frac{1}{\|X_{:j_s}\|^2} X_{:j_s}^\top (Y - X\mathrm{B}^{(s-1)}) - \frac{\lambda/\|X_{:j_s}\|^2}{\|\phi(\mathrm{B}^{(s-1)})\|}\phi(\mathrm{B}^{(s-1)})\right) \ . \end{aligned} \tag{3.31}$$

We are unable to show that $(X\beta^{(t)})_{t\in\mathbb{N}}$ is an asymptotic VAR sequence, because the term $\phi(\mathrm{B})/\|\phi(\mathrm{B})\|$ cannot be linearized with respect to $X\mathrm{B}$ directly. Introducing $\Psi \triangleq e_{j_s}^\top - \frac{1}{\|X_{:j_s}\|^2} X_{:j_s}^\top X$, so that $\phi(\mathrm{B}) = \phi(\hat{\mathrm{B}}) + \Psi(\mathrm{B} - \hat{\mathrm{B}})$, one has the following linearization though:

$$\frac{\phi(\mathrm{B})}{\|\phi(\mathrm{B})\|} = \frac{\phi(\hat{\mathrm{B}})}{\|\phi(\hat{\mathrm{B}})\|} + \frac{1}{\|\phi(\hat{\mathrm{B}})\|}\left(\Psi(\mathrm{B} - \hat{\mathrm{B}}) - \frac{\Psi(\mathrm{B} - \hat{\mathrm{B}})\phi(\hat{\mathrm{B}})^\top \phi(\hat{\mathrm{B}})}{\|\phi(\hat{\mathrm{B}})\|^2}\right) + o(\mathrm{B} - \hat{\mathrm{B}}) \ , \tag{3.32}$$

which does not allow to exhibit a VAR structure, as B should appear only on the right.

Despite the latter negative result, empirical results of Section 3.5 show that dual extrapolation still provides a tighter dual point in the identified support regime. Celer empirical adaptation to multitask Lasso consists in using $d_j^{(t)} = (1 - \|X_{:j}^\top \Theta^{(t)}\|)/\|X_{:j}\|$ with the dual iterate $\Theta^{(t)} \in \mathbb{R}^{n\times q}$. The inner solver is cyclic block coordinate descent (BCD), and the extrapolation coefficients are obtained by solving Equation (2.14), which is an easy to solve matrix least-squares problem.

3.4 Working sets

Being able to construct a better dual point leads to a tighter gap and a smaller upper bound in Equation (3.6), hence to more features being discarded and a greater speed-up for Gap Safe screening rules. As we detail in this section, it also helps to better prioritize features, and to design an efficient working set policy.

3.4.1 Improved working sets policy

In the context of this manuscript, a working set approach starts by solving Problem (3.1) restricted to a small set of features $\mathcal{W}^{(0)} \subset [p]$ (the working set), then defines iteratively new working sets $\mathcal{W}^{(t)}$ and solves a sequence of growing problems (Kowalski et al., 2011;

Boisbunon et al., 2014; Santis et al., 2016). It is easy to see that when $\mathcal{W}^{(t)} \subsetneq \mathcal{W}^{(t+1)}$ and when the subproblems are solved up to the precision required for the whole problem, then working sets techniques converge.

It is easy to see that every screening rule which writes:

$$\forall j \in [p], \quad d_j > \tau \Rightarrow \hat{\beta}_j = 0 \ , \tag{3.33}$$

allows to define a working set policy. For example for Gap Safe rules,

$$d_j = d_j(\theta) \triangleq \frac{1 - |X_{:j}^\top \theta|}{\|X_{:j}\|} \ , \tag{3.34}$$

is defined as a function of a dual point $\theta \in \Delta_X$. The value d_j can be seen as measuring the importance of feature j, and so given an initial size $p^{(1)}$ the first working set can be defined as:

$$\mathcal{W}^{(1)} = \{j_1, \dots, j_{p^{(1)}}\} \ , \tag{3.35}$$

with $d_{j_1}(\theta) \leq \dots \leq d_{j_{p^{(1)}}}(\theta) < d_j(\theta), \forall j \notin \mathcal{W}^{(0)}$, *i.e.,* the indices of the $p^{(1)}$ smallest values of $d(\theta)$. As in Figure 1.7, this is an abuse of notation, as j_1 should be $j_1^{(1)}$ to highlight the dependency on the iteration. The latter notation unfortunately leads to the rather inelegant notation $d_{j_{p^{(1)}}^{(1)}}$, therefore we take the liberty of omitting the exponent notation in $j_1, \dots, j_{p^{(1)}}$. The reader should keep in mind that j_1, etc. are overwritten at each iteration.

Once the working set has been defined, the *subproblem solver* is launched on $X_{\mathcal{W}^{(1)}}$. New primal and dual iterates are returned, which allow to recompute d_j's and define iteratively:

$$\mathcal{W}^{(t+1)} = \{j_1, \dots, j_{p^{(t+1)}}\} \ , \tag{3.36}$$

where we impose $d_j(\theta) = -1$ when $\beta_j^{(t)} \neq 0$ to keep the active features in the next working set. As in Chapter 2, we choose $p^{(t)} = \min(p, 2\|\beta^{(t)}\|_0)$ to ensure a fast initial growth of the working set, and avoid growing too much when the support is nearly identified. The stopping criterion for the inner solver on $\mathcal{W}^{(t)}$ is to reach a gap lower than a fraction $\rho = 0.3$ of the duality gap for the whole problem, $\mathcal{P}(\beta^{(t)}) - \mathcal{D}(\theta^{(t)})$. These adaptive working set policies are commonly used in practice (Johnson and Guestrin, 2015, 2018).

The results of Section 3.3 justify the use of dual extrapolation for any sparse GLM, thus enabling us to generalize Celer to the whole class of models (Algorithm 3.2).

3.4.2 Newton-Celer

When using a squared ℓ_2 loss, the curvature of the loss is constant: for the Lasso and multitask Lasso, the Hessian does not depend on the current iterate. This is however not true for other GLM data fitting terms, *e.g.,* Logistic regression, for which taking into account the second order information proves to be very useful for fast convergence (Hsieh et al., 2014). To leverage this information, we can use a prox-Newton method (Lee et al., 2012; Scheinberg and Tang, 2013) as inner solver; an advantage of dual extrapolation is that it can be combined with *any* inner solver, as we detail below. For reproducibility and completeness, we first briefly detail the Prox-Newton procedure used. In the following and in Algorithms 3.4 to 3.6 we focus on a single subproblem

Algorithm 3.2 Celer for Problem (3.1)

input : $X, y, \lambda, \beta^{(0)}, \theta^{(0)}$
param: $K = 5, p^{(1)} = 100, \epsilon$, `MAX_WS`
init : $\mathcal{W}^{(0)} = \varnothing$

if $\beta^{(0)} \neq \mathbf{0}_p$ **then** $p^{(1)} = |\operatorname{supp}(\beta^{(0)})|$ `// warm start`
for $t = 1, \dots,$ `MAX_WS` **do**
  compute $\theta^{(t)}_{\text{res}}$ `// Equation (3.7)`
  if solver is Prox-Celer **then**
    do K passes of CD on the support of $\beta^{(t)}$, extrapolate to produce $\theta^{(t-1)}_{\text{accel}}$
    $\theta^{(t-1)}_{\text{inner}} = \arg\max_{\theta \in \{\theta^{(t-1)}, \theta^{(t-1)}_{\text{inner}}\}} \mathcal{D}(\theta)$
  $\theta^{(t)} = \arg\max_{\theta \in \{\theta^{(t-1)}, \theta^{(t-1)}_{\text{inner}}, \theta^{(t)}_{\text{res}}\}} \mathcal{D}(\theta)$
  $g^{(t)} = \mathcal{P}(\beta^{(t-1)}) - \mathcal{D}(\theta^{(t)})$ `// global gap`
  if $g^{(t)} \leq \epsilon$ **then** break
  $\epsilon^{(t)}, \mathcal{W}^{(t)} =$ `create_WS()` `// get tolerance and working set with Algorithm 3.3`
  `// Subproblem solver is Algorithm 3.1 or 3.4 for Prox-Celer:`
  get $\tilde{\beta}^{(t)}, \theta^{(t)}_{\text{inner}}$ with subproblem solver applied to $(X_{\mathcal{W}^{(t)}}, y, \lambda, (\beta^{(t-1)})_{\mathcal{W}^{(t)}}, \epsilon^{(t)})$
  $\theta^{(t)}_{\text{inner}} = \theta^{(t)}_{\text{inner}} / \max(1, \|X^\top \theta^{(t)}_{\text{inner}}\|_\infty)$
  set $\beta^{(t)} = \mathbf{0}_p$ and $(\beta^{(t)})_{\mathcal{W}^{(t)}} = \tilde{\beta}^{(t)}$
return $\beta^{(t)}, \theta^{(t)}$

Algorithm 3.3 `create_WS`

input : $X, y, \lambda, \beta^{(t-1)}, \theta^{(t)}, \mathcal{W}^{(t-1)}, g^{(t)}$
param: $p^{(1)} = 100, \rho = 0.3$
init : $d = \mathbf{0}_p$

for $j = 1, \dots, p$ **do**
  if $\beta^{(t-1)}_j \neq 0$ **then** $d^{(t)}_j = -1$
  else $d^{(t)}_j = (1 - |X_{:j}^\top \theta^{(t)}|)/\|X_{:j}\|$
$\epsilon^{(t)} = \rho g^{(t)}$
if $t \geq 2$ **then** $p^{(t)} = \min(2\|\beta^{(t-1)}\|_0, p)$
$\mathcal{W}^{(t)} = \{j \in [p] : d^{(t)}_j$ among $p^{(t)}$ smallest values of $d^{(t)}\}$
return $\epsilon^{(t)}, \mathcal{W}^{(t)}$

optimization, so for lighter notation we assume that the design matrix X is already restricted to features in the working set. The reader should be aware that in the rest of this section, β, X and p in fact refers to $\beta_{\mathcal{W}^{(t)}}$, $X_{:\mathcal{W}^{(t)}}$, and $p^{(t)}$.

Writing the data-fitting term $f(\beta) = F(X\beta)$, we have $\nabla^2 f(\beta) = X^\top D X$, where $D \in \mathbb{R}^{n \times n}$ is diagonal with $f_i''(\beta^\top \mathbf{x}_i)$ as its i-th diagonal entry. Using $H = \nabla^2 f(\beta^{(t)})$ we can approximate the primal objective by[2]:

$$f(\beta^{(t)}) + \nabla f(\beta^{(t)})^\top (\beta - \beta^{(t)}) + \frac{1}{2}(\beta - \beta^{(t)})^\top H (\beta - \beta^{(t)}) + \lambda \left\|\beta\right\|_1 \ . \tag{3.37}$$

[2] H and D should read $H^{(t)}$ and $D^{(t)}$ as they depend on $\beta^{(t)}$; we omit the exponent for brevity.

Algorithm 3.4 PROX-NEWTON SUBPROBLEM SOLVER (illustrated on logistic regression)

input : $X = [X_{:1}|\dots|X_{:p}] \in \mathbb{R}^{n\times p}, y \in \mathbb{R}^n, \lambda, \beta^{(0)} \in \mathbb{R}^p, \epsilon$
param: $\texttt{MAX_CD} = 20, \texttt{MAX_BACKTRACK} = 10, K = 5$
init : $\Delta\beta = \mathbf{0}_p, X\Delta\beta = \mathbf{0}_n, \theta^{(0)} = \mathbf{0}_n, D = \mathbf{0}_{n\times n}, L = \mathbf{0}_p,$
for $t = 1, \dots, T$ **do**
  for $i = 1, \dots, n$ **do** $D_{ii} = f_i''(\beta^\top \mathbf{x}_i)\Big(= \exp(y_i\beta^\top \mathbf{x}_i)/(1+\exp(y_i\beta^\top \mathbf{x}_i))^2\Big)$
  for $j = 1, \dots, p$ **do** $L_j = \langle X_{:j}, X_{:j}\rangle_D\Big(= \sum_{i=1}^n x_{ij}^2 \exp(y_i\beta^\top \mathbf{x}_i)/(1+\exp(y_i\beta^\top \mathbf{x}_i))^2\Big)$
  if $t = 1$ **then** $\texttt{MAX_CD} = 1$
  else $\texttt{MAX_CD} = 20$
  $\Delta\beta = \texttt{newton_direction}(X, y, \beta^{(t-1)}, D, L = (L_1, \dots, L_p), \texttt{MAX_CD})$
  $\alpha^{(t)} = \texttt{backtracking}(\Delta\beta, X\Delta\beta, y, \lambda, \texttt{MAX_BACKTRACK})$
  $\beta^{(t)} = \beta^{(t-1)} + \alpha^{(t)} \times \Delta\beta$
  $\theta_{\text{res}}^{(t)} = -\nabla F(X\beta^{(t)})/\lambda \left(= -y/(\lambda \mathbf{1}_n + \lambda \exp(y \odot X\beta^{(t)}))\right)$
  $\theta_{\text{res}}^{(t)} = \theta_{\text{res}}^{(t)} / \max(1, \|X^\top \theta_{\text{res}}^{(t)}\|_\infty)$
  $\theta^{(t)} = \arg\max_{\theta \in \{\theta^{(t-1)}, \theta_{\text{res}}\}} \mathcal{D}(\theta)$
  if $\mathcal{P}(\beta^{(t)}) - \mathcal{D}(\theta^{(t)}) < \epsilon$ **then**
    break
return $\beta^{(t)}$, $\theta^{(t)}$

Algorithm 3.5 `newton_direction` (illustrated on logistic regression)

input : $X = [X_{:1}|\dots|X_{:p}] \in \mathbb{R}^{n\times p}, y \in \mathbb{R}^n, \beta \in \mathbb{R}^p, D \in \mathbb{R}^{n\times n}, L \in \mathbb{R}^p, \texttt{MAX_CD}$
param: $\epsilon, \texttt{MIN_CD} = 2$
init : $\Delta\beta = \mathbf{0}_p, X\Delta\beta = \mathbf{0}_n$
for $k = 1, \dots, \texttt{MAX_CD}$ **do**
  $\tau = 0$ // stopping condition
  for $j = 1, \dots, p$ **do**
    $u_j = \beta_j + (\Delta\beta)_j$
    $\tilde{u}_j = \mathrm{ST}\left(\beta_j + (\Delta\beta)_j - \frac{1}{L_j}\left(X_{:j}^\top \nabla F(X\beta^{(t)}) - \langle X_{:j}, X\Delta\beta\rangle_D\right), \frac{\lambda}{L_j}\right)$ // see (3.41)
    $(\Delta\beta)_j = \tilde{u}_j - \beta_j$
    $X\Delta\beta \mathrel{+}= (\tilde{u}_j - u_j) X_{:j}$
    $\tau \mathrel{+}= (\tilde{u}_j - u_j)^2 \times L_j^2$
  if $\tau \le \epsilon$ and $k \ge \texttt{MIN_CD}$ **then** break
return $\Delta\beta$

Minimizing this approximation yields the direction $\Delta^{(t)}$ for the *proximal Newton* step:

$$\Delta^{(t)} + \beta^{(t)} = \arg\min_\beta \frac{1}{2}\left\|\beta - \beta^{(t)} + H^{-1}\nabla f(\beta^{(t)})\right\|_H^2 + \lambda\left\|\beta\right\|_1 \quad . \tag{3.38}$$

Then, a step size $\alpha^{(t)}$ is found by backtracking line search (Algorithm 3.6), and:

$$\beta^{(t+1)} = \beta^{(t)} + \alpha^{(t)}\Delta^{(t)} \quad . \tag{3.39}$$

Solving (3.38) amounts to solving the following Lasso problem:

$$u = \arg\min_u \frac{1}{2}\|\tilde{y} - \tilde{X}u\|_2^2 + \lambda \|u\|_1 \quad , \tag{3.40}$$

where $\tilde{X} = D^{1/2}X$, $\tilde{y} = D^{1/2}X\beta^{(t)} - D^{-1/2}X^{\dagger\top}X^\top \nabla F(X\beta^{(t)})$ and $X^\dagger$ is the pseudoinverse of X. While this may seem costly computationally, it turns out that the terms $X^\dagger, \tilde{y}$ and $\tilde{X}$ are not needed to solve (3.40) with coordinate descent. A coordinate descent update for (3.40) reads:

$$u_j \leftarrow \mathrm{ST}\left(u_j + \frac{1}{l_j}\tilde{X}_{:j}^\top\left(\tilde{y} - \tilde{X}u\right), \frac{\lambda}{l_j}\right) \quad , \tag{3.41}$$

where

$$\tilde{X}_{:j}^\top(\tilde{y} - \tilde{X}u) = X_{:j}^\top DX\beta^{(t)} - X_{:j}^\top \nabla F(X\beta^{(t)}) - X_{:j}^\top DXu \ , \tag{3.42}$$

$$l_j = X_{:j}^\top DX_{:j} \ . \tag{3.43}$$

Indeed, the update only involves X, y and inner products weighted by D. The algorithm is summarized in Algorithm 3.5.

Contrary to coordinate descent, Newton steps do not lead to an asymptotic VAR, which is required to guarantee the success of dual extrapolation. To address this issue, we compute K passes of cyclic coordinate descent restricted to the support of the current estimate β before defining a working set (Algorithm 3.2, line 5). The K values of $X\beta$ obtained allow for the computation of both θ_{accel} and θ_{res}. The motivation for restricting the coordinate descent to the support of the current estimate β comes from the observation that dual extrapolation proves particularly useful once the support is identified. The Prox-Newton solver we use is detailed in Algorithm 3.4. When Algorithm 3.2 is used with Algorithm 3.4 as inner solver, we refer to it as the Newton-Celer variant.

Values of parameters and implementation details In practice, Prox-Newton implementations such as GLMNET (Friedman et al., 2010), newGLMNET (Yuan et al., 2012) or QUIC (Hsieh et al., 2014) only solve the direction approximately in Equation (3.38). How inexactly the problem is solved depends on some heuristic values. For reproducibility, we expose the default values of these parameters as inputs to the algorithms. Importantly, the variable `MAX_CD` is set to 1 for the computation of the first Prox-Newton direction. Experiments have indeed revealed that a rough Newton direction for the first update was sufficient and resulted in a substantial speed-up. Other parameters are set based on existing Prox-Newton implementations such as Blitz.

3.5 Experiments

In this section, we numerically illustrate the benefits of dual extrapolation on various data sets. Implementation is done in Python, Cython (Behnel et al., 2011) and numba (Lam et al., 2015) for the low-level critical parts. The solvers exactly follow the `scikit-learn` API (Pedregosa et al., 2011; Buitinck et al., 2013), so that Celer can be used as a drop-in replacement in existing code. The package is available under BSD3

Algorithm 3.6 `backtracking` (illustrated on logistic regression)

input : $\Delta\beta, X\Delta\beta, \lambda$
param: `MAX_BACKTRACK` $= 20$
init : $\alpha = 1$

for $k = 1, \dots,$ `MAX_BACKTRACK` **do**
  $\delta = 0$
  for $j = 1, \dots, p$ **do**
    if $\beta_j + \alpha \times (\Delta\beta)_j < 0$ **then** $\delta \mathrel{-}= \lambda(\Delta\beta)_j$
    else if $\beta_j + \alpha \times (\Delta\beta)_j > 0$ **then** $\delta \mathrel{+}= \lambda(\Delta\beta)_j$
    else if $\beta_j + \alpha \times (\Delta\beta)_j = 0$ **then** $\delta \mathrel{-}= \lambda \left|(\Delta\beta)_j\right|$
  $\theta = \nabla F(X\beta + \alpha \times X\Delta\beta) \left(= -y \odot \sigma(-y \odot (X\beta + \alpha \times X\Delta\beta))\right)$
  $\delta \mathrel{+}= (X\Delta\beta)^\top \theta$
  if $\delta < 0$ **then** break
  else $\alpha = \alpha/2$
return α

license at https://github.com/mathurinm/celer, with documentation and examples at https://mathurinm.github.io/celer.

In all this section, the estimator-specific $\lambda_{\max}$ refers to the smallest value giving a null solution (for instance $\lambda_{\max} = \|X^\top y\|_\infty$ in the Lasso case).

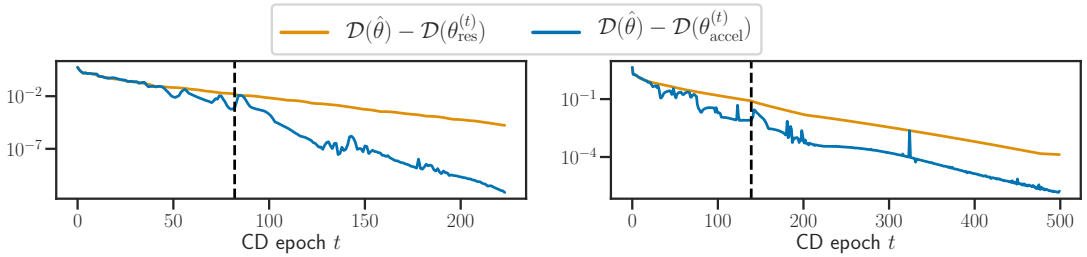
Table 3.1 – Characteristics of datasets used

name	n	p	q	density
leukemia	72	7129	-	1
news20	19 996	632 983	-	$6.1\,10^{-4}$
rcv1_train	20 242	19 960	-	$3.7\,10^{-3}$
Finance	16 087	1 668 738	-	$3.4\,10^{-3}$
Magnetoencephalography (MEG)	305	7498	49	1

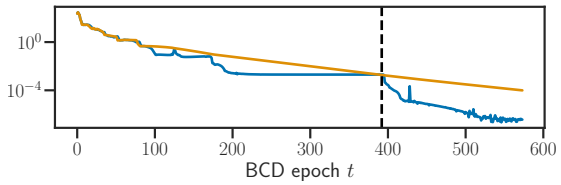
3.5.1 Illustration of dual extrapolation

For the Lasso (Figure 3.1a), Logistic regression (Figure 3.1b) and Multitask Lasso (Figure 3.1c), we illustrate the applicability of dual extrapolation. For all problems, the figures show that θ_{accel} gives a better dual objective after sign identification. They also show that the behavior is stable before identification. The peaks correspond to numerical errors. We choose to present the bare result of extrapolation, but peaks would not appear if we applied the robustifying procedure (2.27), as it forces the dual objective to be monotonic.

In particular, Figure 3.1c shows that dual extrapolation works in practice for the Multitask Lasso, even though there is no such result as sign identification, and we are not able to exhibit a VAR behavior for $(X\mathrm{B}^{(t)})_{t\in\mathbb{N}}$.



(a) Lasso, on *leukemia* for $\lambda = \lambda_{\max}/5$. (b) Log. reg., on *leukemia* for $\lambda = \lambda_{\max}/10$.



(c) Multitask Lasso, on MEG data for $\lambda = \lambda_{\max}/20$.

Figure 3.1 – Dual objectives with classical and proposed approach, for Lasso (top), Logistic regression (middle), Multitask Lasso (bottom). The dashed line marks sign identification (support identification for Multitask Lasso).

3.5.2 Improved screening and working set policy

In order to have a stopping criterion scaling with n, the solvers are stopped when the duality gap goes below $\epsilon \times F(\mathbf{0}_n)$.

Logistic regression

In this section, we evaluate separately the first order solvers (Gap Safe, Gap Safe with extrapolation, Celer with coordinate descent as inner solver), and the Prox-Newton solvers: Blitz, Newton-Celer with working set but without using dual extrapolation (PN WS), and Newton-Celer.

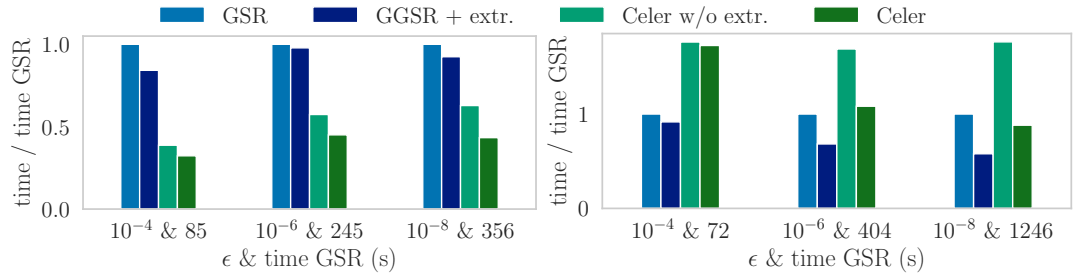


Figure 3.2 – Time to compute a Logistic regression path from $\lambda_{\max}$ to $\lambda_{\max}/100$ on the *news20* dataset (left: coarse grid of 10 values, right: fine grid of 100 values). $\lambda_{\max}/100$ gives 5319 non-zero coefficients.

Figure 3.2 shows that when cyclic coordinate descent is used, extrapolation improves the performance of screening rules, and that using a dual-based working set policy further reduces the computational burden.

Figure 3.3 shows the limitation of dual extrapolation when second order information is taken into account with a Prox-Newton: because the Prox-Newton iterations do not create a VAR sequence, it is necessary to perform some passes of coordinate descent to

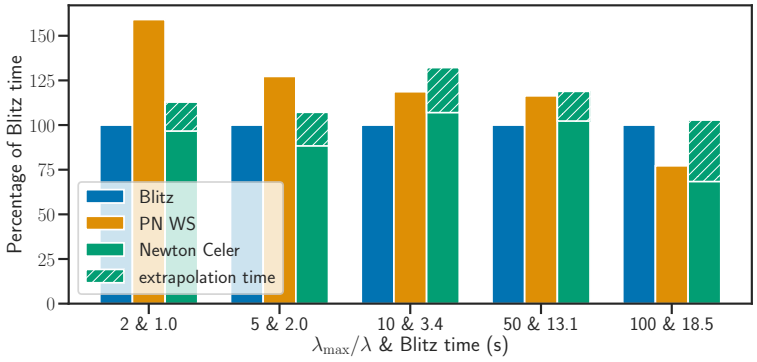


Figure 3.3 – Time to solve a Logistic regression problem for different values of λ, on the *rcv1 dataset* ($\epsilon = 10^{-6}$).

create θ_{accel}, as detailed in Section 3.4.2. This particular experiment reveals that this additional time unfortunately mitigates the gains observed in better working sets and stopping criterion.

Multitask Lasso

The data for this experiment uses magnetoencephalography (MEG) recordings which are collected for neuroscience studies. Here we use data from the *sample* dataset of the MNE software (Gramfort et al., 2014). Data were obtained using auditory stimulation. There are $n = 305$ sensors, $p = 7498$ source locations in the brain, and the measurements are time series of length $q = 49$. Using a Multitask Lasso formulation allows to reconstruct brain activitiy exhibiting a stable sparsity pattern across time (Gramfort et al., 2012). The inner solver for Celer is block coordinate descent, which is also used for the Gap Safe solver (Ndiaye et al., 2015).

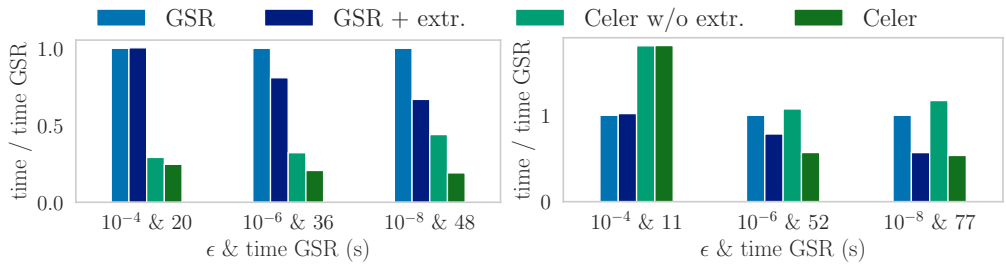


Figure 3.4 – Time to compute a Multitask Lasso path from $\lambda_{\max}$ to $\lambda_{\max}/100$ on MEG data (left: coarse grid of 10 values, right: fine grid of 100 values). $\lambda_{\max}/100$ gives 254 non-zero rows for $\hat{B}$.

While Figure 3.1c showed that for the Multitask Lasso the dual extrapolation performance also gives an improved duality gap, here Figure 3.4 shows that the working set policy of Celer performs better than Gap Safes rules with strong active warm start. We could not include Blitz in the benchmark as there is no standard public implementation for this problem.

