Appl. Numer. Math. 208 (2025) 284–296

Contents lists available at ScienceDirect

Applied Numerical Mathematics

journal homepage: www.elsevier.com/locate/apnum

Research Paper

An augmented Lagrangian approach for cardinality constrained minimization applied to variable selection problems

N. Krejić [a], E.H.M. Krulikovski [b], M. Raydan [b,*]

[a] Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia
[b] Center for Mathematics and Applications (NovaMath), FCT NOVA, 2829-516 Caparica, Portugal

ARTICLE INFO

Keywords:
Cardinality constraints
Variable selection
Constrained linear least-squares
Augmented Lagrangian method

ABSTRACT

To solve convex constrained minimization problems, that also include a cardinality constraint, we propose an augmented Lagrangian scheme combined with alternating projection ideas. Optimization problems that involve a cardinality constraint are NP-hard mathematical programs and typically very hard to solve approximately. Our approach takes advantage of a recently developed and analyzed continuous formulation that relaxes the cardinality constraint. Based on that formulation, we solve a sequence of smooth convex constrained minimization problems, for which we use projected gradient-type methods. In our setting, the convex constraint region can be written as the intersection of a finite collection of convex sets that are easy and inexpensive to project. We apply our approach to a variety of over and under determined constrained linear least-squares problems, with both synthetic and real data that arise in variable selection, and demonstrate its effectiveness.

1. Introduction

Let us start by considering the following general cardinality-constrained optimization problem

$$\begin{array}{ll} \min\limits_{x} & f(x) \\ \text{subject to:} & x \in \Omega \quad \text{and} \quad \|x\|_0 \leq \alpha, \end{array} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, $\Omega \subset \mathbb{R}^n$ is a closed and convex set, and $\alpha \in \mathbb{N}$ is an expected cardinality upper bound which is fixed in advance. The L_0 (quasi) norm $\|x\|_0$ denotes the number of nonzero components of the vector x. The constraint $\|x\|_0 \leq \alpha$ is called the cardinality constraint, and we assume that $\alpha < n$ since otherwise the cardinality constraint is not required.

The main difficulty for solving (1) is that the cardinality constraint involves the L_0 (quasi) norm, which is not a norm, nor continuous neither convex. In fact, optimization problems with cardinality constraints are NP-hard problems (see, e.g., [11,23,44]) and, due to the non-tractability of the zero norm, they are usually very difficult to solve approximately. Nevertheless, for low-dimensional problems they can be solved by global techniques from combinatorial optimization; see, e.g., [9,31,50] and references therein.

* Corresponding author.
E-mail addresses: natasak@uns.ac.rs (N. Krejić), e.krulikovski@fct.unl.pt (E.H.M. Krulikovski), m.raydan@fct.unl.pt (M. Raydan).

https://doi.org/10.1016/j.apnum.2023.12.006
Received 15 October 2023; Received in revised form 5 December 2023; Accepted 14 December 2023

Available online 19 December 2023

In this work, we are concerned with a particular version of problem (1), which arises when solving variable (or feature) selection problems:

$$\begin{array}{ll} \min\limits_{x} & f(x) = \frac{1}{2}\|Ax-b\|_2^2 + \lambda_{reg}\|x\|_2^2 \\ \text{subject to:} & Hx = z, \ \ l \le x \le u, \ \ \text{and} \ \ \|x\|_0 \le \alpha, \end{array} \tag{2}$$

where $A \in \mathbb{R}^{m\times n}$, $H \in \mathbb{R}^{p\times n}$, $z \in \mathbb{R}^p$, $b \in \mathbb{R}^m$, $l \in \mathbb{R}^n$, and $u \in \mathbb{R}^n$ are all given. Concerning the vectors l and u, we assume that $-\infty < l_i < u_i < \infty$ for all $1 \le i \le n$. We also assume that $m > p$, $rank(H) = p$, and $\lambda_{reg} > 0$ is a given real regularization parameter. We note that in (2) as a special case of problem (1), the objective function $f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable, and the closed and polyhedral convex set Ω is the intersection of the compact box $\{x \in \mathbb{R}^n \mid l \le x \le u\}$, and the p linear varieties given by $L_i = \{x \in \mathbb{R}^n \mid H_i^\top x = z_i \text{ for all } 1 \le i \le p\}$, where H_i denotes the i-th row of H. In other words, Ω is the intersection of a finite collection of closed and convex simple sets (i.e., it is easy and inexpensive to project onto each one of them). In here we assume that H, z, l and u are given such that Ω is nonempty. We note that the constraint $Hx = z$, that appears in certain applications, represents a very small set of linear equations that cannot be solved in the least-squares sense, but must be satisfied with equality.

The main idea in variable selection ($m > n$) is to reduce as much as possible the number of variables to those that best reproduce the model, i.e., such that

$$\widehat{f}(x) = \frac{1}{2}\|Ax-b\|_2^2$$

is as small as possible on the feasible region. In other words, the goal is to assign the value zero to as many variables as possible that have little or no value in recovering the model response. However, we note that the need to find sparse and meaningful constrained least-squares solutions of determined ($m = n$), over-determined ($m > n$), and under-determined ($m < n$) linear systems $Ax = b$, where the matrix A has been affected by noise, is not only important in variable selection, but also appears in other domains of statistics, in machine learning tasks, and in a few other applied sciences; see, e.g., [22,26,33,43,44,46] and references therein.

In recent years, since it is computationally hard and since the use of the L_1-norm regularization (via Lasso or via Elastic Net, see [48,51]) has proved to have some shortcomings (see, e.g., [10,24]), the variable selection problem has attracted a great deal of attention; see, e.g., [20,24,27,28,31,38,42,45,49]. In our proposal, by imposing the L_0-norm constraint in (2) we can reduce the number of variables to as many as α, from which we can extract valuable information for the use of the model, while allowing the option to discard at least $n - \alpha$ variables that do not provide any relevant information. Clearly, the choice of the parameter α in (2) will play a fundamental role in our work.

Concerning the (Stress) regularization term $\lambda_{reg}\|x\|_2^2$, that appears in the objective function in (2), we note that it does not add any computational difficulty since it is smooth enough and the evaluation of its first and second derivatives represents a negligible computational cost. Moreover, this regularization term can provide additional benefits when combined with a sparsity imposition (such as the cardinality constraint) to solve variable selection problems.

For the general setting in (1), a continuous formulation has been proposed and analyzed in [21] to deal with cardinality constraint optimization problems. In this work, inspired by the penalty approach recently developed in [37] and applied to portfolio optimization problems, we propose a tracktable low-cost computational scheme based on an augmented Lagrangian strategy that, using the development in [21], addresses a continuous formulation of problem (2).

The rest of this document is organized as follows. In Section 2, we present the proposed specialized augmented Lagrangian strategy. In Section 3, we describe a suitable alternating projection scheme as well as a convenient low-cost projected gradient-type method that can be combined with the augmented Lagrangian approach of Section 2. In Section 4, we fully describe the proposed adaptive augmented Lagrangian algorithm and discuss its convergence properties. In Section 5, we present the results from the numerical experiments. For that, we consider a variety of linear systems to be solved via least-squares and we report the obtained results with all the discussed options, to solve the continuous formulation of problem (2). We also describe a possible collaboration between our approach and the Lasso regularization strategy, based on the use of the L_1-norm, to detect an appropriate choice of the parameter α. In Section 6, we present some final comments and perspectives.

2. An augmented Lagrangian strategy

Based on the theoretical development in [21], the continuous counterpart of problem (2) is given by:

$$\begin{array}{ll} \min\limits_{x,y} & f(x) = \widehat{f}(x) + \lambda_{reg}\|x\|_2^2 \\ \text{subject to:} & Hx = z, \ \ l \le x \le u, \\ & e^\top y \ge n - \alpha, \\ & x_i y_i = 0, \ \ \text{for all } 1 \le i \le n, \\ & 0 \le y_i \le 1, \ \ \text{for all } 1 \le i \le n, \end{array} \tag{3}$$

where $y \in \mathbb{R}^n$ is an auxiliary variable vector and $e \in \mathbb{R}^n$ denotes the vector of ones. Notice that the last n constraints of (3) denote a closed and bounded simple box in the auxiliary variable vector $y \in \mathbb{R}^n$. A more difficult formulation substitutes the simple box by a set of binary constraints given by: either $y_i = 0$ or $y_i = 1$ for all $1 \le i \le n$. In that case, the problem is an integer programming problem (much harder to solve) for which there are several algorithmic ideas already developed; see, e.g., [9,11,50]. However, it is not always convenient to use those schemes, especially for large-scale instances.

In here, we will focus on the continuous formulation (3), that will play a key role in our algorithmic proposal. For additional theoretical properties that include the relationship (concerning feasibility and optimality) between the original version (2) and the continuous relaxed version (3), see [21,34,39,40]. Out of those theoretical results, we would like to point out one that establishes a one-to-one correspondence between minimizers of problems (2) and (3), whenever the obtained solution, say $\bar{x}$, satisfies the cardinality constraint with equality, i.e., $\|\bar{x}\|_0 = \alpha$.

