A Family of Penalty Functions for Structured Sparsity

Charles A. Micchelli*
Department of Mathematics
City University of Hong Kong
83 Tat Chee Avenue, Kowloon Tong
Hong Kong
charles_micchelli@hotmail.com

Jean M. Morales
Department of Computer Science
University College London
Gower Street, London WC1E
England, UK
j.morales@cs.ucl.ac.uk

Massimiliano Pontil
Department of Computer Science
University College London
Gower Street, London WC1E
England, UK
m.pontil@cs.ucl.ac.uk

Abstract

We study the problem of learning a sparse linear regression vector under additional conditions on the structure of its sparsity pattern. We present a family of convex penalty functions, which encode this prior knowledge by means of a set of constraints on the absolute values of the regression coefficients. This family subsumes the ℓ_1 norm and is flexible enough to include different models of sparsity patterns, which are of practical and theoretical importance. We establish some important properties of these functions and discuss some examples where they can be computed explicitly. Moreover, we present a convergent optimization algorithm for solving regularized least squares with these penalty functions. Numerical simulations highlight the benefit of structured sparsity and the advantage offered by our approach over the Lasso and other related methods.

1 Introduction

The problem of sparse estimation is becoming increasingly important in machine learning and statistics. In its simplest form, this problem consists in estimating a regression vector $\beta^* \in \mathbb{R}^n$ from a data vector $y \in \mathbb{R}^m$, obtained from the model $y = X\beta^* + \xi$, where X is an $m \times n$ matrix, which may be fixed or randomly chosen and $\xi \in \mathbb{R}^m$ is a vector resulting from the presence of noise. An important rationale for sparse estimation comes from the observation that in many practical applications the number of parameters n is much larger than the data size m, but the vector β^* is known to be sparse, that is, most of its components are equal to zero. Under these circumstances, it has been shown that regularization with the ℓ_1 norm, commonly referred to as the Lasso method, provides an effective means to estimate the underlying regression vector as well as its sparsity pattern, see for example [4, 12, 15] and references therein.

In this paper, we are interested in sparse estimation under additional conditions on the sparsity pattern of β^*. In other words, not only do we expect that β^* is sparse but also that it is *structured sparse*, namely certain configurations of its nonzero components are to be preferred to others. This problem

*C.A. Micchelli is also with the Dept. of Mathematics and Statistics, State University of New York, Albany, USA. We are grateful to A. Argyriou and Y. Ying for valuable discussions. This work was supported by NSF Grant ITR-0312113, Air Force Grant AFOSR-FA9550, and EPSRC Grant EP/D071542/1.

arises is several applications, see [10] for a discussion. The prior knowledge that we consider in this paper is that the vector $|\beta^*|$, whose components are the absolute value of the corresponding components of β^*, should belong to some prescribed convex set Λ. For certain choices of Λ this implies a constraint on the sparsity pattern as well. For example, the set Λ may include vectors with some desired monotonicity constraints, or other constraints on the "shape" of the regression vector. Unfortunately, the constraint that $|\beta^*| \in \Lambda$ is nonconvex and its implementation is computational challenging. To overcome this difficulty, we propose a novel family of penalty functions. It is based on an extension of the ℓ_1 norm used by the Lasso method and involves the solution of a smooth convex optimization problem, which incorporates the structured sparsity constraints. As we shall see, a key property of our approach is that the penalty function equals the ℓ_1 norm of a vector β when $|\beta| \in \Lambda$ and it is strictly greater than the ℓ_1 norm otherwise. This observation suggests that the penalty function encourages the desired structured sparsity property.

There has been some recent research interest on structured sparsity, see [1, 2, 7, 9, 10, 11, 13, 16] and references therein. Closest to our approach are penalty methods built around the idea of mixed $\ell_1 - \ell_2$ norms. In particular, the group Lasso method [16] assumes that the components of the underlying regression vector β^* can be partitioned into prescribed groups, such that the restriction of β^* to a group is equal to zero for most of the groups. This idea has been extended in [10, 17] by considering the possibility that the groups overlap according to certain hierarchical or spatially related structures. A limitation of these methods is that they can only handle sparsity patterns forming a single connected region. Our point of view is different from theirs and provides a means to designing more general and flexible penalty functions which maintain convexity whilst modeling richer model structures. For example, we will demonstrate that our family of penalty functions can model sparsity pattern forming multiple connected regions of coefficients.

The paper is organized as follows. In Section 2 we define the learning method. In particular, we describe the associated penalty function and establish some of its important properties. In Section 3 we provide examples of penalty functions, deriving the explicit analytical form in some important cases, namely the case that the set Λ is a box or the wedge with nonincreasing coordinates. In Section 4 we address the issue of solving the learning method numerically by means of an alternating minimization algorithm. Finally, in Section 5 we provide numerical simulations with this method, showing the advantage offered by our approach.

2 Learning method

In this section, we introduce the learning method and establish some important properties of the associated penalty function. We let $\mathbb{R}_{++}$ be the positive real line and let $\mathbb{N}_n$ be the set of positive integers up to n. We prescribe a convex subset Λ of the positive orthant $\mathbb{R}^n_{++}$ and estimate β^* by a solution of the convex optimization problem

$$\min \left\{ \|X\beta - y\|_2^2 + 2\rho\Omega(\beta|\Lambda) : \beta \in \mathbb{R}^n \right\}, \tag{2.1}$$

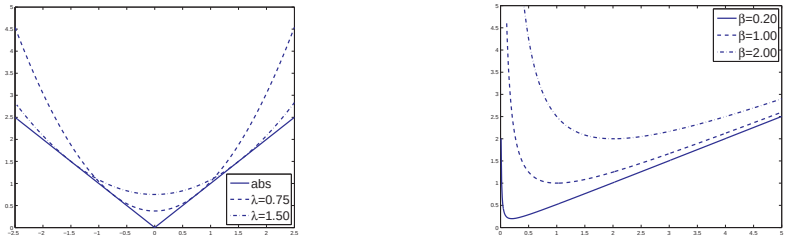
where $\|\cdot\|_2$ denotes the Euclidean norm. The penalty function takes the form

$$\Omega(\beta|\Lambda) = \inf \left\{ \Gamma(\beta, \lambda) : \lambda \in \Lambda \right\} \tag{2.2}$$

and the function $\Gamma : \mathbb{R}^n \times \mathbb{R}^n_{++} \to \mathbb{R}$ is given by the formula $\Gamma(\beta, \lambda) = \frac{1}{2} \sum_{i \in \mathbb{N}_n} \left(\frac{\beta_i^2}{\lambda_i} + \lambda_i \right)$.