3.6 Conclusion

In this chapter, we have generalized the dual extrapolation procedure for the Lasso (Celer) to any ℓ_1- regularized GLM, in particular sparse Logistic regression. We have provided theoretical guarantees based on sign identification of coordinate descent. Experiments show that dual extrapolation yields more efficient Gap Safe screening rules and working sets solver. Finally, we have adapted Celer to make it compatible with prox-Newton solvers, and empirically demonstrated its applicability to the Multi-task Lasso.

These speed improvements are applicable tp the state-of-the-art MxNE approach used for the bio-magnetic inverse problem. We now consider potentially more refined modeling of the noise structure in this problem.

Part II

Concomitant noise estimation for the M/EEG inverse problem

4

Concomitant estimation in multitask and multirepetition framework

It's looks (sic) like a bird, but, it's not a bird.

Contents

4.1 Introduction . . . 86
4.1.1 The (homoscedastic) Concomitant Lasso . . . 87
4.2 Heteroscedastic concomitant estimation . . . 88
4.2.1 SGCL: Working on averaged data . . . 89
4.2.2 CLaR: Exploiting all epochs . . . 90
4.3 Optimization properties of CLaR and SGCL . . . 90
4.3.1 Alternate minimization . . . 90
4.3.2 Duality results . . . 94
4.4 An analysis of CLaR through smoothing . . . 96
4.4.1 Smoothing of Schatten norms . . . 97
4.4.2 Smoothing of the nuclear norm . . . 100
4.5 Conclusion . . . 102

Lasso-type estimators help to fight the statistical curses of high dimension by performing variable selection. When the noise variance is constant, the regularization parameter for which statistical analysis of the Lasso holds is a linear function of the noise standard deviation, which is often unknown in practice. A way to address this dependency is to consider estimators such as the Concomitant Lasso, which jointly optimize over the regression coefficients and the noise level. In applications such as magneto-electroencephalography (M/EEG) where the observations are pooled from different sources to increase sample size, noise levels differ and the homoscedastic assumption of concomitant estimators no longer holds. More complex model are then needed, but the averaging step performed to reduce the noise variance dramatically reduces sample sizes, preventing finer modeling of the noise structure.

In this chapter, we provide new statistical and computational solutions to perform regression with correlated Gaussian noise, in the multitask, multirepetition context of M/EEG. We derive two joint estimators of the regression coefficients and the square root of the noise covariance matrix, computable via a jointly convex optimization problem. Joint convexity allows the estimators to be approximately computed easily with off-the-shelf optimization techniques, a notable asset compared to existing non-convex or hard to solve approaches. In particular, the block-coordinate descent techniques used in the

alternate minimization are amenable to the improvements introduced in Part I. As a theoretical analysis, we connect our optimization problem to the use of nuclear norm as a datafitting term, through the theory of smoothing.

This chapter covers the following publications:

- **M. Massias**, O. Fercoq, A. Gramfort, and J. Salmon. Generalized concomitant multi-task Lasso for sparse multimodal regression. In *AISTATS*, pages 998–1007, 2018a
- Q. Bertrand*, **M. Massias***, A. Gramfort, and J. Salmon. Handling correlated and repeated measurements with the smoothed Multivariate square-root Lasso. In *NeurIPS*, 2019

4.1 Introduction

In many statistical applications, the number of parameters p is much larger than the number of observations n. A popular approach to tackle regression problems in such conditions is to consider convex ℓ_1-type penalties, as popularized by Tibshirani (1996). The use of these penalties relies on a regularization parameter λ trading data fidelity versus sparsity, which requires careful tuning. Unfortunately, Bickel et al. (2009) showed that, in the case of homoscedastic Gaussian noise, the optimal λ linearly depends on the standard deviation of the noise – the *noise level*. Because the latter is rarely known in practice, one can jointly estimate the noise level and the regression coefficients, following pioneering work on concomitant estimation (Huber and Dutter, 1974; Huber, 1981). Adaptations to sparse regression have been analyzed under the names of square-root Lasso (Belloni et al., 2011) or Scaled Lasso (Sun and Zhang, 2012). Alternatively, Städler et al. (2010) considered a joint penalized maximum likelihood approach, using a change of variable to avoid minimization of a non-convex function.

The Concomitant Lasso (Owen, 2007) is jointly convex and admits a "smooth + proximable" structure for the minimization over the regression coefficients. This makes it easy to solve in practice, even in high dimension: while solvers for the "non-smooth + non-smooth" square-root Lasso relied on second order cone programming (Becker et al., 2011), the concomitant solver of Ndiaye et al. (2017a) relies on coordinate descent. It scales gracefully with the dimension p thanks to the use of screening rules (El Ghaoui et al., 2012; Fercoq et al., 2015), making it as fast as a pure Lasso solver.

In the aforementioned contributions, the noise parameter is a single scalar, the variance. Yet, in M/EEG and other applied settings, data of different natures or from different sources must be aggregated to increase the number of observations and improve performance. This often leads to correlation and magnitude variation in the noise: the data may be contaminated with non-white noise (see the statistical analysis of Daye et al. 2012; Wagener and Dette 2012; Kolar and Sharpnack 2012 for non-uniform noise levels).

Wagener and Dette (2012) proposed to estimate the variance with a preliminary adaptive Lasso step, and correct the data-fitting term in a second step. Other works model the log-variance as a linear combination of the features, leading to non-convex objective functions without convergence guarantees. Solvers considered for such approaches require alternate minimization (Kolar and Sharpnack, 2012), possibly in an iterative fashion (Daye et al., 2012), a notable difference with the jointly convex formulation

proposed here, for which one can control global optimality with duality gap certificates.

In our application, the (far from scalar!) structure of the noise covariance of M/EEG data is illustrated in Figure 1.9 (page 34), where we can also see that EEG noise has a standard deviation of the order of a few μV, while gradiometers have a noise standard deviation of a few fT/cm. When pooling such signals together, the absolute value of the noise therefore differs by several orders of magnitude.

To address the correlated noise problem, estimators based on non-convex optimization problems were proposed (Lee and Liu, 2012) and analyzed for sub-Gaussian covariance matrices (Chen and Banerjee, 2017) through penalized Maximum Likelihood Estimation (MLE). Other estimators (Rothman et al., 2010; Rai et al., 2012) assume that the inverse of the covariance (the *precision matrix*) is sparse, but the underlying optimization problems remain non-convex.

Our goal in this chapter is to propose a convex and numerically easy to solve approach to heteroscedastic regression, as the noise covariance matrix estimation is a notorious challenge for M/EEG data (Engemann and Gramfort, 2015a).

We first introduce the Smooth Generalized Concomitant Lasso (SGCL), the most general extension of the Concomitant Lasso to the multitask case. This approach estimates the full square root of the covariance, a statistical challenge when the number of tasks does not dwarf the number of observation. As a proof of concept, we first introduce the block model, where the noise is assumed to be homoscedastic per type of sensor.

In a more refined approach, we then exploit the multiple repetitions structure of M/EEG data: because the SNR is too low, M/EEG measurements are commonly repeated and averaged. Indeed, under the assumption that the signal of interest is corrupted by additive independent realizations of noise, averaging measurements reduces the noise variance by the number of repetitions. Popular estimators for M/EEG usually discard the individual observations, therefore relying on homoscedastic noise models (Ou et al., 2009; Gramfort et al., 2013).

We propose the Concomitant Lasso with Repetitions (CLaR), an estimator that is:

- designed to exploit all available measurements collected during repetitions of experiments,
- defined as the solution of a *convex* minimization problem, handled efficiently by proximal block coordinate descent techniques,
- built thanks to an *explicit* connection with nuclear norm smoothing,
- shown (through extensive benchmarks *w.r.t.* existing estimators) to leverage experimental repetitions to improve support identification

We first introduce briefly the seminal Concomitant Lasso estimator and its cousins. In Section 4.2, we recall the framework of concomitant estimation, and introduce the SGCL and its refinement, CLaR. In Section 4.3, we study the properties of CLaR from the optimizer's point of view, and derive an algorithm to solve it. In Section 4.4, we justify the for through the use of a smoothed nuclear norm as a datafitting term.

4.1.1 The (homoscedastic) Concomitant Lasso

We start by considering the single task setting ($y \in \mathbb{R}^n, \beta \in \mathbb{R}^p$), with an homoscedastic model $y = X\beta^* + \varepsilon$, $\varepsilon \in \mathbb{R}^n$ having i.i.d. entries $\mathcal{N}(0, \sigma_*^2)$. To estimate β^* with an optimal regularization strength independent σ_*, the seminal estimator is the Concomitant Lasso,

proposed under different forms in Owen (2007); Sun and Zhang (2012).

Definition 4.1. *For $\lambda > 0$, the Concomitant Lasso regression coefficients and standard deviation estimators are defined as solutions of the optimization problem*

$$\underset{\beta \in \mathbb{R}^p, \sigma > 0}{\arg\min} \frac{\|y - X\beta\|^2}{2n\sigma} + \frac{\sigma}{2} + \lambda \|\beta\|_1 \quad . \tag{4.1}$$

This estimator is closely connected to the square-root Lasso (Belloni et al., 2011).

Definition 4.2. *For $\lambda > 0$, the square-root Lasso is defined as the solution of:*

$$\underset{\beta \in \mathbb{R}^p}{\arg\min} \frac{\|y - X\beta\|}{\sqrt{n}} + \lambda \|\beta\|_1 \quad . \tag{4.2}$$

The "close connexion" is the following: for a given $\lambda > 0$, any solution of the square-root Lasso $\hat{\beta}_{\surd}$ such that $y - X\hat{\beta}_{\surd} \neq \mathbf{0}_n$ is also a solution of the Concomitant Lasso with same regularization parameter, and associated noise level estimate $\hat{\sigma} = \|y - X\hat{\beta}_{\surd}\|/\sqrt{n}$. As mentioned in the introduction in the case of Tikhonov regression, the square-root Lasso and the Lasso have the same solution paths: the interest of the square-root Lasso resides in an easier study of its regularization parameter.

Finally, notice that the data-fitting term of the square-root Lasso is not smooth (a norm is never differentiable at the origin), making it a challenging "non-smooth + non-smooth" optimization problem, while the Concomitant Lasso has a jointly convex objective function ($(a, b) \mapsto a^2/b$ is convex on $(\mathbb{R}_+^*)^2$: its Hessian is $2/b \cdot vv^\top$ with $v = (1, -a/b)^\top$). Additionally, the objective's dependency in β is amenable to proximal coordinate descent. Nevertheless, numerical issues can arise for the Concomitant Lasso when σ approaches 0; in Ndiaye et al. (2017a) it was proposed to add a constraint on σ in the objective function. Following the terminology introduced in Nesterov (2005), this was coined the Smoothed Concomitant Lasso.

Definition 4.3. *For $\underline{\sigma} > 0$ and $\lambda > 0$, the Smoothed Concomitant Lasso estimator and its associated standard deviation estimator are defined as:*

$$\underset{\beta \in \mathbb{R}^p, \sigma \geq \underline{\sigma}}{\arg\min} \frac{\|y - X\beta\|^2}{2n\sigma} + \frac{\sigma}{2} + \lambda \|\beta\|_1 \quad . \tag{4.3}$$

In these concomitant models, the assumption is that the noise is Gaussian homoscedastic. We aim at generalizing them to the more complex M/EEG noise structure.

4.2 Heteroscedastic concomitant estimation

Notation and probabilistic model For more compact notation, for $\underline{\sigma} > 0$ we denote $\underline{S} = \underline{\sigma}\,\mathrm{Id}_n$. Let r be the number of repetitions of the experiment (*trials* in M/EEG vocabulary). The r observation matrices are denoted $Y^1, \ldots, Y^r \in \mathbb{R}^{n \times q}$ with n the number of samples (sensors) and q the number of tasks (time samples). The mean over the repetitions of the observation matrices is written $\bar{Y} \triangleq \frac{1}{r}\sum_{l=1}^r Y^l$. Let $X \in \mathbb{R}^{n \times p}$ be the design (or gain) matrix, with p features stored column-wise: $X = [X_{:1}|\ldots|X_{:p}]$. The matrix $\mathrm{B}^* \in \mathbb{R}^{p \times q}$ contains the coefficients of the linear regression model. The residuals are defined as $R^l \triangleq Y^l - X\mathrm{B} \in \mathbb{R}^{n \times q}$.

We assume that each measurement follows the common model:

$$\forall l \in [r], \quad Y^l = X\mathrm{B}^* + S^*\mathrm{E}^l \ , \tag{4.4}$$

where the entries of E^l are i.i.d. samples from standard normal distributions, the E^l's are independent, and $S^* \in \mathcal{S}^n_{++}$ is the co-standard deviation matrix. Note that even if the observations $Y^1, \dots, Y^r$ differ because of the noise $\mathrm{E}^1, \dots, \mathrm{E}^r$, both B^* and the noise structure S^* are shared across repetitions. This is a natural assumption for stable physical systems observed with sensor or background noise.

In low signal-to-noise ratio (SNR) situations, a standard way to deal with strong noise is to use the averaged observation $\bar{Y} \in \mathbb{R}^{n \times q}$ instead of the raw observations. The associated model reads:

$$\bar{Y} = X\mathrm{B}^* + \tilde{S}^*\tilde{\mathrm{E}} \ , \tag{4.5}$$

with $\tilde{S}^* \triangleq S^*/\sqrt{r}$ and $\tilde{\mathrm{E}}$ has i.i.d. entries drawn from a standard normal distribution. The SNR (see the rescaled definition we consider in Equation (5.13)) is multiplied by $\sqrt{r}$, yet the number of samples goes from rnq to nq, making it statistically challenging to estimate correctly the $\mathcal{O}(n^2)$ parameters of S^*.

We originally considered Model (4.5) and designed an heteroscedastic estimator on averaged data.

4.2.1 SGCL: Working on averaged data

Contrary to the Concomitant Lasso, the Smooth Generalized Concomitant Lasso estimates the full square root of the covariance of the noise.

Definition 4.4 (SGCL, **Massias** et al. (2018a))**.** *The SGCL estimates the parameters of Model (4.5), by solving:*

$$(\hat{\mathrm{B}}^{\mathrm{SGCL}}, \hat{\mathrm{S}}^{\mathrm{SGCL}}) \in \underset{\substack{\mathrm{B} \in \mathbb{R}^{p \times q} \\ \tilde{S} \succeq \underline{\sigma}/\sqrt{r}\,\mathrm{Id}_n}}{\arg\min} \ \frac{\|\bar{Y} - X\mathrm{B}\|^2_{\tilde{S}^{-1}}}{2nq} + \frac{\mathrm{Tr}(\tilde{S})}{2n} + \lambda \left\|\mathrm{B}\right\|_{2,1} \ . \tag{4.6}$$

As for $\underline{\sigma}$ in the Smoothed Concomitant Lasso, the constraint $S \succeq \underline{\sigma}/\sqrt{r}\,\mathrm{Id}_n$ acts as a regularizer in the dual, and is introduced for numerical stability. To set the value of $\underline{\sigma}$, one can use a proportion of the initial estimation of the noise standard deviation: $\underline{\sigma} = 10^{-\alpha}\|Y\|/\sqrt{nq}$, with for example $\alpha \in \{2, 3\}$. The SGCL corresponds to the most general framework to adapt the Concomitant Lasso to multi-task and non scalar covariances. However, in its general formulation it has an obvious drawback: in practice estimating $\tilde{S}^*$ requires to fit $n(n-1)/2$ parameters with only nq observations, which is problematic if q is not large enough. Hence, additional regularization might be needed to provide an accurate estimator of S^*. For example, the *shrinking* approach of Ledoit and Wolf (2004) estimates the covariance as a weighted average of the identity and the sample covariance matrix, in order to improve its conditioning; the resulting estimator is proven to be well-conditioned and more asymptotically more accurate than the sample covariance (when *both* p and n got to infinity).

Our first direction of research was to assume a more regular structure for $\tilde{S}^*$, motivated by the specificities of the M/EEG inverse problem. More generally, in supervised learning problems where data come from an identified, finite set of sources, we proposed a specification of Model (4.5), when the observations come from K different sources

or K types of sensors (in the M/EEG case: magnetometers, gradiometers and electrodes). In that case, the variant of Model (4.5) called the *block homoscedastic* model constrains $\tilde{S}^*$ to be diagonal, the diagonal being constant over known blocks. For the sake of completeness, this block model and proofs of concept experiments are presented in Appendix B. However, this is a somehow simplistic model, and a better solution is to estimate a full covariance by exploiting all repetitions instead of averaging them.

4.2.2 CLaR: Exploiting all epochs

To leverage the multiple repetitions while taking into account the noise structure, we introduced the Concomitant Lasso with Repetitions (CLaR).

Definition 4.5 (CLaR, Bertrand, **Massias**, Gramfort, and Salmon (2019))**.** *CLaR estimates the parameters of Model (4.4) by solving:*

$$(\hat{\mathrm{B}}^{\mathrm{CLaR}}, \hat{\mathrm{S}}^{\mathrm{CLaR}}) \in \underset{\substack{\mathrm{B}\in\mathbb{R}^{p\times q} \\ S\succeq \underline{\sigma}\,\mathrm{Id}_n}}{\arg\min} \underbrace{\frac{1}{2nqr}\sum_{l=1}^{r}\|Y^l - X\mathrm{B}\|_{S^{-1}}^2 + \frac{\mathrm{Tr}(S)}{2n}}_{\triangleq f(\mathrm{B},S)} + \lambda \left\|\mathrm{B}\right\|_{2,1} \quad , \tag{4.7}$$

where $\lambda > 0$ controls the sparsity of $\hat{\mathrm{B}}^{\mathrm{CLaR}}$ and $\underline{\sigma} > 0$ controls the smallest eigenvalue of $\hat{\mathrm{S}}^{\mathrm{CLaR}}$.

It is clear that CLaR is a generalization the SGCL, in the case of a single repetition with lower noise.

Remark 4.6. *CLaR and SGCL are the same when $r = 1$ and $Y^1 = \bar{Y}$. In the case $r > 1$, note that $\hat{\mathrm{S}}^{\mathrm{CLaR}}$ estimates S^*, while $\hat{\mathrm{S}}^{\mathrm{SGCL}}$ estimates $\tilde{S}^* = S^*/\sqrt{r}$. Since we impose the constraint $\hat{\mathrm{S}}^{\mathrm{CLaR}} \succeq \underline{\sigma}\,\mathrm{Id}_n$, we rescale the constraint so that $\hat{\mathrm{S}}^{\mathrm{SGCL}} \succeq \underline{\sigma}/\sqrt{r}\,\mathrm{Id}_n$ in (4.6) for future comparisons.*

The justification for CLaR is the following: if the quadratic loss $\left\|Y - X\mathrm{B}\right\|^2$ were used, the parameters of Model (4.4) could be estimated by using either $\|\bar{Y} - X\mathrm{B}\|^2$ or $\frac{1}{r}\sum\|Y^l - X\mathrm{B}\|^2$ as a data-fitting term. Yet, both alternatives yield the same solutions as the two terms are equal up to constants. Hence, the quadratic loss does not leverage the multiple repetitions and ignores the noise structure. On the contrary, the more refined data-fitting term of CLaR allows to take into account the individual repetitions, leading to improved performance in applications.

4.3 Optimization properties of CLaR and SGCL

We detail the principal results needed to solve Problem (4.7) numerically, leading to the implementation proposed in Algorithm 4.1. We first recall useful results for alternate minimization of convex composite problems.

4.3.1 Alternate minimization

Proposition 4.7. *CLaR is jointly convex in (B, S). Moreover, f is convex and smooth on the feasible set, and $\|\cdot\|_{2,1}$ is convex and separable in $\mathrm{B}_{j:}$'s, thus minimizing the objective alternatively in S and in $\mathrm{B}_{j:}$'s (see Algorithm 4.1) converges to a global minimum.*

Proof The expression of f is:

$$f(\mathrm{B}, S) = \frac{1}{2nqr} \sum_{1}^{r} \left\| Y^l - X\mathrm{B} \right\|_{S^{-1}}^2 + \frac{1}{2n} \operatorname{Tr}(S) = \frac{1}{2n} \operatorname{Tr}(Z^\top S^{-1} Z) + \frac{1}{2n} \operatorname{Tr}(S) \ , \tag{4.8}$$

with $Z = \frac{1}{\sqrt{qr}} [Y^1 - X\mathrm{B} | \dots | Y^r - X\mathrm{B}]$.

First note that the function $(Z, S) \mapsto \operatorname{Tr} Z^\top S^{-1} Z$ is jointly convex over $\mathbb{R}^{n \times q} \times \mathcal{S}_{++}^n$ (Boyd and Vandenberghe, 2004, Example 3.4). This means that f as a function of (Z, S) is jointly convex. Moreover $\mathrm{B} \mapsto [Y^1 - X\mathrm{B} | \dots | Y^r - X\mathrm{B}]$ is linear in B, thus f is jointly convex in (B, S), meaning that $(\mathrm{B}, S) \mapsto f + \lambda \left\| \cdot \right\|_{2,1}$ is jointly convex in (B, S). Finally, the constraint set is convex and thus Problem (4.7) is convex.

The function f is convex and smooth on the feasible set and $\left\| \cdot \right\|_{2,1}$ is convex in B and separable in $\mathrm{B}_{j:}$'s, thus $f + \lambda \left\| \cdot \right\|_{2,1}$ can be minimized through coordinate descent in S and the $\mathrm{B}_{j:}$'s (Tseng, 2001; Tseng and Yun, 2009). ■

By virtue of Proposition 4.7, to solve Problem (4.7) we only need to consider solving problems with B or S fixed, which we detail in the next propositions.

Definition 4.8 (Clipped Square Root)**.** *For $S \in \mathcal{S}_+^n$ with spectral decomposition $S = U \operatorname{diag}(\gamma_1, \dots, \gamma_n) U^\top$ (U is orthogonal and the γ_i's are positive), let us define the* Clipped Square Root *operator:*

$$\mathrm{ClSqrt}(S, \underline{\sigma}) \triangleq U \operatorname{diag}(\sqrt{\gamma_1} \vee \underline{\sigma}, \dots, \sqrt{\gamma_n} \vee \underline{\sigma}) U^\top \ . \tag{4.9}$$

Proposition 4.9. *Let $\mathrm{B} \in \mathbb{R}^{n \times q}$ be fixed. The minimization of $f(\mathrm{B}, S)$ with respect to S with the constraint $S \succeq \underline{\sigma} \operatorname{Id}_n$ admits the closed-form solution:*

$$S = \mathrm{ClSqrt}\left(\frac{1}{qr} \sum_{l=1}^{r} (Y^l - X\mathrm{B})(Y^l - X\mathrm{B})^\top, \underline{\sigma} \right) \ . \tag{4.10}$$

Proof Minimizing $f(\mathrm{B}, \cdot)$ amounts to solving:

$$\underset{S \succeq \underline{\sigma} \operatorname{Id}_n}{\arg\min} \operatorname{Tr} Z^\top S^{-1} Z + \operatorname{Tr}(S) \ , \quad \text{with } Z = \frac{1}{\sqrt{qr}} [Y^1 - X\mathrm{B} | \dots | Y^r - X\mathrm{B}] \ , \tag{4.11}$$

a problem for which strong duality holds since the objective function is convex and the feasible set has non-empty interior. The associated Lagrangian formulation is:

$$\min_{S \in \mathcal{S}_+^n} \max_{\Lambda \in \mathcal{S}_+^n} \underbrace{\operatorname{Tr} Z^\top S^{-1} Z + \operatorname{Tr}(S) + \operatorname{Tr}(\Lambda^\top (\underline{S} - S))}_{\mathcal{L}(S, \Lambda)} \ . \tag{4.12}$$

Recall that the gradient of $S \mapsto \operatorname{Tr} Z^\top S^{-1} Z$ is $-S^{-1} Z Z^\top S^{-1}$ (on $\mathcal{S}_{++}^n$), and that $ZZ^\top = \frac{1}{qr} \sum_{l=1}^{r} (Y^l - X\mathrm{B})(Y^l - X\mathrm{B})^\top$. The first order optimality conditions read:

$$\begin{cases} \frac{\partial \mathcal{L}(\cdot, \hat{\Lambda})}{\partial S}(\hat{S}) = -\hat{S}^{-1} Z Z^\top \hat{S}^{-1} + \operatorname{Id}_n - \hat{\Lambda} = \mathbf{0}_{n,n} \ , \\ \hat{\Lambda}^\top (\underline{S} - \hat{S}) = \mathbf{0}_{n,n} \ , \\ \hat{\Lambda} \in \mathcal{S}_+^n \ , \\ \hat{S} \succeq \underline{S} \ . \end{cases} \tag{4.13}$$

Let $U \operatorname{diag}(\lambda_1, \dots, \lambda_d, 0, \dots, 0) U^\top$ be an eigenvalue decomposition of $ZZ^\top$, with d the rank of $ZZ^\top$, $\lambda_1 \geq \cdots \geq \lambda_d > 0$ and $UU^\top = \mathrm{Id}_n$.

For $i \in [d]$, let us define $\mu_i \triangleq \sqrt{\lambda_i} \vee \underline{\sigma}$, $S = U \operatorname{diag}(\mu_1, \dots, \mu_d, \underline{\sigma}, \dots, \underline{\sigma}) U^\top$, and $\Lambda = \mathrm{Id}_n - S^{-1} Z Z^\top S^{-1}$. It is clear by construction that $S \succeq \underline{S}$. We also have:

$$\begin{aligned} \Lambda &= U \operatorname{diag}(1, \dots, 1) U^\top - U \operatorname{diag}(\tfrac{\lambda_1}{\mu_1^2}, \dots, \tfrac{\lambda_d}{\mu_d^2}, 0, \dots, 0) U^\top \\ &= U \operatorname{diag}(1 - \tfrac{\lambda_1}{\mu_1^2}, \dots, 1 - \tfrac{\lambda_1}{\mu_1^2}, 1, \dots, 1) U^\top \\ &\succeq \mathbf{0}_{n,n} \ , \end{aligned} \tag{4.14}$$

where the later holds by definition of the μ_i's. Moreover:

$$\Lambda^\top (S - \underline{S}) = U \operatorname{diag}\Big((1 - \tfrac{\lambda_1}{\mu_1^2})(\mu_1 - \underline{\sigma}), \dots, (1 - \tfrac{\lambda_d}{\mu_d^2})(\mu_d - \underline{\sigma}), 0, \dots, 0 \Big) U^\top = \mathbf{0}_{n,n} \ , \tag{4.15}$$

since $\forall i \in [r]$, $(1 - \frac{\lambda_i}{\mu_i^2})(\mu_i - \underline{\sigma}) = 0$ (either the left term or the right term of the LHS is 0 by construction).

This shows that the pair (S, Λ) satisfies all the first order conditions on the Lagrangian, hence that S is solution of Problem (4.11). ■

Proposition 4.10. *For a fixed $S \in \mathcal{S}_{++}^n$, each step of the block minimization of $f(\cdot, S) + \lambda \|\cdot\|_{2,1}$ in the j^{th} line of* B *admits a closed-form solution:*

$$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top S^{-1}(\bar{Y} - X\mathrm{B})}{\|X_{:j}\|_{S^{-1}}^2}, \frac{\lambda n q}{\|X_{:j}\|_{S^{-1}}^2} \right) \ . \tag{4.16}$$

Proof The function to minimize is the sum of a smooth term $f(\cdot, S)$ and a non-smooth but row-wise separable term, $\|\cdot\|_{2,1}$, whose proximal operator can be computed:

- f is $\|X_{:j}\|_{S^{-1}}^2 / nq$-smooth with respect to $\mathrm{B}_{j:}$, with partial gradient $\nabla_j f(\cdot, S) = -\frac{1}{nq} X_{:j}^\top S^{-1}(\bar{Y} - X\mathrm{B})$,
- $\|\mathrm{B}\|_{2,1} = \sum_{j=1}^p \|\mathrm{B}_{j:}\|$ is row-wise separable over B, with:

$$\mathrm{prox}_{\lambda nq / \|X_{:j}\|_{S^{-1}}^2, \|\cdot\|}(\cdot) = \mathrm{BST}\left(\cdot, \frac{\lambda n q}{\|X_{:j}\|_{S^{-1}}^2} \right) \ . \tag{4.17}$$

Hence, proximal block-coordinate descent converges (Tseng and Yun, 2009), and the updates are given by Equation (4.16). The closed-form formula arises since the smooth part of the objective is quadratic and isotropic *w.r.t.* $\mathrm{B}_{j:}$. ■

As for the Lasso, there exists $\lambda_{\max} \geq 0$ such that whenever $\lambda \geq \lambda_{\max}$, the estimated coefficients vanish. This critical value helps to roughly calibrate λ in practice, by choosing it as a fraction of $\lambda_{\max}$.

