Dual formulation of the sparsity constrained optimization problem: application to classification

M. Gaudioso, G. Giallombardo, Jean-Baptiste Hiriart-Urruty

MANUSCRIPT

Dual formulation of the sparsity constrained optimization problem: application to classification

M. Gaudioso[a] and G. Giallombardo[a] and J.-B. Hiriart–Urruty[b]

[a]DIMES, Università della Calabria, 87036 Rende (CS), Italia;[b]Institut de mathématiques, Université Paul Sabatier, 31062 Toulouse, France.

ARTICLE HISTORY
Compiled October 24, 2023

ABSTRACT
We tackle the sparsity constrained optimization problem by resorting to polyhedral k-norm as a valid tool to emulate the ℓ_0-pseudo-norm. The main novelty of the approach is the use of the dual of the k-norm, which allows to obtain a formulation amenable for a relaxation that can be efficiently handled by block coordinate methods. The advantage of the approach is that it does not require the solution of difference-of-convex programs, unlike other k-norm based methods available in the literature. In fact, our block coordinate approach requires, at each iteration, the solution of two convex programs, one of which can be solved in $O(n \log n)$ time. We apply the method to feature selection within the framework of Support Vector Machine classification, and we report the results obtained on some benchmark test problems.

KEYWORDS
Sparse optimization; cardinality constraint; polyhedral k-norm; block coordinate methods

1. Introduction

The sparsity constrained optimization problem consists in minimizing a function $f : \mathbb{R}^n \mapsto \mathbb{R}$, under the constraint that the number of the non-zero components of the solution must not exceed a prefixed integer bound $k > 0$. Denoting by

$$\|\mathbf{x}\|_0 \triangleq \left|\left\{i \in \{1, \ldots, n\} : \, x_i \neq 0\right\}\right|$$

the ℓ_0-pseudo-norm of a vector $\mathbf{x} \in \mathbb{R}^n$, namely, the number of its non-zero components, and letting

$$X_0^k \triangleq \left\{\mathbf{x} \in \mathbb{R}^n : \, \|\mathbf{x}\|_0 \leq k\right\}, \tag{1}$$

the sparsity constrained program can be formulated as

$$\min \left\{f(\mathbf{x}) : \mathbf{x} \in X_0^k\right\}. \tag{2}$$

CONTACT G. Giallombardo. Email: giovanni.giallombardo@unical.it

Sparsity constrained optimization, together with the companion sparse optimization problem

$$\min \left\{ f(\mathbf{x}) + \|\mathbf{x}\|_0 : \mathbf{x} \in \mathbb{R}^n \right\} \tag{3}$$

has received increasing attention in last years, mainly for its potential in effectively dealing with applications in the areas of Machine Learning [13], Compressed Sensing [12], Portfolio Selection [6] and Statistics [4]. For the theoretical study of the optimality conditions of problems involving the ℓ_0-pseudo-norm we refer to [3, 9, 15]. Computational complexity issues have been assessed in [1]. From the computational standpoint, sparsity has been approached in different ways, mainly by adopting models embedding appropriate sets of binary variables [16, 23, 25] or by approximating the ℓ_0-pseudo-norm by means of continuous concave functions [8, 26, 28, 31].

A more recent research stream has focused on the use of polyhedral k-norms to force the solution of an optimization problem to be sparse. The k-norm of any vector $\mathbf{x} \in \mathbb{R}^n$ is defined as the sum of its k largest absolute-value components, and it is indicated as $\|\mathbf{x}\|_{[k]}$. It is related to the ℓ_1- and ℓ_∞-norm by the relations

$$\|\mathbf{x}\|_\infty = \|\mathbf{x}\|_{[1]} \leq \|\mathbf{x}\|_{[2]} \leq \ldots \leq \|\mathbf{x}\|_{[n]} = \|\mathbf{x}\|_1, \tag{4}$$

and, as later explained at the beginning of §2, see formula (9), it can be interpreted as

$$\|\mathbf{x}\|_{[k]} = \max \left\{ \mathbf{x}^\top \mathbf{y} : \|\mathbf{y}\|_1 \leq k,\, \|\mathbf{y}\|_\infty \leq 1 \right\}. \tag{5}$$

In the pioneering work [30] it has been used in tackling overdetermined systems of linear equations. A thorough study of the properties of the k-norms is in [18, 32].

Thanks to the following property, linking the ℓ_0-pseudo-norm to the ℓ_1- and the k-norms,

$$\mathbf{x} \in X_0^k \quad \Longleftrightarrow \quad \|\mathbf{x}\|_1 - \|\mathbf{x}\|_{[k]} = 0, \tag{6}$$

the k-norm has been successfully employed in [20] for dealing with the sparsity constrained optimization. In [17, 19] sparse optimization has been approached along the same guidelines, focusing, in particular, on the application to SVM classification [11] in Machine Learning.

The concept of k-norm is also evoked in some approaches to sparsity available in the Statistics literature [5, 7], mainly in comparison with the Lasso method that is based on adopting the ℓ_1-norm instead of the ℓ_0-pseudo-norm in (3), and it ensures, in general, a reasonable sparsity of the solution.

In this paper we tackle the sparsity constrained optimization problem by elaborating on the approach introduced in [20] and applied in [17], where the use of the polyhedral k-norm is explored as a tool to deal with sparsity. The novelty of our approach consists in the introduction of a model based on the *dual* of the k-norm [18, 21, 30] which, unlike the k-norm-based methods [17, 20], allows us to avoid the need of solving a DC (Difference of convex) optimization problem, while being suitable of treatment via an effective heuristic approach. More specifically, we consider a relaxation of the dual k-norm model that can be treated numerically via a block coordinate approach. It requires, at each iteration, the solution of two convex programs, with one of the two solvable in $O(n \log n)$ time. The proposed heuristics is then tested in the Machine

Learning framework, focusing in particular on Feature Selection problems where the design of sparse classifiers is required.

The paper is organized as follows. In §2 we state our model by introducing a dual formulation of the sparsity constrained program (2). In §3 we discuss about possible approaches to numerically solving the dual formulation, particularly focusing on a penalty-based model for which, in §4, we propose a block coordinate heuristics. In §5 we apply our model, and the related algorithm, to a Feature Selection problem in the Support Vector Machines framework, and we report on computational results obtained on a set of benchmark datasets.

2. Dual formulation of the sparsity-constrained optimization problem

The sparsity constrained optimization problem (2), on the basis of the property (6), has been restated in [20] as

$$\min \left\{ f(\mathbf{x}) : \|\mathbf{x}\|_1 - \|\mathbf{x}\|_{[k]} = 0,\, \mathbf{x} \in \mathbb{R}^n \right\}. \tag{7}$$

We may obtain yet another reformulation of (2) by considering the dual of norm $\|\cdot\|_{[k]}$. Recalling that, given any norm $\|\cdot\|$, the corresponding dual norm $\|\cdot\|^*$ is defined as

$$\|\mathbf{x}\|^* = \max \left\{ \mathbf{x}^\top \mathbf{y} : \|\mathbf{y}\| \leq 1,\, \mathbf{y} \in \mathbb{R}^n \right\}, \tag{8}$$

it has been proved (see, e.g., [18, 30]) that

$$\|\cdot\|_{[k]}^* = \max \left\{ \frac{1}{k} \|\cdot\|_1 , \|\cdot\|_\infty \right\}.$$

By observing that $\|\cdot\| = \left(\|\cdot\|^*\right)^*$, from (8) we write the k-norm as:

$$\|\mathbf{x}\|_{[k]} = \max \left\{ \mathbf{x}^\top \mathbf{y} : \max \left\{ \frac{1}{k} \|\mathbf{y}\|_1, \|\mathbf{y}\|_\infty \right\} \leq 1,\, \mathbf{y} \in \mathbb{R}^n \right\}. \tag{9}$$

We introduce the auxiliary variables $\mathbf{y} \in \mathbb{R}^n$ and define the set

$$\Omega_0^k \triangleq \left\{ (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n : \mathbf{x}^\top \mathbf{y} \geq \|\mathbf{x}\|_1,\, \frac{1}{k} \|\mathbf{y}\|_1 \leq 1,\, \|\mathbf{y}\|_\infty \leq 1 \right\}. \tag{10}$$

Taking into account (9), we state the following problem (in the variables $\mathbf{x}$ and $\mathbf{y}$)

$$f^* = \min \left\{ f(\mathbf{x}) : (\mathbf{x}, \mathbf{y}) \in \Omega_0^k \right\} \tag{11}$$

whose constraint set Ω_0^k is nonconvex due to the presence of the inequality $\mathbf{x}^\top \mathbf{y} \geq \|\mathbf{x}\|_1$. The equivalence of (11) to (2), and thus to (7), is proved in the next proposition.

