The Solution Path of SLOPE

Xavier Dupuis, Patrick J C Tardivel

The Solution Path of SLOPE

Xavier Dupuis
Institut de Mathématiques de Bourgogne
UMR 5584 CNRS
Université de Bourgogne
F-21000 Dijon, France
xavier.dupuis@u-bourgogne.fr

Patrick Tardivel
Institut de Mathématiques de Bourgogne
UMR 5584 CNRS
Université de Bourgogne
F-21000 Dijon, France
patrick.tardivel@u-bourgogne.fr

Abstract

The SLOPE estimator has the particularity of having null components (sparsity) and components that are equal in absolute value (clustering). The number of clusters depends on the regularization parameter of the estimator. This parameter can be chosen as a trade-off between interpretability (with a small number of clusters) and accuracy (with a small mean squared error or a small prediction error). Finding such a compromise requires to compute the solution path, that is the function mapping the regularization parameter to the estimator. We provide in this article an algorithm to compute the solution path of SLOPE and show how it can be used to adjust the regularization parameter.

1 Introduction

The SLOPE estimator (Sorted L One Penalized Estimator [5, 38]) is defined as a solution to the following convex program:

$$\min_{b\in\mathbb{R}^p} \frac{1}{2}\|y - Xb\|_2^2 + \gamma \sum_{i=1}^{p} \lambda_i |b|_{\downarrow i} \tag{1}$$

where $\lambda_1 > 0$, $\lambda_1 \geq \cdots \geq \lambda_p \geq 0$ is a given sequence of penalty parameters, $\gamma > 0$ is the regularization parameter and $|b|_{\downarrow 1} \geq \cdots \geq |b|_{\downarrow p} \geq 0$ are the sorted components of b in absolute value. The SLOPE estimator generalizes both the LASSO estimator (Least Absolute Shrinkage and Selection Operator [36]) for which $\lambda_1 = \cdots = \lambda_p = 1$, and the OSCAR estimator (Octagonal Shrinkage and Clustering Algorithm for Regression [7]) for which the sequence $\lambda_1, \ldots, \lambda_p$ is arithmetic. Note that the penalty term of OSCAR satisfies $\sum_{i=1}^{p} \lambda_i |b|_{\downarrow i} = \lambda_p \|b\|_1 + \frac{\lambda_1 - \lambda_2}{2} \sum_{1\leq i<j\leq p} (|b_i + b_j| + |b_i - b_j|)$, thus OSCAR is a particular generalized LASSO [37]; however, in broad generality SLOPE is not a particular generalized LASSO (as proved in supplementary material).

The SLOPE estimator is gaining popularity among statisticians due to its relevant properties such as minimax rates of the estimation and prediction errors [3, 33], false discovery rate control [5] and dimension reduction of the regression model. The latter property comes from the structure of the solutions to the optimization problem (1), which have null components (sparsity) as well as components equal in absolute value (clustering) [28, 14, 7]. In particular, the sparsity and clustering properties of SLOPE are clear when X is an orthogonal matrix since, in this case, the solution to problem (1) is explicit [5, 12, 30, 35]. When y represents the random response of a linear regression model, sparsity has a well-known statistical interpretation: identification of relevant explanatory variables. Clustering also has a statistical interpretation when the design matrix X is standardized: the explanatory variables having the same regression coefficient have the same impact on the response [29]. On the other hand, without restriction on the design matrix, for a categorical variable having different levels, the equal regression coefficients represent levels that can be grouped together [32, 23].

Preprint. Under review.

Therefore, SLOPE estimator can identify relevant explanatory variables, group explanatory variables having the same impact on the response and, more generally, reduce the dimension of the regression model.

The solution path gives the solution of a penalized optimization problem with respect to the regularization parameter $\gamma > 0$. For instance, the solution path of the LASSO shows that the number of explanatory variables selected by this estimator tends to decrease when the regularization parameter becomes large (see *e.g.* [22, 27]) and computing this path is useful to select the regularization parameter. Similarly, the solution path of SLOPE shows that the number of clusters of explanatory variables selected by this estimator tends to decrease when the regularization parameter becomes large. Moreover computing this path is useful to adjust the regularization parameter by minimizing, for instance, the Stein Unbiased Risk Estimate (SURE) formula [31] or the sum of residual squares on a validation set.

The generalized lasso dual path algorithm [37], implemented in the *genlasso* R package [1], allows to compute the solution path of the generalized LASSO and therefore of OSCAR but not of SLOPE in broad generality; moreover it requires $\ker(X) = \{0\}$. Two articles focus on the solution path of OSCAR: the starting point of their respective algorithm is the ordinary least squared estimator (thus requiring $\ker(X) = \{0\}$) in [34], and a numerical solution of OSCAR in [17]. A recent preprint [26] addresses the solution path of SLOPE, under the assumption $\ker(X) = \{0\}$ to guaranty the uniqueness of the solution and to use the ordinary least squares estimator as a starting point; it gives no theoretical results on the solution path (such as its continuity, the proof that it is piecewise linear, the characterization of its affine components).

In this article, for sequences of penalty parameters $\lambda_1 > \cdots > \lambda_p > 0$, we prove that the solution path of SLOPE is continuous and piecewise linear on $(0, +\infty)$, we characterize its affine components, and we provide an algorithm to compute the exact solution path of SLOPE. Our algorithm does not require neither $\ker(X) = \{0\}$ nor to solve SLOPE with an external solver. We dedicate a section to numerical experiments on real data sets to illustrate: the computation of SLOPE solution paths; the exact minimization of SURE for SLOPE (pointing out differences with the LASSO estimator); the performance of our algorithm compared to *genlasso* to compute the OSCAR solution path; the performance of our algorithm compared to the algorithms considered and implemented in [21] to compute the SLOPE solution for a single regularization parameter γ.

2 Basic notions on SLOPE

Unlike the ℓ_1 norm, in broad generality the sorted ℓ_1 norm is not separable (the sorted ℓ_1 norm cannot be written as a sum of functions of its components). As a result, it is much more challenging to study the SLOPE optimization problem than the LASSO optimization problem. For instance the gradient $X'(y - X\widehat{\beta}^{\text{lasso}})$ of the sum of residual squares at the LASSO solution $\widehat{\beta}^{\text{lasso}}$ gives indications on null components of this estimator. Indeed, $|X_i'(y - X\widehat{\beta}^{\text{lasso}}(\gamma))| < \gamma$ implies $\widehat{\beta}_i^{\text{lasso}}(\gamma) = 0$. Unfortunately, because the sorted ℓ_1 norm is not separable, determining null components based on the the gradient $X'(y - X\widehat{\beta})$ of the sum of residual squares at the SLOPE solution $\widehat{\beta}$ is not straightforward (determining non-null clusters is also difficult). The important notions introduced hereafter allow to overcome this difficulty.

2.1 Sorted ℓ_1 norm and its dual norm

Definition 1 *The* sorted ℓ_1 norm *associated to* $\lambda \in \mathbb{R}^p$ *with* $\lambda_1 \geq \cdots \geq \lambda_p \geq 0$ *and* $\lambda_1 > 0$ *is defined as follows:*

$$J_\lambda(b) = \sum_{i=1}^{p} \lambda_i |b|_{\downarrow i}, \quad b \in \mathbb{R}^p,$$

where $|b|_{\downarrow 1} \geq \ldots \geq |b|_{\downarrow p}$ *are the sorted components of* b *with respect to the absolute value.*

Given a norm $\|\cdot\|$ on $\mathbb{R}^p$, we recall that its dual norm $\|\cdot\|^*$ is defined by $\|v\|^* = \max\{b'v : \|b\| \leq 1\}$, for $v \in \mathbb{R}^p$.

Remark 1 *The* dual sorted ℓ_1 norm *has an explicit expression given in [25] and reminded hereafter:*

$$J_\lambda^*(v) = \max\left\{\frac{\|v\|_{(1)}}{\lambda_1}, \frac{\|v\|_{(2)}}{\sum_{i=1}^2 \lambda_i}, \ldots, \frac{\|v\|_{(p)}}{\sum_{i=1}^p \lambda_i}\right\}, v \in \mathbb{R}^p,$$

where $\|\cdot\|_{(k)}$ *is the* $k-$*norm (the sum of the k largest components in absolute value).*

2.2 SLOPE pattern

The SLOPE pattern introduced in [28], whose definition is reminded below, is a central notion in this article.

Definition 2 *The* SLOPE pattern $\mathrm{patt}(b) \in \mathbb{Z}^p$ *of* $b \in \mathbb{R}^p$ *is defined by*

$$\mathrm{patt}(b)_i = \mathrm{sign}(b_i)\,\mathrm{rank}(|b|)_i, \quad i \in \{1, \ldots, p\},$$

where $\mathrm{rank}(|b|)_i \in \{0, 1, \ldots, k\}$*, k is the number of nonzero distinct values in* $\{|b_1|, \ldots, |b_p|\}$*,* $\mathrm{rank}(|b|)_i = 0$ *if and only if* $b_i = 0$*, and* $\mathrm{rank}(|b|)_i < \mathrm{rank}(|b|)_j$ *if* $|b_i| < |b_j|$*.*

We denote by $\mathcal{P}_p^{\mathrm{slope}} = \mathrm{patt}(\mathbb{R}^p)$ the set of SLOPE patterns. Note in the definition above that $k = \|\mathrm{patt}(b)\|_\infty$ is the number of nonzero clusters of b.

