Accelerating Non-Negative and Bounded-Variable Linear Regression Algorithms with Safe Screening

Cassio F. Dantas, Emmanuel Soubies, Cédric Févotte

Accelerating Non-Negative and Bounded-Variable Linear Regression Algorithms with Safe Screening

Cassio F. Dantas[1 [0000−0002−1934−0625]], Emmanuel Soubies[2 [0000−0003−0571−6983]], and Cédric Févotte[2 [0000−0003−3801−5534]]

[1] INRAE, TETIS, Univ Montpellier, France. cassio.fraga-dantas@inrae.fr
[2] CNRS, IRIT, University of Toulouse, France

Abstract. Non-negative and bounded-variable linear regression problems arise in a variety of applications in machine learning and signal processing. In this paper, we propose a technique to accelerate existing solvers for these problems by identifying saturated coordinates in the course of iterations. This is akin to safe screening techniques previously proposed for sparsity-regularized regression problems. The proposed strategy is *provably safe* as it provides theoretical guarantees that the identified coordinates are indeed saturated in the optimal solution. Experimental results on synthetic and real data show compelling accelerations for both non-negative and bounded-variable problems.

Keywords: Safe screening · linear regression · non-negative · bounded-variable.

1 Introduction

Due to their fundamental importance in many fields, box-constrained linear regression problems—including in the large sense the popular non-negative least squares (NNLS) problem—have received considerable attention for several decades [16,22,6]. They are in particular relevant for under-determined linear regression, when the number of variables is larger than the number of measurements. For example, it has been shown that for a variety of problems, the sole non-negativity constraint can be as efficient as sparsity-based regularization [21],[3, Proposition 4.1].

Several algorithms have been proposed to address such regression problems. Let us in particular mention the seminal active set approach of [16] for NNLS, which generalizes to bounded-variable least squares (BVLS) [22]. Accelerated variants were proposed by [4,23]. Other methods for NNLS include projected gradient and Newton methods [15,19], coordinate descent algorithms [11], or interior point approaches [1]. They come with different strengths and limitations and we refer the reader to [6] for a comparative discussion.

⋆ Code is available at: https://github.com/cassiofragadantas/NN-BV_Screening

Contributions and outline In this work, we propose a generic strategy to accelerate existing solvers for box-constrained linear regression. It relies on the early identification of saturated coordinates (those attaining the box limits in the solution vector) during the course of iterations, akin to safe screening techniques for sparse-regularized regression problems [18,9]. The general optimization problem addressed in the paper is formulated in Section 2. It takes as a special case the *safe squeezing* approach of [10] for ℓ_∞-regularized problems (see the details in Appendix A). In particular, our framework handles non-symmetric bounds (including an infinite upper bound for the nonnegative case) and can deal with a broad class of data-fidelity functions beyond least squares. Then, we show in Section 3 how some saturated coordinates in the solution vector can be *safely* identified from a primal-dual feasible pair of vectors. This allows us to derive a generic dynamic safe screening procedure for box-constrained linear regression problems. In Section 4, we focus on the dual update step. Although a standard dual-scaling can be deployed for the bounded-variable case, it turns out to be ineffective for the non-negative case. As such, we introduce a new dual update strategy, termed as dual translation, to compute relevant feasible dual points. Finally, numerical experiments are reported in Section 5. They demonstrate how our proposed approach can significantly accelerate various solvers from the literature.

Notations For $n \in \mathbb{N}$, we denote by $[n]$ the set $\{1, \dots, n\}$. The ith entry of a vector $\mathbf{z} \in \mathbb{R}^n$ is denoted z_i (or sometimes $[\mathbf{z}]_i$ to avoid ambiguities). Given a subset of indices $\mathcal{S} \subseteq [n]$ with cardinality $|\mathcal{S}| = s$, $\mathbf{z}_\mathcal{S} \in \mathbb{R}^s$ denotes the restriction of $\mathbf{z}$ to its entries indexed by the elements of $\mathcal{S}$. For a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{a}_j \in \mathbb{R}^m$ stands for its jth column and $\mathbf{A}_\mathcal{S} \in \mathbb{R}^{m \times s}$ is the matrix formed out of the columns of $\mathbf{A}$ indexed by the set $\mathcal{S}$. Vector inequalities are taken coordinate-wisely, i.e., $\mathbf{a} \leq \mathbf{b}$ means that $a_i \leq b_i, \forall i$. Given two vectors $\mathbf{l} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^n$, we define the set $[\mathbf{l}, \mathbf{u}] = [l_1, u_1] \times \cdots \times [l_n, u_n]$. We use the notation $[\mathbf{z}]^+$ (resp., $[\mathbf{z}]^-$) to refer to the positive (resp., negative) part operation defined as $\max(0, z_i)$ (resp., $\min(0, z_i)$) for all $i \in [n]$. For a convex function $f : \mathbb{R}^n \to (-\infty, +\infty]$, $f^*(\mathbf{u}) = \sup_{\mathbf{z} \in \mathbb{R}^n} \langle \mathbf{z}, \mathbf{u} \rangle - f(\mathbf{z})$ denotes its Fenchel-Legendre transform (or conjugate function).

2 Box-constrained linear regression

Given a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and a data vector $\mathbf{y} \in \mathbb{R}^m$, we consider the generic box-constrained linear regression problem

$$\mathbf{x}^\star \in \operatorname*{argmin}_{\mathbf{x} \in \mathbb{R}^n} \; P(\mathbf{x}) := \sum_{i=1}^{m} f([\mathbf{A}\mathbf{x}]_i; y_i) \qquad (1)$$
$$\text{s.t. } \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}$$

where $\mathbf{l} \in \mathbb{R}^n$ and $\mathbf{u} \in \bar{\mathbb{R}}^n$ with $\bar{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ the Alexandroff extension of the real number line. Given y, the function $f(z; y)$ is assumed proper, lower semi-continuous, convex, and differentiable with respect to (w.r.t.) z. We also assume

it has Lipschitz gradient with constant $1/\alpha$. We define $F(\mathbf{z};\mathbf{y}) = \sum_i f(z_i;y_i)$, referred to as *loss function*. In the rest of the paper, it will be treated as a function of $\mathbf{z}$ only. Problem (1) encompasses both bounded-variable linear regression (BVLR) when $\mathbf{u} \in \mathbb{R}^n$ and non-negative linear regression (NNLR)[3] when $l_i = 0$ and $u_i = \infty\ \forall i \in [n]$. Moreover, our framework can also account for mixed constraints where some entries of $\mathbf{u}$ are finite and the others are infinite.

3 Early identification of saturated variables

Building upon the Gap safe screening technique [18] for sparse linear regression, we propose a generic approach (Algorithm 1) to accelerate solvers for (1) through the early identification of saturated coordinates (i.e., identification of $j \in [n]$ such that $x_j^\star = l_j$ or $x_j^\star = u_j$ when $u_j < \infty$).

3.1 Dual problem

Let $\mathcal{J}_\infty^u := \{j \in [n] : u_j = \infty\}$ denotes the set of indices for which the upper bound constraint in (1) is infinity. Then, the dual problem of (1) reads

$$\boldsymbol{\theta}^\star = \underset{\boldsymbol{\theta} \in \mathcal{F}_D}{\operatorname{argmax}}\ D(\boldsymbol{\theta}) \tag{2}$$

where the dual objective function and the dual feasible set are given by (see Appendix B)

$$D(\boldsymbol{\theta}) = -\sum_{i=1}^m f^*(-\theta_i;y_i) - \sum_{j=1}^n l_j[\mathbf{A}^\mathsf{T}\boldsymbol{\theta}]_j^- - \sum_{j\in[n]\setminus\mathcal{J}_\infty^u} u_j[\mathbf{A}^\mathsf{T}\boldsymbol{\theta}]_j^+, \tag{3}$$

$$\mathcal{F}_D = \{\boldsymbol{\theta} \in \mathbb{R}^m : \forall j \in \mathcal{J}_\infty^u,\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta} \le 0\}. \tag{4}$$

Note that the dual solution $\boldsymbol{\theta}^\star$ is unique (thanks to the differentiability of f).