Proposition 4.11 (Critical regularization parameter)**.** *For the CLaR estimator we have, with $S_{\max} \triangleq \mathrm{ClSqrt}\left(\frac{1}{qr} \sum_{l=1}^r Y^l Y^{l\top}, \underline{\sigma} \right)$:*

$$\forall \lambda \geq \lambda_{\max} \triangleq \frac{1}{nq} \|X^\top S_{\max}^{-1} \bar{Y}\|_{2,\infty}, \quad \hat{\mathrm{B}}^{\mathrm{CLaR}} = \mathbf{0}_{p,q} \ . \tag{4.18}$$

Algorithm 4.1 ALTERNATE MINIMIZATION FOR CLAR

input : $X, \bar{Y}, \underline{\sigma}, \lambda, f^{\text{dual}}, T$
init : $\mathrm{B} = \mathbf{0}_{p,q}$, $S^{-1} = \underline{\sigma}^{-1}\,\mathrm{Id}_n$, $\bar{R} = \bar{Y}$, $\mathrm{cov}_Y = \frac{1}{r}\sum_{l=1}^{r} Y^l Y^{l\top}$ `// precomputed`

for $t = 1, \dots, T$ **do**
  if $t = 1 \pmod{f^{\text{dual}}}$ **then** `// noise update`
    $RR^\top = \mathrm{RRT}(\mathrm{cov}_Y, Y, X, \mathrm{B})$ `// eq. (4.20)`
    $S = \mathrm{ClSqrt}(\frac{1}{qr}RR^\top, \underline{\sigma})$ `// eq. (4.10)`
    for $j = 1, \dots, p$ **do** $L_j = X_{:j}^\top S^{-1} X_{:j}$
  for $j = 1, \dots, p$ **do** `// coef. update`
    $\bar{R} = \bar{R} + X_{:j}\mathrm{B}_{j:}$ `// partial residual update`
    $\mathrm{B}_{j:} = \mathrm{BST}\Big(\frac{X_{:j}^\top S^{-1}\bar{R}}{L_j}, \frac{\lambda n q}{L_j}\Big)$
    $\bar{R} = \bar{R} - X_{:j}\mathrm{B}_{j:}$ `// residual update`
return B, S

Proof First notice that if $\hat{\mathrm{B}} = 0$, then $\hat{S} = \mathrm{ClSqrt}\left(\frac{1}{qr}\sum_{l=1}^{r} Y^l Y^{l\top}, \underline{\sigma}\right) \triangleq S_{\max}$. Then, according to Fermat's rule:

$$\begin{aligned}
\hat{\mathrm{B}} = \mathbf{0}_{p,q} &\iff \mathbf{0}_{p,q} \in \partial\Big(f(\cdot, S_{\max}) + \lambda\|\cdot\|_{2,1}\Big)(\mathbf{0}_{p,q}) \\
&\iff -\nabla f(\mathbf{0}_{p,q}, S_{\max}) \in \lambda\mathcal{B}_{\|\cdot\|_{2,\infty}} \\
&\iff \frac{1}{nq}\left\|X^\top S_{\max}^{-1}\bar{Y}\right\|_{2,\infty} \triangleq \lambda_{\max} \leq \lambda \ . \qquad (4.19)
\end{aligned}$$

■

Remark 4.12. *Once* $cov_Y \triangleq \frac{1}{r}\sum_1^r Y^l Y^{l\top}$ *is pre-computed, the cost of updating* S *does not depend on* r, i.e., *is the same as working with averaged data. Indeed, with* $R = [Y^1 - X\mathrm{B}|\dots|Y^r - X\mathrm{B}]$, *the following computation can be done in* $\mathcal{O}(qn^2)$*:*

$$\begin{aligned}
RR^\top &= RRT(cov_Y, Y, X, \mathrm{B}) \\
&\triangleq r\,cov_Y + r(X\mathrm{B})(X\mathrm{B})^\top - r\bar{Y}^\top(X\mathrm{B}) - r(X\mathrm{B})^\top\bar{Y} \ . \qquad (4.20)
\end{aligned}$$

Proof

$$\begin{aligned}
RR^\top &= \sum_{l=1}^{r} R^l R^{l\top} \\
&= \sum_{l=1}^{r}(Y^l - X\mathrm{B})(Y^l - X\mathrm{B})^\top \\
&= \sum_{l=1}^{r} Y^l Y^{l\top} - \sum_{1}^{r} Y^l(X\mathrm{B})^\top - \sum_{1}^{r} X\mathrm{B}Y^{l\top} + rX\mathrm{B}(X\mathrm{B})^\top \\
&= r\mathrm{cov}_Y - r\bar{Y}^\top X\mathrm{B} - r(X\mathrm{B})^\top\bar{Y} + rX\mathrm{B}(X\mathrm{B})^\top \ . \qquad (4.21)
\end{aligned}$$

■

Additionally, statistical properties showing the advantages of using CLaR over SGCL can be found in Appendix B.3.1.

Thanks to the convex formulation, convergence of Algorithm 4.1 can be ensured using the duality gap as a stopping criterion (as it guarantees a targeted sub-optimality level). To compute the duality gap, we derive the dual of Problem (4.7) in Proposition 4.13. In addition, convexity allows to leverage acceleration methods such as working sets strategies (Fan and Lv, 2008; Tibshirani et al., 2012; Johnson and Guestrin, 2015) or safe screening rules (El Ghaoui et al., 2012; Ndiaye et al., 2017b), and their improvements detailed in Part I, while retaining theoretical guarantees of convergence. Such techniques are trickier to adapt in the non-convex case, as they could change the local minima reached.

4.3.2 Duality results

Proposition 4.13. *With* $\hat{\Theta} = (\hat{\Theta}^1, \dots, \hat{\Theta}^r)$*, a dual formulation of Problem (4.7) is:*

$$\hat{\Theta} = \underset{(\Theta^1,\dots,\Theta^r)\in\Delta_{X,\lambda}}{\arg\max} \frac{\sigma}{2}\Big(1 - \frac{qn\lambda^2}{r}\sum_{l=1}^{r} \operatorname{Tr}\Theta^l\Theta^{l\top}\Big) + \frac{\lambda}{r}\sum_{l=1}^{r}\Big\langle \Theta^l, Y^l\Big\rangle \ , \tag{4.22}$$

with

$$\bar{\Theta} = \frac{1}{r}\sum_{1}^{r}\Theta^l \tag{4.23}$$

$$\Delta_{X,\lambda} = \Big\{(\Theta^1,\dots,\Theta^r) \in (\mathbb{R}^{n\times q})^r : \|X^\top\bar{\Theta}\|_{2,\infty} \le 1, \Big\|\sum_{l=1}^{r}\Theta^l\Theta^{l\top}\Big\|_2 \le \frac{r}{\lambda^2 n^2 q}\Big\} \ , \tag{4.24}$$

Proof The primal optimum is:

$$\begin{aligned}
p^* &\triangleq \min_{\substack{\mathrm{B}\in\mathbb{R}^{p\times q}\\ S\succeq\underline{\sigma}\,\mathrm{Id}_n}} \frac{1}{2nqr}\sum_{l=1}^{r}\|Y^l - X\mathrm{B}\|_{S^{-1}}^2 + \frac{1}{2n}\operatorname{Tr}(S) + \lambda\big\|\mathrm{B}\big\|_{2,1} \\
&= \min_{\substack{\mathrm{B}\in\mathbb{R}^{p\times q}\\ R^l = Y^l - X\mathrm{B},\ \forall l\in[r]\\ S\succeq\underline{\sigma}\,\mathrm{Id}_n}} \frac{1}{2nqr}\sum_{l=1}^{r}\|R^l\|_{S^{-1}}^2 + \frac{1}{2n}\operatorname{Tr}(S) + \lambda\big\|\mathrm{B}\big\|_{2,1} \\
&= \min_{\substack{\mathrm{B}\in\mathbb{R}^{p\times q}\\ R^1,\dots,R^r\in\mathbb{R}^{n\times q}\\ S\succeq\underline{\sigma}\,\mathrm{Id}_n}} \max_{\Theta^1,\dots,\Theta^r\in\mathbb{R}^{n\times q}} \frac{1}{2nqr}\sum_{l=1}^{r}\|R^l\|_{S^{-1}}^2 + \frac{1}{2n}\operatorname{Tr}(S) \\
&\qquad + \lambda\big\|\mathrm{B}\big\|_{2,1} + \frac{\lambda}{r}\sum_{l=1}^{r}\Big\langle \Theta^l, Y^l - X\mathrm{B} - R^l\Big\rangle \ .
\end{aligned}$$

Since Slater's conditions are met min and max can be inverted:

$$\begin{aligned} p^* = \max_{\Theta^1,\dots,\Theta^r\in\mathbb{R}^{n\times q}} \min_{\substack{\mathrm{B}\in\mathbb{R}^{p\times q}\\ R^1,\dots,R^r\in\mathbb{R}^{n\times q}\\ S\succeq\underline{\sigma}\,\mathrm{Id}_n}} & \frac{1}{2nqr}\sum_{l=1}^{r}\|R^l\|_{S^{-1}}^2 + \frac{1}{2n}\operatorname{Tr}(S) \\ & + \lambda\left\|\mathrm{B}\right\|_{2,1} + \frac{\lambda}{r}\sum_{l=1}^{r}\left\langle\Theta^l, Y^l - X\mathrm{B} - R^l\right\rangle \\ = \max_{\Theta^1,\dots,\Theta^r\in\mathbb{R}^{n\times q}} \Bigg(\min_{S\succeq\underline{\sigma}\,\mathrm{Id}_n} & \frac{1}{r}\sum_{l=1}^{r}\min_{R^l\in\mathbb{R}^{n\times q}}\Big(\frac{\|R^l\|_{S^{-1}}^2}{2nq} - \langle\Theta^l, R^l\rangle\Big) + \frac{1}{2n}\operatorname{Tr}(S) \\ & + \lambda\min_{\mathrm{B}\in\mathbb{R}^{p\times q}}\Big(\left\|\mathrm{B}\right\|_{2,1} - \langle\bar{\Theta}, X\mathrm{B}\rangle\Big) + \frac{\lambda}{r}\sum_{l=1}^{r}\langle\Theta^l, Y^l\rangle\Bigg)\ . \end{aligned} \tag{4.25}$$

Moreover we have:

$$\min_{R^l\in\mathbb{R}^{n\times q}}\left(\frac{\|R^l\|_{S^{-1}}^2}{2nq} - \langle\Theta^l, R^l\rangle\right) = -\frac{nq\lambda^2}{2}\left\langle\Theta^l{\Theta^l}^\top, S\right\rangle \tag{4.26}$$

$$\min_{\mathrm{B}\in\mathbb{R}^{p\times q}}\Big(\left\|\mathrm{B}\right\|_{2,1} - \langle\bar{\Theta}, X\mathrm{B}\rangle\Big) = -\max\Big(\langle X^\top\bar{\Theta}, \mathrm{B}\rangle - \left\|\mathrm{B}\right\|_{2,1}\Big) = -\iota_{\mathcal{B}_{2,\infty}}(X^\top\bar{\Theta})\ . \tag{4.27}$$

This leads to:

$$\begin{aligned} d^* = \max_{\Theta^1,\dots,\Theta^r\in\mathbb{R}^{n\times q}} \min_{S\succeq\underline{\sigma}\,\mathrm{Id}_n} & -\frac{1}{r}\sum_{l=1}^{r}\tfrac{nq\lambda^2}{2}\langle\Theta^l{\Theta^l}^\top, S\rangle - \lambda\iota_{\mathcal{B}_{2,\infty}}(X^\top\bar{\Theta}) + \frac{\operatorname{Tr}(S)}{2n} \\ & + \frac{\lambda}{r}\sum_{l=1}^{r}\langle\Theta^l, Y^l\rangle \\ = \max_{\Theta^1,\dots,\Theta^r\in\mathbb{R}^{n\times q}} \frac{1}{2n}\min_{S\succeq\underline{\sigma}\,\mathrm{Id}_n} & \Big(\langle\mathrm{Id}_n, S\rangle - \frac{qn^2\lambda^2}{r}\sum_{l=1}^{r}\langle\Theta^l{\Theta^l}^\top, S\rangle\Big) - \lambda\iota_{\mathcal{B}_{2,\infty}}(X^\top\bar{\Theta}) \\ & + \frac{\lambda}{r}\sum_{l=1}^{r}\langle\Theta^l, Y^l\rangle \\ = \max_{\Theta^1,\dots,\Theta^r\in\mathbb{R}^{n\times q}} \frac{1}{2n}\min_{S\succeq\underline{\sigma}\,\mathrm{Id}_n} & \Big\langle\mathrm{Id}_n - \tfrac{qn^2\lambda^2}{r}\sum_{l=1}^{r}\Theta^l{\Theta^l}^\top, S\Big\rangle - \lambda\iota_{\mathcal{B}_{2,\infty}}(X^\top\bar{\Theta}) \\ & + \frac{\lambda}{r}\sum_{l=1}^{r}\langle\Theta^l, Y^l\rangle\ . \end{aligned} \tag{4.28}$$

$$\begin{aligned} \min_{S\succeq\underline{\sigma}\,\mathrm{Id}_n}\langle\mathrm{Id}_n - \tfrac{qn^2\lambda^2}{r}\sum_{l=1}^{r}\Theta^l{\Theta^l}^\top, S\rangle & \\ = \begin{cases} \left\langle\mathrm{Id}_n - \tfrac{qn^2\lambda^2}{r}\sum_{l=1}^{r}\Theta^l{\Theta^l}^\top, \underline{\sigma}\,\mathrm{Id}_n\right\rangle\ , & \text{if } \mathrm{Id}_n \succeq \tfrac{qn^2\lambda^2}{r}\sum_{l=1}^{r}\Theta^l{\Theta^l}^\top\ , \\ -\infty\ , & \text{otherwise.} \end{cases} \end{aligned} \tag{4.29}$$

It follows that the dual problem of CLaR is:

$$\max_{(\Theta^1,\dots,\Theta^r)\in\Delta_{X,\lambda}} \frac{\sigma}{2}\Big(1 - \frac{qn\lambda^2}{r}\sum_{l=1}^{r}\operatorname{Tr}\Theta^l\Theta^{l\top}\Big) + \frac{\lambda}{r}\sum_{l=1}^{r}\left\langle \Theta^l, Y^l\right\rangle \ , \tag{4.30}$$

where $\Delta_{X,\lambda} \triangleq \left\{(\Theta^1,\dots,\Theta^r)\in\mathbb{R}^{n\times q\times r} : \|X^\top\bar{\Theta}\|_{2,\infty}\le 1, \|\sum_{l=1}^{r}\Theta^l\Theta^{l\top}\|\le\frac{r}{\lambda^2n^2q}\right\}$. ■

The link equation provides a natural way to construct a dual feasible point from any pair (B, S) in the iterations of Algorithm 4.1. The dual point Θ at iteration t is obtained through a residual rescaling similar to the "naive" procedure detailed in Part I, *i.e.*, $\Theta^l = \frac{1}{nq\lambda}(Y^l - X\mathrm{B})$ (with B the current primal iterate); then the dual point is rescaled to lie in $\Delta_{X,\lambda}$. This strategy would be amenable to the extrapolation improvements of Chapter 3.

Remark 4.14. *Equations (4.9) and (4.10) make it straightforward to compute S^{-1} and $\operatorname{Tr} S$, which we rather store than S for computational efficiency in Algorithm 4.1. At every update of S, it is also beneficial to precompute $S^{-1}X$ and $S^{-1}R$: maintaining $S^{-1}R$ rather than R avoids multiplication by S^{-1} at every BCD step.*

Remark 4.15. *Similarly to the Concomitant Lasso, and contrary to the Lasso, CLaR is equivariant under scaling of the response, in the following sense. Consider the transformation:*

$$Y' = \alpha Y,\ \mathrm{B}' = \alpha\mathrm{B},\ S' = \alpha S, \quad (\alpha > 0) \ ,$$

which leaves Models (4.4) and (4.5) invariant. Then one can check that the solutions of Problem (4.7) are multiplied by the same factor: $\hat{\mathrm{B}}' = \alpha\hat{\mathrm{B}}$ and $\hat{S}' = \alpha\hat{S}$.

4.4 An analysis of CLaR through smoothing

The purpose of this section is to shed some light on the origin of the data-fitting term $\|Y - X\mathrm{B}\|_{S^{-1}}$ used in CLaR and SGCL. This datafitting term was introduced empirically, as a way to generalize concomitant estimation to noise covariance estimation. However, after publication of our works, we became aware of closely related estimators.

Definition 4.16 (Multivariate square-root Lasso)**.** *van de Geer (2016) introduced the Multivariate square-root Lasso estimator, a solution of:*

$$\operatorname*{arg\,min}_{\mathrm{B}\in\mathbb{R}^{p\times q}} \frac{1}{\sqrt{n}}\|Y - X\mathrm{B}\|_{\mathscr{S},1} + \lambda\sum_{i,j}|\mathrm{B}_{i,j}| \ . \tag{4.31}$$

The penalty used on B *is not crucial to this estimator, and can be replaced by $\|\mathrm{B}\|_{2,1}$ to fit our framework without loss of interest.*

As noted by van de Geer (2016, Lemma 3.4), this estimator is also a solution of:

$$\operatorname*{arg\,min}_{\mathrm{B}\in\mathbb{R}^{p\times q},\Sigma\in\mathcal{S}^n_{++}} \frac{1}{n}\|Y - X\mathrm{B}\|_{\Sigma^{-1/2}} + \operatorname{Tr}\Sigma^{1/2} + \lambda\sum_{i,j}|\mathrm{B}_{i,j}| \ , \tag{4.32}$$

provided that the minimum is indeed attained at some $\hat{\Sigma}\in\mathcal{S}^n_{++}$. The statistical analysis of this estimator, also pursued by Molstad (2019); Stucky (2017), is based on this assumption. It suffers from two drawbacks:

- the optimization problem is difficult to solve. Recently, Molstad (2019) proposed two algorithms to solve it: a prox-linear ADMM, and accelerated proximal gradient descent, the latter lacking convergence guarantees since the composite objective has two non-smooth terms. Before that, van de Geer and Stucky (2016) devised a fixed point method, lacking descent guarantees.
- the noise covariance estimate $\hat{\Sigma}$ is never full rank when $q < n$, and even in cases where $q \geq n$, most likely because of the trace penalty, $\hat{\Sigma}$ quickly becomes rank deficient as λ decreases.

As it turns out, our approach can be interpreted as a *smoothing* (recall Proposition 1.9 and Figure 1.3 page 28) technique to replace the non-smooth datafitting term of Problem (4.31) by a smooth approximation.

We start by introducing some elements of smoothing theory (Nesterov, 2005; Beck and Teboulle, 2012) for generic Schatten norms, then focus on the nuclear norm.

4.4.1 Smoothing of Schatten norms

In all this section, the variable is a matrix $Z \in \mathbb{R}^{n \times q}$ unless in the multiple repetition case. To smooth a function, a standard approach is to convolve it with a smooth function (which, as in Figure 1.3, amounts to adding a strongly convex term to its Fenchel transform, then taking the Fenchel conjugate again). Here, the smoothing function we use is an isotropic parabola, and the function we smooth is a generic Schatten norm.

Proposition 4.17. *Let $\omega : \mathbb{R}^d \to \mathbb{R}$ be a convex function with Lipschitz gradient. Let $\underline{\sigma} > 0$ and $\omega_{\underline{\sigma}} \triangleq \underline{\sigma}\omega\left(\frac{\cdot}{\underline{\sigma}}\right)$. Then:*

$$\omega(\cdot) = \tfrac{1}{2}\|\cdot\|^2 + \tfrac{1}{2} \implies \omega_{\underline{\sigma}}^* = \tfrac{\underline{\sigma}}{2}\|\cdot\|^2 - \tfrac{\underline{\sigma}}{2} \ . \tag{4.33}$$

Proof The proof is a direct application of the results on frequently used Fenchel transforms introduced in Equations (1.39), (1.41) and (1.42). ∎

Lemma 4.18. *Let $c \in \mathbb{R}$, $p \in [1, +\infty]$. Let $p^* \in [1, +\infty]$ be the Hölder conjugate of p, satisfying $\frac{1}{p} + \frac{1}{p^*} = 1$. For the choice $\omega = \frac{1}{2}\|\cdot\|^2 + c$:*

$$\left(\|\cdot\|_{\mathscr{S},p} \square \omega_{\underline{\sigma}}\right)(Z) = \frac{1}{2\underline{\sigma}}\left\|Z\right\|^2 + c\underline{\sigma} - \frac{\underline{\sigma}}{2}\left\|\Pi_{\mathcal{B}_{\mathscr{S},p^*}}\left(\frac{Z}{\underline{\sigma}}\right) - \frac{Z}{\underline{\sigma}}\right\|^2 \ .$$

Proof

$$\begin{aligned}
\left(\|\cdot\|_{\mathscr{S},p} \square \omega_{\underline{\sigma}}\right)(Z) &= \left(\|\cdot\|_{\mathscr{S},p} \square \omega_{\underline{\sigma}}\right)^{**}(Z) && \text{(using Equation (1.37))} \\
&= \left(\|\cdot\|_{\mathscr{S},p}^* + \omega_{\underline{\sigma}}^*\right)^*(Z) && \text{(using Equation (1.38))} \\
&= \left(\iota_{\mathcal{B}_{\mathscr{S},p^*}} + \tfrac{\underline{\sigma}}{2}\|\cdot\|^2 - c\underline{\sigma}\right)^*(Z) && \text{(using Equations (1.40) and (4.33))} \\
&= \left(\tfrac{\underline{\sigma}}{2}\|\cdot\|^2 + \iota_{\mathcal{B}_{\mathscr{S},p^*}}\right)^*(Z) + c\underline{\sigma} && \text{(using Equation (1.41))} \ .
\end{aligned} \tag{4.34}$$

We can now compute the last Fenchel transform remaining:

$$\begin{aligned}
\left(\frac{\sigma}{2}\left\|\cdot\right\|^2 + \iota_{\mathcal{B}_{\mathscr{S},p^*}}\right)^*(Z) &= \sup_{U\in\mathbb{R}^{n\times q}}\left(\langle U, Z\rangle - \tfrac{\sigma}{2}\left\|U\right\|^2 - \iota_{\mathcal{B}_{\mathscr{S},p^*}}(U)\right) \\
&= \sup_{U\in\mathcal{B}_{\mathscr{S},p^*}}\left(\langle U, Z\rangle - \tfrac{\sigma}{2}\left\|U\right\|^2\right) \\
&= -\underline{\sigma}\cdot\inf_{U\in\mathcal{B}_{\mathscr{S},p^*}}\left(\tfrac{1}{2}\left\|U\right\|^2 - \langle U, \tfrac{Z}{\underline{\sigma}}\rangle\right) \\
&= -\underline{\sigma}\cdot\inf_{U\in\mathcal{B}_{\mathscr{S},p^*}}\left(\tfrac{1}{2}\left\|U - \tfrac{Z}{\underline{\sigma}}\right\|^2 - \tfrac{1}{2\underline{\sigma}^2}\left\|Z\right\|^2\right) \\
&= \tfrac{1}{2\underline{\sigma}}\left\|Z\right\|^2 - \tfrac{\sigma}{2}\cdot\inf_{U\in\mathcal{B}_{\mathscr{S},p^*}}(\left\|U - \tfrac{Z}{\underline{\sigma}}\right\|^2 \\
&= \tfrac{1}{2\underline{\sigma}}\left\|Z\right\|^2 - \tfrac{\sigma}{2}\left\|\Pi_{\mathcal{B}_{\mathscr{S},p^*}}\left(\tfrac{Z}{\underline{\sigma}}\right) - \tfrac{Z}{\underline{\sigma}}\right\|^2 \quad . \qquad (4.35)
\end{aligned}$$

The result follows by combining Equations (4.34) and (4.35). ■

We detail the result of Lemma 4.18 for peculiar values of p in the next propositions.

