Appl Intell (2016) 45:549–565
DOI 10.1007/s10489-016-0778-y

# Efficient approaches for $\ell_2$-$\ell_0$ regularization and applications to feature selection in SVM

**Hoai An Le Thi[1,2] · Tao Pham Dinh[3] · Mamadou Thiao[2]**

Published online: 2 April 2016


**Abstract** For solving a class of $\ell_2$-$\ell_0$- regularized problems we convexify the nonconvex $\ell_2$-$\ell_0$ term with the help of its biconjugate function. The resulting convex program is explicitly given which possesses a very simple structure and can be handled by convex optimization tools and standard softwares. Furthermore, to exploit simultaneously the advantage of convex and nonconvex approximation approaches, we propose a two phases algorithm in which the convex relaxation is used for the first phase and in the second phase an efficient DCA (Difference of Convex functions Algorithm) based algorithm is performed from the solution given by Phase 1. Applications in the context of feature selection in support vector machine learning are presented with experiments on several synthetic and real-world datasets. Comparative numerical results with standard algorithms show the efficiency the potential of the proposed approaches.

✉ Hoai An Le Thi
lethihoaian@tdt.edu.vn

Pham Dinh Tao
pham@insa-rouen.fr

Mamadou Thiao
mamadou.thiao@univ-lorraine.fr

[1] Department for Management of Science and Technology Development & Faculty of Mathematics Statistics, Ton Duc Thang University, Ho Chi Minh City, Vietnam

[2] Laboratory of Theoretical and Applied Computer Science, University of Lorraine, Ile du Saulcy, 57045 Metz, France

[3] Laboratory of Mathematics, INSA - Rouen, University of Normandie, Avenue de l'Université 76801 Saint-Etienne-du-Rouvray, Cedex, France

**Keywords** Sparsity · Zero norm · Convex relaxation · Biconjugate function · Nonconvex approximation · DC programming · DCA · Feature selection in SVM

## 1 Introduction

Zero-norm, defined as a total number of non-zero elements in a vector, is an important basic concept for modeling data sparsity. Resulting optimization problems are nonsmooth nonconvex programs with many application domains, which have attracted increasing attention from researchers in recent years.

Given a vector $x \in \mathbb{R}^n$. The support of $x$, denoted $supp(x)$ , is the set of the indices of the non-zero components of $x$, say

$$supp(x) = \{i \in \{1, ..., n\} : \ x_i \neq 0\},$$

and the zero norm of $x$, denoted $\ell_0$-norm, is defined as

$$\|.\|_0 := \text{cardinality of } supp(x).$$

Note that although one uses the term "norm" to design $\|.\|_0$, $\|.\|_0$ is not a norm in the mathematical sense. Indeed, for all $x \in \mathbb{R}^n$ and $\lambda \neq 0$, one has $\|\lambda x\|_0 = \|x\|_0$ , which is not true for a norm.
Formally, a so called $\ell_0$-regularized problem takes the form

$$\min \left\{ \phi(x, y) + \rho \|x\|_0 : (x, y) \in \mathbb{R}^n \times \mathbb{R}^p \right\}, \quad (1)$$

where the function $\phi$ corresponds to a given criterion and $\rho$ is a positive number, called the regularization parameter, that makes the trade-off between the criterion $\phi$ and the sparsity of $x$.

In some applications, one wants to control the sparsity of solutions (for example, in order to limit the number of assets

to be investigated in portfolio management), the $\ell_0$-term is thus put in constraint, and the corresponding optimization problem is

$$\begin{cases} \min_{x,y} \ \phi(x, y) \\ s.t. \quad \|x\|_0 \leq k, \ (x, y) \in \mathbb{R}^n \times \mathbb{R}^p. \end{cases} \tag{2}$$

These are challenging nonconvex programs in machine learning, image analysis and finance.

In this paper we consider a class of $\ell_0$-regularized problems (1) where the function $\phi$ is defined by

$$\phi(x, y) := f(x, y) + \lambda \|x\|_2^2. \tag{3}$$

Here $\lambda > 0$, and $f$ is a loss function which is assumed to be convex. The $\ell_0$-regularized problem becomes the so called $\ell_2$-$\ell_0$-regularized problem

$$\min_{(x,y)\in\mathbb{R}^n\times\mathbb{R}^p} \left\{ F^{\lambda,\rho}(x, y) := f(x, y) + \lambda \|x\|_2^2 + \rho \|x\|_0 \right\}. \tag{4}$$

If the function $\phi$ is strongly convex in the variable $x$, i.e., there is $\lambda > 0$ such that the function $f(x, y) := \phi(x, y) - \lambda \|x\|_2^2$ is convex in the couple of variables $(x, y)$**,** then the $\ell_0$ -regularized problem (1) can be expressed as the $\ell_2$-$\ell_0$-regularized problem (4) and so the techniques developed in this paper can be used for the $\ell_0$-regularized problem in this case.

Let us mention some important applications in machine learning related to the model (4).

**Feature selection in support vector machine (SVM) learning** Feature selection is one of the fundamental problems in machine learning. In many applications such as text classification, web mining, gene expression, micro-array analysis, combinatorial chemistry, image analysis, etc, data sets contain a large number of features, many of which are irrelevant or redundant. Feature selection is often applied to high-dimensional data, prior to classification learning. The main goal is to select a subset of features of a given data set while preserving or improving the discriminative ability of a classifier. Research on feature-selection methods is very active in recent years, and an excellent review can be found in the book by [16]. We will show that the embedded (feature and classifier are simultaneously determined during the training process) feature selection method for linear classification in SVM learning is an instance of the problem (4).

Given a training data $\{a_i, b_i\}_{i=1,\ldots,m}$ where each $x_i \in \mathbb{R}^n$ is labeled by its class $b_i \in \{+1, -1\}$, the goal of SVM learning is to construct a linear classifier function that discriminates the data points $\Lambda := \{a_i\}_{i=1,\ldots,m}$ with respect to their classes $\{b_i\}_{i=1,\ldots,m}$. A classical way to obtain this classifier consists of minimizing the following loss function, [2, 8],

$$f(w, \gamma) := \frac{1}{m} \sum_{i=1}^{m} \max(0, 1 - b_i(\langle a_i, w\rangle + \gamma)), \tag{5}$$

on $w \in \mathbb{R}^n$ and $\gamma \in \mathbb{R}$. If $(w, \gamma)$ is a solution of this problem, then the classifier is given by $F(x) = \text{sign}(\langle a_i, w\rangle + \gamma)$. Since in many practical applications the data set $\Lambda$ is large, the model based directly on solving (5 ) leads to over-fit. [8] proposed to take into account the margin, between the separating hyperplane $x \mapsto w^T x + \gamma$ and the data points $\{a_i\}_{i=1,\ldots,m}$, and to make it maximal as possible. This results in the classical SVM problem, which is the $\ell_2$ -regularized problem

$$(\ell_2 - SVM) \quad \min_{(w,\gamma)\in\mathbb{R}^n\times\mathbb{R}} \left\{ f(w, \gamma) + \lambda \|w\|_2^2 \right\}.$$

The regularization parameter $\lambda > 0$ makes the trade-off between the classifier criterion $f$ and the amplitude of the margin.

The embedded feature selection in SVM involves determining the separating hyperplane $x \mapsto w^T x + \gamma$ *which uses as few features as possible*, which leads to the following optimization problem like (4):

$$(\ell_2 - \ell_0 - SVM) \quad \min_{(w,\gamma)\in\mathbb{R}^n\times\mathbb{R}} \left\{ f(w, \gamma) + \lambda \|w\|_2^2 + \rho \|w\|_0 \right\}. \tag{6}$$

**Sparse linear regression** Consider a training data set $\{b_i, a_i\}_{i=1}^m$ of $m$ independent and identically distributed samples, composed of explanatory variables $a_i \in \mathbb{R}^n$ (inputs) and response variables $b_i \in \mathbb{R}$ (outputs). Let $b := (b_i)_{i=1,\ldots,m}$ and $A := (a_{i,j})_{i=1,\ldots,m}^{j=1,\ldots,n}$ denote the vector of outputs and the matrix of inputs respectively. Linear regression aims to find a relation which can possibly exist between $A$ and $b$, in other words, relating $b$ to a function of $A$ and a model parameter $x$. Such a model parameter $x$ can be obtained by solving the optimization problem

$$\min_x \quad \frac{1}{m} \|Ax - b\|_2^2. \tag{7}$$

In many practical applications simple least squares regression leads to over-fit. This occurs when the fitted model has many feature variables with (relatively) large weights (i.e., $x_i$ is large). A classical way to remedy to these curses is provided by *regularization methods*, among them the $\ell_2$ regularization technique, called *ridge regression* in the statistical literature [17, 18], is very useful. This technique leads to the convex quadratic program

$$\min_{x\in\mathbb{R}^n} \left\{ \frac{1}{m} \|Ax - b\|_2^2 + \lambda \|x\|_2^2 \right\}. \tag{8}$$

The *sparse linear ridge regression* problem aims to find a sparse solution of the above linear ridge regression model, it takes the form of ( 4):

$$\min_{x\in\mathbb{R}^n}\left\{\frac{1}{m}\|Ax-b\|_2^2+\lambda\|x\|_2^2+\rho\|x\|_0\right\}. \tag{9}$$

This problem has many important applications, among them sparse signal/image recovery and feature selection in classification.