Example 1 *Let* $b = (4.2, -1.3, 0, 1.3, 4.2)'$*. Then* $\mathrm{patt}(b) = (2, -1, 0, 1, 2)'$*.*

Definition 3 *Let* $m \in \mathbb{Z}^p$ *be a SLOPE pattern with* $k = \|m\|_\infty \geq 1$*. The associated* pattern matrix $U_m \in \mathbb{R}^{p\times k}$ *is defined by*

$$(U_m)_{ij} = \mathrm{sign}(m_i)\mathbf{1}_{(|m_i|=k+1-j)} \quad i \in \{1, \ldots, p\},\ j \in \{1, \ldots, k\}.$$

For $k \geq 1$ we denote $\mathbb{R}^{k+} = \{s \in \mathbb{R}^k \colon s_1 > \ldots > s_k > 0\}$. Definition 3 is such that, for $b \in \mathbb{R}^p$ and $m \in \mathbb{Z}^p$ a SLOPE pattern with $k = \|m\|_\infty \geq 1$, we have

$$\mathrm{patt}(b) = m \iff \exists s \in \mathbb{R}^{k+} \text{ such that } b = U_m s.$$

Hereafter, the notation $|m|_\downarrow = (|m|_{\downarrow 1}, \ldots, |m|_{\downarrow p})'$ represents the components of m sorted non-increasingly with respect to the absolute value.

Example 2 *Let* $m = (2, -1, 0, 1, 2)'$*. Then*

$$U_m = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 1 & 0 \end{pmatrix}' \text{ and } U_{|m|_\downarrow} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}'.$$

Definition 4 *Let* $m \in \mathbb{Z}^p$ *be a SLOPE pattern with* $k = \|m\|_\infty \geq 1$*. The* clustered matrix $\tilde{X}_m \in \mathbb{R}^{n\times k}$ *of* $X \in \mathbb{R}^{n\times p}$ *is defined by* $\tilde{X}_m = XU_m$*; the* clustered parameter $\tilde{\lambda}_m \in \mathbb{R}^k$ *of* $\lambda \in \mathbb{R}^p$ *is defined by* $\tilde{\lambda}_m = U'_{|m|_\downarrow}\lambda$*.*

Note that the dimension of the design matrix X is reduced when it is clustered as $\tilde{X}_m$ by a pattern m: a null component $m_i = 0$ leads to discard the column X_i from the design matrix X, and a cluster $K \subset \{1, \ldots, p\}$ of m (set of components of m equal in absolute value) leads to replace the columns $(X_i)_{i\in K}$ by one column equal to the signed sum: $\sum_{i\in K} \mathrm{sign}(m_i)X_i$.

Example 3 *Let* $X = (X_1|X_2|X_3|X_4|X_5)$*,* $m = (2, -1, 0, 1, 2)'$*,* $\lambda = (\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5)' \in \mathbb{R}^5$*. Then the clustered matrix and the clustered parameter are given by:*

$$\tilde{X}_m = (X_1 + X_5| - X_2 + X_4) \text{ and } \tilde{\lambda}_m = \begin{pmatrix} \lambda_1 + \lambda_2 \\ \lambda_3 + \lambda_4 \end{pmatrix}.$$

2.3 Subdifferential of the sorted ℓ_1 norm

The subdifferential of a norm is related to the dual norm $\|\cdot\|^*$ via the following formula [18, p. 180]:

$$\partial\|\cdot\|(b) = \{v \in \mathbb{R}^p : \|v\|^* \leq 1 \text{ and } b'v = \|b\|\}, \quad b \in \mathbb{R}^p.$$

In particular, $\partial\|\cdot\|(b)$ is a face of the dual unit ball. For the sorted ℓ_1 norm, the above formula can be specified further with the pattern matrix and the clustered parameter associtated to $m = \mathrm{patt}(b)$ for $b \neq 0$ [6, 28]:

$$\partial J_\lambda(b) = \left\{v \in \mathbb{R}^p : J_\lambda^*(v) \leq 1 \text{ and } U_m' v = \tilde{\lambda}_m\right\}. \tag{2}$$

Remark 2 *Given $\lambda \in \mathbb{R}^{p+}$, the mapping $m \mapsto \partial J_\lambda(m)$ is a bijection between the set of SLOPE patterns and the set of faces of the unit ball of J_λ^* (the signed permutahedron) [28, Theorem 6]. It is no longer true when $\lambda_1 \geq \cdots \geq \lambda_p \geq 0$ is not a decreasing sequence. Therefore we restrict our study to the case where $\lambda \in \mathbb{R}^{p+}$, i.e. $\lambda_1 > \cdots > \lambda_p > 0$.*

3 Solution, fitted value and gradient paths

3.1 Solution set and fitted value

Given $X \in \mathbb{R}^{n\times p}$, $y \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{p+}$, and $\gamma > 0$, we denote by $\mathcal{S}_{X,y,\lambda}(\gamma)$ (or simply $\mathcal{S}(\gamma)$ when there is no ambiguity) the set of solutions to the SLOPE optimization problem (1), namely:

$$\min_{b\in\mathbb{R}^p} \frac{1}{2}\|y - Xb\|_2^2 + \gamma J_\lambda(b).$$

For any $\gamma > 0$, the objective function of the above problem is continuous and coercive thus the *solution set* $\mathcal{S}(\gamma)$ is nonempty. Moreover, the *fitted value* $\widehat{\mathrm{fit}}(\gamma) = X\widehat{\beta}$ does not depend on $\widehat{\beta} \in \mathcal{S}(\gamma)$. When $\mathcal{S}(\gamma)$ is a singleton, we denote by $\widehat{\beta}(\gamma)$ its unique element. Note that uniqueness is rather a weak assumption, indeed the set

$$\{X \in \mathbb{R}^{n\times p} : \exists y \in \mathbb{R}^n \ \exists \gamma > 0 \text{ such that } \mathcal{S}_{X,y,\lambda}(\gamma) \text{ is not a singleton}\}$$

has zero Lebesgue measure [28, Proposition 3]. Theorem 1 below shows that $\widehat{\mathrm{fit}}(\cdot)$ and $\widehat{\beta}(\cdot)$ are continuous on $(0, +\infty)$ and affine between two regularization parameters for which SLOPE solutions have the same pattern. Affine expressions of these piecewise linear functions are explicit and intervals are caracterized. We denote hereafter by A^+ the Moore-Penrose pseudo-inverse of a matrix A.

Theorem 1 *Let $X \in \mathbb{R}^{n\times p}$, $y \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{p+}$, and $m \in \mathbb{Z}^p$ be a non-null SLOPE pattern with $k = \|m\|_\infty \geq 1$.*

1. *The set $I_m = \{\gamma > 0 : \exists \widehat{\beta} \in \mathcal{S}(\gamma)$ such that $\mathrm{patt}(\widehat{\beta}) = m\}$ is an interval, with the following characterization:*

$$\gamma \in I_m$$
$$\Updownarrow$$
$$\begin{cases} \exists s \in \mathbb{R}^{k+} \text{ such that } \tilde{X}_m' y - \gamma\tilde{\lambda}_m = \tilde{X}_m'\tilde{X}_m s & \text{(positivity condition)} \\ X'(\tilde{X}_m')^+\tilde{\lambda}_m + \frac{1}{\gamma}X'(I_n - (\tilde{X}_m')^+\tilde{X}_m')y \in \partial J_\lambda(m) & \text{(subdifferential condition)} \end{cases}$$

   Moreover, $\widehat{\beta} = U_m s \in \mathcal{S}(\gamma)$ and $\mathrm{patt}(\widehat{\beta}) = m$ for any $s \in \mathbb{R}^{k+}$ satisfying the positivity condition at $\gamma \in I_m$.

2. *The* fitted value path *$\gamma \mapsto \widehat{\mathrm{fit}}(\gamma)$ is continuous and piecewise linear on $(0, +\infty)$, with the following affine expression on I_m:*

$$\widehat{\mathrm{fit}}(\gamma) = (\tilde{X}_m')^+\tilde{X}_m' y - \gamma(\tilde{X}_m')^+\tilde{\lambda}_m, \quad \gamma \in I_m.$$

3. *If $\mathcal{S}(\gamma) = \{\widehat{\beta}(\gamma)\}$ for all $\gamma > 0$, then the* solution path *$\gamma \mapsto \widehat{\beta}(\gamma)$ is continuous and piecewise linear on $(0, +\infty)$, with the following affine expression on I_m:*

$$\widehat{\beta}(\gamma) = U_m(\tilde{X}_m'\tilde{X}_m)^{-1}(\tilde{X}_m' y - \gamma\tilde{\lambda}_m), \quad \gamma \in I_m.$$

The characterization of the interval I_m above is closely related to Theorem 3.1 in [6].

3.2 Gradient path and clusters

A solution of the SLOPE optimization problem is characterized by the following two conditions

$$\widehat{\beta} \in \mathcal{S}(\gamma) \Leftrightarrow \begin{cases} J_\lambda^*(X'(y - X\widehat{\beta})) \leq \gamma \\ \widehat{\beta}' X'(y - X\widehat{\beta}) = \gamma J_\lambda(\widehat{\beta}) \end{cases}$$

Note that $X'(y - X\widehat{\beta}) = X'(y - \widehat{\mathrm{fit}}(\gamma))$ is the gradient at $\widehat{\beta}$ of the sum of residual squares $b \mapsto \frac{1}{2}\|y - Xb\|_2^2$. Subsequently, we call gradient path the expression $\gamma > 0 \mapsto X'(y - \widehat{\mathrm{fit}}(\gamma))$. The set of inequalities describing the ball of radius γ for the dual sorted ℓ_1 norm which are saturated by the gradient is :

$$\mathcal{A}(\gamma) = \left\{ i \in \{1, \dots, p\} : \frac{\|X'(y - \widehat{\mathrm{fit}}(\gamma))\|_{(i)}}{\sum_{j=1}^{i} \lambda_j} = \gamma \right\}.$$

According to Theorem 2 below, the set $\mathcal{A}(\gamma)$ provides the number of non-zero clusters, the size of these clusters as well as the number of non-zero components.

