Structured sparsity through convex optimization

Francis Bach
INRIA - SIERRA Project-Team
23, avenue d'Italie
75013 Paris, France
`francis.bach@inria.fr`

Rodolphe Jenatton
INRIA - SIERRA Project-Team
23, avenue d'Italie
75013 Paris, France
`rodolphe.jenatton@inria.fr`

Julien Mairal
Department of Statistics
University of California
Berkeley, CA 94720, USA
`julien@stat.berkeley.edu`

Guillaume Obozinski
INRIA - SIERRA Project-Team
23, avenue d'Italie
75013 Paris, France
`guillaume.obozinski@inria.fr`

September 12, 2011

Abstract

Sparse estimation methods are aimed at using or obtaining parsimonious representations of data or models. While naturally cast as a combinatorial optimization problem, variable or feature selection admits a convex relaxation through the regularization by the ℓ_1-norm. In this paper, we consider situations where we are not only interested in sparsity, but where some structural prior knowledge is available as well. We show that the ℓ_1-norm can then be extended to structured norms built on either disjoint or overlapping groups of variables, leading to a flexible framework that can deal with various structures. We present applications to supervised learning in the context of non-linear variable selection, and to unsupervised learning, for structured sparse principal component analysis, and hierarchical dictionary learning.

1 Introduction

The concept of parsimony is central in many scientific domains. In the context of statistics, signal processing or machine learning, it takes the form of variable or feature selection problems, and is commonly used in two situations: First, to make the model or the prediction more interpretable or cheaper to use, i.e., even if the underlying problem does not admit sparse solutions, one looks for the best sparse approximation. Second, sparsity can also be used given prior knowledge that the model should be sparse.

Sparse linear models seek to predict an output by linearly combining a small subset of the features describing the data. To simultaneously address variable selection and model estimation, ℓ_1-norm regularization has become a popular tool, which benefits both from efficient algorithms [see, e.g., Efron et al., 2004, Lee et al., 2007, Beck and Teboulle, 2009,

Yuan, 2010, Bach et al., 2011, and multiple references therein] and a well-developed theory for generalization properties and variable selection consistency [Zhao and Yu, 2006, Wainwright, 2009, Bickel et al., 2009, Zhang, 2009].

When regularizing with the ℓ_1-norm, each variable is selected individually, regardless of its position in the input feature vector, so that existing relationships and structures between the variables (e.g., spatial, hierarchical or related to the physics of the problem at hand) are merely disregarded. However, in many practical situations the estimation can benefit from some type of prior knowledge, potentially both for interpretability and to improve predictive performance.

This a priori can take various forms: in neuroimaging based on functional magnetic resonance (fMRI) or magnetoencephalography (MEG), sets of voxels allowing to discriminate between different brain states are expected to form small localized and connected areas [Gramfort and Kowalski, 2009, Xiang et al., 2009, and references therein]. Similarly, in face recognition, as shown in Section 4.4, robustness to occlusions can be increased by considering as features, sets of pixels that form small convex regions of the faces. Again, a plain ℓ_1-norm regularization fails to encode such specific spatial constraints [Jenatton et al., 2010c]. The same rationale supports the use of *structured sparsity* for background subtraction [Cevher et al., 2008, Huang et al., 2009a, Mairal et al., 2010b].

Another example of the need for higher-order prior knowledge comes from bioinformatics. Indeed, for the diagnosis of tumors, the profiles of array-based comparative genomic hybridization (arrayCGH) can be used as inputs to feed a classifier [Rapaport et al., 2008]. These profiles are characterized by many variables, but only a few observations of such profiles are available, prompting the need for variable selection. Because of the specific spatial organization of bacterial artificial chromosomes along the genome, the set of discriminative features is expected to consist of specific contiguous patterns. Using this prior knowledge in addition to standard sparsity leads to improvement in classification accuracy [Rapaport et al., 2008]. In the context of multi-task regression, a problem of interest in genetics is to find a mapping between a small subset of loci presenting single nucleotide polymorphisms (SNP's) that have a phenotypic impact on a given family of genes [Kim and Xing, 2010]. This target family of genes has its own structure, where some genes share common genetic characteristics, so that these genes can be embedded into some underlying hierarchy. Exploiting directly this hierarchical information in the regularization term outperforms the unstructured approach with a standard ℓ_1-norm [Kim and Xing, 2010].

These real world examples motivate the need for the design of sparsity-inducing regularization schemes, capable of encoding more sophisticated prior knowledge about the expected sparsity patterns. As mentioned above, the ℓ_1-norm corresponds only to a constraint on *cardinality* and is oblivious of any other information available about the patterns of nonzero coefficients ("nonzero patterns" or "supports") induced in the solution, since they are all theoretically possible. In this paper, we consider a family of sparsity-inducing norms that can address a large variety of structured sparse problems: a simple change of norm will induce new ways of selecting variables; moreover, as shown in Section 3.5 and Section 3.6, algorithms to obtain estimators (e.g., convex optimization methods) and theoretical analyses are easily extended in many situations. As shown in Section 3, the norms we introduce

2

generalize traditional "group ℓ_1-norms", that have been popular for selecting variables organized in non-overlapping groups [Turlach et al., 2005, Yuan and Lin, 2006, Roth and Fischer, 2008, Huang and Zhang, 2010].

The paper is organized as follows: we first review in Section 2 classical ℓ_1-norm regularization in supervised contexts. We then introduce several families of norms in Section 3, and present applications to unsupervised learning in Section 4, namely for sparse principal component analysis in Section 4.4 and hierarchical dictionary learning in Section 4.5. We briefly show in Section 5 how these norms can also be used for high-dimensional non-linear variable selection.

Notations. Throughout the paper, we shall denote vectors with bold lower case letters, and matrices with bold upper case ones. For any integer j in the set $[\![1;p]\!] \triangleq \{1,\ldots,p\}$, we denote the j-th coefficient of a p-dimensional vector $\mathbf{w} \in \mathbb{R}^p$ by $\mathbf{w}_j$. Similarly, for any matrix $\mathbf{W} \in \mathbb{R}^{n \times p}$, we refer to the entry on the i-th row and j-th column as $\mathbf{W}_{ij}$, for any $(i,j) \in [\![1;n]\!] \times [\![1;p]\!]$. We will need to refer to sub-vectors of $\mathbf{w} \in \mathbb{R}^p$, and so, for any $J \subseteq [\![1;p]\!]$, we denote by $\mathbf{w}_J \in \mathbb{R}^{|J|}$ the vector consisting of the entries of $\mathbf{w}$ indexed by J. Likewise, for any $I \subseteq [\![1;n]\!]$, $J \subseteq [\![1;p]\!]$, we denote by $\mathbf{W}_{IJ} \in \mathbb{R}^{|I| \times |J|}$ the sub-matrix of $\mathbf{W}$ formed by the rows (respectively the columns) indexed by I (respectively by J). We extensively manipulate norms in this paper. We thus define the ℓ_q-norm for any vector $\mathbf{w} \in \mathbb{R}^p$ by $\|\mathbf{w}\|_q^q \triangleq \sum_{j=1}^{p} |\mathbf{w}_j|^q$ for $q \in [1,\infty)$, and $\|\mathbf{w}\|_\infty \triangleq \max_{j \in [\![1;p]\!]} |\mathbf{w}_j|$. For $q \in (0,1)$, we extend the definition above to ℓ_q pseudo-norms. Finally, for any matrix $\mathbf{W} \in \mathbb{R}^{n \times p}$, we define the Frobenius norm of $\mathbf{W}$ by $\|\mathbf{W}\|_F^2 \triangleq \sum_{i=1}^{n} \sum_{j=1}^{p} \mathbf{W}_{ij}^2$.

2   Unstructured sparsity via the ℓ_1-norm

Regularizing by the ℓ_1-norm has been a topic of intensive research over the last decade. This line of work has witnessed the development of nice theoretical frameworks [Tibshirani, 1996, Chen et al., 1998, Mallat, 1999, Tropp, 2004, 2006, Zhao and Yu, 2006, Zou, 2006, Wainwright, 2009, Bickel et al., 2009, Zhang, 2009, Negahban et al., 2009] and the emergence of many efficient algorithms [Efron et al., 2004, Nesterov, 2007, Friedman et al., 2007, Wu and Lange, 2008, Beck and Teboulle, 2009, Wright et al., 2009, Needell and Tropp, 2009, Yuan et al., 2010]. Moreover, this methodology has found quite a few applications, notably in compressed sensing [Candes and Tao, 2005], for the estimation of the structure of graphical models [Meinshausen and Bühlmann, 2006] or for several reconstruction tasks involving natural images [e.g., see Mairal, 2010, for a review]. In this section, we focus on supervised learning and present the traditional estimation problems associated with sparsity-inducing norms such as the ℓ_1-norm (see Section 4 for unsupervised learning).

In supervised learning, we predict (typically one-dimensional) outputs y in $\mathcal{Y}$ from observations $\mathbf{x}$ in $\mathcal{X}$; these observations are usually represented by p-dimensional vectors with $\mathcal{X} = \mathbb{R}^p$. M-estimation and in particular regularized empirical risk minimization are well suited to this setting. Indeed, given n pairs of data points $\{(\mathbf{x}^{(i)}, y^{(i)}) \in \mathbb{R}^p \times \mathcal{Y}; \ i=1,\ldots,n\}$,

we will consider the estimators solving the following form of convex optimization problem

$$\min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \ell(y^{(i)}, \mathbf{w}^\top \mathbf{x}^{(i)}) + \lambda \Omega(\mathbf{w}), \tag{2.1}$$

where ℓ is a loss function and $\Omega : \mathbb{R}^p \to \mathbb{R}$ is a sparsity-inducing—typically nonsmooth and non-Euclidean—norm. Typical examples of differentiable loss functions are the square loss for least squares regression, i.e., $\ell(y, \hat{y}) = \frac{1}{2}(y - \hat{y})^2$ with y in $\mathbb{R}$, and the logistic loss $\ell(y, \hat{y}) = \log(1 + e^{-y\hat{y}})$ for logistic regression, with y in $\{-1, 1\}$. We refer the readers to Shawe-Taylor and Cristianini [2004] and to Hastie et al. [2001] for more complete descriptions of loss functions.