Proposition 2.1. *Let $\overline{\mathbf{x}} \in \mathbb{R}^n$, then $\overline{\mathbf{x}} \in X_0^k$ if and only if there exists $\overline{\mathbf{y}} \in \mathbb{R}^n$ such that $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_0^k$.*

Proof. We first consider a pair $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_0^k$ and observe that

$$|\overline{y}_i| \leq 1 \quad \forall i \in \{1, \ldots, n\} \qquad \text{and} \qquad \left| \left\{ i \in \{1, \ldots, n\} : |\overline{y}_i| = 1 \right\} \right| \leq k,$$

since $\|\overline{\mathbf{y}}\|_1 \leq k$ and $\|\overline{\mathbf{y}}\|_\infty \leq 1$. Furthermore, accounting also for $\overline{\mathbf{x}}^\top \overline{\mathbf{y}} \geq \|\overline{\mathbf{x}}\|_1$, and recalling (4) and (5), we have that

$$\|\overline{\mathbf{x}}\|_1 \leq \overline{\mathbf{x}}^\top \overline{\mathbf{y}} \leq \|\overline{\mathbf{x}}\|_{[k]} \leq \|\overline{\mathbf{x}}\|_1,$$

from which it follows that $\|\overline{\mathbf{x}}\|_1 = \|\overline{\mathbf{x}}\|_{[k]}$ and in turn, from (6), that $\overline{\mathbf{x}} \in X_0^k$. Next, considering any $\overline{\mathbf{x}} \in X_0^k$ and defining a vector $\overline{\mathbf{y}} \in \mathbb{R}^n$ such that

$$\overline{y}_i = \begin{cases} 1 & \text{if} \quad \overline{x}_i > 0 \\ -1 & \text{if} \quad \overline{x}_i < 0 \\ 0 & \text{otherwise,} \end{cases}$$

for every $i \in \{1, \ldots, n\}$, it is easy to verify that $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_0^k$. □

In the following we refer to problem (11) as the *dual formulation of the sparsity constrained optimization problem.*

3. Numerical treatment of the dual formulation

We discuss our approach to numerically tackle problem (11). A discussion about similarities and differences between the k-norm and the dual k-norm approach to sparsity constrained optimization is reported in the Appendix. In the following we assume convexity of the objective function $f : \mathbb{R}^n \mapsto \mathbb{R}$.

We relax in (11) the (convex) constraint $\|\mathbf{y}\|_\infty \leq 1$. Letting

$$\Omega_1^k \triangleq \left\{ (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^n \times \mathbb{R}^n : \mathbf{x}^\top \mathbf{y} \geq \|\mathbf{x}\|_1,\ \frac{1}{k}\|\mathbf{y}\|_1 \leq 1 \right\} \supset \Omega_0^k, \tag{12}$$

we consider the following program

$$\min \left\{ f(\mathbf{x}) + \rho(\|\mathbf{y}\|_\infty - 1) : (\mathbf{x}, \mathbf{y}) \in \Omega_1^k \right\}, \tag{13}$$

where we have introduced the penalty parameter $\rho > 0$. Note that the objective function of the above problem is convex, and the nonconvexity is confined in the constraint set Ω_1^k due to $\mathbf{x}^\top \mathbf{y} \geq \|\mathbf{x}\|_1$. In the following propositions we provide some characterizations of the feasible solutions $(\mathbf{x}, \mathbf{y})$ in (13).

Proposition 3.1. *Let*

$$X_1^k \triangleq \left\{ \mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_\infty \geq \frac{1}{k}\|\mathbf{x}\|_1 \right\} \tag{14}$$

and take any $\overline{\mathbf{x}} \in \mathbb{R}^n$. *Then,* $\overline{\mathbf{x}} \in X_1^k$ *if and only if there exists* $\overline{\mathbf{y}} \in \mathbb{R}^n$ *such that* $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$.

Proof. Consider first a pair $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \mathbb{R}^n \times \mathbb{R}^n$ and observe that from $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$ it follows

$$\|\overline{\mathbf{x}}\|_1 \leq \overline{\mathbf{x}}^\top \overline{\mathbf{y}} \leq \|\overline{\mathbf{x}}\|_\infty \|\overline{\mathbf{y}}\|_1 \leq k\|\overline{\mathbf{x}}\|_\infty,$$

namely, $\overline{\mathbf{x}} \in X_1^k$. On the other hand, take $\overline{\mathbf{x}} \in X_1^k$ and assume, without loss of generality, that $\|\overline{\mathbf{x}}\|_\infty = |\overline{x}_1|$. Now, let $\overline{\mathbf{y}} \in \mathbb{R}^n$ be defined, for every $i \in \{1, \ldots, n\}$, as

$$\bar{y}_i = \begin{cases} k\,\mathrm{sgn}(x_i) & \text{if } i = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

and observe that

$$\|\overline{\mathbf{y}}\|_1 = k \qquad \text{and} \qquad \overline{\mathbf{x}}^\top \overline{\mathbf{y}} = k|\overline{x}_1| = k\|\overline{\mathbf{x}}\|_\infty \geq \|\overline{\mathbf{x}}\|_1,$$

from which it follows that $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$. □

Proposition 3.2. *For any* $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$ *it holds that if* $\overline{\mathbf{x}} \neq \mathbf{0}$ *then* $\|\overline{\mathbf{y}}\|_\infty \geq 1$.

Proof. Consider a pair $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$ such that $\overline{\mathbf{x}} \neq \mathbf{0}$, and assume that $\|\overline{\mathbf{y}}\|_\infty < 1$. Hence, from $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$ it follows that

$$\|\overline{\mathbf{x}}\|_1 \leq \overline{\mathbf{x}}^\top \overline{\mathbf{y}} \leq \|\overline{\mathbf{x}}\|_1 \|\overline{\mathbf{y}}\|_\infty < \|\overline{\mathbf{x}}\|_1,$$

where the last strict inequality, returning a contradiction, holds since $\|\overline{\mathbf{x}}\|_1 > 0$. □

Remark 1. We observe that for any $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$, from Proposition 3.1 it follows that

$$\sum_{i \notin I_{[k]}(\overline{\mathbf{x}})} |\overline{x}_i| \leq \sum_{i \in I_{[k]}(\overline{\mathbf{x}})} \big(\|\overline{\mathbf{x}}\|_\infty - |\overline{x}_i|\big),$$

which links the sum of the non-maximal components (in modulus) of $\overline{\mathbf{x}}$ to the variability of the maximal ones. Indeed, we note that, for any $1 < k < n$, vectors of the type $\overline{\mathbf{x}}^\top = (\pm\delta, \pm\delta, \ldots, \pm\delta)$, for some $\delta \in \mathbb{R}$, do not belong to X_1^k.

Some properties of problem (13) are listed in the following proposition whose proof is straightforward.