Note that Γ is convex on its domain because each of its summands are likewise convex functions. Hence, when the set Λ is convex it follows that $\Omega(\cdot|\Lambda)$ is a convex function and (2.1) is a convex optimization problem. An essential idea behind our construction of this function, is that, for every $\lambda \in \mathbb{R}_{++}$, the quadratic function $\Gamma(\cdot, \lambda)$ provides a smooth approximation to $|\beta|$ from above, which is exact at $\beta = \pm\lambda$. We indicate this graphically in Figure 1-a. This fact follows immediately by the arithmetic-geometric mean inequality, namely $(a + b)/2 \geq \sqrt{ab}$. Using the same inequality it also follows that the Lasso problem corresponds to (2.1) when $\Lambda = \mathbb{R}^n_{++}$, that is it holds that $\Omega(\beta|\mathbb{R}^n_{++}) = \|\beta\|_1 := \sum_{i \in \mathbb{N}_n} |\beta_i|$. This important special case motivated us to consider the general method described above. The utility of (2.2) is that upon inserting it into (2.1) results in an optimization problem over λ and β with a continuously differentiable objective function. Hence, we have succeeded in expressing a nondifferentiable convex objective function by one which is continuously differentiable on its domain.

The next proposition provides a justification of the penalty function as a means to incorporate structured sparsity and establish circumstances for which the penalty function is a norm.



(a) (b)

Figure 1: (a): Function $\Gamma(\cdot, \lambda)$ for some values of λ; (b): Function $\Gamma(\beta, \cdot)$ for some values of β.

Proposition 2.1. *For every $\beta \in \mathbb{R}^n$, it holds that $\|\beta\|_1 \leq \Omega(\beta|\Lambda)$ and the equality holds* if and only if $|\beta| := (|\beta_i| : i \in \mathbb{N}_n) \in \Lambda$. *Moreover, if Λ is a nonempty convex cone then the function $\Omega(\cdot|\Lambda)$ is a norm and we have that $\Omega(\beta|\Lambda) \leq \omega\|\beta\|_1$, where $\omega := \max\{\Omega(e_k|\Lambda) : k \in \mathbb{N}_n\}$ and $\{e_k : k \in \mathbb{N}_n\}$ is the canonical basis of $\mathbb{R}^n$.*

Proof. By the arithmetic-geometric inequality we have that $\|\beta\|_1 \leq \Gamma(\beta, \lambda)$, proving the first assertion. If $|\beta| \in \overline{\Lambda}$, there exists a sequence $\{\lambda^k : k \in \mathbb{N}\}$ in Λ, such that $\lim_{k\to\infty} \lambda^k = |\beta|$. Since $\Omega(\beta|\Lambda) \leq \Gamma(\beta, \lambda^k)$ it readily follows that $\Omega(\beta|\Lambda) \leq \|\beta\|_1$. Conversely, if $|\beta| \in \overline{\Lambda}$, then there is a sequence $\{\lambda^k : k \in \mathbb{N}\}$ in Λ, such $\gamma(\beta, \lambda^k) \leq \|\beta_1\| + 1/k$. This inequality implies that some subsequence of this sequence converges to a $\overline{\lambda} \in \overline{\Lambda}$. Using the arithmetic-geometric we conclude that $\overline{\lambda} = |\beta|$ and the result follows. To prove the second part, observe that if Λ is a nonempty convex cone, namely, for any $\lambda \in \Lambda$ and $t \geq 0$ it holds that $t\lambda \in \Lambda$, we have that Ω is positive homogeneous. Indeed, making the change of variable $\lambda' = \lambda/|t|$ we see that $\Omega(t\beta|\Lambda) = |t|\Omega(\beta|\Lambda)$. Moreover, the above inequality, $\Omega(\beta|\Lambda) \geq \|\beta\|_1$, implies that if $\Omega(\beta|\Lambda) = 0$ then $\beta = 0$. The proof of the triangle inequality follows from the homogeneity and convexity of Ω, namely $\Omega(\alpha+\beta|\Lambda) = 2\Omega\left((\alpha+\beta)/2|\Lambda\right) \leq \Omega(\alpha|\Lambda)+\Omega(\beta|\Lambda)$. Finally, note that $\Omega(\beta|\Lambda) \leq \omega\|\beta\|_1$ if and only if $\omega = \max\{\Omega(\beta|\Lambda) : \|\beta\|_1 = 1\}$. Since Ω is convex the maximum above is achieved at an extreme point of the ℓ_1 unit ball. ■

This proposition indicates that the function $\Omega(\cdot|\Lambda)$ penalizes less vectors β which have the property that $|\beta| \in \Lambda$, hence encouraging structured sparsity. Indeed, any permutation of the coordinates of a vector β with the above property will incur in the same or a larger value of the penalty term. Moreover, for certain choices of the set Λ, some of which we describe below, the penalty function will encourage vectors which not only are sparse but also have sparsity patterns $(1_{\{|\beta_i|>0\}} : i \in \mathbb{N}_n) \in \Lambda$, where $1_{\{\cdot\}}$ denotes the indicator function.

We end this section by noting that a normalized version of the group Lasso penalty [16] is included in our setting as a special case. If $\{J_\ell : \ell \in \mathbb{N}_k\}$, $k \in \mathbb{N}_n$ form a partition of the index set $\mathbb{N}_n$, the corresponding group Lasso penalty is defined as $\Omega_{\mathrm{GL}}(\beta) = \sum_{\ell\in\mathbb{N}_k} \sqrt{|J_\ell|}\, \|\beta_{J_\ell}\|_2$, where, for every $J \subseteq \mathbb{N}_n$, we use the notation $\beta_J = (\beta_j : j \in J)$. It is a easy matter to verify that $\Omega_{\mathrm{GL}}(\cdot) = \Omega(\cdot|\Lambda)$ for $\Lambda = \{\lambda : \lambda \in \mathbb{R}^n_{++}, \lambda_j = \theta_\ell,\ j \in J_\ell,\ \ell \in \mathbb{N}_k,\ \theta_\ell > 0\}$.

3 Examples of the penalty function

We proceed to discuss some examples of the set $\Lambda \subseteq \mathbb{R}^n_{++}$ which may be used in the design of the penalty function $\Omega(\cdot|\Lambda)$. All but the first example fall into the category that Λ is a polyhedral cone, that is $\Lambda = \{\lambda : \lambda \in \mathbb{R}^n_{++}, A\lambda \geq 0\}$, where A is an $m \times n$ matrix. Thus, in view of Proposition 2.1 the function $\Omega(\cdot|\Lambda)$ is a norm.

The first example corresponds to the prior knowledge that the magnitude of the components of the regression vector should be in some prescribed intervals.

Example 3.1. *We choose $a, b \in \mathbb{R}^n$, $0 < a \leq b$ and define the corresponding box as $B[a, b] := \bigotimes_{i\in\mathbb{N}_n} [a_i, b_i]$.*

The theorem below establishes the form of the box penalty; see also [8, 14] for related penalty functions. To state our result, we define, for every $t \in \mathbb{R}$, the function $(t)_+ = \max(0, t)$.