Within the context of least-squares regression, ℓ_1-norm regularization is known as the Lasso [Tibshirani, 1996] in statistics and as basis pursuit in signal processing [Chen et al., 1998]. For the Lasso, formulation (2.1) takes the form

$$\min_{\mathbf{w} \in \mathbb{R}^p} \left[\frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_1 \right], \tag{2.2}$$

and, equivalently, basis pursuit can be written[1]

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^p} \left[\frac{1}{2} \|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2^2 + \lambda \|\boldsymbol{\alpha}\|_1 \right]. \tag{2.3}$$

These two equations are obviously identical but we write them both to show the correspondence between notations used in statistics and in signal processing. In statistical notations, we will use $\mathbf{X} \in \mathbb{R}^{n \times p}$ to denote a set of n observations described by p variables (covariates), while $\mathbf{y} \in \mathbb{R}^n$ represents the corresponding set of n targets (responses) that we try to predict. For instance, $\mathbf{y}$ may have discrete entries in the context of classification. With notations of signal processing, we will consider an m-dimensional signal $\mathbf{x} \in \mathbb{R}^m$ that we express as a linear combination of p dictionary elements composing the dictionary $\mathbf{D} \triangleq [\mathbf{d}^1, \ldots, \mathbf{d}^p] \in \mathbb{R}^{m \times p}$. While the design matrix $\mathbf{X}$ is usually assumed fixed and given beforehand, we shall see in Section 4 that the dictionary $\mathbf{D}$ may correspond either to some pre-defined basis [e.g., see Mallat, 1999, for wavelet bases] or to a representation that is actually *learned* as well [Olshausen and Field, 1996].

Geometrical intuitions for the ℓ_1-norm ball. While we consider in (2.1) a regularized formulation, we could have considered an equivalent *constrained* problem of the form

$$\min_{\mathbf{w} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \ell(y^{(i)}, \mathbf{w}^\top \mathbf{x}^{(i)}) \quad \text{such that} \quad \Omega(\mathbf{w}) \leq \mu, \tag{2.4}$$

for some $\mu \in \mathbb{R}_+$: It is indeed the case that the solutions to problem (2.4) obtained when varying μ is the same as the solutions to problem (2.1), for some of λ_μ depending on μ [e.g., see Section 3.2 in Borwein and Lewis, 2006].

---

[1]Note that the formulations which are typically encountered in signal processing are either $\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1$ s.t. $\mathbf{x} = \mathbf{D}\boldsymbol{\alpha}$, which corresponds to the limiting case of Eq. (2.3) where $\lambda \to 0$ and $\mathbf{x}$ is in the span of the dictionary $\mathbf{D}$, or $\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_1$ s.t. $\|\mathbf{x} - \mathbf{D}\boldsymbol{\alpha}\|_2 \leq \eta$ which is a constrained counterpart of Eq. (2.3) leading to the same set of solutions (see the explanation following Eq. (2.4)).

At optimality, the opposite of the gradient of f evaluated at any solution $\hat{\mathbf{w}}$ of (2.4) must belong to the normal cone to $\mathcal{B} = \{\mathbf{w} \in \mathbb{R}^p; \ \Omega(\mathbf{w}) \leq \mu\}$ at $\hat{\mathbf{w}}$ [Borwein and Lewis, 2006]. In other words, for sufficiently small values of μ (i.e., ensuring that the constraint is active) the level set of f for the value $f(\hat{\mathbf{w}})$ is tangent to $\mathcal{B}$. As a consequence, important properties of the solutions $\hat{\mathbf{w}}$ follow from the geometry of the ball $\mathcal{B}$. If Ω is taken to be the ℓ_2-norm, then the resulting ball $\mathcal{B}$ is the standard, isotropic, "round" ball that does not favor any specific direction of the space. On the other hand, when Ω is the ℓ_1-norm, $\mathcal{B}$ corresponds to a diamond-shaped pattern in two dimensions, and to a double pyramid in three dimensions. In particular, $\mathcal{B}$ is anisotropic and exhibits some singular points due to the non-smoothness of Ω. Since these singular points are located along axis-aligned linear subspaces in $\mathbb{R}^p$, if the level set of f with the smallest feasible value is tangent to $\mathcal{B}$ at one of those points, sparse solutions are obtained. We display on Figure 1 the balls $\mathcal{B}$ for both the ℓ_1- and ℓ_2-norms. See Section 3 and Figure 2 for extensions to structured norms.

(a) ℓ_2-norm ball   (b) ℓ_1-norm ball

Figure 1: Comparison between the ℓ_2-norm and ℓ_1-norm balls in three dimensions, respectively on the left and right figures. The ℓ_1-norm ball presents some singular points located along the axes of $\mathbb{R}^3$ and along the three axis-aligned planes going through the origin.

3 Structured Sparsity-Inducing Norms

In this section, we consider structured sparsity-inducing norms that induce estimated vectors that are not only sparse, as for the ℓ_1-norm, but whose support also displays some structure known a priori that reflects potential relationships between the variables.

3.1 Sparsity-Inducing Norms with Disjoint Groups of Variables

The most natural form of structured sparsity is arguably *group sparsity*, matching the a priori knowledge that pre-specified disjoint blocks of variables should be selected or ignored simultaneously. In that case, if $\mathcal{G}$ is a collection of groups of variables, forming a partition
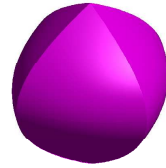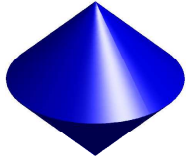
(a) ℓ_1/ℓ_2-norm ball without overlaps: $\Omega(\mathbf{w}) = \|\mathbf{w}_{\{1,2\}}\|_2 + |\mathbf{w}_3|$

(b) ℓ_1/ℓ_2-norm ball with overlaps: $\Omega(\mathbf{w}) = \|\mathbf{w}_{\{1,2,3\}}\|_2 + |\mathbf{w}_1| + |\mathbf{w}_2|$

Figure 2: Comparison between two mixed ℓ_1/ℓ_2-norm balls in three dimensions, without and with overlapping groups of variables, respectively on the left and right figures. The singular points appearing on these balls describe the sparsity-inducing behavior of the underlying norms Ω.

of $[\![1; p]\!]$, we define Ω as

$$\Omega(\mathbf{w}) = \sum_{g \in \mathcal{G}} \|\mathbf{w}_g\|_q \quad \text{for any } q \in (1, \infty]. \tag{3.1}$$

This norm is usually referred to as a mixed ℓ_1/ℓ_q-norm, and in practice, popular choices for q are $\{2, \infty\}$. As desired, regularizing with Ω leads variables in the same group to be selected or set to zero simultaneously (see Figure 2 for a geometric interpretation). In the context of least-squares regression, this regularization is known as the group Lasso [Turlach et al., 2005, Yuan and Lin, 2006]. It has be shown to improve the prediction performance and/or interpretability of the learned models when the block structure is relevant [Roth and Fischer, 2008, Stojnic et al., 2009, Lounici et al., 2009, Huang and Zhang, 2010]. Moreover, applications of this regularization scheme arise also in the context of multi-task learning [Obozinski et al., 2009, Quattoni et al., 2009, Liu et al., 2009] to account for features shared across tasks, and multiple kernel learning [Bach, 2008a] for the selection of different kernels (see also Section 5).

It might seem that the case of groups that overlap would be unnecessarily complex. It turns out, in reality, that appropriate collections of overlapping groups allow to encode quite interesting forms of structured sparsity. In fact, the idea of constructing sparsity-inducing norms from overlapping groups will be key. We present two different constructions based on overlapping groups of variables that are essentially complementary of each other in sections 3.2 and 3.3.

3.2 Sparsity-Inducing Norms with Overlapping Groups of Variables

In this section, we consider a direct extension of the norm introduced in the previous section to the case of overlapping groups; we give an informal overview of the structures that it can encode and examples of relevant applied settings. For more details see Jenatton et al. [2009].

Starting from the definition of Ω in Eq. (3.1), it is natural to study what happens when the set of groups $\mathcal{G}$ is allowed to contain elements that overlap. In fact, and as shown by Jenatton et al. [2009], the sparsity-inducing behavior of Ω remains the same: when regularizing by Ω, some entire groups of variables g in $\mathcal{G}$ are set to zero. This is reflected in the set of non-smooth extreme points of the unit ball of the norm represented on Figure 2-(b). While the resulting patterns of nonzero variables—also referred to as *supports*, or *nonzero patterns*—were obvious in the non-overlapping case, it is interesting to understand here the relationship that ties together the set of groups $\mathcal{G}$ and its associated set of possible nonzero patterns. Let us denote by $\mathcal{P}$ the latter set. For any norm of the form (3.1), it is still the case that variables belonging to a given group are encouraged to be set simultaneously to zero; as a result, the possible zero patterns for solutions of (2.1) are obtained by forming unions of the basic groups, which means that the possible supports are obtained by taking the intersection of a certain number of complements of the basic groups.

Moreover, under mild conditions [Jenatton et al., 2009], given any *intersection-closed*[2] family of patterns $\mathcal{P}$ of variables (see examples below), it is possible to build an ad-hoc set of groups $\mathcal{G}$—and hence, a regularization norm Ω—that enforces the support of the solutions of (2.1) to belong to $\mathcal{P}$.

These properties make it possible to *design* norms that are adapted to the structure of the problem at hand, which we now illustrate with a few examples.