Theorem 1. *([39, Theorem 4]) Let $(\bar{x}, \bar{y})$ be a local minimizer of the relaxed problem (3). Then $\|\bar{x}\|_0 = \alpha$ if and only if $\bar{y}$ is unique, that is, if there exist exactly one $\bar{y}$ such that $(\bar{x}, \bar{y})$ is a local minimizer of (3). In this case, the components of $\bar{y}$ are binary (i.e., $\bar{y}_i = 0$ or $\bar{y}_i = 1$ for all $1 \le i \le n$) and $\bar{x}$ is a local minimizer of (2).*

We note that the so-called Hadamard constraint ($x \circ y = 0$, i.e., $x_i y_i = 0$ for all $1 \le i \le n$) defines a nonconvex set in $\mathbb{R}^{2n}$. We also note that the remaining constraints in (3), namely: $H_i^\top x = z_i$ for all $1 \le i \le p$, $l \le x \le u$, $e^\top y \ge n - \alpha$ and $0 \le y_i \le 1$ for all $1 \le i \le n$, are convex simple sets, and so an alternating projection scheme can be conveniently applied to project onto their intersection. In that sense, $x \circ y = 0$ is the only complicated constraint in (3). Hence, for solving the continuous formulation (3) we can apply a suitable iterative low-cost convex constrained scheme, such as a projected gradient-type method, in which the objective function includes $f(x)$ plus some convenient penalization terms that guarantee that $x \circ y = 0$ at the solution.

To penalize the non-satisfaction of the Hadamard constraint, let us consider the function $h : \mathbb{R}^{2n} \to \mathbb{R}^n$, defined as $h(x, y) = x \circ y$. If we add to the objective function of problem (3) the term

$$\frac{\tau}{2}\|h(x,y)\|_2^2 = \frac{\tau}{2}\sum_{i=1}^{n}(x_i^2 y_i^2), \tag{4}$$

where τ is a positive real parameter, then for $\tau > 0$ sufficiently large, $\|h(x,y)\|_2^2$ will be zero once (3) has been solved, and the Hadamard constraint will be satisfied. The key idea behind this classical penalization strategy is to solve a sequence of smooth convex constrained optimization problems (adding (4) to the objective function of (3)) for an increasing sequence of τ's. However, in general, when the penalty parameter τ becomes very large solving the constrained minimization subproblems could be very difficult.

A robust tool to avoid the ill-conditioning usually associated with large values of τ, and to effectively solve constrained optimization problems, is the augmented Lagrangian approach; see, e.g., [8,12] as two standard references. Based on that approach, the main idea for our proposal is to penalize only the Hadamard constraint $x \circ y = 0$ in (3) using the classical augmented Lagrangian function, given by

$$L(x, y, \hat{\lambda}, \tau) = \widehat{f}(x) + \lambda_{reg}\|x\|_2^2 + \sum_{i=1}^{n}\hat{\lambda}_i x_i y_i + \frac{\tau}{2}\sum_{i=1}^{n}(x_i^2 y_i^2), \tag{5}$$

where $\hat{\lambda}_i \in \mathbb{R}$ is the Lagrange multiplier associated with the equality constraint $x_i y_i = 0$, and $\tau > 0$ is the penalty parameter that appears in (4). Hence, for solving (3), the augmented Lagrangian approach will be applied to the following problem

$$\begin{array}{ll} \min\limits_{x,y} & L(x, y, \hat{\lambda}, \tau) \\ \text{subject to:} & (x, y) \in \widehat{\Omega}, \end{array} \tag{6}$$

where $\widehat{\Omega} = \{(x, y) \in \mathbb{R}^{2n} \ : \ Hx = z, \ l \le x \le u, \ e^\top y \ge n - \alpha, \ 0 \le y_i \le 1 \ \text{ for all } \ 1 \le i \le n\}$.

3. The SPG method and Dykstra's alternating projection algorithm

Projected Gradient (PG) methods provide an interesting low-cost option for solving (6). They are simple and easy to code, and avoid the need for matrix factorizations (no Hessian matrix is used). There have been many different variations of the early PG methods. In particular, a well-established and effective scheme is the so-called Spectral Projected Gradient (SPG) method; see Birgin et al. [13–16]).

The SPG algorithm starts with $(x_0, y_0) \in \mathbb{R}^{2n}$, and moves at every iteration j along the internal projected gradient direction $d_j = P_{\widehat{\Omega}}((x_j, y_j) - \beta_j \nabla L(x_j, y_j, \hat{\lambda}, \tau)) - (x_j, y_j)$, where $d_j \in \mathbb{R}^{2n}$ and β_j is the well-known spectral choice of step length (see [16]):

$$\beta_j = \frac{\langle s_{j-1}, s_{j-1}\rangle}{\langle s_{j-1}, (\nabla L(x_j, y_j, \hat{\lambda}, \tau) - \nabla L(x_{j-1}, y_{j-1}, \hat{\lambda}, \tau))\rangle},$$

and $s_{j-1} = (x_j, y_j) - (x_{j-1}, y_{j-1})$. In the case of rejection of the first trial point, $(x_j, y_j) + d_j$, the next ones are computed along the same direction, i.e., $(x_+, y_+) = (x_j, y_j) + \lambda d_j$, using a nonmonotone line search to choose $0 < \lambda \le 1$ such that the following condition holds

$$L(x_+, y_+, \hat{\lambda}, \tau) \le \max_{0 \le l \le \min\{j, M-1\}} L(x_{k-l}, y_{k-l}, \hat{\lambda}, \tau) + \gamma\lambda\langle d_j, \nabla L(x_j, y_j, \hat{\lambda}, \tau)\rangle,$$

where $M \ge 1$ is a given integer and γ is a small positive number. Therefore, the projection onto $\widehat{\Omega}$ must be performed only once per iteration. The SPG iterations stop when

$$\|P_{\widehat{\Omega}}((x_j, y_j) - \nabla L(x_j, y_j, \hat{\lambda}, \tau)) - (x_j, y_j)\|_2 \leq tol, \tag{7}$$

where $0 < tol < 1$ is a stopping tolerance. A key feature of the SPG method is to accept the initial spectral step-length as often as possible while ensuring global convergence. For this reason, the SPG method employs a non-monotone line search that does not impose functional decrease at every iteration. More details can be found in [13] and [14].

A convenient scheme for finding, at each SPG iteration, the required projections onto $\widehat{\Omega}$ is Dykstra's alternating projection algorithm [18]. Roughly speaking, Dykstra's algorithm projects in a clever way onto the easy convex sets, say Ω_i $(1 \leq i \leq q)$, individually to complete a cycle which is repeated iteratively. The algorithm has been adapted and used for solving a wide variety of different applications. For a review on Dykstra's method, its properties and applications, as well as many other alternating projection schemes; see, e.g., [29].

From a given vector $\tilde{x}$, Dykstra's algorithm generates two sequences: the iterates $\{x^i_\ell\}$ and the increments $\{I^i_\ell\}$. These sequences are defined by the following recursive formulae:

$$\begin{aligned} x^0_\ell &= x^q_{\ell-1}, \\ x^i_\ell &= P_{\Omega_i}(x^{i-1}_\ell - I^i_{\ell-1}) \qquad i = 1, 2, \ldots, q, \\ I^i_\ell &= x^i_\ell - (x^{i-1}_\ell - I^i_{\ell-1}) \qquad i = 1, 2, \ldots, q, \end{aligned} \tag{8}$$

for $\ell \in \mathbb{Z}^+$ with initial values $x^q_0 = \tilde{x}$ and $I^i_0 = 0$ for $i = 1, 2, \ldots, q$.

The sequence of increments plays a fundamental role in the convergence of the sequences $\{x^i_\ell\}$ to $x^* = P_{\widehat{\Omega}}(\tilde{x})$, the projection of $\tilde{x}$ onto $\widehat{\Omega}$. Concerning the strong convergence of algorithm (8), Boyle and Dykstra [18] established that for any given $\tilde{x}$ and any $i = 1, 2, \ldots, q$, $\|x^i_\ell - x^*\|_2 \to 0$ as $\ell \to \infty$.

About the rate of convergence, it is well-known that Dykstra's algorithm exhibits a linear rate of convergence in the polyhedral case ([29]), which is the case when solving (6). Finally, the stopping criterion associated with Dykstra's algorithm is a delicate issue. A discussion about this topic and the development of some robust stopping criteria are fully described in [17]. Based on that, in here we will stop the iterations when

$$\sum_{i=1}^{q} \|I^i_{\ell-1} - I^i_\ell\|_2^2 \leq \varepsilon, \tag{9}$$

where $\varepsilon > 0$ is a small given tolerance.

4. Adaptive augmented Lagrangian algorithm

In general an augmented Lagrangian method consists of a sequence of outer iterations. At each outer iteration, for a given current pair (x, y), a given vector of Lagrange multipliers $\hat{\lambda} \in \mathbb{R}^n$, and a given $\tau > 0$, an (approximate) solution of (6) is obtained using an inner iterative scheme. Then the Lagrange multipliers and the penalty parameter are conveniently updated before performing the next outer iteration. The advantage of this strategy is that the subproblems can be solved using low-cost iterative schemes that can deal with a very large number of variables.