Theorem 3.1. *We have that*

$$\Omega(\beta|B[a,b]) = \|\beta\|_1 + \sum_{i\in\mathbb{N}_n}\left(\frac{1}{2a_i}(a_i-|\beta_i|)_+^2 + \frac{1}{2b_i}(|\beta_i|-b_i)_+^2\right).$$

Moreover, the components of the vector $\lambda(\beta) := \operatorname{argmin}\{\Gamma(\beta,\lambda) : \lambda \in B[a,b]\}$ *are given by the equations* $\lambda_i(\beta) = |\beta_i| + (a_i - |\beta_i|)_+ - (|\beta_i| - b)_+$, $i \in \mathbb{N}_n$.

Proof. Since $\Omega(\beta|B[a,b]) = \sum_{i\in\mathbb{N}_n}\Omega(\beta_i|[a_i,b_i])$ it suffices to establish the result in the case $n=1$. We shall show that if $a,b,\beta\in\mathbb{R}$, $a\le b$ then

$$\Omega(\beta|[a,b]) = |\beta| + \frac{1}{2a}(a-|\beta|)_+^2 + \frac{1}{2b}(|\beta|-b)_+^2. \tag{3.1}$$

Since both sides of the above equation are continuous functions of β it suffices to prove this equation for $\beta\in\mathbb{R}\backslash\{0\}$. In this case, the function $\Gamma(\beta,\cdot)$ is strictly convex in the second argument, and so, has a unique minimum in $\mathbb{R}_{++}$ at $\lambda=|\beta|$, see also Figure 1-b. Moreover, if $|\beta|\le a$ the constrained minimum occurs at $\lambda=a$, whereas if $|\beta|\ge b$, it occurs at $\lambda=b$. This establishes the formula for $\lambda(\beta)$. Consequently, we have that

$$\Omega(\beta|[a,b]) = |\beta|1_{\{a\le|\beta|\le b\}} + \frac{1}{2}\left(\frac{\beta^2}{a}+a\right)1_{\{|\beta|<a\}} + \frac{1}{2}\left(\frac{\beta^2}{b}+b\right)1_{\{|\beta|>b\}}.$$

Equation (3.1) now follows by a direct computation. ■

Note that the function in equation (3.1) is a concatenation of two quadratic functions, connected together with a linear function. Thus, the box penalty will favor sparsity only for $a=0$, case that is defined by a limiting argument.

The second example implements the prior knowledge that the coordinates of the vector λ are ordered in a non increasing fashion.

Example 3.2. *We define the wedge as* $W = \{\lambda : \lambda\in\mathbb{R}_{++}^n,\ \lambda_j\ge\lambda_{j+1},\ j\in\mathbb{N}_{n-1}\}$.

We say that a partition $\mathcal{J} = \{J_\ell : \ell\in\mathbb{N}_k\}$ of $\mathbb{N}_n$ is *contiguous* if for all $i\in J_\ell, j\in J_{\ell+1}$, $\ell\in\mathbb{N}_{k-1}$, it holds that $i<j$. For example, if $n=3$, partitions $\{\{1,2\},\{3\}\}$ and $\{\{1\},\{2\},\{3\}\}$ are contiguous but $\{\{1,3\},\{2\}\}$ is not.

Theorem 3.2. *For every* $\beta\in(\mathbb{R}\backslash\{0\})^n$ *there is a unique contiguous partition* $\mathcal{J}=\{J_\ell : \ell\in\mathbb{N}_k\}$ *of* $\mathbb{N}_n$, $k\in\mathbb{N}_n$, *such that*

$$\Omega(\beta|W) = \sum_{\ell\in\mathbb{N}_k}\sqrt{|J_\ell|}\,\|\beta_{J_\ell}\|_2. \tag{3.2}$$

Moreover, the components of the vector $\lambda(\beta) = \operatorname{argmin}\{\Gamma(\beta,\lambda) : \lambda\in W\}$ *are given by*

$$\lambda_j(\beta) = \frac{\|\beta_{J_\ell}\|_2}{\sqrt{|J_\ell|}},\ j\in J_\ell,\ \ell\in\mathbb{N}_k \tag{3.3}$$

and, for every $\ell\in\mathbb{N}_k$ *and subset* $K\subset J_\ell$ *formed by the first* $k<|J_\ell|$ *elements of* J_ℓ, *it holds that*

$$\frac{\|\beta_K\|_2}{\sqrt{k}} > \frac{\|\beta_{J_\ell\backslash K}\|_2}{\sqrt{|J_\ell|-k}}. \tag{3.4}$$

The partition $\mathcal{J}$ appearing in the theorem is determined by the set of inequalities $\lambda_j\ge\lambda_{j+1}$ which are an equality at the minimum. This set is identified by examining the Karush-Kuhn-Tucker optimality conditions [3] of the optimization problem (2.2) for $\Lambda=W$. The detailed proof is reported in the supplementary material. Equations (3.3) and (3.4) indicate a strategy to compute the partition associated with a vector β. We explain how to do this in Section 4.

An interesting property of the Wedge penalty is that it has the form of a group Lasso penalty (see the discussion at the end of Section 2) with groups not fixed *a-priori* but depending on the location of the vector β. The groups are the elements of the partition $\mathcal{J}$ and are identified by certain convex

constraints on the vector β. For example, for $n = 2$ we obtain that $\Omega(\beta|W) = \|\beta\|_1$ if $|\beta_1| > |\beta_2|$ and $\Omega(\beta|W) = \sqrt{2}\|\beta\|_2$ otherwise. For $n = 3$, we have that

$$\Omega(\beta|W) = \begin{cases} \|\beta\|_1, & \text{if } |\beta_1| > |\beta_2| > |\beta_3| & \mathcal{J} = \{\{1\},\{2\},\{3\}\} \\ \sqrt{2(\beta_1^2+\beta_2^2)} + |\beta_3|, & \text{if } |\beta_1| \le |\beta_2| \text{ and } \beta_1^2+\beta_2^2 > 2\beta_3^2 & \mathcal{J} = \{\{1,2\},\{3\}\} \\ |\beta_1| + \sqrt{2(\beta_2^2+\beta_3^2)}, & \text{if } |\beta_2| \le |\beta_3| \text{ and } 2\beta_1^2 > \beta_2^2+\beta_3^2 & \mathcal{J} = \{\{1\},\{2,3\}\} \\ \sqrt{3(\beta_1^2+\beta_2^2+\beta_3^2)}, & \text{otherwise} & \mathcal{J} = \{\{1,2,3\}\} \end{cases}$$

where we have also reported the partition involved in each case.

The next example is an extension of the wedge set which is inspired by previous work on the group Lasso estimator with hierarchically overlapping groups [17]. It models vectors whose magnitude is ordered according to a graphical structure. Within this context, the wedge corresponds to the set associated with a line graph.

Example 3.3. *We let A be the incidence matrix of a directed graph and choose $\Lambda = \{\lambda : \lambda \in \mathbb{R}^n_{++}, A\lambda \ge 0\}$.*

We have confirmed that Theorem 3.2 extends to the case that the graph is a tree but the general case is yet to be understood. We postpone this discussion to a future occasion.