Theorem 2 *Let $\lambda \in \mathbb{R}^{p+}$, $X \in \mathbb{R}^{n \times p}$, $y \in \mathbb{R}^n$, $\gamma > 0$ and $\widehat{\beta} \in \mathcal{S}(\gamma)$.*

1. *Let $1 \leq k_1 \leq \cdots \leq k_l \leq p$ be a subdivision such that:*

   $$|\mathrm{supp}(\widehat{\beta})| = k_l \text{ and } |\widehat{\beta}|_{\downarrow 1} = \cdots = |\widehat{\beta}|_{\downarrow k_1} > \cdots > |\widehat{\beta}|_{\downarrow k_{l-1}+1} = \cdots = |\widehat{\beta}|_{\downarrow k_l} > 0$$

   (i.e. $\widehat{\beta}$ has l non-null clusters, the cluster of the largest value has k_1 elements and so on and $\widehat{\beta}$ has k_l non-null components). Then, $\{k_1, \dots, k_l\} \subset \mathcal{A}(\gamma)$.

2. *Conversely, if $\{k_1, \dots, k_l\} = \mathcal{A}(\gamma)$ then*

   $$|\widehat{\beta}|_{\downarrow 1} = \cdots = |\widehat{\beta}|_{\downarrow k_1} \geq \cdots \geq |\widehat{\beta}|_{\downarrow k_{l-1}+1} = \cdots = |\widehat{\beta}|_{\downarrow k_l} \geq |\widehat{\beta}|_{\downarrow k_l+1} = \cdots = |\widehat{\beta}|_{\downarrow p} = 0$$

   (i.e. the number of non-null clusters of $\widehat{\beta}$ is smaller or equal to l and the number of non-null components is smaller or equal to k_l).

There are links between Theorem 2 and screening rules for SLOPE [13, 20] which identify some null components of this estimator. For instance, running Algorithm 1 in [20] with $|X'(y - \widehat{\mathrm{fit}}(\gamma)|_\downarrow$ returns that a SLOPE solution has at most $\max\{\mathcal{A}(\gamma)\}$ non-zero components. Otherwise, Theorem 4.1 in [13] is closely related to the following implication: $|\widehat{\beta}|_{\downarrow i} \neq 0 \Rightarrow \exists k \geq i, k \in \mathcal{A}(\gamma)$.

4 Algorithms to compute the solution path

To keep this section simple we assume that $\mathcal{S}(\gamma) = \{\widehat{\beta}(\gamma)\}$ for all $\gamma > 0$. Let $J_\lambda^*(X'y) = \gamma_0 > \gamma_1 > \dots \gamma_r > \gamma_{r+1} = 0$ be a subdivision such that $\gamma \mapsto \widehat{\beta}(\gamma)$ is affine with pattern $m^{(i)}$ on the interval (γ_{i+1}, γ_i) for $i = 0, \dots, r$ (*i.e* the interior of $I_{m^{(i)}}$ is (γ_{i+1}, γ_i)).

First, let us explain how to compute the SLOPE solution path on $[\gamma_1, \gamma_0]$. By construction of $m^{(0)}$ the following implication holds

$$\forall \gamma \in (\gamma_1, \gamma_0) \quad \mathrm{patt}(\widehat{\beta}(\gamma)) = m^{(0)} \Rightarrow \frac{1}{\gamma} X'(y - \widehat{\mathrm{fit}}(\gamma)) \in \partial J_\lambda(m^{(0)}).$$

Moreover, since $\gamma > 0 \mapsto \widehat{\mathrm{fit}}(\gamma)$ is continuous, $\mathrm{fit}(\gamma_0) = 0$ and $\partial J_\lambda(m^{(0)})$ is a closed set, we get

$$\frac{1}{\gamma_0} X'(y - \widehat{\mathrm{fit}}(\gamma_0)) = \frac{1}{\gamma_0} X'y \in \partial J_\lambda(m^{(0)}). \tag{3}$$

Algorithm 1 provides the pattern $M(\frac{1}{\gamma_0} X'y)$ of the smallest face of the signed permutahedron containing $\frac{1}{\gamma_0} X'y$. Therefore, by construction

$$\partial J_\lambda\left(M\left(\frac{1}{\gamma_0} X'y\right)\right) \subset \partial J_\lambda(m^{(0)}) \Rightarrow \left\|M\left(\frac{1}{\gamma_0} X'y\right)\right\|_\infty \leq \|m^{(0)}\|_\infty$$

According to (4), if $\frac{X'y}{\gamma_0}$ lies onto a facet of the signed permutahedron, we have $m^{(0)} = M\left(\frac{1}{\gamma_0} X'y\right)$.

Algorithm 1 Pattern of the smallest face containing a vector

Require: $\lambda \in \mathbb{R}^{p+}$ and $z \in \mathbb{R}^p$ such that $J^*_\lambda(z) \leq 1$

Define the set of saturated inequalities as follows:

$$\mathcal{A}(z) = \left\{ i \in \{1, \dots, p\} : \frac{\|z\|_{(i)}}{\sum_{j=1}^{i} \lambda_j} = 1 \right\}.$$

If $\mathcal{A}(z) = \emptyset$ set $M(z) = (0, \dots, 0) \in \mathbb{R}^p$, otherwise define $M(z) \in \mathcal{P}_p^{\text{slope}}$ as follows:

$$\forall j \in \{1, \dots, p\}\ M_j(z) = \text{sign}(z_j) \sum_{i \in \mathcal{A}(z)} \mathbf{1}(|z_j| \geq \lambda_i).$$

return $M(z)$

Example 4 *We illustrate of the solution path of OSCAR for* $y = (15, 5)' \in \mathbb{R}^2$, $\lambda = (6, 4, 2)' \in \mathbb{R}^{3+}$ *and*

$$X = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \end{pmatrix}.$$

Largest node γ_0: *We have* $X'y = (35, 25, 5)'$*, therefore* $\gamma_0 = J^*_\lambda(X'y) = 6$.

Pattern $m^{(0)}$ in the left neighborhood of γ_0: *Since* $\frac{1}{\gamma_0} X'y = (35/6, 25/6, 0)'$ *lies in the relative interior of* $\partial J_\lambda(1, 1, 0)' = [(6, 2)', (4, 6)'] \times [-2, 2]$ *then* $m^{(0)} = M(\frac{1}{\gamma_0} X'y) = (1, 1, 0)'$.

Expression of $\widehat{\beta}(\gamma)$ in the left neighborhod of γ_0: *According to statement 3 in Theorem 1 when* $\gamma < \gamma_0 = 6$ *is sufficiently close to* γ_0 *we have* $\widehat{\beta}(\gamma) = (\frac{30-5\gamma}{9}, \frac{30-5\gamma}{9}, 0)'$.

We tried the package genlasso *to compute this solution path. Since* $\dim(\ker(X)) \neq 0$*,* genlasso *add a small ridge term* $\epsilon \|b\|_2^2$ *to the objective function (the default value is* $\epsilon = 10^{-4}$*); thus* genlasso *solves*

$$\begin{aligned} & \min_{b \in \mathbb{R}^3} \frac{1}{2} \|y - Xb\|_2^2 + \gamma J_\lambda(b) + \epsilon \|b\|_2^2, \\ = & \min_{b \in \mathbb{R}^3} \frac{1}{2} \left\| \begin{pmatrix} y \\ 0 \end{pmatrix} - \begin{pmatrix} X \\ \sqrt{2\epsilon} I_3 \end{pmatrix} b \right\|_2^2 + \gamma J_\lambda(b). \end{aligned} \tag{4}$$

We computed the solution path of problem (4) *when* $\epsilon = 10^{-4}$ *with our algorithm and* genlasso*. Surprisingly, the solution path computed with* genlasso *is correct when* $\gamma \geq 5$ *but wrong when* $\gamma < 5$*. Comparatively to the original problem (without adding the ridge term), when* $\epsilon = 10^{-4}$ *the solution path have more nodes (especially small nodes). Moreover these paths are extremely different when* γ *is small since* $\epsilon \|b\|_2^2$ *dominates* $\gamma J_\lambda(b)$.

Algorithm 2 uses the characterisation of $I_{m^{(0)}}$, based on the positivity and subdifferential conditions, to provide both the node γ_1 as well as the pattern $m^{(1)}$.

Using iteratively Algorithm 2 allows to compute entirely the SLOPE solution path.

5 Numerical experiments

The code of the implementation in Python of our algorithm and of the experiments below is available at `https://github.com/x-dupuis/slope-path`. The computations were carried out on an Apple M1 Pro chip (8-core CPU and 14-core GPU) and 16GB of unified memory.