From (3) and (4), we see that when $\mathbf{u} \in \mathbb{R}^n$ (BVLR), the dual problem is unconstrained, that is $\mathcal{F}_D = \mathbb{R}^m$. In contrast, when $\mathbf{l} = \mathbf{0}$ and all the entries of $\mathbf{u}$ are equal to ∞ (NNLR), the dual cost function simplifies as $D(\boldsymbol{\theta}) = -\sum_{i=1}^m f_i^*(-\theta_i;y_i)$ and the dual feasible set reads $\mathcal{F}_D = \{\boldsymbol{\theta} \in \mathbb{R}^m : \mathbf{A}^\mathsf{T}\boldsymbol{\theta} \le 0\}$.

Remark 1. Note that if $\mathbf{u} = -\mathbf{l} = \delta\mathbf{1}_m$, the second and third terms of (3) sum up to $\delta\|\mathbf{A}^\mathsf{T}\boldsymbol{\theta}\|_1$ and we obtain a form of Lasso problem. More generally, the dual problem of BVLR can be seen as a further generalization of the Lasso, where the penalization on the vector $\mathbf{A}^\mathsf{T}\boldsymbol{\theta} \in \mathbb{R}^n$ is weighted by $\mathbf{l}$ (resp., $\mathbf{u}$) for its negative (resp., positive) entries.

[3] We use the abbreviation LR (for linear regression) to emphasize that our framework is more general than least-squares (LS) regression and can deal with a broader class of functions f than the quadratic distance used in the LS case.

3.2 Safe identification of saturated variables

From the primal (1) and dual (2) problems, the first-order primal-dual optimality conditions for a primal-dual solution pair $(\mathbf{x}^\star, \boldsymbol{\theta}^\star) \in [\mathbf{l}, \mathbf{u}] \times \mathcal{F}_D$ are given by (see Appendix C):

$$\forall i \in [m],\ \theta_i^\star = -f'([\mathbf{A}\mathbf{x}^\star]_i; y_i), \tag{5}$$

$$\forall j \in [n],\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta}^\star \in \begin{cases} (-\infty, 0] \text{ if } x_j^\star = l_j \\ [0, +\infty) \text{ if } x_j^\star = u_j \\ \{0\} \quad \text{if } x_j^\star \in (l_j, u_j) \end{cases}. \tag{6}$$

Let us emphasize that, for NNLR, the second case of (6) never occurs (as $x_j^\star < \infty = u_j$).

Upon knowledge of the dual solution $\boldsymbol{\theta}^\star$, the optimality condition (6) (known as sub-differential inclusion) constitutes a natural criterion to identify saturated coordinates of the primal solution $\mathbf{x}^\star$. More precisely, we have

$$\begin{aligned} \forall j \in [n],\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta}^\star < 0 &\implies x_j^\star = l_j, \\ \forall j \in [n]\backslash\mathcal{J}_\infty^u,\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta}^\star > 0 &\implies x_j^\star = u_j. \end{aligned} \tag{7}$$

Although this criterion cannot be used in practice as the dual solution $\boldsymbol{\theta}^\star$ is not known in advance, a relaxed version of it can be obtained with only a partial knowledge of the location of $\boldsymbol{\theta}^\star$. More precisely, upon knowledge of a region $\mathcal{R} \subset \mathbb{R}^m$ such that $\boldsymbol{\theta}^\star \in \mathcal{R}$—referred to as *safe region*—we can define a weaker version of (7) as

$$\begin{aligned} \forall j \in [n],\ \max_{\boldsymbol{\theta}' \in \mathcal{R}} \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta}' < 0 &\implies x_j^\star = l_j, \\ \forall j \in [n]\backslash\mathcal{J}_\infty^u,\ \min_{\boldsymbol{\theta}' \in \mathcal{R}} \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta}' > 0 &\implies x_j^\star = u_j. \end{aligned} \tag{8}$$

Clearly, the smaller the region $\mathcal{R} \ni \boldsymbol{\theta}^\star$, the larger the number of saturated variables that can be identified. A convenient and efficient choice for $\mathcal{R}$ is presented in the next section.

3.3 The Gap safe sphere

The Gap safe sphere—initially proposed by [18] in the context of sparse linear regression—is defined for any primal-dual pair $(\mathbf{x}, \boldsymbol{\theta}) \in [\mathbf{l}, \mathbf{u}] \times \mathcal{F}_D$ by

$$\mathcal{B}(\boldsymbol{\theta}, r), \text{ with } r = \sqrt{\frac{2\,\mathrm{Gap}(\mathbf{x}, \boldsymbol{\theta})}{\alpha}} \tag{9}$$

where the duality gap is given by

$$\mathrm{Gap}(\mathbf{x}, \boldsymbol{\theta}) = P(\mathbf{x}) - D(\boldsymbol{\theta}). \tag{10}$$

Let us recall that α in (9) is the inverse of the Lipschitz constant of the gradient of f. Equivalently, this means that $-f^\star$ (and hence D) is α-strongly concave. This safe region leads to state-of-the-art screening performances due, in particular, to two key properties.

– A simple geometry which allows to simplify (8) as

$$\begin{array}{rl} \forall j \in [n],\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta} < -r\|\mathbf{a}_j\|_2 & \implies x_j^\star = l_j, \\ \forall j \in [n]\backslash\mathcal{J}_\infty^u,\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta} > \ \ r\|\mathbf{a}_j\|_2 & \implies x_j^\star = u_j. \end{array} \tag{11}$$

  This significantly limits the computational overhead that is introduced when testing the validity of this criterion (lines 11 and 12 in Algorithm 1).
– It vanishes when the duality gap tends to zero. In other words, if strong duality holds (i.e., $P(\mathbf{x}^\star) - D(\boldsymbol{\theta}^\star) = 0$), then the radius of the Gap safe sphere vanishes as the iterates $\{\mathbf{x}^k, \boldsymbol{\theta}^k\}$ converge to $\{\mathbf{x}^\star, \boldsymbol{\theta}^\star\}$.

A proof that $\mathcal{B}(\boldsymbol{\theta}, r)$ is indeed a safe region (i.e., that $\boldsymbol{\theta}^\star \in \mathcal{B}(\boldsymbol{\theta}, r)$) was provided by [18, Theorem 6] in the context of sparse linear regression. Actually, this proof can be directly applied to the linear regression problems with box constraints considered in the present paper.

Remark 2. Note that the Gap safe sphere can also be defined using *local* strong concavity bounds of D computed on well-chosen subsets of the domain [9, Theorem 5]. This would allow to extend the applicability of Algorithm 1 (following [9]) to a more general class of functions f such as the β-divergences with $\beta \in [1, 2)$, that includes in particular the popular Kullback-Leibler divergence [7].

3.4 Resulting screening algorithm

The proposed safe screening approach for Problem (1) is presented in Algorithm 1. It can be deployed with any iterative solver for (1), as indicated by the generic notation

$$\{\mathbf{x}, \boldsymbol{\eta}\} \leftarrow \texttt{PrimalUpdate}(F(\mathbf{A}\cdot + \mathbf{z}; \mathbf{y}); \mathbf{x}, \boldsymbol{\eta}).$$

This has to be understood as performing few iterations of a given primal solver on $F(\mathbf{A}\cdot + \mathbf{z}; \mathbf{y})$ from the initial point $\mathbf{x}$. The vector $\boldsymbol{\eta}$ contains the hyperparameters of the solver (e.g., step sizes) and we will make the role of $\mathbf{z} \in \mathbb{R}^m$ explicit hereafter.