Next, we note that the wedge may equivalently be expressed as the constraint that the difference vector $D^1(\lambda) := (\lambda_{j+1} - \lambda_j : j \in \mathbb{N}_{n-1})$ is less than or equal to zero. Our next example extends this observation by using the higher order difference operator, which is given by the formula $D^k(\lambda) = \Big(\lambda_{j+k} + \sum_{\ell \in \mathbb{N}_k} (-1)^\ell \binom{k}{\ell} \lambda_{j+k-\ell} : j \in \mathbb{N}_{n-k}\Big)$.

Example 3.4. *For every $k \in \mathbb{N}_n$ we define the set $W^k := \{\lambda : \lambda \in \mathbb{R}^n_{++},\ D^k(\lambda) \ge 0\}$.*

The corresponding penalty $\Omega(\cdot|W^k)$ encourages vectors whose sparsity pattern is concentrated on at most k different contiguous regions. The case $k = 1$ essentially corresponds to the wedge, while the case $k = 2$ includes vectors which have a convex "profile" and whose sparsity pattern is concentrated either on the first elements of the vector, on the last, or on both.

We end this section by discussing a useful construction which may be applied to generate new penalty functions from available ones. It is obtained by composing a set $\Theta \subseteq \mathbb{R}^k_{++}$ with a linear transformation, modeling the sum of the components of a vector, across the elements of a prescribed partition $\{P_\ell : \ell \in \mathbb{N}_k\}$ of $\mathbb{N}_n$. That is, we let $\Lambda = \{\lambda : \lambda \in \mathbb{R}^n_{++},\ (\sum_{j \in P_\ell} \lambda_j : \ell \in \mathbb{N}_k) \in \Theta\}$. We use this construction in the composite wedge experiments in Section 5.

4 Optimization method

In this section, we address the issue of implementing the learning method (2.1) numerically. Since the penalty function $\Omega(\cdot|\Lambda)$ is constructed as the infimum of a family of quadratic regularizers, the optimization problem (2.1) reduces to a simultaneous minimization over the vectors β and λ. For a fixed $\lambda \in \Lambda$, the minimum over $\beta \in \mathbb{R}^n$ is a standard Tikhonov regularization and can be solved directly in terms of a matrix inversion. For a fixed β, the minimization over $\lambda \in \Lambda$ requires computing the penalty function (2.2). These observations naturally suggests an alternating minimization algorithm, which has already been considered in special cases in [1]. To describe our algorithm we choose $\epsilon > 0$ and introduce the mapping $\phi^\epsilon : \mathbb{R}^n \to \mathbb{R}^n_{++}$, whose i-th coordinate at $\beta \in \mathbb{R}^n$ is given by $\phi^\epsilon_i(\beta) = \sqrt{\beta_i^2 + \epsilon}$. For $\beta \in (\mathbb{R}\backslash\{0\})^n$, we also let $\lambda(\beta) = \operatorname{argmin}\{\Gamma(\beta,\lambda) : \lambda \in \Lambda\}$. The alternating minimization algorithm is defined as follows: choose, $\lambda^0 \in \Lambda$ and, for $k \in \mathbb{N}$, define the iterates

$$\beta^k = \operatorname{diag}(\lambda^{k-1})(\operatorname{diag}(\lambda^{k-1})X^\top X + \rho I)^{-1} y \tag{4.1}$$

$$\lambda^k = \lambda(\phi^\epsilon(\beta^k)). \tag{4.2}$$

The following theorem establishes convergence of this algorithm. Its proof is presented in the supplementary material.

Theorem 4.1. *If the set Λ is convex and, for all $a, b \in \mathbb{R}$ with $0 < a < b$, the set $\Lambda_{a,b} := [a,b]^n \cap \Lambda$ is a nonempty, compact subset of the interior of Λ then the iterations (4.1)–(4.2) converges to the vector*

Initialization: $k \leftarrow 0$
Input: $\beta \in \mathbb{R}^n$; **Output:** $J_1, \ldots, J_k$
for $t = 1$ **to** n **do**
  $J_{k+1} \leftarrow \{t\}$; $k \leftarrow k+1$
  while $k > 1$ **and** $\frac{\|\beta_{J_{k-1}}\|_2}{\sqrt{|J_{k-1}|}} \leq \frac{\|\beta_{J_k}\|_2}{\sqrt{|J_k|}}$
    $J_{k-1} \leftarrow J_{k-1} \cup J_k$; $k \leftarrow k-1$
  end
end

Figure 2: Iterative algorithm to compute the wedge penalty

$\gamma(\epsilon) := \operatorname{argmin}\left\{\|y - X\beta\|_2^2 + 2\rho\Omega(\phi^\epsilon(\beta)|\Lambda) : \beta \in \mathbb{R}^n\right\}$. *Moreover, any convergent subsequence of the sequence* $\{\gamma\left(\frac{1}{\ell}\right) : \ell \in \mathbb{N}\}$ *converges to a solution of the optimization problem* (2.1)*.*

The most challenging step in the alternating algorithm is the computation of the vector λ^k. Fortunately, if Λ is a second order cone, problem (2.2) defining the penalty function $\Omega(\cdot|\Lambda)$ may be reformulated as a second order cone program (SOCP), see e.g. [5]. To see this, we introduce an additional variable $t \in \mathbb{R}^n$ and note that

$$\Omega(\beta|\Lambda) = \min\left\{\sum_{i\in\mathbb{N}_n} t_i + \lambda_i : \|(2\beta_i, t_i - \lambda_i)\|_2 \leq t_i + \lambda_i, t_i \geq 0,\ i \in \mathbb{N}_n,\ \lambda \in \Lambda\right\}.$$

In particular, in all examples in Section 3, the set Λ is formed by linear constraints and, so, problem (2.2) is an SOCP. We may then use available tool-boxes to compute the solution of this problem. However, in special cases the computation of the penalty function may be significantly facilitated by using the analytical formulas derived in Section 3. Here, for simplicity we describe how to do this in the case of the wedge penalty. For this purpose we say that a vector $\beta \in \mathbb{R}^n$ is admissible if, for every $k \in \mathbb{N}_n$, it holds that $\|\beta_{\mathbb{N}_k}\|_2/\sqrt{k} \leq \|\beta\|_2/\sqrt{n}$.

The proof of the next lemma is straightforward and we do not elaborate on the details.