In here, since $L(x, y, \hat{\lambda}, \tau)$ has continuous first derivatives and $\widehat{\Omega}$ is the intersection of a finite number of easy-to-project convex sets, the subproblems will be conveniently solved by the SPG method described in Section 3. The required projection on the feasible convex set $\widehat{\Omega}$, at each iteration of the SPG method, will be obtained using Dykstra's alternating projection algorithm also described in Section 3. In a general convex constrained minimization setting, the combination of the SPG method and Dykstra's algorithm has been analyzed in [15], and recently applied to solve portfolio problems subject to cardinality constraints [37]. We note that the SPG method has been proposed and analyzed to solve the subproblems of an augmented Lagrangian approach for general nonlinear programming problems [25]. It is also worth mentioning that different strategies to adapt the augmented Lagrangian approach to solve cardinality constrained optimization problems have been recently published [32,35].

We now present our Adaptive Augmented Lagrangian SPG (AAL-SPG) Algorithm for solving (3).

Adaptive Augmented Lagrangian SPG (AAL-SPG) Algorithm.

Step 0 : Given $\alpha \in [1, n-1]$, $(x_0, y_0) \in \mathbb{R}^{2n}$ as the initial point, $-\infty < \hat{\lambda}_{\min} < \hat{\lambda}_{\max} < \infty$, $\tau_0 > 0$, $\lambda_{reg} > 0$, $\hat{\lambda}_0 \in [\hat{\lambda}_{\min}, \hat{\lambda}_{\max}]^n \subset \mathbb{R}^n$, $0 < tol_1 < 1$, and $0 < tol_2 < 1$. Choose $\delta_0 \geq 1$ and set $k = 0$.

Step 1 : From (x_k, y_k) apply the SPG method until (7) holds with tolerance tol_1 to solve (6), using Dykstra's algorithm to find the projection onto $\widehat{\Omega}$ at each SPG iteration. Let (x_{k+1}, y_{k+1}) be the obtained solution.

Step 2 : If

$$\|h(x_{k+1}, y_{k+1})\|_2^2 \leq tol_2 \, \|x_{k+1}\|_2^2 \, \|y_{k+1}\|_2^2 \quad \text{and} \quad |\widehat{f}(x_{k+1}) - \widehat{f}(x_k)| \leq tol_2, \quad \text{then Stop.}$$

Otherwise, set $\hat{\lambda}_{k+1} = P_{[\hat{\lambda}_{\min}, \hat{\lambda}_{\max}]^n}\left(\hat{\lambda}_k + \tau_k h(x_{k+1}, y_{k+1})\right)$ and

Step 2a : If $\|h(x_{k+1}, y_{k+1})\|_2 \leq \dfrac{1}{\delta_k}\|h(x_k, y_k)\|$ set $\tau_{k+1} = \tau_k$

Step 2b : Otherwise, compute $\delta_{k+1} > \delta_k$ and set $\tau_{k+1} = \delta_{k+1}\tau_k$.

Step 3 : Set $k = k + 1$ and return to Step 1.

Remark 1. Some justifications and practical considerations of the AAL-SPG Algorithm are in order:

1. To keep a balanced trade-off between feasibility and optimality, that is, between $\widehat{f}(x)$ and $\|h(x,y)\|_2^2$, it is convenient to choose the initial parameter τ_0 using the largest eigenvalue of $A^\top A$, which is the Hessian of $\widehat{f}(x)$. For that, we can proceed as follows. Set $z = A^\top Ae$, where e is the vector of all ones, and $\tau_{aux} = z^\top A^\top Az/(z^\top z)$, i.e., the Rayleigh-quotient of $A^\top A$ with the vector z, that produces a good estimate of the largest eigenvalue of $A^\top A$. Then we set
$$\tau_0 = \frac{1}{\tau_{aux}} n(n-\alpha). \tag{10}$$
We note that the calculations to obtain τ_0 are two matrix-vector products with the matrix A and two matrix-vector products with the matrix $A^\top$ (the first pair of products is applied to the vector e and the second pair is applied to the vector z), plus the inner product $z^\top z$. Hence, the required complexity to obtain τ_0 is $4mn + n$ floating point operations (flops). We also note that τ_0 is computed only once at the beginning of the iterative process (at Step 0 of Algorithm AAL-SPG).
2. We need to drive $\|h(x,y)\|_2$ down to zero during the convergence process. For that in Step 2b we increase the penalization parameter as follows:
$$\tau_{k+1} = \delta_{k+1}\tau_k \quad \text{where} \quad \delta_{k+1} = \delta_k + \frac{(n-\alpha)}{2n} \min\left(1, \widehat{f}(x_k), \widehat{f}(x_{k-1})\right) \quad \text{and} \quad \delta_0 = 1. \tag{11}$$
Since $\alpha < n$ we have that $n \geq n - \alpha > 0$ and then $\frac{1}{2} \geq \frac{n-\alpha}{2n} > 0$. Therefore, $\delta_{k+1} > \delta_k > \cdots > \delta_0 = 1$ which in turn implies that $\tau_{k+1} > \tau_k$. We note that in practice this formula increases the penalty parameter in a controlled way. In all the reported experiments in Section 5, the sequence $\{\tau_k\}$ given by (11) was enough to guarantee that the Hadamard product goes down to zero.
3. In Step 2, we modify the penalty parameter τ_k and Lagrangian parameter $\hat{\lambda}_k$ based on the following considerations: If $\|h(x_k, y_k)\| > (1/\delta_k)\|h(x_{k-1}, y_{k-1})\|$, the current iterate has a tendency to deviate from the feasible region. As a consequence the penalty parameter τ_{k+1} needs to be increased. Otherwise, if $\|h(x_k, y_k)\| \leq (1/\delta_k)\|h(x_{k-1}, y_{k-1})\|$, the current iterate approaches the feasibility in a proper way, which indicates that the penalty parameter τ_k does not need to be increased. Nevertheless, it is convenient to update the Lagrangian parameter $\hat{\lambda}_k$ at all iterations.
4. In Step 2, for $1 \leq i \leq n$, $(\hat{\lambda}_{k+1})_i$ is obtained using the safeguard box defined by $\hat{\lambda}_{\min}$ and $\hat{\lambda}_{\max}$ as follows:
$$(\hat{\lambda}_{k+1})_i = \min\{\max\{\hat{\lambda}_{\min}, (\hat{\lambda}_k + \tau_k h(x_{k+1}, y_{k+1}))_i\}, \hat{\lambda}_{\max}\}.$$
5. The computational complexity of the AAL-SPG algorithm per iteration centers on the cost of evaluating the gradient of $L(x, y, \hat{\lambda}, \tau)$ in (5), which is performed only once at each SPG iteration, and that cost is mainly spent evaluating the term $\nabla \widehat{f}(x_k) = A^\top(Ax_k - b)$. The evaluation of the rest of the gradient terms of the objective function in (5) requires $O(n)$ flops. Notice that $\|h(x_{k+1}, y_{k+1})\|_2$, used at Step 2 of the AAL-SPG algorithm, also requires $O(n)$ flops. Now, the evaluation of $\nabla \widehat{f}(x_k)$ involves two matrix-vector products, one with A and the other one with A^T, and that adds up to $2nm$ flops. Note that the function evaluation $\widehat{f}(x_k)$ that is also needed at each SPG iteration comes for free once the gradient has been evaluated. We also note that projecting onto each convex set that makes up $\widehat{\Omega}$ in (6) requires a pair of inner products, which also represent $O(n)$ flops. Therefore, the total cost per iteration of the SPG scheme is $2nm + O(n)$ flops.

We now present the convergence properties of Algorithm AAL-SPG, which are directly obtained from the convergence theory established for the augmented Lagrangian method described and analyzed in [25], with the obvious modifications to be adapted to our algorithmic framework. Therefore, their proof is omitted. The following theorem unifies theorems 4.1 and 4.2 in [25], which focus on the issues of feasibility and optimality, respectively. At this point, we would like to recall that a point is said to be *regular* if the gradients of the active constraints are linearly independent.

Theorem 2. *If $(\bar{x}, \bar{y})$ is a limit point of a sequence generated by Algorithm AAL-SPG, and every (x, y) in $\widehat{\Omega}$ is a regular point, then $(\bar{x}, \bar{y})$ is a first-order stationary point of the feasibility problem*

$$\min \|h(x,y)\|_2^2 \ \textit{such that} \ (x,y) \in \widehat{\Omega}.$$

Moreover, if $h(\bar{x}, \bar{y}) = 0$ and $(\bar{x}, \bar{y})$ is a regular point of problem (3), then $(\bar{x}, \bar{y})$ is a first-order stationary point of (3).