**Sparse fisher linear discriminant analysis** Discriminant analysis captures the relationship between multiple independent variables and a categorical dependent variable in the usual multivariate way, by forming a composite of the independent variables. Given a set of $m$ independent and identically distributed samples composed of explanatory variables $a_i \in \mathbb{R}^n$ and binary response variables $b_i \in \{-1, 1\}$. The idea of Fisher linear discriminant analysis is to determine a projection of variables onto a straight line that best separates the two classes. The line is determined so as to maximize the ratio of the variances of between and within classes in this projection, i.e., maximize the function $f(\zeta) = \frac{\langle\zeta, S_B\zeta\rangle}{\langle\zeta, S_W\zeta\rangle}$, where $S_B$ and $S_W$ are, respectively, the between and within classes scatter matrix (which are symmetric positive semidefinite) given by

$$S_B := (s_+ - s_-)(s_+ - s_-)^T \ , S_W = S_+ + S_-,$$

$$S_+ = \sum_{i=1,b_i=+1}^{m}(a_i - s_+)(a_i - s_+)^T,$$
$$S_- = \sum_{i=1,b_i=-1}^{m}(a_i - s_-)(a_i - s_-)^T.$$

Here, for $j \in \{\pm\}$, $s_j$ is the mean vector of class $j$, $l_j$ is the number of labeled samples in class $j$. If $\zeta$ is an optimal solution of the problem, then the classifier is given by $F(a) = \zeta^T a + c, c = -0.5\zeta^T(s_+ + s_-)$.
The sparse Fisher Discriminant model is defined by ($\rho > 0$ )

$$\begin{cases}\min_{(\zeta)} & \zeta^T S_W\zeta + \rho\|\zeta\|_0\\ s.t. & \zeta^T(s_+ - s_-) = b,\end{cases} \tag{10}$$

which takes the form of (4) (with the additional constraint $\zeta^T(s_+ - s_-) = b$ which is imposed to avoid multiplicity of solutions) when $S_W$ is a symmetric positive definite matrix. Indeed, let $\lambda_{min}(S_W) > 0$ be the smallest eigenvalue of $S_W$. For any $0 < \lambda < \lambda_{min}(S_W)$, the matrix $S_W - \lambda I$ is positive definite, and then the function $\zeta^T S_W\zeta - \lambda\|\zeta\|_2^2$ is convex. The problem (10) can be expressed as

$$\begin{cases}\min_{(\zeta)} & \underbrace{\zeta^T S_W\zeta - \lambda\|\zeta\|_2^2}_{\text{convex}} + \lambda\|\zeta\|_2^2 + \rho\|\zeta\|_0\\ s.t. & \zeta^T(s_+ - s_-) = 1.\end{cases} \tag{11}$$

During the last two decades, research is very active in models and methods optimization involving the zero-norm. Works can be divided into three categories depending on how to treat the zero-norm: convex approximation, nonconvex approximation, and exact reformulation via Difference of Convex functions (DC) programming.

The best known convex approach is the $\ell_1$ regularization approach proposed in [41] in the context of linear regression, called LASSO (Least Absolute Shrinkage and Selection Operator), which consists in replacing the $\ell_0$ term $\|w\|_0$ by $\|w\|_1$, the $\ell_1$-norm of the vector $w$. Since its introduction, several works have been developed to study the $\ell_1$ -regularization technique, from the theoretical point of view to efficient computational methods (see [17], Chapter 18). The LASSO penalty has been shown to be, in certain cases, inconsistent for variable selection and biased [46]. Hence, the Adaptive LASSO is introduced in [46] in which adaptive weights are used for penalizing different coefficients in the $\ell_1$ -penalty.

In parallel, nonconvex approximation approaches (the $\ell_0$ term $\|w\|_0$ is approximated by a nonconvex function) were extensively developed

A variety of sparsity-inducing penalty functions have been proposed to approximate the $\ell_0$ term: exponential concave function [3], $\ell_p$-norm with $0 < p < 1$ [11] and $p < 0$ [37], Smoothly Clipped Absolute Deviation (SCAD) [10], Logarithmic function [43], Capped-$\ell_1$ [33]. The shared properties of these approaches are that the nonconvexregularization used for approximating the $\ell_0$ norm are DC functions, and the resulting optimization problems are DC programs.

Using these approximations, several algorithms have been developed for resulting optimization problems, most of them are in the context of feature selection in classification, sparse regressions or more especially for sparse signal recovery: Successive Linear Approximation (SLA) algorithm [3], DCA (Difference of Convex functions Algorithm) based algorithms [6, 7, 12, 15, 19, 21, 22, 26, 30–32], Local Linear Approximation (LLA) [47], Two-stage $\ell_1$ [45], Adaptive Lasso [46], reweighted-$\ell_1$ algorithms [4]), reweighted- $\ell_2$ algorithms such as Focal Underdetermined System Solver (FOCUSS) ([36, 37]), Iteratively reweighted least squares (IRLS) and Local Quadratic Approximation (LQA) algorithm [10, 47].

Very recently, in a more general framework, Le Thi et all [24] offered a unifying nonconvex approximation approach, with solid theoretical tools as well as efficient algorithms based on DC programming and DCA, to tackle the zero-norm and sparse optimization. A common DC approximation of the zero-norm including all standard sparse inducing penalty functions was proposed and four DCA schemes were developed that cover all standard algorithms in nonconvex sparse approximation approaches as special versions.

In the third category, called the exact reformulation nonconvex approach, the $\ell_0$-regularized problem is reformulated as a continuous nonconvex program. The $\ell_0$-regularized problem is first equivalently formulated as a combinatorial optimization problem by using the binary variables $u_i = 0$ if $x_i = 0$ and $u_i = 1$ if $x_i \neq 0$, and then the last problem is reformulated as a DC program via an exact penalty technique. Works in this direction were developed in [19, 21, 40].

Convex regularization approaches involve convex optimization problems for which several standards methods are available. Nonconvex approaches can produce good sparsity, but the resulting optimization problems are still difficult since they are nonconvex and a local minimum may not be a global one. The development of new models and algorithms for minimizing the zero-norm is always a challenge for researchers in optimization and machine learning.

**Our contributions** Our main contributions are threefold. First, we investigate a new convex approach for solving the $\ell_2$-$\ell_0$-regularized problem (4). We propose a tight convex minorant function of $F^{\lambda,\rho}$ by convexifying the nonconvex term $\lambda \|.\|_2^2 + \rho \|.\|_0$ in $F^{\lambda,\rho}$ with the help of biconjugate function technique in nonconvex programming and explicitly computing the greatest convex minorant of this term. We show that the proposed convex relaxation is a special hard-thresholding operation. Secondly, to exploit simultaneously the advantage of convex and nonconvex approximation approaches, we propose a combined convex - nonconvex regularization approach. In the first phase, the convex relaxation is used and in the second phase an efficient DCA based algorithm is applied on DC approximate problems from the solution given by Phase 1. Third, as an application of our method, we implement it in the context of feature selection in Support Vector Machine learning. In the two-phase method, by the convex relaxation, the first phase performs better than $\ell_2 - \ell_1$ regularization on classification while, with a "good" approximation of the $\ell_0$ -norm, the second phase can produce better sparsity. The proposed methods are compared with two standard approaches for ($\ell_2$-$\ell_0$-SVM): the convex regularization ($\ell_2$-$\ell_1$-SVM) and the nonconvex approximation ($\ell_2$-$Exp$-SVM) studied in [31]. We also compare these methods with the classical $\ell_2$-regularized SVM ($\ell_2$-SVM). Numerical results, on tested datasets, show the efficiency of the proposed approaches and their superiority over the competitive methods.

Besides the main contributions concerning solution methods, we also study, in a natural way, the link between optimal solutions of both the resulting convex relaxation problem and the $\ell_2$-$\ell_0$ -regularized problem (4). More precisely we establish a sufficient condition so that an optimal solution of the convex relaxation problem solves the original problem (4). It turns out that this condition is quite strong and it does not hold when $\rho$, the coefficient parameter of $\ell_0$, is quite large (however $\rho$ should not be small when a sparse solution is desired !) This result motivates us to investigate a combined convex relaxation - nonconvex approximation approach. In fact, since the solution obtained from the convex problem is just an approximate solution to $\ell_2$-$\ell_0$ -regularized problem (4), further refinement for the solution via DCA is strongly recommended to produce good sparsity.

The paper is organized as follows. The convex relaxation technique is developed in Section 2. In the first two subsections of this section, we introduce a convex lower bound of the $\ell_2$-$\ell_0$ term and describe the resulting convex relaxation problem of (4). In the next two subsections, we state the link between this technique and hard-thresholding operation and sufficient global optimality conditions for the nonconvex problem (4) while in the last subsection we give a short discussion about numerical methods for the convex relaxation problem. The two phase algorithm is discussed in Section 3 which is started by a short presentation of DC programming and DCA. Section 4 deals with the application of the proposed approaches on feature selection in SVM and, finally, Section 5 concludes the paper.

Before beginning, let us introduce some notations that will be used in the paper.

*Notations:* For a vector $x \in \mathbb{R}^n$, its components are $x_i$, $i = 1, ..., n$. The vector $e$ stands for the vector of ones and $\langle x, y\rangle := x^T y$ is the standard Euclidean inner product with the corresponding norm $\|.\|_2$, while $\|.\|_1$ the $\ell_1$ norm. For a scalar $s \in \mathbb{R}$, $|s|$ denotes the absolute value of $s$, $s^+ := \max(0, s)$, $s^- := \max(0, -s)$. For a vector $x \in \mathbb{R}^n$, $|x|$, $x^+$ and $x^-$ denote the previous operations componentwise. In the sequel $|.|_0$ is $\|.\|_0$ in the one-dimensional case. In convex analysis, let $\Gamma_0(\mathbb{R}^n)$ be the convex cone of all convex functions $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ semicontinuous and proper (i.e. dom $f := \{x \in \mathbb{R}^n : f(x) < +\infty\}$ is nonempty) on $\mathbb{R}^n$. For a convex function $h$, the subdifferential of $h$ at $x_0$, denoted by $\partial h(x_0)$, is defined by

$$\partial h(x_0) := \{y \in \mathbb{R}^n : h(x) \geq h(x_0) + \langle x - x_0, y\rangle, \forall x \in \mathbb{R}^n\}.$$

For a proper function $g$ admitting an affine minorant on $\mathbb{R}^n$, its conjugate function $g^*$ is defined by

$$g^*(y) := \sup\left\{\langle y, x\rangle - g(x) : x \in \mathbb{R}^n\right\}, \tag{12}$$

and its biconjugate is the function $g^{**} := (g^*)^*$. Recall that $g^{**}$ is the greatest proper convex lower semicontinuous minorant of $g$ on $\mathbb{R}^n$ and $f \in \Gamma_0(\mathbb{R}^n)$ if and only if $f = f^{**}$.