Lemma 4.1. *If* $\beta \in \mathbb{R}^n$ *and* $\delta \in \mathbb{R}^p$ *are admissible and* $\|\beta\|_2/\sqrt{n} \leq \|\delta\|_2/\sqrt{p}$ *then* (β, δ) *is admissible.*

The iterative algorithm presented in Figure 2 can be used to find the partition $\mathcal{J} = \{J_\ell : \ell \in \mathbb{N}_k\}$ and, so, the vector $\lambda(\beta)$ described in Theorem 3.2. The algorithm processes the components of vector β in a sequential manner. Initially, the first component forms the only set in the partition. After the generic iteration $t-1$, where the partition is composed of k sets, the index of the next components, t, is put in a new set J_{k+1}. Two cases can occur: the means of the squares of the sets are in strict descending order, or this order is violated by the last set. The latter is the only case that requires further action, so the algorithm merges the last two sets and repeats until the sets in the partition are fully ordered. Note that, since the only operation performed by the algorithm is the merge of admissible sets, Lemma 4.1 ensures that after each step t the current partition satisfies the conditions (3.4). Moreover, the *while* loop ensures that after each step the current partition satisfies, for every $\ell \in \mathbb{N}_{k-1}$, the constraints $\|\beta_{J_\ell}\|_2\sqrt{|J_\ell|} > \|\beta_{J_{\ell+1}}\|_2\sqrt{|J_{\ell+1}|}$. Thus, the output of the algorithm is the partition $\mathcal{J}$ defined in Theorem 3.2. In the actual implementation of the algorithm, the means of squares of each set can be saved. This allows us to compute the mean of squares of a merged set as a weighted mean, which is a constant time operation. Since there are $n-1$ consecutive terms in total, this is also the maximum number of merges that the algorithm can perform. Each merge requires exactly one additional test, so we can conclude that the running time of the algorithm is linear.

5 Numerical simulations

In this section we present some numerical simulations with the proposed method. For simplicity, we consider data generated noiselessly from $y = X\beta^*$, where $\beta^* \in \mathbb{R}^{100}$ is the true underlying regression vector, and X is an $m \times 100$ input matrix, m being the sample size. The elements of X are generated i.i.d. from the standard normal distribution, and the columns of X are then normalized such that their ℓ_2 norm is 1. Since we consider the noiseless case, we solve the interpolation problem $\min\{\Omega(\beta) : y = X\beta\}$, for different choices of the penalty function Ω. In practice, we solve problem (2.1) for a tiny value of the parameter $\rho = 10^{-8}$, which we found to be sufficient to ensure that the

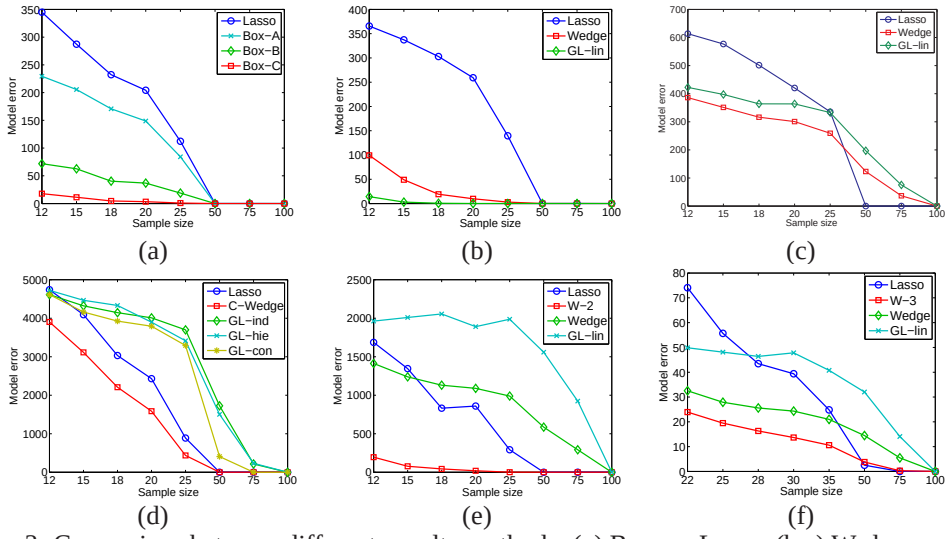


Figure 3: Comparison between different penalty methods: (a) Box vs. Lasso; (b,c) Wedge vs. Hierarchical group Lasso; (d) Composite wedge; (e) Convex; (f) Cubic. See text for more information

error term in (2.1) is negligible at the minimum. All experiments were repeated 50 times, generating each time a new matrix X. In the figures we report the average of the model error $\mathbb{E}[\|\hat{\beta} - \beta^*\|_2^2]$ of the vector $\hat{\beta}$ learned by each method, as a function of the sample size m. In the following, we discuss a series of experiments, corresponding to different choices for the model vector β^* and its sparsity pattern. In all experiments, we solved the optimization problem (2.1) with the algorithm presented in Section 4. Whenever possible we solved step (4.2) using the formulas derived in Section 3 and resorted to the solver CVX (*http://cvxr.com/cvx/*) in the other cases.

Box. In the first experiment the model is 10-sparse, where each nonzero component, in a random position, is an integer uniformly sampled in the interval $[-10, 10]$. We wish to show that the more accurate the prior information about the model is, the more precise the estimate will be. We use a box penalty (see Theorem 3.1) constructed "around" the model, imagining that an oracle tells us that each component $|\beta_i^*|$ is bounded within an interval. We consider three boxes $B[a, b]$ of different sizes, namely $a_i = (r - |\beta_i^*|)_+$ and $b_i = (|\beta_i^*| - r)_+$ and radii $r = 5, 1$ and 0.1, which we denote as Box-A, Box-B and Box-C, respectively. We compare these methods with the Lasso – see Figure 3-a. As expected, the three box penalties perform better. Moreover, as the radius of a box diminishes, the amount of information about the true model increases, and the performance improves.

Wedge. In the second experiment, we consider a regression vector, whose components are nonincreasing in absolute value and only a few are nonzero. Specifically, we choose a 10-sparse vector: $\beta_j^* = 11 - j$, if $j \in \mathbb{N}_{10}$ and zero otherwise. We compare the Lasso, which makes no use of such ordering information, with the wedge penalty $\Omega(\beta|W)$ (see Example 3.2 and Theorem 3.2) and the hierarchical group Lasso in [17], which both make use of such information. For the group Lasso we choose $\Omega(\beta) = \sum_{\ell \in \mathbb{N}_{100}} \|\beta_{J_\ell}\|$, with $J_\ell = \{\ell, \ell+1, \ldots, 100\}$, $\ell \in \mathbb{N}_{100}$. These two methods are referred to as "Wedge" and "GL-lin" in Figure 3-b, respectively. As expected both methods improve over the Lasso, with "GL-lin" being the best of the two. We further tested the robustness of the methods, by adding two additional nonzero components with value of 10 to the vector β^* in a random position between 20 and 100. This result, reported in Figure 3-c, indicates that "GL-lin" is more sensitive to such a perturbation.