Proposition 3.3. *Let* $\rho \geq 0$ *and*

$$h(\rho) \triangleq \min\big\{f(\mathbf{x}) + \rho(\|\mathbf{y}\|_\infty - 1) : (\mathbf{x}, \mathbf{y}) \in \Omega_1^k\big\} = f(\mathbf{x}(\rho)) + \rho(\|\mathbf{y}(\rho)\|_\infty - 1).$$

The following properties hold:

i) $h(\rho) \leq f^*, \forall \rho \geq 0$;
ii) $h(\rho)$ *is concave;*
iii) $g = (\|\mathbf{y}(\rho)\|_\infty - 1)$ *is a supergradient of* h, *that is* $g \in \partial h(\rho)$;
iv) Letting $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \Omega_1^k$ *then* $\bar{g} = (\|\overline{\mathbf{y}}\|_\infty - 1)$ *is an* ε*-supergradient of* h, *that is* $\bar{g} \in \partial_\varepsilon h(\rho)$ *for* $\varepsilon = f(\overline{\mathbf{x}}) + \rho(\|\overline{\mathbf{y}}\|_\infty - 1) - h(\rho)$.

In the next section we introduce our heuristic approach based on an alternate search (or block coordinate) approach.

4. The alternate search approach

Aiming to tackle the penalized problem (13), more precisely the problem

$$\min\big\{f(\mathbf{x})+\rho\|\mathbf{y}\|_\infty : (\mathbf{x},\mathbf{y})\in\Omega_1^k\big\}, \tag{16}$$

where the constant term $-\rho$ in the objective function has been neglected, we introduce a heuristic block-coordinate method of the descent type, see [29]. It consists in alternating the following minimization over $\mathbf{x}\in\mathbb{R}^n$, for a given $\overline{\mathbf{y}}\in\mathbb{R}^n$,

$$\min\big\{f(\mathbf{x}) : \mathbf{x}^\top\overline{\mathbf{y}}\geq\|\mathbf{x}\|_1,\ \mathbf{x}\in\mathbb{R}^n\big\}, \tag{$P(\overline{\mathbf{y}})$}$$

with the following one over $\mathbf{y}\in\mathbb{R}^n$

$$\min\big\{\|\mathbf{y}\|_\infty : \overline{\mathbf{x}}^\top\mathbf{y}\geq\|\overline{\mathbf{x}}\|_1,\ \|\mathbf{y}\|_1\leq k,\ \mathbf{y}\in\mathbb{R}^n\big\}, \tag{$P(\overline{\mathbf{x}})$}$$

for a given $\overline{\mathbf{x}}\in\mathbb{R}^n$.

Algorithm 1 Heuristic Block-Coordinate Minimization Algorithm (`HeurBC`)

Input: an integer $k>1$, a scalar $\epsilon>0$, and a vector $\overline{\mathbf{y}}\in\mathbb{R}^n$ satisfying (17)
Output: a pair $(\mathbf{x}^*,\mathbf{y}^*)\in\Omega_1^k$

1: set $\mathbf{y}^{(0)}=\overline{\mathbf{y}}$ and $s=1$ ▷ Initialization
2: solve $(P(\mathbf{y}^{(s-1)}))$ and obtain its minimizer $\mathbf{x}^{(s)}$ ▷ Minimization over $\mathbf{x}$
3: solve $(P(\mathbf{x}^{(s)}))$ and obtain its minimizer $\mathbf{y}^{(s)}$ ▷ Minimization over $\mathbf{y}$
4: **if** $\|\mathbf{y}^{(s)}\|_\infty>\|\mathbf{y}^{(s-1)}\|_\infty-\epsilon$ **then** ▷ Stopping test
5: set $\mathbf{x}^*=\mathbf{x}^{(s)}$, $\mathbf{y}^*=\mathbf{y}^{(s)}$, and **exit** ▷ Return $(\mathbf{x}^*,\mathbf{y}^*)\in\Omega_1^k$
6: **else** ▷ Sufficient decrease of $\|\mathbf{y}^{(s)}\|_\infty$
7: set $s=s+1$ and **go to** Step 2 ▷ Iterate the procedure
8: **end if**

The block-coordinate heuristics (`HeurBC`) is presented in Algorithm 1, where the input data are the integer sparsity parameter $k>1$, the sufficient decrease parameter $\epsilon>0$, and the starting vector $\overline{\mathbf{y}}\in\mathbb{R}^n$ satisfying

$$\|\overline{\mathbf{y}}\|_1\leq k \qquad\text{and}\qquad \|\overline{\mathbf{y}}\|_\infty>1. \tag{17}$$

In the following propositions we summarize the relevant properties of the sequence $\{(\mathbf{x}^{(s)},\mathbf{y}^{(s)})\}$ generated by the block-coordinate minimization heuristics.

Proposition 4.1. *Let $\{(\mathbf{x}^{(s)},\mathbf{y}^{(s)})\}$ be the sequence generated by Algorithm 1, then*

(i) both $\{(\mathbf{x}^{(s)},\mathbf{y}^{(s-1)})\}$ and $\{(\mathbf{x}^{(s)},\mathbf{y}^{(s)})\}$ are contained in Ω_1^k;
(ii) $\|\mathbf{y}^{(s-1)}\|_\infty<1\Longrightarrow\mathbf{x}^{(s)}=\mathbf{0}$;
(iii) $\mathbf{x}^{(s)}\neq\mathbf{0}\Longrightarrow\|\mathbf{y}^{(s)}\|_\infty\geq1$ and $\mathbf{x}^{(s)}=\mathbf{0}\Longrightarrow\mathbf{y}^{(s)}=\mathbf{0}$;
(iv) $\|\mathbf{y}^{(s-1)}\|_\infty>\|\mathbf{y}^{(s)}\|_\infty$;

Proof. (i) These are straightforward consequences of the formulations $(P(\mathbf{y}^{(s-1)}))$ and $(P(\mathbf{x}^{(s)}))$ combined with the selection of $\mathbf{y}^{(0)}$ such that $\|\mathbf{y}^{(0)}\|_1 \leq k$;
(ii) Recalling the constraint $\|\mathbf{x}\|_1 \leq \mathbf{x}^\top \mathbf{y}^{(s-1)}$ in $P(\mathbf{y}^{(s-1)})$, the results easily follows by observing that $\mathbf{x}^\top \mathbf{y}^{(s-1)} \leq \|\mathbf{x}\|_1 \|\mathbf{y}^{(s-1)}\|_\infty$;
(iii) The former is a consequence of the constraint $\|\mathbf{x}^{(s)}\|_1 \leq \mathbf{x}^{(s)\top}\mathbf{y}$ in $(P(\mathbf{x}^{(s)}))$ combined with $\mathbf{x}^{(s)\top}\mathbf{y} \leq \|\mathbf{x}^{(s)}\|_1 \|\mathbf{y}\|_\infty$, while the latter follows from feasibility of $\mathbf{y}^{(s)} = \mathbf{0}$ in $(P(\mathbf{x}^{(s)}))$ if $\mathbf{x}^{(s)} = \mathbf{0}$;
(iv) This easily follows from feasibility of $\mathbf{y}^{(s-1)}$ in $(P(\mathbf{x}^{(s)})$.

□

Proposition 4.2. *Let $\{(\mathbf{x}^{(s)}, \mathbf{y}^{(s)})\}$ be the sequence generated by Algorithm 1, then the algorithm terminates after finitely many iterations at $(\mathbf{x}^*, \mathbf{y}^*) \in \Omega_1^k$ such that either $(\mathbf{x}^*, \mathbf{y}^*) = (\mathbf{0}, \mathbf{0})$ or $\mathbf{x}^* \neq \mathbf{0}$ and $\|\mathbf{y}^*\|_\infty = \alpha \geq 1$, the equality holding if and only if $\mathbf{x}^* \in X_0^k$.*

Proof. Termination of the algorithm after finitely many iterations is ensured by monotonicity, consequence of Proposition 4.1(iv), and boundedness from below of the sequence $\{\|\mathbf{y}^{(s)}\|_\infty\}$, while the properties of $(\mathbf{x}^*, \mathbf{y}^*)$ are consequences of parts (i) and (iii) of Proposition 4.1. Furthermore, if $\alpha = 1$ then from (10) it follows that $(\mathbf{x}^*, \mathbf{y}^*) \in \Omega_0^k$, hence $\mathbf{x}^* \in X_0^k$ due to Proposition 2.1. □