In Algorithm 1, $\Theta : \mathbb{R}^n \to \mathcal{F}_D$ denotes a function that computes a dual feasible point from the current primal point $\mathbf{x}$. Details on how to define this function will be provided in Section 4. The quantity $\mathcal{A} \subseteq [n]$ refers the *preserved set*, which is the complement of the set of screened coordinates. Starting from $\mathcal{A} = [n]$, it is dynamically reduced (line 15) by removing the components that are surely identified as being saturated in the solution vector $\mathbf{x}^*$ (lines 11 and 12), according to the safe rule (11). At the same time, these identified saturated components in $\mathbf{x}$ are permanently set to their optimal value (line 13), and their contribution to the vector of measurements is stored in $\mathbf{z} \in \mathbb{R}^m$ (line 14).

It follows that the computation of

$$\mathbf{A}\mathbf{x} = \mathbf{A}_\mathcal{A}\mathbf{x}_\mathcal{A} + \mathbf{A}_{\mathcal{A}^c}\mathbf{x}_{\mathcal{A}^c} = \mathbf{A}_\mathcal{A}\mathbf{x}_\mathcal{A} + \mathbf{z} \tag{12}$$

can be reduced from $O(mn)$ to $O(m(|\mathcal{A}| + 1))$ in the next calls to the primal solver. As such, the more saturated components are identified, the larger is the speed improvement in the next calls to the primal solver.

Algorithm 1 Generic screening procedure for Problem (1)

1: **Initialize** $\mathcal{A} = [n]$, $\mathbf{x} \in [\mathbf{l}, \mathbf{u}]$, $\varepsilon_{\text{gap}} > 0$
2: **Set** $\boldsymbol{\eta}$ according to the solver,
3: **Compute** $\alpha > 0$, a strong concavity bound of D
4: **Initialize** $\mathbf{z} = \mathbf{0}$
5: **repeat**
6: — *Solver update restricted to the preserved set* —
7: $\{\mathbf{x}_{\mathcal{A}}, \boldsymbol{\eta}\} \leftarrow \texttt{PrimalUpdate}(F(\mathbf{A}_{\mathcal{A}} \cdot + \mathbf{z}; \mathbf{y}); \mathbf{x}_{\mathcal{A}}, \boldsymbol{\eta})$
8: — *Dynamic safe screening* —
9: $\boldsymbol{\theta} \leftarrow \Theta(\mathbf{x}) \in \mathcal{F}_D$ ▷ *Dual update*
10: $r \leftarrow \sqrt{2\,\text{Gap}(\mathbf{x}, \boldsymbol{\theta})/\alpha}$ ▷ *Safe radius*
11: $\mathcal{S}_l \leftarrow \{j \in \mathcal{A} \mid \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta} < -r\|\mathbf{a}_j\|_2\}$ ▷ *Lower-saturated set*
12: $\mathcal{S}_u \leftarrow \{j \in \mathcal{A} \backslash \mathcal{J}_\infty^u \mid \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta} > r\|\mathbf{a}_j\|_2\}$ ▷ *Upper-saturated set*
13: $\mathbf{x}_{\mathcal{S}_l} \leftarrow \mathbf{l}_{\mathcal{S}_l}$, $\mathbf{x}_{\mathcal{S}_u} \leftarrow \mathbf{u}_{\mathcal{S}_u}$ ▷ *Set of saturated entries*
14: $\mathbf{z} \leftarrow \mathbf{z} + \mathbf{A}_{\mathcal{S}_l \cup \mathcal{S}_u} \mathbf{x}_{\mathcal{S}_l \cup \mathcal{S}_u}$ ▷ *Update saturated part*
15: $\mathcal{A} \leftarrow \mathcal{A} \backslash (\mathcal{S}_l \cup \mathcal{S}_u)$ ▷ *Update preserved set*
16: **until** $\text{Gap}(\mathbf{x}, \boldsymbol{\theta}) < \varepsilon_{\text{gap}}$

Remark 3. The introduction of the variable $\mathbf{z}$ in Algorithm 1 is convenient to present a generic algorithm that encompasses the complete class of functions f considered in this work. Yet, this additional variable can be discarded for some loss functions. For example, when $F(\mathbf{Ax}; \mathbf{y}) = \|\mathbf{Ax} - \mathbf{y}\|_2^2$, line 14 can be replaced by $\mathbf{y} \leftarrow \mathbf{y} - \mathbf{A}_{\mathcal{S}_l \cup \mathcal{S}_u} \mathbf{x}_{\mathcal{S}_l \cup \mathcal{S}_u}$, thus avoiding the use of $\mathbf{z}$ as well as the addition operation in (12).

It is worth mentioning that the acceleration of the primal iterates provided by the screening procedure has to be balanced with the computational overhead of the screening step itself. This mainly concerns the cost of computing the inner products $\mathbf{a}_j^T\boldsymbol{\theta}$ at lines 11 and 12 which, all together, have a complexity of $O(|\mathcal{A}|m)$. Fortunately, provided a suitable choice of the dual update function $\Theta(\mathbf{x})$ (see Section 4), most standard primal solvers already require the computation of these inner products. They can thus be reused for free in the screening step.

Finally, let us emphasize that Algorithm 1 can be simplified in the NNLR case as shown in Algorithm 2. Indeed, as $\mathcal{J}_\infty^u = [n]$, we always have $\mathcal{S}_u = \emptyset$. Moreover, because $\mathbf{l} = \mathbf{0}$, the vector $\mathbf{z}$ remains always zero.

4 Computing a dual feasible point

A crucial step of the proposed screening procedure lies in the computation of a dual feasible point (function $\Theta : \mathbb{R}^n \to \mathcal{F}_D$ at line 9 of Algorithm 1). One can see from the definition of the Gap safe sphere in (9) that the closer $\Theta(\mathbf{x})$ is to $\boldsymbol{\theta}^\star$, the smaller the safe region is likely to be. Moreover, the computation of $\Theta(\mathbf{x})$ should be cheap in order to minimize the computational load of the screening step. In this section, we present ways of defining Θ so as to meet these two desirable properties. Without loss of generality, we focus only on the BVLR and NNLR cases, the case with mixed constraints being easily deduced from the latter two.

Algorithm 2 Screening procedure for NNLR

Initialize $\mathcal{A} = [n]$, $\mathbf{x} \geq 0$, $\varepsilon_{\text{gap}} > 0$
Set $\boldsymbol{\eta}$ according to the solver,
Compute $\alpha > 0$, a strong concavity bound of D
repeat
  — *Solver update restricted to the preserved set* —
  $\{\mathbf{x}_{\mathcal{A}}, \boldsymbol{\eta}\} \leftarrow \texttt{PrimalUpdate}(F(\mathbf{A}_{\mathcal{A}}\cdot; \mathbf{y}); \mathbf{x}_{\mathcal{A}}, \boldsymbol{\eta})$
  — *Dynamic safe screening* —
  $\boldsymbol{\theta} \leftarrow \Theta(\mathbf{x}) \in \mathcal{F}_D$ ▷ *Dual update*
  $r \leftarrow \sqrt{2\,\text{Gap}(\mathbf{x}, \boldsymbol{\theta})/\alpha}$ ▷ *Safe radius*
  $\mathcal{S}_l \leftarrow \{j \in \mathcal{A} \mid \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta} < -r\|\mathbf{a}_j\|_2\}$ ▷ *Lower-saturated set*
  $\mathbf{x}_{\mathcal{S}_l} \leftarrow \mathbf{0}$ ▷ *Set of saturated entries*
  $\mathcal{A} \leftarrow \mathcal{A} \backslash \mathcal{S}_l$ ▷ *Update preserved set*
until $\text{Gap}(\mathbf{x}, \boldsymbol{\theta}) < \varepsilon_{\text{gap}}$