One-dimensional interval pattern. Given p variables organized in a sequence, using the set of groups $\mathcal{G}$ of Figure 3, it is only possible to select *contiguous nonzero patterns*. In this case, we have $|\mathcal{G}| = O(p)$. Imposing the contiguity of the nonzero patterns can be relevant in the context of variable forming time series, or for the diagnosis of tumors, based on the profiles of CGH arrays [Rapaport et al., 2008], since an bacterial artificial chromosome will be inserted as a single continuous block into the genome.

Two-dimensional convex support. Similarly, assume now that the p variables are organized on a two-dimensional grid. To constrain the allowed supports $\mathcal{P}$ to be the set of all rectangles on this grid, a possible set of groups $\mathcal{G}$ to consider is represented in the top of Figure 4. This set is relatively small since $|\mathcal{G}| = O(\sqrt{p})$. Groups corresponding to half planes with additional orientations (see Figure 4 bottom) may be added to "carve out" more general convex patterns. See an illustration in Section 4.4.

---

[2] A set $\mathcal{A}$ is said to be intersection-closed, if for any $k \in \mathbb{N}$, and for any $(a_1, \ldots, a_k) \in \mathcal{A}^k$, we have $\bigcap_{i=1}^{k} a_i \in \mathcal{A}$.
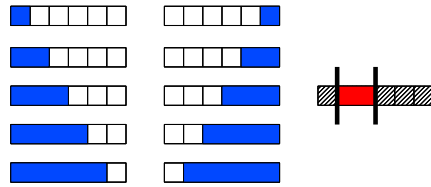
Figure 3: (Left) The set of blue groups to penalize in order to select contiguous patterns in a sequence. (Right) In red, an example of such a nonzero pattern with its corresponding zero pattern (hatched area).
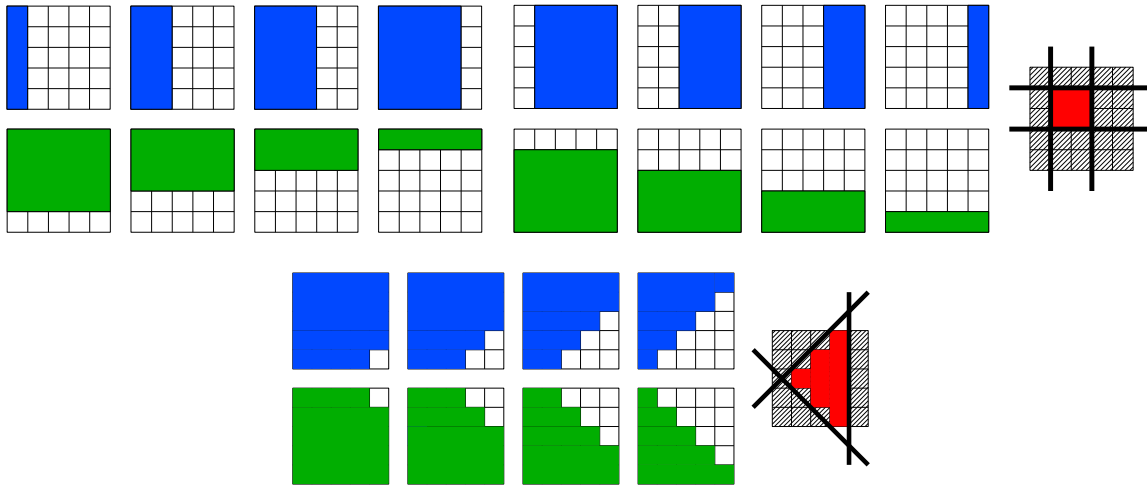

Figure 4: Top: Vertical and horizontal groups: (Left) the set of blue and green groups to penalize in order to select rectangles. (Right) In red, an example of nonzero pattern recovered in this setting, with its corresponding zero pattern (hatched area). Bottom: Groups with $\pm\pi/4$ orientations: (Left) the set of blue and green groups with their (not displayed) complements to penalize in order to select diamond-shaped patterns.

Two-dimensional block structures on a grid. Using sparsity-inducing regularizations built upon groups which are composed of variables together with their spatial neighbors leads to good performances for background subtraction [Cevher et al., 2008, Baraniuk et al., 2010, Huang et al., 2009a, Mairal et al., 2010b], topographic dictionary learning [Kavukcuoglu et al., 2009, Mairal et al., 2011], wavelet-based denoising [Rao et al., 2011a].

Hierarchical structure. A fourth interesting example assumes that the variables are organized in a hierarchy. Precisely, we assume that the p variables can be assigned to the nodes of a tree $\mathcal{T}$ (or a forest of trees), and that a given variable may be selected only if all its ancestors in $\mathcal{T}$ have already been selected. This hierarchical rule is exactly respected when using the family of groups displayed on Figure 5. The corresponding penalty was first used in Zhao et al. [2009]; one of it simplest instance in the context of regression is the sparse group Lasso [Sprechmann et al., 2010, Friedman et al., 2010]; it has found numerous

8

applications, for instance, wavelet-based denoising [Zhao et al., 2009, Baraniuk et al., 2010, Huang et al., 2009a, Jenatton et al., 2010b], hierarchical dictionary learning for both topic modelling and image restoration [Jenatton et al., 2010a,b], log-linear models for the selection of potential orders [Schmidt and Murphy, 2010], bioinformatics, to exploit the tree structure of gene networks for multi-task regression [Kim and Xing, 2010], and multi-scale mining of fMRI data for the prediction of simple cognitive tasks [Jenatton et al., 2011].
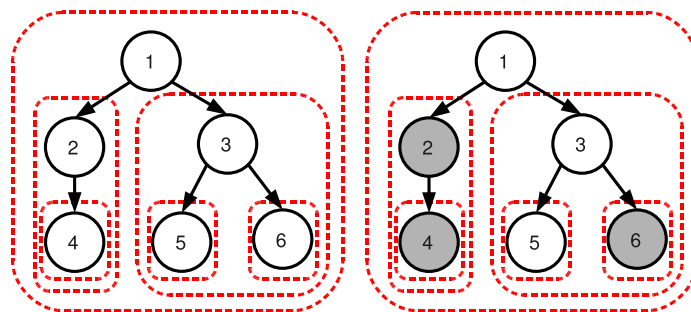

Figure 5: Left: set of groups $\mathcal{G}$ (dashed contours in red) corresponding to the tree $\mathcal{T}$ with $p = 6$ nodes represented by black circles. Right: example of a sparsity pattern induced by the tree-structured norm corresponding to $\mathcal{G}$: the groups $\{2, 4\}, \{4\}$ and $\{6\}$ are set to zero, so that the corresponding nodes (in gray) that form subtrees of $\mathcal{T}$ are removed. The remaining nonzero variables $\{1, 3, 5\}$ form a rooted and connected subtree of $\mathcal{T}$. This sparsity pattern obeys the two following equivalent rules: (i) if a node is selected, the same goes for all its ancestors. (ii) if a node is not selected, then its descendant are not selected.

Extensions. Possible choices for the sets of groups $\mathcal{G}$ are not limited to the aforementioned examples: more complicated topologies can be considered, for example three-dimensional spaces discretized in cubes or spherical volumes discretized in slices (see an application to neuroimaging by Varoquaux et al. [2010]), and more complicated hierarchical structures based on directed acyclic graphs can be encoded as further developed in Section 5.

3.3   Norms for overlapping groups: a latent variable formulation.

The family of norms defined in Eq. (3.1) is adapted to *intersection-closed* sets of nonzero patterns. However, some applications exhibit structures that can be more naturally modelled by *union-closed* families of supports. This idea was developed by Jacob et al. [2009] who, given a set of groups $\mathcal{G}$, introduced the following norm

$$\Omega_{\text{union}}(\mathbf{w}) \triangleq \min_{\mathbf{v} \in \mathbb{R}^{p \times |\mathcal{G}|}} \sum_{g \in \mathcal{G}} d_g \|\mathbf{v}^g\|_q \quad \text{such that} \quad \begin{cases} \sum_{g \in \mathcal{G}} \mathbf{v}^g = \mathbf{w}, \\ \forall g \in \mathcal{G}, \ \mathbf{v}_j^g = 0 \text{ if } j \notin g. \end{cases} \tag{3.2}$$

The norm defined above provides a different generalization of the ℓ_1/ℓ_q norm to the case of overlapping groups than the norm presented in Section 3.2. In fact, it is easy to see that

solving Eq. (2.1) with the norm Ω_{union} is equivalent to solving

$$\min_{(\mathbf{v}^g \in \mathbb{R}^{|g|})_{g \in \mathcal{G}}} \sum_{i=1}^{n} \ell\big(y^{(i)}, \sum_{g \in \mathcal{G}} \mathbf{v}_g^{g\top} \mathbf{x}_g^{(i)}\big) + \lambda \sum_{g \in \mathcal{G}} d_g \|\mathbf{v}^g\|_q \qquad (3.3)$$

and setting $\mathbf{w} = \sum_{g \in \mathcal{G}} \mathbf{v}^g$. This last equation shows that using the norm Ω_{union} can be interpreted as implicitly duplicating the variables belonging to several groups and regularizing with a weighted ℓ_1/ℓ_q norm for disjoint groups in the expanded space. It should be noted that a careful choice of the weights is much more important in the situation of overlapping groups than in the case of disjoint groups.

This latent variable formulation pushes some of the vectors $\mathbf{v}^g$ to zero while keeping others with no zero components, hence leading to a vector $\mathbf{w}$ with a support which is in general the union of the selected groups. Interestingly, it can be seen as a convex relaxation of a non-convex penalty encouraging similar sparsity patterns which was introduced by Huang et al. [2009b] and which we present in section 3.4.

Graph Lasso. One type of a priori knowledge commonly encountered takes the form of graph defined on the set of input variables, which is such that connected variables are more likely to be simultaneously relevant or irrelevant; this type of prior is common in genomics where regulation, co-expression or interaction networks between genes (or their expression level) used as predictors are often available. To favor the selection of neighbors of a selected variable, it is possible to consider the edges of the graph as groups in the previous formulation (see Jacob et al. [2009], Rao et al. [2011b]).