Some further remarks about the relevant features of Algorithm 1 are in order. First, we note that the initialization of $\mathbf{y}^{(0)}$ such that $\|\mathbf{y}^{(0)}\|_1 \leq k$ and $\|\mathbf{y}^{(0)}\|_\infty \geq 1$ ensure, respectively, that the feasible region of $(P(\mathbf{x}^{(1)}))$ is not empty, due to Proposition 3.1 since $\mathbf{x}^{(1)} \in X_1^k$, and that the trivial termination implied by Proposition 4.1(ii) is prevented. Next, focusing on the nontrivial outcome $(\mathbf{x}^*, \mathbf{y}^*)$ such that $\|\mathbf{y}^*\|_\infty = \alpha \geq 1$, we observe that

$$\|\mathbf{y}^*\|_1 \leq k \qquad \Longrightarrow \qquad \mathbf{x}^{*\top}\mathbf{y}^* \leq \|\mathbf{y}^*\|_\infty \|\mathbf{x}^*\|_{[k]} = \alpha \|\mathbf{x}^*\|_{[k]}$$

from which, recalling the constraints in $(P(\mathbf{x}^*))$, we obtain

$$\|\mathbf{x}^*\|_1 \leq \mathbf{x}^{*\top}\mathbf{y}^* \leq \alpha \|\mathbf{x}^*\|_{[k]}.$$

As a consequence, at $\mathbf{x}^* \neq \mathbf{0}$ it holds that

$$1 \leq \frac{\|\mathbf{x}^*\|_1}{\|\mathbf{x}^*\|_{[k]}} \leq \alpha$$

which provides a bound on the relative weight of the non-maximal components of $\mathbf{x}^*$ and confirms again that $\alpha = 1$ if and only if $\mathbf{x}^* \in X_0^k$ due to (6). Finally, to highlight the heuristic nature of the proposed approach, we observe that, since the penalty parameter ρ does not play any role, the algorithm cannot be cast into some iterative penalty-function approach by increasing ρ to force feasibility of the relaxed constraint.

Note that property *iv)* of Proposition 3.3 provides an interpretation of $(\alpha - 1)$ in terms of ε-superdifferential of function $h(\rho)$.

Problem $(P(\overline{\mathbf{x}}))$ is obviously convex, as is problem $(P(\overline{\mathbf{y}}))$ due to the convexity assumption made on $f(\cdot)$. In the remainder of the section we will focus on problem $(P(\overline{\mathbf{x}}))$ showing that its solution can be obtained in $O(n \log n)$ time via an ad hoc sorting algorithm.

4.1. Solving the subproblem $(\mathbf{P}(\mathbf{x}^{(s)}))$

Problem $(P(\mathbf{x}^{(s)}))$ can be put in the form of a structured linear program by observing that there exists an optimal solution $\overline{\mathbf{y}}$ where

$$\mathbf{x}^{(s)\top}\overline{\mathbf{y}} = \sum_{i=1}^{n} |x_i^{(s)}|\,|\overline{y}_i|.$$

Thus, without loss of generality, in the following we address the properties of the problem:

$$\min\left\{\|\mathbf{y}\|_\infty : \mathbf{w}^\top\mathbf{y} \geq \|\mathbf{w}\|_1,\, \mathbf{e}^\top\mathbf{y} \leq k,\, \mathbf{y} \geq \mathbf{0},\, \mathbf{y} \in \mathbb{R}^n\right\}, \qquad (P(\mathbf{w}))$$

where $\mathbf{w}$ is any nonnegative vector in $\mathbb{R}^n$ whose components are ordered in decreasing order. Problem $(P(\mathbf{w}))$ is in turn equivalent to the following linear program $(P_0(\mathbf{w}))$, where the additional scalar variable y_0 has been introduced to represent $\|\mathbf{y}\|_\infty$:

$$\min\left\{y_0 : \mathbf{w}^\top\mathbf{y} \geq \|\mathbf{w}\|_1,\, \mathbf{e}^\top\mathbf{y} \leq k,\, y_0\mathbf{e} - \mathbf{y} \geq 0,\, \mathbf{y} \geq \mathbf{0},\, \mathbf{y} \in \mathbb{R}^n\right\}, \qquad (P_0(\mathbf{w}))$$

Focusing on the nontrivial case where $\mathbf{w} \neq \mathbf{0}$, from Proposition 4.2 it follows that the minimum $\overline{y}_0$ of $(P_0(\mathbf{w}))$ is such that $\overline{y}_0 \geq 1$, with the equality holding, due to Proposition 2.1, if and only if $\|\mathbf{w}\|_0 \leq k$. Hence, in the remainder of the section we will work under the assumptions that

$$\|\mathbf{w}\|_0 > k \quad (\Longleftrightarrow y_0^* > 1) \qquad (18)$$

and

$$\|\mathbf{w}\|_\infty \geq \frac{1}{k}\|\mathbf{w}\|_1 \qquad (19)$$

the latter being consequence of Proposition 4.1(i), taking into account the characterization of $\mathbf{x}^{(s)}$ given by Proposition 3.1.

In particular, we first consider the following dual problem of $(P_0(\mathbf{w}))$

$$\max\left\{\|\mathbf{w}\|_1\mu - k\sigma : \mathbf{w}\mu - \mathbf{e}\sigma - \boldsymbol{\lambda} \leq \mathbf{0},\, \mathbf{e}^\top\boldsymbol{\lambda} = 1,\, \boldsymbol{\lambda} \geq \mathbf{0},\, \mu \geq 0,\, \sigma \geq 0,\, \boldsymbol{\lambda} \in \mathbb{R}^n\right\}, \qquad (D_0(\mathbf{w}))$$

then we state the complementary slackness conditions for the pair $(P_0(\mathbf{w}))$-$(D_0(\mathbf{w}))$:

$$\begin{aligned}
\mu(\mathbf{w}^\top\mathbf{y} - \|\mathbf{w}\|_1) &= 0 && (20)\\
\sigma(k - \mathbf{e}^\top\mathbf{y}) &= 0 && (21)\\
\lambda_i(y_0 - y_i) &= 0 \quad \forall i \in \{1,\ldots,n\} && (22)\\
y_i(w_i\mu - \sigma - \lambda_i) &= 0 \quad \forall i \in \{1,\ldots,n\}. && (23)
\end{aligned}$$

Denoting the optimal solutions of $(P_0(\mathbf{w}))$ and $(D_0(\mathbf{w}))$ by $(\overline{y}_0, \overline{\mathbf{y}})$ and $(\overline{\mu}, \overline{\sigma}, \overline{\boldsymbol{\lambda}})$, respectively, we note that assumption (18) implies $\overline{\mu} > 0$, since otherwise the dual objective function would be prevented to take strictly positive values. Furthermore,

we can derive from (20)-(23) the following relevant properties:

$$\overline{\lambda}_i = 0 \qquad \forall i \in \{1,\dots,n\} \quad \text{such that} \quad \overline{y}_i < \overline{y}_0 \tag{24}$$

$$\frac{\overline{\sigma}}{\overline{\mu}} = w_i \qquad \forall i \in \{1,\dots,n\} \quad \text{such that} \quad 0 < \overline{y}_i < \overline{y}_0 \tag{25}$$

$$\frac{\overline{\sigma}}{\overline{\mu}} \leq w_i \qquad \forall i \in \{1,\dots,n\} \quad \text{such that} \quad \overline{y}_i = \overline{y}_0, \tag{26}$$

where (24) follows from (22), (25) follows from (23) combined with $\overline{\mu} > 0$, and (26) follows from (23) combined with nonnegativity of $\overline{\boldsymbol{\lambda}}$.