Proposition 4.19 (Schatten 2-norm (Frobenius norm))**.** *For the choice* $\omega = \frac{1}{2}\left\|\cdot\right\|^2 + \frac{1}{2}$*, and for* $Z \in \mathbb{R}^{n\times q}$ *then:*

$$\left(\left\|\cdot\right\| \square\, \omega_{\underline{\sigma}}\right)(Z) = \min_{\sigma\geq\underline{\sigma}}\left(\tfrac{1}{2\sigma}\left\|Z\right\|^2 + \tfrac{\sigma}{2}\right) = \begin{cases} \frac{\left\|Z\right\|^2}{2\underline{\sigma}} + \frac{\underline{\sigma}}{2} \ , & \text{if } \left\|Z\right\| \leq \underline{\sigma} \ , \\ \left\|Z\right\| \ , & \text{if } \left\|Z\right\| > \underline{\sigma} \ . \end{cases} \qquad (4.36)$$

Notice that this is simply the Huber function (with parameter $\underline{\sigma}$*) applied to* $\left\|Z\right\|$*.*

Proof Let us recall that $\left\|\cdot\right\| = \left\|\cdot\right\|_{\mathscr{S},2}$. Therefore:

$$\Pi_{\mathcal{B}_{\mathscr{S},2}}\left(\frac{Z}{\underline{\sigma}}\right) = \begin{cases} \frac{Z}{\underline{\sigma}} \ , & \text{if } \left\|Z\right\| \leq \underline{\sigma} \ , \\ \frac{Z}{\left\|Z\right\|} \ , & \text{if } \left\|Z\right\| > \underline{\sigma} \ . \end{cases} \qquad (4.37)$$

By combining Equation (4.37) and lemma 4.18 with $p = p^* = 2$, and $c = \frac{1}{2}$, the later yields:

$$\left(\left\|\cdot\right\| \square\, \omega_{\underline{\sigma}}\right)(Z) = \begin{cases} \frac{1}{2\underline{\sigma}}\left\|Z\right\|^2 + \frac{\underline{\sigma}}{2} \ , & \text{if } \left\|Z\right\| \leq \underline{\sigma} \ , \\ \left\|Z\right\| \ , & \text{if } \left\|Z\right\| > \underline{\sigma} \ . \end{cases}$$

■

Proposition 4.20 (Schatten infinity-norm (spectral norm))**.** *For the choice* $\omega = \frac{1}{2}\left\|\cdot\right\|^2 + \frac{1}{2}$ *and for* $Z \in \mathbb{R}^{n\times q}$ *with singular value decomposition* $V\,\mathrm{diag}(\gamma_1, \ldots, \gamma_{n\vee q})W^\top$*, then:*

$$\left(\left\|\cdot\right\|_{\mathscr{S},\infty} \square\, \omega_{\underline{\sigma}}\right)(Z) = \begin{cases} \frac{1}{2\underline{\sigma}}\left\|Z\right\|^2 + \frac{\underline{\sigma}}{2} \ , & \text{if } \left\|Z\right\|_{\mathscr{S},1} \leq 1 \ , \\ \frac{\underline{\sigma}}{2}\sum_{i=1}^{n\wedge q}\left(\frac{\gamma_i^2}{\underline{\sigma}^2} - \nu^2\right)_+ + \frac{\underline{\sigma}}{2} \ , & \text{if } \left\|Z\right\|_{\mathscr{S},1} > 1 \ , \end{cases}$$

where $\nu \geq 0$ *is defined by the implicit equation:*

$$\left\|\left(\mathrm{ST}\left(\tfrac{\gamma_1}{\underline{\sigma}}, \nu\right), \ldots, \mathrm{ST}\left(\tfrac{\gamma_{n\wedge q}}{\underline{\sigma}}, \nu\right)\right)\right\|_1 = 1 \ . \qquad (4.38)$$

Proof The Hölder conjugate of $p = +\infty$ is $p^* = 1$. We remind that $\Pi_{\mathcal{B}_{\mathscr{S},1}}$, the projection over $\mathcal{B}_{\mathscr{S},1}$, is given by (Beck, 2017, Example 7.31):

$$\Pi_{\mathcal{B}_{\mathscr{S},1}}\left(\frac{Z}{\underline{\sigma}}\right) = \begin{cases} \frac{Z}{\underline{\sigma}} \ , & \text{if } \left\|Z\right\|_{\mathscr{S},1} \leq \underline{\sigma} \ , \\ V \operatorname{diag}\Big(\operatorname{ST}(\frac{\gamma_i}{\underline{\sigma}}, \gamma)\Big) W^\top \ , & \text{if } \left\|Z\right\|_{\mathscr{S},1} > \underline{\sigma} \ , \end{cases} \tag{4.39}$$

the scalar γ being defined by the implicit equation:

$$\left\| \Big(\operatorname{ST}(\tfrac{\gamma_1}{\underline{\sigma}}, \gamma), \dots, \operatorname{ST}(\tfrac{\gamma_{n\wedge q}}{\underline{\sigma}}, \gamma) \Big) \right\|_1 = 1 \ . \tag{4.40}$$

By combining Equation (4.37) and Lemma 4.18 (with $p^* = \infty, c = \frac{1}{2}$) it follows that:

$$(\|\cdot\| \square \omega_{\underline{\sigma}})(Z) = \begin{cases} \frac{1}{2\underline{\sigma}} \left\|Z\right\|^2 + \frac{\underline{\sigma}}{2} \ , & \text{if } \left\|Z\right\|_{\mathscr{S},1} \leq \underline{\sigma} \ , \\ \frac{1}{2\underline{\sigma}} \left\|Z\right\|^2 + \frac{\underline{\sigma}}{2} - \frac{\underline{\sigma}}{2} \left\| \Pi_{\mathcal{B}_{\mathscr{S},1}}\left(\frac{Z}{\underline{\sigma}}\right) - \frac{Z}{\underline{\sigma}} \right\|^2 \ , & \text{if } \left\|Z\right\|_{\mathscr{S},1} > \underline{\sigma} \ . \end{cases} \tag{4.41}$$

Let us compute $\left\| \Pi_{\mathcal{B}_{\mathscr{S},1}}\left(\frac{Z}{\underline{\sigma}}\right) - \frac{Z}{\underline{\sigma}} \right\|^2$. If $\left\|Z\right\|_{\mathscr{S},1} > \underline{\sigma}$ we have:

$$\begin{aligned} \left\| \Pi_{\mathcal{B}_{\mathscr{S},1}}\left(\frac{Z}{\underline{\sigma}}\right) - \frac{Z}{\underline{\sigma}} \right\|^2 &= \frac{1}{\underline{\sigma}^2} \left\| V \operatorname{diag}\Big((\gamma_i - \nu\underline{\sigma})_+ - \gamma_i \Big) W^\top \right\|^2 \qquad \text{(using Equation (4.39))} \\ &= \frac{1}{\underline{\sigma}^2} \sum_{i=1}^{n\wedge q} \Big((\gamma_i - \nu\underline{\sigma})_+ - \gamma_i \Big)^2 \\ &= \frac{1}{\underline{\sigma}^2} \Big(\sum_{\gamma_i \geq \nu\underline{\sigma}} \gamma^2 \underline{\sigma}^2 + \sum_{\gamma_i < \nu\underline{\sigma}} \gamma_i^2 \Big) \ . \end{aligned} \tag{4.42}$$

By plugging Equation (4.42) into Equation (4.41) it follows that if $\left\|Z\right\|_{\mathscr{S},1} > \underline{\sigma}$,

$$\begin{aligned} \Big(\|\cdot\| \square \omega_{\underline{\sigma}} \Big)(Z) &= \frac{1}{2\underline{\sigma}} \sum_{i=1} \gamma_i^2 + \frac{\underline{\sigma}}{2} - \frac{1}{2\underline{\sigma}} \sum_{\gamma_i \geq \nu\underline{\sigma}} \nu^2 \underline{\sigma}^2 - \frac{1}{2\underline{\sigma}} \sum_{\gamma_i < \nu\underline{\sigma}} \gamma_i^2 \\ &= \frac{1}{2\underline{\sigma}} \sum_{\gamma_i \geq \nu\underline{\sigma}} \Big(\gamma_i^2 - \nu^2 \underline{\sigma}^2 \Big) + \frac{\underline{\sigma}}{2} \\ &= \frac{\underline{\sigma}}{2} \sum_{i=1}^{n\wedge q} \Big(\frac{\gamma_i^2}{\underline{\sigma}^2} - \nu^2 \Big)_+ + \frac{\underline{\sigma}}{2} \ . \end{aligned} \tag{4.43}$$

Proposition 4.20 follows by plugging Equation (4.43) for the case $\left\|Z\right\|_{\mathscr{S},1} > \underline{\sigma}$, and the fact that when $\left\|Z\right\|_{\mathscr{S},1} \leq \underline{\sigma}$ the result is straightforward. ∎

Remark 4.21. *Since* $\nu \mapsto \left\| \Big(\operatorname{ST}(\frac{\gamma_1}{\underline{\sigma}}, \nu), \dots, \operatorname{ST}(\frac{\gamma_{n\wedge q}}{\underline{\sigma}}, \nu) \Big) \right\|_1$ *is decreasing and piecewise linear, the solution of Equation (4.38) can be computed exactly in* $\mathcal{O}(n \wedge q \ \log(n \wedge q))$ *operations.*

We have a particular interest in the Schatten 1-norm.

4.4.2 Smoothing of the nuclear norm

The next propositions are key to our framework and show the connection between the SGCL, CLaR and the Schatten 1-norm used in the Multivariate square-root Lasso. First, we derive a formula for the smoothing of this norm. Let us define the following smoothing function:

$$\omega_{\underline{\sigma}} \triangleq \frac{1}{2}\left(\|\cdot\|^2 + n \wedge q\right)\underline{\sigma} \ . \tag{4.44}$$

Proposition 4.22. *The $\omega_{\underline{\sigma}}$-smoothing of the Schatten 1-norm,* i.e., $\|\cdot\|_{\mathscr{S},1} \square \omega_{\underline{\sigma}}$, *is the solution of the following smooth optimization problem:*

$$(\|\cdot\|_{\mathscr{S},1} \square \omega_{\underline{\sigma}})(Z) = \min_{S \succeq \underline{\sigma} \mathrm{Id}_n} \tfrac{1}{2}\left\|Z\right\|_{S^{-1}}^2 + \tfrac{1}{2}\operatorname{Tr}(S) \ . \tag{4.45}$$

Proof Let $V \operatorname{diag}(\gamma_1, \ldots, \gamma_{n\wedge q}) W^\top$ be the singular value decomposition of Z. We remind that $\Pi_{\mathcal{B}_{\mathscr{S},\infty}}$, the projection over the unit ball of the Schatten ∞-norm $\mathcal{B}_{\mathscr{S},\infty}$, is given by (see Beck 2017, Example 7.31):

$$\begin{aligned}\Pi_{\mathcal{B}_{\mathscr{S},\infty}}\left(\frac{Z}{\underline{\sigma}}\right) &= V \operatorname{diag}\left(\Pi_{\mathcal{B}_\infty}\left(\tfrac{\gamma_1}{\underline{\sigma}}, \ldots, \tfrac{\gamma_{n\wedge q}}{\underline{\sigma}}\right)\right) W^\top \\ &= V \operatorname{diag}\left(\tfrac{\gamma_1}{\underline{\sigma}} \wedge 1, \ldots, \tfrac{\gamma_{n\wedge q}}{\underline{\sigma}} \wedge 1\right) W^\top \ ,\end{aligned} \tag{4.46}$$

where we used that $\gamma_i/\underline{\sigma} \geq 0$. Then we have:

$$\begin{aligned}\left\|\Pi_{\mathcal{B}_{\mathscr{S},\infty}}\left(\frac{Z}{\underline{\sigma}}\right) - \frac{Z}{\underline{\sigma}}\right\|^2 &\overset{(4.46)}{=} \left\|V \operatorname{diag}\left(\tfrac{\gamma_1}{\underline{\sigma}} \wedge 1 - \tfrac{\gamma_1}{\underline{\sigma}}, \ldots, \tfrac{\gamma_{n\wedge q}}{\underline{\sigma}} \wedge 1 - \tfrac{\gamma_{n\wedge q}}{\underline{\sigma}}\right) W^\top\right\|^2 \\ &= \sum_{i=1}^{n\wedge q}\left(\tfrac{\gamma_i}{\underline{\sigma}} \wedge 1 - \tfrac{\gamma_i}{\underline{\sigma}}\right)^2 \\ &= \frac{1}{\underline{\sigma}^2}\sum_{i=1}^{n\wedge q}\left(\gamma_i \wedge \underline{\sigma} - \gamma_i\right)^2 \ .\end{aligned} \tag{4.47}$$

By combining Equation (4.47) and lemma 4.18 with $p^* = \infty, c = \frac{n\wedge q}{2}$, the later yields:

$$\left(\|\cdot\|_{\mathscr{S},1} \square \omega_{\underline{\sigma}}\right)(Z) = (n \wedge q)\frac{\underline{\sigma}}{2} + \frac{1}{2\underline{\sigma}}\sum_{\gamma_i \leq \underline{\sigma}} \gamma_i^2 - \frac{1}{2}\sum_{\gamma_i \geq \underline{\sigma}} \underline{\sigma} + \sum_{\gamma_i \geq \underline{\sigma}} \gamma_i \ . \tag{4.48}$$

Moreover:

$$\begin{aligned}\min_{S \succeq \underline{\sigma}\mathrm{Id}_n} \frac{1}{2}\operatorname{Tr}[Z^\top S^{-1} Z] + \frac{1}{2}\operatorname{Tr}(S) &= \frac{1}{2}\sum_{i=1}^{n\wedge q}\frac{\gamma_i^2}{\gamma_i \vee \underline{\sigma}} + \frac{1}{2}\sum_{i=1}^{n\wedge q}\gamma_i \vee \underline{\sigma} \\ &= \frac{1}{2\underline{\sigma}}\sum_{\gamma_i \leq \underline{\sigma}}\gamma_i^2 + \frac{1}{2}\sum_{\gamma_i \geq \underline{\sigma}}\gamma_i + \frac{1}{2}\sum_{\gamma_i \leq \underline{\sigma}}\underline{\sigma} + \frac{1}{2}\sum_{\gamma_i \geq \underline{\sigma}}\gamma_i \\ &= \frac{1}{2\underline{\sigma}}\sum_{\gamma_i \leq \underline{\sigma}}\gamma_i^2 + \sum_{\gamma_i \geq \underline{\sigma}}\gamma_i + \frac{1}{2}\sum_{\gamma_i \leq \underline{\sigma}}\underline{\sigma} + (n \wedge q)\tfrac{\underline{\sigma}}{2} - (n \wedge q)\tfrac{\underline{\sigma}}{2} \\ &= \frac{1}{2\underline{\sigma}}\sum_{\gamma_i \leq \underline{\sigma}}\gamma_i^2 + \sum_{\gamma_i \geq \underline{\sigma}}\gamma_i + (n \wedge q)\frac{\underline{\sigma}}{2} - \frac{1}{2}\sum_{\gamma_i \geq \underline{\sigma}}\underline{\sigma} \ ,\end{aligned} \tag{4.49}$$

and identifying Equations (4.48) and (4.49) leads to the result. ■

It is now easy to generalize this result to a matrix Z containing stacked residuals.

Proposition 4.23 (Schatten 1-norm (nuclear norm) with repetitions)**.** *Let $Z^1, \dots, Z^r$ be matrices in $\mathbb{R}^{n\times q}$, then we define $Z \in \mathbb{R}^{n\times qr}$ by $Z \triangleq [Z^1|\dots|Z^r]$. For the choice $\omega(Z) = \frac{1}{2}\left\|Z\right\|^2 + \frac{n\wedge qr}{2}$, then:*

$$\left(\|\cdot\|_{\mathscr{S},1} \square\, \omega_{\underline{\sigma}}\right)(Z) = \min_{S \succeq \underline{\sigma}\,\mathrm{Id}_n} \frac{1}{2}\sum_{l=1}^{r} \mathrm{Tr}\left(Z^{l\top} S^{-1} Z^l\right) + \frac{1}{2}\mathrm{Tr}(S) \ . \tag{4.50}$$

Proof The result is a direct application of Proposition 4.22, with $Z = [Z^{(1)}|\dots|Z^{(r)}]$. It suffices to notice that $\mathrm{Tr}\, Z^\top S^{-1} Z = \sum_{l=1}^r \mathrm{Tr}\left(Z^{l\top} S^{-1} Z^l\right)$. ■

Proposition 4.24. *Any solution of Problem (4.7), $(\hat{\mathrm{B}}, \hat{S}) = (\hat{\mathrm{B}}^{\mathrm{CLaR}}, \hat{S}^{\mathrm{CLaR}})$ is also a solution of:*

$$\hat{\mathrm{B}} = \operatorname*{arg\,min}_{\mathrm{B}\in\mathbb{R}^{p\times q}} \left(\|\cdot\|_{\mathscr{S},1} \square\, \omega_{\underline{\sigma}}\right)(Z) + \lambda n \left\|\mathrm{B}\right\|_{2,1}$$
$$\hat{S} = \mathrm{ClSqrt}\left(\tfrac{1}{rq} ZZ^\top, \underline{\sigma}\right) \ , \ \textit{where } Z = [Y^1 - X\mathrm{B}|\dots|Y^r - X\mathrm{B}].$$

Proof Proposition 4.24 follows from Proposition 4.23 by choosing:

$$Z = \tfrac{1}{\sqrt{rq}}[Y^1 - X\mathrm{B}|\dots|Y^r - X\mathrm{B}] \ , \tag{4.51}$$

and by taking the arg min over B. ■

As mentioned above, properties similar to Proposition 4.24 can be traced back to van de Geer (2016, Lemma 3.4), where the following variational formulation was used to prove oracle inequalities for the multivariate square-root Lasso: if $ZZ^\top \succ \mathbf{0}_{n,n}$,

$$\left\|Z\right\|_{\mathscr{S},1} = \min_{S\in\mathcal{S}^n_{++}} \frac{1}{2}\left\|Z\right\|^2_{S^{-1}} + \frac{1}{2}\mathrm{Tr}(S) \ . \tag{4.52}$$

In other words Proposition 4.24 generalizes van de Geer (2016, Lemma 3.4) for all matrices Z, getting rid of the condition $ZZ^\top \succ \mathbf{0}_{n,n}$. Our formulation in Proposition 4.22 is motivated by computational aspects, as it helps to address the combined non-differentiability of the data-fitting term $\|\cdot\|_{\mathscr{S},1}$ and the penalty $\|\cdot\|_{2,1}$ term. Other alternatives to exploit the multiple repetitions without simply averaging them, could consist in investigating other Schatten p-norms:

$$\operatorname*{arg\,min}_{\mathrm{B}\in\mathbb{R}^{p\times q}} \tfrac{1}{\sqrt{rq}}\left\|[Y^1 - X\mathrm{B}|\dots|Y^r - X\mathrm{B}]\right\|_{\mathscr{S},p} + \lambda n \left\|\mathrm{B}\right\|_{2,1} \ . \tag{4.53}$$

Without smoothing, problems of the form given in Equation (4.53) have the drawback of having two non-differentiable terms, and calling for primal-dual algorithms (Chambolle and Pock, 2011) with costly proximal operators. Even if the non-smooth Schatten 1-norm is replaced by the formula in (4.52), numerical challenges remain: S can approach 0 arbitrarily, hence, the gradient *w.r.t.* S of the data-fitting term is not Lipschitz over the optimization domain. A similar problem was raised for the concomitant Lasso by Ndiaye et al. (2017a) who used smoothing techniques to address it. Here we replaced the nuclear norm by its smoothed version $\|\cdot\|_{\mathscr{S},1} \square\, \omega_{\underline{\sigma}}$.

4.5 Conclusion

In this chapter, we have proposed the SGCL and CLaR, two new sparse regression estimators designed to deal with heterogeneous observations coming from different origins and corrupted by different levels of noise – in the context of repeated observations for CLaR. Despite the joint estimation of the regression coefficients as well as the noise level, the problem considered is jointly convex, thus guaranteeing global convergence which one can check by duality gap certificates. The resulting optimization problem can be solved efficiently with state-of-the-art convex solvers, and the algorithmic cost is the same as for single repetition data. The theory of smoothing connects CLaR to the Schatten 1-Lasso in a principled manner, which opens the way to the use of more sophisticated datafitting terms.

In the next chapter, we evaluate the benefits of CLaR for support recovery in heteroscedastic context against a large number of competitors, both on simulations and on empirical MEG data.

5

Experimental validation of smoothed concomitant estimation

Tous les fleuves se jettent dans
la mer et la mer ne regorge pas.

Contents

5.1 Alternative estimators . . . 103
5.1.1 Multi-Task Lasso (MTL) . . . 104
5.1.2 $\ell_{2,1}$-Maximum Likelihood ($\ell_{2,1}$-MLE) . . . 104
5.1.3 Multivariate Regression with Covariance Estimation (MRCE) . . . 105
5.1.4 Algorithms summary . . . 106
5.2 CLaR . . . 107
5.2.1 Right auditory stimulations . . . 112
5.2.2 Left visual stimulations . . . 113
5.2.3 Right visual stimulations . . . 113
5.3 Preprocessing steps for realistic and real data . . . 114
5.4 Time comparison . . . 115

In this chapter, the practical benefits of convex and smooth concomitant estimation for multitask regression are demonstrated on toy datasets, realistic simulated data and real neuroimaging data.

Section 5.1 presents the alternative approaches to heteroscedastic multitask regression. Empirical properties of CLaR are highlighted in Section 5.2, where the estimators performance on real data is also evaluated.

5.1 Alternative estimators

We compare CLaR (Bertrand et al., 2019) to several estimators: SGCL (**Massias** et al., 2018a), the (smoothed) $\ell_{2,1}$-Maximum Likelihood ($\ell_{2,1}$-MLE) and a version of the $\ell_{2,1}$-MLE with multiple repetitions ($\ell_{2,1}$-MLER), an $\ell_{2,1}$ penalized version of the Multivariate Regression with Covariance Estimation (Rothman et al. 2010, $\ell_{2,1}$-MRCE), a version of $\ell_{2,1}$-MRCE with repetitions ($\ell_{2,1}$-MRCER) and the Multi-Task Lasso (Obozinski et al. 2010, MTL). The cost of an epoch of block coordinate descent and the cost of computing the duality gap for each algorithm are summarized in Table 5.1. The updates of each algorithms are summarized in Table 5.2.

CLaR and SGCL were introduced in the previous chapter. Let us first introduce the definitions of the alternative estimation procedures.

5.1.1 Multi-Task Lasso (MTL)

The Multitask Lasso (Obozinski et al., 2010) is the classical sparse multitask estimator used when the additive noise is supposed to be homoscedastic (without correlation). It is obtained by solving:

$$\hat{\mathrm{B}}^{\mathrm{MTL}} \in \underset{\mathrm{B}\in\mathbb{R}^{p\times q}}{\arg\min} \frac{1}{2nq}\|\bar{Y} - X\mathrm{B}\|^2 + \lambda\|\mathrm{B}\|_{2,1} \ . \tag{5.1}$$

Remark 5.1. *It can be seen that trying to use all the repetitions in the MTL leads to MTL itself because* $\|\bar{Y} - X\mathrm{B}\|^2 = \frac{1}{r}\sum_l \|Y^l - X\mathrm{B}\|^2$ *up to constant terms in* B.

5.1.2 $\ell_{2,1}$-Maximum Likelihood ($\ell_{2,1}$-MLE)

Here we study a penalized Maximum Likelihood Estimator (Chen and Banerjee, 2017) ($\ell_{2,1}$-MLE). When minimizing $\ell_{2,1}$-Maximum Likelihood the natural parameters of the problem are the regression coefficients B and the precision matrix Σ^{-1}. Since real M/EEG covariance matrices are not full rank, one has to be algorithmically careful when Σ becomes singular. To avoid such numerical errors and to be consistent with the smoothed estimator proposed in Chapter 4, let us define the (smoothed) $\ell_{2,1}$-MLE as:

$$(\hat{\mathrm{B}}^{\ell_{2,1}-\mathrm{MLE}}, \hat{\Sigma}^{\ell_{2,1}-\mathrm{MLE}}) \in \underset{\substack{\mathrm{B}\in\mathbb{R}^{p\times q}\\ \Sigma\succeq\underline{\sigma}^2/r^2\,\mathrm{Id}_n}}{\arg\min} \left\|\bar{Y} - X\mathrm{B}\right\|^2_{\Sigma^{-1}} - \log\det(\Sigma^{-1}) + \lambda\left\|\mathrm{B}\right\|_{2,1} \ , \tag{5.2}$$

and its repetitions version ($\ell_{2,1}$-MLER):

$$(\hat{\mathrm{B}}^{\ell_{2,1}\mathrm{MLER}}, \hat{\Sigma}^{\ell_{2,1}\mathrm{MLER}}) \in \underset{\substack{\mathrm{B}\in\mathbb{R}^{p\times q}\\ \Sigma\succeq\underline{\sigma}^2\,\mathrm{Id}_n}}{\arg\min} \sum_1^r \left\|Y^l - X\mathrm{B}\right\|^2_{\Sigma^{-1}} - \log\det(\Sigma^{-1}) + \lambda\left\|\mathrm{B}\right\|_{2,1} \ . \tag{5.3}$$

Problems (5.2) and (5.3) are not convex because the objective functions are not convex in $(\mathrm{B}, \Sigma^{-1})$, however they are biconvex, *i.e.,* convex in B and convex in Σ^{-1}. Alternate minimization can be used to solve Problems (5.2) and (5.3), but without guarantees to converge toward a global minimum.

Minimization in $\mathrm{B}_{j:}$ As for CLaR and SGCL the updates in $\mathrm{B}_{j:}$'s for $\ell_{2,1}$-MLE and $\ell_{2,1}$-MLER read:

$$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top \Sigma^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|^2_{\Sigma^{-1}}}, \frac{\lambda nq}{\|X_{:j}\|^2_{\Sigma^{-1}}}\right) \ . \tag{5.4}$$

Minimization in Σ^{-1}: for $\ell_{2,1}$-MLE (*resp.* $\ell_{2,1}$-MLER) the update in Σ reads:

$$\Sigma = \mathrm{Cl}(\Sigma^{\mathrm{emp}}, \underline{\sigma}^2) \quad (\textit{resp.}\ \Sigma = \mathrm{Cl}(\Sigma^{\mathrm{emp},r}, \underline{\sigma}^2)) \ , \tag{5.5}$$

with $\Sigma^{\mathrm{emp}} \triangleq \frac{1}{q}(\bar{Y} - X\mathrm{B})(\bar{Y} - X\mathrm{B})^\top$ (*resp.* $\Sigma^{\mathrm{emp},r} \triangleq \frac{1}{rq}\sum_{l=1}^r (Y^l - X\mathrm{B})(Y^l - X\mathrm{B})^\top$).

Let us prove the last result. Minimizing Problem (5.2) in Σ^{-1} amounts to solving:

$$\hat{\Sigma}^{-1} \in \underset{\mathbf{0}_{n,n}\prec\Sigma^{-1}\preceq 1/\underline{\sigma}^2}{\arg\min} \left\langle \Sigma^{\mathrm{emp}}, \Sigma^{-1} \right\rangle - \log\det(\Sigma^{-1}) \ . \tag{5.6}$$

Theorem 5.2. *Let* $\Sigma^{\text{emp}} = U \operatorname{diag}(\sigma_i^2) U^\top$ *be an eigenvalue decomposition of* Σ^{emp}*, a solution to Problem (5.6) is given by:*

$$\hat{\Sigma}^{-1} = U \operatorname{diag}\left(\frac{1}{\sigma_i^2 \vee \underline{\sigma}^2}\right) U^\top \ . \tag{5.7}$$

Theorem 5.2 is intuitive: the solution of the smoothed optimization problem (5.6) is the solution of the non-smoothed problem, where the eigenvalues of the solution have been lifted to satisfy the constraint.

Proof The KKT conditions of Problem (5.6) for conic programming (see Boyd and Vandenberghe 2004, p. 267) state that the optimum in the primal $\hat{\Sigma}^{-1}$ and the optimum in the dual $\hat{\Gamma}$ should satisfy:

$$\Sigma^{\text{emp}} - \hat{\Sigma} + \hat{\Gamma} = \mathbf{0}_{n,n} \ , \qquad \hat{\Gamma}^\top (\hat{\Sigma}^{-1} - \frac{1}{\underline{\sigma}^2} \operatorname{Id}_n) = \mathbf{0}_{n,n} \ ,$$
$$\hat{\Gamma} \in \mathcal{S}_+^n \ , \qquad \mathbf{0}_{n,n} \prec \hat{\Sigma}^{-1} \preceq \frac{1}{\underline{\sigma}^2} \ .$$

Since Problem (5.6) is convex these conditions are also sufficient. We exhibit a primal-dual pair $(\hat{\Sigma}^{-1}, \hat{\Gamma})$ satisfying the KKT conditions. Let $\Sigma^{\text{emp}} = U \operatorname{diag}(\sigma_i^2) U^\top$ be an eigenvalue decomposition of Σ^{emp}, one can check that:

$$\hat{\Sigma}^{-1} = U \operatorname{diag}(\frac{1}{\sigma_i^2 \vee \underline{\sigma}^2}) U^\top \ ,$$
$$\hat{\Gamma} = U \operatorname{diag}(\sigma_i^2 \vee \underline{\sigma}^2 - \sigma_i^2) U^\top \ .$$

verify the KKT conditions, leading to the desired result. ■

5.1.3 Multivariate Regression with Covariance Estimation (MRCE)

Introduced by Rothman et al. (2010), this estimator jointly estimates the regression coefficients (assumed to be sparse) and the precision matrix (*i.e.,* the inverse of the covariance matrix), which is also assumed to be sparse. Originally in Rothman et al. (2010) the sparsity enforcing term on the regression coefficients is a matrix ℓ_1-norm, also denoted as $\|\cdot\|_1$ in this section:

$$(\hat{\mathrm{B}}^{\text{MRCE}}, \hat{\Sigma}^{\text{MRCE}}) \in \underset{\substack{\mathrm{B} \in \mathbb{R}^{p \times q} \\ \Sigma^{-1} \succ \mathbf{0}_{n,n}}}{\arg\min} \ \|\bar{Y} - X\mathrm{B}\|_{\Sigma^{-1}}^2 - \log\det(\Sigma^{-1}) + \lambda \|\mathrm{B}\|_1 + \mu \|\Sigma^{-1}\|_1 \ . \tag{5.8}$$

Problem (5.8) is not convex, but can be solved heuristically (see Rothman et al. 2010 for details) by coordinate descent for the udpdates in B and solving a Graphical Lasso problem (Friedman et al., 2008) for the update in Σ^{-1}. The ℓ_1-norm being not well suited for our problem, we introduce an $\ell_{2,1}$ version of MRCE:

$$(\hat{\mathrm{B}}^{\ell_{2,1}\text{MRCE}}, \hat{\Sigma}^{\ell_{2,1}\text{MRCE}}) \in \underset{\substack{\mathrm{B} \in \mathbb{R}^{p \times q} \\ \Sigma^{-1} \succ \mathbf{0}_{n,n}}}{\arg\min} \ \|\bar{Y} - X\mathrm{B}\|_{\Sigma^{-1}}^2 - \log\det(\Sigma^{-1}) + \lambda \|\mathrm{B}\|_{2,1} + \mu \|\Sigma^{-1}\|_1 \ . \tag{5.9}$$

In order to combine $\ell_{2,1}$-MRCE to take take advantage of all the repetitions, one can think of the following estimator:

$$(\hat{\mathrm{B}}^{\ell_{2,1}\mathrm{MRCER}}, \hat{\Sigma}^{\ell_{2,1}\mathrm{MRCER}}) \in \underset{\substack{\mathrm{B}\in\mathbb{R}^{p\times q} \\ \Sigma^{-1}\succ\mathbf{0}_{n,n}}}{\arg\min} \sum_{1}^{r} \|Y^l - X\mathrm{B}\|^2_{\Sigma^{-1}} - \log\det(\Sigma^{-1}) + \lambda\|\mathrm{B}\|_{2,1} + \mu\|\Sigma^{-1}\|_1 \ . \tag{5.10}$$

As for Problem (5.8), Problems (5.9) and (5.10) can be heuristically solved with alternated block coordinate descent and Graphical Lasso steps.