We note that $\widehat{\Omega}$ is the intersection of two compact boxes, one half-space and p hyperplanes, and so it is a compact polyhedral convex set. Hence, the constant positive linear dependence (CPLD) condition always holds, and so any stationary point of (6) satisfies the KKT conditions; see, e.g., [2–4]. Moreover, since the function $L(x, y, \hat{\lambda}, \tau)$ in (5) is continuously differentiable, if the feasible set $\widehat{\Omega}$ is nonempty then problem (6) attains global solutions. Therefore, if at every outer iteration of Algorithm AAL-SPG the SPG method converges to a tol_1-global solution of (6), then stronger convergence results can be established. In that case, if the penalty parameter τ_k remains bounded for all k, then the limit point $(\bar{x}, \bar{y})$ is feasible (i.e., $(\bar{x}, \bar{y}) \in \widehat{\Omega}$ and $h(\bar{x}, \bar{y}) = 0$) and $(\bar{x}, \bar{y})$ is a tol_1-global solution

of problem (3), that is $f(\bar{x}) \leq f(x) + tol_1$ for all feasible (x, y). These results, as well as many others related to different assumptions and also to practical issues, are analyzed in detail in [12, Chapters 5, 6, and 7].

5. Computational results

To add understanding and illustrate the feasibility and effectiveness of the AAL-SPG algorithm to solve problem (3) we present several computational experiments. All the experiments were performed using Matlab R2022a with double precision on an Intel® Quad-Core i7-1165G7 at 4.70 GHz with 16GB of RAM memory.

In our implementation of Algorithm AAL-SPG we set the initial vector y as follows: $(y_0)_i = 0$ if $|(x_0)_i| \geq 10^{-8}$ and 1 otherwise. The choice of the initial vector x_0 will be described below. We set $tol_1 = 0.5 \times 10^{-8}$, $tol_2 = 10^{-8}$, $\lambda_{reg} = 0.01$, $\hat{\lambda}_{\min} = -10^{10}$, $\hat{\lambda}_{\max} = 10^{10}$, and $\hat{\lambda}_0 = e - y_0$. For choosing the related 2-norm regularization parameter λ_{reg} we tried a simple discrete (cross-validation) strategy, starting from 1 and dividing by 10 each time until we reached 10^{-4}. For each choice we ran all our experiments and noted that $\lambda_{reg} = 10^{-2}$ induced the best overall performance. Concerning the nonmonotone line search strategy used by the SPG method, we set $\gamma = 10^{-4}$ and $M = 10$. Each SPG iteration uses Dykstra's alternating projection scheme to obtain the required projection onto $\widehat{\Omega}$, and this internal iterative process is stopped when (9) is satisfied with $\varepsilon = 10^{-9}$. To update τ_k we consider, in Step 2b, $\tau_k = \min\left(\delta_k \tau_{k-1}, 10^8\right)$. We note that at any iteration $k \geq 1$, Step 1 of Algorithm AAL-SPG starts from (x_k, y_k), which is the previous solution of (6), obtained using τ_k. Let us recall that to stop the SPG iterations we use (7). It is worth recalling that if $\|P_{\widehat{\Omega}}((x, y) - \nabla f(x, y)) - (x, y)\| = 0$, then $(x, y) \in \widehat{\Omega}$ is stationary for problem (6); see, e.g., [13,15].

In general, for our experiments, we consider a linear regression model $b = Ax + \xi$ with response vector $b \in \mathbb{R}^m$. The vector b and the matrix $A \in \mathbb{R}^{m \times n}$ are given, while the vector $\xi \in \mathbb{R}^m$ is unknown. In particular, we allow deviations from equality, that is, we consider that $Ax - b = \xi = \sigma\epsilon$ and we choose $0 \leq \sigma < \max\{|b_1|, \ldots, |b_m|\} = \|b\|_\infty$ as analyzed in [33]. In Matlab, we set $\xi = \sigma\epsilon = normrnd(0, \sigma, [1, n])$. Hence, σ is the standard deviation, which controls the effect of the noise. Hence, by setting $\sigma = 0$ we can explore the behavior of our approach without noise. In our experiments, we explore different values of σ in the set $\sigma \in \{0, 0.01, 0.5, 1, 1.5\}$.

To obtain the initial point x_0 we apply the SPG scheme, starting at the vector e of all ones, and stopping it when (7) is satisfied with $tol = 10^{-5}$ (low precision), to the unconstrained function:

$$\frac{1}{2}\|Ax - b\|_2^2 + 0.01\,\|x\|_2^2 + \tilde{\lambda} \sum_{i=1}^{n} g_i(x),$$

where $g_i(x) = x_i\, tanh(x_i/\mu)$ with $\mu = 100$ and $\tilde{\lambda} = 0.1\,\|A^\top b\|_\infty$. Once the SPG scheme stops, say at x^*_{temp}, we set $(x_0)_i = 0$ if $|(x^*_{temp})_i| < 10^{-5}$, and $(x_0)_i = (x^*_{temp})_i$ otherwise. The main motivation for this choice is to obtain a suitable sparsity in x_0 by imposing an approximated continuously differentiable Elastic-Net low-cost regularization strategy to the minimization of $(1/2)\|Ax - b\|_2^2$. Concerning the smoothness of the considered function, we note that $g_i'(x) = (x/\mu) * sech^2(x/\mu) + tanh(x/\mu)$. The fact that $\sum_{i=1}^n g_i(x)$ is a good approximation of $\|x\|_1$ is proposed and discussed in [5,6]. This specialized choice of x_0 has proved to be effective in all our numerical experiments.

To explore the behavior of the AAL-SPG algorithm when solving (6), we will vary the parameters $\sigma \geq 0$ and α. In addition, for each problem we indicate the maximum number of iterations $\max_{iter}$ and the values of m and n. Once the algorithm stops, we report the final penalization parameter τ, the number of outer iterations $Iter$, the obtained residual norm $(1/2)\|Ax^* - b\|_2^2$, and the cardinality of the solution x^*, i.e., $\|x^*\|_0$. For each experiment we also report the residual values that can be considered satisfactory. To do this, we plot the considered parameters α versus the residual norms obtained, and identify the so-called elbow point or L-curve corner, at which the curvature of that graph reaches a maximum, that is, the point in which the slope of the curve changes drastically. Indeed, in most cases, that curve exhibits a typical L shape, and the corner of the L represents a compromise between the imposed sparsity and the quality of the solution obtained. This idea for detecting residual values that can be considered satisfactory is clearly inspired by the well-known L-curve strategy, originally developed to identify the convenient Tikhonov regularization parameter for ill-posed problems [30].

In Subsection 5.1 we study a determined case ($m = n$) and also an over-determined case ($n < m$). We work with under-determined linear systems in Subsection 5.2, that is, experiments with $n > m$. Finally, in Subsection 5.3 we describe some experiments in which we explore a collaboration between our approach and the use of the L_1-norm.

5.1. Determined and over-determined linear systems

Experiment 1. Here, to solve problem (6), we consider one real example with bounds $l_i = -10$ and $u_i = 10$ for all i, and such that the matrix $A = \tilde{A} \in \mathbb{R}^{12 \times 12}$ is the matrix of the countries data set, which is available at https://hastie.su.domains/ElemStatLearn/datasets/countries.data and fully described in [36]. The method of obtaining it was by distributing a questionnaire in a political science class and asking students to provide subjective dissimilarity coefficients between 12 countries. The students' averages were used to obtain the final dissimilarity coefficients. The output of that process is the following square matrix

Table 1
Performance of the AAL-SPG algorithm when A and b are given in Experiment 1 for $n = 12$ and $m = 12$, $\max_{iter} = 200$, and different values of σ and α.

σ	α	τ	$Iter$	$(1/2)\|Ax^* - b\|_2^2$	$\|x^*\|_0$
0	1	10^8	18	2.4×10^{3}	1
	2	10^8	17	605.7	2
	3	10^8	18	98.6	3
	4	10^8	21	38.4	4
	5	10^8	27	11.3	5
	6	276.0	15	6.8×10^{-7}	6
	7	180.4	16	6.7×10^{-7}	6
	8	39.6	25	1.3×10^{-7}	6
0.01	1	10^8	18	2.4×10^{3}	1
	2	10^8	17	607.6	2
	3	10^8	18	98.7	3
	4	10^8	23	38.2	4
	5	10^8	23	11.3	5
	6	50.8	26	1.17×10^{-6}	6
	7	48.3	28	1.15×10^{-6}	6
	8	353.1	18	1.12×10^{-6}	6
1.5	1	10^8	17	2.5×10^{3}	1
	2	10^8	17	710.2	2
	3	10^8	18	127.8	3
	4	10^8	20	57.6	4
	5	10^8	20	15.3	5
	6	10^8	24	0.65	6
	7	5.8×10^{6}	29	0.25	6
	8	1.5×10^{6}	31	0.18	6