As a consequence of the above observations, we can get an insight into the structure of $(\overline{y}_0, \overline{\mathbf{y}})$ and $(\overline{\mu}, \overline{\sigma}, \overline{\boldsymbol{\lambda}})$. In fact, we note that assumption (18), combined with $\mathbf{e}^\top \mathbf{y} \leq k$, implies that no more than $k-1$ components of $\overline{\mathbf{y}}$ can take value $\overline{\mathbf{y}}_0$. Thus, there exists some index $\ell \in \{1,\dots,k-1\}$ such that $\overline{\mathbf{y}}_i = \overline{\mathbf{y}}_0$ for every $i \in \{1,\dots,\ell\}$. Moreover, by defining

$$I(\overline{\mathbf{y}}) \triangleq \left\{ i \in \{1,\dots,n\} : 0 < \overline{y}_i \right\}$$

and

$$I_0(\overline{\mathbf{y}}) \triangleq \left\{ i \in I(\overline{\mathbf{y}}) : \overline{y}_i < \overline{y}_0 \right\}$$

then (25) and (26) imply that

$$j \in I_0(\overline{\mathbf{y}}) \qquad \Longleftrightarrow \qquad j = \arg\min\{w_i : i \in I(\overline{\mathbf{y}})\}.$$

Now we present an approach to find a primal-dual couple of feasible solutions satisfying the complementary slackness conditions (24)-(26). Given an index set $I = \{1, 2, \dots, h\}$ for some $h \in \{2,\dots,k\}$, we define the vector $(y_0, \mathbf{y})$ such that $y_{h+1} = \cdots = y_n = 0$ and $y_0 = y_1 = \cdots = y_{h-1}$. Then, in order to satisfy constraints $\mathbf{w}^\top \mathbf{y} \geq \|\mathbf{w}\|_1$ and $\mathbf{e}^\top \mathbf{y} \leq k$ it suffices to find y_0 and y_h such that

$$\|\mathbf{w}\|_{[h-1]} y_0 + w_h y_h = \|\mathbf{w}\|_1 \tag{27}$$

and

$$(h-1) y_0 + y_h = k. \tag{28}$$

Thus, we obtain

$$y_i = \begin{cases} \dfrac{\|\mathbf{w}\|_1 - k w_h}{\|\mathbf{w}\|_{[h]} - h w_h} & i = 0,\dots,h-1 \\ k - (h-1)\dfrac{\|\mathbf{w}\|_1 - k w_h}{\|\mathbf{w}\|_{[h]} - h w_h} & i = h \\ 0 & i = h+1,\dots,n. \end{cases} \tag{29}$$

We observe that there exists at least one index $h \in \{2, \ldots, k\}$ such that $\|\mathbf{w}\|_{[h]} - hw_h > 0$, since otherwise one would have $w_1 = w_2 = \cdots = w_k$ and the assumptions (18) and (19) would be violated. Moreover, feasibility of a solution $(y_0, \mathbf{y})$ obtained according to (29) is ensured if $y_h \geq 0$ and $y_i \geq y_h$, for every $i \in \{0, \ldots, h-1\}$. Such a solution, whose objective function value is

$$y_0 = \frac{\|\mathbf{w}\|_1 - kw_h}{\|\mathbf{w}\|_{[h]} - hw_h}$$

is just the optimal solution $(\overline{y}_0, \overline{\mathbf{y}})$, as we show next. In fact, setting the dual variables as

$$\overline{\mu} = \frac{1}{\|\mathbf{w}\|_{[h]} - hw_h} > 0, \quad \overline{\sigma} = \frac{w_h}{\|\mathbf{w}\|_{[h]} - hw_h} > 0$$

and

$$\overline{\lambda}_i = \begin{cases} w_i\overline{\mu} - \overline{\sigma} = \dfrac{w_i - w_h}{\|\mathbf{w}\|_{[h]} - hw_h} \geq 0, & \forall i \in \{1, \ldots, h\} \\ 0 & \forall i \in \{h+1, \ldots, n\} \end{cases}$$

we obtain a dual feasible solution, since $\mathbf{e}^\top \overline{\boldsymbol{\lambda}} = 1$, with the corresponding objective function value

$$\|\mathbf{w}\|_1 \overline{\mu} - k\overline{\sigma} = \frac{\|\mathbf{w}\|_1 - kw_h}{\|\mathbf{w}\|_{[h]} - hw_h},$$

which coincides with the primal objective function value. Hence, primal-dual optimality follows from satisfaction of the complementary slackness conditions (20)-(23).

Summing up, since from duality it follows that the only primal feasible solutions candidate to be optimal are those where y_0 coincides with at most $(k-1)$ components of $\mathbf{y}$, ordered according to nonincreasing values of w_i, the optimal solution can be found by a simple sweeping algorithm that checks for feasibility such type of primal solutions, as we show in Algorithm 2. It is easy to verify that, due to the the initial sorting of the components of $\mathbf{w}$, and taking into account that the sweeping cost is linear in k, the total computational cost of the algorithm is $O(n \log n)$.

5. Computational experience in the Support Vector Machine framework

In the Support Vector Machine (SVM) framework for binary classification, two (labeled) point-sets $\mathcal{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_\ell\}$ and $\mathcal{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_m\}$ in $\mathbb{R}^n$ are given, the objective being to find a hyperplane, associated with a couple $(\mathbf{x}, \gamma) \in \mathbb{R}^n \times \mathbb{R}$, strictly separating them. Thus, it is required that the following inequalities hold true:

$$\mathbf{a}_i^\top \mathbf{x} \leq \gamma - 1, \qquad \forall i \in \{1, \ldots, \ell\}, \tag{30}$$

$$\mathbf{b}_j^\top \mathbf{x} \geq \gamma + 1, \qquad \forall j \in \{1, \ldots, m\}. \tag{31}$$

Algorithm 2 Sweeping algorithm

Input: a vector $\mathbf{w} \in \mathbb{R}^n$ ordered according to decreasing values of w_i, an integer $k \geq 2$
Output: a solution $(\overline{y}_0, \overline{\mathbf{y}}) \in \Omega_1^k$

set $h = k$ ▷ Initialization
for $i = 0 \dots (h-1)$ **do** ▷ Calculate a tentative solution
    set $\overline{y}_i = \frac{\|\mathbf{w}\|_1 - k w_h}{\|\mathbf{w}\|_{[h]} - h w_h}$
end for
set $\overline{y}_h = k - (h-1)\overline{y}_0$
for $i = (h+1) \dots n$ **do**
    set $\overline{y}_i = 0$
end for
if $(\|\mathbf{w}\|_{[h]} - h w_h > 0)$ **and** $(\overline{y}_0 \geq \overline{y}_h)$ **and** $(\overline{y}_h \geq 0)$ **then** ▷ Feasibility test
    exit ▷ Return $(\overline{y}_0, \overline{\mathbf{y}}) \in \Omega_1^k$
else
    set $h = h - 1$ and **go to** Step 2 ▷ Iterate the procedure
end if

Since such a hyperplane may not exist, whenever $\mathrm{conv}\mathcal{A} \cap \mathrm{conv}\mathcal{B} \neq \emptyset$, the following convex piecewise linear and nonnegative error function of $(\mathbf{x}, \gamma)$ is defined

$$\mathrm{err}(\mathbf{x}, \gamma) = \sum_{i=1}^{\ell} \max\left\{0, \mathbf{a}_i^\top \mathbf{x} - \gamma + 1\right\} + \sum_{j=1}^{m} \max\left\{0, -\mathbf{b}_j^\top \mathbf{x} + \gamma + 1\right\}. \tag{32}$$

We note that $\mathrm{err}(\mathbf{x}, \gamma)$ is equal to zero if and only if $(\mathbf{x}, \gamma)$ defines a strictly separating hyperplane satisfying (30)-(31). In the SVM approach the following convex problem

$$\min\left\{C \cdot \mathrm{err}(\mathbf{x}, \gamma) + \|\mathbf{x}\| : \mathbf{x} \in \mathbb{R}^n,\, \gamma \in \mathbb{R}\right\}, \tag{33}$$

is solved, where the norm of $\mathbf{x}$ is added to the error function aiming to obtain a maximum-margin separation, and C is a positive trade-off parameter.

