



The Smooth-Lasso and other $\ell_1 + \ell_2$ -penalized methods

Mohamed Hebiri, Sara A. van de Geer

► To cite this version:

Mohamed Hebiri, Sara A. van de Geer. The Smooth-Lasso and other $\ell_1 + \ell_2$ -penalized methods. 2010. hal-00462882v1

HAL Id: hal-00462882

<https://hal.science/hal-00462882v1>

Preprint submitted on 10 Mar 2010 (v1), last revised 7 Oct 2011 (v2)

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

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

The Smooth-Lasso and other $\ell_1 + \ell_2$ -penalized methods

Mohamed Hebiri and Sara van de Geer

Abstract

We consider the linear regression problem in the high dimensional setting, i.e., the number p of covariates can be much larger than the sample size n . In such a situation one often assumes sparsity of the regression vector, i.e., that it contains many zero components. We propose a Lasso-type estimator $\hat{\beta}^{Quad}$ (where ‘*Quad*’ stands for quadratic), which is based on two penalty terms. The first one is the ℓ_1 norm of the regression coefficients used to exploit the sparsity of the regression as done by the Lasso estimator, whereas the second is a quadratic penalty term introduced to capture some additional information on the setting of the problem. We detail two special cases: the Elastic-Net $\hat{\beta}^{EN}$, introduced in [31], deals with sparse problems where correlations between variables may exist; and the S-Lasso¹ $\hat{\beta}^{SL}$, which responds to sparse problems where successive regression coefficients are known to vary slowly (in some situations, this can also be interpreted in terms of correlations between successive coefficients). From a theoretical point of view, we establish variable selection consistency results and show that $\hat{\beta}^{Quad}$ achieves a Sparsity Inequality, i.e., a bound in terms of the number of non-zero components of the ‘true’ regression vector. These results are provided under a weaker assumption on the Gram matrix than the one used by the Lasso. In some (bad) situations this guarantees a significant improvement over the Lasso. Furthermore, a simulation study is conducted and shows that when we consider the estimation accuracy, the S-Lasso $\hat{\beta}^{SL}$ performs better than known methods as the Lasso, the Elastic-Net $\hat{\beta}^{EN}$, and the Fused-Lasso (introduced in [23]), specifically when the regression vector is ‘smooth’, i.e., when the variations between successive coefficients of the unknown parameter of the regression are small. The study also reveals that the theoretical calibration of the tuning parameters imply a S-Lasso solution with close performance to the S-Lasso when the tuning parameters are chosen by 10 fold cross validation.

Keywords: Lasso, Elastic-Net, LARS, Sparsity, Variable selection, Restricted eigenvalues, High-dimensional data.

AMS 2000 subject classifications: Primary 62J05, 62J07; Secondary 62H20, 62F12.

1 Introduction

We focus on the usual linear regression model:

$$y_i = x_i \beta^* + \varepsilon_i, \quad i = 1, \dots, n, \quad (1)$$

where the design $x_i = (x_{i,1}, \dots, x_{i,p}) \in \mathbb{R}^p$ is deterministic, $\beta^* = (\beta_1^*, \dots, \beta_p^*)' \in \mathbb{R}^p$ is the unknown parameter and $\varepsilon_1, \dots, \varepsilon_n$, are independent identically distributed (i.i.d.) centered Gaussian random variables with known variance σ^2 . We wish to estimate β^* in the sparse

¹The S-Lasso estimator has initially been introduced in the paper titled *Regularization with the Smooth-Lasso procedure*, in [12]. Results can be found there for the this method which are not provided here, such as the theoretical performance when $p \leq n$ and a simulation study from a variable selection point of view.

case, that is when many of its unknown components equal zero. Thus only a subset of the design covariates $(X_j)_j$ is truly of interest where $X_j = (x_{1,j}, \dots, x_{n,j})'$, $j = 1, \dots, p$. Moreover we are interested in the high dimensional problem where $p \gg n$, so that we can consider p depending on n . In such a framework, two main issues arise: i) the interpretability of the resulting prediction; ii) the control of the variance in the estimation. Regularization is therefore needed. For this purpose we use selection type procedures of the following form:

$$\tilde{\beta} = \underset{\beta \in \mathbb{R}^p}{\text{Argmin}} \{ \|Y - X\beta\|_n^2 + \text{pen}(\beta) \}, \quad (2)$$

where $X = (x'_1, \dots, x'_n)'$, $Y = (y_1, \dots, y_n)'$ and $\text{pen} : \mathbb{R}^p \rightarrow \mathbb{R}$ is a positive convex function called the penalty. For any vector $a = (a_1, \dots, a_n)'$, we have adopted the notation $\|a\|_n^2 = n^{-1} \sum_{i=1}^n |a_i|^2$ (we denote by $\langle \cdot, \cdot \rangle_n$ the corresponding inner product in \mathbb{R}^n). The choice of the penalty appears to be crucial. Although well-suited for variable selection purpose, concave-type penalties (see for instance [8, 11, 24]) are often computationally hard to optimize. Lasso-type procedures (modifications of the ℓ_1 penalized least square (Lasso) estimator introduced by [22]) have been extensively studied during the last few years. Between many others, see [2, 3, 6, 29] and references inside. Such procedures seem to respond to our objective as they perform both regression parameters estimation and variable selection with low computational cost. We will explore this type of procedures in our study.

In this paper, we propose a novel estimator, denoted by $\hat{\beta}^{Quad}$, which is modification of the Lasso. It is defined as the solution of the optimization problem (2) when the penalty function is a combination of the Lasso penalty (i.e., $\sum_{j=1}^p |\beta_j|$) and the quadratic penalty $\beta' \mathbf{J} \mathbf{J} \beta$ for some $p \times m$ matrix \mathbf{J} ($m \in \mathbb{N}^*$). We add this second term to the Lasso procedure for two major issues. First we exploit this second penalty in order to take into account some prior information on the data or the regression vector that the Lasso may not (as correlation between variables or a specified structure on the regression vector). Second the quadratic penalty is introduced to overcome (or to reduce) theoretical problems observed by the Lasso estimator. Indeed, in several works ([2, 3, 14, 17, 26, 28, 29, 30] among others) conditions to guarantee good performance in prediction, estimation or variable selection for the Lasso procedure are given. See also [25] for an overview of the conditions used to establish the theoretical results according to the Lasso. It was shown that the Lasso does not ensure good performance when high correlations exist between the covariates. We establish theoretical results for $\hat{\beta}^{Quad}$ that states that this estimator guarantees good performance under a weaker assumption than the Lasso estimator. The improvement is specifically observed when the Lasso achieves poor results. Two particular cases of the estimator $\hat{\beta}^{Quad}$ are mainly considered: the Elastic-Net, introduced in [31] to deal with problems where correlations between variables exist. It is defined with the quadratic penalty term $\sum_{j=1}^p \beta_j^2$. The second and novel procedure is called the *Smooth-lasso* (*S-lasso*) estimator. It is defined with the ℓ_2 -fusion penalty, i.e., $\sum_{j=2}^p (\beta_j - \beta_{j-1})^2$. The ℓ_2 -fusion penalty was first introduced in [13]. This term helps to tackle situations where correlations between successive variables exist or the regression vector is structured such that its coefficients vary slowly. Let us say in this case that the regression vector is 'smooth'. Note however that our theoretical study takes into account a large amount of procedures such as the closely related procedure 'Weighted Fusion' introduced in [9], as detailed in Remark 1.

From a practical point of view, some problems are also encountered when we solve the Lasso criterion (for instance with the LARS algorithm [10]). Indeed this algorithm fails to select a complete group of correlated covariates. Two major lacks follow. First the Lasso is

not consistent neither in variable selection nor in estimation (bad reconstitution of β^*). In this paper we focus on the estimation issue. We consider the case where the regression vector β^* is structured. We invoke the *S-lasso* estimator to respond to such problems where the covariates are ranked so that the regression vector is ‘smooth’ (i.e., the vector β^* consists in small variations in its successive components). We will see through simulations that such situations support the use of the *S-lasso* estimator. This estimator is inspired by the *Fused-Lasso* ([23]). Both S-Lasso and Fused-Lasso combine a ℓ_1 -penalty with a fusion term ([13]). The fusion term is suggested to make successive coefficients as close as possible to each other. The main difference between the two procedures is that we use the ℓ_2 distance between the successive coefficients (i.e., the ℓ_2 -fusion penalty) whereas the Fused-Lasso uses the ℓ_1 distance (i.e., the ℓ_1 -fusion penalty: $\sum_{j=2}^p |\beta_j - \beta_{j-1}|$). Hence, compared to the Fused-Lasso, we sacrifice sparsity in changes between successive coefficients in the estimation of β^* in favor of an easier optimization due to the strict convexity of the ℓ_2 distance. This implies a large reduction of computational cost. However, sparsity is yet ensured by the Lasso penalty. The ℓ_2 -fusion penalty helps to provide ‘smooth’ solutions. Consequently, even if there is no perfect match between successive coefficients our results are still interpretable. From a theoretical point of view, the ℓ_2 distance also helps us to provide theoretical properties for the S-Lasso which in some situations appears to outperform the Lasso and the Elastic-Net ([31]), another Lasso-type procedure. Let us mention that variable selection consistency of the Fused-Lasso and the corresponding Fused adaptive Lasso has also been studied in [20] but in a different context from the one in the present paper. The results obtained in [20] are established not only under the sparsity assumption, but the model is also supposed to be *blocky*, that is the non-zero coefficients are represented in a block fashion with equal values inside each block.

Many techniques have been proposed to solve the weaknesses of the Lasso. The Fused-Lasso procedure is one of them and we give here some of the most popular methods; the Adaptive Lasso was introduced by [30], which is similar to the Lasso but with adaptive weights used to penalize each regression coefficient separately. This procedure reaches under certain (strong) conditions, ‘Oracles Properties’ (i.e., consistency in variable selection and asymptotic normality. See [30]). Another approach in the Relaxed Lasso ([16]), which aims to doubly-control the Lasso estimate: one parameter to control variable selection and the other to control shrinkage of the selected coefficients. To overcome the problem due to the correlation between covariates, group variable selection has been proposed by [27] with the Group-Lasso procedure which selects groups of correlated covariates instead of single covariates at each step. A first step to the variable selection consistency study has been proposed in [1] and Sparsity Inequalities were given in [7, 15]. Another choice of penalty has been proposed with the Elastic-Net ([31]). It is in a unified fashion that we shall treat the S-Lasso and the Elastic-Net from a theoretical point of view.

The rest of the paper is organized as follows. In the next section, we introduce the estimator $\hat{\beta}^{Quad}$ defined with the Lasso penalty on one hand and a quadratic penalty on the other hand. We also provide a way to solve the $\hat{\beta}^{Quad}$ problem with the attractive property of piecewise linearity of its regularization path. Consistency in estimation and variable selection in the high dimensional case are considered in Section 3. We finally give experimental results in Section 4 which display the S-Lasso performance against some popular methods. All proofs are postponed to the Appendix section.

2 The S-Lasso procedure

As described above, we define the S-Lasso estimator $\hat{\beta}^{SL}$ as the solution of the optimization problem (2) when the penalty function is:

$$\text{pen}(\beta) = \lambda|\beta|_1 + \mu \sum_{j=2}^p (\beta_j - \beta_{j-1})^2, \quad (3)$$

where λ and μ are two positive parameters that control on one hand the sparsity of our estimator and its smoothness on the other hand. For any vector $a = (a_1, \dots, a_p)'$, we have used the notation $|a|_1 = \sum_{j=1}^p |a_j|$. Note that when $\mu = 0$, the solution is the Lasso estimator so that it appears as a special case of the S-Lasso estimator. In a more general point of view we consider the following penalty

$$\text{pen}(\beta) = \lambda|\beta|_1 + \mu\beta' \mathbf{J}' \mathbf{J} \beta, \quad (4)$$

where \mathbf{J} is any $p \times p$ matrix. This penalty is a combination of the Lasso penalty and a quadratic penalty. Let us call $\hat{\beta}^{Quad}$ the solution of the minimization problem (2)-(4). Note that the S-Lasso penalty can be seen as a particular case of the penalty (4) with J defined by

$$\mathbf{J} = \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ 1 & -1 & \ddots & \ddots & \vdots \\ 0 & 1 & -1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 & -1 \end{pmatrix}, \quad (5)$$

and that the Elastic-Net corresponds to the case where \mathbf{J} is the identity matrix.

Remark 1. For any $j, k \in \{1, \dots, p\}$, denote by $s_{j,k} = \text{sign}\left(\frac{X_j' X_k}{n}\right)$ the sign of the sample correlation between predictor variables j and k . Denote also by $w_{j,k} \geq 0$ some predictor correlation driven weights. Given this notation, the Weighted Fusion introduced in [9] corresponds to the case where the k -th diagonal terms of \mathbf{J} equals $w_{j,k}$ and $(\mathbf{J})_{k,j} = (\mathbf{J})_{j,k} = -s_{j,k}w_{j,k}$ for $j \neq k$.