$$\tilde{A} = \begin{bmatrix} Belgium & Brazil & China & Cuba & Egypt & France & India & Israel & USA & URSS & Yugoslavia & Zaire \\ 0.00 & 5.58 & 7.00 & 7.08 & 4.83 & 2.17 & 6.42 & 3.42 & 2.50 & 6.08 & 5.25 & 4.75 \\ 5.58 & 0.00 & 6.50 & 7.00 & 5.08 & 5.75 & 5.00 & 5.50 & 4.92 & 6.67 & 6.83 & 3.00 \\ 7.00 & 6.50 & 0.00 & 3.83 & 8.17 & 6.67 & 5.58 & 6.42 & 6.25 & 4.25 & 4.50 & 6.08 \\ 7.08 & 7.00 & 3.83 & 0.00 & 5.83 & 6.92 & 6.00 & 6.42 & 7.33 & 2.67 & 3.75 & 6.67 \\ 4.83 & 5.08 & 8.17 & 5.83 & 0.00 & 4.92 & 4.67 & 5.00 & 4.50 & 6.00 & 5.75 & 5.00 \\ 2.17 & 5.75 & 6.67 & 6.92 & 4.92 & 0.00 & 6.42 & 3.92 & 2.25 & 6.17 & 5.42 & 5.58 \\ 6.42 & 5.00 & 5.58 & 6.00 & 4.67 & 6.42 & 0.00 & 6.17 & 6.33 & 6.17 & 6.08 & 4.83 \\ 3.42 & 5.50 & 6.42 & 6.42 & 5.00 & 3.92 & 6.17 & 0.00 & 2.75 & 6.92 & 5.83 & 6.17 \\ 2.50 & 4.92 & 6.25 & 7.33 & 4.50 & 2.25 & 6.33 & 2.75 & 0.00 & 6.17 & 6.67 & 5.67 \\ 6.08 & 6.67 & 4.25 & 2.67 & 6.00 & 6.17 & 6.17 & 6.92 & 6.17 & 0.00 & 3.67 & 6.50 \\ 5.25 & 6.83 & 4.50 & 3.75 & 5.75 & 5.42 & 6.08 & 5.83 & 6.67 & 3.67 & 0.00 & 6.92 \\ 4.75 & 3.00 & 6.08 & 6.67 & 5.00 & 5.58 & 4.83 & 6.17 & 5.67 & 6.50 & 6.92 & 0.00 \end{bmatrix}.$$

We start by setting $\tilde{x} = (0,0,1,0,0,0,1,1,0,1,1,1)^\top$ (notice that $\|\tilde{x}\|_0 = 6$), and then we set $b = A\tilde{x}$. It is possible to solve $Ax = b$ in Matlab by using $x = linsolve(A, b)$, obtaining $\tilde{x}$ as the solution of the linear system. We note in Table 1 that, for $\sigma = 0$ and $\alpha \geq 6$, we obtain exactly $\tilde{x}$. Moreover, choosing $\alpha > 6$ we obtain a solution vector whose cardinality is again 6. Applying the L-curve strategy described above, we note that for $(1/2)\|Ax^* - b\|_2^2 \leq 10^{-6}$ the solutions can be considered satisfactory, that corresponds to choosing $\alpha \geq 6$ in the cardinality constraint.

Experiment 2. [Prostate cancer data set] This data set is considered in [47,48,51], and available at https://hastie.su.domains/ElemStatLearn/datasets/prostate.data, to investigate the correlation between the level of prostate specific antigen (*lpsa*) and a number of clinical measures obtained from $m = 97$ patients, who were about to receive a radical prostatectomy. The $n = 8$ factors are: log cancer volume (*lcavol*), log prostate weight (*lweight*), age, log benign prostatic hyperplasia amount (*lbph*), seminal vesicle invasion (*svi*), log capsular penetration (*lcp*), Gleason score (*gleason*), and percentage Gleason scores 4 or 5 (*pgg45*). Therefore, we consider the following linear model

$$x_1\, lcavol + x_2\, lweight + x_3\, age + x_4\, lbph + x_5\, svi + x_6\, lcp + x_7\, gleason + x_8\, pgg45 = lpsa.$$

For validation of the considered model, that is, to check if the obtained solution vector x^* is sufficiently good to classify a new patient with prostate cancer, we see if the entries different from zero in x^* are significant features to obtain the lpsa. We proceed by dividing the data set into two subsets: approximately 80% for training and the rest for testing. The vector $b_{initial}$ is equal to $lpsa$ and the matrix $A_{initial}$ consists of 8 columns and 97 rows. The initial 77 rows of $A_{initial}$ and $b_{initial}$ are our training set, that is, A and b for our considered optimization problem (6), for which we set $l_i = -10^5$ and $u_i = 10^5$ for all i.

Table 2
Performance of the AAL-SPG algorithm when $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$ are given in Experiment 2 for $n = 8$ and $m = 77$, $\max_{iter} = 1000$, $\sigma = 0$, and different values of α.

α	τ	$Iter$	$(1/2)\|Ax^* - b\|_2^2$	$\|x^*\|_0$	Selected features
1	10^8	524	23.3	1	2
2	10^8	723	23.3	1	2
3	10^8	117	15.0	2	1, 2
4	10^8	235	15.0	2	1, 2
5	10^8	93	14.2	3	1, 2, 4
6	10^8	359	13.9	4	1, 2, 4, 5
7	10^8	45	13.2	7	1, 2, 3, 4, 5, 6, 7
8	10^8	172	13.1	7	1, 2, 3, 4, 5, 6, 7

Table 3
Performance of the AAL-SPG algorithm applied to the test set of Experiment 2 ($n = 8$ and $m = 20$), when $\sigma = 0$ and $\alpha = 5$.

$A_{test}x^*$	b_{test}
3.02	3.44
3.01	3.46
2.88	3.51
2.00	3.52
2.61	3.53
3.13	3.57
2.84	3.57
2.33	3.59
3.25	3.63
2.53	3.68
2.54	3.71
3.59	3.98
2.62	3.99
3.19	4.03
3.00	4.13
3.22	4.39
3.63	4.68
3.03	5.14
3.19	5.48
3.52	5.58

The results are reported in Table 2. We note that for any choice of α the first and second entries of x^* are the most important relevant features. In addition, the fourth entry is the relevant feature to take into account if we accept 3 features to represent the model. The fact that those 3 features appear in the chosen list for all values of $\alpha \geq 3$ is a clear indication that those are the 3 important features to be considered for recovering the model. In fact, applying the L-curve strategy, we observe that for $(1/2)\|Ax^* - b\|_2^2 \leq 15$ the solutions can be considered satisfactory, which corresponds to choosing $\alpha \geq 2$ in the cardinality constraint.

Now, we proceed for the test phase, where we take the test set (the vector b_{test} and A_{test}, i.e., the last $20 = 97 - 77$ entries of b and A respectively), for which we already know a prior its classification, and check if x^* correctly classifies such a set. Based on the previous observations, we take into account the 3 most important features (entries 1, 2, and 4) and for that we use the vector x^* obtained for $\alpha = 5$. The result of the *lpsa* at x^*, that is, the entries of the vector $A_{test}x^*$ (as well as the entries of the vector b_{test}) are reported in Table 3. Note that in each entry we obtain values close to b_{test}, which is the true value of *lpsa*. Therefore, the solution x^* obtained of the AAL-SPG algorithm for all α is effective for classifying new patients.

Let us remark that the scheme used in this experiment of considering a training data set and a test data set is very useful in variable selection, and has a favorable impact on machine learning applications.

5.2. Under-determined linear systems

Experiment 3. In this case, $l_i = -10$ and $u_i = 10$ for all i, and A is synthetically constructed from a multivariate normal distribution $N_n(\mu, \Sigma)$ with $\mu = 0$. This distribution is a generalization of the univariate normal distribution to two or more variables. It has two parameters, a mean vector μ and a covariance matrix Σ, that are analogous to the mean and variance parameters of a univariate normal distribution. The choice of different Σ has no major impact on the performance of our approach, and here we consider the correlation matrix $\Sigma = I_{30}$. We consider a true-features vector $\tilde{x} \in \mathbb{R}^{30}$ given by

$$\tilde{x} = (0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0)^\top,$$

with $\|\tilde{x}\|_0 = 10$. Then the response $b \in \mathbb{R}^m$ is calculated as $b = A\tilde{x}$. In addition, we construct the noise $\xi = \sigma\epsilon$ from a univariate normal distribution $N(0, \sigma)$. In Matlab, we set $\xi = \sigma\epsilon = normrnd(0, \sigma, [1, n])$. The matrix $A \in \mathbb{R}^{15\times 30}$ has full rank $rank(A) = 15$, $cond(A) = 8$, $\sigma_{\max}(A) \approx 9.2$ and $\sigma_{\min}(A) \approx 1.1$. The numerical results are reported in Table 4, and in Fig. 1 we show the value of α versus the residual norm $(\frac{1}{2}\|Ax^* - b\|_2^2)$ without noise ($\sigma = 0$). The behavior observed in Fig. 1, showing a monotone increase in the residual norm when α decreases, has been observed in all our experiments. Using the L-curve strategy, we can also observe in Fig. 1 that for $(1/2)\|Ax^* - b\|_2^2 \leq 4.5$ the solutions can be considered satisfactory, that corresponds to selecting $\alpha \geq 6$.

Experiment 4. [Colon cancer data set] The matrix $A \in \mathbb{R}^{62\times 2000}$, and vector b are obtained in https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html. This data set was described and used in [1]. In this case, $l_i = -100$ and $u_i = 100$ for all i. The results are reported in Table 5. Its worth noticing that if α is reduced then $\|x^*\|_0$ is also reduced, and the selected set of variables, as expected, is a subset of the previous (larger) selected set. We note, applying the L-curve strategy, that for $(1/2)\|Ax^* - b\|_2^2 \leq 12$ the solutions can be considered satisfactory, which corresponds to choosing $\alpha \geq 10$.