In this classification framework, sparse optimization comes into play in case feature selection is pursued, with the ℓ_0-pseudo-norm looking as the most suitable tool, although the ℓ_1-norm has been often considered as a good approximation. In the following, we will focus on the sparsity constrained counterpart of (33)

$$\min\left\{\mathrm{err}(\mathbf{x}, \gamma) : \|\mathbf{x}\|_0 \leq k,\, \mathbf{x} \in \mathbb{R}^n,\, \gamma \in \mathbb{R}\right\}, \tag{34}$$

where the ℓ_0-pseudo-norm is adopted in order to keep the number of relevant features of the SVM not larger than a positive integer k. We will tackle problem (34) by means of the approach described in §4, assuming that problem $(P(\overline{\mathbf{y}}))$ to be solved at Step 2 of Algorithm 1 has the following structure

$$\min\left\{\mathrm{err}(\mathbf{x}, \gamma) : \mathbf{x}^\top \overline{\mathbf{y}} \geq \|\mathbf{x}\|_1,\, \mathbf{x} \in \mathbb{R}^n,\, \gamma \in \mathbb{R}\right\}. \tag{35}$$

We have evaluated the computational behavior of our SVM-based feature selection

Table 1. Details of datasets

#	Name	Reference	ℓ	m	n
1	Breast-Cancer	[10]	444	239	10
2	Diabetes	[10]	268	500	8
3	Heart	[10]	150	120	13
4	Ionosphere	[10]	126	225	34

model by testing it on 4 well known datasets whose relevant details are listed in Table 1. The experimental plan is based on the tenfold cross-validation protocol adopted to train the classifier, by randomly partitioning every dataset into 10 groups of equal size. Then, 10 different blocks (the training sets) are built, each containing 9 out of 10 groups. Every block is used to train the classifier, using the left out group as the testing-set that returns the percentage of points that are correctly classified (test correctness).

We have implemented the block-coordinate algorithm in Python 3.6 and run the computational experiments on a 2.80 GHz Intel(R) Core(TM) i7 computer. The LP solver of IBM ILOG CPLEX 20.1 has been used to solve linear programs at Step 2 of Algorithm 1. We randomly sample a starting point satisfying (17), and we embed the whole algorithm into multi-start procedure repeated 30 times per fold, adopting as the output $(\mathbf{x}^*, \gamma^*)$ the one returned by the run that has the lowest training error. The sufficient decrease parameter ϵ has been tuned to 0.001.

Numerical results are reported in Tables 2–5, where we list the percentage correctness averaged over the 10 folds of both the testing (`AvgTest`) and the training (`AvgTrain`) phases, for several values of the parameter k. Moreover, we report the values `ft0`, `ft-2`, `ft-4`, and `ft-9`, representing the percentage average of features for which the corresponding component of the minimizer $\mathbf{x}^*$ is larger than 1, 10^{-2} , 10^{-4}, 10^{-9}, respectively. Hence, small values of `ft-9` denote high sparsity of $\mathbf{x}^*$. In fact $(1 - \texttt{ft-9})$ can be interpreted as the percentage of the zero-valued components. We also report the `cpu` time (measured in seconds) regarding the execution time of the block-coordinate minimization algorithm in the training phase, averaged over the 10 training folds.

The numerical results demonstrate the ability of the proposed method, even adopting relevant restriction of the cardinality parameter k, of providing good solutions, in terms of the number of active features, without severe reduction of the classification correctness.

For comparison purpose we report in Table 6 our results and those presented in [16] and in [17], where a mixed integer programming formulation treated via Lagrangian relaxation and a k-norm based approach to Feature Selection are adopted, respectively. We indicate our algorithm as `HeurBC` and we report the results in terms of average testing correctness (`AvgTest`) and average percentage of non-zero components (`ft-9`). For the algorithm `HeurBC` we also report the particular value of k adopted.

The comparisons show that `HeurBC`, as well as the other tested algorithms, provide a satisfactory trade-off between correctness and sparsity, without clear dominance of any of the three.

A better insight on the behavior of `HeurBC` can be gained by considering the results provided by the benchmark package `LibLinear`[1] [14] adopted to solve model (33) equipped with an ℓ_1 norm, i.e., an SVM with LASSO regularization. Such results, reported in Tables 7–10 for different values of the regularization parameter C, show, as expected, an acceptable performance in terms of sparsity enforcement when C

[1]We have adopted the method available in the Python `scikit-learn` library.

decreases, although this effect appears weaker than the one provided by `HeurBC`.

In the remaining experiments, we have adopted an ANOVA F-value feature selection procedure, available in the Python scikit-learn library combining the `SelectKBest` method with the `f_classif` function, as a pre-processing phase for both our `HeurBC` and the ℓ_1-`LibLinear` algorithms, for values of k strictly lower than n. Such an approach allows to reduce the size of the input space by getting rid of the $n-k$ lowest scored features. The results are reported in Tables 11-14 for `HeurBC` and in Tables 15-18 for ℓ_1-`LibLinear`. As for the latter, we only report results obtained with $C = 1$, as for smaller values we observed no improvement on the solution sparsity. In both cases the preliminary feature selection provides some improvement, more sensible when ℓ_1-`LibLinear` is adopted. Nonetheless, we observe that unlike ℓ_1-`LibLinear`, the use of `HeurBC` always allows to further improve sparsity after the preliminary feature-selection phase. Finally, it is worth noting that for the Ionosphere dataset `HeurBC` produces a result not too worse than the strong one reported in [24].

References

[1] E. Amaldi, V. Kann (1998) On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems, *Theoretical Computer Science* 209(1-2):237–260.

[2] L.T.H. An, P.D. Tao (2005) The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems, *Annals of Operational Research* 133:23–46.

[3] A. Beck, Y.C. Eldar (2013) Sparsity constrained nonlinear optimization: Optimality conditions and algorithms, *SIAM Journal on Optimization* 23(3):1480–1509.

[4] D. Bertsimas, A. King (2017) Logistic regression. From Art to science, *Statistical Science* 32(3):367–384.

[5] D. Bertsimas, M.S. Copenhaver, R. Mazumder (2017) The trimmed Lasso: sparsity and robustness, *arXiv:1708.04527.*

[6] D. Bienstock (1996) Computational study of a family of mixed-integer quadratic programming problems, *Mathematical Programming* 74:121–140.

[7] M. Bogdan, E. van den Berg, C. Sabatti, W. Su, E.J. Candès (2015) Slope–adaptive variable selection via convex optimization, *Annals of Applied Statistics* 9(3):1103-1140.

[8] P.S. Bradley, O.L. Mangasarian, W.N. Street (1998) Feature selection via mathematical programming, *INFORMS Journal on Computing*, 10(2):209–217.

[9] O.P. Burdakov, C. Kanzow, A. Schwartz (2016) Mathematical programs with cardinality constraints: Reformulation by complementarity-type conditions and a regularization method, *SIAM Journal on Optimization* 26(1):397–425.

[10] C-C. Chang, C-J. Lin (2011) LIBSVM : a library for support vector machines, *ACM Transactions on Intelligent Systems and Technology* 2(27):1–27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

[11] N. Cristianini, J. Shawe–Taylor (2000) *An introduction to support vector machines and other kernel-based learning methods*, Cambridge University Press.

[12] D.L. Donoho (2006) Compressed sensing, *IEEE Transactions on Information Theory* 52:1289–1306.

[13] J.G. Dy, C.E. Brodley, S. Wrobel (2004) Feature selection for unsupervised learning, *Journal of Machine Learning Research* 5:845–889.

[14] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin (2008) LIBLINEAR: A library for large linear classification, *Journal of Machine Learning Research* 9, 1871–1874.

[15] M. Feng, J.E. Mitchell, J-S. Pang, X. Shen, A. Wächter (2018) Complementarity formulations of ℓ_0-norm optimization problems, *Pacific Journal of Optimization* 14(2):273–305.

[16] M. Gaudioso, E. Gorgone, M. Labbé, A. M. Rodríguez-Chía (2017) Lagrangian relaxation for SVM feature selection, *Computers and Operations Research*, 87:137–145.