We use two real data sets:

- the *Wine Quality* data set[1] describes the quality of red "Vinho Verde" wines [10]. Each column of $X \in \mathbb{R}^{1599 \times 11}$ represents a physicochemical measurement (density, pH, alcohol, etc.) and $y \in \mathbb{R}^{1599}$ represents wine quality scores (between 0 and 10);

[1] available at `https://archive.ics.uci.edu/dataset/186/wine+quality`

Algorithm 2 Next node and next pattern

Require: $X \in \mathbb{R}^{n\times p}, \lambda \in \mathbb{R}^{p+}, \gamma_i, m^{(i)}$ and $s(\gamma) = (\tilde{X}'_{m^{(i)}}\tilde{X}_{m^{(i)}})^{-1}(\tilde{X}'_{m^{(i)}}y - \gamma\tilde{\lambda}_{m^{(i)}})$

Let $k = \|m^{(i)}\|_\infty$ and set $\gamma_{\text{fuse}} = 0$ if $s(\gamma) \in \mathbb{R}^{k+}$ for all $\gamma \in [0, \gamma_i)$, otherwise compute

$$\gamma_{\text{fuse}} = \sup\{\gamma \in [0, \gamma_i) : s(\gamma) \notin \mathbb{R}^{k+}\}.$$

if $\gamma_{\text{fuse}} = 0$ **then**

set $\gamma_{\text{split}} = 0$ if $X'(y - \tilde{X}_{m^{(i)}}s(\gamma)) \in \gamma\partial J_\lambda(m^{(i)})$ for all $\gamma \in [0, \gamma_i)$, otherwise compute

$$\gamma_{\text{split}} = \sup\{\gamma \in [0, \gamma_i) : X'(y - \tilde{X}_{m^{(i)}}s(\gamma)) \notin \gamma\partial J_\lambda(m^{(i)})\}.$$

if $\gamma_{\text{slpit}} = 0$ **then**

return The solution path is entirely computed.

else

$\gamma_{i+1} = \gamma_{\text{split}}$

Compute $m^{(i+1)} = M(\frac{1}{\gamma_{i+1}}X'(y - \tilde{X}_{m^{(i)}}s(\gamma_{i+1})))$ with Algorithm 1.

return $\gamma_{i+1}, m^{(i+1)}$

end if

else if $\gamma_{\text{fuse}} > 0$ and $X'(y - \tilde{X}_{m^{(i)}}s(\gamma_{\text{fuse}})) \in \gamma_{\text{fuse}}\partial J_\lambda(m^{(i)})$ **then**

$\gamma_{i+1} = \gamma_{\text{fuse}}$

$m^{(i+1)} = \text{patt}(U_{m^{(i)}}s(\gamma_{\text{fuse}}))$

return $\gamma_{i+1}, m^{(i+1)}$

else

Compute

$$\gamma_{i+1} = \sup\{\gamma \in [\gamma_{\text{fuse}}, \gamma_i) : X'(y - \tilde{X}_{m^{(i)}}s(\gamma)) \notin \gamma\partial J_\lambda(m^{(i)})\}.$$

Compute $m^{(i+1)} = M(\frac{1}{\gamma_{i+1}}X'(y - \tilde{X}_{m^{(i)}}s(\gamma_{i+1})))$ with Algorithm 1.

return $\gamma_{i+1}, m^{(i+1)}$

end if

- the *Riboflavin* data set[2] describes the riboflavin production with *Bacillus subtilis* [9]. Each column of $X \in \mathbb{R}^{71\times 4088}$ represents a gene expression measurement and $y \in \mathbb{R}^{71}$ represents production rates.

The matrices X are mean-centered ($\forall j, \sum_i X_{ij} = 0$) and standardized ($\forall j, \sum_i X_{ij}^2 = n$), the vectors y are mean-centered ($\sum_i y_i = 0$).

5.1 Full paths computation

We illustrate here the computation of SLOPE solution paths on the *Wine Quality* data set. For this numerical experiment we take $\lambda = (1, \sqrt{2}-1, \sqrt{3}-\sqrt{2}, \ldots, \sqrt{11}-\sqrt{10})$, so that the unit ball of the sorted ℓ_1 norm is quasi-spherical [26]. Figure 1 provides the solution path of SLOPE as well as the solution path of LASSO (computed via the homotopy algorithm in [22]).

5.2 Exact minimization of SURE

The Stein Unbiaised Risk Estimate (SURE) formula is an unbiased estimator of the prediction error ($\mathbb{E}(\|X\widehat{\beta} - X\beta\|_2^2)$ where $\widehat{\beta}$ is an estimator of β). For LASSO and SLOPE, unbiased estimators for the prediction error are reported hereafter [24, 40]

$$\text{sure}(\gamma) = \begin{cases} \|y - X\widehat{\beta}(\gamma)\|_2^2 - n\sigma^2 + 2\sigma^2\|\text{patt}(\widehat{\beta}(\gamma))\|_\infty & \text{when } \widehat{\beta}(\gamma) \text{ is a SLOPE estimator} \\ \|y - X\widehat{\beta}(\gamma)\|_2^2 - n\sigma^2 + 2\sigma^2|\text{supp}(\widehat{\beta}(\gamma))| & \text{when } \widehat{\beta}(\gamma) \text{ is a LASSO estimator} \end{cases}$$

[2] available at `https://www.annualreviews.org/doi/suppl/10.1146/annurev-statistics-022513-115545`

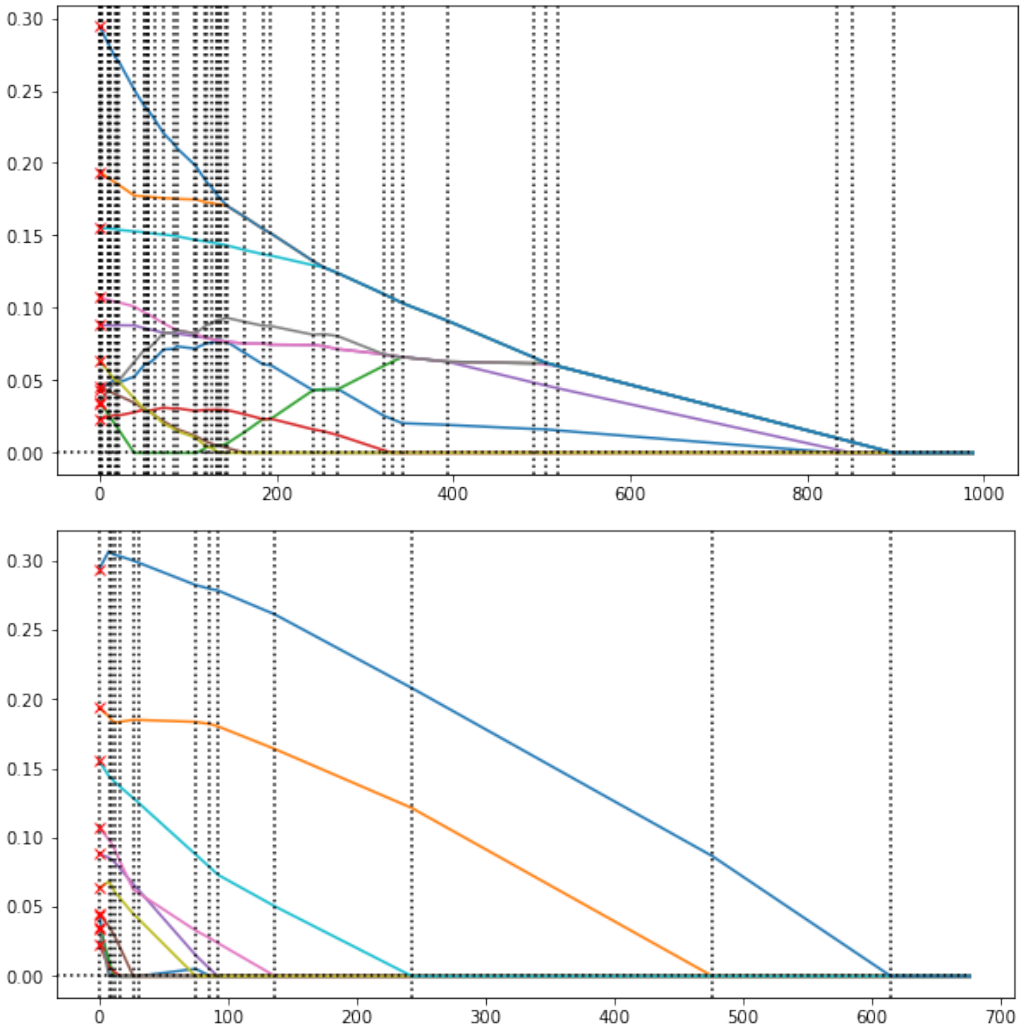

Figure 1: Solution paths in absolute value of SLOPE (top) and LASSO (bottom) as functions of $\gamma > 0$. On top some curves partially superimpose or partially coincide with the x-axis, illustrating the clustering and sparsity properties of SLOPE. At the bottom some curves just partially coincide with the x-axis, illustrating the sparsity property of LASSO.

where σ^2 is the variance of residuals. A usual way to select the regularization parameter γ is to minimize $\mathrm{sure}(\gamma)$ [11, 4]. For both SLOPE and LASSO the solution path is piecewise linear, therefore the SURE formula is quadratic between two adjacents nodes (*i.e* the SURE formula restricted to the interval (γ_{i+1}, γ_i) is quadratic). As a result, solving exactly the solution path allows to minimize exactly the SURE formula[3]. For this numerical experiment we substitute σ^2 in the expression of $\mathrm{sure}(\gamma)$ by $\widehat{\sigma^2} = \|(I_n - X(X'X)^{-1}X')y\|/1588 = 0.4197$. Note that when γ is very large the SURE formula satisfies $\mathrm{sure}(\gamma) = \|y\|_2^2 - 1599\widehat{\sigma^2} = 371.1382$. Moreover when γ tends to 0, both SLOPE and LASSO converge to the ordinary least squares estimator therefore $\lim_{\gamma \to 0} \mathrm{sure}(\gamma) = 11\widehat{\sigma^2} = 4.6162$. We report the regularization parameter minimizing the SURE formula in Table 1.