Update in $\mathrm{B}_{j:}$ The minimization is the same as for $\ell_{2,1}$-MLE and $\ell_{2,1}$-MLER:

$$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top \Sigma^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|^2_{\Sigma^{-1}}}, \frac{\lambda nq}{\|X_{:j}\|^2_{\Sigma^{-1}}}\right) \ . \tag{5.11}$$

Update in Σ^{-1} Solving Problem (5.9) in Σ^{-1} amounts to solving:

$$\mathrm{glasso}(\Sigma, \mu) \triangleq \underset{\Sigma^{-1}\succ\mathbf{0}_{n,n}}{\arg\min} \ \langle \Sigma^{\mathrm{emp}}, \Sigma^{-1}\rangle - \log\det(\Sigma^{-1}) + \mu\|\Sigma^{-1}\|_1 \ . \tag{5.12}$$

This is a well known and well studied problem (Friedman et al., 2008) that can be solved through coordinate descent. For our implementation we used the `scikit-learn` (Pedregosa et al., 2011) version of the Graphical Lasso. Note that solving the Graphical Lasso on very ill-conditioned empirical covariance matrices such as Σ^{emp} is very long: we thus only considered $\ell_{2,1}$-MRCER were the Graphical Lasso is applied on $\Sigma^{\mathrm{emp},r}$.

5.1.4 Algorithms summary

Each optimization problem is solved with block coordinate descent, whether there is theoretical guarantees for it to converge toward a global minimum (for convex formulations, CLaR, SGCL and MTL), or not (for non-convex formulations, $\ell_{2,1}$-MLE, $\ell_{2,1}$-MLER, $\ell_{2,1}$-MRCER). The cost for the updates for each algorithm can be found in Table 5.1. The formula for the updates in $\mathrm{B}_{j:}$'s and S or Σ for each algorithm can be found in Table 5.2.

Let f^{dual} be the number of updates of B for one update of S or Σ.

Table 5.1 – Algorithms cost in time summary

	CD epoch cost	convex	dual gap cost
CLaR	$\mathcal{O}(\frac{n^3+qn^2}{f^{\mathrm{dual}}} + pn^2 + pnq)$	✓	$\mathcal{O}(rnq + p)$
SGCL	$\mathcal{O}(\frac{n^3+qn^2}{f^{\mathrm{dual}}} + pn^2 + pnq)$	✓	$\mathcal{O}(nq + p)$
$\ell_{2,1}$-MLER	$\mathcal{O}(\frac{n^3+qn^2}{f^{\mathrm{dual}}} + pn^2 + pnq)$	✗	not convex
$\ell_{2,1}$-MLE	$\mathcal{O}(\frac{n^3+qn^2}{f^{\mathrm{dual}}} + pn^2 + pnq)$	✗	not convex
$\ell_{2,1}$-MRCER	$\mathcal{O}(\frac{\mathcal{O}(\mathrm{glasso})}{f^{\mathrm{dual}}} + pn^2 + pnq)$	✗	not convex
MTL	$\mathcal{O}(npq)$	✓	$\mathcal{O}(nq + p)$

Recalling that $\Sigma^{\text{emp}} \triangleq \frac{1}{q}(\bar{Y} - X\mathrm{B})(\bar{Y} - X\mathrm{B})^\top$ and $\Sigma^{\text{emp},r} \triangleq \frac{1}{rq}\sum_{l=1}^{r}(Y^l - X\mathrm{B})(Y^l - X\mathrm{B})^\top$, a summary of the updates in S or Σ and $\mathrm{B}_{j:}$'s for each algorithm is given in Table 5.2.

Comments on Table 5.2 The updates in S/Σ and $\mathrm{B}_{j:}$'s are given in Table 5.2. Although the updates may look similar, all the algorithms can lead to very different results, as Figures 5.6, 5.8, 5.10 and 5.12 will illustrate.

Table 5.2 – Algorithms updates summary

	update in $\mathrm{B}_{j:}$	update in S or Σ
CLaR	$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top S^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|_{S^{-1}}^2}, \frac{\lambda nq}{\|X_{:j}\|_{S^{-1}}^2}\right)$	$S = \mathrm{ClSqrt}(\Sigma^{\text{emp},r}, \underline{\sigma})$
SGCL	$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top S^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|_{S^{-1}}^2}, \frac{\lambda nq}{\|X_{:j}\|_{S^{-1}}^2}\right)$	$S = \mathrm{ClSqrt}(\Sigma^{\text{emp}}, \underline{\sigma})$
$\ell_{2,1}$-MLER	$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top \Sigma^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|_{\Sigma^{-1}}^2}, \frac{\lambda nq}{\|X_{:j}\|_{\Sigma^{-1}}^2}\right)$	$\Sigma = \mathrm{Cl}(\Sigma^{\text{emp},r}, \underline{\sigma}^2)$
$\ell_{2,1}$-MLE	$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top \Sigma^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|_{\Sigma^{-1}}^2}, \frac{\lambda nq}{\|X_{:j}\|_{\Sigma^{-1}}^2}\right)$	$\Sigma = \mathrm{Cl}(\Sigma^{\text{emp}}, \underline{\sigma}^2)$
$\ell_{2,1}$-MRCER	$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top \Sigma^{-1}(Y - X\mathrm{B})}{\|X_{:j}\|_{\Sigma^{-1}}^2}, \frac{\lambda nq}{\|X_{:j}\|_{\Sigma^{-1}}^2}\right)$	$\Sigma = \mathrm{glasso}(\Sigma^{\text{emp},r}, \mu)$
MTL	$\mathrm{B}_{j:} = \mathrm{BST}\left(\mathrm{B}_{j:} + \frac{X_{:j}^\top (Y - X\mathrm{B})}{\|X_{:j}\|^2}, \frac{\lambda nq}{\|X_{:j}\|^2}\right)$	no update in S/Σ

5.2 CLaR

The Python code, with Numba compilation (Lam et al., 2015), is released as an open source package: https://github.com/QB3/CLaR. It includes the code to run the SGCL as a special instance of CLaR.

We compare CLaR to other estimators: SGCL (**Massias** et al., 2018a), an $\ell_{2,1}$ version of MLE (Chen and Banerjee, 2017; Lee and Liu, 2012) ($\ell_{2,1}$-MLE), a version of the $\ell_{2,1}$-MLE with multiple repetitions ($\ell_{2,1}$-MLER), an $\ell_{2,1}$ penalized version of MRCE (Rothman et al., 2010) with repetitions ($\ell_{2,1}$-MRCER) and the Multi-Task Lasso (MTL, Obozinski et al. 2010). The cost of an epoch of block coordinate descent is summarized in Table 5.1 in Section 5.1.4 for each algorithm, along with the duality gap cost when available. All competitors are detailed in Section 5.1.

Synthetic data Here we demonstrate the ability of our estimator to recover the support *i.e.,* the ability to identify the predictive features. There are $n = 150$ observations, $p = 500$ features, $q = 100$ tasks. The design X is random with Toeplitz-correlated features with parameter $\rho_X = 0.6$ (correlation between $X_{:i}$ and $X_{:j}$ is $\rho_X^{|i-j|}$), and its columns have unit Euclidean norm. The true coefficient B^* has 30 non-zeros rows whose entries are independent and normally centered distributed. S^* is a Toeplitz matrix with parameter ρ_S. The SNR is fixed and constant across all repetitions:

$$\mathrm{SNR} \triangleq \|X\mathrm{B}^*\| / \sqrt{r}\|X\mathrm{B}^* - \bar{Y}\| \ . \tag{5.13}$$

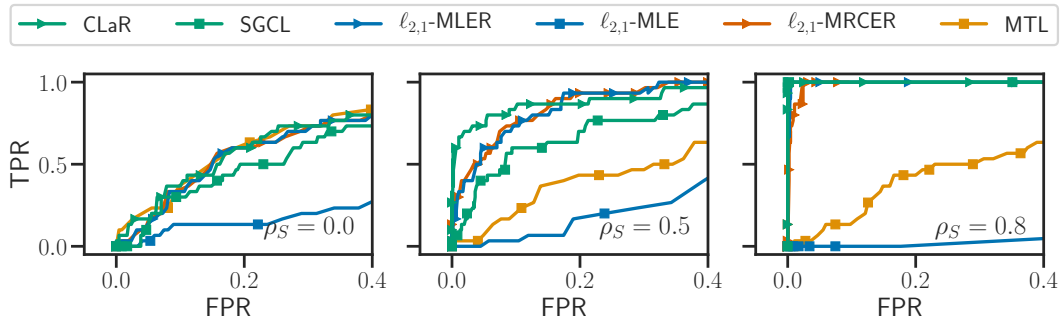


Figure 5.1 – *Influence of noise structure.* ROC curves of support recovery ($\rho_X = 0.6$, SNR $= 0.03$, $r = 20$) for different ρ_S values.

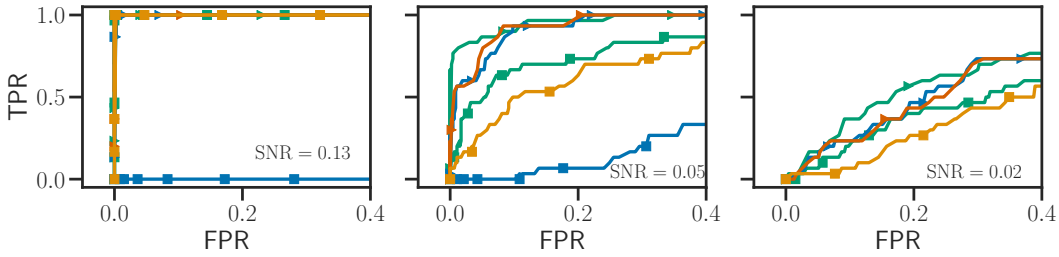


Figure 5.2 – *Influence of SNR.* ROC curves of support recovery ($\rho_X = 0.6$, $\rho_S = 0.4$, $r = 20$) for different SNR values.

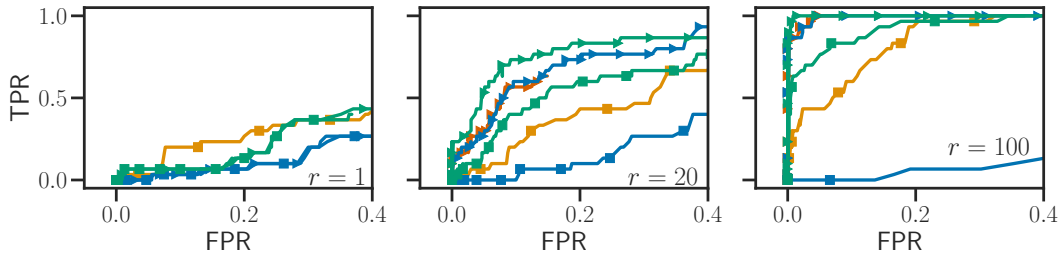


Figure 5.3 – *Influence of the number of repetitions.* ROC curves of support recovery ($\rho_X = 0.6$, SNR $= 0.03$, $\rho_S = 0.4$) for different r values.

For Figures 5.1 to 5.3, the figure of merit is the ROC curve, *i.e.,* the true positive rate (TPR) against the false positive rate (FPR). For each estimator, the ROC curve is obtained by varying the value of the regularization parameter λ on a geometric grid of 160 points, from $\lambda_{\max}$ (the smallest regularization strength leading to $\mathbf{0}_{p,q}$ being solution), estimator-specific) to $\lambda_{\min}$, the latter also being estimator-specific and chosen to obtain a FPR larger than 0.4.

Influence of noise structure. Figure 5.1 represents the ROC curves for different values of ρ_S. As ρ_S increases, the noise becomes more and more heteroscedastic. From left to right, the performance of heteroscedastic solvers (CLaR, SGCL, $\ell_{2,1}$-MRCER, $\ell_{2,1}$-MRCE, $\ell_{2,1}$-MLER) increases as they are designed to exploit correlations in the noise, while the performance of MTL decreases, as its homoscedastic model becomes less and less valid.

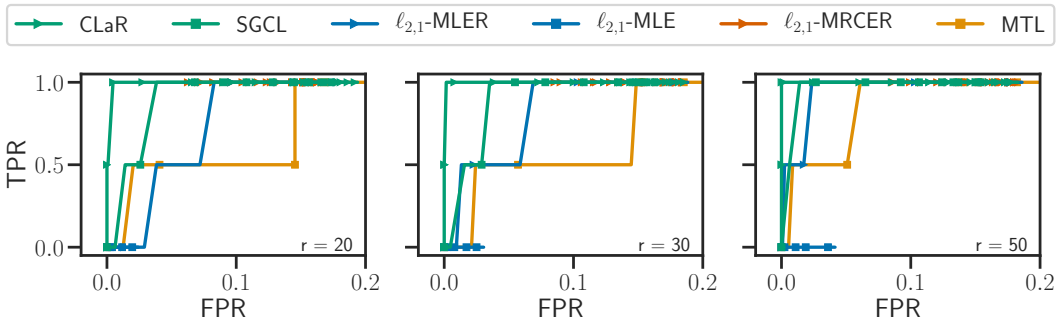


Figure 5.4 – *Influence of the number of repetitions.* ROC curves with empirical X and S and simulated B^* (amp = 2 nA.m), for different number of repetitions.

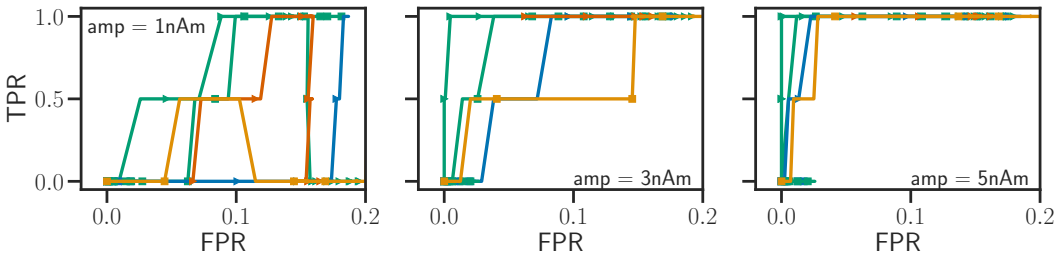


Figure 5.5 – *Amplitude influence.* ROC curves with empirical X and S and simulated B^* ($r = 50$), for different signal amplitudes.

Influence of SNR. On Figure 5.2 we can see that when the SNR is high (left), all estimators (except $\ell_{2,1}$-MLE) reach the (0, 1) point. This means that for each algorithm (except $\ell_{2,1}$-MLE), there exists a value of λ such that the estimated support is exactly the true one. However, when the SNR decreases (middle), the performance of SGCL and MTL starts to drop, while that of CLaR, $\ell_{2,1}$-MLER and $\ell_{2,1}$-MRCER remains stable (CLaR performing better), highlighting their capacity to leverage multiple repetitions of measurements to handle the noise structure. Finally, when the SNR is too low (right), all algorithms perform poorly, but CLaR, $\ell_{2,1}$-MLER and $\ell_{2,1}$-MRCER still performs better.

Influence of the number of repetitions. Figure 5.3 shows ROC curves of all compared approaches for different r, starting from $r = 1$ (left) to 100 (right). Even with $r = 20$ (middle) CLaR outperforms the other estimators, and when $r = 100$, CLaR can better leverage the large number of repetitions.

Realistic data We now evaluate the estimators on realistic magneto- and electroencephalography (M/EEG) data. We recall (see Section 1.2) that the M/EEG recordings measure the electrical potential and magnetic fields induced by the active neurons. Data are time series of length q with n sensors and p sources mapping to locations in the brain. Because the propagation of the electromagnetic fields is driven by the linear Maxwell equations, one can assume that the relation between the measurements $Y^1, \dots, Y^r$ and the amplitudes of sources in the brain B^* is linear.

The M/EEG inverse problem consists in identifying B^*. Because of the limited number of sensors (a few hundreds in practice), as well as the physics of the problem, the M/EEG inverse problem is severely ill-posed and needs to be regularized. Moreover,

the experiments being usually short (less than 1 s.) and focused on specific cognitive functions, the number of active sources is expected to be small, *i.e.,* B^* is assumed to be row-sparse. This plausible biological assumption motivates the framework of Part II.

Dataset. We use the *sample* dataset from the MNE software (Gramfort et al., 2014). The experimental conditions are here auditory stimulations in the right or left ears, leading to two main foci of activations in bilateral auditory cortices (*i.e.,* 2 non-zeros rows for B^*). For this experiment, we keep only the gradiometer magnetic channels. After removing one channel corrupted by artifacts, this leads to $n = 203$ signals. The length of the temporal series is $q = 100$, and the data contains $r = 50$ repetitions. We choose a source space of size $p = 1281$ which corresponds to about 1 cm distance between neighboring sources. The orientation is fixed, and normal to the cortical mantle.

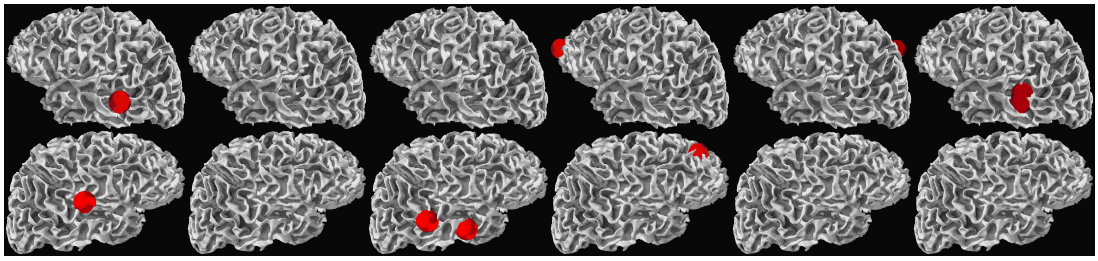
Realistic MEG data simulations. We use here true empirical values for X and S by solving Maxwell equations and computing an empirical co-standard deviation matrix on pre-stimulus data. To generate realistic MEG data we simulate neural responses B^* with 2 non-zeros rows corresponding to areas known to be related to auditory processing (Brodmann area 22). Each non-zero row of B^* is chosen as a sinusoidal signal with realistic frequency (5 Hz) and amplitude (amp $\sim 1 - 10$ nA.m). We finally simulate r MEG signals following the model $Y^l = X\mathrm{B}^* + S^*\mathrm{E}^l$, E^l being matrices with i.i.d. normal entries.

The signals being contaminated with correlated noise, if one wants to use homoscedastic solvers it is necessary to whiten the data first (and thus to have an estimation of the covariance matrix, the later often being unknown). In this experiment we demonstrate that without this whitening process, the homoscedastic solver MTL fails, as well as solvers which does not take in account the repetitions: SGCL and $\ell_{2,1}$-MLE. In this scenario CLaR, $\ell_{2,1}$-MLER and $\ell_{2,1}$-MRCER do succeed in recovering the sources, CLaR leading to the best results. As for the synthetic data, Figures 5.4 and 5.5 are obtained by varying the estimator-specific regularization parameter λ from $\lambda_{\max}$ to $\lambda_{\min}$ on a geometric grid.

Influence of the number of repetitions. Figure 5.4 shows ROC curves for different number of repetitions r. When the number of repetitions is high (right, $r = 50$), the algorithms taking into account all the repetitions (CLaR, $\ell_{2,1}$-MLER, $\ell_{2,1}$-MRCER) perform best, almost hitting the $(0, 1)$ corner, whereas the algorithms which do not take into account all the repetitions ($\ell_{2,1}$-MLE, MTL, SGCL) perform poorly. As soon as the number of repetitions decreases (middle and left) the performances of all the algorithms except CLaR start dropping severely. CLaR is once again the algorithm taking the most advantage of the number of repetitions.

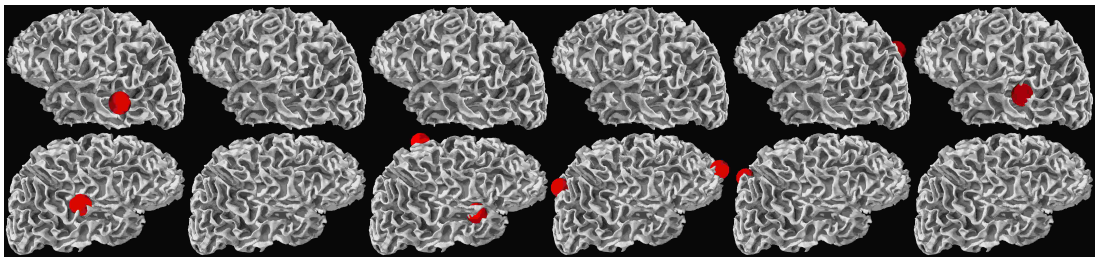
Amplitude influence. Figure 5.5 shows ROC curves for different values of the amplitude of the signal. When the amplitude is high (right), all the algorithms perform well, however when the amplitude decreases (middle) only CLaR leads to good results, almost hitting the $(0, 1)$ corner. When the amplitude gets lower (left) all algorithms perform worse, CLaR still yielding the best results.

Real data As above, we use the *sample* dataset from MNE, keeping only the magnetometer magnetic channels ($n = 102$ signals). We choose a source space of size $p = 7498$ (about 5 mm between neighboring sources). The orientation is fixed, and normal to the cortical mantle. As for realistic data, X is the empirical design matrix, but this time we use the empirical measurements $Y^1, \ldots, Y^r$. The experiment are left or right auditory



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.6 – *Real data, left auditory stimulations ($n = 102$, $p = 7498$, $q = 76$, $r = 63$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after left auditory stimulations .



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

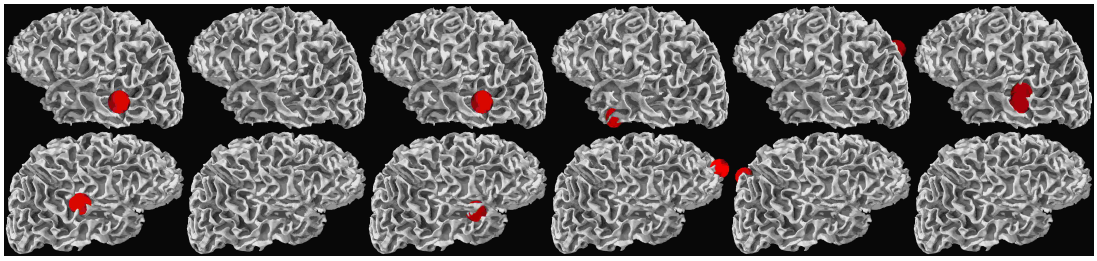
Figure 5.7 – *Real data, right auditory stimulations ($n = 102$, $p = 7498$, $q = 76$, $r = 33$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after right auditory stimulations.

or visual stimulations. As two sources are expected (one in each hemisphere, in bilateral auditory cortices), we vary λ by dichotomy between $\lambda_{\max}$ (returning 0 sources) and a $\lambda_{\min}$ (returning more than 2 sources), until finding a value of λ giving exactly 2 sources. Results are provided in Figures 5.6 and 5.7. Running times of each algorithm are of the same order of magnitude and can be found in Section 5.4.

Comments on Figure 5.6, left auditory stimulations. Sources found by the algorithms are represented by red spheres. SGCL, $\ell_{2,1}$-MLE and $\ell_{2,1}$-MRCER completely fail, finding sources that are not in the auditory cortices at all (SGCL sources are deep, thus not in the auditory cortices, and cannot be seen). MTL and $\ell_{2,1}$-MLER do find sources in auditory cortices, but only in one hemisphere (left for MTL and right for $\ell_{2,1}$-MLER). CLaR is the only one that finds one source in each hemisphere in the auditory cortices as expected.

Comments on Figure 5.7, right auditory stimulations. In this experiment we only keep $r = 33$ repetitions (out of 63 available) and it can be seen that only CLaR finds correct sources, MTL finds sources only in one hemisphere and all the other algorithms do find sources that are not in the auditory cortices. This highlights the robustness of CLaR, even with a limited number of repetitions, confirming previous experiments (see Figure 5.3).

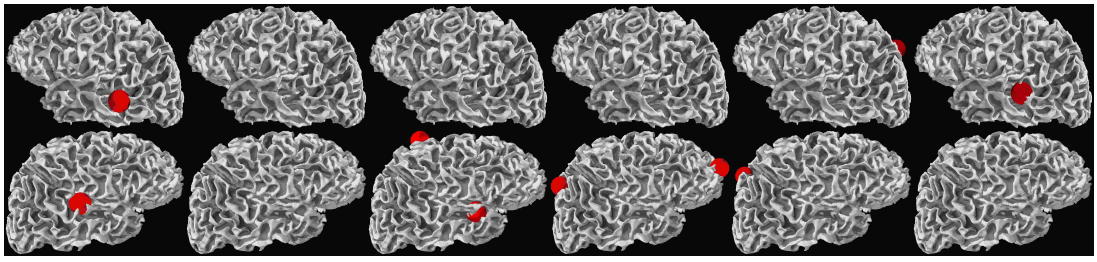
5.2.1 Right auditory stimulations



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.8 – *Real data ($n = 102$, $p = 7498$, $q = 76$, $r = 65$)*: sources found in the left hemisphere (top) and the right hemisphere (bottom) after right auditory stimulations.

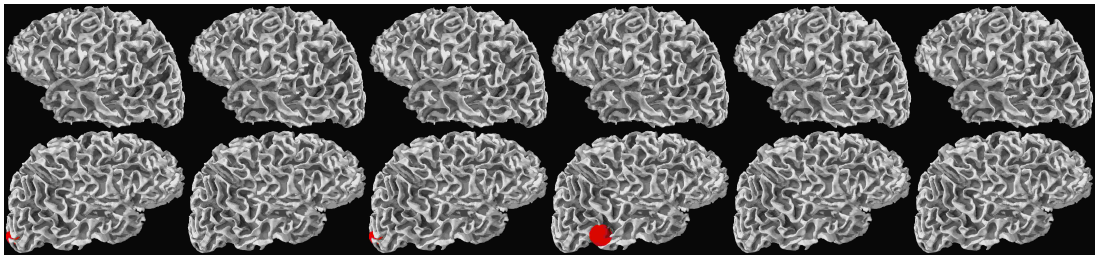
Figures 5.8 and 5.9 show the solution given by each algorithm on real data after right auditory stimulations. As two sources are again expected (one in each hemisphere, in bilateral auditory cortices), we repeat the procedure detailed in the previous experiment to find a suitable value of λ per estimator. Figure 5.8 (*resp.* Figure 5.9) shows the solution given by the algorithms taking in account all the repetitions (*resp.* only half of the repetitions). When the number of repetitions is high (Figure 5.8) only CLaR and $\ell_{2,1}$-MLER find one source in each auditory cortices, MTL does find sources only in one hemisphere, all the other algorithms fail by finding sources not in the auditory cortices at all. Moreover when the number of repetitions is decreasing (Figure 5.9) $\ell_{2,1}$-MLER fails and only CLaR does find 2 sources, one in each hemisphere. Once again CLaR is more robust and performs better, even when the number of repetitions is lower.



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.9 – *Real data ($n = 102$, $p = 7498$, $q = 76$, $r = 33$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after right auditory stimulations.

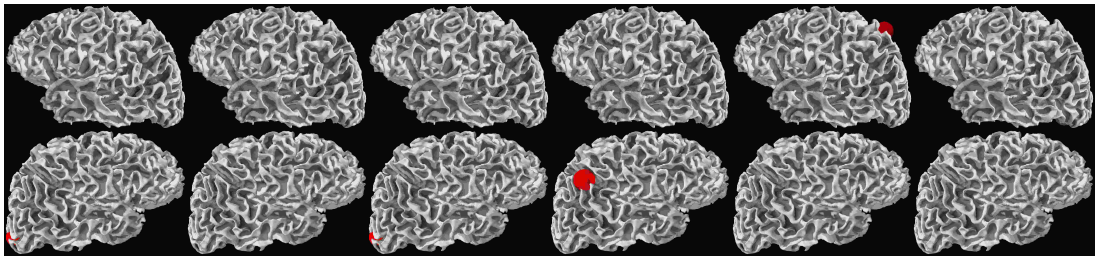
5.2.2 Left visual stimulations



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.10 – *Real data ($n = 102$, $p = 7498$, $q = 48$, $r = 71$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after left visual stimulations.