Now we deal with the solution $\hat{\beta}^{Quad}$ of (2)-(4) and its computational cost. The following lemma shows that the S-Lasso criterion can be expressed as a Lasso criterion by augmenting the data artificially.

Lemma 1. Given the dataset (X, Y) and the tuning parameters (λ, μ) . Define the extended dataset (\tilde{X}, \tilde{Y}) and $\tilde{\varepsilon}$ by

$$\tilde{X} = \begin{pmatrix} X \\ \sqrt{n\mu} \mathbf{J} \end{pmatrix}, \quad \text{and} \quad \tilde{Y} = \begin{pmatrix} Y \\ \mathbf{0} \end{pmatrix}, \quad \text{and} \quad \tilde{\varepsilon} = \begin{pmatrix} \varepsilon \\ -\sqrt{n\mu} \mathbf{J} \beta^* \end{pmatrix},$$

where $\mathbf{0}$ is a vector of size p containing only zeros, $\varepsilon = (\varepsilon_1, \dots, \varepsilon_n)'$ is the noise vector and \mathbf{J} is the $p \times p$ matrix given by the penalty (4) (\mathbf{J} is given by (5) in the case of the S-Lasso estimator). Then we have $\tilde{Y} = \tilde{X} \beta^* + \tilde{\varepsilon}$, and the estimator $\hat{\beta}^{Quad}$, solution of the minimization problem (2) with the penalty given by (4) (in the case of the S-Lasso, the penalty is given by (3)), is also the minimizer of the following Lasso-criterion

$$\frac{1}{n} \left| \tilde{Y} - \tilde{X} \beta \right|_2^2 + \lambda |\beta|_1. \quad (6)$$

This result is a consequence of simple algebra. It motivates the following comments on the estimator $\hat{\beta}^{Quad}$.

Remark 2 (*Regularization paths*). LARS is an iterative algorithm introduced in [10]. A modification of LARS can be used to construct $\hat{\beta}^{Quad}$. For a fixed μ (appearing in (3)), it constructs at each step an estimator based on the correlation between covariates and the current residual. Each step corresponds to a value of λ . Then for a fixed μ , we get the evolution of the coefficients values of $\hat{\beta}^{Quad}$ when λ varies. This evolution describes the regularization paths of $\hat{\beta}^{Quad}$ which are piecewise linear ([21]). This property implies that (again for fixed μ) the S-Lasso problem can be solved with the same computational cost as the ordinary least square (OLS) estimate using the LARS algorithm.

Remark 3 (*Implementation*). The number of covariates that the LARS algorithm and its Lasso version can select is limited by the number n of rows in the matrix X . Applied to the augmented data (\tilde{X}, \tilde{Y}) introduced in Lemma 1, the Lasso modification of the LARS algorithm is able to select all the p covariates. Then we are no longer limited by the sample size as for the Lasso ([10]).

3 Theoretical results when dimension p is larger than sample size n

In this section, we study the performance of the estimator $\hat{\beta}^{Quad}$ in the high dimensional case. In particular, we provide a non-asymptotic bound on the squared risk. We also provide a bound on the ℓ_2 estimation error of $\hat{\beta}^{Quad}$. This last result implies in particular the variable consistency of $\hat{\beta}^{Quad}$. The results of this section are proved in Appendix B. These theoretical contributions rely partly on Lemma 1. Moreover, the tuning parameters λ and μ will actually be chosen depending on the sample size n . We emphasize this dependency by adding a subscript n to these parameters.

3.1 Sparsity Inequality

Now we establish a Sparsity Inequality (SI) achieved by the estimator $\hat{\beta}^{Quad}$, that is a bound on the squared risk that takes into account the sparsity of the regression vector β^* . More precisely, we prove that the rate of convergence of $\hat{\beta}^{Quad}$ is $|\mathcal{A}^*| \log(n)/n$, where \mathcal{A}^* is the sparsity set $\mathcal{A}^* = \{j : \beta_j^* \neq 0\}$. This is the same rate as the one for the Lasso. Nevertheless, using the estimator $\hat{\beta}^{Quad}$ is attractive compared to using the Lasso since the main assumption, associated to $\hat{\beta}^{Quad}$, on the Gram matrix $\Psi^n := n^{-1}X'X$ is weaker than the Lasso one.

The general case: The first result we establish here considers the case where the matrix

$$\tilde{J} = J'J$$

is any $p \times p$ matrix. Let us first establish the assumptions needed, and the setup of this contribution. We define the regularization parameters λ_n and μ_n in the following way:

$$\lambda_n = \kappa\sigma\sqrt{\frac{\log(p)}{n}}, \quad \text{and} \quad \mu_n = \lambda_n \frac{1}{4|\tilde{J}\beta^*|_\infty}, \quad (7)$$

where $\kappa > 8\sqrt{2}$.

Remark 4. The tuning parameter μ_n , given by (7), depends on the unknown regression vector β^* . In practice, we deal with the calibration of this parameter thanks to a validation criterion such as cross validation.

Our assumption on the Gram matrix Ψ^n involves the symmetric $p \times p$ matrix K_n defined by $K_n = \Psi^n + \mu_n \tilde{J}$. Given the augmented dataset defined in Lemma 1, we note that $K_n = n^{-1} \tilde{X}' \tilde{X}$, which can be seen as an augmented Gram matrix. Let $\Theta \subset \{1, \dots, p\}$ a set of indices. Using this notation, we formulate the following assumption:

Assumption $B(\Theta)$: There is a constant $\phi > 0$ such that, for any $\Delta \in \mathbb{R}^p$ that satisfies

$$\sum_{j \notin \Theta} |\Delta_j| \leq 4\sqrt{|\Theta|} \sqrt{\sum_{j \in \Theta} \Delta_j^2}, \text{ we have}$$

$$\Delta' K_n \Delta \geq \phi \sum_{j \in \Theta} \Delta_j^2. \quad (8)$$

First of all, we note that Assumption $B(\Theta)$ is inspired by the Restricted Eigenvalue Assumption introduced in [2]. The main difference is that in that paper, the authors consider the case where $K_n = \Psi^n$, which matches with the Lasso estimator (that is $\mu_n = 0$ in our setting). Another minor difference is that the set on which the assumption should hold is larger in Assumption $B(\Theta)$ than in the Restricted Eigenvalue Assumption. Indeed, in Assumption $B(\Theta)$, the considered vectors Δ should be such that $\sum_{j \notin \Theta} |\Delta_j| \leq \text{cst} \cdot \sqrt{|\Theta|} \sqrt{\sum_{j \in \Theta} \Delta_j^2}$, whereas in [2], the authors only need to consider vectors Δ such that $\sum_{j \notin \Theta} |\Delta_j| \leq \text{cst} \cdot \sum_{j \in \Theta} |\Delta_j|$ (see also [25]). Finally, let us mention that only small subsets of indices Θ are considered in Assumption $B(\Theta)$. In particular we will consider $\Theta = \mathcal{A}^*$, the true sparsity set or $\Theta = \mathcal{B}$, a set of indices which strictly includes \mathcal{A}^* and which is not much larger. That is $|\mathcal{B}| \leq \text{cst} \cdot |\mathcal{A}^*|$. Let us now explain briefly the meaning of this hypothesis. In the case, where K_n is invertible, the condition (8) is always satisfied for any $\Delta \in \mathbb{R}^p$ with ϕ larger than the smallest eigenvalue of K_n . In the sequel, the established theoretical results will involve this quantity ϕ , and the smaller ϕ , the worse will be the performance of the method. For the Lasso estimator, which corresponds to the case $K_n = \Psi^n$ (that is $\mu_n = 0$), ϕ may be very small. In Assumption $B(\Theta)$, ϕ can be set larger thanks to a good calibration of μ_n . This helps to improve the performance of the method. Hence, larger values for μ_n are desired, in order to control suitably the eigenvalues of K_n .

Let us first provide the main result on the general case.

Theorem 1. Let \mathcal{A}^* be the sparsity set and let the tuning parameters (λ_n, μ_n) be defined as in (7). Suppose that $p \geq n$. If Assumption $B(\mathcal{A}^*)$ holds, then with probability greater than $1 - u_{n,p}$, we have

$$\left\| X\beta^* - X\hat{\beta}^{Quad} \right\|_n^2 \leq \frac{16}{3\phi} \lambda_n^2 |\mathcal{A}^*|,$$

$$(\beta^* - \hat{\beta}^{Quad})' \tilde{J} (\beta^* - \hat{\beta}^{Quad}) \leq \frac{64 |\tilde{J} \beta^*|_\infty}{3\phi} \lambda_n |\mathcal{A}^*|,$$

and

$$|\beta^* - \hat{\beta}^{Quad}|_1 \leq \frac{16}{\sqrt{3}\phi} \lambda_n |\mathcal{A}^*|,$$

where $u_{n,p} = p^{1-\kappa^2/128}$ with κ , the constant appearing in (7).

The proof of Theorem 1 is based on the ‘argmin’ definition of the estimator $\hat{\beta}^{Quad}$ and some technical concentration inequalities. It provides bounds on the prediction and the ℓ_1 estimation errors with high probability. Similar bounds were provided for the Lasso estimator by [2]. Let us mention that the constants are not optimal. We focused our attention on the dependency on n (and then on p and $|\mathcal{A}^*|$). It turns out that our results are near optimal. For instance, for the ℓ_2 risk, the S-Lasso estimator reaches nearly the optimal rate $\frac{|\mathcal{A}^*|}{n} \log(\frac{p}{|\mathcal{A}^*|} + 1)$ up to a logarithmic factor ([5, Theorem 5.1]). Moreover, Theorem 1 states a control on an error which is linked to the expected prior information which suggested the use of the estimator $\hat{\beta}^{Quad}$. Further details are given at the end of this section when we deal with the special cases: the Elastic-Net and the S-Lasso.

Remark 5. *Theorem 1 improves the performance of the Lasso thanks to the quantity ϕ introduced in Assumption B(\mathcal{A}^*). Let us denote by ϕ^0 the ϕ obtained when $\mu_n = 0$ (corresponding to the Assumption B(\mathcal{A}^*) in the Lasso case). Since this quantity appears in the bound of the squared error, we observe that the improvement using $\hat{\beta}^{Quad}$ is significant in particular when ϕ^0 is very small (much smaller than μ_n). Indeed, let us consider the clearer case, where \tilde{J} is diagonal (for instance the identity matrix corresponding to the Elastic-Net estimator). Then, Assumption B(\mathcal{A}^*) guaranties that ϕ is at least $\mu_n = cst \cdot \lambda_n$ and then, for instance, the prediction error $\|X\beta^* - X\hat{\beta}^{Quad}\|_n^2$ is bounded by $cst \cdot \sqrt{\log(p)/n} |\mathcal{A}^*|$. Although not optimal, this bound is much better than to one achieved by the Lasso.*

Sparse matrix \tilde{J} : Theorem 1 takes into account the case of general matrices $\tilde{J} = \mathbf{J}'\mathbf{J}$. We can improve the results whenever \tilde{J} is sparse. This includes the Elastic-Net (where \tilde{J} is the identity matrix) and the S-Lasso (where \tilde{J} is non-zero only on its diagonal and its upper and lower diagonals) as detailed below. Indeed, the above paragraph reveals the importance of the quantity ϕ in the performance of the considered estimator. In particular, we want ϕ be as large as possible. This can be established by increasing μ_n . In this sparse case, we define the tuning parameters by

$$\lambda_n = \kappa \sigma \sqrt{\frac{\log(p)}{n}}, \quad \text{and} \quad \mu_n = \lambda_n \frac{\sqrt{|\mathcal{A}^*|}}{|\tilde{J}\beta^*|_2}, \quad (9)$$

with here $\kappa > 4\sqrt{2}$. Thanks to this calibration of the tuning parameters, it turns out that a better error control can be obtained, for instance when we deal with diagonal matrix \tilde{J} . This is the statement of Theorem 2 below. The result also needs to consider a set of indices Θ larger than \mathcal{A}^* in Assumption B(Θ). For this purpose, let $\mathcal{B} \subset \{1, \dots, p\}$ be a set of indices such that it includes \mathcal{A}^* , the true sparsity set. This set depends on \tilde{J} and on \mathcal{A}^* . More precisely \mathcal{B} contains the indices of components which interferes in the product $\beta^{*'} \tilde{J} u$ for a given $u \in \mathbb{R}^p$. This set is not too large compared to \mathcal{A}^* when we consider the case where \tilde{J} is sparse. For instance, in the case of the Elastic-Net, $\mathcal{B} = \mathcal{A}^*$, and in the case of the S-Lasso (that we will detail later), the set \mathcal{B} is such that $|\mathcal{B}| \leq 3|\mathcal{A}^*|$. Of course, the definition of \mathcal{B} depends on \mathcal{A}^* , but here we are only interested in the magnitude of $|\mathcal{B}|$. Thanks to the sparsity of \tilde{J} , we can assume that there exists a constant $c_{\tilde{J}} \geq 1$ such that $|\mathcal{B}| \leq c_{\tilde{J}} |\mathcal{A}^*|$. Given this new notation we can establish the results for sparse matrices \tilde{J} :

Theorem 2 (\tilde{J} sparse). *Let \mathcal{A}^* be the sparsity set and consider the augmented dataset (\tilde{X}, \tilde{Y}) defined in Lemma 1. Let the tuning parameters (λ_n, μ_n) be defined as in (9). Suppose that*

$p \geq n$. Suppose that Assumption $B(\mathcal{B})$ is satisfied with a set $\mathcal{B} \supset \mathcal{A}^*$ such that $|\mathcal{B}| \leq c_{\tilde{J}}|\mathcal{A}^*|$ for a given constant $c_{\tilde{J}} \geq 1$. Then with probability greater than $1 - u_{n,p}$, we have

$$\begin{aligned} \left\| X\beta^* - X\hat{\beta}^{Quad} \right\|_n^2 &\leq \frac{16}{3\phi} \lambda_n^2 |\mathcal{A}^*|, \\ (\beta^* - \hat{\beta}^{Quad})' \tilde{J} (\beta^* - \hat{\beta}^{Quad}) &\leq \frac{16|\tilde{J}\beta^*|_2}{3\phi} \lambda_n \sqrt{|\mathcal{A}^*|}, \end{aligned} \quad (10)$$

and

$$|\beta^* - \hat{\beta}^{Quad}|_1 \leq \frac{16}{\sqrt{3}\phi} \lambda_n |\mathcal{A}^*|,$$

where $u_{n,p} = p^{1-\kappa^2/32}$ with κ , the constant appearing in (9).