## 2 A Convex relaxation technique

Let $\tau := \sqrt{\frac{\rho}{\lambda}}$ and $\mu := \frac{1}{\lambda}$. To simplify the presentation, in what follows we consider the $\ell_2$ -$\ell_0$-regularized problem (4) in the form

$$(P^{\mu,\tau}) \quad v := \min_{(x,y)\in\mathbb{R}^n\times\mathbb{R}^p} \left\{ F^{\mu,\tau}(x,y) := \mu f(x,y) + \|x\|_2^2 + \tau^2 \|x\|_0 \right\}. \tag{13}$$

The $\ell_0$ term $\|.\|_0$ in (13) makes the problem nonconvex, discontinuous, NP-hard and intractable directly in general. To circumvent these difficulties, we propose to replace the nonconvex $\ell_2$-$\ell_0$ regularized term

$$\|.\|_2^2 + \tau^2 \|.\|_0 \tag{14}$$

by its convex biconjugate function (its *greatest convex lower semicontinuous minorant* on $\mathbb{R}^n$)

$$(\|.\|_2^2 + \tau^2 \|.\|_0)^{**}, \tag{15}$$

to build the following convex relaxation of (4)

$$(\mathrm{CR}) \quad \begin{cases} \min\ G^{\mu,\tau}(x,y) := \mu f(x,y) + (\|.\|_2^2 + \tau^2 \|.\|_0)^{**}(x) \\ s.t.\ \ x \in \mathbb{R}^n,\ y \in \mathbb{R}^m. \end{cases} \tag{16}$$

As $(\|.\|_2^2 + \tau^2 \|.\|_0)^{**} \le (\|.\|_2^2 + \tau^2 \|.\|_0)$, we always have $G^{\mu,\tau}(x,y) \le F^{\mu,\tau}(x,y)$ and the optimal value of (CR) is a lower bound of $v$.

### 2.1 Computation of $(\|.\|_2^2 + \tau^2 \|.\|_0)^{**}$

**Proposition 1** *The biconjugate of the $\ell_2$-$\ell_0$ regularized function is computed by*

$$(\|.\|_2^2 + \tau^2 \|.\|_0)^{**}(x) = \|x\|_2^2 - \left\|(\tau e - |x|)^+\right\|_2^2 + \tau^2 n. \tag{17}$$

*Proof* Let $\varphi : \mathbb{R} \to \mathbb{R}$ be the function defined by $\varphi(r) := r^2 + \tau^2 |r|_0$. We have $\|x\|_2^2 + \tau^2 \|x\|_0 = \sum_{i=1}^n \varphi(x_i)$ separable and $\varphi$ is nonnegative, finite and lower semicontinuous on $\mathbb{R}$. According to the well known result on the conjugate and biconjugate of a separable function [38] we have

$$(\|.\|_2^2 + \tau^2 \|.\|_0)^{**}(x) = \sum_{i=1}^n \varphi^{**}(x_i). \tag{18}$$

$\varphi^{**}$ is the upper envelope of all affine minorants of $\varphi$ on $\mathbb{R}$, i.e., for $t \in \mathbb{R}$,

$$\begin{aligned} \varphi^{**}(t) &= \sup\{at + b :\ a \in \mathbb{R}, b \in \mathbb{R},\ az + b \\ &\le \varphi(z), \forall z \in \mathbb{R}\} \\ &= \sup\left\{at + b :\ a \in \mathbb{R}, b \in \mathbb{R},\ az + b \le z^2 \right. \\ &\quad \left. + \tau^2 |z|_0, \forall z \in \mathbb{R}\right\}. \end{aligned}$$

The condition

$$az + b \le z^2 + \tau^2 |z|_0, \forall z \in \mathbb{R}$$

is equivalent to

$$(b \le 0 \text{ and } az + b \le z^2 + \tau^2, \forall z \in \mathbb{R}, z \ne 0)$$

which is also equivalent to

$$(b \le 0 \text{ and } az + b \le z^2 + \tau^2, \forall z \in \mathbb{R}).$$

Using the discriminant of the second degree polynomial $z^2 - az - b + \tau^2$, the condition $az + b \le z^2 + \tau^2$, $\forall z \in \mathbb{R}$ can be rewritten as $\Delta := a^2 - 4(\tau^2 - b) \le 0$. Then we obtain

$$\begin{aligned} \varphi^{**}(r) &= \sup\left\{at + b :\ a \in \mathbb{R}, b \le 0,\ a^2 - 4(\tau^2 - b) \le 0\right\} \\ &= \sup\left\{at + b :\ a \in \mathbb{R}, b \le 0, b \le \tfrac{4\tau^2 - a^2}{4}\right\} \\ &= \sup\left\{at + \tfrac{1}{4}\min(4\tau^2 - a^2, 0) :\ a \in \mathbb{R}\right\} \\ &= \sup\left\{a|t| + \tfrac{1}{4}\min(4\tau^2 - a^2, 0) :\ a \ge 0\right\} \\ &= \sup\left\{a|t| + \tfrac{1}{4}(4\tau^2 - a^2) :\ a \ge 2\tau\right\} \\ &= \sup\left\{-\tfrac{1}{4}a^2 + a|t| + \tau^2 :\ a \ge 2\tau\right\} \\ &= \begin{cases} t^2 + \tau & \text{if } |r| \ge \tau \\ 2\tau|t| & \text{otherwise} \end{cases} \\ &= t^2 - \left[(\tau - |t|)^+\right]^2 + \tau^2. \end{aligned}$$

Combining this and (18) we get (17). □

On Fig. 1, we illustrate the functions $\|.\|_2^2 + \tau^2 \|.\|_0$, $(\|.\|_2^2 + \tau^2 \|.\|_0)^{**}$ and the convex approach $\ell_2$-$\ell_1$, $\|.\|_2^2 + \tau^2 \|.\|_1$, in the one-dimensional case ($n = 1$).

We are now in a position to give the explicit formulation of the convex relaxation program (CR) of (13).

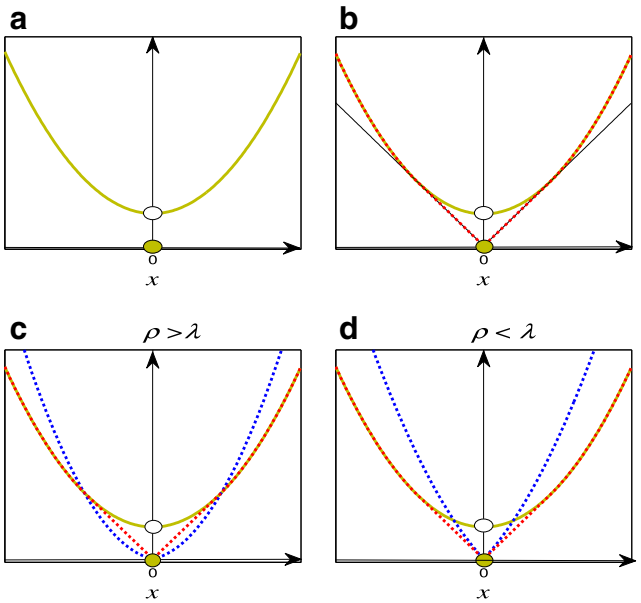


**Fig. 1** Graph of $\varphi$ in green, $\varphi^{**}$ in red and the $\ell_2$-$\ell_1$ approximation in blue

### 2.2 Convex relaxation formulation of (4)

From (17), the explicit formulation of (CR) can be written as

$$\begin{cases} \min\ \mu f(x,y) + \|x\|_2^2 - \left\|(\tau e - |x|)^+\right\|_2^2 + \tau^2 n \\ s.t.\quad x \in \mathbb{R}^n,\ y \in \mathbb{R}^p. \end{cases} \tag{19}$$

The formulation of (19) can be refined according to the definition of the function $f$ in order to get an efficient solver for the resulting convex program. For example, we can express the term $\|x\|_2^2 - \left\|(\tau e - |x|)^+\right\|_2^2$ as a convex quadratic function and get another formulation of (19) as shown in (24) below. This formulation is interesting when the function $f$ is quadratic or linear, because it becomes a convex quadratic program for which several efficient solvers are available.

Let $K$ be the closed convex cone $\mathbb{R}^n_+$, then its polar cone $K^o := \{y \in \mathbb{R}^n : \langle x, y\rangle \le 0, \forall x \in K\}$ is $\mathbb{R}^n_-$ and there hold the following well known properties:

i) For $u \in \mathbb{R}^n$, $u^+ = \max(0, u)$ is the projection of $u$ on $K$, i.e. the solution of

$$\min\{\|u - x\|_2 : x \in K\}$$

and $-u^- = -\max(0, -u)$ is the projection of $u$ on $K^o$, i.e. the solution of

$$\min\{\|u - x\|_2 : x \in K^o\}$$

$$ii) u = u^+ - u^-,\ |u| = u^+ + u^-,\ \langle u^+, u^-\rangle = 0, \text{ and } \|u\|_2^2 = \|u^+\|_2^2 + \|u^-\|_2^2. \tag{20}$$

Therefore, (19) can be rewritten as

$$\begin{array}{l} \min\ \mu f(x,y) + \|x\|_2^2 - \|\tau e - |x|\|_2^2 + \|(\tau e - |x|)^+ - (\tau e - |x|)\|_2^2 \\ s.t.\quad x \in \mathbb{R}^n,\ y \in \mathbb{R}^p. \end{array} \tag{21}$$

It follows from (20) that the problem (21) is equivalent to

$$\begin{array}{l} \min\ \mu f(x,y) + \|x\|_2^2 - \|\tau e - |x|\|_2^2 + \|u - (\tau e - |x|)\|_2^2 \\ s.t.\quad x \in \mathbb{R}^n,\ y \in \mathbb{R}^p,\ u \in \mathbb{R}^n_+ \end{array} \tag{22}$$

in the sense that $(\bar{x}, \bar{y})$ *is an optimal solution of* (21) *iff* $(\bar{x}, \bar{y}, \bar{u} = [\tau e - |\bar{x}|]^+)$ *is an optimal solution of* (22).

The last problem can be written in a simpler form

$$\begin{cases} \min\ \mu f(x,y) + \|x\|_2^2 + \|u\|_2^2 + 2\,|x|^T u - 2\tau e^T u \\ s.t.\quad (x,y,u) \in \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^n_+ \end{cases}$$

which is equivalent to

$$\begin{cases} \min\ \mu f(x,y) + \|\varsigma\|_2^2 + \|u\|_2^2 + 2\varsigma^T u - 2\tau e^T u \\ s.t.\quad |x| \le \varsigma,\ (x,\varsigma,y,u) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^n_+ \end{cases} \tag{23}$$

or again

$$\begin{cases} \min\ G^{\mu,\tau}(x,y,\varsigma,\vartheta) := \mu f(x,y) + q(\varsigma,u) \\ s.t.\quad |x| \le \varsigma,\ (x,\varsigma,y,u) \in C \times \mathbb{R}^p \times \mathbb{R}^n_+, \end{cases} \tag{24}$$

where

$$q(\varsigma,u) := \|\varsigma\|_2^2 + \|u\|_2^2 + 2\varsigma^T u - 2\tau e^T u;$$
$$C := \left\{(x,\varsigma) \in \mathbb{R}^n \times \mathbb{R}^n : |x| \le \varsigma\right\}.$$

### 2.3 Link with the hard-threshold operation

Let $HT_s(.)$ be defined by

$$HT_s(|\theta|) = s^2 - (|\theta| - s)^2 I(|\theta| < s), \tag{25}$$

with $I(|\theta| < s) := 1$ if $|\theta| < s$ and 0 otherwise.
One can see (19) as an approximated problem of (13) in which the term $\|x\|_0$ is replaced by

$$\frac{1}{\tau^2}\sum_{i=1}^n HT_\tau(|x_i|). \tag{26}$$

Hence, this convex relaxation is a special hard-threshold operation. In general, a hard-thresholding operation of $\ell_0$-norm involves a nonconvex program for which only local solutions are guaranteed by iterative methods and there are some difficulties for setting the threshold parameter $s$. The nice effect of our approach (19) resides in the fact that the resulting program is *convex* with its explicit threshold parameter $s = \tau$.