The explanatory variables "fixed acidity" (corresponding to X_1) and "pH" (corresponding to X_9) are the most correlated ones (the largest off-diagonal components of $X'X$, in absolute value,

[3]One may similarly minimize exactly the sum of residual squares on a validation set $\gamma > 0 \mapsto \|y^{\mathrm{val}} - X^{\mathrm{val}}\widehat{\beta}(\gamma)\|_2^2$.

	γ_{sure}	$\text{sure}(\gamma_{\text{sure}})$
SLOPE	18.6292	3.4641
LASSO	11.7602	4.1297

Table 1: Minimizer and minimum of the SURE formula for both SLOPE and LASSO. The minimum is lower for SLOPE than LASSO, suggesting that SLOPE is a slightly better estimator for the prediction error than LASSO.

is $|X_1' X_9| = 1092.0821$). The explanatory variables "fixed acidity" and "density" (corresponding to X_8) are also strongly correlated ($|X_1' X_8| = 1068.2076$). These three variables are clustered by the SLOPE estimator $\widehat{\beta}(\gamma_{\text{sure}})$ (corresponding to the cluster "4" in $\text{patt}(\widehat{\beta}(\gamma_{\text{sure}})) = (4, -8, -1, 2, -5, 3, -6, -4, -4, 7, 9)'$) whereas the LASSO estimator $\widehat{\beta}^{\text{lasso}}(\gamma_{\text{sure}})$ only selects one: the "pH" (actually $\widehat{\beta}_1^{\text{lasso}}(\gamma_{\text{sure}})) = \widehat{\beta}_8^{\text{lasso}}(\gamma_{\text{sure}})) = 0$). Clustering property of SLOPE for highly correlated variables had been discussed in [14] and intuitively we believe that this property is beneficial for the prediction error.

5.3 Full path solvers benchmark

For this benchmark we focus on the solution path of OSCAR as, in the literature, no algorithm for solving the solution path of SLOPE is available online (the code for solving the solution path of SLOPE in the preprint [26] is not available). A natural competitor to our algorithm is *genlasso*. Hereafter X and y are provided by the *Wine Quality* data set and λ is an arithmetic progression where $\lambda_1 = 4$ and $\lambda_{11} = 1$. In table 2 we compare the time needed to compute the solution path as well as the value of the objective function of OSCAR at $\gamma \in \left\{ \frac{J_\lambda^*(X'y)}{2}, \frac{J_\lambda^*(X'y)}{10} \right\}$.

	genlasso	slope path (our)
Time	4.96e-01	**1.31e-02**
Value at $\frac{J_\lambda^*(X'y)}{2}$	483.4367	483.4367
Value at $\frac{J_\lambda^*(X'y)}{10}$	379.8561	**378.5511**

Table 2: Time in seconds to compute the solution path and value of the objective function. Our algorithm is much faster than *genlasso*. Moreover, the value obtained with our algorithm is lower than the one obtained with *genlasso* at $\gamma = \frac{J_\lambda^*(X'y)}{10}$, illustrating that the solution provided by *genlasso* is not accurate.

Comparison between *genlasso* and our algorithm on the Riboflavin data set is not tractable; indeed the D matrix such that $J_\lambda(b) = \|Db\|_1$ belongs to $\mathbb{R}^{4088^2 \times 4088}$ and even if it is sparse, the package *genlasso* cannot handle such a big matrix.

5.4 SLOPE solvers benchmark

Computing the full solution path of SLOPE on $(0, +\infty)$ is a more ambitious task than solving the SLOPE optimization problem for a single regularization parameter γ. Therefore, given such a γ, we can compute the solution path on $[\gamma, +\infty)$ and thus define a SLOPE solver (called *slope path* hereafter). We compare it to the following algorithms implemented in the extensive benchmark of SLOPE solvers [21]:

- Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [2];
- Anderson acceleration for Proximal Gradient Descent (Anderson PGD) [39];
- Alternating Direction Method of Multipliers (ADMM) [8] with the augmented Lagrangian parameter $\rho = 100$;
- Coordinate Descent for SLOPE (hybrid CD) [21].

We used their code[4] and set as stopping criterion a primal-dual gap smaller than 1e-12 (which is satisfied by our algorithm all along the path). When λ is an arithmetic progression where $\lambda_1 = 4$ and $\lambda_p = 1$ the benchmarks on the two real data sets are reported in tables 3 and 4.

γ	$\frac{J^*_\lambda(X'y)}{2}$	$\frac{J^*_\lambda(X'y)}{10}$
FISTA	1.36e-02	4.47e-02
Anderson PGD	5.26e-03	7.02e-02
ADMM ($\rho = 100$)	2.38e-02	7.18e-03
hybrid CD	2.39e-03	7.91e-03
slope path (our)	**6.58e-04**	**4.51e-03**

Table 3: Time in seconds to compute the solution for the *Wine Quality* data set. In this case, where $p = 11$ is small, our algorithm is the fastest one.

γ	$\frac{J^*_\lambda(X'y)}{2}$	$\frac{J^*_\lambda(X'y)}{10}$
FISTA	9.01e+01	-
Anderson PGD	1.45e+01	-
ADMM ($\rho = 100$)	-	4.93e+00
hybrid CD	4.19e-02	**8.84e-01**
slope path (our)	**3.83e-02**	3.72e+00

Table 4: Time in seconds to compute the solution for the *Riboflavin* data set. In this case where $p = 4088$ is large, our algorithm is still the fastest one when γ is large ($\gamma = \frac{J^*_\lambda(X'y)}{2}$) but is over-performed by hybrid CD when γ is small ($\gamma = \frac{J^*_\lambda(X'y)}{10}$). The missing values correspond to algorithms not reaching the required primal-dual gap (1e-12).

6 Conclusion and future works

One of the main result in this article is Theorem 1 proving that the SLOPE solution path is piecewise linear and providing the characterization of the intervals where the path is affine. Moreover algorithms 1 and 2 allow to solve exactly this path. The computational time of our numerical scheme depends mainly on the number of nodes. In our illustration on real data sets, the number of intervals is not too large. However the number of intervals where the path is affine is bounded by the number of SLOPE patterns in $\mathbb{R}^p$ and potentially, similarly as for LASSO [22], this huge upper bound might be reached. Therefore solving the solution path of SLOPE on $(0, +\infty)$ might be intractable for some pathological examples and, in such a situation, our algorithm can only compute partially the solution path. A first algorithmic perspective would be to generalize this method to a wide class of penalized estimators. Indeed, the crucial notion of SLOPE pattern might be generalized to a polyhedral gauge penalty [16] (the SLOPE pattern is just the pattern associated to the sorted ℓ_1 (polyhedral) norm). Another methodological perspective is to derive, based on Theorem 2, screening rules identifying null components and clusters for SLOPE.

Acknowledgments

The Institut de Mathématiques de Bourgogne (IMB) receives support from the EIPHI Graduate School (contract ANR-17-EURE-0002). Patrick Tardivel receives support from the region Bourgogne-Franche-Comté (EPADM project).

[4] available at `https://github.com/jolars/slopecd`

7 Appendix

Proof of Theorem 1

1: I_m is an interval) Hereafter we suppose that $I_m \neq \emptyset$. Let $\gamma_0, \gamma_1 \in I_m$ and pick $\widehat{\beta}(\gamma_0) \in \mathcal{S}(\gamma_0)$, $\widehat{\beta}(\gamma_1) \in \mathcal{S}(\gamma_1)$ such that $\mathrm{patt}(\widehat{\beta}(\gamma_0)) = \mathrm{patt}(\widehat{\beta}(\gamma_1)) = m$. Let $\alpha \in [0,1]$, $\bar{\gamma} = \alpha\gamma_0 + (1-\alpha)\gamma_1$ and $\bar{\beta} = \alpha\widehat{\beta}(\gamma_0) + (1-\alpha)\widehat{\beta}(\gamma_1)$ then $\mathrm{patt}(\bar{\beta}) = m$. Indeed, if $m = 0$ then clearly $\mathrm{patt}(\bar{\beta}) = 0$. Otherwise, let $k = \|m\|_\infty \geq 1$ then $\widehat{\beta}(\gamma_0) = U_m s_0$ for some $s_0 \in \mathbb{R}^{k+}$, $\widehat{\beta}(\gamma_1) = U_m s_1$ for some $s_1 \in \mathbb{R}^{k+}$ therefore $\bar{\beta} = U_m \bar{s}$ where $\bar{s} = \alpha s_0 + (1-\alpha)s_1 \in \mathbb{R}^{k+}$. To prove that I_m is an interval it remains to show that $\bar{\beta} \in \mathcal{S}(\bar{\gamma})$. Because both $\widehat{\beta}(\gamma_0)$ and $\widehat{\beta}(\gamma_1)$ are SLOPE minimizers, we have

$$X'(y - X\widehat{\beta}(\gamma_0)) \in \gamma_0 \partial J_\lambda(m) \text{ and } X'(y - X\widehat{\beta}(\gamma_1)) \in \gamma_1 \partial J_\lambda(m).$$

By construction of $\bar{\beta}$ the following equality occurs:

$$\alpha X'(y - X\widehat{\beta}(\gamma_0)) + (1-\alpha)X'(y - X\widehat{\beta}(\gamma_1)) = X'(y - X\bar{\beta}).$$

Moreover, since $\partial J_\lambda(m)$ is a convex set, we have $\alpha\gamma_0 \partial J_\lambda(m) + (1-\alpha)\gamma_1 \partial J_\lambda(m) \subset \bar{\gamma}\partial J_\lambda(m)$. Consequently, $X'(y - X\bar{\beta}) \in \bar{\gamma}\partial J_\lambda(m) = \bar{\gamma}\partial J_\lambda(\bar{\beta})$ thus $\bar{\beta} \in \mathcal{S}(\bar{\gamma})$.

