Contents lists available at ScienceDirect

Computers and Operations Research

Cost-sensitive Feature Selection for Support Vector Machines

S. Benítez-Peña [a,b,*], R. Blanquero [a,b], E. Carrizosa [a,b], P. Ramírez-Cobo [a,c]

[a] *IMUS, Universidad de Sevilla, Sevilla 41012, Spain*
[b] *Departamento de Estadística e Investigación Operativa, Universidad de Sevilla, Sevilla 41012, Spain*
[c] *Departamento de Estadística e Investigación Operativa, Universidad de Cádiz, Puerto Real, Cádiz 11510, Spain*

ARTICLE INFO

ABSTRACT

Feature Selection is a crucial procedure in Data Science tasks such as Classification, since it identifies the relevant variables, making thus the classification procedures more interpretable, cheaper in terms of measurement and more effective by reducing noise and data overfit. The relevance of features in a classification procedure is linked to the fact that misclassifications costs are frequently asymmetric, since false positive and false negative cases may have very different consequences. However, off-the-shelf Feature Selection procedures seldom take into account such cost-sensitivity of errors.

In this paper we propose a mathematical-optimization-based Feature Selection procedure embedded in one of the most popular classification procedures, namely, Support Vector Machines, accommodating asymmetric misclassification costs. The key idea is to replace the traditional margin maximization by minimizing the number of features selected, but imposing upper bounds on the false positive and negative rates. The problem is written as an integer linear problem plus a quadratic convex problem for Support Vector Machines with both linear and radial kernels.

The reported numerical experience demonstrates the usefulness of the proposed Feature Selection procedure. Indeed, our results on benchmark data sets show that a substantial decrease of the number of features is obtained, whilst the desired trade-off between false positive and false negative rates is achieved.

© 2018 Elsevier Ltd. All rights reserved.

1. Introduction

Supervised Classification is one of the most important tasks in Data Science, e.g. Bertsimas et al. (2016b) and Provost and Fawcett (2013), full of challenges from a Mathematical Optimization perspective, e.g. Bartlett et al. (2006), Ben-Tal et al. (2011), Boţ and Lorenz (2011), Bradley et al. (1999), Carrizosa et al. (2016, 2017a), Carrizosa and Romero-Morales (2013), Corne et al. (2012), Marron et al. (2007), Meisel and Mattfeld (2010), Panagopoulos et al. (2016), Plastria and Carrizosa (2012), Richtárik and Takáč (2016), Sánchez et al. (2016) and Shen et al. (2003). In its most basic version, we are given a set I of individuals, each represented by a vector (x_i, y_i), where $x_i \in \mathbb{R}^N$ is the so-called feature vector, and $y_i \in \mathcal{C} = \{-1, 1\}$ is the membership of individual i. A classifier Ψ, i.e., a function $\Psi : \mathbb{R}^N \longrightarrow \mathcal{C}$, is sought to assign labels $c \in \mathcal{C}$ to incoming individuals for which the feature vector x is known but the label y is unknown and estimated through $\Psi(x)$.

The different classification procedures differ in the way the classifier Ψ is obtained from the data set I. A frequent approach consists of reducing the search of the classifier to the resolution of an optimization problem, see Carrizosa and Romero-Morales (2013). This is the case, among many others, of the state-of-the-art classifier for binary classification, known as Support Vector Machines (SVM), Carrizosa and Romero-Morales (2013), Cristianini and Shawe-Taylor (2000) and Vapnik (1995, 1998), addressed in this paper.

In SVM *with linear kernel*, Ψ takes the form

$$\Psi(x) = \begin{cases} 1, & \text{if } \boldsymbol{w}^\top x + \beta \geq 0 \\ -1, & \text{else,} \end{cases} \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^N$ and $\beta \in \mathbb{R}$ are obtained as the optimal solution of the following convex quadratic programming formulation with linear constraints

$$\begin{aligned} \min_{\boldsymbol{w}, \beta, \xi} \quad & \boldsymbol{w}^\top \boldsymbol{w} + C \sum_{i \in I} \xi_i \\ s.t. \quad & y_i(\boldsymbol{w}^\top x_i + \beta) \geq 1 - \xi_i, \quad i \in I \\ & \xi_i \geq 0 \qquad\qquad\qquad\quad i \in I. \end{aligned} \tag{2}$$

* Corresponding author at: Departamento de Estadística e Investigación Operativa, Universidad de Sevilla, Sevilla, 41012, Spain

E-mail addresses: sbenitez1@us.es (S. Benítez-Peña), rblanquero@us.es (R. Blanquero), ecarrizosa@us.es (E. Carrizosa), pepa.ramirez@uca.es (P. Ramírez-Cobo).