Theorem 2 states that $\hat{\beta}^{Quad}$ achieves the same SI as in Theorem 1 with a value of the tuning parameter μ_n of the form $\mu_n = cst \cdot \sqrt{\frac{\log(p)}{n}} |\mathcal{A}^*|$. This rate for μ_n is the one of the general case times $\sqrt{|\mathcal{A}^*|}$. Let us consider the situation where the Lasso estimator has bad performance (that is ϕ^0 is very small; cf. Remark 5) and let $\hat{\beta}^{Quad}$ be the Elastic-Net to simplify. Then we have $\phi \geq \mu_n$ and the bound of the squared error in Theorem 2 becomes $cst \cdot \sqrt{\frac{\log(p)}{n}} |\mathcal{A}^*|$. Here, $\hat{\beta}^{Quad}$ does not reach the optimal rate but it outperforms the Lasso. Moreover exploiting the sparsity of \tilde{J} , it also improves to previous results obtained in Theorem 1. Apart from the considerations on the quantity ϕ , we observe a changing in the bound of $(\beta^* - \hat{\beta}^{Quad})' \tilde{J} (\beta^* - \hat{\beta}^{Quad})$ in Theorem 2. Indeed, the bound in Theorem 2 involves the term $|\tilde{J}\beta^*|_2 \sqrt{|\mathcal{A}^*|}$ whereas in Theorem 1 appears $|\tilde{J}\beta^*|_\infty |\mathcal{A}^*|$ which is obviously larger. We then have a better control on this error exploiting the sparsity of the matrix \tilde{J} . Let us now consider two special cases of these results:

Elastic-Net: Corresponding to the case where \tilde{J} equals the identity matrix, the Elastic-Net satisfies a Sparsity Inequality with $\mathcal{B} = \mathcal{A}^*$. Then we observe that it achieves this SI with a weaker assumption on the Gram matrix than the Lasso. Indeed, the Elastic-Net involves a perturbation on the diagonal term of the Gram matrix of order μ_n . This makes Assumption $B(\mathcal{A}^*)$ to be satisfied with a better constant ϕ . These results improve the results obtained in [4], where both of the Elastic-Net and the Lasso impose the same assumption on the Gram matrix. However, let us mention that the theoretical study does not show that the Elastic-Net is particularly useful when correlation between variables exist. Finally, we observe that in this case, Equation (10) is nothing but a SI on the ℓ_2 estimation error $|\beta^* - \hat{\beta}^{Quad}|_2^2$. Note that the rate $\lambda_n \sqrt{|\mathcal{A}^*|}$ is not optimal, but has the advantage to not requiring a more restrictive assumption than Assumption $B(\mathcal{A}^*)$. Imposing Assumption $B(\mathcal{B})$ to be satisfied with a set \mathcal{B} larger \mathcal{A}^* , a better rate can be reached.

S-Lasso: In this case, $|\tilde{J}\beta^*|_2$ is intuitively small since the S-Lasso essentially responds to problems where the regression vector is expected to be ‘smooth’. This means that the successive regression coefficients are close and then $|\mathbf{J}\beta^*|_2 = \beta^{*'} \tilde{J}\beta^* = \sum_{j=2}^p (\beta_j^* - \beta_{j-1}^*)^2$ is obviously small (we have the following worst case relation: $|\tilde{J}\beta^*|_2 \leq 2\sqrt{10} |\mathbf{J}\beta^*|_2 \leq$

$7|\mathbf{J}\beta^*|_2$). Note also that in this case Assumption $B(\Theta)$ is satisfied with a set $\Theta = \mathcal{B}$ less than three times larger than \mathcal{A}^* . This set can be expressed by

$$\mathcal{B} = \{j \in \{2, \dots, p-1\} : \beta_j^* \neq 0, \quad \beta_{j-1}^* \neq 0 \quad \text{or} \quad \beta_{j+1}^* \neq 0\},$$

and Theorem 2 holds with $c_{\tilde{J}} = 3$. Moreover, Equation (10) can be seen as a control on the ‘smoothness’ error $\sum_{j=2}^p (\delta_j - \delta_{j-1})^2$, where δ_j is the components difference $\beta_j^* - \hat{\beta}_j^{Quad}$.

Remark 6. For the *S-Lasso*, the matrix \tilde{J} is tridiagonal with its off-diagonal terms equal to -1 . If we do not consider the diagonal terms, we remark that Ψ^n and K_n differ only in the terms on the second diagonals (i.e., $(K_n)_{j-1,j} \neq (\Psi^n)_{j-1,j}$ for $j = 2, \dots, p$ as soon as $\mu_n \neq 0$). Terms in the second diagonals of Ψ^n correspond to correlations between successive covariates. When high correlations exist between successive covariates, a suitable choice of μ_n makes Assumption $B(\mathcal{B})$ satisfied. Hence, this assumption fits well with the setup where correlations between successive variables interfere. In many situations, we expect that the variables are ranked, such that not only the regression vector is ‘smooth’, but also successive covariates are correlated. In this case the *S-Lasso* estimator is particularly useful. We also observe how the ‘smoothness’ of the regression vector influences the control of the correlation on one hand (see Assumption $B(\mathcal{B})$), and the prediction and the estimation errors on the other hand (as ϕ depends on $|\mathbf{J}\beta^*|_2$). Similarly to the *Elastic-Net*, the *S-Lasso* improves the *Lasso* results, but here specifically in problems where correlations are considered between successive variables.

Remark 7. In situations where one can expect some structure on the regression vector, the second term of the penalty attempts to catch this structure. As a consequence the case where \tilde{J} is sparse is promising since the value of the tuning parameter μ_n is larger.

Remark 8. From the proofs of Theorems 1 and 2, the value of the tuning constant κ in the definition of λ_n can be taken respectively larger than $4\sqrt{2}$ and $2\sqrt{2}$ instead of $8\sqrt{2}$ and $4\sqrt{2}$. Such tuning is possible if we consider only the prediction error $\left\|X\beta^* - X\hat{\beta}^{Quad}\right\|_n^2$. In the sequel, a bound on the ℓ_2 estimation error $|\beta^* - \hat{\beta}^{Quad}|_2$ can be obtained thanks to Assumption $B(\Theta)$, which has to hold with a slightly larger set Θ than the ones considered in the above theorems.

3.2 Variable selection

Now we deal with variable selection. Let us first mention that a big amount of work has been done on the topic of variable selection of the *Lasso*. One important observation is that one has to make a compromise between identifying low signal level (that is, small in absolute value coefficients β_j^* , $j \in \mathcal{A}^*$) and imposing a large restriction on the Gram matrix Ψ^n , which sometimes seems to be not realistic. Moreover, the question of the identifiability of β^* has also to be considered. Our approach consists in the choice of the middle road, that is, involving the less restrictive assumption on the Gram matrix that permit us to recover reasonably low signal level. For this purpose we first provide a bound on the sup-norm $|\beta_{\mathcal{A}^*}^* - \hat{\beta}_{\mathcal{A}^*}^{Quad}|_\infty$, where for any p -dimensional vector a and subset $\Theta \subset \{1, \dots, p\}$, the notation a_Θ means that $(a_\Theta)_j = a_j$ for any $j \in \Theta$ and zero otherwise. Thanks to the theorems stated in the previous

section, one can easily use the ℓ_1 estimation error $|\beta^* - \hat{\beta}^{Quad}|_1$ to get a bound on the sup-norm error. Nevertheless, this implies that only ‘high’ levels of signal can be reconstituted, i.e., coefficients β_j^* , $j \in \mathcal{A}^*$ such that $|\beta_j^*| \geq cst \cdot \lambda_n |\mathcal{A}^*|$. For this reason, we present a result on the ℓ_2 estimation error $|\beta^* - \hat{\beta}^{Quad}|_2$ which by the sequel ables us to recover signals such $|\beta_j^*| \geq cst \cdot \lambda_n \sqrt{|\mathcal{A}^*|}$ with the same Restricted Eigenvalue assumption. Let us mention that $\lambda_n \sqrt{|\mathcal{A}^*|}$ is not the best level which can be recovered. One can also get rid of the term $\sqrt{|\mathcal{A}^*|}$ through a quite restrictive assumption on the correlations between variables such as Mutual Coherence assumption: $\max_{j \in \mathcal{A}^*} \max_{\substack{k \in \{1, \dots, p\} \\ k \neq j}} |(K_n)_{j,k}| \leq \frac{t}{|\mathcal{A}^*|}$, where t is a small constant. Let us first state the assumption on the regression parameter.

Assumption C: *The true regression vector β^* is such that*

$$\min_{j \in \mathcal{A}^*} |\beta_j^*| > \tilde{c} \lambda_n \sqrt{|\mathcal{A}|},$$

where $\tilde{c} = \frac{4}{\sqrt{3}\phi}$ and ϕ is the constant appearing in Assumption B(\mathcal{A}^*).

Note that the constant \tilde{c} is the same as the one used in Proposition 1 below. Here again, we observe how important the quantity ϕ is. We want it to be as large as possible.

Proposition 1. *Let us consider the same setting as in Theorem 1: consider the linear regression model (1). Let $\lambda_n = \kappa \sigma \sqrt{\log(p)/n}$ and $\mu_n = \lambda_n / (4|\tilde{J}\beta^*|_\infty)$ with $\kappa > 8\sqrt{2}$. Suppose that $p \geq n$. Under Assumptions B(\mathcal{A}^*)-C, and with probability larger than $1 - p^{-1 - \frac{\kappa^2}{128}}$, we have*

$$|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_\infty \leq |\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2 \leq \tilde{c} \lambda_n \sqrt{|\mathcal{A}|},$$

where $\tilde{c} = \frac{4}{\sqrt{3}\phi}$ and ϕ is the constant appearing in Assumption B(\mathcal{A}^*). Moreover, we have

$$\mathbb{P} \left(\text{Sgn}(\hat{\beta}_{\mathcal{A}}^{Quad}) \neq \text{Sgn}(\beta_{\mathcal{A}}^*) \right) \leq p^{-1 - \kappa^2/128}.$$

Proposition 1 is a trivial consequence of Theorem 1. A small proof is given in the Appendix section. The previous proposition directly underlines that under the Restrictive Eigenvalue Assumption B(\mathcal{A}^*), all non-zero components of β^* are detected by $\hat{\beta}^{Quad}$. Actually, in the setting of Proposition 1, $\hat{\beta}^{Quad}$ contains too many non-zero components. More restrictions are needed in order to ensure the variable selection consistency of $\hat{\beta}^{Quad}$. Here is an additional assumption on the Gram matrix which controls the correlations between the truly relevant variables and those which are not.

Assumption D: *We assume that*

$$\max_{j \in \mathcal{A}^*} \max_{k \notin \mathcal{A}^*} |(K_n)_{j,k}| \leq \frac{t}{|\mathcal{A}^*|},$$

where t is a positive term smaller than $\frac{\sqrt{3}\phi}{128}$.

This assumption is quite close to the Mutual Coherence assumption which involves the Gram matrix Φ^n instead of K_n . In addition, the Mutual Coherence assumption makes a restriction on correlations between all covariates.

Theorem 3. *Let consider the linear regression model (1). Let $\lambda_n = \kappa\sigma\sqrt{\log(p)/n}$ and $\mu_n = \lambda_n/(4|\tilde{J}\beta^*|_\infty)$ with $\kappa > 16$. Suppose that $p \geq n$. Under Assumptions $B(\mathcal{A}^*)$ -C and also Assumption D, we have*

$$\mathbb{P}(\hat{\mathcal{A}} \not\subseteq \mathcal{A}^*) \leq 2p^{2-\kappa^2/128},$$

and then

$$\mathbb{P}\left(\text{Sgn}(\hat{\beta}^{Quad}) \neq \text{Sgn}(\beta^*)\right) \leq 2p^{2-\kappa^2/128}.$$

The second point is a consequence of the first and of Proposition 1. There are essentially two differences between the setting of Theorem 3 and Proposition 1. First, we need a more restrictive assumption on the correlations between variables. However, this restriction is only between relevant variables and irrelevant ones. This is a ‘quite’ reasonable assumption to identify the relevant variables, that is, the non-zero components of the vector β^* . Second, the minimal value of λ_n is larger in this last theorem. This suggests that we need a larger value of this tuning parameter to set to zero the irrelevant components.