### 2.4 Optimality conditions: links between the convex relation (19), the $\ell_2$-regularized and $\ell_2$- $\ell_0$-regularized problems

Consider the convex relaxation (19) of $(P^{\mu,\tau})$, its optimality condition can be expressed as follows

$$(0,0) \in \partial G^{\mu,\tau}(x^*, y^*) \subset \mathbb{R}^n \times \mathbb{R}^p. \tag{27}$$

This is equivalent to the following condition: there exists $u \in \mathbb{R}^n$ such that

$$\begin{array}{l} (u,0) \in \mu\partial f(x^*, y^*) \text{ and} \\ \quad -u \in \mu\partial\left[\|.\|_2^2 - \left\|(\tau e - |.|)^+\right\|_2^2\right](x^*). \end{array} \tag{28}$$

Let $(x^*, y^*)$ be an optimal solution of the convex relaxation problem (19). It is interesting to study when $(x^*, y^*)$ becomes an optimal solution to the original problem $(P^{\mu,\tau})$. Since $G^{\mu,\tau}(x^*, y^*) \le \nu \le F^{\mu,\tau}(x,y)\ \forall (x,y) \in \mathbb{R}^n \times \mathbb{R}^p$, it is clear that if $G^{\mu,\tau}(x^*, y^*) = F^{\mu,\tau}(x^*, y^*)$, say $(\|.\|_2^2 + \tau^2\|.\|_0)^{**}(x^*) = (\|.\|_2^2 + \tau^2\|.\|_0)(x^*)$, then $(x^*, y^*)$ is also an optimal solution to $(P^{\mu,\tau})$. Hence the following useful results are immediate .

**Proposition 2** *1) Let* $(x^*, y^*)$ *be an optimal solution of* (19)*, i.e.,* $(x^*, y^*)$ *satisfying* (28)*. If*

$$\min\{|x_i^*| : i \in \sup p(x^*)\} \ge \tau, \tag{29}$$

*then* $(x^*, y^*)$ *is an optimal solution of* $(P^{\mu,\tau})$.
*2)* $(0, y^*)$ *is an optimal solution of* (19) *if and only if there exists* $u \in [-2\tau, 2\tau]^n$ *satisfying* $(u, 0) \in \partial f(0, y^*)$. *In this case,* $(0, y^*)$ *is also an optimal solution of* $(P^{\mu,\tau})$.

*Proof* 1) is straightforward because the condition $|x_i^*| \geq \tau$ for all $i \in \text{supp}(x^*)$ implies that $(\|.\|_2^2 + \tau^2 \|.\|_0)^{**}(x^*) = (\|.\|_2^2 + \tau^2 \|.\|_0)(x^*)$ and then $G^{\mu,\tau}(x^*, y^*) = F^{\mu,\tau}(x^*, y^*) = \nu$.

2. It is clear that with $x^* = 0$, the second condition in (28) becomes $u \in [-2\tau, 2\tau]^n$. The proof of the first part is then complete. The second part comes from the fact that when $x^* = 0$ one has $G^{\mu,\tau}(x^*, y^*) = F^{\mu,\tau}(x^*, y^*)$. □

Now let $(x^{\mu,0,*}, y^{\mu,0,*})$ be an optimal solution of the $\ell_2$-regularized problem (i.e. (13) without the $\ell_0$ -term)

$$(P^{\mu,0}) \quad \min \left\{ \mu f(x, y) + \|x\|_2^2 : \ (x, y) \in \mathbb{R}^n \times \mathbb{R}^p \right\}.$$

The following proposition shows the link between the solutions of $\ell_2$ -regularized and $\ell_2 - \ell_0$-regularized problems and gives another sufficient optimality condition for the problem $(P^{\mu,\tau})$.

**Proposition 3** $(x^{\mu,0,*}, y^{\mu,0,*})$ *is an optimal solution to* $(P^{\mu,\tau})$ *for all* $\mu > 0$, $\tau > 0$ *satisfying*

$$\min \left\{ |x_i^{\mu,0,*}| : \ i \in supp(x^{\mu,0,*}) \right\} \geq \tau \ . \tag{30}$$

*Proof* Let $0 < \tau \leq \min \left\{ |x_i^{\mu,0,*}| : \ i \in \text{supp}(x^{\mu,0,*}) \right\}$. From 1) of Proposition 2, it suffices to show that $(x^{\mu,0,*}, y^{\mu,0,*})$ is an optimal solution of (19). Clearly, $(x^{\mu,0,*}, y^{\mu,0,*})$ is an optimal solution of the convex program $(P^{\mu,0})$ iff

$$(0, 0) \in \partial f(x^{\mu,0,*}, y^{\mu,0,*}) + (2x^{\mu,0,*}, 0).$$

It is clear that $(x^{\mu,0,*}, y^{\mu,0,*})$ and $u^* := -2x^{\mu,0,*}$ satisfy the optimality condition (28) for (19). Thus $(x^{\mu,0,*}, y^{\mu,0,*})$ is an optimal solution of (19). □

*Remark 1*

i) Proposition 3 gives an interesting interpretation in the use of the $\ell_2$-regularized problem in practice. It shows that, in some cases, the ridge regularization in learning methods affects not only on predictor but also on sparsity.
ii) Meanwhile, we observe that the conditions (29) and (30) are too strong, they hold only when $\tau = \sqrt{\rho/\lambda}$ is quite small, i.e., when the $\ell_0$-regularized term does not play an important role in the $\ell_2 - \ell_0$ regularized problem. For example, in our first experiment we check the condition (29) on the same dataset when $\tau$ varies, (29) holds in 8/18 cases with $\tau^2$ taking a value in the set $\{0.1, 0.2, 0.5, 1, 2, 10\}$. In other words, to produce sparse solutions $\tau$ should not be small, in this case an optimal solution of the corresponding convex program (19) is only an approximate solution to the original problem $(P^{\mu,\tau})$. Such a solution could be refined by considering a better approximation of the $\ell_2 - \ell_0$ term. This shoul be donne by using a nonconvex approximation, since the biconjugate of the $\ell_2$-$\ell_0$ regularized function is its tightness convex lower bound. Hence we are motivated to develop a two phase algorithm that combines convex relaxation - nonconvex approximation approaches.

### 2.5 On solution methods for the convex program (19 )

Since (19) is a convex program, one can use several standard algorithms and software in convex programming for solving it. For machine learning applications where one is often faced with large scale setting problems, it is important to develop fast and scalable algorithms. Such approaches should exploit the structure and properties of the convex function $f$. Efficient specific numerical methods should be developed for each problem when the function $f$ is given. Although convex programming has been studied for about a century, much effort has been put recently into developing fast and scalable algorithms to deal with large scale problems. While some convex regularizations involve convex quadratic programs (QP) for which standard QP solvers can be certainly used, many first-order methods have been developed in the last years for large scale convex problems, e.g. the coordinate gradient descent [42], the fast iterative shrinkage-thresholding algorithms [1], smoothing proximal gradient methods [5].

Since convex programs constitute a nice class of DC programs for which DCAs converge to optimal solutions, DCA can be used to solve the convex program ( 19). Assume that there exists a nonnegative number $\eta$ such that the function $\frac{1}{2}\eta\|(x, y)\|^2 - \mu f(x, y)$ is convex (in many practical problems such a $\eta$ exists and can be easily computed; for example, when $f$ is a smooth function with Lipschitz continuous gradient, we can take $\eta := \mu L$, where $L$ is the Lipschitz constant of $\nabla f$). Then we can derive a DCA scheme which is the first order method based on the projection onto $C$ and onto $\mathbb{R}^n_+$.

In this paper, as we focus on the tightness of the proposed convex regularization and its effect in the combined convex-nonconvex approaches, we simply use, in our experiment on feature selection in SVM, the CPLEX software to solve the convex program (24). It is in fact a quadratic program (note that this software uses efficient techniques for large scale setting such as interior points methods).

In the next section we will present DCA for solving a nonconvex approximate problem of (4).

## 3 Nonconvex approximation approaches

As nonconvex approximation approaches produce, in general, good sparsity, we can improve the convex regularization approach by solving, in the second step, a resulting nonconvex approximation problem from the solution given by the convex approach.

Nonconvex approximation approaches for sparse optimization involving a DC function and the $\ell_0$ term have been intensively studied in [24] in the unified DC programming framework. Considering a class of DC approximation functions of the zero-norm including all usual sparse inducing approximation functions, the authors have proved several novel and elegant results concerning the consistency between global (resp. local) minimizers of the approximate problem and the original problem, the equivalence between these two problems in some cases, etc, and have developed various DCA schemes that cover all standard nonconvex approximation algorithms as special versions.

In this section, we adapt the first DCA scheme proposed in [24] for solving the problem (4) where the function $f$ is convex (but not "real" DC as considered in [24]). For some practical problems this DCA scheme enjoys interesting convergence properties and it has been shown to be the most efficient among DCA based algorithms proposed in [24]) for feature selection in SVM.

Before presenting this DCA based algorithm, let us describe the philosophy of DCA.