Patterns consisting of a small number of intervals. A quite similar situation occurs, when one knows a priori — typically for variables forming sequences (times series, strings, polymers)— that the support should consist of a small number of connected subsequences. In that case, one can consider the sets of variables forming connected subsequences (or connected subsequences of length at most k) as the overlapping groups.

3.4 Related Approaches to Structured Sparsity

Norm design through submodular functions. Another approach to structured sparsity relies on submodular analysis [Bach, 2010a]. Starting from a non-decreasing, submodular[3] set-function F of the supports of the parameter vector $\mathbf{w}$ — i.e., $\mathbf{w} \mapsto F(\{j \in [\![1;p]\!];\ \mathbf{w}_j \neq 0\})$ — a structured sparsity-inducing norm can be built by considering its convex envelope (tightest convex lower bound) on the unit ℓ_∞-norm ball. By selecting the appropriate set-function F, similar structures to those described above can be obtained. This idea can be further extended to symmetric, submodular set-functions of the level sets of $\mathbf{w}$, that is, $\mathbf{w} \mapsto \max_{\nu \in \mathbb{R}} F(\{j \in [\![1;p]\!];\ \mathbf{w}_j \geq \nu\})$, thus leading to different types of structures [Bach, 2010b], allowing to shape the level sets of $\mathbf{w}$ rather than its support.

---

[3]Let S be a finite set. A function $F : 2^S \to \mathbb{R}$ is said to be submodular if for any subset $A, B \subseteq S$, we have the inequality $F(A \cap B) + F(A \cup B) \leq F(A) + F(B)$; see Bach [2010a] and references therein.
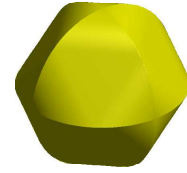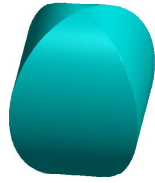
(a) Unit ball for $\mathcal{G} = \big\{\{1,3\}, \{2,3\}\big\}$

(b) $\mathcal{G} = \big\{\{1,3\}, \{2,3\}, \{1,2\}\big\}$

Figure 6: Two instances of unit balls of the latent group Lasso regularization Ω_{union} for two or three groups of two variables. Their singular points lie on axis aligned circles, corresponding to each group, and whose convex hull is exactly the unit ball. It should be noted that the ball on the left is quite similar to the one of Fig. 2b except that its "poles" are flatter, hence discouraging the selection of $\mathbf{x}_3$ without either $\mathbf{x}_1$ or $\mathbf{x}_2$.

Non-convex approaches. We end this review of structured sparsity approaches by a couple of non-convex formulations. In the same spirit as the norm (3.2), Huang et al. [2009a] considered the penalty

$$\psi(\mathbf{w}) \triangleq \min_{\mathcal{H} \subseteq \mathcal{G}} \sum_{g \in \mathcal{H}} \omega_g, \text{ such that } \{j \in [\![1;p]\!]; \ \mathbf{w}_j \neq 0\} \subseteq \bigcup_{g \in \mathcal{H}} g,$$

where $\mathcal{G}$ is a given set of groups, and $\{\omega_g\}_{g \in \mathcal{G}}$ is a set of positive weights which defines a *coding length*. In other words, the penalty ψ measures from an information-theoretic viewpoint, "how much it costs" to represent $\mathbf{w}$. Finally, in the context of compressed sensing, the work of Baraniuk et al. [2010] also focuses on union-closed families of supports, although without information-theoretic considerations.

3.5 Convex optimization with proximal methods

In this section, we briefly review *proximal methods* which are convex optimization methods particularly suited to the norms we have defined. They essentially allow to solve the problem regularized with a new norm at low implementation and computational costs. For a more complete presentation of optimization techniques adapted to sparsity-inducing norms, see Bach et al. [2011].

Proximal methods constitute a class of first-order techniques typically designed to solve problem (2.1) [Nesterov, 2007, Beck and Teboulle, 2009, Combettes and Pesquet, 2010]. They take advantage of the structure of (2.1) as the sum of two convex terms, only one of

11

which is assumed smooth. Thus, we will typically assume that the loss function ℓ is convex differentiable, with Lipschitz-continuous gradients (such as the logistic or square loss), while Ω will only be assumed convex.

Proximal methods have become increasingly popular over the past few years, both in the signal processing [e.g., Becker et al., 2009, Wright et al., 2009, Combettes and Pesquet, 2010, and numerous references therein] and in the machine learning communities [e.g., Bach et al., 2011, and references therein]. In a broad sense, these methods can be described as providing a natural extension of gradient-based techniques when the objective function to minimize has a non-smooth part. Proximal methods are iterative procedures. Their basic principle is to linearize, at each iteration, the function f around the current estimate $\hat{\mathbf{w}}$, and to update this estimate as the (unique, by strong convexity) solution of the following *proximal problem*:

$$\min_{\mathbf{w} \in \mathbb{R}^p} \left[f(\hat{\mathbf{w}}) + (\mathbf{w} - \hat{\mathbf{w}})^{\top} \nabla f(\hat{\mathbf{w}}) + \lambda \Omega(\mathbf{w}) + \frac{L}{2} \|\mathbf{w} - \hat{\mathbf{w}}\|_2^2 \right]. \tag{3.4}$$

The role of the added quadratic term is to keep the update in a neighborhood of $\hat{\mathbf{w}}$ where f stays close to its current linear approximation; $L > 0$ is a parameter which is an upper bound on the Lipschitz constant of ∇f.

Provided that we can solve efficiently the proximal problem (3.4), this first iterative scheme constitutes a simple way of solving problem (2.1). It appears under various names in the literature: proximal-gradient techniques [Nesterov, 2007], forward-backward splitting methods [Combettes and Pesquet, 2010], and iterative shrinkage-thresholding algorithm [Beck and Teboulle, 2009]. Furthermore, it is possible to guarantee convergence rates for the function values [Nesterov, 2007, Beck and Teboulle, 2009], and after k iterations, the precision be shown to be of order $O(1/k)$, which should contrasted with rates for the subgradient case, that are rather $O(1/\sqrt{k})$.

This first iterative scheme can actually be extended to "accelerated" versions [Nesterov, 2007, Beck and Teboulle, 2009]. In that case, the update is not taken to be exactly the result from (3.4); instead, it is obtained as the solution of the proximal problem applied to a well-chosen linear combination of the previous estimates. In that case, the function values converge to the optimum with a rate of $O(1/k^2)$, where k is the iteration number. From Nesterov [2004], we know that this rate is optimal within the class of first-order techniques; in other words, accelerated proximal-gradient methods can be as fast as without non-smooth component.

We have so far given an overview of proximal methods, without specifying how we precisely handle its core part, namely the computation of the proximal problem, as defined in (3.4).

Proximal Problem.   We first rewrite problem (3.4) as

$$\min_{\mathbf{w} \in \mathbb{R}^p} \; \frac{1}{2} \left\| \mathbf{w} - \left(\hat{\mathbf{w}} - \frac{1}{L} \nabla f(\hat{\mathbf{w}}) \right) \right\|_2^2 + \frac{\lambda}{L} \Omega(\mathbf{w}).$$

Under this form, we can readily observe that when $\lambda = 0$, the solution of the proximal problem is identical to the standard gradient update rule. The problem above can be more

generally viewed as an instance of the *proximal operator* [Moreau, 1962] associated with $\lambda\Omega$:

$$\text{Prox}_{\lambda\Omega} : \mathbf{u} \in \mathbb{R}^p \mapsto \underset{\mathbf{v} \in \mathbb{R}^p}{\arg\min} \frac{1}{2}\|\mathbf{u} - \mathbf{v}\|_2^2 + \lambda\Omega(\mathbf{v}).$$

For many choices of regularizers Ω, the proximal problem has a closed-form solution, which makes proximal methods particularly efficient. If Ω is chosen to be the ℓ_1-norm, the proximal operator is simply the soft-thresholding operator applied elementwise [Donoho and Johnstone, 1995]. It turns out that for the norms defined in this paper, we can compute in a large number of cases the proximal operator exactly and efficiently [see more details in Bach et al., 2011]. Most of these norms and the associated proximal problems are implemented in the open-source software SPAMS[4].

In summary, with proximal methods, generalizing algorithms from the ℓ_1-norm to a structured norm requires only to be able to compute the corresponding proximal operator, which can be done efficiently in many cases.

3.6 Theoretical analysis

Sparse methods are traditionally analyzed according to three different criteria; it is often assumed that the data were generated by a sparse loading vector $\mathbf{w}^*$. Denoting $\hat{\mathbf{w}}$ a solution of the M-estimation problem in Eq. (2.1), traditional statistical consistency results aim at showing that $\|\mathbf{w}^* - \hat{\mathbf{w}}\|$ is small for a certain norm $\|\cdot\|$; model consistency considers the estimation of the support of $\mathbf{w}^*$ as a criterion, while, prediction efficiency only cares about the prediction of the model, i.e., with the square loss, the quantity $\|\mathbf{X}\mathbf{w}^* - \mathbf{X}\hat{\mathbf{w}}\|_2^2$ has to be as small as possible.

A striking consequence of assuming that $\mathbf{w}^*$ has many zero components is that for the three criteria, consistency is achievable even when p is much larger than n [Zhao and Yu, 2006, Wainwright, 2009, Bickel et al., 2009, Zhang, 2009].

However, to relax the often unrealistic assumption that the data are generated by a sparse loading vector, and also because a good predictor, especially in the high-dimensional setting, can possibly be much sparser than any potential true model generating the data, prediction efficiency is often formulated under the form of *oracle inequalities*, where the performance of the estimator is upper bounded by the performance of any function in a fixed complexity class, reflecting approximation error, plus a complexity term characterizing the class and reflecting the hardness of estimation in that class. We refer the reader to van de Geer [2010] for a review and references on oracle results for the Lasso and the group Lasso.