[17] M. Gaudioso, E. Gorgone, J.-B. Hiriart-Urruty (2019) Feature selection in SVM via polyhedral k-norm *Optimization Letters*, 14:19–36.
[18] M. Gaudioso, J.-B. Hiriart-Urruty (2022) Deforming $\|\cdot\|_1$ into $\|\cdot\|_\infty$ via polyhedral norms: a pedestrian approach, *SIAM Review*, 64(3):713–727.
[19] M. Gaudioso, G. Giallombardo, G. Miglionico (2023) Sparse optimization via vector k-norm and DC programming with an application to feature selection for Support Vector Machines, *Computational Optimization and Applications*, 86(2):745–766.
[20] J. Gotoh, A. Takeda, K. Tono (2018) DC formulations and algorithms for sparse optimization problems, *Mathematical Programming*, 169:141–176.
[21] A. B. Hempel, P. J. Goulart (2014) A Novel Method for Modelling Cardinality and Rank Constraints, *53rd IEEE Conference on Decision and Control*, Los Angeles, Cal., USA Dec. 15–17, pp. 4322–4327.
[22] J.-B. Hiriart-Urruty (1986) Generalized differentiability/duality and optimization for problems dealing with differences of convex functions, *Lecture Notes in Economic and Mathematical Systems*, 256:37–70, Springer Verlag.
[23] S. Maldonado, J. Pérez, R. Weber, M. Labbé (2014) Feature selection for Support Vector Machines via Mixed Integer Linear Programming, *Information Sciences*, 279:163–175.
[24] O. L. Mangasarian (2006) Exact 1-Norm Support Vector Machines via Unconstrained Convex Differentiable Minimization, *Journal of Machine Learning Research*, 7:1517–1530.
[25] M. Pilanci, M. J. Wainwright, L. El Ghaoui (2015) Sparse learning via Boolean relaxations, *Mathematical Programming*, 151:63–87.
[26] F. Rinaldi, F. Schoen, M. Sciandrone (2010) Concave programming for minimizing the zero-norm over polyhedral sets, *Computational Optimization and Applications*, 46:467–486.
[27] A. Strekalovsky (1998) Global optimality conditions for nonconvex optimization, *Journal of Global Optimization* 12:415–434.
[28] E. Soubies, L. Blanc–Féraud, G. Aubert (2017) A unified view of exact continuous penalties for ℓ_2-ℓ_0 minimization, *SIAM Journal on Optimization*, 27(3):2034–2060.
[29] P. Tseng (2001) Convergence of a block coordinate descent method for nondifferentiable minimization, *Journal of Optimization Theory and Applications*, 109(3):475-–494.
[30] G.A. Watson (1992) Linear best approximation using a class of polyhedral norms, *Numerical Algorithms*, 2:321–336.
[31] J. Weston, A. Elisseeff, B. Schölkopf, M. Tipping (2003) Use of the zero-norm with linear models and kernel methods, *Journal of Machine Learning Research*, 3:1439–1461.
[32] B. Wu, C. Ding, D. Sun, K-C. Toh (2014) On the Moreau-Yosida regularization of the vector $k-$norm related functions, *SIAM Journal on Optimization* 24(2):766-794.

Appendix

We recall first how problem (7) is treated in [20], then we discuss yet another penalization of our dual formulation (11) and we show the equivalence of the two approaches.

In [20] a DC (Difference of Convex) decomposition, upon appropriate penalization, is adopted. In particular, by introducing the penalty parameter $\rho > 0$, problem (7) is replaced by the unconstrained optimization model

$$\min \left\{ f(\mathbf{x}) + \rho\big(\|\mathbf{x}\|_1 - \|\mathbf{x}\|_{[k]}\big) : \mathbf{x} \in \mathbb{R}^n \right\} \tag{36}$$

whose objective function is in the form $f_1(\cdot) - f_2(\cdot)$ with

$$f_1(\mathbf{x}) = f(\mathbf{x}) + \rho\|\mathbf{x}\|_1 \qquad \text{and} \qquad f_2(\mathbf{x}) = \rho\|\mathbf{x}\|_{[k]}$$

which are both convex. By applying any method based on successive linearizations of function $f_2(\cdot)$, see [2, 27], one comes out with a sequence of convex programs, each one obtained by taking an affine approximation of function f_2 rooted at the current estimate $\mathbf{x}^{(j)}$ of a local minimizer. In DCA [2], for example, the iterate point $\mathbf{x}^{(j+1)}$ would be calculated as

$$\mathbf{x}^{(j+1)} = \arg\min \left\{ f(\mathbf{x}) + \rho\|\mathbf{x}\|_1 - \rho\Big(\|\mathbf{x}^{(j)}\|_{[k]} + \mathbf{g}^{(j)\top}(\mathbf{x} - \mathbf{x}^{(j)})\Big) : \mathbf{x} \in \mathbb{R}^n \right\},$$

where $\mathbf{g}^{(j)} \in \partial\|\mathbf{x}^{(j)}\|_{[k]}$. We recall that denoting by

$$I_{[k]}(\mathbf{x}^{(j)}) \triangleq \{i_1^{(j)}, \ldots, i_k^{(j)}\}$$

the index set of the k largest components (in modulus) of $\mathbf{x}^{(j)}$, a subgradient $\mathbf{g}^{(j)}$ of the vector k-norm at $\mathbf{x}^{(j)}$ can be calculated by setting (see [17]), for every $i \in \{1, \ldots, n\}$, the component $g_i^{(j)}$ according to

$$g_i^{(j)} = \begin{cases} 1 & \text{if} \quad i \in I_{[k]}(\mathbf{x}^{(j)}) \ \text{ and } \ x_i^{(j)} \geq 0 \\ -1 & \text{if} \quad i \in I_{[k]}(\mathbf{x}^{(j)}) \ \text{ and } \ x_i^{(j)} < 0 \\ 0 & \text{otherwise.} \end{cases}$$

Coming to the dual formulation (11), we consider here an alternative penalty function approach w.r.t. to the one adopted in Section 3. Here we penalize the nonsmooth and nonconvex constraint

$$\mathbf{x}^\top \mathbf{y} \geq \|\mathbf{x}\|_1.$$

Denoting again by $\rho > 0$ the penalty parameter, and letting

$$Y_1^k \triangleq \big\{\mathbf{y} \in \mathbb{R}^n : \|\mathbf{y}\|_1 \leq k, \, \|\mathbf{y}\|_\infty \leq 1\big\}, \tag{37}$$

we obtain the problem

$$\min \left\{ f(\mathbf{x}) + \rho\Big(\|\mathbf{x}\|_1 - \mathbf{x}^\top \mathbf{y}\Big) : \mathbf{x} \in \mathbb{R}^n, \, \mathbf{y} \in Y_1^k \right\} \tag{38}$$

equivalent to (36) as we show next. Taking any optimal solution $\mathbf{x}^*$ to (36), we denote by $(\mathbf{x}^*, \mathbf{y}^*)$ a feasible solution to (38) obtained by setting, for every $i \in \{1, \ldots, n\}$,

$$y_i^* = \begin{cases} 1 & \text{if} \quad i \in I_{[k]}(\mathbf{x}^*) \ \text{ and } \ x_i^* \geq 0 \\ -1 & \text{if} \quad i \in I_{[k]}(\mathbf{x}^*) \ \text{ and } \ x_i^* < 0 \\ 0 & \text{otherwise.} \end{cases} \tag{39}$$

Hence, adopting the same penalty parameter ρ, the objective functions of the two problems take the same value at $\mathbf{x}^*$ and $(\mathbf{x}^*, \mathbf{y}^*)$, respectively, since $\mathbf{x}^{*\top}\mathbf{y}^* = \|\mathbf{x}^*\|_{[k]}$,

and that $(\mathbf{x}^*, \mathbf{y}^*)$ is a minimizer for (38) due to (9). On the other hand, let $(\mathbf{x}^*, \mathbf{y}^*)$ be any optimal solution to problem (38) and note that from optimality it follows

$$\mathbf{y}^* = \arg\max \left\{ \mathbf{x}^{*\top} \mathbf{y} : \mathbf{y} \in Y_1^k \right\}, \tag{40}$$

that is $\mathbf{x}^{*\top} \mathbf{y}^* = \|\mathbf{x}^*\|_{[k]}$, see (9). Hence, at $\mathbf{x}^*$ and $(\mathbf{x}^*, \mathbf{y}^*)$, respectively, the objective functions of the two problems take the same value and $\mathbf{x}^*$ must be optimal for (36).