### 3.1 Philosophy of DCA

DCA [20, 23, 34, 35] aims to solve a nonconvex program of the form

$$\inf\{F(x) := G(x) - H(x) : x \in \mathbb{R}^n\} \qquad (P_{dc})$$

where $G, H \in \Gamma_0(\mathbb{R}^n)$ (the convex cone of all lower semi-continuous proper convex functions defined on $\mathbb{R}^n$ and taking values in $\mathbb{R} \cup \{+\infty\}$. A convex constrained DC problem with the constraint $x \in C$ can be rewritten in the form $(P_{dc})$ by adding the indicator function of $C$, denoted by $\chi_C$, $\chi_C(x) = 0$ if $x \in C$, and $+\infty$ otherwise) into $G$:

$$\inf\{F(x) := G(x) - H(x) : x \in C\} \Leftrightarrow \inf\{(\chi_C(x) + G(x)) - H(x) : x \in \mathbb{R}^n\}$$

The main idea of DCA is simple: each iteration of DCA approximates the concave part $-H$ by its affine majorization (that corresponds to taking $y^k \in \partial H(x^k)$) and solves the resulting convex program:

**DCA - general scheme initializations** let $x^0 \in \mathbb{R}^n$ be a guess, set $k := 0$.
**repeat**
1. calculate $y^k \in \partial H(x^k)$.
2. calculate $x^{k+1} \in \arg\min\{G(x) - \langle x, y^k\rangle : x \in \mathbb{R}^n\}$ $(P_k)$.
3. $k = k + 1$.
**until** convergence of $\{x^k\}$.

It has been proved in [23, 34, 35] that DCA is a descent method without linesearch, and either the sequence $x^k$ converges after a finitely number of iterations to a critical point of $G - H$, or if the infinite sequence $\{x^k\}$ is bounded and the optimal value of problem $(P_{dc})$ is finite then every limit point $x^*$ of the sequence $\{x^k\}$ is a critical point of $G - H$.

The construction of DCA, and so its efficiency, depends on the choice of the functions $G$ and $H$ and the so called DC composition $G - H$. The flexibility of DCA according to the choice of DC decomposition is a crucial point to design efficient DCA based algorithms. It is worth noticing that with suitable DC decomposition DCA recovers most of standard methods in convex and nonconvex programming, in particular the three popular methods in machine learning, namely the EM (Expectation-Maximization) ([9] ), the SLA (Succesive Linear Approximation) ([3]) and the CCCP (Convex-Concave Procedure) ([44]).

DCA has been successfully applied to many (smooth or nonsmooth) large-scale nonconvex programs in various domains of applied sciences, in particular in Machine Learning (see e.g. [7, 12, 19, 21, 22, 25, 27–32, 48–52]) for which they provided quite often global solutions and proved to be more robust and efficient than standard methods.

### 3.2 A DCA based algorithm for nonconvex approximation problems

By the definition, the step function $|.|_0 : \mathbb{R} \to \mathbb{R}$ is given by $|t|_0 = 1$ for $t \neq 0$ and 0 otherwise. Then $\|x\|_0 = \sum_{i=1}^n |x_i|_0$. The idea of approximation methods is to replace the discontinuous step function by a continuous approximation function, denoted $r_\theta$, where $\theta > 0$ is a parameter controlling the tightness of approximation.

By the way, the original problem

$$\min_{(x,y)\in\mathbb{R}^n\times\mathbb{R}^p} \left\{ F^{\lambda,\rho}(x, y) := f(x, y) + \lambda\|x\|_2^2 + \rho\,\|x\|_0 \right\}.$$

becomes

$$\min\left\{ F_\theta(x, y) = f(x, y) + \lambda\|x\|_2^2 + \rho \sum_{i=1}^n r_\theta(x_i) : (x, y) \in \mathbb{R}^n\times\mathbb{R}^p \right\}. \tag{31}$$

With the following DC decomposition of $r_\theta$:

$$r_\theta(t) = \eta|t| - (\eta|t| - r_\theta(t)) \quad \forall t \in \mathbb{R}, \tag{32}$$

where $\eta$ is a positive number such that $\psi(t) = \eta|t| - r_\theta(t)$ is convex (the existence of such a $\eta$ has been proved in [24] ), a DC formulation of the problem (31) is given by

$$\min_{x,y}\{F_\theta(x, y) := G(x, y) - H(x, y)\}, \tag{33}$$

where

$$G(x, y) = f(x, y) + \lambda\|x\|_2^2 + \rho\eta\|x\|_1,$$

$$H(x, y) = \rho\sum_{i=1}^{n}(\eta|x_i| - r_\theta(x_i)),$$

Note that the use of the DC approximation $r_\theta$ of the form (32) aims at introducing $\rho\eta\|.\|_1$ in the DC program (33) . It has been stated in [24] a list of continuous such functions $r_\theta$, which contains all standard DC approximations and the explicit computation of their corresponding subdifferential $\partial\psi$ . Following the generic DCA scheme described above, DCA applied to (33) is given by Algorithm 1 below, for a given sparse inducing function $r_\theta$.

**Algorithm 1** DCA for solving (31)

Initialize $(x^0, y^0) \in \mathbb{R}^n \times \mathbb{R}^p$, $k \leftarrow 0$
**repeat**
1. Compute $\overline{z}_i^k \in \lambda\partial\psi(x_i^k)\ \forall i = 1, \ldots, n.$
2. Compute
$$(x^{k+1}, y^{k+1}) \in \arg\min_{(x,y)\in K}\Big\{f(x, y) + \lambda\|x\|_2^2 + \rho\eta\|x\|_1 - \rho\langle\overline{z}^k, x\rangle\Big\}$$
3. $k \leftarrow k + 1$.
**until** Stopping criterion

### 3.3 The two-phase algorithm

As mentioned in the introduction, convex regularization approaches involve convex optimization problems which are so far "easy" to solve. However, even if our convex relaxation is a special hard-threshold operation (hence it can promote sparsity), nonconvex approaches are still needed to produce better sparsity (see Remark 1). But the resulting optimization problems are very hard. Several sparse inducing nonconvex functions and corresponding algorithms are proposed in the literature, they are all special versions of DCA (see [24]). Due to its local character, finding a good starting point is important for DCA to reach global solutions. Using the solution of the convex relaxation problem seems to be a good fit for that purpose. It is therefore suggested to design a two-phase algorithm combining convex and nonconvex approaches. In the first phase the convex relaxation is performed and in the second phase an efficient DCA based algorithm is used for the nonconvex approximation problem, starting from the solution given by Phase 1. One can see that the solution given by the convex relaxation is refined (to be sparser) via the second phase via a closer (nonconvex) approximation of the $\ell_2$-$\ell_0$-term. So, this method can exploit simultaneously the advantage of both convex and nonconvex approximation approaches.

## 4 Application to feature selection in SVM

### 4.1 Convex relaxation formulation for feature selection in SVM model (6)

As mentioned in Section 1, the ($\ell_2$-$\ell_0$-SVM) problem takes the form (with the use of $\mu$ and $\tau$ instead to $\lambda$ and $\rho$):

$$\begin{cases}\min \frac{\mu}{m}\sum_{i=1}^{m}\max(0, 1-b_i(\langle a_i, w\rangle+\gamma))+\|w\|_2^2+\tau^2\|w\|_0\\ s.t.\ \ (w, \gamma) \in \mathbb{R}^n \times \mathbb{R}\end{cases} \tag{34}$$

or again

($\ell_2$-$\ell_0$-SVM)
$$\begin{cases}\min\ F^{\mu,\tau}(w, \gamma, \xi) := \frac{\mu}{m}e^T\xi + \|w\|_2^2 + \tau^2\|w\|_0\\ s.t.\ \ b_i(a_i^T w + \gamma) \geq 1 - \xi_i,\ i = 1, ..., m,\\ \qquad (w, \gamma, \xi) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}_+^m\end{cases} \tag{35}$$

Its convex relaxation formulation (24) becomes

(CR-SVM)
$$\begin{cases}\min\ G^{\mu,\tau}(w, \gamma, u, \varsigma, \xi) := \frac{\mu}{m}e^T\xi + q(\varsigma, u)\\ s.t.\ \ b_i(a_i^T w + \gamma) \geq 1 - \xi_i,\ i = 1, ..., m,\\ \qquad |w| \leq \varsigma, (w, \gamma, u, \varsigma, \xi) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}_+^n \times \mathbb{R}^n \times \mathbb{R}_+^m\end{cases} \tag{36}$$

which is a convex quadratic program.

### 4.2 A combined convex-nonconvex approximation approach: the two-phase algorithm

For feature selection in classification, convex regularization approaches perform better in classification while nonconvex approximation approaches, with a "good" approximation of the $\ell_0$-norm and based on an efficient algorithm for nonconvex resulting optimization problem, produce better sparsity. In order to get both quality and sparsity of the classifier, we use a two-phase algorithm.

A state-of-the-art algorithm for the problem ($\ell_2$-$\ell_0$-SVM) is the DCA scheme developed in [31] with the concave exponential approximation proposed in [3]:

$$r_\alpha(x) = \begin{cases}1 - \varepsilon^{-\alpha x} & \text{if } x \geq 0,\\ 1 - \varepsilon^{\alpha x} & \text{if } x < 0,\end{cases} \tag{37}$$

In this approach, the resulting nonconvex problem takes the form

($\ell_2$-$Exp$-SVM)
$$\begin{cases} \min \ \frac{\mu}{m} e^T \xi + \|w\|_2^2 + \tau^2 e^T (e - exp(-\alpha\,|w|)) \\ s.t. \ \ b_i(a_i^T w + \gamma) \geq 1 - \xi_i, \ i = 1, ..., m, \\ \quad (w, \gamma, \xi) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^m_+, \end{cases}$$

and the DCA based algorithm requires solving one quadratic program at each iteration (see [31] for more details).

It has been shown in [30] and [24] that, for the concave exponential approximation [3], the DC decomposition (32) is better than the one used in [31]. Therefore we employ this DC decomposition and Algorithm 1 for solving the approximate problem ($\ell_2$-$Exp$-SVM). Algorithm 1 with the function $r$ defined in (37) is described as follows.

**Algorithm 2** Algorithm $(\ell_2 - \ell_0)$-DCA1 (DCA for solving (31))

**Initializations:** let $\epsilon > 0$ be given and $X^0 = (w^0, \xi^0, b^0)$ be an initial point. Set $k \leftarrow 0$

**repeat**

1. Compute, for $j = 1..n$

$$\overline{w}_j^k = \begin{cases} \alpha(1 - \varepsilon^{-\alpha w_j^k}) & \text{if} \ \ w_j^k \geq 0 \\ -\alpha(1 - \varepsilon^{\alpha w_j^k}) & \text{if} \ \ w_j^k < 0 \end{cases} \tag{38}$$

and set $Y^k = (\overline{w}^k, 0, 0, 0)$.

2. Compute $X^{k+1}$, an optimal solution of the following convex quadratic program

$$\Leftrightarrow \min_{w,b,\xi,t} \begin{cases} \frac{\mu}{m} e^T \xi + \|w\|_2^2 + \tau^2 \sum_{j=1}^n t_j - \tau^2 \sum_{j=1}^n \overline{w}_j^k w_j \\ s.t. \ \ (w, b, \xi) \in \Omega, t_j \geq \alpha w_j, t_j \geq -\alpha w_j, j = 1..n. \end{cases} \tag{39}$$

3. $k \leftarrow k + 1$.