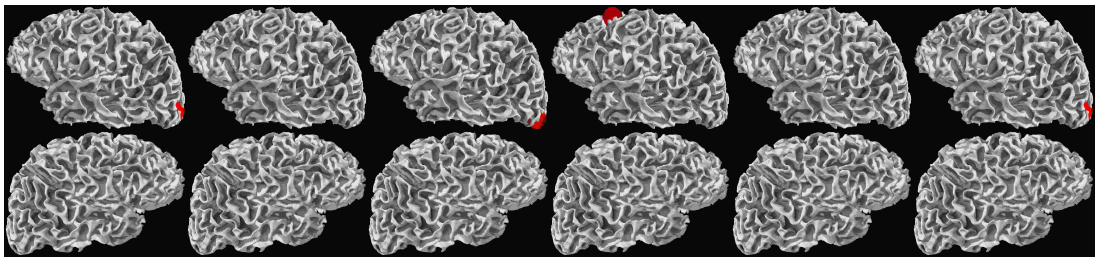
Figures 5.10 and 5.11 show the results for each algorithm after left visual stimulations. As one source is expected (in the right hemisphere), we vary λ by dichotomy between $\lambda_{\max}$ (returning 0 source) and a $\lambda_{\min}$ (returning more than 1 source), until finding a value of λ giving exactly 1 source. When the number of repetitions is high (Figure 5.10) only CLaR and $\ell_{2,1}$-MLER do find a source in the visual cortex. When the number of repetitions decreases, CLaR and $\ell_{2,1}$-MLER still find one source in the visual cortex, other algorithms fail. This highlights this importance to take in account the repetitions.



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.11 – *Real data ($n = 102$, $p = 7498$, $q = 48$, $r = 36$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after left visual stimulations.

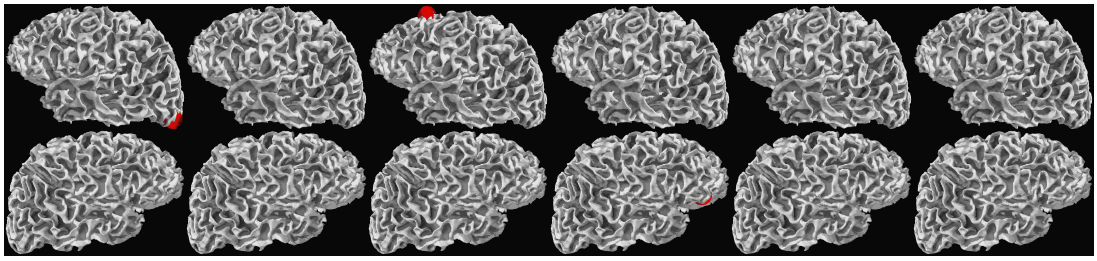
5.2.3 Right visual stimulations



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.12 – *Real data ($n = 102$, $q = 7498$, $q = 48$, $r = 61$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after right visual stimulations.

Figures 5.12 and 5.13 show the results for each algorithm after right visual stimulations. As one source is expected (in the left hemisphere), we vary λ by dichotomy between $\lambda_{\max}$ (returning 0 sources) and a $\lambda_{\min}$ (returning more than 1 source), until finding a lambda giving exactly 1 source. When the number of repetitions is high (Figure 5.12) only CLaR, $\ell_{2,1}$-MLER and MTL do find a source in the visual cortex. When the number of repetitions decreases (Figure 5.13), only CLaR finds one source in the visual cortex, other algorithms fail. This highlights once again the robustness of CLaR, even with a limited number of repetitions.



(a) CLaR (b) SGCL (c) $\ell_{2,1}$-MLER (d) $\ell_{2,1}$-MLE (e) $\ell_{2,1}$-MRCER (f) MTL

Figure 5.13 – *Real data ($n = 102$, $q = 7498$, $q = 48$, $r = 31$)* Sources found in the left hemisphere (top) and the right hemisphere (bottom) after right visual stimulations.

Finally, we describe the preprocessing pipeline used for the realistic and real data (see Section 5.3). We then propose time comparison for all the algorithms (see Section 5.4).

5.3 Preprocessing steps for realistic and real data

When using multi-modal data without whitening, one has to rescale properly data, indeed data needs to have the same order of magnitude, otherwise some mode (for example EEG data) could be (almost) completely ignored by the optimization algorithm. The preprocessing pipeline used to rescale realistic data (Figures 5.4 and 5.5) and real data (Figures 5.6, 5.8, 5.10 and 5.12) is described in Algorithm 5.1.

Algorithm 5.1 Preprocessing steps for realistic and real data

input : $X, Y^1, \ldots, Y^r$

// rescale each line of X

for $i = 1, \ldots, n$ **do**
  for $l = 1, \ldots, r$ **do**
    $Y^l_{i:} \leftarrow Y^l_{i:} / \|X_{i:}\|$
  $X_{i:} \leftarrow X_{i:} / \|X_{i:}\|$

// rescale each column of X

for $j = 1, \ldots, q$ **do**
  $X_{:j} \leftarrow X_{:j} / \|X_{:j}\|$

return $X, Y^1, \ldots, Y^r$

5.4 Time comparison

The goal of this small experiment is to show that our algorithm (CLaR) is as costly as a Multi-Task Lasso or other competitors (in the M/EEG context, *i.e.*, n not too large). The time taken by each algorithm to produce Figure 5.6 (real data, left auditory stimulations) is given in Figure 5.14. In this experiment the stoppping tolerance is set to $\epsilon = 10^{-3}$, the safe stopping criterion is duality gap $< \epsilon$ (only available for convex optimization problems). The heuristic stopping criterion is to stop when the objective no longer decrease steeply enough, *i.e.*, we stop when $\mathcal{P}(\mathrm{B}^{(t)}, \Sigma^{(t)}) - \mathcal{P}(\mathrm{B}^{(t+1)}, \Sigma^{(t+1)}) < \epsilon/10$. The safe stopping criterion is only available for CLaR, SGCL and MTL (it takes too much time – more than 10 min – for SGCL to get to a duality gap inferior to ϵ).

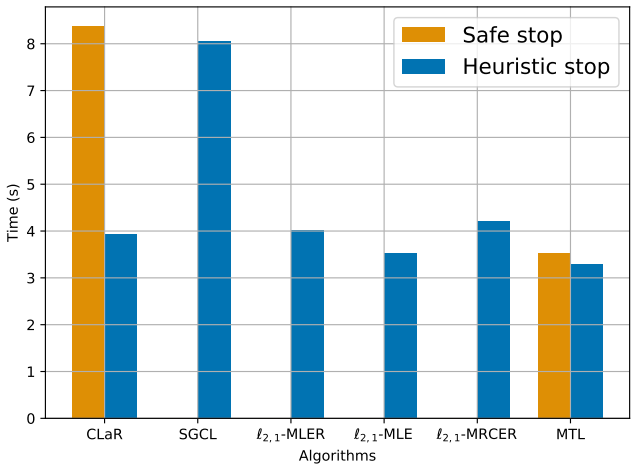


Figure 5.14 – *Time comparison, real data,* $n = 102$, $p = 7498$, $q = 54$, $r = 56$. Time for each algorithm to produce Figure 5.6.

Comment on Figure 5.14 Figure 5.14 shows that if we use the heuristic stopping criterion, CLaR is as fast the other algorithm. In addition CLaR has a safe stopping criterion which only take 2 to 3 more time than the heuristic one (less than 10 sec).

Conclusion and perspectives

Winter does not last forever.
Spring comes, snows melt.

In this thesis, we have introduced better numerical solutions to the $\ell_{2,1}$-regularized bio-magnetic inverse problem. First, we designed a better dual point construction for coordinate descent solvers used in ℓ_1-type optimization. This point helps to identify active features faster than previous approaches. Celer, the solver based on this construction, shows improvements of Gap Safe screening rules and working set algorithms in an extended benchmark. It is available as an easy to use Python package, for reproducibility and practical impact. Initially designed for the Lasso, we then applied this method to popular problems such as sparse Logistic regression and Multitask Lasso, whose range of application go far beyond the field of brain imaging.

Then, we introduced concomitant noise estimators, to address the complex structure of magneto-electroencephalographic measurements. We handle the multiple repetition setting of M/EEG experiments for a better estimation of the noise matrix. Contrary to existing "noise aware" estimators, they are jointly convex, and the smoothing of the nuclear norm makes the underlying optimization problem easy to solve. The smoothing analysis we provide for CLaR paves the way for a simplified use of Schatten norms as datafitting term in sparse problems.

The performance obtained for dual extrapolation is visible after support identification. In our Machine Learning setting, we have empirically observed that this identification usually happens quickly. However, it may not be the case in other situations, and modifying our extrapolation procedure for it to perform well before support identification is a perspective to make it more efficient. We devoted our efforts to the construction of a better dual point, as we had previously observed that the duality gap provided a massive overestimation of the suboptimality. Now that this bottleneck has been removed, a promising perspective is to apply extrapolation in the primal, and to generalize to more schemes than cyclic coordinate descent.

Working set methods start with a small number of features. In homotopy methods, the solvers start with a high value of λ, giving a very sparse solution, then decrease it gradually. Connections between these techniques (parallel between geometric grid of regularizers and the popular geometric growth of the working set size) may help us to justify the multiple heuristics upon which efficient working set algorithms are based.

CLaR depends on a noise parameter controlling the smallest eigenvalue of the noise covariance. Unfortunately, compared to its minimal noise level interpretation in the Concomitant Lasso, this parameter has no simple explanation in the matrix case. We are currently working on a statistical analysis of CLaR, shedding some light on the optimal way to set its value.

Combination of the two parts, *i.e.,* dual extrapolation for CLaR, as been implemented as a proof of concept; however the preliminary step is to better understand the putative VAR nature of Multitask Lasso iterates.

Appendices

A

Choice of parameters in Celer

A.1 Additional experiments

A.1.1 Choice of f and K

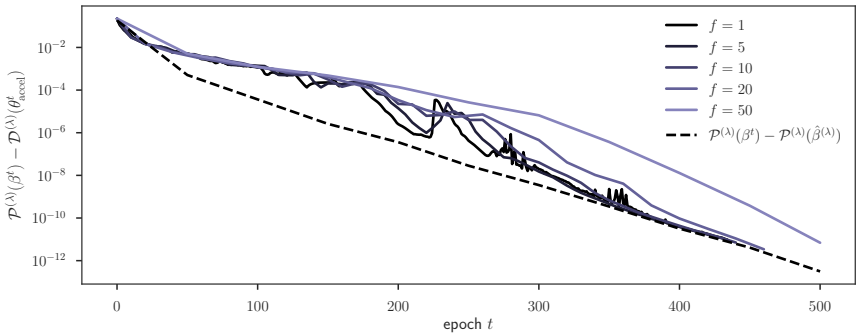


Figure A.1 – Duality gap evaluated with θ_{accel} as a function of the parameter f, for $K = 5$.

Figure A.1 shows that if the residuals used to extrapolate are too close (small f), the performance of acceleration is too noisy (though the duality gap still converges to the true suboptimality gap). For residuals too far apart (large f), the convergence towards $\hat{\theta}$ is slower and the duality gap does not reach the true suboptimality gap as it should ideally. $f^{\text{dual}} = 10$ provides the best performance.

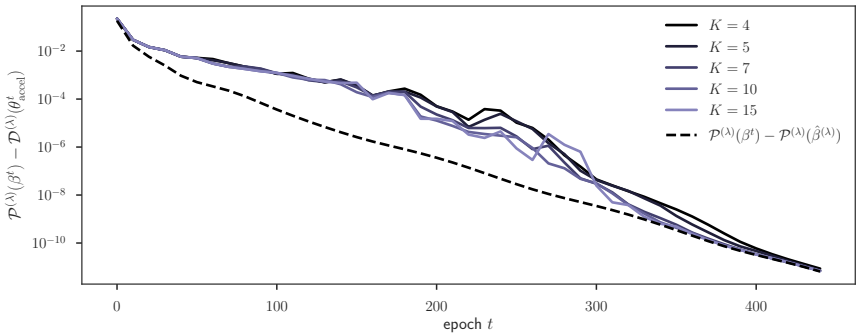


Figure A.2 – Duality gap evaluated with θ_{accel} as a function of the parameter K, for $f^{\text{dual}} = 10$.

Figure A.2 shows that the choice of K is not critical: all performances are nearly equivalent. Hence, we keep the default choice $K = 5$ proposed in Scieur et al. (2016).

A.1.2 Working set size policy

In this section, we demonstrate how the growth policy we chose in Equation (2.39) behaves better than others. We consider two types of growth: geometric of factor γ:

$$p^{(t)} = \min(\gamma \|\beta^{(t-1)}\|_0, p) \ , \tag{A.1}$$

and linear of factor γ:

$$p^{(t)} = \min(\gamma + \|\beta^{(t-1)}\|_0, p) \ . \tag{A.2}$$

We implement these two strategies with factor 2 and 4 for the geometric, and 10 and 50 for the linear. We consider two scenarios:

- *undershooting*, with $p^{(1)} = 10$ much smaller than the true support size $\|\hat{\beta}\|_0 = 983$ (obtained with $\lambda = \lambda_{\max}/20$),
- *overshooting*, with $p^{(1)} = 500$ much larger than the true support size $\|\hat{\beta}\|_0 = 63$ (obtained with $\lambda = \lambda_{\max}/5$).

Figure A.3 shows that, when the first working set is too small (choice of $p^{(1)} = 10$), the approximate solutions are dense and the subsequent $\mathcal{W}_{(t)}$ grow in size. Amongst the four strategies considered, the geometric growth with factor 2 quickly reaches the targeted support size (contrary to the linear strategies), and does not create way too large WS like the geometric strategy with factor 4 does.

Figure A.4 shows that, if the initial guess is too large, using $\|\beta^{(t-1)}\|_0$ instead of $|\mathcal{W}^{(t-1)}|$ immediately decreases the size of $\mathcal{W}_{(1)}$, thus correcting the initial mistake and avoiding solving too large subproblems.

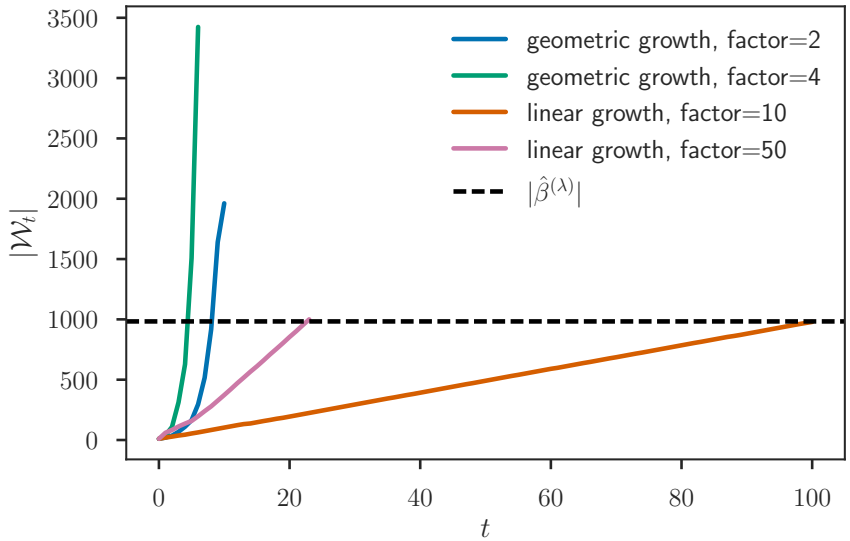


Figure A.3 – Size of working sets defined by Celer with linear or geometric growth, when the support size is underestimated.

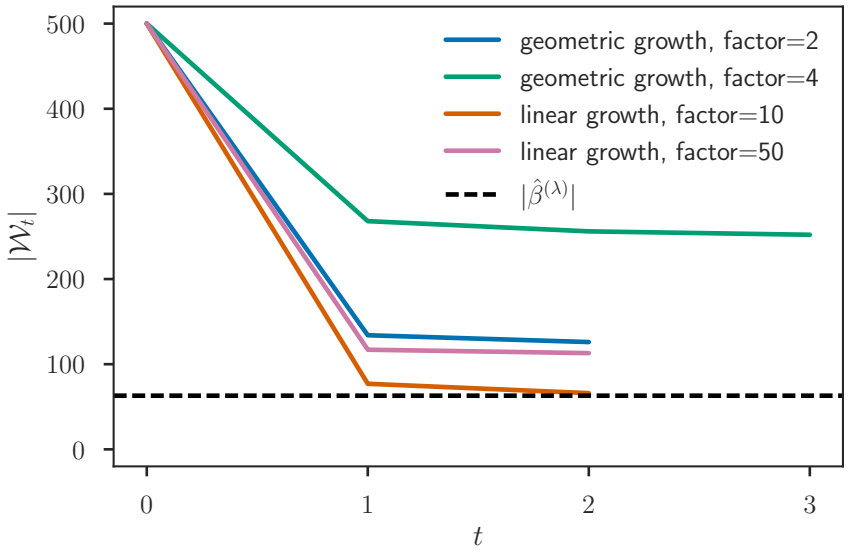


Figure A.4 – Size of working sets defined by Celer with linear or geometric growth, when the support size is overestimated.

A.1.3 Path on other dataset

Table A.1 reproduces the results of Section 2.5.4 on another dataset: bcTCGA, obtained from the The Cancer Genome Atlas (TCGA) Research Network[1]. For this dense dataset, $n = 536$ and $p = 17\,323$ (unregularized intercept column added). The grid goes from $\lambda_{\max}$ to $\lambda_{\max}/100$. The conclusions from Section 2.5.4 still hold.

Table A.1 – Computation time (in seconds) for Celer (no pruning) and Blitz to reach a given precision ϵ for a Lasso path on a dense grid, on the bcTCGA dataset.

ϵ	10^{-2}	10^{-4}	10^{-6}	10^{-8}
Celer	6	45	160	255
Blitz	22	101	252	286

Note that for the lowest precision the Blitz solver stops running due to internal stopping criterion measuring primal decrease and time spent on working set, but the evaluated duality gap when the solver stops is not always lower than ϵ along the path.

[1] http://cancergenome.nih.gov/

B
Concomitant estimation

B.1 Block homosecedastic model

Formally, if the k-th group of sensors is composed of n_k sensors ($\sum_1^K n_k = n$), with design matrix $X^k \in \mathbb{R}^{n_k \times p}$, observation matrix $Y^k \in \mathbb{R}^{n_k \times q}$ and noise level $\sigma_k^* > 0$, the block homoscedastic model is a combination of K homoscedastic models:

$$\forall k \in [K], \quad Y^k = X^k \mathrm{B}^* + \sigma_k^* \mathrm{E}^k \ , \tag{B.1}$$

with the entries of E^k independently sampled from $\mathcal{N}(0, 1)$. In the following we denote

$$X = \begin{pmatrix} X^1 \\ \vdots \\ X^K \end{pmatrix}, \ Y = \begin{pmatrix} Y^1 \\ \vdots \\ Y^K \end{pmatrix}, \ \mathrm{E} = \begin{pmatrix} \mathrm{E}^1 \\ \vdots \\ \mathrm{E}^K \end{pmatrix}, \text{ and } S^* = \begin{pmatrix} \sigma_1^* \, \mathrm{Id}_{n_1} & & 0 \\ & \ddots & \\ 0 & & \sigma_K^* \, \mathrm{Id}_{n_K} \end{pmatrix} \in \mathcal{S}_{++}^n.$$

Following this model, we call multi-task Smoothed Block Homoscedastic Concomitant Lasso (multi-task SBHCL) the estimator similar to the one of Problem (4.6) with the additional constraint that S is a diagonal matrix $\mathrm{diag}(\sigma_1 \, \mathrm{Id}_{n_1}, \dots, \sigma_K \, \mathrm{Id}_{n_K})$, with K constraints $\sigma_k \geq \underline{\sigma}_k$:

$$\underset{\substack{\mathrm{B} \in \mathbb{R}^{p \times q}, \\ \sigma_1, \dots, \sigma_K \in \mathbb{R}_{++}^K \\ \sigma_k \geq \underline{\sigma}_k, \forall k \in [K]}}{\arg\min} \ \sum_{k=1}^K \left(\frac{\|Y^k - X^k \mathrm{B}\|^2}{2nq\sigma_k} + \frac{n_k \sigma_k}{2n} \right) + \lambda \left\| \mathrm{B} \right\|_{2,1} \ . \tag{B.2}$$

Since Problem (B.2) does not admit a closed-form solution, we also propose an iterative solver, along with a stopping condition based on the duality gap, which is derived for this problem in Theorem B.1.

• When the constraints on the σ_k's are not saturated at optimality, formulation Problem (B.2) has an equivalent square-root Lasso (Belloni et al., 2011) formulation:

$$\underset{\mathrm{B} \in \mathbb{R}^{p \times q}}{\arg\min} \ \frac{1}{nq} \sum_{k=1}^K \sqrt{n_k} \|Y^k - X^k \mathrm{B}\| + \lambda \left\| \mathrm{B} \right\|_{2,1} \ . \tag{B.3}$$

• To fix the values of the lower bounds on the noise levels σ_k, we use an arbitrary proportion of the initial estimation of the noise variances per block *i.e.*, $\underline{S} = 10^{-\alpha} \, \mathrm{diag}(\|Y^1\| / \sqrt{n_1 q} \, \mathrm{Id}_{n_1}, \dots, \|Y^K\| / \sqrt{n_K q} \, \mathrm{Id}_{n_K})$. $\alpha = 3$ is used in the experiments.

Theorem B.1. *The dual formulation of Problem (B.2) is*

$$\hat{\Theta} = \underset{\Theta \in \Delta'_{X,\lambda}}{\arg\max} \ \langle Y, \lambda \Theta \rangle + \sum_{k=1}^K \frac{\underline{\sigma}_k}{2} \left(\frac{n_k}{n} - nq\lambda^2 \|\Theta^k\|^2 \right) \ ,$$

Algorithm B.1 ALTERNATE MIN. FOR THE BLOCK MODEL

input : $X^1, \dots, X^K, Y^1, \dots, Y^K, \underline{\sigma}_1, \dots, \underline{\sigma}_K, \lambda, T$
init : $\mathrm{B} = 0_{p,q}$, $\forall k \in [K], \sigma_k = \|Y^k\|/\sqrt{n_k q}, R^k = Y^k$, $\forall k \in [K], \forall j \in [p], L_{k,j} = \|X_j^k\|_2^2$

for $t = 1, \dots, T$ **do**
  for $j = 1, \dots, p$ **do**
    for $k = 1, \dots, K$ **do**
      $R^k \leftarrow R^k + X_j^k \mathrm{B}_j$ // residual update
    $\mathrm{B}_j \leftarrow \mathrm{BST}\left(\sum_{k=1}^K \frac{X_j^{k\top} R^k}{\sigma_k}, \lambda n q / \sum_{k=1}^K \frac{L_{k,j}}{\sigma_k} \right)$ // block soft-thresholding
    for $k = 1, \dots, K$ **do**
      $R^k \leftarrow R^k - X_j^k \mathrm{B}_j$ // residual update
      $\sigma_k \leftarrow \underline{\sigma}_k \vee \frac{\|R^k\|}{\sqrt{n_k q}}$ // std dev update
return $\mathrm{B}, \sigma_1, \dots, \sigma_k$

where

$$\Delta'_{X,\lambda} \triangleq \left\{ \Theta \in \mathbb{R}^{n \times q} : \|X^\top \Theta\|_{2,\infty} \leq 1, \forall k \in [K], \|\Theta^k\| \leq \tfrac{\sqrt{n_k}}{n\lambda\sqrt{q}} \right\}.$$

Proposition B.2. *When optimizing Problem (B.2) with* B *fixed, then*

$$\hat{S} = \mathrm{diag}(\hat{\sigma}_1 \mathrm{Id}_{n_1}, \dots, \hat{\sigma}_K \mathrm{Id}_{n_K}) \ , \tag{B.4}$$

with residuals $R^k = Y^k - X^k \hat{\mathrm{B}}$ *and* $\hat{\sigma}_k = \underline{\sigma}_k \vee (\|R^k\|/\sqrt{n_k q})$.

Proposition B.3. *For the multi-task SBHCL the critical parameter is*

$$\lambda_{\max} \triangleq \tfrac{1}{nq} \|X^\top S_{\max}^{-1} Y\|_{2,\infty} \ , \tag{B.5}$$

where $S_{\max} = \mathrm{diag}(\sigma_1^{\max} \mathrm{Id}_{n_1}, \dots, \sigma_K^{\max} \mathrm{Id}_{n_K})$ *and* $\forall k \in [K], \sigma_k^{\max} = \underline{\sigma}_k \vee (\|Y^k\|/\sqrt{n_k q})$.

The strategy of Algorithm 4.1 can also be applied to the multi-task SBHCL. Because of the special form of S, the computations are lighter and the standard deviations σ_k's can be updated at each coordinate descent update. Indeed, updating all the σ_k's may seem costly, since a naive implementation requires to recompute all the residual norms $\|R^k\|$, where $R^k = Y^k - X^k \mathrm{B}$, which is $\mathcal{O}(nq)$. However, it is possible to store the values of $\|R^k\|^2$ and update them at each B_j update with a $\mathcal{O}(kq)$ cost. Indeed, if we denote $\tilde{\mathrm{B}}_j$ and $\tilde{R}^k$ the values before the update, we have:

$$\begin{aligned} R^k &= \tilde{R}^k + X_j^{k\top}(\tilde{\mathrm{B}}_j - \mathrm{B}_j) \\ \|R^k\|^2 &= \|\tilde{R}^k\|^2 + 2\langle (\tilde{\mathrm{B}}_j - \mathrm{B}_j), \tilde{R}^{k\top} X_j^k \rangle + \|\tilde{\mathrm{B}}_j - \mathrm{B}_j\|^2 L_{j,k} \end{aligned}$$

and all the quantities $\tilde{R}^{k\top} X_j^k$ are already computed for the soft-thresholding step. As $k \leq n$, this makes the cost of one B_j update of Algorithm B.1 $\mathcal{O}(nq)$, the same cost as for the $\ell_{2,1}$ regularized Lasso, *a.k.a.* multi-task Lasso (MTL) (Obozinski et al., 2010).

B.2 Experiments

B.2.1 Block homoscedastic model illustration

To demonstrate the benefits of handling non-homoscedastic noise, we now present experiments using both simulations and real M/EEG data. First, we show that taking into account multiple noise levels improves both prediction performance and support identification. We then illustrate on M/EEG data that the estimates of the noise standard deviations using multi-task SBHCL match the expected behavior when increasing the SNR of the data. We also demonstrate empirically the benefit of our proposed multi-task SBHCL to reduce the variance of the estimation. The implementation is done in Python/Cython and is available at https://github.com/mathurinm/SHCL.

We consider the case where the block structure of the noise is known by the practitioner. Therefore, all experiments use the block homoscedastic setting. Note that this is relevant with the M/EEG framework where the variability of the noise is due to different data acquisitions sensors that are known.

B.2.2 Prediction performance

We first study the impact of the multi-task SBHCL on prediction performance, evaluated on left-out data.

The experiment setup is as follows. There are $n = 300$ observations, $p = 1000$ features and $q = 100$ tasks. The design X is random with Toeplitz-correlated features with parameter $\rho = 0.7$ (correlation between features i and j is $\rho^{|i-j|}$). The true coefficient matrix B^* has 20 non-zero rows, whose entries are independently and normally (centered and reduced) distributed. We simulate data coming from $K = 3$ sources (each one containing 100 observations) whose respective noise levels are σ^*, $2\sigma^\star$ and $5\sigma^\star$. The standard deviation $\sigma^\star$ is chosen so that the signal-to-noise ratio

$$\mathrm{SNR} \triangleq \|Y\|/\|X\mathrm{B}^*\| = 1 \ .$$

The two estimators are trained for λ varying on a logarithmic grid of 15 values between the critical parameter[1] $\lambda_{\max}$ and $\lambda_{\max}/10$. The training set contains 150 samples ($n_1 = n_2 = n_3 = 50$ of each data source) and the test set consists of the remaining 150.

Figure B.1 shows prediction performance for the Smooth Concomitant Lasso (SCL), which estimates a single noise level for all blocks, and the multi-task SBHCL. Since each block has a different noise level, for each block k and each estimator, we report the Root Mean Squared error (RMSE, $\|Y^k - X^k\hat{\mathrm{B}}\|/\sqrt{qn_k}$) normalized by the oracle RMSE ($\|Y^k - X^k\mathrm{B}^*\|/\sqrt{qn_k}$). After taking the log, zero value means a perfect estimation, a positive value means under-fitting of the block, while a negative value corresponds to over-fitting. Figure B.1 reports normalized RMSE values on both the training and the test data.