Remark 9. *The results of Theorem 3 can also be obtained under the more restrictive Mutual Coherence assumption: $\max_{j \in \mathcal{A}^*} \max_{\substack{k \in \{1, \dots, p\} \\ k \neq j}} |(K_n)_{j,k}| \leq \frac{\tilde{t}}{|\mathcal{A}^*|}$. Here even the correlations between relevant variables are restricted but this restriction makes possible to recover even smaller signal. That is, we can detect coefficients of β^* such that $|\beta_j^*| \geq cst \cdot \sqrt{\log(p)/n}$.*

Remark 10. *We presented in Theorem 3 a variable selection result under an assumption on the perturbed Gram matrix K_n , which is a combination of the Restricted Eigenvalue and the Mutual Coherence assumptions. However, the above results do not allow large value of the tuning parameter μ_n . Indeed, we recall that when \tilde{J} is sparse, the rate of μ_n can reach $\sqrt{\frac{\log(p)}{n}}|\mathcal{A}^*|$. Let us mention that such a rate can be calibrated here to obtain variable selection consistency if we use a hard thresholded version of $\hat{\beta}^{Quad}$. That is, we set to zero components of $\hat{\beta}^{Quad}$ which are smaller in absolute value to a threshold, obtained thanks to the bound on the ℓ_2 estimator $|\hat{\beta}^{Quad} - \beta^*|_2$. We get such a bound without Assumption D, but paying the price of a little more restrictive version of the Restricted Eigenvalue Assumption $B(\Theta)$ (that is, Θ must be larger).*

4 Experimental Results

In this section we present the experimental performance of the estimator $\hat{\beta}^{Quad}$. In particular, we focus on two special cases: the Elastic-Net and the S-Lasso defined respectively with the penalties $\text{pen}^{EN}(\beta) = \lambda|\beta|_1 + \mu|\beta|_2^2$ and $\text{pen}^{SL}(\beta) = \lambda|\beta|_1 + \mu \sum_{j=2}^p (\beta_j - \beta_{j-1})^2$. The Elastic-Net is useful when high correlations between variables appears, whereas the S-Lasso is devoted to problems where the regression vector β^* is ‘smooth’ (small variations in the values of the successive components of β^*). We essentially are interested in the performance of these estimators w.r.t. their estimation accuracy, i.e., in terms of the estimation error $|\hat{\beta} - \beta^*|_2$, when β^* is known (simulated data). Indeed, the introduction of $\hat{\beta}^{Quad}$ is motivated by a priori knowledge on the structure of the parameter β^* , or on the correlation between variables, and the purpose here is to see how this information can be taken into account to improve the reconstruction of the vector β^* . As benchmarks, we use the Lasso and the Fused-Lasso estimators, since the first is the reference method and the second is close in spirit to the S-Lasso estimator. Indeed, it is designed to produce solutions with equal values

of the successive components of β^* ('blocky') [23]. Note also that in the pioneer paper of the Elastic-Net, a 'corrected' version of this estimator is proposed [31]. There is as yet no theoretical support for this method. Moreover, it outperforms the 'non-corrected' Elastic-Net (this 'non-corrected' Elastic-Net is denoted by naive in [31]) in only a very few of the situations we consider in this paper. We omitted the results for these 'corrected' versions to avoid digressions.

Except for the Fused-Lasso solution, all of the Lasso, the S-Lasso and the Elastic-Net solutions can be computed thanks to the LARS algorithm (cf. Lemma 1). However, we will not use this LARS algorithm in this study. Indeed, in order to be fair with all the methods, we used the same algorithm for all of them. We exploit an algorithm² which is an implementation of a general convex program given by [19].

In all our experiments, the tuning parameters are chosen based on the 10 fold cross validation criterion, but we also display the results obtained based on the theoretical values. Note that for the Fused-Lasso, we considered the same theoretical values of the tuning parameters as for the S-Lasso as they are motivated by similar applications (this choice seems arbitrary but to our knowledge, no precise study has been made for the Fused-Lasso in the context we are considering). On the other hand, both Elastic-Net and the S-Lasso involve a sparse matrix \tilde{J} in the definition of the estimator $\hat{\beta}^{Quad}$. Then the theoretical values of the tuning parameters are $\lambda = \kappa\sigma\sqrt{\log(p)/n}$ and $\mu = \lambda\sqrt{\mathcal{A}^*}/|\tilde{J}\beta^*|_2$, for a positive constant κ , in accordance with the second part of Section 3.1. These quantities depend on unknown parameters. They can be used only in the simulation study, and otherwise one needs to estimate $|\tilde{J}\beta^*|_2$. Note moreover that the constant κ is fixed equal to $2\sqrt{2}$ in all the simulations (cf. Remark 8).

The different methods are applied to several simulation examples. They also have been applied to a *pseudo-real dataset* generated from the riboflavin dataset.

4.1 Synthetic data

There are several parameters: the dimension p , the sample size n and the noise level σ . They will be specified during the experiments. The first one is classical and has been introduced in the original paper of the Lasso [22]. The second one comes from the paper by [31]. Here we are interested to observe the performance of the procedures when groups of variables appear. The last two studies aim to determine the behavior of the methods when the regression vector is 'smooth'.

Example (a) $[\sigma/\rho]$: No particularities. We fix $p = 8$ and $n = 20$. Here only β_1, β_2 and β_5 are nonzero and equal respectively 3, 1.5, 2. Moreover, the design correlation matrix Σ is defined by $\Sigma_{j,k} = \rho^{-|j-k|}$ for $(j, k) \in \{1, \dots, 35\}^2$ and $\Sigma_{j,k} = \mathbb{I}(j = k)$ otherwise where $\rho \in]0, 1[$ and $\mathbb{I}(\cdot)$ is the indicator function.

Example (b) $[p/n/\sigma]$: Groups. We have $\beta_j = 3$ for $j \in \{1, \dots, 15\}$ and zero otherwise. We construct three groups of correlated variables: $\Sigma_{j,j} = 1$ for every $j \in \{1, \dots, p\}$; for $j \neq k$, $\Sigma_{j,k} \approx 1$ (actually $\Sigma_{j,k} = \frac{1}{1+0.01}$, due to an extra noise variable) when (j, k) belongs to $\{1, \dots, 5\}^2, \{6, \dots, 10\}^2$ and $\{11, \dots, 15\}^2$ and zero otherwise.

²provided by J. Mairal: <http://www.di.ens.fr/~mairal/index.php>

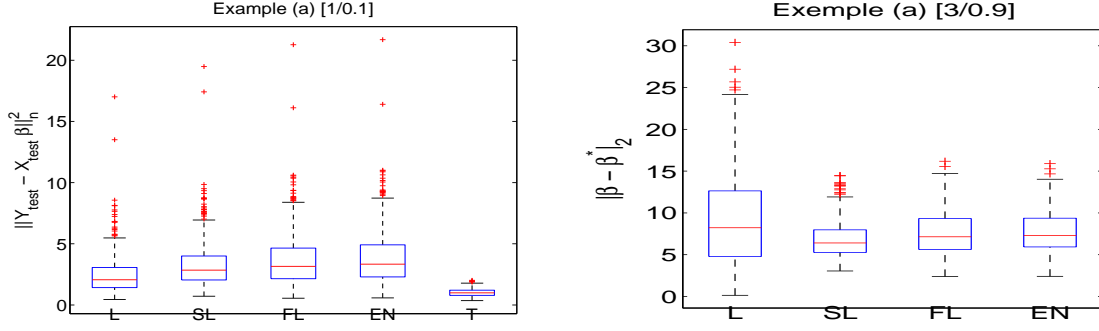


Figure 1: Performance of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) applied to *Example (a)* and based on 500 replications. The tuning parameters are chosen based on the theoretical study. *Left*: Evaluation of the prediction error $\|Y_{test} - X_{test}\hat{\beta}\|_n^2$, in comparison with the performance of the truth (T), i.e., $\|Y_{test} - X_{test}\beta^*\|_n^2$. *Right*: Evaluation of the ℓ_2 estimation error $\|\hat{\beta} - \beta^*\|_2$.

Example (c) $[p/n/\sigma]$: Smooth regression vector. The regression vector is given by $\beta_j = (3 - 0.2j)^2$ for $j = 1, \dots, 15$ and zero otherwise. Moreover, the correlations are described by $\Sigma_{j,k} = \exp(-|j - k|)$ for $(j, k) \in \{1, \dots, p\}^2$.

Example (d) $[p/n/\sigma]$: High sparsity index and smooth regression vector. The regression vector is such that $\beta_j = (4 + 0.1j)^2$ for $j \in \{1, \dots, 40\}$ and zero otherwise, and the correlations are the same as in *Example (c)*.

Except when $p = 500$ where we run only 100 replications, we based all the experiments on 500 replications.

Results. The performance of the estimator $\hat{\beta}$ (which can be the Lasso, the S-Lasso, the Elastic-Net or the Fused-Lasso) in terms of the prediction error $\|Y_{test} - X_{test}\hat{\beta}\|_n^2$ (on a test set (Y_{test}, X_{test}) of size n) and the ℓ_2 estimation error $\|\hat{\beta} - \beta^*\|_2$ are illustrated by boxplots in Figure 1 to Figure 4. For some of these experiments, the corresponding computational cost (in seconds) of each method are reported in Table 1. In what follows, we first compare the methods to each other in terms of their accuracy. Then we compare them in terms of their computational costs. Finally we provide some numerical justifications to the theoretical calibration of the tuning parameters of the S-Lasso procedure.

Methods comparison in terms of performance: Let us consider the different examples separately.

– *Example (a)*: when we consider the procedures induced by the cross validation criterion (for the choice of the tuning parameter), we notice that none of them outperforms the others even when $\rho = 0.9$ (quite large correlation between successive variables). This is observed for both prediction and estimation errors. It is essentially due to the good behavior of the Lasso in such a situation where the regression vector is sparse but without any particular structure. Actually, this conclusion holds in almost all the cases even when the tuning parameters are chosen based on the theoretical study. However, two observations can be made. First,

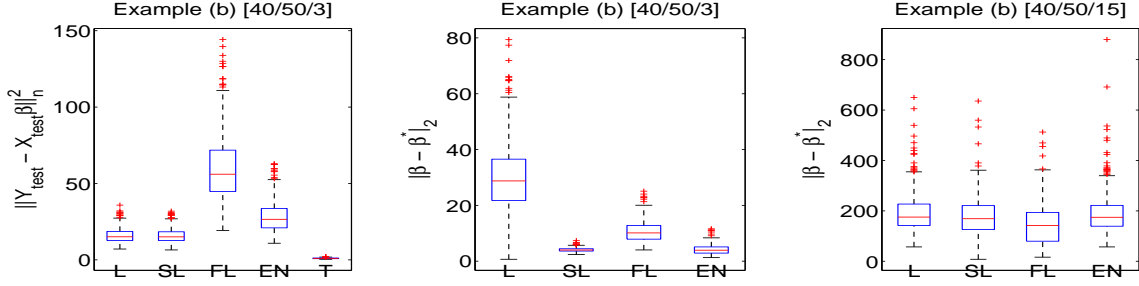


Figure 2: Performance of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) applied to *Example (b)* and based on 500 replications. The tuning parameters are chosen based on the theoretical study in the first two plots, and by 10 fold cross validation in the third. *Left*: Evaluation of the prediction error $\|Y_{test} - X_{test}\hat{\beta}\|_n^2$, in comparison with the performance of the truth (T), i.e., $\|Y_{test} - X_{test}\beta^*\|_n^2$. *Center-Right*: Evaluation of the ℓ_2 estimation error $\|\hat{\beta} - \beta^*\|_2$.

when both of ρ and σ are small, the Lasso estimator performs slightly better than the other methods, Moreover when ρ is large, a small improvement can be observed using the Lasso modification methods when we care about the estimation error. This is illustrated in Figure 1 (left and right respectively) where we display the performance of the methods in terms of the prediction error in *Example (a)* [1/0.1] (left) and in terms of the estimation error in *Example (a)* [3/0.9] (right). For this example, the Lasso seems to be the best method since it involves only one tuning parameter. It moreover has a lower (mean) computational cost equal to 0.18 seconds for the Lasso (based on the cross validation criterion) as displayed in Table 1. The S-Lasso, the Elastic-Net and the Fused-Lasso computational costs are respectively 3.7, 3.6 and 4.2 seconds.