**until** $\|X^{k+1} - X^k\| \leq \epsilon \|X^k\|$.

Note that DCA has been successfully applied in several works on feature selection in classification [3, 4, 22, 30–32], and sparse signal recovery [12, 26], in particular it furnished a good sparse solution. Here, we hope that the two-phase algorithm performs classification like convex relaxation approach and produces sparsity like DCA applied on ($\ell_2$-$Exp$-SVM).

**Two-phase algorithm** (for solving ($\ell_2$-$\ell_0$ -SVM))

**Phase 1**. Solve the convex program (CR-SVM) to get an optimal solution
$(w^{CR}, \gamma^{CR}, \vartheta^{CR}, \varsigma^{CR}, \xi^{CR})$.

**Phase 2**. Apply Algorithm $(\ell_2 - \ell_0)$-DCA1 from the starting point $(w^{CR}, \gamma^{CR})$.

## 5 Numerical experiments

For evaluating the effectiveness of the proposed approaches (the convex relaxation approach named CR-SVM and the two-phase algorithm) we execute numerical experiments on several datasets and compare them with two state-of-the-art algorithms for the ($\ell_2$-$\ell_0$ regularized SVM: the convex approach ($\ell_2$-$\ell_1$-SVM)

($\ell_2$-$\ell_1$-SVM)
$$\begin{cases} \min \ \frac{\mu}{m} e^T \xi + \|w\|_2^2 + \tau^2 \|w\|_1 \\ s.t. \ \ b_i(a_i^T w + \gamma) \geq 1 - \xi_i, \ i = 1, ..., m \\ \quad (w, \gamma, \xi) \in \mathbb{R}^n \times \mathbb{R} \times \mathbb{R}^m_+ \end{cases}$$

and the nonconvex approach developed in [31]. We also consider the ($\ell_2$-SVM) approach which measures more or less the difficulty degree of classification task in test problems. All algorithms are coded in $C++$ and tested on an Intel Core$^{TM}$ I7 ($2 \times 2.2$ Ghz) processor of 4 Gb RAM. The three convex approaches solve one convex quadratic problem while the nonconvex approach requires solving one convex quadratic program at each iteration. We use CPLEX solver library 13.2 for solving convex quadratic programs. The number of nonzero features in $w$ are determined by $card\{j : |w_j| > 10^{-5}\}$.

### 5.1 Data

We consider 8 datasets which can be found at the web-site of UCI Machine Learning Repository, and 3 micro-array datasets - Leukemia cancer [13], prostate cancer [39] and Lung cancer [14]. They are described in detail in the Tables 1 and 2.

We consider also a synthetic data in which six features of 202 were relevant. The probability of $y = 1$ or $-1$ was equal. The first (resp. second) three features

**Table 1** UCI datasets

| Datasets | Note | #samples | #features | #class+/ #class- |
|---|---|---|---|---|
| ionosphere | INO | 351 | 34 | 225/126 |
| cleveland | CLE | 297 | 13 | 160/137 |
| pima | PIM | 768 | 8 | 500/268 |
| breast cancer wisconsin | BCW | 683 | 9 | 444/239 |
| sona | SON | 208 | 60 | 111/97 |
| internet advertisements | ADV | 3279 | 1558 | 459/2820 |
| spambase | SPA | 4601 | 57 | 1813/2788 |
| gisette | GIS | 7000 | 5000 | 3500/3500 |

**Table 2** Micro-array datasets

| Data sets | no. of features | no. of training samples | no. of testing samples |
|---|---|---|---|
| Leukemia | 7129 | 38 | 34 |
| Prostate cancer | 12600 | 102 | 34 |
| Lung cancer | 12533 | 32 | 149 |

$x_1, x_2, x_3$ (resp. $x_4, x_5, x_6$) were drawn as $x_i = y\mathcal{N}(i, 1)$ (resp. $x_i = \mathcal{N}(0, 1)$) with a probability of 0.7, otherwise the first (resp. second) three as $x_i = \mathcal{N}(0, 1)$ (resp. $x_i = y\mathcal{N}(i - 3, 1)$). The remaining features are noise $x_i = \mathcal{N}(0, 20)$, $i = 1, ..., 202$.

### 5.2 Experiment 1: the tightness of the convex minorant

In the first experiment, our aim is to evaluate the tightness of the proposed lower bound for solving the $\ell_2$-$\ell_0$-SVM problem ( 35). For this purpose, we measure the optimality gap (in %) which is defined by

$$Gap := \frac{Ub - Lb}{|Ub|} \times 100. \tag{38}$$

Here $Ub := F^{\mu,\tau}(w^{CR}, \gamma^{CR})$ and $Lb := G^{\mu,\tau}(w^{CR}, \gamma^{CR})$ are, respectively, an upper bound and a lower bound of the optimal value of (35) which are given by $(w^{CR}, \gamma^{CR})$, an optimal solution of the problem (CR-SVM).

We consider the Inosphere dataset and run CR-SVM on various problems with different values of $\lambda$ and $\rho$ (the coefficient of the $\ell_2$ -term and $\ell_0$-term). The results are reported in Table 3.

The columns "Gap" in Table 3 show that the lower bound $G^{\mu,\tau}(w^{CR}, \gamma^{CR})$ obtained from CR-SVM is very close to the upper bound $F^{\mu,\tau}(w^{CR}, \gamma^{CR})$ (and then close to the optimal value): the mean of gap is only 0.24%. Moreover, in 8/18 cases this lower bound is exactly the optimal value of the corresponding $\ell_2$-$\ell_0$-SVM problem., i.e. the sufficient optimality condition ( 29) holds and CR-SVM gives an optimal solution to $\ell_2$-$\ell_0$-SVM problem. This shows that our convex approach CR-SVM is very promising for solving this type of problem.

### 5.3 Experiment 2: comparison between the two convex approaches on the synthetic data and the micro-array data

First, we evaluate the performance of the CR-SVM and the $\ell_2 - \ell_1$ approaches in terms of feature selection and classification on the synthetic data. The results are reported in Table 4 for various training set sizes, taking the average test error on 500 samples over 30 runs of each training set size. The set of parameters

$\Lambda := \{0.0001; 0.0002; 0.0003; 0.0004; 0.0005\}$ and

$\Gamma := \{0.1; 0.2; 0.3; 0.4; 0.5\}$

has been used.

From Table 4, we observe that CR-SVM performs better, both in feature selection and classification, than $\ell_2$-$\ell_1$-SVM on this data set. Another important result is that CR-SVM selects a number of features around 6 which represents the number of relevant features on the synthetic data.

Second, we compare the two convex approaches on the three micro-array datasets. Here the training and test sets are explicitly given, and we perform the algorithms on the set of parameters.

$\Lambda := \{0.0001; 0.0002; 0.0003; 0.0004; 0.0005\}$ and $\Gamma := \{0.01; 0.02; 0.03; 0.04; 0.05\}$ to get the best parameters for each algorithm. More precisely, for each $\theta \in \Theta := \Lambda \times \Gamma$, we apply the algorithm on the training set to get classifier and selected features, and then take the best parameter $\theta^* = (\lambda^*, \rho^*)$ as the one corresponding to the

**Table 3** CR-SVM: optimality gap ("Gap"), for solving $\ell_2$-$\ell_0$-SVM on inosphere data

| $\lambda$ | $\rho$ | Gap(%) | $\lambda$ | $\rho$ | Gap(%) |
|---|---|---|---|---|---|
| 0.0001 | 0.0001 | 0.03 | 0.005 | 0.0005 | 0.00 |
| 0.0001 | 0.0005 | 0.33 | 0.005 | 0.01 | 0.00 |
| 0.0001 | 0.001 | 0.62 | 0.005 | 0.05 | 0.00 |
| 0.0005 | 0.0001 | 0.03 | 0.01 | 0.005 | 0.00 |
| 0.0005 | 0.0005 | 0.12 | 0.01 | 0.01 | 0.00 |
| 0.0005 | 0.001 | 0.42 | 0.01 | 0.05 | 0.00 |
| 0.001 | 0.0001 | 0.02 | 0.05 | 0.005 | 0.07 |
| 0.001 | 0.0005 | 0.09 | 0.05 | 0.01 | 0.00 |
| 0.001 | 0.01 | 2.58 | 0.05 | 0.05 | 0.00 |

**Table 4** Synthetic data: number of selected feature (num) and classification error (ERR %)

| | CR-SVM | | $\ell_2$-$\ell_1$-SVM | |
|---|---|---|---|---|
| Training set sizes | Num | ERR | Num | ERR |
| 20 | 15.2 | 11.8 | 43.1 | 18.9 |
| 50 | 7.6 | 1.8 | 19.7 | 4.6 |
| 80 | 4.6 | 1.1 | 20.4 | 2.9 |
| 100 | 5.2 | 1.3 | 22.7 | 2.7 |

**Table 5** Comparative results on micro-array datasets

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.0005 | 0.05 | 0 | 2.94 | 28 | 0.39 | 365.12 |
| $\ell_2 - \ell_1$-SVM | 10 | 5 | 0 | 2.94 | 29 | 0.41 | 310.02 |
| Leukemia dataset | | | | | | | |
| CR-SVM | 0.0005 | 0.05 | 0 | 0 | 45 | 0.36 | 624.12 |
| $\ell_2 - \ell_1$-SVM | 3 | 1 | 0 | 2.94 | 260 | 2.06 | 567.25 |
| Prostate cancer | | | | | | | |
| CR-SVM | 0.0001 | 0.01 | 0 | 0.67 | 24 | 0.19 | 565.12 |
| $\ell_2 - \ell_1$-SVM | 10 | 5 | 0 | 2.01 | 24 | 0.19 | 510.02 |
| Lung cancer | | | | | | | |

**Table 6** Comparative results on PIM dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.011 | 0.015 | 23.75 | 21.93 | 4.00 | 50.00 | 0.057 |
| $\ell_2 - \ell_1$-SVM | 0.003 | 0.017 | 23.36 | 22.81 | 4.00 | 50.00 | 0.036 |
| $\ell_2$-SVM | 0.001 | - | 22.19 | 22.37 | 8.00 | 100.0 | 0.042 |
| $\ell_2$-Exp-SVM | 0.019 | 0.007 | 24.66 | 25.00 | 3.33 | 41.63 | 0.109 |
| Two phase algorithm | 0.019 | 0.007 | 23.17 | 24.12 | 3.67 | 45.88 | 0.131 |