In the structured sparse situations, such high-dimensional phenomenon can also be characterized; essentially, if we assume that $\mathbf{w}^*$ is compatible with the additional prior knowledge that we impose on its support through our norm Ω, then the assumptions required for consistency can usually be relaxed. See Huang and Zhang [2009], Jenatton et al. [2009], Huang et al. [2009a], Jacob et al. [2009], Bach [2010a], for more details in several settings.

---

[4]http://www.di.ens.fr/willow/SPAMS/

4 Sparse principal component analysis and dictionary learning

Unsupervised learning aims at extracting latent representations of the data that are useful for analysis, visualization, denoising or to extract relevant information to solve subsequently a supervised learning problem. Sparsity or structured sparsity are essential to specify, on the representations, constraints that improve their identifiability and interpretability.

4.1 Analysis and synthesis views of PCA

Depending on how the latent representation is extracted or constructed from the data, it is useful to distinguish two points of view. This is illustrated well in the case of PCA.

In the *analysis* view, PCA aims at finding *sequentially* a set of directions in space that explain the largest fraction of the variance of the data. This can be formulated as an iterative procedure in which a one-dimensional projection of the data with maximal variance is found first, then the data are projected on the orthogonal subspace (corresponding to a *deflation* of the covariance matrix), and the process is iterated. In the *synthesis* view, PCA aims at finding a set of vectors, or *dictionary elements* (in a terminology closer to signal processing) such that all observed signals admit a linear decomposition on that set with low reconstruction error. In the case of PCA, these two formulations lead to the same solution (an eigenvalue problem). However, in extensions of PCA, in which either the dictionary elements or the decompositions of signals are constrained to be sparse or structured, they lead to different algorithms with different solutions.

The *analysis* interpretation leads to sequential formulations [d'Aspremont et al., 2008, Moghaddam et al., 2006, Jolliffe et al., 2003] that consider components one at a time and perform a *deflation* of the covariance matrix at each step [see Mackey, 2009]. The *synthesis* interpretation leads to non-convex global formulations [Zou et al., 2006, Mairal et al., 2010a, Moghaddam et al., 2006, Lee et al., 2007] which estimate simultaneously all principal components, typically do not require the orthogonality of the components, and are referred to as matrix factorization problems [Singh and Gordon, 2008, Bach et al., 2008] in machine learning, and dictionary learning in signal processing [Olshausen and Field, 1996].

While we could also impose structured sparse priors in the analysis view, we will consider from now on the synthesis view, that we will introduce with the terminology of dictionary learning.

4.2 Dictionary learning

Given a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ of n columns corresponding to n observations in $\mathbb{R}^m$, the dictionary learning problem is to find a matrix $\mathbf{D} \in \mathbb{R}^{m \times p}$, called *dictionary*, such that each observation can be well approximated by a linear combination of the p columns $(\mathbf{d}^k)_{k \in [\![1;p]\!]}$ of $\mathbf{D}$ called the *dictionary elements*. If $\mathbf{A} \in \mathbb{R}^{p \times n}$ is the matrix of the linear combination

14

coefficients or *decomposition coefficients* (or *codes*), with $\mathbf{a}^k$ the k-th column of $\mathbf{A}$ being the coefficients for the k-th signal $\mathbf{x}^k$, the matrix product $\mathbf{DA}$ is called a decomposition of $\mathbf{X}$.

Learning simultaneously the dictionary $\mathbf{D}$ and the coefficients $\mathbf{A}$ corresponds to a matrix factorization problem [see Witten et al., 2009, and reference therein].

As formulated by Bach et al. [2008] or Witten et al. [2009], it is natural, when learning a decomposition, to penalize or constrain some norms or pseudo-norms of $\mathbf{A}$ and $\mathbf{D}$, say $\Omega_{\mathbf{A}}$ and $\Omega_{\mathbf{D}}$ respectively, to encode prior information — typically sparsity — about the decomposition of $\mathbf{X}$. While in general the penalties could be defined globally on the matrices $\mathbf{A}$ and $\mathbf{D}$, we assume that each column of $\mathbf{D}$ and $\mathbf{A}$ is penalized separately. This can be written as

$$\min_{\substack{\mathbf{A}\in\mathbb{R}^{p\times n},\\ \mathbf{D}\in\mathbb{R}^{m\times p}}} \frac{1}{2nm}\big\|\mathbf{X}-\mathbf{DA}\big\|_{\mathrm{F}}^2 + \lambda\sum_{k=1}^{p}\Omega_{\mathbf{D}}(\mathbf{d}^k),\ \text{s.t. } \Omega_{\mathbf{A}}(\mathbf{a}^i) \leq 1,\ \forall\, i\in [\![1;n]\!], \qquad (4.1)$$

where the regularization parameter $\lambda \geq 0$ controls to which extent the dictionary is regularized. If we assume that both regularizations $\Omega_{\mathbf{A}}$ and $\Omega_{\mathbf{D}}$ are convex, problem (4.1) is convex with respect to $\mathbf{A}$ for fixed $\mathbf{D}$ and vice versa. It is however not *jointly* convex in the pair $(\mathbf{A},\mathbf{D})$, but alternating optimization schemes generally lead to good performance.

4.3 Imposing sparsity

The choice of the two norms $\Omega_{\mathbf{A}}$ and $\Omega_{\mathbf{D}}$ is crucial and heavily influences the behavior of dictionary learning. When no norms are used (e.g., $\lambda = 0$), then we classically recover PCA. When $\Omega_{\mathbf{A}}$ is the ℓ_1-norm and $\Omega_{\mathbf{D}}$ the ℓ_2-norm, we aim at finding a dictionary such that each signal $\mathbf{x}^i$ admits a sparse decomposition on the dictionary. In this context, we are essentially looking for a basis where the data have sparse decompositions, a framework we refer to as *sparse dictionary learning*. On the contrary, when $\Omega_{\mathbf{A}}$ is the ℓ_2-norm and $\Omega_{\mathbf{D}}$ the ℓ_1-norm, the formulation induces sparse principal components, i.e., atoms with many zeros, a framework we refer to as sparse PCA. In Section 4.4 and Section 4.5, we replace the ℓ_1-norm by structured norms introduced in Section 3, leading to structured versions of the above estimation problems.

4.4 Adding structures to principal components

One of PCA's main shortcomings is that, even if it finds a small number of important factors, the factor themselves typically involve all original variables. In the last decade, several alternatives to PCA which find sparse and potentially interpretable factors have been proposed, notably non-negative matrix factorization (NMF) [Lee and Seung, 1999] and sparse PCA (SPCA) [Jolliffe et al., 2003, Zou et al., 2006, Zass and Shashua, 2007, Witten et al., 2009].

However, in many applications, only constraining the size of the supports of the factors does not seem appropriate because the considered factors are not only expected to be sparse but

15

also to have a certain structure. In fact, the popularity of NMF for face image analysis owes essentially to the fact that the method happens to retrieve sets of variables that are partly localized on the face and capture some features or parts of the face which seem intuitively meaningful given our a priori. We might therefore gain in the quality of the factors induced by enforcing directly this a priori in the matrix factorization constraints. More generally, it would be desirable to encode higher-order information about the supports that reflects the *structure* of the data. For example, in computer vision, features associated to the pixels of an image are naturally organized on a grid and the supports of factors explaining the variability of images could be expected to be localized, connected or have some other regularity with respect to that grid. Similarly, in genomics, factors explaining the gene expression patterns observed on a microarray could be expected to involve groups of genes corresponding to biological pathways or set of genes that are neighbors in a protein-protein interaction network.

Based on these remarks and with the norms presented earlier, sparse PCA is readily extended to *structured sparse PCA* (SSPCA), which explains the variance of the data by factors that are not only sparse but also respect some a priori structural constraints deemed relevant to model the data at hand: slight variants of the regularization term defined in Section 3 (with the groups defined in Figure 4) can be used successfully for $\Omega_\mathbf{D}$.

Experiments on face recognition. By definition, dictionary learning belongs to unsupervised learning; in that sense, our method may appear first as a tool for exploratory data analysis, which leads us naturally to *qualitatively* analyze the results of our decompositions (e.g., by visualizing the learned dictionaries). This is obviously a difficult and subjective exercise, beyond the assessment of the consistency of the method in artificial examples where the "true" dictionary is known. For quantitative results, see Jenatton et al. [2009].[5]

We apply SSPCA on the cropped AR Face Database [Martinez and Kak, 2001] that consists of 2600 face images, corresponding to 100 individuals (50 women and 50 men). For each subject, there are 14 non-occluded poses and 12 occluded ones (the occlusions are due to sunglasses and scarfs). We reduce the resolution of the images from 165×120 pixels to 38×27 pixels for computational reasons.

Figure 7 shows examples of learned dictionaries (for $p = 36$ elements), for NMF, unstructured sparse PCA (SPCA), and SSPCA. While NMF finds sparse but spatially unconstrained patterns, SSPCA selects sparse convex areas that correspond to a more natural segment of faces. For instance, meaningful parts such as the mouth and the eyes are recovered by the dictionary.