4.1 Dual update for BVLR

As pointed out in Section 3.1, the dual problem of BVLR is unconstrained, i.e., $\mathcal{F}_D = \mathbb{R}^m$. As such, any point $\boldsymbol{\theta} \in \mathbb{R}^m$ is admissible to be a center for the Gap safe sphere. Following the *dual scaling* idea [18], we propose to define Θ as

$$\Theta(\mathbf{x}) := -\nabla F(\mathbf{A}\mathbf{x}; \mathbf{y}). \tag{13}$$

The notation $\nabla F(\mathbf{A}\mathbf{x}; \mathbf{y})$ refers to the gradient of the function $F(\cdot; \mathbf{y})$ evaluated in $\mathbf{A}\mathbf{x}$. Note that here no scaling of this gradient is required as $\mathcal{F}_D = \mathbb{R}^m$. The rationale behind this choice is twofold. First, we get from the primal-dual link (5) that $\Theta(\mathbf{x}) \to \boldsymbol{\theta}^\star$ as $\mathbf{x} \to \mathbf{x}^\star$. Second, for any first-order primal solver,

$$\nabla P(\mathbf{x}) = \mathbf{A}^\mathsf{T} \nabla F(\mathbf{A}\mathbf{x}; \mathbf{y}) = -\mathbf{A}^\mathsf{T}\Theta(\mathbf{x}) \tag{14}$$

is computed during the primal update step. This vector contains nothing else than the inner products needed for the screening test (lines 11 and 12 of Algorithm 1) and can thus be reused for free.

4.2 Dual update for NNLR

Computing a dual feasible point for the NNLR problem is more involved than for BVLR. Here, $\mathcal{F}_D = \{\boldsymbol{\theta} \in \mathbb{R}^m : \mathbf{A}^\mathsf{T}\boldsymbol{\theta} \leq 0\}$ and dual scaling is no longer possible since,

$$\mathbf{z} \notin \mathcal{F}_D \implies (\rho\mathbf{z}) \notin \mathcal{F}_D \quad \forall \rho > 0. \tag{15}$$

In other words, if $-\nabla F(\mathbf{A}\mathbf{x}; \mathbf{y})$ is not a feasible dual point (i.e., $\exists j$ such that $-\mathbf{a}_j^\mathsf{T} \nabla F(\mathbf{A}\mathbf{x}; \mathbf{y}) > 0$), then any scaled version of it remains not feasible.

Instead, assuming that $\text{Int}(\mathcal{F}_D) \neq \emptyset$ (i.e., that the interior of $\mathcal{F}_D$ is nonempty, see Remark 4), we propose a *dual translation* strategy (by analogy with dual scaling) defined, for any vector $\mathbf{t} \in \text{Int}(\mathcal{F}_D)$, as

$$\Theta(\mathbf{x}) := \Xi_{\mathbf{t}}(-\nabla F(\mathbf{A}\mathbf{x}; \mathbf{y})) \tag{16}$$

where $\Xi_{\mathbf{t}}$ is the translation in the direction of vector $\mathbf{t}$

$$\Xi_{\mathbf{t}}(\mathbf{z}) := \mathbf{z} + \left(\max_{j \in [n]} \frac{(\mathbf{a}_j^\mathsf{T} \mathbf{z})^+}{|\mathbf{a}_j^\mathsf{T} \mathbf{t}|} \right) \mathbf{t}. \tag{17}$$

Proposition 1 shows that this Θ indeed maps $\mathbf{x}$ onto $\mathcal{F}_D$ and leads to the desired convergence property. Moreover, similarly to (14), this dual translation allows one to reuse in the screening test some quantities computed during the primal update. Note that the additional inner products $\mathbf{A}^\mathsf{T}\mathbf{t}$ can be pre-computed, which keeps $\mathbf{A}^\mathsf{T}\Theta(\mathbf{x})$ as cheap as with the standard dual scaling procedure.

Proposition 1 (Validity of the dual translation). *Let Θ be defined as in* (16)*. Then, for any primal point $\mathbf{x} \in \mathbb{R}^n_{\geq 0}$, we have $\Theta(\mathbf{x}) \in \mathcal{F}_D$. Moreover, $\Theta(\mathbf{x}) \to \boldsymbol{\theta}^\star$ as $\mathbf{x} \to \mathbf{x}^\star$.*

Proof. It is sufficient to show that $\forall \mathbf{z} \in \mathbb{R}^m$, $\Xi_{\mathbf{t}}(\mathbf{z}) \in \mathcal{F}_D$, i.e., $\mathbf{A}^\mathsf{T}\Xi_{\mathbf{t}}(\mathbf{z}) \leq 0$. Denoting $\epsilon = \max_{k \in [n]} \frac{(\mathbf{a}_k^\mathsf{T}\mathbf{z})^+}{|\mathbf{a}_k^\mathsf{T}\mathbf{t}|}$, we get, by definition of Ξ that, $\forall j \in [n]$

$$\begin{aligned}
\mathbf{a}_j^\mathsf{T} \Xi_{\mathbf{t}}(\mathbf{z}) &= \mathbf{a}_j^\mathsf{T}(\mathbf{z} + \epsilon \mathbf{t}) = \mathbf{a}_j^\mathsf{T}\mathbf{z} + \epsilon \mathbf{a}_j^\mathsf{T}\mathbf{t} \\
&= \mathbf{a}_j^\mathsf{T}\mathbf{z} + \left(\max_{k \in [n]} \frac{(\mathbf{a}_k^\mathsf{T}\mathbf{z})^+}{|\mathbf{a}_k^\mathsf{T}\mathbf{t}|} \right) \mathbf{a}_j^\mathsf{T}\mathbf{t} \\
&\leq \mathbf{a}_j^\mathsf{T}\mathbf{z} + \frac{\mathbf{a}_j^\mathsf{T}\mathbf{z}}{|\mathbf{a}_j^\mathsf{T}\mathbf{t}|} \mathbf{a}_j^\mathsf{T}\mathbf{t} = 0
\end{aligned}$$

where we used the fact that $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$, i.e., $\mathbf{a}_j^\mathsf{T}\mathbf{t} < 0 \ \forall j \in [n]$. Finally, by continuity of $\nabla F(\mathbf{A}\cdot; \mathbf{y})$ we get that

$$-\nabla F(\mathbf{A}\mathbf{x}; \mathbf{y}) \to -\nabla F(\mathbf{A}\mathbf{x}^\star; \mathbf{y}) \underset{(5)}{=} \boldsymbol{\theta}^\star \text{ as } \mathbf{x} \to \mathbf{x}^\star.$$

Then, the continuity of $\Xi_{\mathbf{t}}$ together with $\Xi_{\mathbf{t}}(\boldsymbol{\theta}^\star) = \boldsymbol{\theta}^\star$ (as $\boldsymbol{\theta}^\star \in \mathcal{F}_D$) proves that $\Theta(\mathbf{x}) \to \boldsymbol{\theta}^\star$ as $\mathbf{x} \to \mathbf{x}^\star$.