**Table 7** Comparative results on BCW dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.019 | 0.019 | 3.52 | 4.11 | 7.00 | 77.78 | 0.142 |
| $\ell_2 - \ell_1$-SVM | 0.001 | 0.009 | 2.71 | 3.23 | 8.00 | 88.89 | 0.172 |
| $\ell_2$-SVM | 0.017 | - | 2.86 | 2.93 | 9.00 | 100 | 0.151 |
| $\ell_2$-Exp-SVM | 0.001 | 0.017 | 4.56 | 5.33 | 2.33 | 25.89 | 0.057 |
| Two phase algorithm | 0.001 | 0.017 | 3.66 | 3.81 | 2.67 | 29.67 | 0.218 |

**Table 8** Comparative results on CLE dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.019 | 0.019 | 15.51 | 16.26 | 7.67 | 59.00 | 0.037 |
| $\ell_2 - \ell_1$-SVM | 0.003 | 0.019 | 14.16 | 15.91 | 9.33 | 71.77 | 0.042 |
| $\ell_2$-SVM | 0.019 | - | 13.99 | 16.61 | 13.00 | 100.0 | 0.031 |
| $\ell_2$-Exp-SVM | 0.005 | 0.019 | 23.59 | 23.65 | 1.00 | 7.69 | 0.023 |
| Two phase algorithm | 0.005 | 0.019 | 15.68 | 17.27 | 5.00 | 38.46 | 0.078 |

**Table 9** Comparative results on INO dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.019 | 0.019 | 11.54 | 15.10 | 10.33 | 30.38 | 0.058 |
| $\ell_2 - \ell_1$-SVM | 0.001 | 0.019 | 9.69 | 13.96 | 15.00 | 44.12 | 0.067 |
| $\ell_2$-SVM | 0.003 | - | 5.98 | 13.11 | 33.00 | 97.06 | 0.057 |
| $\ell_2$-Exp-SVM | 0.011 | 0.019 | 13.15 | 14.06 | 3.22 | 9.47 | 0.075 |
| Two phase algorithm | 0.011 | 0.019 | 10.83 | 15.1 | 5.00 | 17.71 | 0.145 |

**Table 10** Comparative results on SPA dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.005 | 0.019 | 23.21 | 23.97 | 4.33 | 7.60 | 32.182 |
| $\ell_2 - \ell_1$-SVM | 0.001 | 0.017 | 24.03 | 24.82 | 5.00 | 8.77 | 45.417 |
| $\ell_2$-SVM | 0.001 | - | 10.71 | 12.82 | 57.00 | 100.0 | 20.63 |
| $\ell_2$-Exp-SVM | 0.019 | 0.017 | 27.65 | 29.04 | 1.00 | 1.75 | 0.952 |
| Two phase algorithm | 0.019 | 0.017 | 17.66 | 18.67 | 3.33 | 5.84 | 29.38 |

**Table 11** Comparative results on SON dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.005 | 0.019 | 16.35 | 39.87 | 9.00 | 15.00 | 0.045 |
| $\ell_2 - \ell_1$-SVM | 0.001 | 0.019 | 15.15 | 40.83 | 12.67 | 21.12 | 0.067 |
| $\ell_2$-SVM | 0.003 | - | 11.55 | 44.69 | 60.00 | 100.0 | 0.031 |
| $\ell_2$-Exp-SVM | 0.011 | 0.017 | 24.43 | 32.54 | 3.56 | 5.93 | 0.298 |
| Two phase algorithm | 0.011 | 0.017 | 14.91 | 47.59 | 6.00 | 10.00 | 0.203 |

**Table 12** Comparative results on ADV dataset

| Algorithm | $\lambda^*$ | $\rho^*$ | Class. error | | Select. features | | CPU |
|---|---|---|---|---|---|---|---|
| | | | Train | Test | Number | % | |
| CR-SVM | 0.001 | 0.019 | 3.98 | 4.57 | 7.33 | 0.47 | 207.437 |
| $\ell_2 - \ell_1$-SVM | 0.001 | 0.019 | 5.58 | 6.25 | 3.33 | 0.21 | 193.956 |
| $\ell_2$-SVM | 0.001 | - | 1.11 | 4.33 | 835.00 | 53.59 | 90.579 |
| $\ell_2$-Exp-SVM | 0.011 | 0.009 | 6.95 | 7.14 | 2.00 | 0.13 | 389.648 |
| Two phase algorithm | 0.011 | 0.009 | 3.96 | 4.85 | 5.33 | 0.34 | 588.262 |

best criterion on the test set. As we are interested on both accuracy of classification and sparsity of classifier, the best evaluated criterion used in our experiment is the smallest value of (ERRt+FS)/ACC, where ERRt, FS, ACC denote, respectively, the percentage of classification error on the test set, the percentage of selected features and the percentage of classification accuracy on the test set (i.e., ACC = 100-ERRt). The results are reported in Table 5.

On the three micro-array data sets, we observe that CR-SVM performs better feature selection than the the $\ell_2$-$\ell_1$-SVM on the data sets Leukemia (28 features versus 29 features) and Prostate (45 versus 260 ) and performs equally as well as $\ell_2$-$\ell_1$-SVM on the Lung Cancer. In term of generalization error on the test sets, CR-SVM, $\ell_2$ -$\ell_1$-SVM give equal generalization error on Leukemia (2.94%). On the Prostate data, CR-SVM is more competitive with 0% of generalization versus 2.94% for the $\ell_2$-$\ell_1$-SVM. And finally, CR-SVM* performs better than the others with only 0.67% of generalization error. As for time consuming, $\ell_2$-$\ell_1$-SVM is slightly better than CR-SVM.

### 5.4 Experiment 3: comparison of all approaches on UCI datasets

In this experiment we compare the efficiency of the proposed convex approach (CR-SVM) and the two-phase algorithm with the convex approaches ($\ell_2 - \ell_1$-SVM and $\ell_2$-SVM), as well as the nonconvex approach [31] (denoted $\ell_2$-Exp-SVM).

We fix a finite set of parameters $\Theta := \{\lambda, \rho\}$ and use the ten-fold cross-validation for the choice of the best parameters for each algorithm. More precisely, we divide the dataset into 10 equal parts. For each fold, we set 9 parts as the training set and one as the test set. By changing the test set we get ten folds.

For each $\theta \in \Theta$, we apply the algorithm on each of 10 folds to determine, on each fold, the classifier and selected features on the train set and compute the classification error (ERR) on the test set as well as on the train set. Then the average result on 10 folds is used for determining the best parameters $\theta^* \in \Theta$.

Here, we fix the parameter $\theta^* = (\lambda^*, \rho^*) \in \Theta$ that gives the best average evaluated criterion which is, as in the experiment on the micro-array datasets, (ERRt+FS)/ACC.

The set of parameters for the cross-validation procedure is

$$\Lambda := \{0.001; 0.003; 0.005; 0.007; 0.009\};$$
$$\Gamma := \{0.01; 0.03; 0.05; 0.07; 0.09\}.$$

For $\ell_2$-Exp-SVM, the parameter $\alpha$ of concave approximation function is set to 5 as proposed in [3]. Since $\ell_2$ -Exp-SVM is a local approach which depends on the choice of initial point, for each run, we perform it 10 times from random initial points and report the average results. The other algorithms are performed one time, because they do not depend on initial points.

In Tables 6, 7, 8, 9, 10, 11, 12 we report the best average results on 10 folds (ERR on the train set (Train) and on the test set (Test), the number (Number) and the corresponding percent (%) of selected features (Select. features)). We also indicate the values $\lambda^*$ and $\rho^*$ corresponding to these best results.

We observe from the numerical results that

i) Among the convex approaches, CR-SVM is better than $\ell_2 - \ell_1$ -SVM on both classification and feature selection: in all datasets (except for ADV) CR-SVM suppresses more features than $\ell_2 - \ell_1$-SVM while the ERRt are smaller on 4/7 datasets and very slightly (less than 1%) larger on 3/7 datasets. As for $l_2$, it deals more or less with sparsity on 2 out of 7 dataset (INO and ADV) where it selects 97% and 53.9% features.

ii) Not surprisingly, nonconvex approach $\ell_2$-Exp-SVM is better than convex approaches on feature selection while it is worse than convex approaches on classification.

iii) The two-phase algorithm improves considerably the accuracy of classification of $\ell_2$-Exp-SVM while the

number of selected features of the former is slightly larger than the one of the later. We observe that the two-phase algorithm performs classification like CR-SVM while it selects features like $\ell_2$-Exp-SVM. Hence this algorithm realizes the trade-off between accuracy and sparsity and it is the best approach for simultaneously performing feature selection and classification.

## 6 Conclusion

We have proposed a new convex relaxation technique for minimizing a class of functions involving the zero norm that includes several important problems in machine learning. The approach is based on computing the biconjugate of the nonconvex $\ell_2$-$\ell_0$ regularization function which is its tightest convex minorant. It is worth to note the nice effect of our approach compared with hard-thresholding algorithms: with an appropriate choice of threshold parameter our resulting program is convex while, in general, the hard-threshold approximation of $\ell_0$ involves a nonconvex program. This new and efficient way to deal with the $\ell_0$ norm constitutes the most important contribution of the paper.

Secondly, the idea of combining the two convex - nonconvex approaches is interesting. The two-phase algorithm is a promising approach that we suggest to use for feature selection and classification as well as for other sparse optimization problems. The proposed approaches have been successfully applied to the feature selection in SVM via experiments on several datasets.

The new results developed in this paper open the door to several research issues.

Firstly, the tightness of the proposed lower bound suggests us to develop global algorithms based on this lower bound for the nonconvex $\ell_2$-$\ell_0$ problem and/or for extension cases, say nonconvex programs involving $\ell_0$ norm.

Secondly, this convex relaxation can be useful for the combined convex relaxation - nonconvex approximation approaches. Thirdly, the global optimality condition should be exploited in the above mentioned approaches to check (and/or to get) the globality of solutions. Fourthly, numerical methods for efficiently solving large scale convex relaxation problems should be developed for practical applications, i.e. for various forms of the function $f$ in the considered $\ell_2$-$\ell_0$ problem.

Works on these issues are under progress.

**Acknowledgments** This research is funded by Foundation for Science and Technology Development of Ton Duc Thang University (FOSTECT), website: http://fostect.tdt.edu.vn, under Grant FOSTECT.2015.BR.15.