---

[5]A Matlab toolbox implementing our method can be downloaded from http://www.di.ens.fr/~jenatton/.
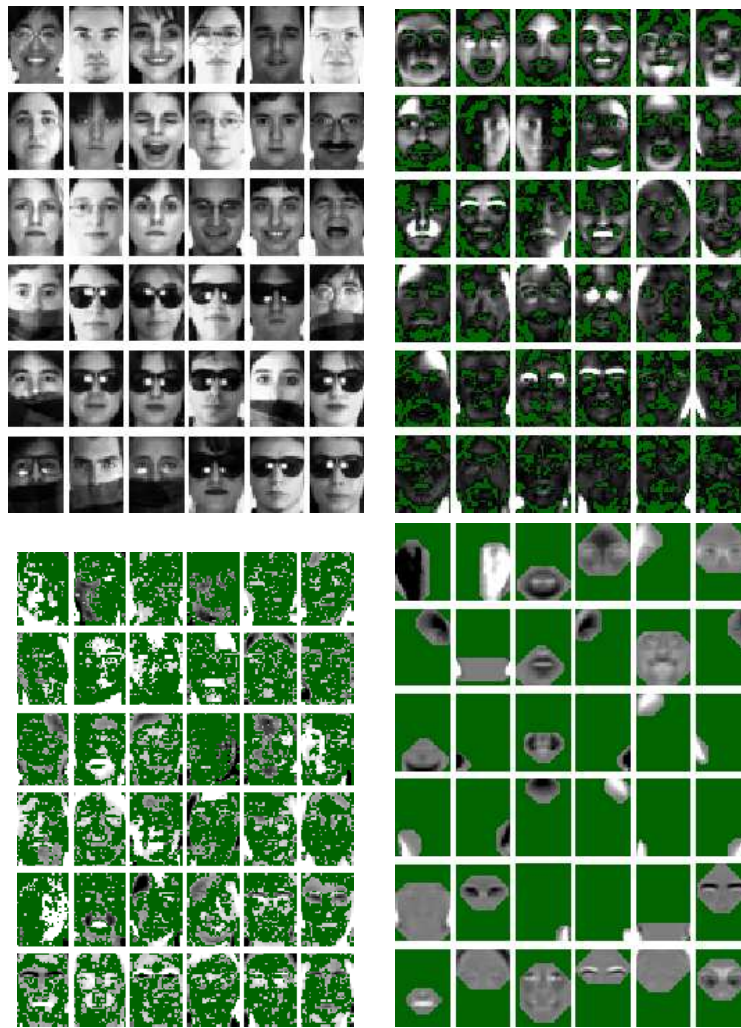
16

Figure 7: Top left, examples of faces in the datasets. The three remaining images represent learned dictionaries of faces with $p=36$: NMF (top right), SPCA (bottom left) and SSPCA (bottom right). The dictionary elements are sorted in decreasing order of explained variance. While NMF gives sparse spatially unconstrained patterns, SSPCA finds convex areas that correspond to more natural face segments. SSPCA captures the left/right illuminations and retrieves pairs of symmetric patterns. Some displayed patterns do not seem to be convex, e.g., nonzero patterns located at two opposite corners of the grid. However, a closer look at these dictionary elements shows that convex shapes are indeed selected, and that small numerical values (just as regularizing by ℓ_2-norm may lead to) give the visual impression of having zeroes in convex nonzero patterns. This also shows that if a nonconvex pattern has to be selected, it will be, by considering its convex hull.

4.5   Hierarchical dictionary learning

In this section, we consider sparse dictionary learning, where the structured sparse prior knowledge is put on the decomposition coefficients, i.e., the matrix $\mathbf{A}$ in Eq. (4.1), and present an application to text documents.

Text documents.   The goal of probabilistic topic models is to find a low-dimensional representation of a collection of documents, where the representation should provide a semantic description of the collection. Approaching the problem in a parametric Bayesian framework, latent Dirichlet allocation (LDA), Blei et al. [2003] models documents, represented as vectors of word counts, as a mixture of a predefined number of *latent topics*, defined as multinomial distributions over a fixed vocabulary. The number of topics is usually small compared to the size of the vocabulary (e.g., 100 against 10 000), so that the topic proportions of each document provide a compact representation of the corpus.

In fact the problem addressed by LDA is fundamentally a matrix factorization problem. For instance, Buntine [2002] argued that LDA can be interpreted as a Dirichlet-multinomial counterpart of factor analysis. We can actually cast the problem in the dictionary learning formulation that we presented before[6]. Indeed, suppose that the signals $\mathbf{X} = [\mathbf{x}^1, \ldots, \mathbf{x}^n]$ in $\mathbb{R}^{m \times n}$ are each the so-called *bag-of-word* representation of each of n documents over a vocabulary of m words, i.e., $\mathbf{x}^i$ is a vector whose k-th component is the empirical frequency in document i of the k-th word of a fixed lexicon. If we constrain the entries of $\mathbf{D}$ and $\mathbf{A}$ to be nonnegative, and the dictionary elements $\mathbf{d}^j$ to have unit ℓ_1-norm, the decomposition $(\mathbf{D}, \mathbf{A})$ can be interpreted as the parameters of a topic-mixture model. Sparsity here ensures that a document is described by a small number of topics.

Switching to structured sparsity allows in this case to organize automatically the dictionary of topics in the process of learning it. Assume that $\Omega_{\mathbf{A}}$ in Eq. (4.1) is a tree-structured regularization, such as illustrated on Figure 5; in this case, in the light of Section 3.2, if the decomposition of a document involves a certain topic, then all ancestral topics in the tree are also present in the topic decomposition. Since the hierarchy is shared by all documents, the topics close to the root participate in every decomposition, and given that the dictionary is learned, this mechanism forces those topics to be quite generic—essentially gathering the lexicon which is common to all documents. Conversely, the deeper the topics in the tree, the more specific they should be. It should be noted that such hierarchical dictionaries can also be obtained with generative probabilistic models, typically based on non-parametric Bayesian prior over trees or paths in trees, and which extend the LDA model to topic hierarchies [Blei et al., 2010, Adams et al., 2010].

Visualization of NIPS proceedings.   We qualitatively illustrate our approach on the NIPS proceedings from 1988 through 1999 [Griffiths and Steyvers, 2004]. After removing words appearing fewer than 10 times, the dataset is composed of 1714 articles, with a vocabulary of 8274 words. As explained above, we enforce both the dictionary and the sparse coefficients to be non-negative, and constrain the dictionary elements to have a

---

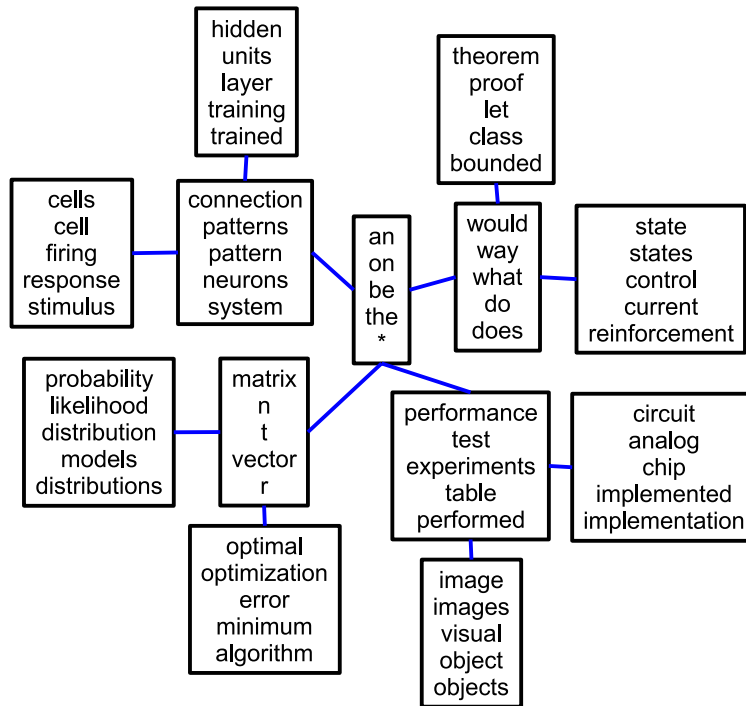[6]Doing so we simply trade the multinomial likelihood with a least-square formulation.

Figure 8: Example of a topic hierarchy estimated from 1714 NIPS proceedings papers (from 1988 through 1999). Each node corresponds to a topic whose 5 most important words are displayed. Single characters such as n, t, r are part of the vocabulary and often appear in NIPS papers, and their place in the hierarchy is semantically relevant to children topics.

unit ℓ_1-norm. Figure 8 displays an example of a learned dictionary with 13 topics, obtained by using a tree-structured penalty (see Section 3.2) on the coefficients $\mathbf{A}$ and by selecting manually $\lambda = 2^{-15}$. As expected and similarly to Blei et al. [2010], we capture the stopwords at the root of the tree, and topics reflecting the different subdomains of the conference such as neurosciences, optimization or learning theory.

5   High-dimensional non-linear variable selection

In this section, we show how structured sparsity-inducing norms may be used to provide an efficient solution to the problem of high-dimensional *non-linear* variable selection. Namely, given p variables $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$, our aim is to find a non-linear function $\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_p)$ which depends only on a few variables. First approaches to the problem have considered restricted functional forms such as $\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_p) = \mathbf{f}_1(\mathbf{x}_1) + \cdots + \mathbf{f}_p(\mathbf{x}_p)$, where each $\mathbf{f}_1, \ldots, \mathbf{f}_p$ are univariate non-linear functions [Ravikumar et al., 2008, Bach, 2008b]. However, many non-linear functions cannot be expressed as sums of functions of these forms. Additional interactions have been added leading to functions of the form $\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_p) = \sum_{J \subset \{1, \ldots, p\}, \ |J| \leqslant 2} \mathbf{f}_J(\mathbf{x}_J)$ [Lin and Zhang, 2006]. While second-order interactions make the class of functions larger, our aim in this section is to consider functions which

19

can be expressed a sparse linear combination of the form $\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_q) = \sum_{J \subset \{1, \ldots, p\}} \mathbf{f}_J(\mathbf{x}_J)$, i.e., a combination of functions defined on potentially larger subsets of variables.

The main difficulties associated with this problem are that (1) each function $\mathbf{f}_J$ has to be estimated, leading to a non-parametric problem, and (2) there are exponentially many such functions. We propose however an approach that overcomes both difficulties in the next section, based on the ideas that estimating functions rather than vectors can be tackled with estimators in reproducing kernel Hilbert spaces (see Section 5.1), and that the complexity issues can be addressed by imposing some structure among all the subsets $J \subset \{1, \ldots, p\}$ (see Section 5).