Composite wedge. Next we consider a more complex experiment, where the regression vector is sparse within different contiguous regions $P_1, \ldots, P_{10}$, and the ℓ_1 norm on one region is larger than the ℓ_1 norm on the next region. We choose sets $P_i = \{10(i-1)+1, \ldots, 10i\}$, $i \in \mathbb{N}_{10}$ and generate a 6-sparse vector β^* whose i-th nonzero element has value $31 - i$ (decreasing) and is in a random position in P_i, for $i \in \mathbb{N}_6$. We encode this prior knowledge by choosing $\Omega(\beta|\Lambda)$ with $\Lambda = \left\{\lambda \in \mathbb{R}^{100} : \|\lambda_{P_i}\|_1 \geq \|\lambda_{P_{i+1}}\|_1,\ i \in \mathbb{N}_9\right\}$. This method constraints the sum of the sets to be nonincreasing and may be interpreted as the composition of the wedge set with an average operation across the sets P_i, see the discussion at the end of Section 3. This method, which is referred to as "C-Wedge" in Figure 3-d, is compared to the Lasso and to three other versions of the group Lasso. The

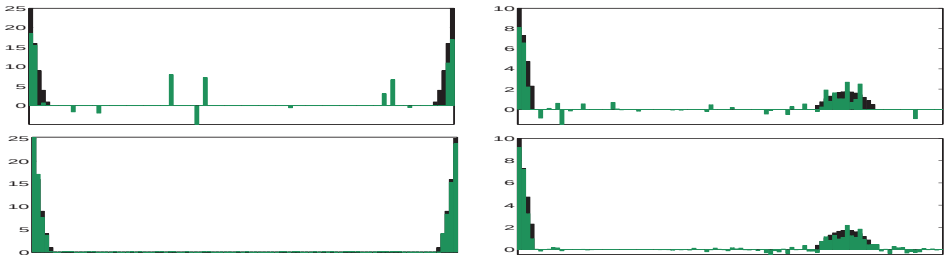


Figure 4: Lasso vs. penalty $\Omega(\cdot|W^2)$ (left) and $\Omega(\cdot|W^3)$ (Right); see text for more information.

first is a standard group Lasso with the nonoverlapping groups $J_i = P_i$, $i \in \mathbb{N}_{10}$, thus encouraging the presence of sets of zero elements, which is useful because there are 4 such sets. The second is a variation of the hierarchical group Lasso discussed above with $J_i = \cup_{j=i}^{10} P_j$, $i \in \mathbb{N}_{10}$. A problem with these approaches is that the ℓ_2 norm is applied at the level of the individual sets P_i, which does not promote sparsity within these sets. To counter this effect we can enforce contiguous nonzero patterns within each of the P_i, as proposed by [10]. That is, we consider as the groups the sets formed by all sequences of $q \in \mathbb{N}_9$ consecutive elements at the beginning or at the end of each of the sets P_i, for a total of 180 groups. These three groupings will be referred to as "GL-ind", "GL-hie'', "GL-con" in Figure 3-d, respectively. This result indicates the advantage of "C-Wedge" over the other methods considered. In particular, the group Lasso methods fall behind our method and the Lasso, with "GL-con" being slight better than "GL-ind" and "GL-hie". Notice also that all group Lasso methods gradually diminish the model error until they have a point for each dimension, while our method and the Lasso have a steeper descent, reaching zero at a number of points which is less than half the number of dimensions.

Convex and Cubic. To show the flexibility of our framework, we consider two further examples of sparse regression vectors with additional structured properties. In the first example, most of the components of this vector are zero, but the first and the last few elements follow a discrete convex trend. Specifically, we choose $\beta^* = (5^2, 4^2, 3^2, 2^2, 1, 0, \ldots, 0, 1, 2^2, 3^2, 4^2, 5^2) \in \mathbb{R}^{100}$. In this case, we expect the penalty function $\Omega(\beta|W^2)$ to outperform the Lasso, because it favors vectors with convex shape. Results are shown in Figure3-e, where this penalty is named "W-2". In lack of other specific methods to impose this convex shape constraint, and motivating by the fact that the first few components decrease, we compare it with two methods that favors a learned vector that is decreasing: the Wedge and the group Lasso with $J_k = \{k, \ldots, 100\}$ for $k \in \mathbb{N}_{100}$. These methods and the Lasso fail to use the prior knowledge of convexity, and are outperformed by using the constraint set W^2. The second example considers the case where $|\beta^*| \in W^3$, namely the differences of the second order are decreasing. This vector is constructed from the cubic polynomial $p(t) = -t(t-1.5)(t+6.5)$. The polynomial is evaluated at 100 equally spaced (0.1) points, starting from -7. The resulting vector starts with 5 nonzero components and has then a bump of another 15 elements. We use our method with the penalty $\Omega(\beta|W^3)$, which is referred to as "W-3" in the Figure. The model error, compared again with "W-1" and group Lasso linear, is shown in Figure 3-f. Finally, Figure 4 displays the regression vector found by the Lasso and the vector learned by "W-2" (left) and by the Lasso and "W-3" (right), in a single run with sample size of 15 and 35, respectively. The estimated vectors (green) are superposed to the true vector (black). Our method provides a better estimate than the Lasso in both cases.

Conclusion

We proposed a family of penalty functions that can be used to model structured sparsity in linear regression. We provided theoretical, algorithmic and computational information about this new class of penalty functions. Our theoretical observations highlight the generality of this framework to model structured sparsity. An important feature of our approach is that it can deal with richer model structures than current approaches while maintaining convexity of the penalty function. Our practical experience indicates that these penalties perform well numerically, improving over state of the art penalty methods for structure sparsity, suggesting that our framework is promising for applications. In the future, it would be valuable to extend the ideas presented here to learning nonlinear sparse regression models. There is also a need to clarify the rate of convergence of the algorithm presented here.

References

[1] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73(3):243–272, 2008.

[2] R.G. Baraniuk, V. Cevher, M.F. Duarte, and C. Hegde. Model-based compressive sensing. *Information Theory, IEEE Transactions on*, 56(4):1982 –2001, 2010.

[3] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

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

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[6] J.M. Danskin. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

[7] J. Huang, T. Zhang, and D. Metaxas. Learning with structured sparsity. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 417–424. ACM, 2009.

[8] L. Jacob. Structured priors for supervised learning in computational biology. 2009. Ph.D. Thesis.

[9] L. Jacob, G. Obozinski, and J.-P. Vert. Group lasso with overlap and graph lasso. In *International Conference on Machine Learning (ICML 26)*, 2009.

[10] R. Jenatton, J.-Y. Audibert, and F. Bach. Structured variable selection with sparsity-inducing norms. arXiv:0904.3523v2, 2009.

[11] S. Kim and E.P. Xing. Tree-guided group lasso for multi-task regression with structured sparsity. Technical report, 2009. arXiv:0909.1373.

[12] K. Lounici. Sup-norm convergence rate and sign concentration property of Lasso and Dantzig estimators. *Electronic Journal of Statistics*, 2:90–102, 2008.

[13] K. Lounici, M. Pontil, A.B Tsybakov, and S. van de Geer. Taking advantage of sparsity in multi-task learning. In *Proc. of the 22nd Annual Conference on Learning Theory (COLT)*, 2009.