As it can be observed, the RMSE for the multi-task SBHCL is lower on every block of the test set, meaning that it has better prediction performance. By attributing a higher noise standard deviation to the noisiest block (block 3), the multi-task SBHCL is able to down-weight the impact of these samples on the estimation, while still benefiting from it.

[1] Note that $\lambda_{\max}$ is model specific

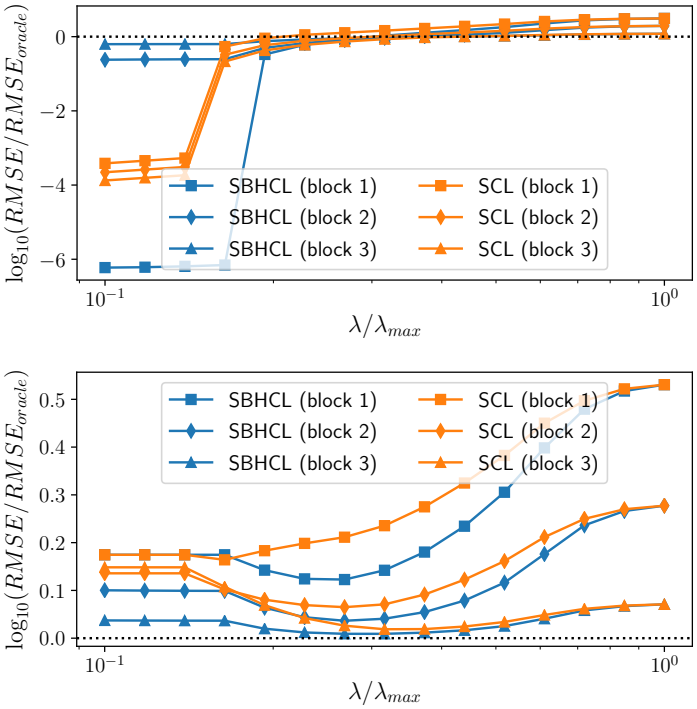


Figure B.1 – RMSE normalized by oracle RMSE, per block, for the multi-task SBHCL and Smooth Concomitant Lasso (SCL), on training (top) and testing (bottom) set, for various values of λ. The flexibility of the block homoscedastic model enables the multi-task Smoothed Block Homoscedastic Concomitant Lasso to reach a lower RMSE on every block of the test set.

While the 3 normalized RMSE have similar behaviors on the test set for the SCL, for low values of λ, the multi-task SBHCL overfits more on the least noisy block. However this does not result in degraded prediction performance on the test set, neither for this block nor for others, and the prediction is even better on the noisiest block. Indeed, the SCL overfits more on the the noisiest block, which has a greater impact on prediction (as overfitting on noiseless data would lead to perfect parameter inference). When the regularization parameter becomes too low, taking into account different noise levels allows our estimator to limit the impact of overfitting by favoring the most reliable source. This experiments shows that our formulation is appealing for parameter selection, as the best left-out prediction is obtained for similar values of λ.

B.2.3 Support recovery

In this experiment, we demonstrate the superior performance of the multi-task SBHCL for support recovery, *i.e.,* its ability to correctly identify the predictive features. The experimental setup is the same as in Appendix B.2.2, except that the support of B^* is of size 50. We also vary $\rho \in \{0.1, 0.9\}$ and the SNR $\in \{1, 5\}$ (additional results are included in Appendix A.1).

The five estimators compared on Figure B.2 are the multi-task SBHCL, the SCL, the MTL, and also the MTL and the SCL trained on the least noisy block (*i.e.,* the most favorable block). Following the empirical evaluation from (Bühlmann and Mandozzi, 2014), the figure of merit is the ROC curve, *i.e.,* the true positive rate as a function of

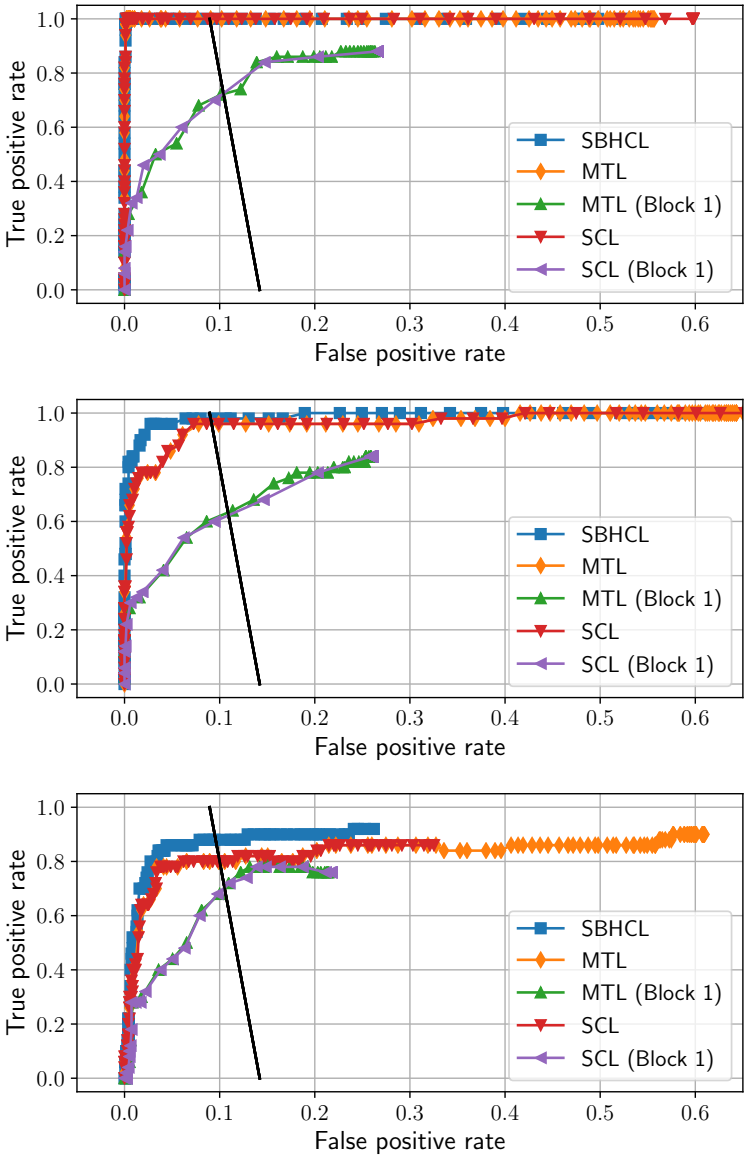


Figure B.2 – ROC curves of true support recovery for the SBHCL, the MTL and SCL on all blocks, and the MTL and SCL on the least noisy block. The black curve marks the limit of supports of size $0.9n$. Top: SNR $= 5$, $\rho = 0.1$, middle: SNR$=1$, $\rho = 0.1$, bottom: SNR $= 1$, $\rho = 0.9$.

the false positive rate. The curve is obtained by varying the value of λ (lower values leading to larger predicted support and therefore potentially more false positives).

We can see that when the SNR is sufficiently high (top graph with $SNR = 5$), the multi-task SBHCL, the SCL and the MTL successfully recover the true support, while the MTL or SCL trained on the least noisy block with only one third of the data fails. However, when the SNR is lower (middle graph with SNR $= 1$), the multi-task SBHCL still achieves almost perfect support identification, while the performance of the MTL and SCL decreases. The performance is naturally even worse when using only one block of samples. Finally, when the features are more correlated ($\rho = 0.9$) and the conditioning of X is degraded, the multi-task SBHCL, despite not perfectly recovering

Table B.1 – Mean values of pAUC for the main estimators, across ten simulations of X and Y.

SNR	1	1	5
ρ	0.1	0.9	0.1
SBHCL	0.92 ± 0.12	0.86 ± 0.05	0.98 ± 0.02
MTL	0.79 ± 0.08	0.71 ± 0.07	0.99 ± 0.00
MTL (block 1)	0.44 ± 0.04	0.48 ± 0.05	0.48 ± 0.04

the true support, still has superior performance. Note also that unsurprisingly the MTL and the SCL lead to almost perfectly the same ROC curves as both estimators (if $\underline{\sigma}$ is small enough) have the same solution path. Any difference between SCL and MTL in our graph is due to the choice of a discrete set of λ values.

To study the stability of Figure B.2, we repeat the simulation 10 times. Since the curves are not guaranteed to reach TPR = 1, it is not possible to use AUC as a scalar figure of merit. As we are usually interested in sparse estimators when the recovered support is small, we follow Bühlmann and Mandozzi (2014, Fig. 1-4), and limit the study to estimated supports of size inferior to $0.9n$ (*i.e.,* the part to the left of the black curve). This leads to the use of pAUC or "0.9–performance": the area under the ROC curve, but restricted to the left of the black line, and normalized by its maximal value. The mean pAUCs for 10 repetitions, for all estimators in the different settings are in Table B.1.

B.2.4 Results on joint M/EEG real data

We now evaluate our estimator on magneto- and electroencephalography (M/EEG) data. The data consists of M/EEG recordings, which measure the electric potential and magnetic field induced by active neurons. Data are time-series so that n corresponds to the number of sensors and q to the number of consecutive time instants in the data. Thanks to their high temporal resolution, M/EEG help to elucidate where and precisely when cognitive processes happen in the brain. The so-called M/EEG inverse problem, which consists in identifying active brain regions, can be cast as a high-dimensional sparse regression problem. Because of the limited number of sensors, as well as the physics of the problem, this problem is severely ill-posed, and regularization is needed to provide solutions which are both biologically plausible and robust to measurement noise (Wipf et al., 2008; Haufe et al., 2008; Gramfort et al., 2013). As foci of neural activity are observed from a distance by M/EEG and since only a small number of brain regions are involved in a cognitive task during a short time interval, it is common to employ sparsity-promoting regularizations. Amongst these, the ℓ_1/ℓ_2 penalty has been successfully applied to the M/EEG inverse problem in either time (Ou et al., 2009) or frequency domain (Gramfort et al., 2013).

The experimental condition considered is a monaural auditory stimulation in the right ear of the subject. The same subject undergoes the same stimulation 61 times, and the M/EEG measurements are recorded from 0.2 s before to 0.5 s after the stimulus. The data (from the MNE software (Gramfort et al., 2014)) thus contains 61 repetitions (*trials*) of this stimulation.

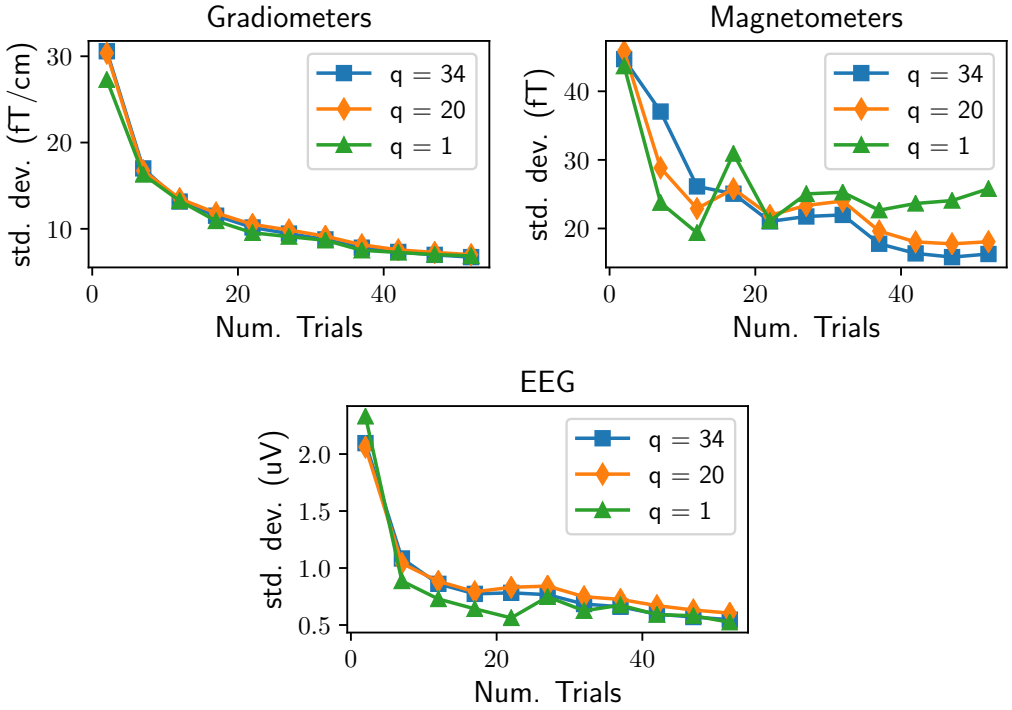


Figure B.3 – Noise standard deviation estimated on auditory data for $q = 1$, $q = 20$ and $q = 34$ time instants using the SBHCL estimator. Data consist of combined MEG gradiometers ($n_1 = 203$ sensors) and magnetometers ($n_2 = 102$ sensors), as well as EEG ($n_3 = 59$ sensors). We used $\lambda = 0.03\lambda_{\max}$.

In the experimental setup we have 204 gradiometers, 102 magnetometers and 60 EEG electrodes. We have discarded one magnetometer and one electrode corrupted by strong artifacts. We therefore have $K = 3$ sensor types with $n_1 = 203$, $n_2 = 102$ and $n_3 = 59$ (so $n = 364$). X is obtained by numerically solving the M/EEG forward problem using $p = 1884$ candidate sources distributed over the cortical surface ($X \in \mathbb{R}^{364\times1884}$). The orientations of the dipoles are assumed known and normal to the cortical mantle.

The measurements for $q = 1$ (single time measurements) are selected 75 ms after the stimulus onset, and between 60 and 115 ms (*resp.* 70 and 102 ms) after the stimulus for $q = 34$ (resp $q = 20$). This time interval corresponds to the main cortical response to the auditory stimulation.

For a number t of repetitions of experiment (t ranging from 2 to 56), we create an observation matrix Y_t by averaging the first t trials. By doing so, the noise standard deviations of each block should be proportional to $1/\sqrt{t}$. We then run the multi-task SBHCL with fixed λ, equal to 3% of $\lambda_{\max}$. Figure B.3 shows the noise standard deviation estimated by the multi-task SBHCL, when ran on a single time instant (single task), 20 and 34 time instants.

We can see that the estimated values are plausible: they have the correct orders of magnitude, as well as the expected $1/\sqrt{t}$ decrease. We also see that taking more tasks into account leads to more stable noise estimation: for magnetometers, the curve is smoother for $q = 34$ than for $q = 20$ and $q = 1$. Indeed, using more tasks reduces the variance of the estimation.

B.3 Statistical properties

B.3.1 Statistical comparison

In this subsection, we show the statistical interest of using all repetitions of the experiments instead of using a mere averaging as SGCL would do (recall that the later is equivalent to CLaR with $r = 1$ and $Y^1 = \bar{Y}$, see Remark 4.6).

Let us introduce Σ^*, the true covariance matrix of the noise (*i.e.,* $\Sigma^* = S^{*2}$ with our notation). In SGCL and CLaR alternate minimization consists in a succession of estimations of B^* and Σ^* (more precisely $S = \mathrm{ClSqrt}(\Sigma, \underline{\sigma})$ is estimated along the process). In this section we explain why the estimation of Σ^* provided by CLaR has better statistical properties than that of SGCL. For that, we can compare the estimates of Σ^* one would obtain provided that the true parameter B^* is known by both SGCL and CLaR. In such "ideal" scenario, the associated estimators of Σ^* could be written:

$$\hat{\Sigma}^{\mathrm{CLaR}} \triangleq \frac{1}{qr}\sum_{l=1}^{r}(Y^l - X\hat{\mathrm{B}})(Y^l - X\hat{\mathrm{B}})^\top \ , \tag{B.6}$$

$$\hat{\Sigma}^{\mathrm{SGCL}} \triangleq \frac{1}{qr}\Big(\sum_{l=1}^{r} Y^l - X\hat{\mathrm{B}}\Big)\Big(\sum_{l=1}^{r} Y^l - X\hat{\mathrm{B}}\Big)^\top , \tag{B.7}$$

with $\hat{\mathrm{B}} = \mathrm{B}^*$, and satisfy the following properties:

Proposition B.4. *Provided that the true signal is known, and that the covariance estimator* $\hat{\Sigma}^{\mathrm{CLaR}}$ *and* $\hat{\Sigma}^{\mathrm{SGCL}}$ *are defined thanks to Equations* (B.6) *and* (B.7)*, then one can check that*

$$\mathbb{E}(\hat{\Sigma}^{\mathrm{CLaR}}) = \mathbb{E}(\hat{\Sigma}^{\mathrm{SGCL}}) = \Sigma^* \ , \tag{B.8}$$

$$\mathrm{cov}(\hat{\Sigma}^{\mathrm{CLaR}}) = \frac{1}{r}\,\mathrm{cov}(\hat{\Sigma}^{\mathrm{SGCL}}) \ . \tag{B.9}$$

Proposition B.4 states that $\hat{\Sigma}^{\mathrm{CLaR}}$ and $\hat{\Sigma}^{\mathrm{SGCL}}$ are unbiased estimators of Σ^* but our newly introduced CLaR, improves the estimation of the covariance structure by a factor r, the number of repetitions performed.

Empirically[2], we have also observed that $\hat{\Sigma}^{\mathrm{CLaR}}$ has larger eigenvalues than $\hat{\Sigma}^{\mathrm{SGCL}}$, leading to a less biased estimation of S^* after clipping the singular values.

Let us recall that:

$$\Sigma^{\mathrm{SGCL}} = \frac{1}{qr}\Big(\sum_{l=1}^{r} R^l\Big)\Big(\sum_{l=1}^{r} R^l\Big)^\top \qquad \text{and} \qquad \Sigma^{\mathrm{CLaR}} = \frac{1}{qr}\sum_{l=1}^{r} R^l R^{l\top} \ . \tag{B.10}$$

Proof If $\mathrm{B} = \mathrm{B}^*$, $R^l = S^*\mathrm{E}^l$, where E^l are random matrices with normal i.i.d. entries, and the result trivially follows.

If $\hat{\mathrm{B}} = \mathrm{B}^*$, $Y^l - X\hat{\mathrm{B}} = S^*\mathrm{E}^l$, where the E^l's are random matrices with normal i.i.d. entries.

Now, on the one hand:

$$\hat{\Sigma}^{\mathrm{SGCL}} = \frac{1}{qr}\Big(\sum_{l=1}^{r} S^*\mathrm{E}^l\Big)\Big(\sum_{l=1}^{r} S^*\mathrm{E}^l\Big)^\top \ .$$

[2]In that case we plug $\hat{\mathrm{B}} = \hat{\mathrm{B}}^{\mathrm{CLaR}}$ (*resp.* $\hat{\mathrm{B}} = \hat{\mathrm{B}}^{\mathrm{CLaR}}$) in Proposition B.4.

Since $\frac{1}{\sqrt{r}} \sum_{l=1}^{r} S^* \mathrm{E}^l \underset{law}{\sim} S^* \mathrm{E}$ it follows that

$$\hat{\Sigma}^{\mathrm{SGCL}} \underset{law}{\sim} \frac{1}{q} S^* \mathrm{E} (S^* \mathrm{E})^\top,$$
$$\mathrm{cov}(\hat{\Sigma}^{\mathrm{SGCL}}) = \frac{1}{q^2} \mathrm{cov}(S^* \mathrm{E} (S^* \mathrm{E})^\top) \ .$$

On the other hand:

$$\hat{\Sigma}^{\mathrm{CLaR}} = \frac{1}{qr} \sum_{l=1}^{r} S^* \mathrm{E}^l (S^* \mathrm{E}^l)^\top \ .$$

Since the E^l's are independent it follows that:

$$\begin{aligned} \mathrm{cov}(\hat{\Sigma}^{\mathrm{CLaR}}) &= \frac{1}{r^2 q^2} \sum_{l=1}^{r} \mathrm{cov}\left(S^* \mathrm{E}^l (S^* \mathrm{E}^l)^\top \right) = \frac{1}{r^2 q^2} \sum_{l=1}^{r} \mathrm{cov}\left(S^* \mathrm{E} (S^* \mathrm{E})^\top \right) \\ &= \frac{1}{r q^2} \mathrm{cov}\left(S^* \mathrm{E} (S^* \mathrm{E})^\top \right) = \frac{1}{r} \mathrm{cov}\left(\hat{\Sigma}^{\mathrm{SGCL}} \right) \ . \end{aligned}$$

■

Bibliography

P. Ablin, J.-F. Cardoso, and A. Gramfort. Faster independent component analysis by preconditioning with Hessian approximations. *IEEE Trans. Signal Process.*, 66(15): 4040–4049, 2018. page 33

A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. In *NeurIPS*, pages 41–48, 2006. page 24

Ü. Aydin, J. Vorwerk, M. Dümpelmann, P. Küpper, H. Kugel, M. Heers, J. Wellmer, C. Kellinghaus, J. Haueisen, S. Rampp, et al. Combined EEG/MEG can outperform single modality EEG or MEG source reconstruction in presurgical epilepsy diagnosis. *PloS one*, 10(3):e0118753, 2015. page 33

F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Convex optimization with sparsity-inducing norms. *Foundations and Trends in Machine Learning*, 4(1):1–106, 2012. pages 28, 44

H. H. Bauschke and P. L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. Springer, New York, 2011. pages 26, 27, 53

A. Beck. *First-Order Methods in Optimization*, volume 25. SIAM, 2017. pages 99, 100

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009. pages 24, 29, 45

A. Beck and M. Teboulle. *Gradient-based algorithms with applications to signal-recovery problems*, pages 42–88. Cambridge Univ. Press, Cambridge, 2010. page 24

A. Beck and M. Teboulle. Smoothing and first order methods: A unified framework. *SIAM J. Optim.*, 22(2):557–580, 2012. page 97

S. R. Becker, E. J. Candès, and M. C. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Math. Program. Comput.*, 3(3):165–218, 2011. page 86

S. Behnel, R. Bradshaw, C. Citro, L. Dalcin, D. S. Seljebotn, and K. Smith. Cython: The best of both worlds. *Computing in Science Engineering*, 13(2):31 –39, 2011. pages 57, 78

A. Belloni, V. Chernozhukov, and L. Wang. Square-root Lasso: pivotal recovery of sparse signals via conic programming. *Biometrika*, 98(4):791–806, 2011. pages 24, 25, 66, 86, 88, 125

Q. Bertrand, **M. Massias**, A. Gramfort, and J. Salmon. Handling correlated and repeated measurements with the smoothed multivariate square-root Lasso. In *NeurIPS*, 2019. pages 90, 103

P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of Lasso and Dantzig selector. *Ann. Statist.*, 37(4):1705–1732, 2009. pages 24, 44, 86

K. Bleakley and J.-P. Vert. The group fused lasso for multiple change-point detection. *arXiv preprint arXiv:1106.4199*, 2011. page 26

A. Boisbunon, R. Flamary, and A. Rakotomamonjy. Active set strategy for high-dimensional non-convex sparse optimization problems. In *ICASSP*, pages 1517–1521, 2014. pages 44, 75

R. Bollapragada, D. Scieur, and A. d'Aspremont. Nonlinear acceleration of momentum and primal-dual algorithms. *ArXiv e-print*, abs/1810.04539, 2018. page 51

A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval. A dynamic screening principle for the lasso. In *EUSIPCO*, 2014. pages 55, 66

A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval. Dynamic screening: accelerating first-order algorithms for the Lasso and Group-Lasso. *IEEE Trans. Signal Process.*, 63(19):20, 2015. page 55

S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004. pages 68, 91, 105

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011. page 24

P. Bühlmann and J. Mandozzi. High-dimensional variable screening and bias in subsequent inference, with an empirical comparison. *Computational Statistics*, 29(3): 407–430, Jun 2014. pages 128, 130

L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. Vanderplas, A. Joly, B. Holt, and G. Varoquaux. API design for machine learning software: experiences from the scikit-learn project. *arXiv e-prints*, 2013. page 78

E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory*, 52(2):489–509, 2006. pages 24, 26

E. J. Candès, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted l_1 minimization. *J. Fourier Anal. Applicat.*, 14(5-6):877–905, 2008. pages 24, 25

A. Chambolle and T. Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.*, 40(1):120–145, 2011. pages 24, 101

R. Chartrand. Exact reconstruction of sparse signals via nonconvex minimization. *IEEE Trans. Signal Process. Lett.*, 14(10):707–710, 2007. page 24

S. Chen and A. Banerjee. Alternating estimation for structured high-dimensional multi-response models. In *NeurIPS*, pages 2838–2848, 2017. pages 87, 104, 107

S. S. Chen and D. L. Donoho. Atomic decomposition by basis pursuit. In *SPIE*, 1995. pages 24, 44

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.*, 20(1):33–61, 1998. page 23

J. F. Claerbout and F. Muir. Robust modeling with erratic data. *Geophysics*, 38(5): 826–844, 1973. page 23

D. Cohen. Magnetoencephalography: evidence of magnetic fields produced by alpha-rhythm currents. *Science*, 161(3843):784–786, 1968. page 33

P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, volume 49 of *Springer Optim. Appl.*, pages 185–212. Springer, New York, 2011. pages 24, 29

A. M. Dale, A. K. Liu, B. R. Fischl, R. L. Buckner, J. W. Belliveau, J. D. Lewine, and E. Halgren. Dynamic statistical parametric mapping: combining fMRI and MEG for high-resolution imaging of cortical activity. *Neuron*, 26(1):55–67, 2000. page 36

J. Daye, J. Chen, and H. Li. High-dimensional heteroscedastic regression with an application to eQTL data analysis. *Biometrics*, 68(1):316–326, 2012. page 86

D. L. Donoho. Compressed sensing. *IEEE Trans. Inf. Theory*, 52(4):1289–1306, 2006. pages 24, 26

D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994. page 23

J. H. Duyn. The future of ultra-high field MRI and fMRI for study of the human brain. *Neuroimage*, 62(2):1241–1248, 2012. page 31

R. L. Dykstra. An algorithm for restricted least squares regression. *J. Amer. Statist. Assoc.*, 78(384):837–842, 1983. page 53

M. A. Efroymson. Multiple regression analysis. In *Mathematical methods for digital computers*, pages 191–203. Wiley, New York, 1960. page 23

L. El Ghaoui, V. Viallon, and T. Rabbani. Safe feature elimination in sparse supervised learning. *J. Pacific Optim.*, 8(4):667–698, 2012. pages 44, 66, 86, 94

D. A. Engemann and A. Gramfort. Automated model selection in covariance estimation and spatial whitening of MEG and EEG signals. *NeuroImage*, 108:328–342, 2015a. page 87

D.-A. Engemann and A. Gramfort. Automated model selection in covariance estimation and spatial whitening of MEG and EEG signals. *NeuroImage*, 108:328–342, 2015b. page 34

D.-A. Engemann, D. Strohmeier, E. Larson, and A. Gramfort. Mind the noise covariance when localizing brain sources with M/EEG. In *Pattern Recognition in NeuroImaging (PRNI)*, pages 9–12, 2015. page 34

R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, 2008. pages 44, 61, 66

J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Amer. Statist. Assoc.*, 96(456):1348–1360, 2001. page 24

J. Fan and J. Lv. Sure independence screening for ultrahigh dimensional feature space. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 70(5):849–911, 2008. pages 66, 94

M. Fazel. *Matrix rank minimization with applications*. PhD thesis, Stanford University, 2002. page 24

O. Fercoq and P. Richtárik. Accelerated, parallel and proximal coordinate descent. *SIAM J. Optim.*, 25(3):1997 – 2013, 2015. pages 60, 66, 67

O. Fercoq, A. Gramfort, and J. Salmon. Mind the duality gap: safer rules for the lasso. In *ICML*, pages 333–342, 2015. pages 44, 55, 66, 68, 86

L. E. Frank and J. H. Friedman. A statistical view of some chemometrics regression tools. *Technometrics*, 35(2):109–135, 1993. page 24

J. Friedman, T. J. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *Ann. Appl. Stat.*, 1(2):302–332, 2007. pages 45, 66

J. Friedman, T. J. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008. pages 24, 105, 106

J. Friedman, T. J. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.*, 33(1):1, 2010. pages 44, 60, 66, 78

W. J. Fu. Penalized regressions: the bridge versus the lasso. *J. Comput. Graph. Statist.*, 7(3):397–416, 1998. page 44

G. Gasso, A. Rakotomamonjy, and S. Canu. Recovering sparse signals with a certain family of nonconvex penalties and DC programming. *IEEE Trans. Signal Process.*, 57(12):4686–4698, 2009. page 25