5.1 Multiple kernel learning: From linear to non-linear predictions

Reproducing kernel Hilbert spaces are arguably the simplest spaces for the non-parametric estimation of non-linear functions since most learning algorithms for linear models are directly ported to any RKHS via simple kernelization. We therefore start be reviewing learning from a single and later multiple reproducing kernels, since our approach will be based on combining functions from multiple (actually a hierarchy) of RKHSes. For more details, see Bach [2008a].

Single kernel learning. Let us assume that the n input data-points $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}$ belong to a set $\mathcal{X}$ (not necessarily $\mathbb{R}^p$), and consider predictors of the form $\langle f, \Phi(\mathbf{x}) \rangle$ where $\Phi : \mathcal{X} \rightarrow \mathcal{F}$ is a map from the input space to a reproducing kernel Hilbert space $\mathcal{F}$ (associated to the kernel function k), which we refer to as the feature space. These predictors are linearly parameterized, but may depend non-linearly on $\mathbf{x}$. We consider the following estimation problem:

$$\min_{\mathbf{f} \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \ell(y^{(i)}, \langle \mathbf{f}, \Phi(\mathbf{x}^{(i)}) \rangle) + \frac{\lambda}{2} \|\mathbf{f}\|_{\mathcal{F}}^2,$$

where $\|.\|_{\mathcal{F}}$ is the Hilbertian norm associated to $\mathcal{F}$. The representer theorem [Kimeldorf and Wahba, 1971] states that, for all loss functions (potentially nonconvex), the solution $\mathbf{f}$ admits the expansion $\mathbf{f} = \sum_{i=1}^{n} \boldsymbol{\alpha}_i \Phi(\mathbf{x}^{(i)})$, so that, replacing $\mathbf{f}$ by its new expression, we can now minimize

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \frac{1}{n} \sum_{i=1}^{n} \ell(y^{(i)}, (\mathbf{K}\boldsymbol{\alpha})_i) + \frac{\lambda}{2} \boldsymbol{\alpha}^\top \mathbf{K} \boldsymbol{\alpha},$$

where $\mathbf{K}$ is the *kernel matrix*, an $n \times n$ matrix whose element (i, j) is equal to $\langle \Phi(\mathbf{x}^{(i)}), \Phi(\mathbf{x}^{(j)}) \rangle = k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$. This optimization problem involves the observations $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}$ only through the kernel matrix $\mathbf{K}$, and can thus be solved, as long as $\mathbf{K}$ can be evaluated efficiently. See Shawe-Taylor and Cristianini [2004] for more details.

Multiple kernel learning (MKL). We can now assume that we are given m Hilbert spaces $\mathcal{F}_j$, and look for predictors of the form $\mathbf{f}(\mathbf{x}) = \mathbf{g}_1(\mathbf{x}) + \cdots + \mathbf{g}_m(\mathbf{x})$, where[7] each

---

[7] Notice that the function $\mathbf{g}_j$ is not restricted to depend only on $\mathbf{x}_j$ as before.

$\mathbf{g}_j \in \mathcal{F}_j$. In order to have many $\mathbf{g}_j$ equal to zero, we can penalize $\mathbf{f}$ using a sum of norms similar to the group Lasso penalties introduced earlier, namely $\sum_{j=1}^{m} \|\mathbf{g}_j\|_{\mathcal{F}_j}$. This leads to selection of functions. Moreover, it turns out that the optimization problems may be expressed also in terms of the m kernel matrices, and it is equivalent to learn a sparse linear combination $\hat{\mathbf{K}} = \sum_{j=1}^{m} \eta_j \mathbf{K}_j$ (with many η's equal to zero) of kernel matrices with then $\boldsymbol{\alpha}$ solution of the single kernel learning problem for $\hat{\mathbf{K}}$. For more details, see Bach [2008a].

From MKL to sparse generalized additive models. As shown above, the MKL framework is defined with any set of m RKHSes defined on the same base set $\mathcal{X}$. When the base set is itself defined as a cartesian product of p base sets, i.e., $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p$, then it is common to consider $m = p$ RKHSes which are each of them defined on a single $\mathcal{X}_i$, leading to the desired functional form $\mathbf{f}_1(\mathbf{x}_1) + \cdots + \mathbf{f}_p(\mathbf{x}_p)$. To overcome the limitation of this functional form we need to consider a more complex expansion.

5.2 Hierarchical kernel learning

In this section, we consider functional expansions with up to $m = 2^p$ terms corresponding to different RKHSes, each defined on a cartesian product of a subset of the p separate input spaces. Specifically, we consider functions of the form $\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_p) = \sum_{J \subset \{1,\ldots,p\}} \mathbf{f}_J(\mathbf{x}_J)$ with $\mathbf{f}_J$ chosen to live in a RKHS $\mathcal{F}_J$ defined on variables $\mathbf{x}_J$. Penalizing by the norm $\sum_{J \subset \{1,\ldots,p\}} \|\mathbf{f}_J\|_{\mathcal{F}_J}$ would in theory lead to an appropriate selection of functions from the various RKHSes (and to learning a sparse linear combination of the corresponding kernel matrices). However, in practice, there are 2^p such predictors, which is not algorithmically feasible.

This is where structured sparsity comes into play. In order to obtain polynomial-time algorithms and theoretically controlled predictive performance, we may add an extra constraint to the problem. Namely, we endow the power set of $\{1, \ldots, p\}$ with the partial order of the inclusion of sets, and in this directed acyclic graph (DAG), we require that predictors $\mathbf{f}$ select a subset only after all of its ancestors have been selected. This can be achieved in a convex formulation using a structured-sparsity inducing norm of the type presented in section 3.2, but defined by a hierarchy of groups as follows

$$\Omega\big[(\mathbf{f}_H)_{H \subset \{1,\ldots,p\}}\big] = \sum_{J \subset \{1,\ldots,p\}} \left(\sum_{H \supset J} \|\mathbf{f}_H\|_2^2 \right)^{1/2}.$$

As illustrated in Figure 9, this norm corresponds to overlapping groups of variables defined on the directed acyclic graphs of all subsets of $\{1, \ldots, p\}$. We will explain briefly how introducing this norm may lead to polynomial time algorithms and what theoretical guarantees are associated with it.

Polynomial-time estimation algorithm. While we are, a priori, still facing an estimation problem with 2^p functions, it can be solved using an active set method, which considers adding a component $\mathbf{f}_J \in \mathcal{F}_J$ (resp. $\mathbf{K}_J$) to the active set of predictors (resp. kernels). The
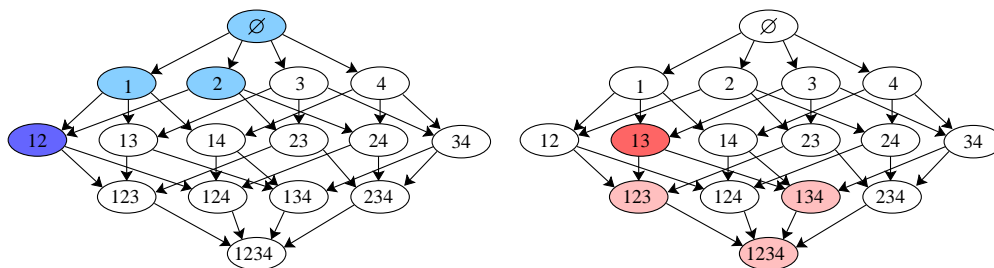
Figure 9: Power set of the set $\{1,\ldots,4\}$: in blue, an authorized set of selected subsets. In red, an example of a group used within the norm (a subset and all of its descendants in the DAG).

two crucial aspects are (1) to add the right kernel, i.e., choose among the 2^p which one to add, and (2) when to stop. As shown in Bach [2008c], these steps may be carried out efficiently for certain collections of RKHSes $\mathcal{F}_J$, in particular those for which we are able to compute efficiently (i.e., in polynomial time in p) the sum $\sum_{J\subset\{1,\ldots,p\}}\mathbf{K}_J$. This is the case, for example, for Gaussian kernels $k_J(\mathbf{x}_J,\mathbf{x}'_J)=\exp(-\gamma\|\mathbf{x}_J-\mathbf{x}'_J\|_2)$.

Theoretical analysis. Bach [2008c] showed that under appropriate assumptions, estimation under high-dimensional scaling, i.e., for $p\gg n$ but $\log p=O(n)$, is possible in this situation, in spite of the fact that the number of terms in the expansion is now potentially doubly exponential in n.

6 Conclusion

In this paper, we reviewed several approaches for structured sparsity, based on convex optimization and the design of appropriate sparsity-inducing norms. Analyses and algorithms for the traditional ℓ_1-norm can readily be extended to these new norms, making them an efficient and flexible tools for introducing prior knowledge in high-dimensional statistical problems. We also presented several applications to supervised and unsupervised learning problems, where the proper use of additional knowledge leads to improved interpretability of the sparse estimates and/or increased predictive performance.

Acknowledgements

References

Ryan Adams, Zoubin Ghahramani, and Michael Jordan. Tree-structured stick breaking for hierarchical data. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 19–27. 2010.

F. Bach. Consistency of the group lasso and multiple kernel learning. *Journal of Machine Learning Research*, 9:1179–1225, 2008a.

F. Bach. Consistency of the group Lasso and multiple kernel learning. *Journal of Machine Learning Research*, pages 1179–1225, 2008b.

F. Bach. Exploring large feature spaces with hierarchical multiple kernel learning. In *Neural Information Processing Systems*, volume 21, 2008c.

F. Bach. Structured sparsity-inducing norms through submodular functions. In *Advances in Neural Information Processing Systems*, 2010a.