1: characterization of I_m) The proof of this characterization is closely related to the proof of Theorem 3.1 in [6].

Necessity. If $\gamma \in I_m$, then there exists $\widehat{\beta} \in \mathcal{S}(\gamma)$ such that $\mathrm{patt}(\widehat{\beta}) = m$. Consequently, $\widehat{\beta} = U_m s$ for some $s \in \mathbb{R}^{k+}$. Because $\widehat{\beta}$ is a element of $\mathcal{S}(\gamma)$ whose pattern is m then $X'(y - \widehat{\mathrm{fit}}(\gamma)) \in \gamma\partial J_\lambda(\widehat{\beta}) = \gamma\partial J_\lambda(m)$. Multiplying this inclusion by U_m', we get $\tilde{X}_m'(y - \widehat{\mathrm{fit}}(\gamma)) = \gamma\tilde{\lambda}_m$ and so

$$\tilde{X}_m' y - \gamma\tilde{\lambda}_m = \tilde{X}_m'\widehat{\mathrm{fit}}(\gamma) = \tilde{X}_m'\tilde{X}_m s. \tag{5}$$

The positivity condition is proven.

We apply $(\tilde{X}_m')^+$ from the left to (5) and use the fact that $(\tilde{X}_m')^+\tilde{X}_m'$ is the projection onto $\mathrm{col}(\tilde{X}_m)$. Since $\widehat{\mathrm{fit}}(\gamma) \in \mathrm{col}(\tilde{X}_m)$, we have $(\tilde{X}_m')^+\tilde{X}_m'\widehat{\mathrm{fit}}(\gamma) = \widehat{\mathrm{fit}}(\gamma)$. Thus,

$$(\tilde{X}_m')^+\tilde{X}_m' y - \gamma(\tilde{X}_m')^+\tilde{\lambda}_m = \widehat{\mathrm{fit}}(\gamma).$$

The above equality gives the subdifferential condition:

$$\begin{aligned} \partial J_\lambda(m) \ni \frac{1}{\gamma}X'(y - \widehat{\mathrm{fit}}(\gamma)) &= \frac{1}{\gamma}X'(y - ((\tilde{X}_m')^+\tilde{X}_m' y - \gamma(\tilde{X}_m')^+\tilde{\lambda}_m)) \\ &= X'(\tilde{X}_m')^+\tilde{\lambda}_m + \frac{1}{\gamma}X'(I_n - (\tilde{X}_m')^+\tilde{X}_m')y. \end{aligned} \tag{6}$$

Sufficiency. Assume that the positivity condition and the subdifferential conditions hold true. Then, by the positivity condition, one may pick $s \in \mathbb{R}^{k+}$ for which

$$\gamma\tilde{\lambda}_m = \tilde{X}_m' y - \tilde{X}_m'\tilde{X}_m s. \tag{7}$$

Let us show that $U_m s \in \mathcal{S}(\gamma)$. By definition of U_m, we have $\mathrm{patt}(U_m s) = m$ thus $\partial J_\lambda(U_m s(\gamma)) = \partial J_\lambda(m)$. Moreover, using (6) and (7) one may deduce

$$\begin{aligned} \partial J_\lambda(U_m s) &\ni \frac{1}{\gamma}X'(y - (\tilde{X}_m')^+\tilde{X}_m' y + \gamma(\tilde{X}_m')^+\tilde{\lambda}_m) \\ &= \frac{1}{\gamma}X'(y - (\tilde{X}_m')^+\tilde{X}_m' y + (\tilde{X}_m')^+(\tilde{X}_m y - \tilde{X}_m'\tilde{X}_m s)) \\ &= \frac{1}{\gamma}X'(y - XU_m s). \end{aligned}$$

Consequently $U_m s \in \mathcal{S}(\gamma)$.

2 and 3: continuity) Let $\gamma \in (0, +\infty)$, $(\gamma_n)_{n\in\mathbb{N}}$ be a sequence converging to γ and $\widehat{\beta}(\gamma_n) \in S_{X,\gamma_n J_\lambda}(y)$. Both sequences $(\widehat{\beta}(\gamma_n))_{n\in\mathbb{N}}$ and $(\widehat{\mathrm{fit}}(\gamma_n))_{n\in\mathbb{N}}$ are bounded therefore, up to

extract a subsequence, one may assume that both $(\widehat{\beta}(\gamma_n))_{n\in\mathbb{N}}$ and $(\widehat{\mathrm{fit}}(\gamma_n))_{n\in\mathbb{N}}$ converge respectively to a limit point $l \in \mathbb{R}^p$ and $Xl \in \mathbb{R}^n$. Let $\widehat{\beta}(\gamma) \in \mathcal{S}(\gamma)$. Because $\widehat{\beta}(\gamma_n)$ is a minimizer, the following inequality occurs.

$$\frac{1}{2}\|y - \widehat{\mathrm{fit}}(\gamma_n)\|_2^2 + \gamma_n J_\lambda(\widehat{\beta}(\gamma_n)) \leq \frac{1}{2}\|y - \widehat{\mathrm{fit}}(\gamma)\|_2^2 + \gamma_n J_\lambda(\widehat{\beta}(\gamma)).$$

Taking the limit in the above expression gives

$$\frac{1}{2}\|y - Xl\|_2^2 + \gamma J_\lambda(l) \leq \frac{1}{2}\|y - \widehat{\mathrm{fit}}(\gamma)\|_2^2 + \gamma J_\lambda(\widehat{\beta}(\gamma)).$$

Because $\widehat{\beta}(\gamma) \in \mathcal{S}(\gamma)$, one may deduce that $l \in \mathcal{S}(\gamma)$ and thus $Xl = \widehat{\mathrm{fit}}(\gamma)$. Therefore, the unique limit point of the bounded sequence $(\widehat{\mathrm{fit}}(\gamma_n))_{n\in\mathbb{N}}$ is $\widehat{\mathrm{fit}}(\gamma)$. Consequently, $\lim_{n\to+\infty} \widehat{\mathrm{fit}}(\gamma_n) = \widehat{\mathrm{fit}}(\gamma)$ and thus the function $\gamma \in (0, +\infty) \mapsto \widehat{\mathrm{fit}}(\gamma)$ is continuous. Similarly, if $\mathcal{S}(\gamma)$ is a singleton then $l = \widehat{\beta}(\gamma)$, the unique limit point of the bounded sequence $(\widehat{\beta}(\gamma_n))_{n\in\mathbb{N}}$ is $\widehat{\beta}(\gamma)$ and thus $\lim_{n\to+\infty} \widehat{\beta}(\gamma_n) = \widehat{\beta}(\gamma)$. Therefore the function $\gamma \in (0, +\infty) \mapsto \widehat{\beta}(\gamma)$ is continuous.

2) When $\gamma \in I_m$ then multiplying both side of the positivity condition by $(\tilde{X}'_m)^+$ and using the fact that $(\tilde{X}'_m)^+\tilde{X}'_m$ is the projection onto $\mathrm{col}(\tilde{X}_m)$ gives

$$(\tilde{X}'_m)^+\tilde{X}'_m y - \gamma(\tilde{X}'_m)^+\tilde{\lambda}_m = (\tilde{X}'_m)^+\tilde{X}'_m\tilde{X}_m s = \tilde{X}_m s = \widehat{\mathrm{fit}}(\gamma).$$

3) The proof of statement 3) relies on Lemma 1 proved in supplementary material.

Lemma 1 *Let $X \in \mathbb{R}^{n\times p}$, $y \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}^{p+}$. There exists $\widehat{\beta} \in \mathcal{S}(\gamma)$ for which the pattern $m = \mathrm{patt}(\widehat{\beta})$ satisfies* $\ker(\tilde{X}_m) = \{0\}$.

Consequently, when $\gamma \in I_m$ and $\mathcal{S}(\gamma)$ is a singleton then $\ker(\tilde{X}_m) = \{0\}$, where $m = \mathrm{patt}(\widehat{\beta}(\gamma))$. Since $\tilde{X}'_m\tilde{X}_m$ is invertible, the positivity condition gives

$$\widehat{\beta}(\gamma) = U_m s = U_m(\tilde{X}'_m\tilde{X}_m)^{-1}(\tilde{X}'_m y - \gamma\tilde{\lambda}_m).$$

Basic notions on subdifferential, permutahedron and signed permutahedron

The results of this section will be useful to establish the proof of Proposition 1. We denote by S_p the set of permutations on the set $\{1, \ldots, p\}$. Given $\lambda \in \mathbb{R}^{p+}$, the subdifferential calculus of the sorted ℓ_1 norm satisfies the following properties [12, 28, 35]:

Subdifferential at 0: signed permutahedron The following equality holds:

$$\partial J_\lambda(0) = \mathrm{conv}\{(\sigma_1\lambda_{\pi(1)}, \ldots, \sigma_p\lambda_{\pi(p)}), \sigma_1, \ldots, \sigma_p \in \{-1, 1\}, \pi \in S_p\}.$$

The V-polytope $P^{\pm}(\lambda_1, \ldots, \lambda_p) := \mathrm{conv}\{(\sigma_1\lambda_{\pi(1)}, \ldots, \sigma_p\lambda_{\pi(p)}), \sigma_1, \ldots, \sigma_p \in \{-1, 1\}, \pi \in S_p\}$ is called the *signed permutahedron* and can be described as a H-polytope as follows [15]:

$$P^{\pm}(\lambda_1, \ldots, \lambda_p) = \left\{ x \in \mathbb{R}^p : \forall j \in \{1, \ldots, p\}, \sum_{i=1}^{j} |x|_{\downarrow i} \leq \sum_{i=1}^{j} \lambda_i \right\}.$$

This polytope is actually the unit ball of the dual sorted ℓ_1 norm [25].