– *Example (b)*: with *Example (a)*, this example is the less favorable for the S-Lasso. Indeed, here the fifteen first coefficients equal 3. Then the value of the coefficients drops down directly to 0. There is a breaking point in the ‘smoothness’ in the true regression vector. Figure 5 displays the best reconstitution of the regression vector β^* using the S-Lasso solution (which minimizes the ℓ_2 estimation error since β^* is known). We observe the borders problems (breaking point in the ‘smoothness’) that the S-Lasso can meet due to the ℓ_2 fusion penalty term. However, even in this case, it seems that all the procedures perform in a similar way when the tuning parameters are chosen by cross validation. When the noise level is large ($\sigma = 15$), let us nevertheless mention a (very) small improvement using the corrected versions of the S-Lasso and the Elastic-Net. Figure 2 (right) illustrates the performance of the methods in terms of the estimation error when they are applied to *Example (b)* [40/50/15]. The Fused-Lasso outperforms a little the other methods in this example (with this noise level) when we deal with the estimation performance.

On the other hand, when the methods are based on the theoretical calibration of the tuning parameters, two observations can be made regardless the noise level ($1 \leq \sigma \leq 15$): the S-Lasso and the Lasso perform better than the other methods in terms of the prediction error; the S-Lasso and the Elastic-Net provide good results whereas the Lasso guarantees poor performance in terms of estimation error. This is illustrated in Figure 2 (left and center respectively) when the methods are applied to *Example (b)* [40/50/3]. Note moreover that a similar illustration is also obtained when $p = 100$ and $n = 40$. Then the behavior of the different methods seems to be stable with the parameters p , n and σ . This example is quite interesting since it points out that a good method for the prediction objective can be less

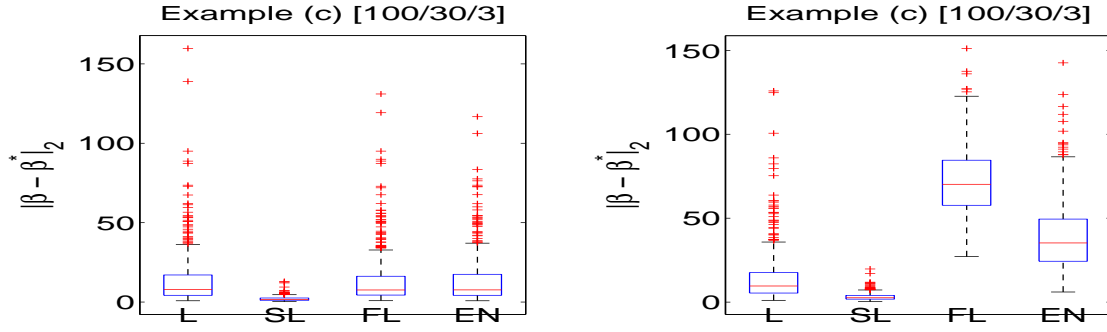


Figure 3: Evaluation of the ℓ_2 estimation error $|\hat{\beta} - \beta^*|_2$ of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) applied to *Example (c)* and based on 500 replications. *Left*: The tuning parameters are chosen by 10 fold cross validation. *Right*: The tuning parameters are chosen based on the theoretical study.

efficient for the estimation objective (see the performance of the Lasso and the Elastic-Net).
– *Example (c)*: we consider several values of the sample size n and the dimension p . It turns out that here again, when $p < n$ all the methods behave in the same way when the tuning parameters are chosen by cross validation (the S-Lasso induces just a small improvement). However when $p > n$ the S-Lasso is by far better than the other methods. This is illustrated by Figure 3 (left) where ℓ_2 estimation error of each method applied to *Example (c)* [100/30/3] is displayed. The same plot is obtained for the prediction error.

Moreover when the tuning parameters are calibrated according to the theoretical study, the S-Lasso performs the best and the Fused-Lasso the worst. This appears to be true whatever the values of the parameters p , n and σ . See for instance Figure 3 (right) where the different methods are applied to *Example (c)* [100/30/3] and for the estimation task (the same is obtained for the prediction objective).

Note that in this example, the Fused-Lasso and the Elastic-Net appear to be useless.

– *Example (d)*: this is with *Example (c)* the most favorable situation for the S-Lasso estimator where the regression vector is ‘smooth’ with a large amount of non-zero components. The S-Lasso estimator seems to dominate its opponents in all the cases, and regardless of the sample size n , the dimension p or the noise level σ . This observation holds for the ℓ_2 estimation and the prediction errors. Note that when the tuning parameters are chosen by cross validation, the Lasso, the Fused-Lasso and the Elastic-Net have quite close performance. Figure 4 illustrates this fact when $p < n$ for the estimation error (left: cross validation; center-left: theory). Moreover, Figure 4 (center-right and right) displays the performance of the methods when $p > n$ when the tuning parameters are based on the theoretical study (note that ranking of the methods does not change from the case $p < n$ when the tuning parameters are chosen by cross validation). Here an interesting observation follows from the experiments on *Example (d)* [100/30/3] (Figure 4-left). Indeed, here the sparsity index $|\mathcal{A}^*| = 40$ and it is then larger than the sample size $n = 30$. In this case, the Lasso has poor performance. However, the S-Lasso is still good. Moreover, there even exists a pair (λ, μ) (the pair minimizing the ℓ_2 estimation error since β^* is known) such that we have a good reconstitution on the regression vector β^* (see Figure 5-right).

Methods comparison in terms of computational cost: Table 1 displays the computational cost (in seconds) of each method on several examples. First note that the Fused-Lasso has the

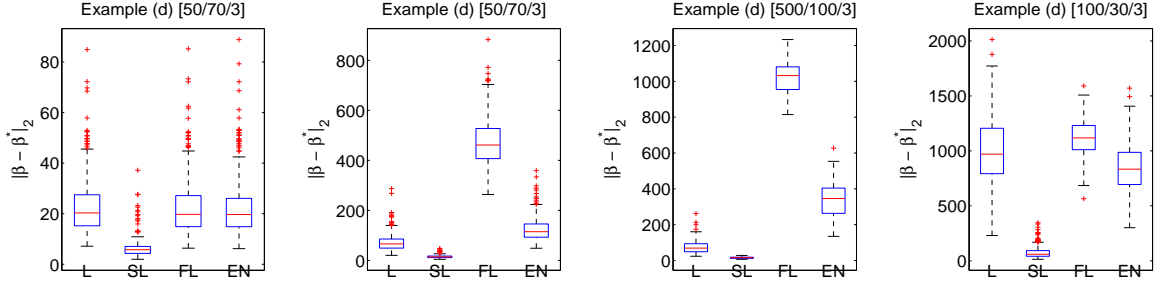


Figure 4: Evaluation of the ℓ_2 estimation error $|\hat{\beta} - \beta^*|_2$ of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) applied to *Example (c)* and based on 500 replications. *Left*: The tuning parameters are chosen by 10 fold cross validation. *Center-left*; *Center-right*; *Right*: The tuning parameters are chosen based on the theoretical study.

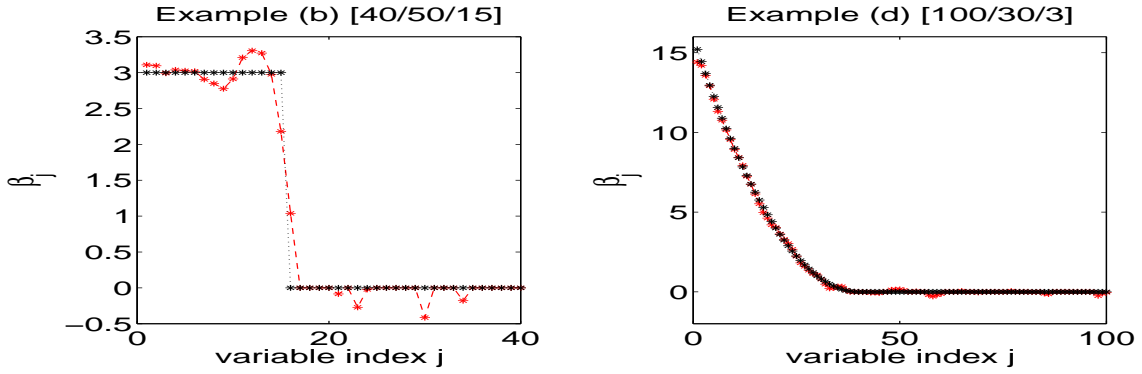


Figure 5: Best reconstitution of the regression vector β^* (black curve) by the SL-Lasso estimator (red curve). *Left*: Application to *Example (b)* [40/50/15]. *Right*: Application to *Example (d)* [100/30/3].

largest computational cost in all the simulations whereas the Lasso has the smallest. The Elastic-Net and the S-Lasso have intermediate computational costs but stay reasonable compared to the Fused-Lasso. More precisely, when the tuning parameters are chosen by cross validation, we remark that the computational costs of the S-Lasso and the Elastic-Net are about 30 times larger than the Lasso. This is partly explained by the number of values explored for the tuning parameter μ (a grid with 20 elements). Actually, even for fixed λ and μ , the computation cost of the Lasso is (a little) smaller than the computation costs of the S-Lasso and the Elastic-Net. This is observed for instance when we consider the solutions computed when the tuning parameters are chosen based on the theoretical study. The reason is that the S-Lasso and the Elastic-Net are solved thanks to a Lasso program applied to augmented data (cf. Lemma 1). Except on *Example (a)* where the increase of computational cost using the S-Lasso and the Elastic-Net is not justified (since the improvement using the Lasso-type methods is quite small), in most of the considered situations it is quite interesting to use the Elastic-Net and even more interesting to use the S-Lasso estimator. This is due to the ‘smoothness’ of the true regression vector.

On the other hand, the Fused-Lasso has a large computation cost due to the ℓ_1 -fusion penalty which is not strictly convex. Moreover, it does not improve enough the Lasso estimator in the situation we considered in this paper (as observed in the previous part).

In view of the computational costs related to *Example (a)* (the first two columns in Table 1), let us finally remark that these costs increase with ρ , the correlation level between variables,

Table 1: Computation costs in seconds of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) on the examples illustrated in Figure 1. The parameter $Tuning = Th$ or $Tuning = Cv$ depending on whether we consider the methods with the tuning parameters based on the theoretical issue or on the 10 fold cross validation respectively.

METH.	TUNING	$Ex.(A) [1/0.1]$	$Ex.(A) [3/0.9]$	$Ex.(B) [40/50/15]$	$Ex.(C) [30/50/3]$	$Ex.(D) [30/50/3]$
L	$Th \cdot 10^{-4}$	1.1 ± 0.1	8 ± 41	5 ± 2	33 ± 64	457 ± 243
	Cv	0.18 ± 0.01	0.5 ± 0.2	0.5 ± 0.1	1.1 ± 0.3	12.3 ± 4.9
SL	$Th \cdot 10^{-4}$	5.1 ± 6.4	8 ± 28	6 ± 6	48 ± 81	967 ± 441
	Cv	3.7 ± 0.1	11.1 ± 1.3	10.2 ± 2.0	36.2 ± 9.1	648.3 ± 219.2
FL	$Th \cdot 10^{-4}$	2.6 ± 0.3	10.0 ± 30.0	20 ± 12	518 ± 271	5996 ± 2019
	Cv	4.2 ± 0.2	14.1 ± 1.6	38.3 ± 5.8	245.6 ± 64.3	$\simeq 3 \cdot 10^3$
EN	$Th \cdot 10^{-4}$	4.7 ± 3.5	9 ± 43	5 ± 3	41 ± 60	1022 ± 432
	Cv	3.6 ± 0.2	11.0 ± 1.3	10.2 ± 2.0	35.2 ± 8.9	637.3 ± 214.0

and σ , the noise level. We observe for instance that the mean computational cost of the Lasso estimator (when the tuning parameter is chosen by cross validation) is 1.1 seconds when $\rho = 0.1$ and $\sigma = 1$ and increases to 8 seconds when $\rho = 0.9$ and $\sigma = 3$.

S-Lasso; theory vs. cross validation: Figure 6 resumes the comparison between the S-Lasso based on a theoretical choice of the tuning parameters (denoted by this part S-LassoTh) and the S-Lasso where the tuning parameters are based on 10 fold cross validation (denoted here by S-Lasso^{Cv}). First we can observe that the performance of both S-LassoTh and S-Lasso^{Cv} are close. Moreover given the results in the part ‘Methods comparison in terms of performance’, they both perform in a good way. However, it seems that S-Lasso^{Cv} outperforms S-LassoTh when we deal with the prediction task. This seems quite intuitive since by definition, the cross validation criterion attempts to provide good estimator for the prediction objective. According to the ℓ_2 estimation goal, we cannot conclude the superiority of one of the estimator on the other. Nevertheless, in the high dimensional setting *Example* (d) [500/100/ σ], it seems that S-Lasso^{Cv} begins to become better.

Hence it turns out that the theoretical choice for μ ($\mu = \frac{\lambda\sqrt{A^*}}{|J\beta^*|_2}$) provides good performance both in terms of ℓ_2 estimation error and test error. Moreover, they are often close to the performance of the S-Lasso estimator based on the cross validation criterion. This is quite interesting since the computational cost of S-LassoTh is much smaller than S-Lasso^{Cv}. This study is actually more a verification of our theoretical choices of the tuning parameters than a rule to apply in practice. Indeed, since the theoretical choice of μ depends on β^* , the corresponding estimator S-LassoTh is unusable in real data problems.