We remark that problem (38) can be tackled by means of an alternate search approach, consisting in alternating minimization over $\mathbf{x}$ keeping $\mathbf{y}$ fixed and vice versa. Observing that the minimization with respect to $\mathbf{y}$ can be solved in closed form, see (39), it is easy to verify that such an approach coincides with DCA for problem (36).

Table 2. Breast Cancer Dataset: `HeurBC` results

k	AvgTest (%)	AvgTrain (%)	ft0 (%)	ft-2 (%)	ft-4 (%)	ft-9 (%)	cpu (s)
10	95.75	97.15	65.00	65.00	65.00	65.00	8.036
8	96.63	96.93	60.00	60.00	60.00	60.00	8.925
6	94.14	94.41	45.00	45.00	45.00	45.00	8.622
4	87.66	86.83	26.00	26.00	26.00	26.00	8.864

Table 3. Diabetes Dataset: `HeurBC` results

k	AvgTest (%)	AvgTrain (%)	ft0 (%)	ft-2 (%)	ft-4 (%)	ft-9 (%)	cpu (s)
8	73.69	74.22	41.25	41.25	41.25	41.25	10.523
6	73.18	73.90	38.75	38.75	38.75	38.75	10.133
5	70.97	72.21	31.25	31.25	31.25	31.25	9.734
4	70.45	71.76	22.50	22.50	22.50	22.50	9.356

Table 4. Heart Dataset: `HeurBC` results

k	AvgTest (%)	AvgTrain (%)	ft0 (%)	ft-2 (%)	ft-4 (%)	ft-9 (%)	cpu (s)
13	84.44	84.53	73.85	73.85	73.85	73.85	4.297
11	81.48	84.03	60.00	60.00	60.00	60.00	4.287
9	81.11	84.07	56.92	56.92	56.92	56.92	4.722
4	80.00	80.21	31.54	31.54	31.54	31.54	6.372

Table 5. Ionosphere Dataset: `HeurBC` results

k	AvgTest (%)	AvgTrain (%)	ft0 (%)	ft-2 (%)	ft-4 (%)	ft-9 (%)	cpu (s)
34	88.87	93.13	84.71	84.71	84.71	84.71	7.767
25	88.04	92.31	71.47	71.47	71.47	71.47	8.686
17	87.45	91.04	61.47	61.47	61.47	61.47	8.480
6	86.06	88.92	47.94	47.94	47.94	47.94	13.892

Table 6. Comparison of `HeurBC` against `Algo[16]` and `Algo[17]`

Dataset	HeurBC			Algo[16]		Algo[17]	
	k	AvgTest	ft-9	AvgTest	ft-9	AvgTest	ft-9
Breast Cancer	6	94.14	45.00	96.41	71.00	93.17	34.00
Diabetes	8	73.69	41.25	76.01	87.50	75.57	43.75
Heart	9	81.11	56.92	83.95	82.31	82.32	50.00
Ionosphere	17	87.45	61.47	87.93	67.65	86.35	13.82

Table 7. Breast Cancer Dataset: ℓ_1-`LibLinear` results

C	AvgTest (%)	AvgTrain (%)	ft0 (%)	ft-2 (%)	ft-4 (%)	ft-9 (%)	cpu (s)
1.000	96.90	97.20	95.00	95.00	95.00	95.00	0.014
0.100	96.50	96.80	79.00	79.00	79.00	79.00	0.002
0.010	94.40	94.80	50.00	50.00	50.00	50.00	0.000
0.005	91.20	91.40	40.00	40.00	40.00	40.00	0.000

Table 8. Diabetes Dataset: ℓ_1-`LibLinear` results

C	AvgTest (%)	AvgTrain (%)	ft0 (%)	ft-2 (%)	ft-4 (%)	ft-9 (%)	cpu (s)
1.000	77.20	78.00	97.50	97.50	97.50	97.50	0.011
0.100	77.10	77.40	75.00	75.00	75.00	75.00	0.003
0.010	71.50	72.40	61.25	61.25	61.25	61.25	0.002
0.005	63.90	64.40	32.50	32.50	32.50	32.50	0.002

Table 9. Heart Dataset: ℓ_1-`LibLinear` results

C	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
1.000	84.10	85.10	98.46	98.46	98.46	98.46	0.003
0.100	83.00	85.00	81.54	81.54	81.54	81.54	0.002
0.010	77.80	77.80	23.85	23.85	23.85	23.85	0.000
0.005	76.30	76.30	8.46	8.46	8.46	8.46	0.002

Table 10. Ionosphere Dataset: ℓ_1-`LibLinear` results

C	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
1.000	90.30	93.70	85.59	85.59	85.59	85.59	0.117
0.100	88.00	90.20	53.53	53.53	53.53	53.53	0.014
0.010	72.10	72.70	8.82	8.82	8.82	8.82	0.006
0.005	74.90	74.90	6.18	6.18	6.18	6.18	0.000

Table 11. Breast Cancer Dataset: ANOVA F-score and `HeurBC` results

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
8	94.43	94.50	46.00	46.00	46.00	46.00	8.945
6	96.19	96.32	40.00	40.00	40.00	40.00	7.578
4	85.99	86.13	17.00	17.00	17.00	17.00	7.722

Table 12. Diabetes Dataset: ANOVA F-score and `HeurBC` results

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
6	74.48	74.20	37.50	37.50	37.50	37.50	10.678
5	72.79	72.79	30.00	30.00	30.00	30.00	9.947
4	73.05	73.13	25.00	25.00	25.00	25.00	14.566

Table 13. Heart Dataset: ANOVA F-score and `HeurBC` results

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
11	83.33	85.23	57.69	57.69	57.69	57.69	4.15
9	80.37	82.55	40.00	40.00	40.00	40.00	3.898
4	71.48	70.70	15.39	15.39	15.39	15.39	2.694

Table 14. Ionosphere Dataset: ANOVA F-score and `HeurBC` results

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
25	88.03	91.90	57.65	57.65	57.65	57.65	8.072
17	85.18	88.89	36.77	36.77	36.77	36.77	7.144
6	85.76	86.96	12.06	12.06	12.06	12.06	5.681

Table 15. Breast Cancer Dataset: ANOVA F-score and ℓ_1-`LibLinear` results ($C = 1.0$)

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
8	97.07	97.27	80.00	80.00	80.00	80.00	0.002
6	97.07	97.20	60.00	60.00	60.00	60.00	0.000
4	95.75	96.13	40.00	40.00	40.00	40.00	0.002

Table 16. Diabetes Dataset: ANOVA F-score and ℓ_1-`LibLinear` results ($C = 1.0$)

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
7	77.20	77.66	87.50	87.50	87.50	87.50	0.000
6	77.20	77.66	75.00	75.00	75.00	75.00	0.002
5	77.20	77.30	62.50	62.50	62.50	62.50	0.000

Table 17. Heart Dataset: ANOVA F-score and ℓ_1-`LibLinear` results ($C = 1.0$)

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
11	84.82	85.84	84.62	84.62	84.62	84.62	0.002
9	84.07	85.89	69.23	69.23	69.23	69.23	0.000
4	83.33	84.07	30.77	30.77	30.77	30.77	0.000

Table 18. Ionosphere Dataset: ANOVA F-score and ℓ_1-`LibLinear` results ($C = 1.0$)

k	`AvgTest` (%)	`AvgTrain` (%)	`ft0` (%)	`ft-2` (%)	`ft-4` (%)	`ft-9` (%)	`cpu` (s)
25	88.30	92.05	73.53	73.53	73.53	73.53	0.002
17	86.61	88.86	50.00	50.00	50.00	50.00	0.002
6	85.77	86.42	17.65	17.65	17.65	17.65	0.002