Remark 4 (Comment on $\mathrm{Int}(\mathcal{F}_D) \neq \emptyset$). One may wonder to what extend such a condition is restrictive in practice. From the expression of $\mathcal{F}_D$ in (4), we get that $\mathrm{Int}(\mathcal{F}_D) \neq \emptyset$ is equivalent to

$$\exists \boldsymbol{\omega} \in \mathbb{R}^m \ s.t. \ \mathbf{A}^\mathsf{T}\boldsymbol{\omega} < 0. \tag{18}$$

In other words, the columns of $\mathbf{A}$ must belong to the interior of a half space of $\mathbb{R}^m$ containing the origin on its boundary. Hence, if $\mathrm{Int}(\mathcal{F}_D) = \emptyset$ (i.e., (18) fails), we have $\mathbf{0} \in \mathrm{conv}\{\mathbf{a}_j\}_{j=1}^n$ and the NNLS problem is ill-posed as it admits infinitely many solutions. If furthermore $\mathbf{0} \in \mathrm{Int}\left(\mathrm{conv}\{\mathbf{a}_j\}_{j=1}^n\right)$, we have $\mathrm{cone}\{\mathbf{a}_j\}_{j=1}^n = \mathbb{R}^m$ meaning that the non-negativity constraint is useless [21]. Note that the latter case corresponds to $\mathcal{F}_D = \{\mathbf{0}\}$. To conclude, relevant NNLS problems satisfy $\mathrm{Int}(\mathcal{F}_D) \neq \emptyset$.

It remains to discuss how one can determine a direction $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$. Although there is no systematic approach for a general $\mathbf{A} \in \mathbb{R}^{m\times n}$, this can be achieved on a case-by-case basis for many relevant matrices $\mathbf{A}$.

Proposition 2. *The following types of matrix* $\mathbf{A}$ *ensure that* $\mathrm{Int}(\mathcal{F}_D) \neq \emptyset$. *Moreover, a vector* $\mathbf{t} \in \mathcal{F}_D$ *can be easily computed.*

1. $\mathbf{A} \in \mathbb{R}^{m\times n}$ *with* $\mathrm{rank}(\mathbf{A}) = n \leq m$. *Then, for any* $\mathbf{b} \in \mathbb{R}^n_{<0}$, *all the solutions of* $\mathbf{A}^\mathsf{T}\mathbf{t} = \mathbf{b}$ *(there exists at least one) satisfy* $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$.
2. $\mathbf{A} \in \mathbb{R}^{m\times m}$ *orthogonal (i.e.,* $\mathbf{A}^\mathsf{T}\mathbf{A} = \mathbf{I}$*). Then, any negative linear combination of the columns of* $\mathbf{A}$, *i.e.,* $\mathbf{t} = \sum_{j\in[n]} \beta_j \mathbf{a}_j$ *for* $\boldsymbol{\beta} \in \mathbb{R}^n_{<0}$, *satisfies* $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$.
3. $\mathbf{A} \in \mathbb{R}^{m\times n}_{\geq 0}$ *with non-negative entries and no column with only zeros. Then, any negative vector* $\mathbf{t} \in \mathbb{R}^m_{<0}$ *satisfies* $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$.
4. $\mathbf{A} \in \mathbb{R}^{m\times n}$ *such that* $\mathbf{A}^\mathsf{T}\mathbf{A}$ *contains a column (say the* j*th) with all entries positive. Then,* $\mathbf{t} = -\mathbf{a}_j$ *satisfies* $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$.

Proof. We prove each case independently.

1. The existence of solutions for $\mathbf{A}^\mathsf{T}\mathbf{t} = \mathbf{b}$ comes from the fact that the columns of $\mathbf{A}^\mathsf{T}$ span $\mathbb{R}^n$ (as $\mathrm{rank}(\mathbf{A}) = n$). Then, as $\mathbf{b} \in \mathbb{R}^n_{<0}$, we have $\mathbf{A}^\mathsf{T}\mathbf{t} = \mathbf{b} < 0$ and thus $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$.

2. Due to the orthogonality of $\mathbf{A}$, we have that $\forall i \neq j$, $\mathbf{a}_i^\mathsf{T}\mathbf{a}_j = 0$. Hence, for any $\boldsymbol{\beta} \in \mathbb{R}^n_{<0}$, we have

$$\forall i \in [n], \quad \mathbf{a}_i^\mathsf{T}\left(\sum_{j\in[n]} \beta_j \mathbf{a}_j\right) = \beta_i \|\mathbf{a}_i\|_2^2 < 0$$

which shows that $\mathbf{t} = \sum_{j\in[n]} \beta_j \mathbf{a}_j \in \mathrm{Int}(\mathcal{F}_D)$.

3. Given that $\forall i \in [n]$, $\mathbf{a}_i \in \mathbb{R}^m_{\geq 0}\backslash\{\mathbf{0}\}$, we have, for any negative vector $\mathbf{t} \in \mathbb{R}^m_{<0}$, that $\forall i \in [n]$, $\mathbf{a}_i^\mathsf{T}\mathbf{t} < 0$ which shows that $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$.

4. Direct consequence of the assumption on $\mathbf{A}$.

5 Experiments

In this paper, we restrict ourselves to the popular bounded-variable ($\mathbf{l} < \mathbf{u} < \infty$) and non-negative ($\mathbf{l} = \mathbf{0} < \mathbf{u} = \infty$) least squares problems, i.e., $f(z; y) = \frac{1}{2}(z-y)^2$. In this case, the conjugate function is given by $f^*(\theta; y) = \frac{1}{2}((y+\theta)^2 - y^2)$. In the NNLS experiments, we assumed $\mathbf{A} \in \mathbb{R}^{m\times n}_{\geq 0}$, which corresponds to traditional scenarios. As such, unless otherwise stated, the dual translation vector is set to $\mathbf{t} = -\mathbf{1}$ (according to Proposition 2).

We use projected gradient descent [19] and the Chambolle-Pock primal-dual algorithm [5] to solve the BVLS problem. For NNLS, we consider both the coordinate descent (CD) method of [11] and the `lsqnonneg` routine of MATLAB (a variant of the original active set algorithm of [16]) denoted simply *Active Set* hereafter. The algorithms are stopped when the duality gap falls below 10^{-6}. For

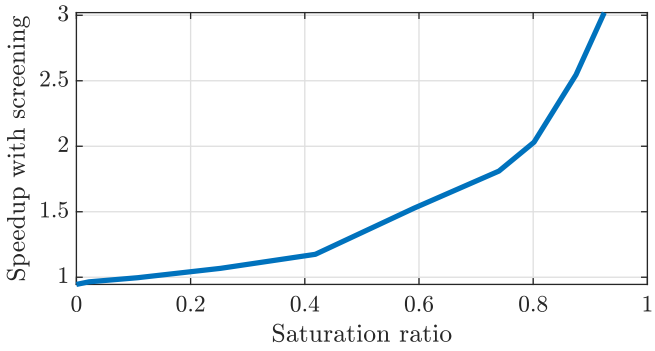

Fig. 1. Speedup w.r.t. saturation ratio for a BVLS problem of size $m = 4000$, $n = 2000$, solved with projected gradient descent. The matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and vector $\mathbf{y} \in \mathbb{R}^m$ have been generated according to $a_{ij} \sim \mathcal{N}(0, 1)$ and $y_i \sim \mathcal{N}(0, 1)$. The saturation ratio is controlled by varying the size of the box $b[-\mathbf{1}, \mathbf{1}]$.

all baselines without screening, the duality gap has been computed offline so as not to impact the measured execution times.

Section 5.1 reports results that illustrate how screening performance varies with specific experimental parameters. Then, Section 5.2 is devoted to the evaluation of the proposed screening with real datasets.

5.1 Understanding screening behaviour

Influence of the saturation ratio. Figure 1 presents the typical evolution of the speedup factor achieved with screening as a function of the saturation ratio (i.e., s/n where $s \in \mathbb{N}$ is the number of saturated components in $\mathbf{x}^\star$). As expected, the higher the number of saturated components, the best the screening performance. Yet, there is a critical value of the saturation ratio under which the computational overhead of screening dominates the acceleration of the primal update, leading to an overall "speedup" below 1.