Subdifferential at a constant vector: permutahedron Let $c > 0$. Then the following equality holds:

$$\partial J_\lambda(c, \ldots, c) = \mathrm{conv}\{(\lambda_{\pi(1)}, \ldots, \lambda_{\pi(p)}), \pi \in S_p\}.$$

The V-polytope $P(\lambda_1, \ldots, \lambda_p) := \mathrm{conv}((\lambda_{\pi(1)}, \ldots, \lambda_{\pi(p)}), \pi \in S_p)$ is called the *permutahedron* and can be described as an H-polytope as follows [15, 25]:

$$P(\lambda_1, \ldots, \lambda_p) = \left\{ b \in \mathbb{R}^p : \sum_{i=1}^{p} b_i = \sum_{i=1}^{p} \lambda_i \text{ and } \sum_{i=1}^{j} b_{\downarrow i} \leq \sum_{i=1}^{j} \lambda_i \ \forall j \in \{1, \ldots, p-1\} \right\}. \quad (8)$$

Subdifferential computation rule Let $b \in \mathbb{R}^p$ be such that $b_1 \geq \cdots \geq b_k > b_{k+1} \geq \cdots \geq b_p \geq 0$. Then

$$\partial J_\lambda(b) = \partial J_{\lambda_1,\ldots,\lambda_k}(b_1,\ldots,b_k) \times \partial J_{\lambda_{k+1},\ldots,\lambda_p}(b_{k+1},\ldots,b_p). \tag{9}$$

Proof of Theorem 2

Let $\pi \in S_p$ and $\epsilon \in \{-1,1\}^p$ be such that

$$|\widehat{\beta}|_\downarrow = (\epsilon_1 \widehat{\beta}_{\pi(1)}, \ldots, \epsilon_p \widehat{\beta}_{\pi(p)}),$$

and let ϕ be the orthogonal transformation defined as follows:

$$\forall x \in \mathbb{R}^p \quad \phi(x) = (\epsilon_1 x_{\pi(1)}, \ldots, \epsilon_p x_{\pi(p)}).$$

Proof of 1) Because $\widehat{\beta} \in \mathcal{S}(\gamma)$ is a SLOPE minimizer, the following equivalence holds:

$$\frac{1}{\gamma} X'(y - \widehat{\mathrm{fit}}(\gamma)) \in \partial J_\lambda(\widehat{\beta}) \Leftrightarrow \phi\left(\frac{1}{\gamma} X'(y - \widehat{\mathrm{fit}}(\gamma))\right) \in \phi(\partial J_\lambda(\widehat{\beta})) = \partial J_\lambda(|\widehat{\beta}|_\downarrow).$$

Since the components of $|\widehat{\beta}|_\downarrow$ are decreasing, $\partial J_\lambda(|\widehat{\beta}|_\downarrow)$ is a Cartesian product of permutahedra with potentially a signed permutahedron (if $\widehat{\beta}$ has a null component) [12, 28]. Specifically, we have

$$\partial J_\lambda(|\widehat{\beta}|_\downarrow) = \begin{cases} P(\lambda_1,\ldots,\lambda_{k_1}) \times \cdots \times P(\lambda_{k_{l-1}+1},\ldots,\lambda_{k_l}) \text{ if } k_l = p, \\ P(\lambda_1,\ldots,\lambda_{k_1}) \times \cdots \times P(\lambda_{k_{l-1}+1},\ldots,\lambda_{k_l}) \times P^{\pm}(\lambda_{k_l+1},\ldots,\lambda_p) \text{ if } k_l < p. \end{cases}$$

According to (8), if $b \in P(\lambda_1,\ldots,\lambda_{k_1}) \times \cdots \times P(\lambda_{k_{l-1}+1},\ldots,\lambda_{k_l})$, then the following equalities hold:

$$\forall i \in \{k_1,\ldots,k_l\}, \ \sum_{j=1}^{i} b_j = \|b\|_{(i)} = \sum_{j=1}^{i} \lambda_j.$$

Finally, since the $i-$norm $\|.\|_{(i)}$ is invariant by the transformation ϕ, one may deduce the following equalities:

$$\forall i \in \{k_1,\ldots,k_l\}, \ \frac{\left\|\phi\left(\frac{1}{\gamma} X'(y - \widehat{\mathrm{fit}}(\gamma))\right)\right\|_{(i)}}{\sum_{j=1}^{i} \lambda_j} = \frac{\left\|\frac{1}{\gamma} X'(y - \widehat{\mathrm{fit}}(\gamma))\right\|_{(i)}}{\sum_{j=1}^{i} \lambda_j} = 1.$$

Proof of 2) First, let us establish for $b \in \mathbb{R}^p$ such that $b_1 \geq \cdots \geq b_p > 0$ the following inclusion:

$$\partial J_\lambda(b) \subset \mathrm{conv}\left\{(\lambda_{\pi(1)},\ldots,\lambda_{\pi(p)}), \pi \in \mathcal{S}_p\right\} = P(\lambda_1,\ldots,\lambda_p). \tag{10}$$

Since the sorted ℓ_1 norm is polyhedral, namely

$$J_\lambda(b) = \max\{\sum_{i=1}^{p} \epsilon_i \lambda_{\pi(i)} b_i : \epsilon_1,\ldots,\epsilon_p \in \{-1,1\}, \pi \in \mathcal{S}_p\},$$

its subdifferential is given by

$$\partial J_\lambda(b) = \mathrm{conv}\left\{(\epsilon_1 \lambda_{\pi(1)},\ldots,\epsilon_p \lambda_{\pi(p)}), \epsilon_1,\ldots,\epsilon_p \in \{-1,1\}, \pi \in \mathcal{S}_p : \sum_{i=1}^{p} \epsilon_i \lambda_{\pi(i)} b_i = J_\lambda(b)\right\}.$$

Moreover, if $\epsilon_{i_0} = -1$ for some $i_0 \in \{1,\ldots,p\}$, then

$$\sum_{i=1}^{p} \epsilon_i \lambda_{\pi(i)} b_i < \lambda_{i_0} b_{i_0} + \sum_{i \neq i_0} \epsilon_i \lambda_{\pi(i)} b_i \leq J_\lambda(b).$$

Therefore $(\epsilon_1 \lambda_{\pi(1)},\ldots,\epsilon_p \lambda_{\pi(p)}) \notin \partial J_\lambda(x)$, which proves inclusion (3).

Now, let us assume that there exists $i \notin \mathcal{A}(\gamma)$ such that

$$\begin{cases} |\widehat{\beta}|_{\downarrow i} > |\widehat{\beta}|_{\downarrow i+1} & \text{if } i \leq p-1, \\ |\widehat{\beta}|_{\downarrow i} > 0 & \text{if } i = p. \end{cases}$$

Then, according to (9) and (3), we have $\partial J_{\lambda_1,\ldots,\lambda_i}(|\widehat{\beta}|_{\downarrow 1},\ldots,|\widehat{\beta}|_{\downarrow i}) \subset P(\lambda_1,\ldots,\lambda_i)$. Consequently

$$\frac{\left\|\phi\left(\frac{1}{\gamma}X'(y-\widehat{\mathrm{fit}}(\gamma))\right)\right\|_{(i)}}{\sum_{j=1}^{i}\lambda_j} = \frac{\left\|\frac{1}{\gamma}X'(y-\widehat{\mathrm{fit}}(\gamma))\right\|_{(i)}}{\sum_{j=1}^{i}\lambda_j} = 1.$$

Therefore $i \in \mathcal{A}(\gamma)$, which leads to a contradiction.

Existence of a SLOPE minimizer having less than $\mathrm{rk}(X)$ non-null clusters

Lemma 2 below provides a statement more precise than both [19, Theorem 2.1] and [28, Corollary 9], proving that, under the assumption of uniqueness, the unique element $\widehat{\beta}$ of $\mathcal{S}(\gamma)$ has a number of non-null clusters smaller or equal to $\mathrm{rk}(X)$ (*i.e.* $\|\mathrm{patt}(\widehat{\beta})\|_\infty \leq \mathrm{rk}(X)$).

Lemma 2 *Let $X \in \mathbb{R}^{n\times p}$, $y \in \mathbb{R}^n$, $\lambda \in \mathbb{R}^{p+}$ and $\gamma > 0$. Then either $\mathcal{S}(\gamma) = \{0\}$ or there exists $\widehat{\beta} \in \mathcal{S}(\gamma)$ for which the pattern $m = \mathrm{patt}(\widehat{\beta})$ satisfies $\|m\|_\infty = \mathrm{rk}(\tilde{X}_m)$.*

Note that by construction of $\tilde{X}_m = XU_m$, $\mathrm{rk}(\tilde{X}_m) \leq \mathrm{rk}(X)$. Moreover, $\|m\|_\infty = \mathrm{rk}(\tilde{X}_m)$ if and only if $\ker(\tilde{X}_m) = \{0\}$.