Conclusion of the experimental results. The S-Lasso has good performance when the regression vector is ‘smooth’ (*Examples* (c)-(d)). Nevertheless, even in situations made in favor of the Elastic-Net and the Fused-Lasso (*Examples* (b)), the S-Lasso performs similarly to the other methods when the tuning parameters are chosen based on the cross validation criterion. The S-Lasso is even better in these examples when the methods are constructed based on the theoretical considerations.

Moreover all the results according to the procedures for which the tuning parameters are chosen based on the theoretical study is a little unfair in disfavor of the Fused-Lasso. Indeed the rates of the tuning parameters have been calibrated based on a study made for the estimator $\hat{\beta}^{Quad}$ (the Elastic-Net and the S-Lasso are two particular cases of this estimator).

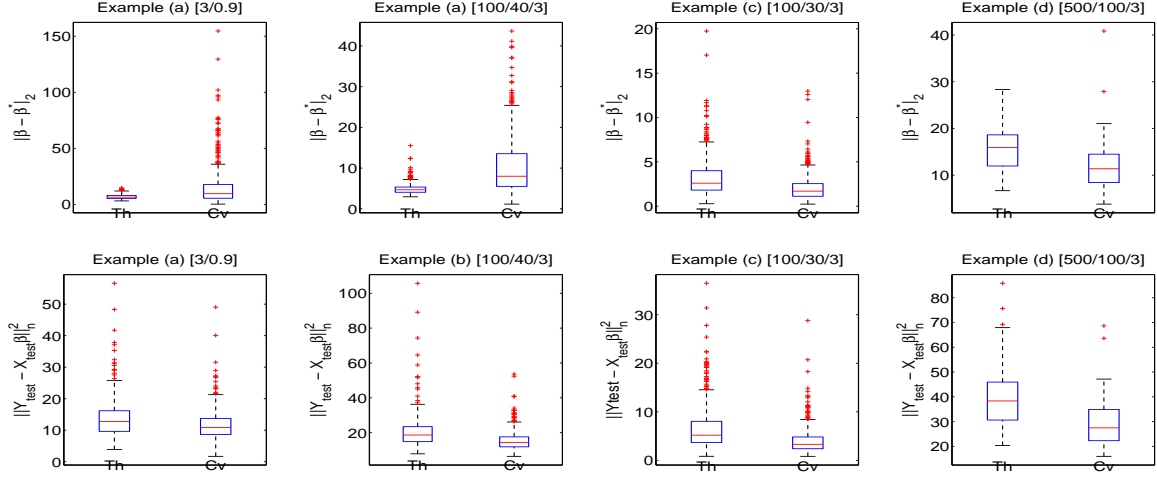


Figure 6: Evaluation of the ℓ_2 estimation error $|\hat{\beta} - \beta^*|_2$ of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) based on 500 replications. *Left*: The tuning parameters are chosen by 10 fold cross validation. That is the mean values are $\lambda =$ for all the methods, $\mu =$ for the S-Lasso and the Fused-Lasso and finally $\mu =$ for the Elastic-Net. *Right*: The tuning parameters are chosen based on the theoretical study. That is the mean values are $\lambda =$ for all the methods, $\mu =$ for the S-Lasso and the Fused-Lasso and finally $\mu =$ for the Elastic-Net.

For the Lasso estimator, we also used the usual rate for λ . Even if the Fused-Lasso seems to be close to the S-Lasso, it turns out that similar choices for the tuning parameters lead to the worst results for the Fused-Lasso.

Based on results on *Examples* (c)-(d) it seems that the Fused-Lasso and the Elastic-Net imply a large bias when the regression vector is smooth for large values of μ (also observed in [9]). They do not improve sufficiently the performance of the Lasso estimator in such situations. Even the ‘corrected’ Elastic-Net does not provide better results since the artificial correction seems to work for a small number of pairs (λ, μ) that have to be chosen very carefully.

4.2 Pseudo-real dataset

We apply all the methods we previously studied on artificially dataset generated from the riboflavin data. These data are about riboflavin (vitamin B2) production by *B. subtilis*. They kindly have been provided to us by DSM Nutritional Products (Switzerland). In the original data, the real-valued response variable is the logarithm of the riboflavin production rate, and there are $p = 4088$ covariates measuring the logarithm of the expression level of 4088 genes that cover essentially the whole genome of *Bacillus subtilis*. The sample size is $n = 71$.

Here we are not interested in the riboflavin production, but only in a covariates matrix X coming from a real application. We use this design matrix to generate an artificial response vector thanks to a ‘smooth’ regression vector as in Equation (1). Let us mention that this trick to generate pseudo-real datasets has already been used in [18]. In what follows, we consider two different applications based on the real covariates matrix provided by the riboflavin dataset. In the first application, says *Application 1*, let us define X as the 1023 first covariates of the riboflavin dataset. Moreover let us define the regression vector β^* , such that $\beta_j^* = 10 \cdot \exp - \frac{1}{1 - ((j-125)/125.1)^2}$ for $j = 1, \dots, 250$ (cf. Figure 8), and the noise level $\sigma = 3$. Hence $n = 71$ and $p = 1023$ and then this is a high-dimensional setting with $p \gg n$, where the number

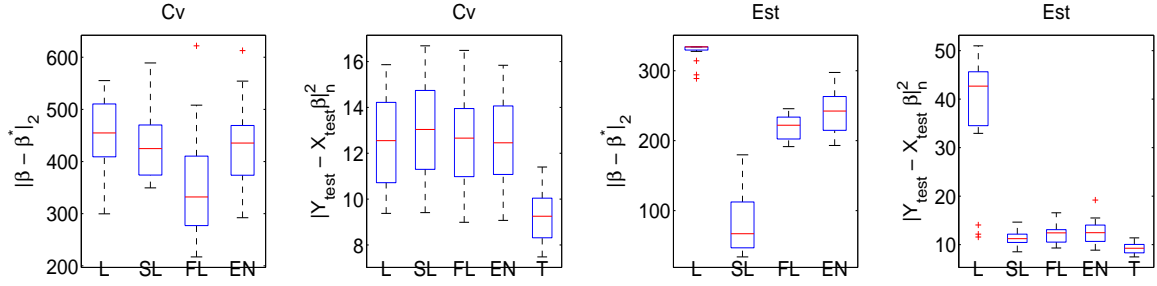


Figure 7: Evaluation of the ℓ_2 estimation error $|\hat{\beta} - \beta^*|_2$ and the prediction error $\|Y_{test} - X_{test}\hat{\beta}\|_n^2$ of the Lasso (L), the S-Lasso (SL), the Fused-Lasso (FL) and the Elastic-Net (EN) applied to *Example (c)* and based on 20 replications of *Application 2*. *Left; Center-left*: The tuning parameters are chosen by 10 fold cross validation. *Center-right; Right*: The tuning parameters minimize the estimation error.

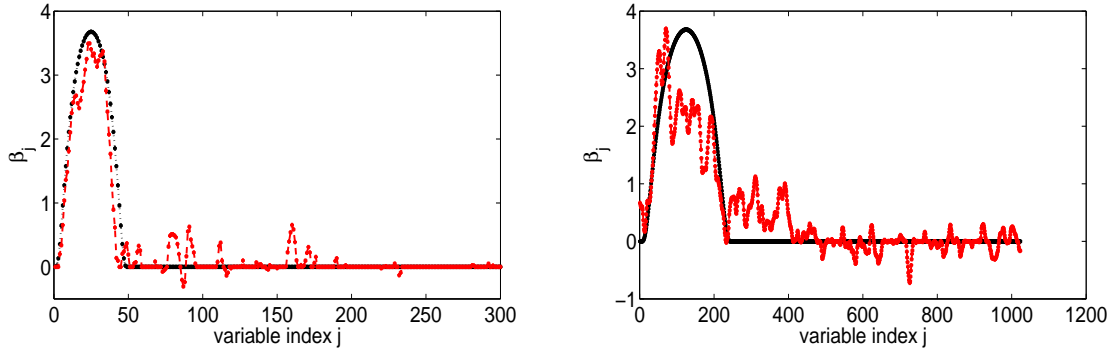


Figure 8: Best reconstitution of the regression vector β^* (black curve) by the SL-Lasso estimator (red curve). *Left*: On *Application 2*. *Right*: On *Application 1*.

of non-zero components (the sparsity index \mathcal{A}^*) is larger than the sample size n . According to the second application, says *Application 2*, we restrict X to the 300 first covariates of the riboflavin dataset. The regression vector β^* is such that $\beta_j^* = 10 \cdot \exp -\frac{1}{1 - ((j-25)/25.1)^2}$ for $j = 1, \dots, 50$ (cf. Figure 8), and the noise level $\sigma = 3$. This is a more usual high-dimensional case where the sparsity index \mathcal{A}^* is smaller than the sample size n .

Let us now detail the obtained results for different experiences. First we mention that, with the exception of the S-Lasso, all the methods provides an estimation of the regression vector which is characterized by large variations in the values of the successive components when μ is small (for the Elastic-Net and the Fused-Lasso), and by large bias when μ is large. Hence we focus here on the S-Lasso estimator. Nevertheless, we display the comparison of all the methods in terms of accuracy in Figures 7 when the methods are applied to *Application 2*. Even though the S-Lasso estimator is outperformed when the tuning parameter is chosen by cross validation (by the Fused-Lasso for the estimation error and by all the methods for the prediction; cf. Figures 7 (left and center-left)), it turns out that we can disclose a S-Lasso solution which performs better than the other methods as displayed in Figures 7 (center-right and right). One of the best solution of the S-Lasso estimator in this *Application 2* can also be seen in Figure 8 (left). We observe how the S-Lasso succeed to reconstruct the ‘smooth’ regression vector β^* .

Finally, let us consider *Application 1*, and let’s recall that the sparsity index is here larger than the sample size. Figure 8 (right) displays the best reconstitution of the regression vector on

this much difficult problem. We observe that the S-Lasso succeed only partly to reconstruct the true regression vector. On the simulation study, we met a similar close situation in *Example (d)* [100/30/3] (cf Figure 5), where the S-Lasso perfectly estimated β^* . However, the situation here is even more difficult since the sparsity index is much larger than the sample size and since many high and negative correlations appear between the covariates in the riboflavin dataset.

5 Conclusion

In this paper, we introduced the Lasso-type estimator $\hat{\beta}^{Quad}$ which consists in two penalty terms: the ℓ_1 penalty which ensures sparsity; and a quadratic penalty which captures some structure in the regression vector. We showed that this estimator satisfies good theoretical performance specifically when the Lasso estimator might fail. As particular cases we considered the Elastic-Net and the S-Lasso. We pointed the interest to use such methods respectively when correlations between variables exist and when the regression vector is ‘smooth’.

In practice, we considered the performance of the S-Lasso estimator compared to the Lasso, the Elastic-Net and the Fused-Lasso in terms of prediction and estimation accuracy. We illustrated the superiority of the S-Lasso in several simulation experiments where the regression vector has a particular structure. We also observed that the theoretical calibration of the tuning parameters provides close performance as when they are chosen by 10 fold cross validation. The methods have also been applied to *pseudo real* examples where based on the riboflavin dataset.

According to some simulation studies (as in *Example (d)* [100/30/ σ]), an interesting point would be whether the S-Lasso satisfies Sparsity Inequalities which can take into account the ‘smoothness’ of the regression vector β^* (if such an assumption is made). This is the topic of future works.

6 Proofs

We first provide a concentration result:

Lemma 2. *Let $0 < \tau < 1$, be a real number. Let $\Lambda_{n,p}$ be the random event defined by $\Lambda_{n,p} = \{\max_{j=1,\dots,p} 2|V_j| \leq \tau\lambda_n\}$ where $V_j = n^{-1} \sum_{i=1}^n x_{i,j}\varepsilon_i$. Let us choose a $\kappa > 2\sqrt{2}/\tau$ and $\lambda_n = \kappa\sigma\sqrt{n^{-1}\log(p)}$. Then*

$$\mathbb{P}\left(\max_{j=1,\dots,p} 2|V_j| \leq \tau\lambda_n\right) \geq 1 - p^{1-\kappa^2\tau^2/8}.$$

Proof. Since $V_j \sim \mathcal{N}(0, n^{-1}\sigma^2)$ for any $j \in \{1, \dots, p\}$, an elementary Gaussian inequality gives

$$\begin{aligned} \mathbb{P}\left(\max_{j=1,\dots,p} |V_j| \geq \tau\lambda_n/2\right) &\leq p \max_{j=1,\dots,p} \mathbb{P}(|V_j| \geq \tau\lambda_n/2) \\ &\leq p \exp\left(-\frac{n}{2\sigma^2} \left(\frac{\tau\lambda_n}{2}\right)^2\right) \\ &= p^{1-\kappa^2\tau^2/8}. \end{aligned}$$

This ends the proof. □

Proof of Theorem 1. We provide a first result which may help the legibility of the paper. For any vector $b \in \mathbb{R}^p$, let $b_{\mathcal{A}}$ be the vector in \mathbb{R}^p such that $(b_{\mathcal{A}})_j = b_j$ if $j \in \mathcal{A}$ and zero otherwise. Then the following proposition states that the squared risk and the ℓ_1 -estimation error are controlled by the restricted ℓ_2 -estimation error $|\beta_{\mathcal{A}^*}^* - \hat{\beta}_{\mathcal{A}^*}^{SL}|_2$.