F. Bach. Shaping level sets with submodular functions. Technical report, Preprint arXiv:1012.1501, 2010b.

F. Bach, J. Mairal, and J. Ponce. Convex sparse matrix factorizations. Technical report, Preprint arXiv:0812.1869, 2008.

F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. Technical Report 00613125, HAL, 2011.

R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56:1982–2001, 2010.

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

S. Becker, J. Bobin, and E. Candes. NESTA: A Fast and Accurate First-order Method for Sparse Recovery. Technical report, Preprint arXiv:0904.3367, 2009.

P. Bickel, Y. Ritov, and A. Tsybakov. Simultaneous analysis of Lasso and Dantzig selector. *Annals of Statistics*, 37(4):1705–1732, 2009.

D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

D. Blei, T. L. Griffiths, and M. I. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2):1–30, 2010.

J. M. Borwein and A. S. Lewis. *Convex Analysis and Nonlinear Optimization: Theory and Examples.* Springer, 2006.

W. L. Buntine. Variational Extensions to EM and Multinomial PCA. In *Proceedings of the European Conference on Machine Learning (ECML)*, 2002.

E.J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.

V. Cevher, M. F. Duarte, C. Hegde, and R. G. Baraniuk. Sparse signal recovery using markov random fields. In *Advances in Neural Information Processing Systems*, 2008.

S. S. Chen, D. L. Donoho, and M. A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

P. L. Combettes and J.-C. Pesquet. Proximal splitting methods in signal processing. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*. Springer, 2010.

A. d'Aspremont, F. Bach, and L. El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9:1269–1294, 2008.

D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, 1995.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–451, 2004.

J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1(2):302–332, 2007.

J. Friedman, T. Hastie, and R. Tibshirani. A note on the group lasso and a sparse group lasso. *preprint*, 2010.

A. Gramfort and M. Kowalski. Improving M/EEG source localization with an inter-condition sparse prior. In *IEEE International Symposium on Biomedical Imaging*, 2009.

T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl 1):5228, 2004.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, 2001.

J. Huang and T. Zhang. The benefit of group sparsity. Technical report, 2009. Preprint arXiv:0901.2962.

J. Huang and T. Zhang. The benefit of group sparsity. *Annals of Statistics*, 38(4):1978–2004, 2010.

J. Huang, T. Zhang, and D. Metaxas. Learning with structured sparsity. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009a.

J. Huang, Z. Zhang, and D. Metaxas. Learning with structured sparsity. In *Proc. Intl. Conf. Machine Learning*, 2009b.

L. Jacob, G. Obozinski, and J.-P. Vert. Group Lasso with overlaps and graph Lasso. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

R. Jenatton, J.-Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. Technical report, Preprint arXiv:0904.3523, 2009.

R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010a.

R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for hierarchical sparse coding. Technical report, Preprint arXiv:1009.2139v2, 2010b. To appear in Journal Machine Learning Research.

R. Jenatton, G. Obozinski, and F. Bach. Structured sparse principal component analysis. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010c.

R. Jenatton, A. Gramfort, V. Michel, G. Obozinski, F. Bach, and B. Thirion. Multi-scale mining of fMRI data with hierarchical structured sparsity. In *International Workshop on Pattern Recognition in Neuroimaging (PRNI)*, 2011.

I. T. Jolliffe, N. T. Trendafilov, and M. Uddin. A modified principal component technique based on the Lasso. *Journal of Computational and Graphical Statistics*, 12(3):531–547, 2003.

K. Kavukcuoglu, M. A. Ranzato, R. Fergus, and Y. Le-Cun. Learning invariant features through topographic filter maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

S. Kim and E. P. Xing. Tree-guided group Lasso for multi-task regression with structured sparsity. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2010.

G. S. Kimeldorf and G. Wahba. Some results on Tchebycheffian spline functions. *J. Math. Anal. Applicat.*, 33:82–95, 1971.

D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

H. Lee, A. Battle, R. Raina, and A.Y. Ng. Efficient sparse coding algorithms. *Neural Information Processing Systems*, 19:801, 2007.

Y. Lin and H. H. Zhang. Component selection and smoothing in multivariate nonparametric regression. *Annals of Statistics*, 34(5):2272–2297, 2006.

H. Liu, M. Palatucci, and J. Zhang. Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

K. Lounici, M. Pontil, A. B. Tsybakov, and S. van de Geer. Taking advantage of sparsity in multi-task learning. Technical report, Preprint arXiv:0903.1468, 2009.

L. Mackey. Deflation methods for sparse PCA. In *Advances in Neural Information Processing Systems*, 2009.

J. Mairal. *Sparse coding for machine learning, image processing and computer vision.* PhD thesis, École normale supérieure de Cachan - ENS Cachan, 2010. Available at `http://tel.archives-ouvertes.fr/tel-00595312/fr/`.

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(1):19–60, 2010a.

J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Network flow algorithms for structured sparsity. In *Advances in Neural Information Processing Systems*, 2010b.

J. Mairal, R. Jenatton, G. Obozinski, and F. Bach. Convex and network flow optimization for structured sparsity. Technical report, Preprint arXiv:1104.1872, 2011. In submission to Journal Machine Learning Research.

S. G. Mallat. *A wavelet tour of signal processing.* Academic Press, 1999.

A. M. Martinez and A. C. Kak. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233, 2001.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *Annals of Statistics*, 34(3):1436–1462, 2006.

B. Moghaddam, Y. Weiss, and S. Avidan. Spectral bounds for sparse PCA: Exact and greedy algorithms. In *Advances in Neural Information Processing Systems*, 2006.

J. J. Moreau. Fonctions convexes duales et points proximaux dans un espace hilbertien. *C. R. Acad. Sci. Paris Sér. A Math.*, 255:2897–2899, 1962.

D. Needell and J. A. Tropp. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.

S. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. In *Advances in Neural Information Processing Systems*, 2009.

Y. Nesterov. *Introductory lectures on convex optimization: a basic course.* Kluwer Academic Publishers, 2004.

Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain, 2007.

G. Obozinski, B. Taskar, and M. I. Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Statistics and Computing*, pages 1–22, 2009.

B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

A. Quattoni, X. Carreras, M. Collins, and T. Darrell. An efficient projection for ℓ_1/ℓ_∞ regularization. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2009.

N. S. Rao, R. D. Nowak, S. J. Wright, and N. G. Kingsbury. Convex approaches to model wavelet sparsity patterns. In *International Conference on Image Processing (ICIP)*, 2011a.

N.S. Rao, R.D. Nowak, S.J. Wright, and N.G. Kingsbury. Convex approaches to model wavelet sparsity patterns. *Arxiv preprint arXiv:1104.4385*, 2011b.

F. Rapaport, E. Barillot, and J.-P. Vert. Classification of arrayCGH data using fused SVM. *Bioinformatics*, 24(13):i375–i382, Jul 2008.

P. Ravikumar, H. Liu, J. Lafferty, and L. Wasserman. SpAM: Sparse additive models. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

V. Roth and B. Fischer. The group-Lasso for generalized linear models: uniqueness of solutions and efficient algorithms. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

M. Schmidt and K. Murphy. Convex structure learning in log-linear models: Beyond pairwise potentials. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

A. P. Singh and G. J. Gordon. A Unified View of Matrix Factorization Models. In *Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases-Part II*, 2008.

P. Sprechmann, I. Ramirez, G. Sapiro, and Y.C. Eldar. Collaborative Hierarchical Sparse Modeling. *Arxiv preprint arXiv:1003.0400*, 2010.

M. Stojnic, F. Parvaresh, and B. Hassibi. On the reconstruction of block-sparse signals with an optimal number of measurements. *IEEE Transactions on Signal Processing*, 57 (8):3075–3085, 2009.

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B*, pages 267–288, 1996.

J. A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, 2004.

J. A. Tropp. Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, 52(3), 2006.

B. A. Turlach, W. N. Venables, and S. J. Wright. Simultaneous variable selection. *Technometrics*, 47(3):349–363, 2005.

S.A. van de Geer. Convex approaches to model wavelet sparsity patterns. In *Proceedings of the International Congress of Mathematicians*, volume 4, pages 2351–2369, 2010.

G. Varoquaux, R. Jenatton, A. Gramfort, G. Obozinski, B. Thirion, and F. Bach. Sparse structured dictionary learning for brain resting-state activity modeling. In *NIPS Workshop on Practical Applications of Sparse Modeling: Open Issues and New Directions*, 2010.

M. J. Wainwright. Sharp thresholds for noisy and high-dimensional recovery of sparsity using ℓ_1- constrained quadratic programming. *IEEE Transactions on Information Theory*, 55:2183–2202, 2009.

D. M. Witten, R. Tibshirani, and T. Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515, 2009.

S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7):2479–2493, 2009.

T. T. Wu and K. Lange. Coordinate descent algorithms for lasso penalized regression. *Annals of Applied Statistics*, 2(1):224–244, 2008.

Z. J. Xiang, Y. T. Xi, U. Hasson, and P. J. Ramadge. Boosting with spatial regularization. In *Advances in Neural Information Processing Systems*, 2009.

G. X. Yuan, K. W. Chang, C. J. Hsieh, and C. J. Lin. Comparison of optimization methods and software for large-scale l1-regularized linear classification. *Journal of Machine Learning Research*, 11:3183–3234, 2010.

M. Yuan. High dimensional inverse covariance matrix estimation via linear programming. *Journal of Machine Learning Research*, 11:2261–2286, 2010.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society. Series B*, 68(1):49–67, 2006.

R. Zass and A. Shashua. Nonnegative sparse PCA. In *Advances in Neural Information Processing Systems*, 2007.

T. Zhang. Some sharp performance bounds for least squares regression with l1 regularization. *Annals of Statistics*, 37(5A):2109–2144, 2009.

P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.

P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *Annals of Statistics*, 37(6A):3468–3497, 2009.

H. Zou. The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

H. Zou, T. Hastie, and R. Tibshirani. Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):265–286, 2006.