[14] A.B. Owen. A robust hybrid of lasso and ridge regression. In *Prediction and discovery: AMS-IMS-SIAM Joint Summer Research Conference, Machine and Statistical Learning: Prediction and Discovery*, volume 443, page 59, 2007.

[15] S.A. van de Geer. High-dimensional generalized linear models and the Lasso. *Annals of Statistics*, 36(2):614, 2008.

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

[17] P. Zhao, G. Rocha, and B. Yu. Grouped and hierarchical model selection through composite absolute penalties. *Annals of Statistics*, 37(6A):3468–3497, 2009.

A Appendix

In this appendix we provide the proof of Theorems 3.2 and 4.1.

A.1 Proof of Theorem 3.2

Before proving the theorem we require some additional notation. Given any two disjoint subsets $J, K \subseteq \mathbb{N}_n$ we define the region

$$Q_{J,K} = \left\{ \beta : \beta \in \mathbb{R}^n, \frac{\|\beta_J\|_2^2}{|J|} > \frac{\|\beta_K\|_2^2}{|K|} \right\}.$$

Note that the boundary of this region is determined by the zero set of a homogeneous polynomial of degree two. We also need the following construction.

Definition A.1. *For every subset $S \subseteq \mathbb{N}_{n-1}$ we set $k = |S| + 1$ and label the elements of S in increasing order as $S = \{j_\ell : \ell \in \mathbb{N}_{k-1}\}$. We associate with the subset S a contiguous partition of $\mathbb{N}_n$, given by $\mathcal{J}(S) = \{J_\ell : \ell \in \mathbb{N}_k\}$, where we define $J_\ell := \{j_{\ell-1} + 1, j_\ell\}$ and set $j_0 = 0$ and $j_k = n$.*

A subset S of $\mathbb{N}_{n-1}$ also induces two regions in $\mathbb{R}^n$ which play a central role in the identification of the wedge penalty. First, we describe the region which "crosses over" the induced partition $\mathcal{J}(S)$. This is defined to be the set

$$O_S := \bigcap \left\{ Q_{J_\ell, J_{\ell+1}} : \ell \in \mathbb{N}_{k-1} \right\}. \tag{A.1}$$

In other words, $\beta \in O_S$ if the average of the square of its components within each region J_ℓ strictly decreases with ℓ. The next region which is essential in our analysis is the "stays within" region, induced by the partition $\mathcal{J}(S)$. This region requires the notation $J_{\ell,q} := \{j : j \in J_\ell, j \leq q\}$ and is defined by the equation

$$I_S := \bigcap \left\{ \overline{Q}_{J_\ell, J_{\ell,q}} : q \in J_\ell, \ell \in \mathbb{N}_k \right\}, \tag{A.2}$$

where $\overline{Q}$ denotes the closure of the set Q. In other words, all vectors β within this region have the property that, for every set $J_\ell \in \mathcal{J}(S)$, the average of the square of a first segment of components of β within this set is not greater than the average over J_ℓ. We note that if S is the empty set the above notation should be interpreted as $O_S = \mathbb{R}^n$ and

$$I_S = \bigcap \{ \overline{Q}_{\mathbb{N}_n, \mathbb{N}_q} : q \in \mathbb{N}_n \}.$$

We also introduce, for every $S \in \mathbb{N}_{n-1}$ the sets

$$U_S := O_S \cap I_S \cap (\mathbb{R} \backslash \{0\})^n.$$

We shall prove the following slightly more general version the Theorem 3.2

Theorem A.1. *The collection of sets $\mathcal{U} := \{U_S : S \subseteq \mathbb{N}_{n-1}\}$ forms a partition of $(\mathbb{R}\backslash\{0\})^n$. For each $\beta \in (\mathbb{R}\backslash\{0\})^n$ there is a unique $S \in \mathbb{N}_{n-1}$ such that $\beta \in U_S$, and*

$$\Omega(\beta|W) = \sum_{\ell \in \mathbb{N}_k} \sqrt{|J_\ell|}\, \|\beta_{J_\ell}\|_2, \tag{A.3}$$

where $k = |S| + 1$. Moreover, the components of the vector $\lambda(\beta) := \operatorname{argmin}\{\Gamma(\beta, \lambda) : \lambda \in W\}$ is given by the equations $\lambda_j(\beta) = \mu_\ell,\ j \in J_\ell,\ \ell \in \mathbb{N}_k$, where

$$\mu_\ell = \frac{\|\beta_{J_\ell}\|_2}{\sqrt{|J_\ell|}}. \tag{A.4}$$

Proof. First, let us observe that there are $n-1$ inequality constraints defining W. It readily follows that all vectors in this constraint set are *regular*, in the sense of optimization theory, see [3, p. 279]. Hence, we can appeal to [3, Prop. 3.3.4, p. 316 and Prop. 3.3.6, p. 322], which state that $\lambda \in \mathbb{R}^n_{++}$

is a solution to the minimum problem determined by the wedge penalty, if and only if there exists a vector $\alpha = (\alpha_i : i \in \mathbb{N}_{n-1})$ with nonnegative components such that

$$-\frac{\beta_j^2}{\lambda_j^2} + 1 + \alpha_{j-1} - \alpha_j = 0, \quad j \in \mathbb{N}_n, \tag{A.5}$$

where we set $\alpha_0 = \alpha_n = 0$. Furthermore, the following complementary slackness conditions hold true

$$\alpha_j(\lambda_{j+1} - \lambda_j) = 0, \; j \in \mathbb{N}_{n-1}. \tag{A.6}$$

To unravel these equations, we let $S := \{j : \lambda_j > \lambda_{j+1}, j \in \mathbb{N}_{n-1}\}$, which is the subset of indexes corresponding to the constraints that are not tight. When $k \geq 2$, we express this set in the form $\{j_\ell : \ell \in \mathbb{N}_{k-1}\}$ where $k = |S| + 1$.

As explained in Definition A.1, the set S induces the partition $\mathcal{J}(S) = \{J_\ell : \ell \in \mathbb{N}_k\}$ of $\mathbb{N}_n$. When $k = 1$ our notation should be interpreted to mean that S is empty and the partition $\mathcal{J}(S)$ consists only of $\mathbb{N}_n$. In this case, it is easy to solve the equations (A.5) and (A.6). In fact, all components of the vector λ have a common value, say $\mu > 0$, and by summing both sides of equation (A.5) over $j \in \mathbb{N}_n$ we obtain that $\mu^2 = \|\beta\|_2^2/n$. Moreover, summing both sides of the same equation over $j \in \mathbb{N}_q$ we obtain that $\alpha_q = -\sum_{j\in\mathbb{N}_q} \beta_j^2/\mu^2 + q$ and, since $\alpha_q \geq 0$ we conclude that $\beta \in I_S = U_S$.