Influence of the problem parameters. We report in Tables 1 and 2 execution times for NNLS and BVLS problems respectively with increasing size and fixed saturation ratio. For both active set and coordinate descent algorithms, we observe that the screening performance (i.e., the speed improvement) increases with the size and the level of indeterminacy of the problem. Speedups are obtained consistently and independently of the problem instance (BV or NN) and the chosen solver. Most tested solvers benefit from speedups of around five times, except for the Active Set solver which, by its own nature, is less prone to screening approaches (as they already manipulate reduced sets of coordinates).

Table 1. Execution times and speedup for a NNLS problem with fixed number of rows $m = 2000$ and different number of columns n. For each instance of the problem, the matrix $\mathbf{A} \in \mathbb{R}^{m \times n}_{\geq 0}$ has been generated using $a_{ij} = |\eta|$ with $\eta \sim \mathcal{N}(0, 1)$. The data vector $\mathbf{y} \in \mathbb{R}^{m}_{\geq 0}$ has been obtained as $\mathbf{y} = \mathbf{A}\bar{\mathbf{x}} + \boldsymbol{\epsilon}$ where $\bar{\mathbf{x}} \in \mathbb{R}^{n}_{\geq 0}$ is such that $\|\bar{\mathbf{x}}\|_0/n = 0.05$ with non-zero entries $\bar{x}_j$ distributed like a_{ij} and $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is such that $\epsilon_i \sim \mathcal{N}(0, 1)$.

	n	Baseline [s]	Screening [s]	Speedup
Coord. Descent	1000	2.19	0.71	3.08
	2000	10.2	2.09	4.87
	4000	64.28	9.52	6.75
	6000	146.12	18.63	7.84
Active Set	1000	0.11	0.09	1.25
	2000	0.16	0.13	1.23
	4000	0.33	0.25	1.31
	6000	0.36	0.26	1.38

Table 2. Execution times and speedup for a BVLS problem with $m = 1000$ and same setup as in Table 1, except that $\bar{x}_j \sim \mathcal{U}(0, 1)$ with bounds $\mathbf{l} = \mathbf{0}$ and $\mathbf{u} = \mathbf{1}$.

	n	Baseline [s]	Screening [s]	Speedup
Proj. Grad.	500	9.41	1.71	5.49
	1000	27.98	4.33	6.47
	2000	127.21	18.82	6.76
	3000	347.05	48.46	7.16
Primal dual	500	0.26	0.08	3.41
	1000	0.84	0.19	4.52
	2000	2.92	0.59	4.97
	3000	5.20	0.95	5.48

Influence of choice of the dual point. In Figure 2, we report for a NNLS problem the screening ratio (i.e., the number of identified saturated components relatively to the size n) as a function of the iteration number, for different choices of the dual translation vector $\mathbf{t} \in \mathrm{Int}(\mathcal{F}_D)$. Clearly, the choice of $\mathbf{t}$ affects the screening performance. Although the existence of an optimal choice remains an open question, the reported results allow for some intuition. Indeed, denoting $\mathbf{a}_+$ (resp., $\mathbf{a}_-$) the column of $\mathbf{A} \in \mathbb{R}^{m \times n}_{\geq 0}$ that correlates the most (resp., the least) with all other columns, we can see that setting $\mathbf{t} = -\mathbf{a}_+$ leads to significantly better screening performance than $\mathbf{t} = -\mathbf{a}_-$. Other relevant choices for this example are $\mathbf{t} = -\mathbf{1}$ (used in all remaining NNLS examples) and $\mathbf{t} = -\frac{1}{n}\sum_j \mathbf{a}_j$. This suggests the conjecture that a relevant $\mathbf{t}$ should be close to the "central axis" of $\mathrm{cone}\{\mathbf{a}_j\}_{j=1}^n$.

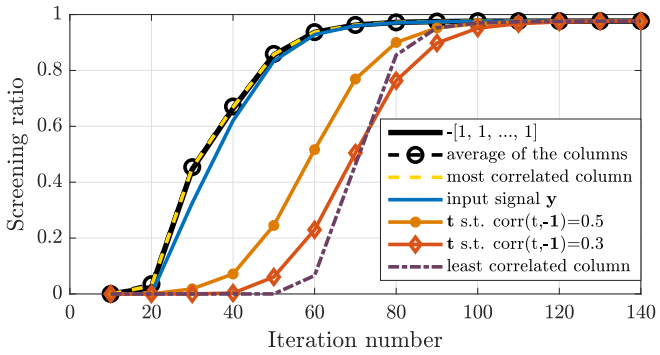

Fig. 2. Screening performance with different choices for the translation vector $\mathbf{t}$ on a NNLS problem with the NIPS papers dataset (described in Section 5.2).

Limits of screening: oracle dual point. To evaluate the practical limits of the proposed screening approach, we perform experiments in which a perfect dual update is performed artificially. The results in Figure 3 show that, although significant acceleration is already obtained with the usual dual update, there is still room for improvement. The gap to the optimal bound could be reduced in practice with better dual point estimations. In the NNLS case specifically, this could be achieved by defining better dual directions or maybe even with a completely different approach than the proposed dual translation (see, e.g. [17]).

5.2 Performance in applicative scenarios

BVLS for hyperspectral unmixing. In this experiment, the data vector $\mathbf{y}$ is the observed light reflectance spectrum vector of a random pixel from the Cupitre hyperspectral image [14]. The columns of $\mathbf{A} \in \mathbb{R}_{\geq 0}^{188\times 342}$ are set as the reflectance spectra of pure materials from the USGS High Resolution Spectral Library [8], using the same pre-processing as in [2]. The goal of the regression problem is thus to identify and determine the proportions (so-called abundances) of the materials that compose $\mathbf{y}$. Physical observation constraints dictate that the abundances should lie in the interval $[0, 1]$, leading to a BVLS problem [2].

The convergence and screening ratio curves of the projected gradient algorithm with and without screening are presented in Figure 4. We observe that, as the screening ratio progressively grows, the iterations become faster and the convergence curve eventually detaches from the baseline.

NNLS for archetypal analysis. The NIPS papers dataset contains word counts from 2484 papers published at the NIPS conference between 1988 to 2003 [12]. We removed any all-zero columns or rows from the original data matrix and normalized its columns. The input data $\mathbf{y} \in \mathbb{R}_{\geq 0}^{2483}$ is taken as a random sample of the dataset and all remaining samples form the columns of $\mathbf{A} \in \mathbb{R}_{\geq 0}^{2483\times 14035}$.

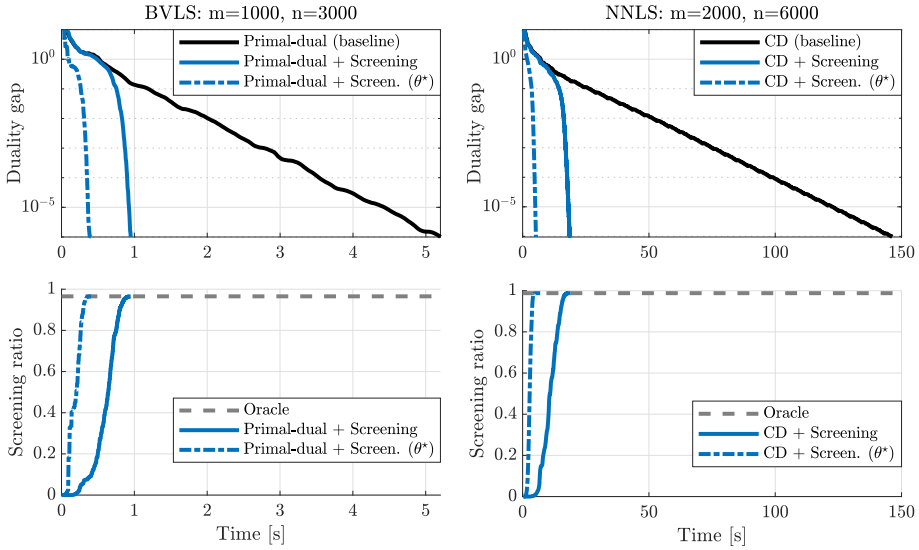

Fig. 3. Duality gap convergence (top) and screening ratio (bottom) through time for the simulation setups described in Tables 1 and 2. Left: BVLS problem with primal-dual solver. Right: NNLS problem with coordinate descent solver. The dashed curves correspond to the screening approach artificially informed with an optimal dual point ($\boldsymbol{\theta}^\star$), which reaches a speedup of 12.8 (left) and 27.8 (right).