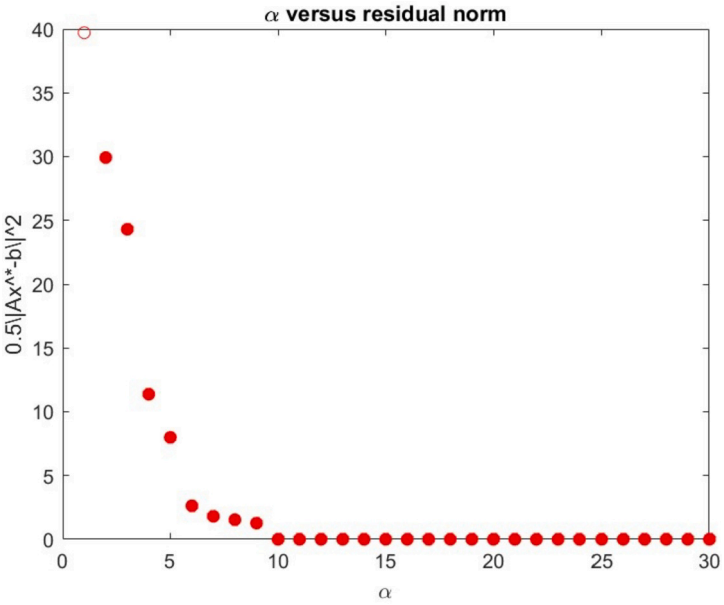

Fig. 1. Value of α versus $\frac{1}{2}\|Ax^* - b\|^2$ with $\sigma = 0$, by using the AAL-SPG algorithm applied to Experiment 3.

Table 4
Performance of the AAL-SPG algorithm applied to (6) when A and b are given in Experiment 3 for $n = 30$ and $m = 15$, $\max_{iter} = 500$, and different values of σ and α.

σ	α	τ	$Iter$	$(1/2)\|Ax^* - b\|_2^2$	$\|x^*\|_0$
0	1	2.8×10^3	23	39.7	1
	2	650.3	22	29.9	2
	3	10^8	34	24.3	3
	4	1.7×10^3	23	13.8	4
	5	1.0×10^3	40	7.6	5
	6	10^8	65	4.47	6
	7	10^8	63	3.10	7
	8	10^8	63	1.00	8
	10	6.2×10^3	42	0.62	10
	15	10^8	96	0.02	13
	20	2.3×10^3	500	3.7×10^{-3}	18
	25	472.6	500	1.4×10^{-3}	19
	29	20.2	500	1.4×10^{-3}	21
0.1	1	1.2×10^7	27	39.7	1
	5	10^8	59	7.8	5
	10	10^8	27	0.65	10
	15	1.3×10^5	43	0.04	14
0.5	1	10^8	25	40.2	1
	5	277.3	34	8.1	5
	10	5.6×10^4	29	0.73	10
	15	1.1×10^4	66	0.06	13

Table 5
Behavior of the AAL-SPG scheme when $A \in \mathbb{R}^{62\times2000}$ (colon cancer dataset) and $\max_{iter} = 700$ for different values of α.

α	τ	$Iter$	$(1/2)\|Ax^* - b\|_2^2$	$\|x^*\|_0$
3	10^8	700	30.43773	2
5	10^8	700	29.91580	4
15	10^8	700	11.85664	10
30	10^8	700	7.238638	23
40	10^8	700	3.387789	33

Table 6
Performance of the AAL-SPG algorithm when A and b are given in Experiment 5 for $n = 60$ and $m = 12$, $\max_{iter} = 400$, and $\sigma = 0$; and also for different values of σ when $\alpha = 5$ and 25.

σ	α	τ	$Iter$	$(1/2)\|Ax^* - b\|_2^2$	$\|x^*\|_0$
0	5	10^8	19	441.8	5
	6	10^8	63	441.8	5
	7	10^8	98	441.8	5
	10	10^8	21	94.7	10
	20	10^8	400	38.4	15
	30	71.1	75	2.4×10^{-6}	25
	40	687.0	60	2.4×10^{-6}	25
	55	490.5	400	2.2×10^{-6}	30
0.1	5	10^8	79	458.6	4
	25	10^8	400	0.03	19
0.5	5	10^8	400	459.4	4
	25	10^8	400	57.1	12
1	5	10^8	400	459.5	4
	25	10^8	400	56.1	13
1.5	5	10^8	400	443.4	5
	25	10^8	400	97.7	10

Table 7
Performance of the AAL-SPG algorithm when A and b are described in Experiment 6, and $\max_{iter} = 500$. In here, the percentage of null entries at x^* is denoted by $nc(x^*)$.

Label	α	τ	$Iter$	$(1/2)\|Ax^* - b\|_2^2$	$\|x^*\|_0$	$nc(x^*)$
SC6	100	10^8	500	14.2	100	97.56%
	200	10^8	183	11.7	163	96.02%
	400	10^8	107	8.6	325	92.07%
	1200	10^8	74	7.2	443	89.18%
	1400	10^8	500	7.2	443	89.18%
	2100	10^8	20	6.9	453	88.94%
	2400	10^8	20	6.9	453	88.94%
SC8	100	10^8	347	2.21	15	99.68%
	300	10^8	164	2.1	17	99.63%
	900	10^8	315	1.7	20	99.57%
	1500	10^8	101	1.1	25	99.47%
	1600	10^8	110	0.65	40	99.15%
	2100	10^8	152	0.29	120	97.45%
	2500	10^8	110	0.28	145	96.92%
	2900	10^8	174	0.27	202	95.70%
	3200	10^8	88	0.26	246	94.77%
	3300	10^8	71	0.26	249	94.70%
	3400	10^8	70	0.26	272	94.22%
	3600	10^8	79	0.25	347	92.62%
	4000	5×10^7	43	0.25	474	89.92%

Experiment 5. We consider the matrix $A = [\tilde{A}\tilde{A}\tilde{A}\tilde{A}\tilde{A}] \in \mathbb{R}^{12\times60}$, where $\tilde{A}$ is the matrix of the Countries data set, already described in Experiment 1. In this case, $l_i = -2$ and $u_i = 2$ for all i. We obtain b such that $b = A\tilde{x}$ with $\tilde{x}_1 = \tilde{x}_5 = \tilde{x}_{10} = \tilde{x}_{15} = \tilde{x}_{25} = \tilde{x}_{35} = 1$ and for all other entries $\tilde{x}_i = 0$, that is, $\|\tilde{x}\|_0 = 6$. For this experiment, applying the L-curve strategy, we notice that for $(1/2)\|Ax^* - b\|_2^2 \leq 38.5$ the solutions can be considered satisfactory, that corresponds to choosing $\alpha \geq 15$ (Table 6).

Experiment 6. We consider two problems: SC6 [7] and SC8 [41]. For SC6, with $(n_{row}, n_{col}) = (1024, 4096)$, $\sigma_{\max}(A) = 1.5$, $\sigma_{\min}(A) = 0.5$, and $\text{rank}(A) = 1024$. For SC8, with $(n_{row}, n_{col}) = (64, 4702)$, $\sigma_{\max}(A) = 46.9$, $\sigma_{\min}(A) = 5.8e-16$, and $\text{rank}(A) = 64$. In both case, $l_i = -10^5$ and $u_i = 10^5$ for all i. The obtained results are reported in Table 7. Applying the L-curve strategy, we note that, in the case of the SC6 problem, for $(1/2)\|Ax^* - b\|_2^2 \leq 9$ the solutions can be considered satisfactory, that corresponds to $\|x^*\|_0 \geq 325$. In the case of the SC8 problem, for $(1/2)\|Ax^* - b\|_2^2 \leq 0.65$ the solutions can be considered satisfactory, which corresponds to fixing $\alpha \geq 40$.

At this point, it is worth making some general comments about the behavior of the AAL-SPG algorithm in the presence of noise. It can be observed that in some experiments, for some small values of the forced cardinality α (i.e., when $\|x^*\|_0$ is small), in the presence of noise the obtained residual norm is not satisfactory. However, in those cases it turns out that for the same small value

Table 8
Using $\alpha = \alpha_{lasso}$, obtained after solving (12), to apply the AAL-SPG algorithm when $A \in \mathbb{R}^{62\times2000}$ (Colon cancer), $A \in \mathbb{R}^{1024\times4096}$ (SC6), and $A \in \mathbb{R}^{64\times4702}$ (SC8).

	Lasso		AAL-SPG (α_{lasso})		AAL-SPG ($\alpha_{lasso}+1$)	
Problem	$\|x^*_{lasso}\|_0$	res_{lasso}	$\|x^*\|_0$	res_{x^*}	$\|x^*\|_0$	res_{x^*}
Colon	32	5.21	30	3.49	32	3.41
SC6	10	34.8	10	22.5	11	21.2
SC8	13	2.59	12	2.29	13	2.28

of $\|x^*\|_0$, and without noise, the residual norm obtained is also not satisfactory. On the contrary, we also observe that when the residual norm is satisfactory without noise, the obtained residual norm by adding noise increases (as expected) by a magnitude of the same order of the imposed noise level, indicated in our tables by $\sigma > 0$.