Proposition 2. *Let $\hat{\beta}^{Quad}$ be the estimator defined by (2)-(4). Let $\lambda_n = \kappa\sigma\sqrt{\frac{\log(p)}{n}}$ and $\mu_n = \frac{\lambda_n}{4|\mathbf{J}\beta^*|_\infty}$, with $A > 2\sqrt{2}$. Let $0 < \tau < 1$ be a real number. On the event $\Lambda_{n,p} = \{\max_{j=1,\dots,p} 2|V_j| \leq \tau\lambda_n\}$ with $V_j = n^{-1} \sum_{i=1}^n x_{i,j}\varepsilon_i$, if $\tau = 1/4$ we have*

$$\frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{SL} \right\|_{n+p}^2 + \frac{\lambda_n}{2} |\beta^* - \hat{\beta}^{SL}|_1 \leq 2\lambda_n \sqrt{|\mathcal{A}^*|} |\beta_{\mathcal{A}^*}^* - \hat{\beta}_{\mathcal{A}^*}^{SL}|_2. \quad (11)$$

Proof. Let first \tilde{X} , \tilde{Y} and $\tilde{\varepsilon}$ be the augmented dataset defined by

$$\tilde{X} = \begin{pmatrix} X \\ \sqrt{n\mu_n}\mathbf{J} \end{pmatrix}, \quad \text{and} \quad \tilde{Y} = \begin{pmatrix} Y \\ \mathbf{0} \end{pmatrix}, \quad \text{and} \quad \tilde{\varepsilon} = \begin{pmatrix} \varepsilon \\ -\sqrt{n\mu_n}\mathbf{J}\beta^* \end{pmatrix},$$

where $\mathbf{0}$ is a vector of size p containing only zeros and \mathbf{J} is the $p \times p$ matrix given by (5). Then we have $\tilde{Y} = \tilde{X}\beta^* + \tilde{\varepsilon}$, and the estimator $\hat{\beta}^{Quad}$, solution of the minimization problem (2) with the penalty given by (4), is also the minimizer of

$$\frac{n+p}{n} \left\| \tilde{Y} - \tilde{X}\beta \right\|_{n+p}^2 + \lambda_n |\beta|_1.$$

Hence, by definition of the estimator $\hat{\beta}^{Quad}$ we can write

$$\begin{aligned} & \frac{n+p}{n} \left\| \tilde{Y} - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 + \lambda_n |\hat{\beta}^{Quad}|_1 \leq \frac{n+p}{n} \left\| \tilde{Y} - \tilde{X}\beta^* \right\|_{n+p}^2 + \lambda_n |\beta^*|_1 \\ \iff & \frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} + \tilde{\varepsilon} \right\|_{n+p}^2 - \frac{n+p}{n} \|\tilde{\varepsilon}\|_{n+p}^2 \leq \lambda_n |\beta^*|_1 - \lambda_n |\hat{\beta}^{Quad}|_1 \\ \iff & \frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 \leq \lambda_n \left[|\beta^*|_1 - |\hat{\beta}^{Quad}|_1 \right] + \frac{2}{n} \tilde{\varepsilon}' \tilde{X} (\beta^* - \hat{\beta}^{Quad}). \end{aligned}$$

Let us now consider the term $\frac{2}{n} \tilde{\varepsilon}' \tilde{X} (\beta^* - \hat{\beta}^{Quad})$. By the definition of \tilde{X} and $\tilde{\varepsilon}$, we have the decomposition $\frac{1}{n} \tilde{\varepsilon}' \tilde{X} (\beta^* - \hat{\beta}^{Quad}) = \frac{1}{n} \varepsilon' X (\beta^* - \hat{\beta}^{Quad}) - \mu_n \beta^{*'} \mathbf{J}' \mathbf{J} (\beta^* - \hat{\beta}^{Quad})$. The first term in this decomposition is quite common in the literature and we treat it using arguments which can be found for instance in [6]. We then need to adapt those arguments in order to deal with the second term of the decomposition $\mu_n \beta^{*'} \mathbf{J}' \mathbf{J} (\beta^* - \hat{\beta}^{Quad})$ in the same time. Recall that $\mathcal{A} = \mathcal{A}^* = \{j : \beta_j^* \neq 0\}$ and that $\mathbf{J}' \mathbf{J} = \tilde{\mathbf{J}}$. Let $0 < \tau < 1$ be a real number. Then, on the event $\Lambda_{n,p} = \{\max_{j=1,\dots,p} 2|V_j| \leq \tau\lambda_n\}$ with $V_j = n^{-1} \sum_{i=1}^n x_{i,j}\varepsilon_i$, we have

$$\begin{aligned} \frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 & \leq \lambda_n \left[|\beta^*|_1 - |\hat{\beta}^{Quad}|_1 \right] + \tau\lambda_n |\beta^* - \hat{\beta}^{Quad}|_1 \\ & \quad - \mu_n \beta^{*'} \tilde{\mathbf{J}} (\beta^* - \hat{\beta}^{Quad}). \end{aligned} \quad (12)$$

The remainder of this prove is linked to the way we choose to treat the term $\mu_n \beta^{*'} \tilde{\mathbf{J}} (\beta^* - \hat{\beta}^{Quad})$ and in particular in the way we choose to link the RHS of Inequality (12) to the quantity $|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2$, where \mathcal{A} is the true sparsity set. Note that

$$-\mu_n \beta^{*'} \tilde{\mathbf{J}} (\beta^* - \hat{\beta}^{Quad}) \leq \mu_n |\tilde{\mathbf{J}} \beta^*|_\infty |\beta^* - \hat{\beta}^{Quad}|_1.$$

Then, if we set $\tau = \frac{1}{4}$ and the tuning parameter $\mu_n = \frac{\lambda_n}{4|\tilde{J}\beta^*|_\infty}$ Inequality (12) becomes

$$\frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 \leq \lambda_n \left[|\beta^*|_1 - |\hat{\beta}^{Quad}|_1 \right] + \frac{\lambda_n}{2} |\beta^* - \hat{\beta}^{Quad}|_1.$$

Add $2^{-1}\lambda_n|\beta^* - \hat{\beta}^{Quad}|_1$ to both sides of the previous inequality and then thanks to the fact that $|\beta_j^* - \hat{\beta}_j^{Quad}| + |\beta_j^*| - |\hat{\beta}_j^{Quad}| = 0$ for any $j \notin \mathcal{A}$ and to the triangular inequality, the above inequality implies that

$$\frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 + \frac{\lambda_n}{2} |\beta^* - \hat{\beta}^{Quad}|_1 \leq 2\lambda_n \sqrt{|\mathcal{A}|} |\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2.$$

□

Let us now proof the main theorem. Thank to Inequality (11) in Proposition 2, we easily obtain that

$$|\beta^* - \hat{\beta}^{Quad}|_1 \leq 4\sqrt{|\mathcal{A}|} |\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2. \quad (13)$$

and the vector $\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}$ is an admissible vector Δ in Assumption $B(\mathcal{A})$. As a consequence, using this assumption in Equation (11), we get on one hand

$$\frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 \leq \frac{2\lambda_n \sqrt{|\mathcal{A}|}}{\sqrt{\phi}} \sqrt{\frac{n+p}{n}} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p},$$

and then, thanks to the trivial inequality $2\alpha\beta \leq \alpha^2/4 + 4\beta^2$ (for $\alpha, \beta \in \mathbb{R}$), we obtain

$$\frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 \leq \frac{16}{3\phi} \lambda_n^2 |\mathcal{A}|. \quad (14)$$

This provides the first part of the result. On the other hand, Inequality (13), combined to Assumption $B(\mathcal{A})$ and Inequality (14), implies that

$$|\beta^* - \hat{\beta}^{Quad}|_1 \leq \frac{16}{\sqrt{3}\phi} \lambda_n |\mathcal{A}|, \quad (15)$$

which is the desired bound on the ℓ_1 estimation error given in Theorem 1. The proof is completed when we use Lemma 2 with $\tau = 1/4$ to control the probability of the event $\Lambda_{n,p}$. □

Proof of Theorem 2. We consider now the case where \tilde{J} is very sparse. The S-Lasso and the Elastic-Net can be considered as a special case. Most of the proof is inspired from the one of Theorem 1. Then a similar reasoning leads to (12) and the only different occurs when we deal with the term $-\mu_n \beta^{*'} \tilde{J}(\beta^* - \hat{\beta}^{Quad})$. We obviously can write

$$-\mu_n \beta^{*'} \tilde{J}(\beta^* - \hat{\beta}^{Quad}) = -\mu_n \beta_{\mathcal{B}}^{*'} \tilde{J}(\beta_{\mathcal{B}}^* - \hat{\beta}_{\mathcal{B}}^{Quad}) \leq \mu_n |\tilde{J}\beta^*|_2 |\beta_{\mathcal{B}}^* - \hat{\beta}_{\mathcal{B}}^{Quad}|_2,$$

and we recall that the set \mathcal{B} includes \mathcal{A} , the true sparsity set and is not much larger. In this case we defined μ_n by $\mu_n = \frac{\lambda_n \sqrt{|\mathcal{A}^*|}}{|\tilde{J}\beta^*|_2}$. Let also $\tau = 1/2$ such that (12) implies

$$\begin{aligned} \frac{n+p}{n} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}^2 + \frac{\lambda_n}{2} |\beta^* - \hat{\beta}^{Quad}|_1 &\leq \lambda_n \sum_{j \in \mathcal{A}} |\beta_j^* - \hat{\beta}_j^{Quad}| \\ &\quad + \mu_n |\tilde{J}\beta^*|_2 |\beta_{\mathcal{B}}^* - \hat{\beta}_{\mathcal{B}}^{Quad}|_2 \\ &\leq \tilde{\tau}_n |\beta_{\mathcal{B}}^* - \hat{\beta}_{\mathcal{B}}^{Quad}|_2, \end{aligned}$$

where $\tilde{\tau}_n = \lambda_n \sqrt{|\mathcal{A}|} + \mu_n |\tilde{J}\beta^*|_2$, since $|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_1 \leq \sqrt{|\mathcal{A}|} |\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2 \leq \sqrt{|\mathcal{A}|} |\beta_{\mathcal{B}}^* - \hat{\beta}_{\mathcal{B}}^{Quad}|_2$. This above intermediate result is the analogous of Proposition 2 in the case where \tilde{J} is sparse. In particular, if we choose μ_n equal to $\frac{\lambda_n \sqrt{|\mathcal{A}|}}{|\tilde{J}\beta^*|_2}$, the quantity $\tilde{\tau}_n$ becomes equal to $2\lambda_n \sqrt{|\mathcal{A}|}$, and we get a similar bound but depending on $|\beta_{\mathcal{B}}^* - \hat{\beta}_{\mathcal{B}}^{Quad}|_2$ instead of $|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2$. Taking into account this changing, we use a similar reasoning as in the proof of Theorem 1 and get the desired result. \square

Proof of Proposition 1. Recall the short notation $\mathcal{A} = \mathcal{A}^*$. Theorem 1 states a bounds on the prediction error and on the ℓ_1 estimation error under Assumption $B(\mathcal{A})$. Thanks to (13) we can use Assumption $B(\mathcal{A})$, which directly implies that the following inequality holds $|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2 \leq \sqrt{\phi^{-1}} \sqrt{\frac{n+p}{n}} \left\| \tilde{X}\beta^* - \tilde{X}\hat{\beta}^{Quad} \right\|_{n+p}$. Combining this inequality with (14), we easily get

$$|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_2 \leq \frac{4}{\sqrt{3}\phi} \lambda_n \sqrt{|\mathcal{A}|}, \quad (16)$$

and this completes the proof of the first part of the Proposition. We now show that $\mathcal{A} = \mathcal{A}^* \subset \hat{\mathcal{A}}$ with high probability. Thanks to (16), we have with high probability $|\beta_{\mathcal{A}}^* - \hat{\beta}_{\mathcal{A}}^{Quad}|_{\infty} \leq U$ where we used $U = \frac{4}{\sqrt{3}\phi} \lambda_n \sqrt{|\mathcal{A}|}$ for short. But

$$|\hat{\beta}_{\mathcal{A}}^{Quad} - \beta_{\mathcal{A}}^*|_{\infty} \leq U \quad \Leftrightarrow \quad \beta_j^* - U \leq \hat{\beta}_j^{Quad} \leq \beta_j^* + U \quad \forall j \in \mathcal{A}.$$