The convergence and screening ratio curves of the active set and coordinate descent algorithms are depicted in Figure 5. While we observe a substantial acceleration for the coordinate descent, its counterpart for the active set method is more subdued. Yet, in both cases, the proposed screening strategy allows to accelerate the considered solvers.

6 Conclusion

In the paper we extended the recently fruitful safe screening framework to the intrinsically distinct family of box-constrained problems. Instead of identifying zero coordinates induced by a sparse regularization term, we manage to safely identify saturated coordinates induced by the constraints. The main technical challenge in the proposed approach lies in the choice of a dual feasible point which is non-trivial when the box limit is allowed to be unbounded (an example being the widespread NNLS problem). We proposed the simple and efficient *dual translation* procedure to tackle this problem and suggested some practical choices for the translation direction. Determining an optimal translation direction is actually a challenging problem that deserves further studies—inasmuch as the quality of the dual point can decisively influence the screening performance.

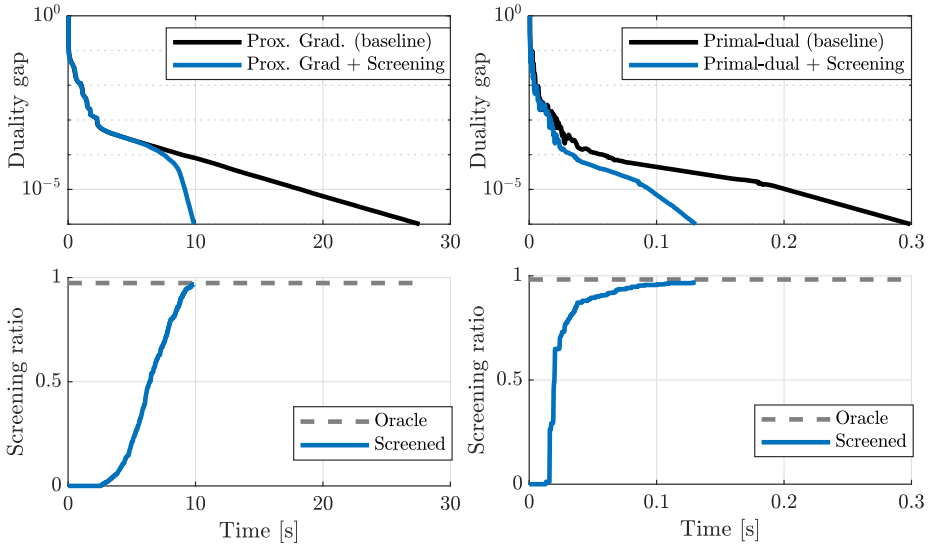

Fig. 4. Convergence and screening ratio for the BVLS hyperspectral experiment. Speedups of 2.79 and 2.30 are achieved respectively for the projected gradient (left) and primal-dual (right) solvers.

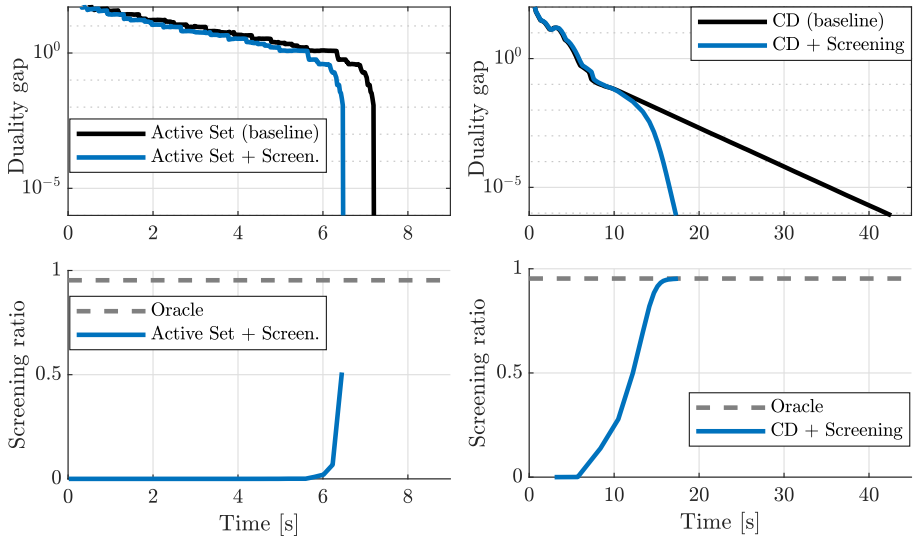

Fig. 5. Convergence and screening ratio for the NNLS NIPS papers experiment. Speedups of 2.44 and 1.12 are achieved respectively for the coordinate descent (left) and the active set (right) solvers.

References

1. Bellavia, S., Macconi, M., Morini, B.: An interior point newton-like method for non-negative least-squares problems with degenerate solution. Numerical Linear Algebra with Applications **13**(10), 825–846 (2006)
2. Bioucas-Dias, J.M., Plaza, A., Dobigeon, N., Parente, M., Du, Q., Gader, P., Chanussot, J.: Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **5**(2), 354–379 (2012)
3. Boyer, C., Chambolle, A., Castro, Y.D., Duval, V., De Gournay, F., Weiss, P.: On representer theorems and convex regularization. SIAM Journal on Optimization **29**(2), 1260–1281 (2019)
4. Bro, R., De Jong, S.: A fast non-negativity-constrained least squares algorithm. Journal of Chemometrics: A Journal of the Chemometrics Society **11**(5), 393–401 (1997)
5. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. Journal of Mathematical Imaging and Vision **40**(1), 120–145 (May 2011). https://doi.org/10.1007/s10851-010-0251-1
6. Chen, D., Plemmons, R.J.: Nonnegativity constraints in numerical analysis. In: The birth of numerical analysis, pp. 109–139. World Scientific (2010)
7. Cichocki, A., Amari, S.: Families of Alpha- Beta- and Gamma- divergences: Flexible and robust measures of similarities. Entropy **12**(6), 1532–1568 (June 2010)
8. Clark, R.N., Swayze, G.A., Wise, R.A., Livo, K.E., Hoefen, T.M., Kokaly, R.F., Sutley, S.J.: USGS digital spectral library splib06a. Tech. rep., US Geological Survey (2007)
9. Dantas, C., Soubies, E., Févotte, C.: Expanding boundaries of Gap Safe screening. Journal of Machine Learning Research (JMLR) **22**(236), 1–57 (2021)
10. Elvira, C., Herzet, C.: Safe squeezing for antisparse coding. IEEE Transactions on Signal Processing **68**, 3252–3265 (2020)
11. Franc, V., Hlaváč, V., Navara, M.: Sequential coordinate-wise algorithm for the non-negative least squares problem. In: International Conference on Computer Analysis of Images and Patterns. pp. 407–414. Springer (2005)
12. Globerson, A., Chechik, G., Pereira, F., Tishby, N.: Euclidean embedding of co-occurrence data. The Journal of Machine Learning Research **8**, 2265–2295 (2007)
13. Hiriart-Urruty, J.B., Lemaréchal, C.: Convex Analysis and Minimization Algorithms II. Springer Berlin Heidelberg (1993)
14. Jia, S., Qian, Y.: Spectral and spatial complexity-based hyperspectral unmixing. IEEE Transactions on Geoscience and Remote Sensing **45**(12), 3867–3879 (2007). https://doi.org/10.1109/TGRS.2007.898443
15. Johansson, B., Elfving, T., Kozlov, V., Censor, Y., Forssén, P.E., Granlund, G.: The application of an oblique-projected landweber method to a model of supervised learning. Mathematical and computer modelling **43**(7-8), 892–909 (2006)
16. Lawson, C.L., Hanson, R.J.: Solving least squares problems. Prentice-Hall Series in Automatic Computation (1974)
17. Massias, M., Vaiter, S., Gramfort, A., Salmon, J.: Dual extrapolation for sparse glms. Journal of Machine Learning Research **21**(234), 1–33 (2020)
18. Ndiaye, E., Fercoq, O., Gramfort, A., Salmon, J.: Gap safe screening rules for sparsity enforcing penalties. Journal of Machine Learning Research **18**(128), 1–33 (Nov 2017)