## References

1. Beck A, Teboulle M (2009) A fast iterative shrinkage thresholding algorithm for linear inverse problems. SIAM J Imag Sci 2(1):183–202
2. Bennett KP, Mangasarian OL (1992) Robust linear programming discrimination of two linearly inseparable. sets Opt Meth Soft 1:23–34
3. Bradley PS, Mangasarian OL (1998) Feature selection via concave minimization and support vector machines. In ICML 1998:82–90
4. Candes E, Wakin M, Boyd S (2008) Enhancing sparsity by reweighted l1 minimization. J Four Anal Appli
5. Chen X, Lin Q, Kim S, Carbonel JC, Xing EP (2012) Smoothing proximal gradient method for general structured sparse regression. Ann Appl Stat 6(2):719–752
6. Chen X, Xu FM, Ye Y (2010) Lower bound theory of nonzero entries in solutions of l2-lp minimization. SIAM J Sci Comp 32(5):2832–2852
7. Collober R, Sinz F, Weston J, Bottou L (2006) Trading convexity for scalability. In: Proceedings of the 23th International Conference on Machine Learning (ICML 2006). Pittsburgh, PA
8. Cortes C, Vapnik V (1995) Support vector networks. Mach Learn 20:273–297
9. Dempster AP, Laird NM (1977) Maximum likelihood from incomplete data via the em algorithm. J Roy Stat Soc B 39:1–38
10. Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. J Amer Stat Ass 96(456):1348–1360
11. Fu WJ (1998) Penalized regression: the bridge versus the lasso. J Comp Graph Stat 7:397–416
12. Gasso G, Rakotomamonjy A, Canu S (2009) Recovering sparse signals with a certain family of nonconvex penalties and dc programming. IEEE Trans Sign Proc 57:4686–4698
13. Golub TR., Slonim DK., Tamayo P, Huard C, Gaasenbeek M, Mesirov JP., Coller H, Loh ML., Downing JR., Caligiuri MA., Bloomfield CD., Lander ES. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Sci 286:531–537
14. Gordon GJ, Jensen RV, Hsiao L, Gullans SR, Blumenstock FE, Ramaswamy R, Richard WG, Sugarbaker DJ, Bueno R (2002) Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. Cancer Res 62:4963–4967
15. Guan A, Gray W (2013) Sparse high-dimensional fractional-norm support vector machine via dc programming. Comput Stat Data Anal 67:136–148
16. Guyon I, Gunn S, Nikravesh M, Zadeh L (2006) Feature Extractions and Applications
17. Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning. springer, Heidelberg 2th edition
18. Hoerl AE, Kennard R (1970) Ridge regression: biased estimation for nonorthogonal problems. Technometrics 12:55–67
19. Le HM, Le Thi HA, Nguyen MC (2015) Sparse semi-supervised support vector machines by DC programming and DCA. Neurocomputing 153:62–76
20. Le Thi HA DC programming and DCA. http://www.lita.univ-lorraine.fr/~lethi/index.php/dca.html

21. Le Thi HA, Le HM, Pham Dinh T Feature selection in machine learning: an exact penalty approach using a difference of convex functions algorithm. Mach learn. doi:10.1007/s10994-014-5455-y. Online July 2014
22. Le Thi HA, Nguyen VV, Ouchani S (2008) Gene selection for cancer classification using DCA. Adv Dat Min Appl LNCS 5139:62–72
23. Le Thi HA, Pham Dinh T (2005) The DC (difference of convex functions) programming and DCA revisited with DC models of real world non convex optimization problems. Ann Oper Res 133:23–46
24. Le Thi HA, Pham Dinh T, Le HM., Vo Xuan T (2015) DC Approximation approaches for sparse optimization. EJOR 44(1):26–46
25. Le Thi HA, Vo Xuan T, Pham Dinh T (2014) Feature selection for linear svms under uncertain data: robust optimization based on difference of convex functions algorithms. Neural Netw 59:36–50
26. Le Thi H, Nguyen B, Le HM (2013) Sparse signal recovery by difference of convex functions algorithms. In in Intelligent Information and Database Systems. Lect Notes Comput Sci 7803:387–397
27. Le Thi HA, Le HM, Pham Dinh T (2007) Fuzzy clustering based on nonconvex optimisation approaches using difference of convex (DC) functions algorithms. Journal of Advances in Data Analysis and Classification 2:1–20
28. Le Thi H, Le HM, Pham Dinh T (2014) New and efficient dca based algorithms for minimum sum-of-squares clustering. Pattern Recogn 47(1):388–401
29. Le Thi H, Le HM, Pham Dinh T, Huynh VN (2013) Block clustering based on DC programming and DCA. Neural Comput 25(10):2776–2807
30. Le Thi HA, Le Hoai M, Nguyen VV (2008) A DC programming approach for feature selection in support vector machines learning. J Adv Dat Anal Class 2:259–278
31. Neumann J, Schnörr C, Steidl G (2005) Combined svm-based feature selection and classification. Mach Learn 61:129–150
32. Ong CS, Le Thi HA Learning with sparsity by difference of convex functions algorithm. J Optimization Methods Software. doi:10.1080/10556788.2011.652630:14. Press 27 February 2012
33. Peleg D, Meir R (2008) A bilinear formulation for vector sparsity optimization. Signal Processing 8(2):375–389
34. Pham Dinh T, Le Thi HA (1997) Convex analysis approaches to dc programming: Theory, algorithms and applications. Acta Mathematica Vietnamica 22(1):287–367
35. Pham Dinh T, Le Thi HA (1998) D.c. optimization algorithms for solving the trust region subproblem. SIAM J Optim:476–505
36. Rao BD, Engan K, Cotter SF, Palmer J, Kreutz-Delgado K (2003) Subset selection in noise based on diversity measure minimization. IEEE Trans Signal Process 51(3):760–770
37. Rao BD, Kreutz-Delgado K (1999) An affine scaling methodology for best basis selection. IEEE Trans Signal Process 47:87–200
38. Rockafellar RT (1970) Convex analysis. Princeton University Press
39. Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Tamayo P, Renshaw AA, D'Amico AV, Richie JP, Lander ES, Loda M, Kantoff PW, Golub TR, Sellers WR (2002) Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1:203–209
40. Thiao M, Pham Dinh T, Le Thi HA (2008) Dc programming approach for a class of nonconvex programs involving l0 norm. In: Modelling Computation and Optimization in Information Systems and Management Sciences, Communications in Computer and Information Science CCIS, Springer, vol 14, pp 358–367
41. Tibshirani R (1996) Regression shrinkage selection via the lasso. J Roy Stat Regression Soc 46:431–439
42. Tseng P, Yun S (2009) A coordinate gradient descent method for nonsmooth separable minimization. Mathematical Programming 117(1):387–423
43. Weston J, Elisseeff A, Scholkopf B, Tipping M (2003) Use of the zero-norm with linear models and kernel methods. J Mach Learn Res 3:1439–1461
44. Yuille AL, Rangarajan A (2002) The Convex Concave Procedure (Cccp) Advances in Neural Information Processing System, vol 14. MIT Press, Cambrige MA
45. Zhang T (2009) Some sharp performance bounds for least squares regression with l1 regularization. Ann Statist 37:2109–2144
46. Zou H (2006) The adaptive lasso and its oracle properties. J Amer Stat Ass 101:1418–1429
47. Zou H, Li R (2008) One-step sparse estimates in nonconcave penalized likelihood models. Ann Statist 36(4):1509–1533
48. Le Thi HA, Nguyen MC (2014) Self-organizing maps by difference of convex functions optimization. Data Min. Knowl. Disc. 28(5-6):1336–1365
49. Le Thi HA, Nguyen MC, Pham Dinh T (2014) ADCprogramming approach for finding Communities in networks. Neural Comput. 26(12):2827–2854
50. Liu Y, Shen X, Doss H (2005) Multicategory $\psi$-learning and support vector machine: computational tools. J. Comput. Graph. Stat. 14:219–236
51. Liu Y, Shen X (2006) Multicategory $\psi$-Learning. J. Am. Stat. Assoc. 101:500–509
52. Weber S, Nagy A, Schüle T, Schnörr C, Kuba A (2006) A benchmark evaluation of large-scale optimization approaches to binary tomography. Proceedings of the Conference on Discrete Geometry on Computer Imagery (DGCI 2006), vol 4245


**Hoai An Le Thi** obtained his PhD with Distinction in Optimization in 1994, his Habilitation in 1997 from university of Rouen, France. She is currently Director of Laboratory of Theoretical & Applied Computer Science, university of Lorraine and serving as full Professor of Exceptional Class. She is adjunct professor at the Ton Duc Thang university. She is the author/co-author of more than 200 journal articles, international conference papers and book chapters, the co-editor of 8 books and 8 special issues of international journals. She has been president of Scientific Committee and president of Organizing Committee as well as member of Scientific Committee of various international conferences, and has been heading of several regional/national/international projects. Her research interests include Machine Learning, Optimization and Operations Research and their applications in Information Systems and various complex Industrial Systems. She is the co-founder (with Pham Dinh Tao) of DC programming and DCA, an innovative approach in nonconvex programming.

**Pham Dinh Tao** earned his Doctor of Sciences on Numerical Analysis and Optimization in 1981 from University Joseph Fourier-Grenoble I, France. Appointed Professor since 1989 at National Institute for Applied Sciences (INSA)-Rouen, France, he is currently Full Professor and Leader of the Team "Modelling, Optimization and Operations Research" (LMI) during the period 1989–2010. His research interests include Numerical Analysis, Optimization and Operations Research [and their applications in Data mining-Machine Learning, Bioinformatics, Image processing and Computer vision, Cryptology, Finance, Mechanics, Communication systems, Transport-Logistics, Petrochemical Engineering, Robotics and Optimal Control, Supply Chain and Management?] where he has introduced DC Programming and DCA in their preliminary form in 1985 and extensively developed, with Le Thi Hoai An, these theoretical and algorithmic tools since 1994 in order for them to become now classic and increasingly popularized by researchers and practitioners all the world over. He is the author of more than 200 papers, editor of 5 books and 7 Special Issues of International Journals, supervisor of 50 PhD theses. He has been Chair of many Scientific Committee and Chair of Organizing Committee of 6 International Conferences, Member of Scientific Committee of many International Conferences.

**Mamadou Thiao** obtained his PhD in 2011. He was a postdoctoral researcher at the Laboratory of Theoretical & Applied Computer Science, University of Lorraine. His research interests include issues related to nonconvex optimization applied on data mining and machine learning.