5.3. Collaboration with the L_1 norm

Applying the L_1-norm regularization strategy, to obtain a sparse solution, can be seen as a preliminary step to detect a convenient sparsity factor, i.e., a meaningful choice of α to be used in our approach. In that sense, in here, we discuss a possible collaboration between Lasso (and also Elastic Net) and the AAL-SPG algorithm for variable selection problems. Using this collaboration, an approximate value of the number of important features or variables can be predicted.

Indeed, Lasso [48], is known to produce sparse solutions when applied to

$$\min_x \frac{1}{2}\|Ax-b\|_2^2 + \lambda_{lasso}\|x\|_1,$$

as long as $\lambda_{lasso} > 0$ is a properly chosen regularization parameter. In here, we set $\lambda_{lasso} = 0.1 * \|A^\top b\|_\infty$, that is, $\lambda_{lasso} = 0.1\|\nabla \widehat{f}(0)\|_\infty$ as suggested in [42]. Similarly, the Elastic Net strategy, originally described in [51], can be seen as an extension of Lasso combining the L_1 and L_2 norms to solve

$$\min_x \frac{1}{2}\|Ax-b\|_2^2 + \lambda_{reg}\|x\|_2^2 + \lambda_{lasso}\|x\|_1. \tag{12}$$

For the proposed collaboration, we start at the same initial guess used by the AAL-SPG algorithm and solve the unconstrained problem (12) to obtain x^*_{lasso}, with cardinality α^*_{lasso}, and residual norm $res_{lasso} = (1/2)\|Ax^*_{lasso} - b\|_2^2$.[1] Then, setting $\alpha = \alpha^*_{lasso}$ and also $\alpha = \alpha^*_{lasso} + 1$ we apply the AAL-SPG algorithm to solve (6), starting at its standard initial guess, to obtain the solution x^* and the residual $res_{x^*} = (1/2)\|Ax^* - b\|_2^2$ in both cases.

The results obtained in this collaborative experiment are reported in Table 8 for 3 of the already considered data sets. It can be observed that for roughly the same number of zero entries (sparsity) in the obtained solution, the one produced by the AAL-SPG algorithm consistently has a lower residual value than the one obtained when solving (12), that is, the solution obtained by AAL-SPG represents a better recovery of the model.

An additional important piece of information, closely related to the recovery of the model, can be extracted by a close inspection of the selected entries (or features) obtained by the methods reported in Table 8. Let us illustrate this issue by considering problem SC6. For SC6, the 10 nonzero entries $(x^*_{lasso})_i$, selected by solving (12) are $i = 1, 3, 65, 66, 68, 129, 322, 323, 2524, 2941$, and the obtained residual is $res_{lasso} = 34.8$. When using $\alpha = \alpha^*_{lasso}$, the 10 nonzero entries $(x^*)_i$ obtained by the AAL-SPG algorithm are $i = 1, 3, 65, 66, 68, 129, 131, 200, 322, 323$, and the obtained residual is $res_{x^*} = 22.5$. Moreover, when using $\alpha = \alpha^*_{lasso} + 1$, the 11 nonzero entries $(x^*)_i$ obtained by the AAL-SPG algorithm are $i = 1, 3, 65, 66, 68, 129, 131, 200, 322, 323, 681$, and the obtained residual is $res_{x^*} = 21.2$. It is clear, after a close inspection, that the entries 131 ans 200 (selected by AAL-SPG) are more relevant to the model than the entries 2524 and 2941 (selected by Lasso). In fact, by swapping those 2 entries and leaving the remaining ones the same, the residual was significantly reduced from 34.8 to 22.5. The additional entry selected by AAL-SPG when $\alpha = \alpha^*_{lasso} + 1$ is the one at $i = 681$, which represents a slight additional reduction of the residual from 22.5 to 21.2. Taking into account all these facts, we conclude that the 8 most relevant features (selected by all methods) are those found in the positions $i = 1, 3, 65, 66, 68, 129, 322$ and 323.

We close this section with some recommendations on how to use the AAL-SPG algorithm, combined with some of the other features that have been described throughout this section, to solve a variable selection problem. For the validation of the linear model, specially when dealing with machine learning applications, the splitting of the data set described in our Experiment 2 is recommendable. Once the model is established, as we have already mentioned, the choice of $1 \le \alpha < n$ is fundamental in our algorithmic framework. For that, unless the specific application offers some natural information about the expected sparsity factor, the collaboration with the L_1 norm is advisable to obtain a small range of suitable values for α. For each of those few α values the AAL-SPG algorithm can be applied. The initial parameters required by the AAL-SPG algorithm can be chosen as described in the first 4 paragraphs of Section 5 and also in the first item of Remark 1. In particular, to choose the parameter $\lambda_{reg} > 0$, more advanced strategies can be used taking into account the specific available data sets; see, e.g., [38,49] and references in there. The result

[1] The solution of (12) is obtained using the code in Julia described in [42], which is available upon request.

obtained from the AAL-SPG algorithm (i.e., the selected variables), for each suitable value of α, offers very valuable information. It is recommendable to inspect and compare them to extract the most desirable common features, as discussed and illustrated in the previous paragraph of this section, for 3 given specific data sets.

6. Conclusions

The direct imposition of a cardinality constraint (using the L_0-norm), to force a preestablished sparsity while ensuring that the most relevant features are detected, has always been a tempting idea for solving variable selection problems. However, due to the non-convexity and discontinuity of the L_0-norm, the resulting optimization formulations are known to be intractable.

In this work we take advantage of a formulation proposed in [21] that allows us to impose the desired cardinality constraint within a continuous constrained optimization problem, at the price of doubling the dimension of the workspace, since it needs to include an auxiliary vector $y \in \mathbb{R}^n$ of the same dimension of the original variable $x \in \mathbb{R}^n$. The feasible set of the obtained formulation involves easy-to-project convex sets except for the Hadamard condition between the two variables ($x \circ y = 0$), which represents a non-convex set. Our algorithmic proposal removes only the Hadamard condition, from the list of constraints, and incorporates it into the objective function via an augmented Lagrangian approach. As a consequence, the augmented Lagrangian subproblems can be solved combining effective and low-cost optimization schemes (namely Dykstra's algorithm and the SPG method).

The algorithm that emerges from this novel combination is described in detail and is applied to a variety of under and over determined real and synthetic feature selection problems. Now, in general, predicting the number of relevant variables is important (see, e.g., [19,52]), i.e., choosing the right value of the entry parameter α in the AAL-SPG algorithm. For that, we also discuss a collaborative strategy that uses the L_1-norm (Lasso or Elastic Net) to predict a convenient sparsity factor $\alpha \in \mathbb{N}$ ($1 \leq \alpha < n$) that needs to be imposed via the cardinality constraint in our approach. In our results, it is consistently observed that by reducing α, the number of non-zero entries in the obtained solution is also reduced. Better yet, if we apply our approach with sparsity factors α_1 and α_2, where $\alpha_1 < \alpha_2$, the set of selected variables using α_1 is a proper subset of those selected when using α_2. As a consequence, the residual obtained when using α_2 is reduced as compared to the residual obtained when using α_1. In other words, by imposing the L_0-norm constraint in our algorithmic proposal, we observe that the recovery of the model is improved as the number of selected variables increases. This coherency is welcome when solving variable selection problems.

Declaration of competing interest

No potential conflict of interest was reported by the authors.

Data availability

The codes and data sets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Acknowledgements

We would like to thank two anonymous reviewers for their constructive comments and suggestions that helped us to improve the final version of this paper. We would like to thank Dr. Paulo J. S. Silva from University of Campinas, Brazil, for providing us with all the Julia routines needed to test the Lasso strategy in Section 5.3. We would also like to thank Dr. Marta Lopes and Dr. Mina Norouzirad from the Center NovaMath at NOVA University of Lisbon, Portugal, for several insightful discussions on regularization strategies for model recovery.

Funding

The first author was financially supported by the Provincial Secretariat for Higher Education and Scientific Research of Vojvodina, grant number 142-451-2593/2021-01/2. The second author was financially supported by Fundação para a Ciência e a Tecnologia (FCT) (Portuguese Foundation for Science and Technology) under the scope of the projects UIDB/MAT/00297/2020, UIDP/MAT/00297/2020 (Centro de Matemática e Aplicações), and UI/297/2020-5/2021. The third author was financially supported by the Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) under the scope of the projects UIDB/MAT/00297/2020 and UIDP/MAT/00297/2020 (Centro de Matemática e Aplicações), and PTDC/CCI-BIO/4180/2020.

References

[1] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Cell Biol. 96 (1999) 6745–6750.

[2] R. Andreani, E.G. Birgin, J.M. Martínez, M.L. Schuverdt, On augmented Lagrangian methods with general lower-level constraints, SIAM J. Optim. 18 (4) (2007) 1286–1309.

[3] R. Andreani, E.G. Birgin, J.M. Martínez, M.L. Schuverdt, Augmented Lagrangian methods under the constant positive linear dependence constraint qualification, Math. Program. 111 (2008) 5–32.