19. Polyak, R.A.: Projected gradient method for non-negative least square. Contemp Math **636**, 167–179 (2015)
20. Rockafellar, R.T.: Convex analysis. Princeton University Press (1970)
21. Slawski, M., Hein, M.: Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization. Electronic Journal of Statistics **7**, 3004–3056 (2013)
22. Stark, P.B., Parker, R.L.: Bounded-variable least-squares: An algorithm and applications. Computational Statistics **10**, 129–129 (1995)
23. Van Benthem, M.H., Keenan, M.R.: Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems. Journal of Chemometrics: A Journal of the Chemometrics Society **18**(10), 441–450 (2004)

Ethical Statement

This work contains no sensible or private data. No direct harmful application to society has been identified by the authors. The research was conducted with a commitment to ethical principles and scientific rigor. No human or animal subject were directly involved in this work. The researchers aimed to ensure that the results of this study are accurate, transparent and reproducible, and that they contribute to the advancement of scientific knowledge and to the benefit of society.

A Relation to ℓ_∞-regularization

A particular case of (1), obtained by setting $\mathbf{u} = -\mathbf{l} = c\mathbf{1}$, $c > 0$, is the ℓ_∞-constrained optimization problem

$$\begin{aligned} \mathbf{x}^\star \in \underset{\mathbf{x}\in\mathbb{R}^n}{\operatorname{argmin}} \; & \sum_{i=1}^{m} f([\mathbf{A}\mathbf{x}]_i; y_i) \\ & \text{s. t. } \|\mathbf{x}\|_\infty \leq c. \end{aligned} \tag{19}$$

The penalized counterpart of the above problem leads to the ℓ_∞-regularized linear regression problem considered by [10]

$$\mathbf{x}^\star \in \underset{\mathbf{x}\in\mathbb{R}^n}{\operatorname{argmin}} \; \sum_{i=1}^{m} f([\mathbf{A}\mathbf{x}]_i; y_i) + \lambda\|\mathbf{x}\|_\infty, \tag{20}$$

where the ℓ_∞-norm penalization is controlled by the parameter $\lambda > 0$, to which corresponds a certain value of parameter c in the previous constrained formulation. Note that, in [10], only the least-squares case is addressed.

B Dual problems derivation

As a direct application of the Fenchel duality formalism [20, Theorem 31.3] and the coordinate-wise separability property of the convex conjugate [13, Ch. X, Prop. 1.3.1] we have the following pair of generic primal-dual problems

$$\mathbf{x}^\star \in \underset{\mathbf{x}\in\mathbb{R}^n}{\operatorname{argmin}} \overbrace{\sum_{i=1}^{m} f([\mathbf{A}\mathbf{x}]_i; y_i)}^{F(\mathbf{A}\mathbf{x};\mathbf{y})} + \Omega(\mathbf{x}), \tag{21}$$

$$\boldsymbol{\theta}^\star = \underset{\boldsymbol{\theta}\in\mathbb{R}^m}{\operatorname{argmax}} - \underbrace{\sum_{i=1}^{m} f^*(-\theta_i; y_i)}_{F^*(-\boldsymbol{\theta};\mathbf{y})} - \Omega^*(\mathbf{A}^\mathsf{T}\boldsymbol{\theta}), \tag{22}$$

where $\Omega(\mathbf{x}) = \mathbb{1}_{\mathbf{x}\in[\mathbf{l},\mathbf{u}]}$. To complete the demonstration, we need to calculate the conjugate function Ω^* of Ω.

From the definition of the conjugate function and the separability of Ω, we have

$$\begin{aligned} \Omega^*(\mathbf{w}) &= \sup_{\mathbf{x}\in\mathbb{R}^n} \langle \mathbf{x}, \mathbf{w}\rangle + \Omega(\mathbf{x}) \\ &= \sum_{j\in[n]} \sup_{x_j\in[l_j,u_j]} x_j w_j \\ &= \sum_{j\in[n]\setminus\mathcal{J}_\infty^u} \sup_{x_j\in[l_j,u_j]} x_j w_j + \sum_{j\in\mathcal{J}_\infty^u} \sup_{x_j>l_j} x_j w_j. \end{aligned}$$

The first term is a sum of sup of linear functions over compact sets. As such, each sup is attained on the boundary. If $w_j < 0$, the sup is attained at $x_j = l_j$ whereas if $w_j > 0$, it is attained at $x_j = u_j$. Overall, the value of each sup is given by $l_j(w_j)^- + u_j(w_j)^+$.

The second term is a sum of sup of linear functions over umbounded sets. Each sup is either attained at $x_j = l_j$ if $w_j < 0$ or equals $+\infty$ if $w_j > 0$. Each sup is thus equal to $l_j(w_j)^- + \mathbb{1}_{w_j \leq 0}$.

Combining all these expressions, we obtain

$$\Omega^*(\mathbf{w}) = \sum_{j \in [n]} l_j(w_j)^- + \sum_{j \in [n] \setminus \mathcal{J}_\infty^u} u_j(w_j)^+ + \mathbb{1}_{\mathbf{w} \leq 0}, \tag{23}$$

which, with (22), completes the proof.

C First-order optimality conditions

First-order optimality conditions for the generic primal-dual pair (21)-(22) are given by [20, Theorem 31.3]:

$$\boldsymbol{\theta}^\star = -\nabla F(\mathbf{A}\mathbf{x}^\star; \mathbf{y}), \tag{24}$$

$$\mathbf{A}^\mathsf{T}\boldsymbol{\theta}^\star \in \partial\Omega(\mathbf{x}^\star). \tag{25}$$

For $F(\mathbf{A}\mathbf{x}; \mathbf{y}) = \sum_{i=1}^m f([\mathbf{A}\mathbf{x}]_i; y_i)$ and $\Omega(\mathbf{x}) = \sum_{j=1}^n \Omega_j(x_j)$ coordinate-wise separable, it simplifies to:

$$\forall i \in [m],\ \theta_i^\star = -f'([\mathbf{A}\mathbf{x}^\star]_i; y_i), \tag{26}$$

$$\forall j \in [n],\ \mathbf{a}_j^\mathsf{T}\boldsymbol{\theta}^\star \in \partial\Omega_j(x_j^\star). \tag{27}$$

Finally, the proof is completed with the expression of the sub-differential of $\Omega_j(x_j) = \mathbb{1}_{x_j \in [l_j, u_j]}$,

$$\partial\Omega_j(x_j) = \begin{cases} (-\infty, 0] \text{ if } & x_j = l_j \\ [0, +\infty) \text{ if } & x_j = u_j \\ \{0\} \quad \text{if } & x_j \in (l_j, u_j) \end{cases}$$

where the second case only occurs when $u_j < \infty$.