C. F. Gauss. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium*, volume 7. Perthes et Besser, 1809. page 21

P. Gloor. Neuronal generators and the problem of localization in electroencephalography: application of volume conductor theory to electroencephalography. *Journal of clinical neurophysiology*, 2(4):327–354, 1985. page 31

J. L. Goffin. On convergence rates of subgradient optimization methods. *Mathematical Programming*, 13(1):329–347, 1977. page 29

A. Gramfort, M. Kowalski, and M. Hämäläinen. Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods. *Phys. Med. Biol.*, 57(7):1937–1961, 2012. pages 36, 81

A. Gramfort, D. Strohmeier, J. Haueisen, M. S. Hämäläinen, and M. Kowalski. Time-frequency mixed-norm estimates: Sparse M/EEG imaging with non-stationary source activations. *NeuroImage*, 70:410–422, 2013. pages 36, 87, 130

A. Gramfort, M. Luessi, E. Larson, D. A. Engemann, D. Strohmeier, C. Brodbeck, L. Parkkonen, and M. S. Hämäläinen. MNE software for processing MEG and EEG data. *NeuroImage*, 86:446 – 460, 2014. pages 81, 110, 130

J. Gross, S. Baillet, G. R. Barnes, R. N. Henson, A. Hillebrand, O. Jensen, K. Jerbi, V. Litvak, B. Maess, R. Oostenveld, et al. Good practice for conducting and reporting MEG research. *NeuroImage*, 65:349–363, 2013. page 33

E. Hale, W. Yin, and Y. Zhang. Fixed-point continuation for ℓ_1-minimization: Methodology and convergence. *SIAM J. Optim.*, 19(3):1107–1130, 2008. pages 69, 70, 71

T. J. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer, New York, second edition, 2009. page 17

T. J. Hastie, A. Montanari, S. Rosset, and R. J. Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*, 2019. page 22

S. Haufe, V. V. Nikulin, A. Ziehe, K.-R. Müller, and G. Nolte. Combining sparsity and rotational invariance in EEG/MEG source reconstruction. *NeuroImage*, 42(2): 726–738, 2008. pages 36, 130

J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex analysis and minimization algorithms. II*, volume 306. Springer-Verlag, Berlin, 1993. page 27

A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. page 22

B. Hong, W. Liu, J. Ye, D. Cai, X. He, and J. Wang. Scaling up sparse support vector machines by simultaneous feature and sample reduction. *J. Mach. Learn. Res.*, 20 (121):1–39, 2019. page 44

C.-J Hsieh, M. Sustik, I. Dhillon, and P. Ravikumar. QUIC: Quadratic approximation for sparse inverse covariance estimation. *J. Mach. Learn. Res.*, 15:2911–2947, 2014. pages 75, 78

J. Huang, P. Breheny, and S. Ma. A selective review of group selection in high-dimensional models. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 27(4), 2012. page 24

M Huang, C. J. Aine, S. Supek, E. Best, D. Ranken, and E. R. Flynn. Multi-start downhill simplex method for spatio-temporal source localization in magnetoencephalography. *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, 108(1):32–44, 1998. page 35

L. Huber, D. A. Handwerker, D. C. Jangraw, G. Chen, A. Hall, C. Stüber, J. Gonzalez-Castillo, D. Ivanov, S. Marrett, M. Guidi, et al. High-resolution CBV-fMRI allows mapping of laminar activity and connectivity of cortical input and output in human M1. *Neuron*, 96(6):1253–1263, 2017. page 31

P. J. Huber. *Robust Statistics*. John Wiley & Sons Inc., 1981. page 86

P. J. Huber and R. Dutter. Numerical solution of robust regression problems. In *Compstat 1974 (Proc. Sympos. Computational Statist., Univ. Vienna, Vienna, 1974)*, pages 165–172. Physica Verlag, Vienna, 1974. page 86

M. Hämäläinen and R. J. Ilmoniemi. Interpreting magnetic fields of the brain: minimum norm estimates. *Medical & biological engineering & computing*, 32(1):35–42, 1994. page 35

A. A. Ivanov. *The theory of approximate methods and their applications to the numerical solution of singular integral equations*, volume 2. Springer Science & Business Media, 1976. page 22

T. B. Johnson. *Scaling Machine Learning via Prioritized Optimization*. PhD thesis, University of Washington, 2018. page 60

T. B. Johnson and C. Guestrin. Blitz: A principled meta-algorithm for scaling sparse optimization. In *ICML*, pages 1171–1179, 2015. pages 44, 66, 75, 94

T. B. Johnson and C. Guestrin. A fast, principled working set algorithm for exploiting piecewise linear structure in convex problems. *arXiv preprint arXiv:1807.08046*, 2018. pages 66, 75

P. Karimireddy, A. Koloskova, S. Stich, and M. Jaggi. Efficient greedy coordinate descent for composite problems. *arXiv preprint arXiv:1810.06999*, 2018. page 66

S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An interior-point method for large-scale ℓ_1-regularized least squares. *IEEE J. Sel. Topics Signal Process.*, 1(4): 606–617, 2007. pages 45, 60

J. Kim and H. Park. Fast active-set-type algorithms for l1-regularized linear regression. In *AISTATS*, pages 397–404, 2010. page 56

K. Kobayashi, T. Akiyama, T. Nakahori, H. Yoshinaga, and J. Gotman. Systematic source estimation of spikes by a combination of independent component analysis and RAP-MUSIC: I: Principles and simulation study. *Clinical Neurophysiology*, 113(5): 713–724, 2002. page 35

K. Koh, S.-J. Kim, and S. Boyd. An interior-point method for large-scale l1-regularized logistic regression. *J. Mach. Learn. Res.*, 8(8):1519–1555, 2007. pages 24, 66

M. Kolar and J. Sharpnack. Variance function estimation in high-dimensions. In *ICML*, pages 1447–1454, 2012. page 86

Z. J. Koles and A. C. Soong. EEG source localization: implementing the spatio-temporal decomposition approach. *Electroencephalography and clinical Neurophysiology*, 107 (5):343–352, 1998. page 35

M. Kowalski, P. Weiss, A. Gramfort, and S. Anthoine. Accelerating ISTA with an active set strategy. In *OPT 2011: 4th International Workshop on Optimization for Machine Learning*, page 7, 2011. pages 66, 74

S. K. Lam, A. Pitrou, and S. Seibert. Numba: A LLVM-based Python JIT Compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, pages 1–6. ACM, 2015. pages 78, 107

O. Ledoit and M. Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *J. Multivariate Anal.*, 88(2):365–411, 2004. pages 34, 89

J. Lee, Y. Sun, and M. Saunders. Proximal Newton-type methods for convex optimization. In *NeurIPS*, pages 827–835, 2012. page 75

W. Lee and Y. Liu. Simultaneous multiple response regression and inverse covariance matrix estimation via penalized Gaussian maximum likelihood. *Journal of multivariate analysis*, 111:241–255, 2012. pages 87, 107

A.-M. Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes*. F. Didot, 1805. page 21

C. Leng, Y. Lin, and G. Wahba. A note on the lasso and related procedures in model selection. *Statistica Sinica*, 16(4):1273, 2006. page 25

K. Lounici. *Estimation statistique en grande dimension, parcimonie et inégalités d'oracle*. PhD thesis, Université Paris Diderot, 2009. page 25

F. Lucka, S. Pursiainen, M. Burger, and C. H. Wolters. Hierarchical Bayesian inference for the EEG inverse problem using realistic FE head models: depth localization and source separation for focal primary currents. *Neuroimage*, 61(4):1364–1382, 2012. page 36

J. Mairal. *Sparse coding for machine learning, image processing and computer vision*. PhD thesis, École normale supérieure de Cachan, 2010. pages 46, 67, 69

S. Makeig, A. J. Bell, T.-Z. Jung, and T. J. Sejnowski. Independent component analysis of electroencephalographic data. In *NeurIPS*, pages 145–151, 1996. page 33

S. Mallat and Z. Zhang. Matching pursuit with time-frequency dictionaries. *IEEE Trans. Image Process.*, 41:3397–3415, 1993. page 23

H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1):77–91, 1952. page 23

M. Massias, A. Gramfort, and J. Salmon. From safe screening rules to working sets for faster lasso-type solvers. In *NIPS-OPT workshop*, 2017.

M. Massias, O. Fercoq, A. Gramfort, and J. Salmon. Generalized concomitant multi-task Lasso for sparse multimodal regression. In *AISTATS*, pages 998–1007, 2018a. pages 89, 103, 107

M. Massias, A. Gramfort, and J. Salmon. Celer: a fast solver for the Lasso with dual extrapolation. In *ICML*, pages 3321–3330, 2018b.

M. Massias, S. Vaiter, A. Gramfort, and J. Salmon. Dual extrapolation for sparse Generalized Linear Models. *arXiv preprint arXiv:1907.05830*, 2019.

K. Matsuura and Y. Okabe. Selective minimum-norm solution of the biomagnetic inverse problem. *IEEE Transactions on Biomedical Engineering*, 42(6):608–615, 1995. page 36

P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman and Hall/CRC Monographs on Statistics and Applied Probability Series. Taylor & Francis, second edition, 1989. pages 18, 66

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *Ann. Statist.*, 34(3):1436–1462, 2006. page 25

A. Miller. *Subset selection in regression*. Chapman and Hall/CRC, 2002. page 23

A. J. Molstad. Insights and algorithms for the multivariate square-root lasso. *arXiv preprint arXiv:1909.05041*, 2019. pages 96, 97

J.-J. Moreau. Proximité et dualité dans un espace hilbertien. *Bull. Soc. Math. France*, 93:273–299, 1965. page 27

V. A. Morozov. *Methods for solving incorrectly posed problems*, volume 2. Springer-Verlag, 1984. page 22

J. C. Mosher and R. M. Leahy. Source localization using recursively applied and projected (RAP) MUSIC. *IEEE Trans. Signal Process.*, 47(2):332–340, 1999. page 35

J. C. Mosher, S. Baillet, and R. M. Leahy. EEG source localization and imaging using multiple signal classification approaches. *Journal of Clinical Neurophysiology*, 16(3): 225–238, 1999. page 35

S. Murakami and Y. Okada. Invariance in current dipole moment density across brain structures and species: Physiological constraint for neuroimaging. *NeuroImage*, 111: 49–58, 2015. page 32

D. Myers and W. Shih. A constraint selection technique for a class of linear programs. *Operations Research Letters*, 7(4):191–195, 1988. page 44

Y. Nardi and A. Rinaldo. Autoregressive process modeling via the lasso procedure. *Journal of Multivariate Analysis*, 102(3):528–549, 2011. page 26

B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 24(2):227–234, 1995. page 23

E. Ndiaye, O. Fercoq, A. Gramfort, and J. Salmon. Gap safe screening rules for sparse multi-task and multi-class models. In *NeurIPS*, pages 811–819, 2015. page 81

E. Ndiaye, O. Fercoq, A. Gramfort, V. Leclère, and J. Salmon. Efficient smoothed concomitant lasso estimation for high dimensional regression. *Journal of Physics: Conference Series*, 904(1):012006, 2017a. pages 86, 88, 101

E. Ndiaye, O. Fercoq, A. Gramfort, and J. Salmon. Gap safe screening rules for sparsity enforcing penalties. *JMLR*, 18(128):1–33, 2017b. pages 44, 55, 61, 66, 68, 94

S. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *ArXiv e-prints*, 2010. page 24

Y. Nesterov. A method for solving a convex programming problem with rate of convergence $O(1/k^2)$. *Soviet Math. Doklady*, 269(3):543–547, 1983. pages 24, 29

Y. Nesterov. Smooth minimization of non-smooth functions. *Math. Program.*, 103(1): 127–152, 2005. pages 88, 97

M. Nikolova. Relationship between the optimal solutions of least squares regularized with ℓ_0-norm and constrained by k-sparsity. *Applied and Computational Harmonic Analysis*, 41(1):237–265, 2016. page 24

P. L. Nunez and R. Srinivasan. *Electric fields of the brain: the neurophysics of EEG*. Oxford university press, 2006. page 32

J. Nutini, M. Schmidt, and W. Hare. "Active-set complexity" of proximal gradient: how long does it take to find the sparsity pattern? *Optimization Letters*, pages 1–11, 2017. page 69

G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, 20(2):231–252, 2010. pages 24, 36, 66, 103, 104, 107, 126

P. Ochs, A. Dosovitskiy, T. Brox, and T. Pock. On iteratively reweighted algorithms for nonsmooth nonconvex optimization in computer vision. *SIAM J. Imaging Sci.*, 8 (1):331–372, 2015. page 25

W. Ockham. *Quaestiones et decisiones in quatuor libros Sententiarum cum centilogio theologico*. 1319. page 22

K. Ogawa, Y. Suzuki, and I. Takeuchi. Safe screening of non-support vectors in pathwise SVM computation. In *ICML*, pages 1382–1390, 2013. pages 44, 66

B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research*, 37(23):3311–3325, 1997. page 26

W. Ou, M. Hämaläinen, and P. Golland. A distributed spatio-temporal EEG/MEG inverse solver. *NeuroImage*, 44(3):932–946, Feb 2009. pages 36, 87, 130

A. B. Owen. A robust hybrid of lasso and ridge regression. *Cont. Math.*, 443:59–72, 2007. pages 25, 86, 88

F. Palacios-Gomez, L. Lasdon, and M. Engquist. Nonlinear optimization by successive linear programming. *Management Science*, 28(10):1106–1120, 1982. page 44

N. Parikh, S. Boyd, E. Chu, B. Peleato, and J. Eckstein. Proximal algorithms. *Foundations and Trends in Machine Learning*, 1(3):1–108, 2013. page 29

L. Parkonnen. *MEG: An Introduction to Methods*. Oxford university press, 2010. page 33

R. Pascual-Marqui et al. Standardized low-resolution brain electromagnetic tomography (sLORETA): technical details. *Methods Find. Exp. Clin. Pharmacol.*, 24:5–12, 2002. page 36

Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Proceedings of 27th Asilomar conference on signals, systems and computers*, pages 40–44, 1993. page 23

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011. pages 59, 78, 106

D. Perekrestenko, V. Cevher, and M. Jaggi. Faster coordinate descent via adaptive importance sampling. In *AISTATS*, pages 869–877, 2017. pages 56, 66

E. J. G. Pitman. Sufficient statistics and intrinsic accuracy. In *Mathematical Proceedings of the Cambridge Philosophical society*, volume 32, pages 567–579, 1936. page 19

R. L. Plackett. Studies in the History of Probability and Statistics. XXIX: The discovery of the method of least squares. *Biometrika*, 59(2):239–251, 1972. page 21

C. Poon, J. Liang, and C. Schoenlieb. Local convergence properties of SAGA/Prox-SVRG and acceleration. In *ICML*, pages 4124–4132, 2018. page 69

P. Rai, A. Kumar, and H. Daume. Simultaneously leveraging output and task structures for multiple-output regression. In *NeurIPS*, pages 3185–3193, 2012. page 87

S. Ramon y Cajal. *Comparative study of the sensory areas of the human cortex*. 1899. page 32

P. Richtárik and M. Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1-2): 1–38, 2014. page 66

S. E. Robinson and J. Vrba. Recent advances in biomagnetism. 1999. page 35

R. T. Rockafellar. *Convex analysis*. Princeton Landmarks in Mathematics. Princeton University Press, Princeton, NJ, 1997. page 28

S. Rosset, J. Zhu, and T. Hastie. Boosting as a regularized path to a maximum margin classifier. *J. Mach. Learn. Res.*, 5:941–973, 2004. page 69

V. Roth and B. Fischer. The group-lasso for generalized linear models: uniqueness of solutions and efficient algorithms. In *ICML*, pages 848–855, 2008. page 66

A. J. Rothman, E. Levina, and J. Zhu. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962, 2010. pages 87, 103, 105, 107

M. De Santis, S. Lucidi, and F. Rinaldi. A fast active set block coordinate descent algorithm for ℓ_1-regularized least squares. *SIAM J. Optim.*, 26(1):781–809, 2016. page 75

F. Santosa and W. W. Symes. Linear inversion of band-limited reflection seismograms. *SIAM Journal on Scientific and Statistical Computing*, 7(4):1307–1330, 1986. page 23

K. Scheinberg and X. Tang. Complexity of inexact proximal Newton methods. *arXiv preprint arxiv:1311.6547*, 2013. page 75

M. Scherg and D. von Cramon. Two bilateral sources of the late AEP as identified by a spatio-temporal dipole model. *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, 62(1):352–44, 1985. page 35

D. Scieur. *Acceleration in Optimization*. PhD thesis, École normale supérieure, 2018. page 47

D. Scieur, A. d'Aspremont, and F. Bach. Regularized nonlinear acceleration. In *NeurIPS*, pages 712–720, 2016. pages 47, 51, 58, 67, 121

S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014. page 17

A. Shibagaki, M. Karasuyama, K. Hatano, and I. Takeuchi. Simultaneous safe screening of features and samples in doubly sparse modeling. In *ICML*, pages 1577–1586, 2016. page 44

N. Simon, J. Friedman, T. J. Hastie, and R. Tibshirani. A sparse-group lasso. *J. Comput. Graph. Statist.*, 22(2):231–245, 2013. ISSN 1061-8600. pages 24, 66

E. Soubies, L. Blanc-Féraud, and G. Aubert. A continuous exact ℓ_0 penalty (CEL0) for least squares regularized problem. *SIAM J. Imaging Sci.*, 8(3):1607–1639, 2015. page 24

N. Städler, P. Bühlmann, and S. van de Geer. ℓ_1-penalization for mixture regression models. *TEST*, 19(2):209–256, 2010. page 86

S. Stich, A. Raj, and M. Jaggi. Safe adaptive importance sampling. In *NeurIPS*, pages 4384–4394, 2017. page 56

D. Strohmeier. *Spatio-Temporal Sparse Priors for MEG/EEG Source Reconstruction*. PhD thesis, Technische Univesität Ilmenau, 2016. page 36

D. Strohmeier, Y. Bekhti, J. Haueisen, and A. Gramfort. The iterative reweighted mixed-norm estimate for spatio-temporal MEG/EEG source reconstruction. *IEEE Trans. Med. Imag.*, 2016. page 36

B. Stucky. *Asymptotic confidence regions and sharp oracle results under structured sparsity*. PhD thesis, ETH Zurich, 2017. page 96

T. Sun and C.-H. Zhang. Scaled sparse linear regression. *Biometrika*, 99(4):879–898, 2012. pages 25, 86, 88

Y. Sun, H. Jeong, J. Nutini, and M. Schmidt. Are we there yet? Manifold identification of gradient-related proximal methods. In *AISTATS*, pages 1110–1119, 2019. page 69

H. L. Taylor, S. C. Banks, and J. F. McCoy. Deconvolution with the ℓ_1 norm. *Geophysics*, 44(1):39–52, 1979. page 23

G. Thompson, F. Tonge, and S. Zionts. Techniques for removing nonbinding constraints and extraneous variables from linear programming problems. *Management Science*, 12(7):588–608, 1966. page 44

R. Tibshirani. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 58(1):267–288, 1996. pages 24, 44, 86

R. Tibshirani, J. Bien, J. Friedman, T. J. Hastie, N. Simon, J. Taylor, and R. J. Tibshirani. Strong rules for discarding predictors in lasso-type problems. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 74(2):245–266, 2012. pages 44, 66, 94

R. J. Tibshirani. The lasso problem and uniqueness. *Electron. J. Stat.*, 7:1456–1490, 2013. pages 69, 70

R. J. Tibshirani. Dykstra's algorithm, ADMM, and coordinate descent: Connections, insights, and extensions. In *NeurIPS*, pages 517–528, 2017. pages 50, 53

A. N. Tikhonov. On the stability of inverse problems. *Dokl. Akad. Nauk SSSR*, 39: 176–179, 1943. page 22

J. A. Tropp. Just relax: convex programming methods for identifying sparse signals in noise. *IEEE Trans. Inf. Theory*, 52(3):1030–1051, 2006. page 23

P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *J. Optim. Theory Appl.*, 109(3):475–494, 2001. pages 44, 45, 66, 91

P. Tseng and S. Yun. Block-coordinate gradient descent method for linearly constrained nonsmooth separable optimization. *J. Optim. Theory Appl.*, 140(3):513, 2009. pages 24, 91, 92

K. Uutela, M. Hämäläinen, and R. Salmelin. Global optimization in the localization of neuromagnetic sources. *IEEE Trans. Med. Imag.*, 45(6):716–723, 1998. page 35

K. Uutela, M. Hämäläinen, and E. Somersalo. Visualization of magnetoencephalographic data using minimum current estimates. *NeuroImage*, 10(2):173–180, 1999. page 36

D. Vainsencher, H. Liu, and T. Zhang. Local smoothness in variance reduced optimization. In *NeurIPS*, pages 2179–2187, 2015. page 44

S. Vaiter, G. Peyré, and J. M. Fadili. Model consistency of partly smooth regularizers. *IEEE Trans. Inf. Theory*, 64(3):1725–1737, 2018. page 74

S. van de Geer. *Estimation and testing under sparsity*, volume 2159 of *Lecture Notes in Mathematics*. Springer, 2016. Lecture notes from the 45th Probability Summer School held in Saint-Four, 2015, École d'Été de Probabilités de Saint-Flour. pages 96, 101

S. van de Geer and B. Stucky. χ^2-confidence sets in high-dimensional regression. In *Statistical analysis for high-dimensional data*, pages 279–306. Springer, 2016. page 97

B. D. Van Veen, W. Van Drongelen, M. Yuchtman, and A. Suzuki. Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *IEEE Transactions on biomedical engineering*, 44(9):867–880, 1997. page 35

J. Wagener and H. Dette. Bridge estimators and the adaptive Lasso under heteroscedasticity. *Math. Methods Statist.*, 21:109–126, 2012. page 86

J. Wang, J. Zhou, P. Wonka, and J. Ye. Lasso screening rules via dual polytope projection. In *NeurIPS*, pages 1070–1078, 2013. pages 44, 66

D. Wipf and S. Nagarajan. A unified Bayesian framework for MEG/EEG source imaging. *NeuroImage*, 44(3):947–966, 2009. page 36

D. P. Wipf, J. P. Owen, H. Attias, K. Sekihara, and S. S. Nagarajan. Estimating the location and orientation of complex, correlated neural activity using MEG. In *NeurIPS*, pages 1777–1784, 2008. pages 36, 130

D. Wrinch and H. Jeffreys. On certain fundamental principles of scientific inquiry. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 42 (249):369–390, 1921. page 22

T. T. Wu and K. Lange. Coordinate descent algorithms for lasso penalized regression. *Ann. Appl. Stat.*, pages 224–244, 2008. page 24

Z. J. Xiang and P. J. Ramadge. Fast lasso screening tests based on correlations. In *ICASSP*, pages 2137–2140, 2012. page 44

Z. J. Xiang, Y. Wang, and P. J. Ramadge. Screening tests for lasso problems. *IEEE Trans. Pattern Anal. Mach. Intell.*, PP(99), 2016. page 66

G. Yuan, C.-H. Ho, and C.-J. Lin. An improved GLMNET for l1-regularized logistic regression. *J. Mach. Learn. Res.*, 13:1999–2030, 2012. page 78

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(1):49–67, 2006. pages 24, 36, 66

C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *Ann. Statist.*, 38(2):894–942, 2010. page 24

T. Zhang. Adaptive forward-backward greedy algorithm for learning sparse representations. *IEEE Trans. Inf. Theory*, 57(7):4689–4708, 2011. page 23

P. Zhao and B. Yu. On model selection consistency of Lasso. *J. Mach. Learn. Res.*, 7: 2541–2563, 2006. page 24

M. Zibulevsky and B. A. Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Neural computation*, 13(4):863–882, 2001. page 26

H. Zou. The adaptive lasso and its oracle properties. *J. Amer. Statist. Assoc.*, 101(476): 1418–1429, 2006. page 25

H. Zou and T. J. Hastie. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 67(2):301–320, 2005. page 24

université PARIS-SACLAY
ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Régression parcimonieuse en grande dimension en présence de bruit coloré hétéroscédastique: application à la localisation de sources M/EEG

Mots clés : Optimisation convexe, parcimonie, hétéroscédasticité, lissage, extrapolation

Résumé : Parmi les techniques d'imagerie cérébrale, la magnéto- et l'électro-encéphalographie se distinguent pour leur faible degré d'invasivité et leur excellente résolution temporelle. La reconstruction de l'activité neuronale à partir de l'enregistrements des champs électriques et magnétiques constitue un problème inverse extrêmement mal posé, auquel il est nécessaire d'ajouter des contraintes pour le résoudre. Une approche populaire, empruntée dans ce manuscript, est de postuler que la solution est parcimonieuse spatialement, ce qui peut s'obtenir par une pénalisation L2/1. Cependant, ce type de régularisation nécessite de résoudre des problèmes d'optimisation non-lisses en grande dimension, avec des méthodes itératives dont la performance se dégrade avec la dimension. De plus, les enregistrements M/EEG sont typiquement corrompus par un fort bruit coloré, allant à l'encontre des hypothèses classiques de résolution des problèmes inverses.
Dans cette thèse, nous proposons d'abord une accélération des algorithmes itératifs utilisés pour résoudre le problème bio-magnetique avec régularisation L2/1. Les améliorations classiques (règles de filtrage et ensemble actifs), tirent parti de la parcimonie de la solution: elles ignorent les sources cérébrales inactives, et réduisent ainsi la dimension du problème. Nous introduisons une nouvelle technique d'ensemble actifs, reposant sur les règles de filtrage les plus performantes actuellement. Nous proposons des techniques duales avancées, qui permettent un contrôle plus fin de l'optimalité et améliorent les techniques d'identification de prédicteurs. Notre construction duale extrapole la structure Vectorielle Autoregressive des itérés duaux, régularité que nous relions aux propriétés d'identification de support des algorithmes proximaux. En plus du problème inverse bio-magnétique, l'approche proposée est appliquée à l'ensemble des modèles linéaires généralisés régularisés L1.
Deuxièmement, nous introduisons de nouveaux estimateurs concomitants pour la régression multitâche, conçus pour traiter du bruit gaussien corrélé. Le problème d'optimisation sous-jacent est convexe, et présente une structure "lisse + proximable" attrayante ; nous lions la formulation de ce problème au lissage des normes de Schatten.

Title : Sparse high dimensional regression in the presence of colored heteroscedastic noise: application to M/EEG source imaging

Keywords : Convex optimization, sparsity, heteroscedasticity, smoothing, extrapolation; magneto-electroencephalography

Abstract : Amongst neuroimaging techniques, magneto- and electroencephalography (M/EEG) stand out for their non-invasiveness and their excellent time resolution. Reconstructing the neural activity from the recordings of magnetic field and electric potentials is a severely ill-posed inverse problem, for which it is popular to assume spatial *sparsity* of the solution, obtained through $\ell_{2,1}$-penalized regression approaches. However, this regularization requires to solve time-consuming high-dimensional optimization problems. Additionally, M/EEG recordings are usually corrupted by strong non-white noise, which breaks the classical statistical assumptions of inverse problems.
In this thesis, we first propose speed improvements of iterative solvers used for the $\ell_{2,1}$-regularized bio-magnetic inverse problem. Typical improvements, screening and working sets, exploit the sparsity of the solution: by identifying inactive brain sources, they reduce the dimensionality of the optimization problem. We introduce a new working set policy, derived from the state-of-the-art Gap safe screening rules, and propose duality improvements, yielding a tighter control of optimality and improving feature identification techniques. Our dual construction extrapolates on an asymptotic Vector AutoRegressive regularity of the dual iterates, which we connect to manifold identification of proximal algorithms. Beyond the $\ell_{2,1}$-regularized bio-magnetic inverse problem, the proposed methods apply to the whole class of sparse Generalized Linear Models.
Second, we introduce new concomitant estimators for multitask regression. We design them to handle non-white Gaussian noise, and to exploit the multiple repetitions nature of M/EEG experiments. The underlying optimization problem is jointly convex in the regression coefficients and the noise variable, with a "smooth + proximable" composite structure. It is therefore solvable via standard alternate minimization, for which we apply the improvements detailed in the first part. We provide a theoretical analysis of our objective function, linking it to the smoothing of Schatten norms.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France