[4] R. Andreani, J.M. Martínez, M.L. Schuverdt, On the relation between the constant positive linear dependence condition and quasinormality constraint qualification, J. Optim. Theory Appl. 125 (2005) 473–485.

[5] Y.J.J. Bagul, A smooth transcendental approximation to $|x|$, Int. J. Math. Sci. Eng. Appl. 11 (II) (2017) 213–217.
[6] Y.J.J. Bagul, B.K. Khairnar, A note on smooth transcendental approximation to $|x|$, Palest. J. Math. 10 (2) (2021) 644–646.
[7] E. van den Berg, M.P. Friedlander, G. Hennenfent, F.J. Herrmann, R. Saab, Ö. Yilmaz, SPARCO: a testing framework for sparse reconstruction, Technical Report TR-2007-20, Department of Computer Science, University of British Columbia, Vancouver, 2007.
[8] D.P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, New York, 1982.
[9] D. Bertsimas, R. Shioda, Algorithm for cardinality-constrained quadratic optimization, Comput. Optim. Appl. 43 (2009) 1–22.
[10] D. Bertsimas, A. King, R. Mazumder, Best subset selection via a modern optimization lens, Ann. Stat. 44 (2) (2016) 813–852.
[11] D. Bienstock, Computational study of a family of mixed-integer quadratic programming problems, Math. Program. 74 (1996) 121–140.
[12] E.G. Birgin, J.M. Martínez, Practical Augmented Lagrangian Methods for Constrained Optimization, SIAM, Philadelphia, 2014.
[13] E.G. Birgin, J.M. Martínez, M. Raydan, Nonmonotone spectral projected gradient methods on convex sets, SIAM J. Optim. 10 (2000) 1196–1211.
[14] E.G. Birgin, J.M. Martínez, M. Raydan, Algorithm 813: SPG – software for convex-constrained optimization, ACM Trans. Math. Softw. 27 (2001) 340–349.
[15] E.G. Birgin, J.M. Martínez, M. Raydan, Inexact spectral projected gradient methods on convex sets, IMA J. Numer. Anal. 23 (2003) 539–559.
[16] E.G. Birgin, J.M. Martínez, M. Raydan, Spectral projected gradient methods: review and perspectives, J. Stat. Softw. 60 (3) (2014).
[17] E.G. Birgin, M. Raydan, Robust stopping criteria for Dykstra's algorithm, SIAM J. Sci. Comput. 26 (2005) 1405–1414.
[18] J.P. Boyle, L. Dykstra, A method for finding projections onto the intersections of convex sets in Hilbert spaces, in: R. Dykstra, T. Robertson, F.T. Wright (Eds.), Advances in Order Restricted Statistical Inference, in: Lecture Notes in Statistics, vol. 37, Springer, New York, 1986, pp. 28–47.
[19] G.E.P. Box, R.D. Meyer, An analysis for unreplicated fractional factorials, Technometrics 28 (1) (1986) 11–18.
[20] A. Buccini, O. De la Cruz Cabrera, C. Koukouvinos, M. Mitrouli, L. Reichel, Variable selection in saturated and supersaturated designs via l_p-l_q minimization, Commun. Stat., Simul. Comput. 52 (9) (2023) 4326–4347, https://doi.org/10.1080/03610918.2021.1961151.
[21] O.P. Burdakov, C. Kanzow, A. Schwartz, Mathematical programs with cardinality constraints: reformulation by complementarity-type conditions and a regularization method, SIAM J. Optim. 26 (1) (2016) 397–425.
[22] E. Candes, T. Tao, The Dantzig selector: statistical estimation when p is much larger than n, Ann. Stat. 35 (6) (2007) 2313–2351.
[23] Y. Chen, Y. Ye, M. Wang, Approximation hardness for a class of sparse optimization problems, J. Mach. Learn. Res. 20 (38) (2019) 1–27.
[24] C.M. Costa, D. Kreber, M. Schmidt, An alternating method for cardinality-constrained optimization: a computational study for the best subset selection and sparse portfolio problems, INFORMS J. Comput. 34 (6) (2022) 2968–2988.
[25] M.A. Diniz-Ehrhard, M.A. Gomes-Ruggiero, J.M. Martínez, S.A. Santos, Augmented Lagrangian algorithms based on the spectral projected gradient method for solving nonlinear programming problems, JOTA 123 (3) (2004) 497–517.
[26] D. Donoho, Y. Tsaig, I. Drori, J. Starck, Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit, IEEE Trans. Inf. Theory 58 (2) (2012) 1094–1121.
[27] M. El Guide, K. Jbilou, C. Koukouvinos, A. Lappa, Comparative study of L_1 regularized logistic regression methods for variable selection, Commun. Stat., Simul. Comput. 51 (9) (2022) 4957–4972.
[28] M. El Guide, K. Jbilou, C. Koukouvinos, A. Lappa, Krylov subspace solvers for L_1 regularized logistic regression method, Commun. Stat., Simul. Comput. 52 (6) (2023) 2738–2751, https://doi.org/10.1080/03610918.2021.1914093.
[29] R. Escalante, M. Raydan, Alternating Projection Methods, SIAM, Philadelphia, 2011.
[30] P.C. Hansen, Analysis of discrete ill-posed problems by means of the L-curve, SIAM Rev. 34 (1992) 561–580.
[31] H. Hazimeh, R. Mazumder, Fast best subset selection: coordinate descent and local combinatorial optimization algorithms, Oper. Res. 68 (5) (2020) 1517–1537.
[32] X. Jia, C. Kanzow, P. Mehlitz, G. Wachsmuth, An augmented Lagrangian method for optimization problems with structured geometric constraints, Math. Program. 199 (2023) 1365–1415.
[33] S. Jokar, M.E. Pfetsch, Exact and approximate sparse solutions of underdetermined linear equations, SIAM J. Sci. Comput. 31 (2008) 23–44.
[34] Ch. Kanzow, A.B. Raharja, A. Schwartz, Sequential optimality conditions for cardinality-constrained optimization problems with applications, Comput. Optim. Appl. 80 (2021) 185–211.
[35] Ch. Kanzow, A.B. Raharja, A. Schwartz, An augmented Lagrangian method for cardinality constrained optimization problems, J. Optim. Theory Appl. 189 (2021) 793–813.
[36] L. Kaufman, P. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, New York, 1990.
[37] N. Krejić, E.H.M. Krulikovski, M. Raydan, A low-cost alternating projection approach for a continuous formulation of convex and cardinality constrained optimization, Oper. Res. Forum 4 (2023) 73, https://doi.org/10.1007/s43069-023-00257-w.
[38] C. Koukouvinos, A. Lappa, M. Mitrouli, P. Roupa, O. Turek, Numerical methods for estimating the tuning parameter in penalized least squares problems, Commun. Stat., Simul. Comput. 51 (4) (2022) 1542–1563.
[39] E.H.M. Krulikovski, A.A. Ribeiro, M. Sachine, On the weak stationarity conditions for mathematical programs with cardinality constraints: a unified approach, Appl. Math. Optim. 84 (2021) 3451–3473.
[40] E.H.M. Krulikovski, A.A. Ribeiro, M. Sachine, A comparative study of sequential optimality conditions for mathematical programs with cardinality constraints, J. Optim. Theory Appl. 192 (2022) 1067–1083.
[41] M. Lichman, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2017, http://archive.ics.uci.edu/ml, Last updated 23 July 2017.
[42] R. Lopes, S.A. Santos, P.J.S. Silva, Accelerating block coordinate descent methods with identification strategies, Comput. Optim. Appl. 72 (2019) 609–640.
[43] A. Miller, Subset Selection in Regression, Chapman and Hall, Melbourne, 1990.
[44] B.K. Natarajan, Sparse approximate solutions to linear systems, SIAM J. Comput. 24 (1995) 227–234.
[45] M. Norouzirad, M. Arashi, Preliminary test and Stein-type shrinkage ridge estimators in robust regression, Stat. Pap. 60 (2019) 1849–1882.
[46] A.K.Md.E. Saleh, M. Arashi, R.A. Saleh, M. Norouzirad, Rank-Based Methods for Shrinkage and Selection with Application to Machine Learning, John Wiley and Sons, Inc., New Jersey, 2022.
[47] T.A. Stamey, J.N. Kabalin, J.E. McNeal, I.M. Johnstone, F. Freiha, E.A. Redwine, N. Yang, Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate, II. Radical prostatectomy treated patients, J. Urol. 141 (5) (1989) 1076–1783.
[48] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc., Ser. B 58 (1996) 267–288.
[49] J.R. Winkler, M. Mitrouli, C. Koukouvinos, The application of regularisation to variable selection in statistical modelling, J. Comput. Appl. Math. 404 (2022) 113884.
[50] X. Zeng, X. Sun, D. Li, Improving the performance of MIQP solvers for quadratic programs with cardinality and minimum threshold constraints: a semidefinite program approach, INFORMS J. Comput. 26 (4) (2014) 690–703.
[51] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, J. R. Stat. Soc., Ser. B, Stat. Methodol. 67 (2) (2005) 301–320.
[52] W.R. Zwick, W.F. Velicer, Comparison of five rules for determining the number of components to retain, Psychol. Bull. 99 (1986) 432–442.