Note that by Assumption C , we have $|\beta_j^*| > U, \forall j \in \mathcal{A}$. Then if we distinguish the case $\beta_j^* > 0$ and the case $\beta_j^* < 0$, we easily conclude that $\beta_j^* > 0$ implies $\hat{\beta}_j^{Quad} > 0$ and $\beta_j^* < 0$ implies $\hat{\beta}_j^{Quad} < 0$. This ables us to write

$$\mathbb{P}(\text{Sgn}(\hat{\beta}_{\mathcal{A}}^{Quad}) = \text{Sgn}(\beta_{\mathcal{A}}^*)) \leq \mathbb{P}(|\hat{\beta}_{\mathcal{A}}^{Quad} - \beta_{\mathcal{A}}^*|_{\infty} \leq U) \leq p^{1-\kappa^2/128},$$

and this naturally implies the that $\mathcal{A} \subset \hat{\mathcal{A}}$ with high probability. \square

Proof of Theorem 3. We now show that $\hat{\mathcal{A}} \subset \mathcal{A}^*$ with high probability. This proof is quite inspired by the one by Bunea [4]. First of all, note that we can write the KKT conditions of the minimization problem (6) as

$$|K_n(\hat{\beta}^{Quad} - \beta^*) - \frac{X'\varepsilon}{n} + \mu_n \tilde{J}\beta^*|_{\infty} \leq \frac{\lambda_n}{2}. \quad (17)$$

Then all the solutions of the criterion (6) share the same active set

$$\hat{\mathcal{A}} = \left\{ j \in \{1, \dots, p\} : |(K_n(\hat{\beta}^{Quad} - \beta^*))_j - \frac{X'_j \varepsilon}{n} + \mu_n(\tilde{J}\beta^*)_j| = \frac{\lambda_n}{2} \right\}.$$

That is, all these solutions have non-zero components at the same positions. We now use this property to show that the estimator $\hat{\beta}^{Quad}$ has non-zero components at the same positions as a well-controlled (but uncomputable) estimator on an event which occurs with high probability. For this purpose, let us consider the criterion

$$F(b) = \|Y - \sum_{j \in \mathcal{A}^*} X_j b_j\|_n^2 + \lambda_n \sum_{j \in \mathcal{A}^*} |b_j| + \mu_n b'_{\mathcal{A}^*} \mathbf{J}'_{\mathcal{A}^*} \mathbf{J}_{\mathcal{A}^*} b_{\mathcal{A}^*},$$

where recall that for any p -dimensional vector a and any set $\Theta \subset \{1, \dots, p\}$, the notation a_Θ means that $(a_\Theta)_j = a_j, \forall j \in \Theta$ and 0 otherwise. Moreover, $\mathbf{J}_{\mathcal{A}^*}$ is such that $(\mathbf{J}_{\mathcal{A}^*})_{j,k} = \mathbf{J}_{j,k}$ if $j, k \in (\mathcal{A}^*)^2$ and 0 otherwise. For now on let us note \mathcal{A} for \mathcal{A}^* for short. Define the estimator

$$\hat{b} = \underset{b \in \mathbb{R}^p : b_{\mathcal{A}^c} = \mathbf{0}_p}{\text{Argmin}} F(b),$$

where $\mathbf{0}_p$ is the zero in \mathbb{R}^p . Since we restricted \hat{b} to be zero when β^* is zero and that this is an information we do not have access to, we mention that the vector is not computable. Let us denote by Γ the following event

$$\Gamma = \bigcap_{k \notin \mathcal{A}} \left\{ \left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) - \frac{X'_k \varepsilon}{n} + \mu_n \sum_{j \in \mathcal{A}} \tilde{J}_{j,k} \beta_j^* \right| < \frac{\lambda_n}{2} \right\}.$$

Observe how the event Γ is inspired by the KKT conditions (17). Actually, on the event Γ , the components \hat{b}_k with $k \notin \mathcal{A}$ equals zero as they do not saturate KKT conditions. This makes the minimization of $F(b)$ over $b \in \mathbb{R}^p : b_{\mathcal{A}^c} = \mathbf{0}_p$ coincide with the minimization of the criterion (6) on Γ . That is, the estimator \hat{b} turns out to be also solution of the original criterion (6) on Γ . But $\hat{\beta}^{Quad}$ is also solution of (6) and then, as we already pointed, this implies that on Γ , both of $\hat{\beta}^{Quad}$ and \hat{b} have non-zero components at the same positions and then, \hat{b} has non-zero components at components $j \in \hat{\mathcal{A}}$. Add the fact that by construction $\hat{b}_{\mathcal{A}^c}$, then $\hat{\mathcal{A}} \subset \mathcal{A}$ on the event Γ . It then remains to prove that the event Γ occurs with high probability. We have

$$\begin{aligned} \mathbf{P}(\hat{\mathcal{A}} \not\subset \mathcal{A}) &\leq \mathbf{P}(\Gamma^c) \\ &\leq \sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) - \frac{X'_k \varepsilon}{n} + \mu_n \sum_{j \in \mathcal{A}} \tilde{J}_{j,k} \beta_j^* \right| \geq \frac{\lambda_n}{2} \right) \\ &\leq \sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) - \frac{X'_k \varepsilon}{n} \right| \geq \frac{\lambda_n}{2} - \mu_n |\tilde{J}\beta^*|_\infty \right) \\ &\leq \sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) - \frac{X'_k \varepsilon}{n} \right| \geq \frac{\lambda_n}{4} \right) \\ &\leq \sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) \right| \geq \frac{\lambda_n}{8} \right) + \sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \frac{X'_k \varepsilon}{n} \right| \geq \frac{\lambda_n}{8} \right) \quad (18) \end{aligned}$$

where we used the fact that for real number a and b , we have $|a| + |b| \geq |a + b|$ in the third inequality and the fact that $\mu_n = \frac{\lambda_n}{4|J\beta^*|_\infty}$ in the forth one. Let us consider the last two terms in the last display separately. i) First, thanks to Lemma 2 with $\tau = 4^{-1}$, we obtain $\sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \frac{X'_k \varepsilon}{n} \right| \geq \frac{\lambda_n}{8} \right) \leq p^{1-\kappa^2/128}$. This imposes that the parameter κ have to be chosen larger than $8\sqrt{2}$; ii) according to $\sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) \right| \geq \frac{\lambda_n}{8} \right)$, we need to control $\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) \right|$. On one hand, Assumption B2 implies that

$$\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) \right| \leq \sum_{j \in \mathcal{A}} |\hat{b}_j - \beta_j^*| t / |\mathcal{A}|. \quad (19)$$

By definition of \hat{b} , we just have to repeat the proof of Theorem 1 but with \hat{b} instead of $\hat{\beta}^{Quad}$ and only on the true sparsity set \mathcal{A} . We get that on the event $\Lambda_{n,\mathcal{A}} = \left\{ \max_{j \in \mathcal{A}} |X'_j \varepsilon| \leq \lambda_n/8 \right\}$, which is the same that $\Lambda_{n,p}$ but using \mathcal{A} instead of $\{1, \dots, p\}$,

$$\sum_{j \in \mathcal{A}} |\hat{b}_j - \beta_j^*| \leq \frac{16}{\sqrt{3}\phi} \lambda_n |\mathcal{A}|.$$

Moreover, similar reasoning as in Lemma 2 leads to $\mathbf{P} \left(\Lambda_{n,\mathcal{A}}^c \right) \leq |\mathcal{A}| p^{-\kappa^2/8} \leq p^{1-\kappa^2/8}$. Combine this result with (19) and to get

$$\begin{aligned} \sum_{k \notin \mathcal{A}} \mathbf{P} \left(\left| \sum_{j \in \mathcal{A}} (K_n)_{j,k} (\hat{b}_j - \beta_j^*) \right| \geq \frac{\lambda_n}{8} \right) &\leq p \mathbf{P} \left(\sum_{j \in \mathcal{A}} |\hat{b}_j - \beta_j^*| \geq \frac{|\mathcal{A}| \lambda_n}{8t} \right) \\ &\leq p \mathbf{P} \left(\sum_{j \in \mathcal{A}} |\hat{b}_j - \beta_j^*| \geq \frac{16}{\sqrt{3}\phi} \lambda_n |\mathcal{A}| \right) \\ &\leq p \mathbf{P} \left(\Lambda_{n,\mathcal{A}}^c \right) \leq p^{2-\kappa^2/128}, \end{aligned}$$

provided that $t \leq \frac{\sqrt{3}\phi}{128}$. We finally conclude by this last inequality, (18) that $\mathbf{P}(\hat{\mathcal{A}} \not\subseteq \mathcal{A}) \leq p^{2-\kappa^2/128} + p^{1-\kappa^2/128} \leq 2p^{2-\kappa^2/128}$. Note that with our choice of $\lambda_n = \kappa\sigma\sqrt{\log(p)/n}$ with $\kappa \geq 16\sqrt{2}$, we guaranty that this probability goes to zero exponentially fast and as a consequence, we get the desired result. \square

References

- [1] F. Bach. Consistency of the group lasso and multiple kernel learning. *J. Mach. Learn. Res.*, 9:1179–1225, 2008.
- [2] P. Bickel, Y. Ritov, and A. Tsybakov. Simultaneous analysis of lasso and Dantzig selector. *Ann. Statist.*, 37(4):1705–1732, 2009.
- [3] F. Bunea. Consistent selection via the lasso for high dimensional approximating regression models. 2008. IMS Collections, B. Clarke and S. Ghosal Editors.
- [4] F. Bunea. Honest variable selection in linear and logistic regression models via ℓ_1 and $\ell_1 + \ell_2$ penalization. *Electron. J. Stat.*, 2:1153–1194, 2008.

- [5] F. Bunea, A. Tsybakov, and M. Wegkamp. Aggregation for Gaussian regression. *Ann. Statist.*, 35(4):1674–1697, 2007.
- [6] F. Bunea, A. B. Tsybakov, and M. H. Wegkamp. Sparsity oracle inequalities for the Lasso. *Electron. J. Stat.*, 1:169–194, 2007.
- [7] C. Chesneau and M. Hebiri. Some theoretical results on the grouped variables lasso. *Math. Methods Statist.*, 17(4):317–326, 2008.
- [8] A. S. Dalalyan and A. B. Tsybakov. Aggregation by exponential weighting and sharp oracle inequalities. In *Learning theory*, volume 4539 of *Lecture Notes in Comput. Sci.*, pages 97–111. Springer, Berlin, 2007.
- [9] Z. John Daye and X. Jessie Jeng. Shrinkage and model selection with correlated variables via weighted fusion. *Computational Statistics & Data Analysis*, 53(4):1284–1298, 2009.
- [10] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Ann. Statist.*, 32(2):407–499, 2004. With discussion, and a rejoinder by the authors.
- [11] J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Amer. Statist. Assoc.*, 96(456):1348–1360, 2001.
- [12] M. Hebiri. Regularization with the smooth-lasso procedure. Preprint Laboratoire de Probabilités et Modèles Aléatoires, 2008.
- [13] S. R. Land and J. H. Friedman. Variable fusion: a new method of adaptive signal regression. *Manuscript*, 1996.
- [14] K. Lounici. Sup-norm convergence rate and sign concentration property of Lasso and Dantzig estimators. *Electron. J. Stat.*, 2:90–102, 2008.
- [15] L. Meier, S. van de Geer, and P. Bühlmann. The group Lasso for logistic regression. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 70(1):53–71, 2008.
- [16] N. Meinshausen. Relaxed Lasso. *Comput. Statist. Data Anal.*, 52(1):374–393, 2007.
- [17] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *Ann. Statist.*, 34(3):1436–1462, 2006.
- [18] N. Meinshausen, L. Meier, and P. Bühlmann. p-values for high-dimensional regression. *J. Amer. Statist. Assoc.*, 104:1671–1681, 2009.
- [19] Yu. Nesterov. Gradient methods for minimizing composite objective function. CORE Discussion Papers 2007076, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), Sep 2007.
- [20] A. Rinaldo. Properties and refinements of the fused lasso. *Ann. Statist.*, 37(5B):2922–2952, 2009.
- [21] S. Rosset and J. Zhu. Piecewise linear regularized solution paths. *Ann. Statist.*, 35(3):1012–1030, 2007.

- [22] R. Tibshirani. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1):267–288, 1996.
- [23] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight. Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 67(1):91–108, 2005.
- [24] A. B. Tsybakov and S. A. van de Geer. Square root penalty: adaptation to the margin in classification and in edge estimation. *Ann. Statist.*, 33(3):1203–1224, 2005.
- [25] S. van de Geer and P. Bühlmann. On the conditions used to prove oracle results for the lasso. *Elect. Journ. Statist.*, 3:1360–1392, 2009.
- [26] M. Wainwright. Sharp thresholds for noisy and high-dimensional recovery of sparsity using ℓ_1 -constrained quadratic programming. Manuscript, 2006.
- [27] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(1):49–67, 2006.
- [28] M. Yuan and Y. Lin. On the non-negative garrote estimator. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 69(2):143–161, 2007.
- [29] P. Zhao and B. Yu. On model selection consistency of Lasso. *J. Mach. Learn. Res.*, 7:2541–2563, 2006.
- [30] H. Zou. The adaptive lasso and its oracle properties. *J. Amer. Statist. Assoc.*, 101(476):1418–1429, 2006.
- [31] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 67(2):301–320, 2005.