Proof: If $0 \in \mathcal{S}(\gamma)$ and since every elements in $\mathcal{S}(\gamma)$ have the same sorted ℓ_1 norm, one may deduce that $\mathcal{S}(\gamma) = \{0\}$. Now, let us assume that $0 \notin \mathcal{S}(\gamma)$. Let $\widehat{\beta} \in \mathcal{S}(\gamma)$ be such that the number of non-null clusters $k = \|\mathrm{patt}(\widehat{\beta})\|_\infty = \|m\|_\infty \geq 1$ is minimal. Let us prove that $\ker(\tilde{X}_m) = \{0\}$. If $\dim(\ker(\tilde{X}_m)) \geq 1$, then pick $h \in \ker(\tilde{X}_m), h \neq 0$. Then set $\widehat{\beta} = U_m s$ where $s \in \mathbb{R}^{k+}$ and $c(t) = \widehat{\beta} + tU_m h = U_m(s+th)$. Since $\tilde{X}_m h = XU_m h = 0$, then $X'(y - Xc(t)) = X'(y - X\widehat{\beta})$. Let $t_{\min} = \inf\{|t| : s + th \notin \mathbb{R}^{k+}\} > 0$; by construction, for $t \in (-t_{\min}, t_{\min})$, $s + th \in \mathbb{R}^{k+}$ and thus $\mathrm{patt}(c(t)) = m$. Consequently,

$$\begin{array}{rll} & \forall t \in (-t_{\min}, t_{\min}) & X'(y - Xc(t)) \in \partial J_\lambda(m) = \partial J_\lambda(c(t)), \\ \Rightarrow & \forall t \in (-t_{\min}, t_{\min}) & c(t) \in \mathcal{S}(\gamma). \end{array}$$

Since $\mathcal{S}(\gamma)$ is a closed set, one may deduce that $c(\pm t_{\min}) \in \mathcal{S}(\gamma)$. Finally, by construction of $t_{\min}$, one of the vectors $s + t_{\min}h$ or $s - t_{\min}h$ does not have k distinct components, therefore $\|\mathrm{patt}(c(t_{\min}))\|_\infty < k$ or $\|\mathrm{patt}(c(-t_{\min}))\|_\infty < k$ which contradicts the fact that $\widehat{\beta} \in \mathcal{S}(\gamma)$ has a minimal number of non-null clusters.

SLOPE is a generalized LASSO if and only if λ is an arithmetic progression

Let $D \in \mathbb{R}^{m\times p}$. The subdifferential at 0 of the function $b \in \mathbb{R}^p \mapsto \|Db\|_1$ is $D'[-1,1]^m$. The polytope $D'[-1,1]^m$ is a zonotope (the image of a cube under an affine transformation). On the other hand the signed permutahedron (the subdifferential at 0 of J_λ) is a zonotope if and only if λ is an arithmetic progression [15, Theorem 4.13]. Consequently, when λ is not an arithmetic progression one cannot pick a matrix $D \in \mathbb{R}^{m\times p}$ such that $J_\lambda(.) = \|D.\|_1$ thus SLOPE is not a generalized LASSO. On the other hand OSCAR (*i.e.* SLOPE when $\lambda_1 \geq \cdots \geq \lambda_p \geq 0$ is an arithmetic progression) is clearly a particular generalized LASSO.

References

[1] Taylor B Arnold and Ryan J Tibshirani. Efficient implementations of the generalized lasso dual path algorithm. *Journal of Computational and Graphical Statistics*, 25(1):1–27, 2016.

[2] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.

[3] Pierre C Bellec, Guillaume Lecué, and Alexandre B Tsybakov. Slope meets lasso: improved oracle bounds and optimality. *The Annals of Statistics*, 46(6B):3603–3642, 2018.

[4] Quentin Bertrand, Quentin Klopfenstein, Mathurin Massias, Mathieu Blondel, Samuel Vaiter, Alexandre Gramfort, and Joseph Salmon. Implicit differentiation for fast hyperparameter selection in non-smooth convex learning. *The Journal of Machine Learning Research*, 23(1): 6680–6722, 2022.

[5] Małgorzata Bogdan, Ewout Van Den Berg, Chiara Sabatti, Weijie Su, and Emmanuel J Candès. Slope—adaptive variable selection via convex optimization. *The annals of applied statistics*, 9 (3):1103, 2015.

[6] Małgorzata Bogdan, Xavier Dupuis, Piotr Graczyk, Bartosz Kołodziejek, Tomasz Skalski, Patrick Tardivel, and Maciej Wilczyński. Pattern recovery by slope. *arXiv preprint arXiv:2203.12086*, 2022.

[7] Howard D Bondell and Brian J Reich. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. *Biometrics*, 64(1):115–123, 2008.

[8] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[9] Peter Bühlmann, Markus Kalisch, and Lukas Meier. High-dimensional statistics with a view toward applications in biology. *Annual Review of Statistics and Its Application*, 1:255–278, 2014.

[10] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4):547–553, 2009.

[11] Charles Dossal, Maher Kachour, MJ Fadili, Gabriel Peyré, and Christophe Chesneau. The degrees of freedom of the lasso for general design matrix. *Statistica Sinica*, pages 809–828, 2013.

[12] Xavier Dupuis and Patrick JC Tardivel. Proximal operator for the sorted ℓ_1 norm: Application to testing procedures based on slope. *Journal of Statistical Planning and Inference*, 221:1–8, 2022.

[13] Clément Elvira and Cédric Herzet. Safe rules for the identification of zeros in the solutions of the slope problem. *SIAM Journal on Mathematics of Data Science*, 5(1):147–173, 2023.

[14] Mario Figueiredo and Robert Nowak. Ordered weighted l1 regularized regression with strongly correlated covariates: Theoretical aspects. In *Artificial Intelligence and Statistics*, pages 930–938. PMLR, 2016.

[15] Thomas Godland and Zakhar Kabluchko. Projections and angle sums of belt polytopes and permutohedra. *Results in Mathematics*, 78(4):140, 2023.

[16] Piotr Graczyk, Ulrike Schneider, Tomasz Skalski, and Patrick Tardivel. Pattern recovery in penalized and thresholded estimation and its geometry. *arXiv preprint arXiv:2307.10158*, 2023.

[17] Bin Gu, Guodong Liu, and Heng Huang. Groups-keeping solution path algorithm for sparse regression with automatic feature grouping. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 185–193, 2017.

[18] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2004.

[19] Philipp J Kremer, Damian Brzyski, Małgorzata Bogdan, and Sandra Paterlini. Sparse index clones via the sorted ℓ_1-norm. *Quantitative finance*, 22(2):349–366, 2022.

[20] Johan Larsson, Malgorzata Bogdan, and Jonas Wallin. The strong screening rule for slope. In *Advances in Neural Information Processing Systems*, pages 14592–14603. Curran Associates, Inc., 2020.

[21] Johan Larsson, Quentin Klopfenstein, Mathurin Massias, and Jonas Wallin. Coordinate descent for slope. In *International Conference on Artificial Intelligence and Statistics*, pages 4802–4821. PMLR, 2023.

[22] Julien Mairal and Bin Yu. Complexity analysis of the lasso regularization path. In *Proceedings of the 29th International Conference on Machine Learning*, pages 353–360, 2012.

[23] Aleksandra Maj-Kańska, Piotr Pokarowski, and Agnieszka Prochenka. Delete or merge regressors for linear model selection. *Electronic Journal of Statistics*, 9(2):1749 – 1778, 2015.

[24] Kentaro Minami. Degrees of freedom in submodular regularization: A computational perspective of stein's unbiased risk estimate. *Journal of Multivariate Analysis*, 175:104546, 2020.

[25] Renato Negrinho and Andre Martins. Orbit regularization. *Advances in neural information processing systems*, 27, 2014.

[26] Shunichi Nomura. An exact solution path algorithm for slope and quasi-spherical oscar. *arXiv preprint arXiv:2010.15511*, 2020.

[27] Saharon Rosset and Ji Zhu. Piecewise linear regularized solution paths. *The Annals of Statistics*, pages 1012–1030, 2007.

[28] Ulrike Schneider and Patrick Tardivel. The geometry of uniqueness, sparsity and clustering in penalized estimation. *Journal of Machine Learning Research*, 23(331):1–36, 2022.

[29] Dhruv B Sharma, Howard D Bondell, and Hao Helen Zhang. Consistent group identification and variable selection in regression with correlated predictors. *Journal of Computational and Graphical Statistics*, 22(2):319–340, 2013.

[30] Tomasz Skalski, Piotr Graczyk, Bartosz Kołodziejek, and Maciej Wilczyński. Pattern recovery and signal denoising by slope when the design matrix is orthogonal. *Probability and Mathematical Statistics*, 42(2):283 – 302, 2022.

[31] Charles M Stein. Estimation of the mean of a multivariate normal distribution. *The annals of Statistics*, 9(6):1135–1151, 1981.

[32] Benjamin G Stokell, Rajen D Shah, and Ryan J Tibshirani. Modelling high-dimensional categorical data using nonconvex fusion penalties. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83(3):579–611, 2021.

[33] Weijie Su and Emmanuel Candes. Slope is adaptive to unknown sparsity and asymptotically minimax. *The Annals of Statistics*, 44(3):1038–1068, 2016.

[34] Atsumori Takahashi and Shunichi Nomura. Efficient path algorithms for clustered lasso and oscar. *arXiv preprint arXiv:2006.08965*, 2020.

[35] Patrick JC Tardivel, Rémi Servien, and Didier Concordet. Simple expressions of the lasso and slope estimators in low-dimension. *Statistics*, 54(2):340–352, 2020.

[36] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[37] Ryan J Tibshirani and Jonathan Taylor. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3):1335–1371, 2011.

[38] Xiangrong Zeng and Mário AT Figueiredo. Decreasing weighted sorted ℓ_1 regularization. *IEEE Signal Processing Letters*, 21(10):1240–1244, 2014.

[39] Junzi Zhang, Brendan O'Donoghue, and Stephen Boyd. Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations. *SIAM J. Optim.*, 30(4):3170–3197, 2020.

[40] Hui Zou, Trevor Hastie, and Robert Tibshirani. On the "degrees of freedom" of the lasso. *The Annals of Statistics*, 35(5):2173–2192, 2007.