We now consider the case that $k \geq 2$. Hence, the vector λ has equal components on each subset J_ℓ, which we denote by μ_ℓ, $\ell \in \mathbb{N}_{k-1}$. The definition of the set S implies that the μ_ℓ are strictly decreasing and equation (A.6) implies that $\alpha_j = 0$, for every $j \in S$. Summing both sides of equation (A.5) over $j \in J_\ell$ we obtain that

$$-\frac{1}{\mu_\ell^2} \sum_{j\in J_\ell} \beta_j^2 + |J_\ell| = 0$$

from which equation (A.4) follows. Since the μ_ℓ are strictly decreasing, we conclude that $\beta \in O_S$. Moreover, choosing $q \in J_\ell$ and summing both sides of equations (A.5) over $j \in J_{\ell,q}$ we obtain that

$$0 \leq \alpha_q = -\frac{\|\beta_{J_{\ell,q}}\|_2^2}{\mu_\ell^2} + |J_{\ell,q}|$$

which implies that $\beta \in \overline{Q}_{J_\ell, J_{\ell,q}}$. Since this holds for every $q \in J_\ell$ and $\ell \in N_k$ we conclude that $\beta \in I_S$ and therefore, it follows that $\beta \in U_S$.

In summary, we have shown that $\beta \in U_S$. In particular, this implies that the collection of sets $\mathcal{U}$ covers $(\mathbb{R}\backslash\{0\})^n$. Next, we show that the elements of $\mathcal{U}$ are disjoint. To this end, we observe that, the computation described above can be *reversed*. That is to say, conversely for *any* $S \subseteq \mathbb{N}_{n-1}$ and $\beta \in U_S$ we conclude that the vectors α and λ define above solve the equations (A.5) and (A.6). Since the wedge penalty function is *strictly convex* we know that equations (A.5) and (A.6) have a unique solution. Now, if $\beta \in U_S \cap U_{S'}$ then it must follow that $\lambda = \lambda'$. Consequently, since the vectors λ and λ' are a constant on any element of their respective partitions $\mathcal{J}(S)$ and $\mathcal{J}(S')$, strictly decreasing from one element to the next in those partition, it must be the case that $S_1 = S_2$. ■

We note that if some components of β are zero we may compute $\Omega(\beta|\Lambda)$ as a limiting process, since the function $\Omega(\cdot|\Lambda)$ is continuous.

Proof of Theorem 4.1 We divide the proof into several steps. To this end, we define

$$E_\epsilon(\beta, \lambda) := \|y - X\beta\|_2^2 + 2\rho\Gamma(\phi^\epsilon(\beta), \lambda)$$

and let $\beta(\lambda) := \operatorname{argmin}\{E_\epsilon(\alpha, \lambda) : \alpha \in \mathbb{R}^n\}$.

Step 1. We define two sequences, $\theta_k = E_\epsilon(\beta^k, \lambda^{k-1})$ and $\nu_k = E_\epsilon(\beta^k, \lambda^k)$ and observe, for any $k \geq 2$, that

$$\theta_{k+1} \leq \nu_k \leq \theta_k \leq \nu_{k-1}. \tag{A.7}$$

These inequalities follow directly from the definition of the alternating algorithm, see equations (4.1) and (4.2).

Step 2. We define the compact set $B = \{\beta : \beta \in \mathbb{R}^n, \|\beta\|_1 \leq \theta_1\}$. From the first inequality in Proposition 2.1, $\|\beta\|_1 \leq \Omega(\beta|\Lambda)$, and inequality (A.7) we conclude, for every $k \in \mathbb{N}$, that $\beta^k \in B$.

Step 3. We define a function $g : \mathbb{R}^n \to \mathbb{R}$ at $\beta \in \mathbb{R}^n$ as

$$g(\beta) = \min \left\{ E_\epsilon(\alpha, \lambda(\phi^\epsilon(\beta))) : \alpha \in \mathbb{R}^n \right\}.$$

We claim that g is continuous on B. In fact, there exists a constant $\kappa > 0$ such that, for every $\gamma^1, \gamma^2 \in B$, it holds that

$$|g(\gamma^1) - g(\gamma^2)| \leq \kappa \|\lambda(\phi^\epsilon(\gamma^1)) - \lambda(\phi^\epsilon(\gamma^2))\|_\infty. \tag{A.8}$$

The essential ingredient in the proof of this inequality is the fact that by our hypothesis on the set Λ there exists constant $\overline{a}$ and $\overline{b}$ such that, for all $\beta \in B$, $\lambda(\phi^\epsilon(\beta)) \in [\overline{a}, \overline{b}]^n$. This fact follows by Danskin's Theorem [6].

Step 4. By step 2, there exists a subsequence $\{\beta^{k_\ell} : \ell \in \mathbb{N}\}$ which converges to $\tilde{\beta} \in B$ and, for all $\beta \in \mathbb{R}^n$ and $\lambda \in \Lambda$, it holds that

$$E_\epsilon(\tilde{\beta}, \lambda(\phi^\epsilon(\tilde{\beta}))) \leq E_\epsilon(\beta, \lambda(\phi^\epsilon(\tilde{\beta}))), \quad E_\epsilon(\tilde{\beta}, \lambda(\phi^\epsilon(\tilde{\beta}))) \leq E_\epsilon(\tilde{\beta}, \lambda). \tag{A.9}$$

Indeed, from step 1 we conclude that there exists $\psi \in \mathbb{R}_{++}$ such that

$$\lim_{k\to\infty} \theta_k = \lim_{k\to\infty} \nu_k = \psi.$$

Under our hypothesis the mapping $\beta \mapsto \lambda(\beta)$ is continuous for $\beta \in (\mathbb{R}\backslash\{0\})^n$, we conclude that

$$\lim_{\ell\to\infty} \lambda^{k_\ell} = \lambda(\phi^\epsilon(\tilde{\beta})).$$

By the definition of the alternating algorithm, we have, for all $\beta \in \mathbb{R}^n$ and $\lambda \in \Lambda$, that

$$\theta_{k+1} = E_\epsilon(\beta^{k+1}, \lambda^k) \leq E_\epsilon(\beta, \lambda^k), \qquad \nu_k = E_\epsilon(\beta^k, \lambda^k) \leq E_\epsilon(\beta^k, \lambda).$$

From this inequality we obtain, passing to limit, inequalities (A.9).

Step 5. The vector $(\tilde{\beta}, \lambda(\phi^\epsilon(\tilde{\beta}))$ is a stationary point. Indeed, since Λ is admissible, by step 3, $\lambda(\phi^\epsilon(\tilde{\beta})) \in \mathrm{int}(\Lambda)$. Therefore, since E_ϵ is continuously differentiable this claim follows from step 4.

Step 6. The alternating algorithm converges. This claim follows from the fact that E_ϵ is strictly convex. Hence, E_ϵ has a unique global minimum in $\mathbb{R}^n \times \Lambda$, which in virtue of inequalities (A.9) is attained at $(\tilde{\beta}, \lambda(\phi^\epsilon(\tilde{\beta})))$.

The last claim in the theorem follows from the fact that the set $\{\gamma(\epsilon) : \epsilon > 0\}$ is bounded and the function $\lambda(\beta)$ is continuous. ■