Here $C > 0$ is the *regularization parameter*, which needs to be tuned, and $\xi_i \geq 0$ is a penalty associated to misclassifying individual i in the so-called training sample I.

An apparently innocent extension of (1) is given by

$$\Psi(x) = \begin{cases} 1, & \text{if } \mathbf{w}^\top \phi(x) + \beta \geq 0 \\ -1, & \text{else,} \end{cases} \tag{3}$$

where $\phi : \mathbb{R}^N \to \mathcal{H}$ maps the original N features into a vector space of higher dimension, and $\mathbf{w}$ and β are obtained by solving an optimization problem formally identical to (2), but taking place in the space $\mathcal{H}$ instead of $\mathbb{R}^N$,

$$\min_{\mathbf{w}, \beta, \xi} \quad \mathbf{w}^\top \mathbf{w} + C \sum_{i \in I} \xi_i$$
$$s.t. \quad y_i(\mathbf{w}^\top \phi(x_i) + \beta) \geq 1 - \xi_i, \quad i \in I \tag{4}$$
$$\xi_i \geq 0 \quad i \in I.$$

In this case, the classifier is usually obtained by solving, instead of (4), its dual,

$$\max_\alpha \quad \sum_{i \in I} \alpha_i - \frac{1}{2} \sum_{i,j \in I} \alpha_i y_i \alpha_j y_j K(x_i, x_j)$$
$$s.t. \quad \sum_{i \in I} \alpha_i y_i = 0 \tag{5}$$
$$0 \leq \alpha_i \leq \frac{C}{2}, \quad i \in I,$$

where $K(x, x') = \phi(x)^\top \phi(x')$ is the so-called *kernel function*. From the optimal solution to (5) and taking into account the complementarity slackness conditions, $\mathbf{w}$ and β in (3) are obtained. In particular,

$$\mathbf{w}^\top \mathbf{w} = \sum_{i,j \in I} \alpha_i y_i \alpha_j y_j K(x_i, x_j), \tag{6}$$

$$\mathbf{w}^\top \phi(x) = \sum_{i \in I} \alpha_i y_i K(x_i, x). \tag{7}$$

See e.g. Carrizosa and Romero-Morales (2013), Cristianini and Shawe-Taylor (2000), Vapnik (1995, 1998) for details.

The classifier uses all the features involved in the problem, both in (1) and (3), which may be rather problematic if measuring the features involve some non-negligible costs. This is particularly relevant when the dimension N of the data set is large. It is then advisable to perform Feature Selection (FS), (Aytug, 2015; Berto-lazzi et al., 2016; Bradley et al., 1998; Carrizosa et al., 2011; Fung and Mangasarian, 2004; Guyon and Elisseeff, 2003; Le Thi et al., 2015; Maldonado and Weber, 2009a; Maldonado et al., 2011a; Weston et al., 2001), in order to reduce the set of features and obtain an appropriate trade-off between classification accuracy and sparsity.

An amount of different FS procedures are found in the literature, some independent of the classification procedure (FS is performed in advance, based e.g. on the correlation between each feature and the label) and others embedded in the classification procedure, like the Holdout SVM (HOSVM), Maldonado and Weber (2009b), Kernel-Penalized SVM (KP-SVM), Maldonado et al. (2011b), or the methods presented in Chan et al. (2007) or Ghaddar and Naoum-Sawaya (2018). Also, one can minimize the number of relevant features or even their cost, as in Maldonado et al. (2017). The embedded method together with whichever of the previous optimization schemes is the approach considered in this paper, since we aim to obtain a SVM-based classifier, and, at the same time, perform the selection of the features. The core idea is the optimization problem to be solved: instead of maximizing the margin, as in the traditional SVM, we seek the classifier with lowest number of features (or cost of the features), but without damaging too much the original performance. In order to be able to control the classifier's performance, we will make use of constraints as in Benítez-Peña et al.. Specifically, the formulation

of the constrained SVM with linear kernel is

$$\min_{\mathbf{w}, \beta, \xi} \quad \mathbf{w}^\top \mathbf{w} + C \sum_{i \in I} \xi_i$$
$$s.t. \quad y_i(\mathbf{w}^\top x_i + \beta) \geq 1 - \xi_i, \quad i \in I$$
$$0 \leq \xi_i \leq M_1(1 - \zeta_i) \quad i \in I \tag{8}$$
$$\mu(\zeta)_\ell \geq \lambda_\ell \quad \ell \in L$$
$$\zeta_i \in \{0, 1\} \quad i \in I,$$

where M_1 is a large number.

In essence, this is simply the formulation for the SVM with linear kernel, to which performance constraints have been added: $\mu(\zeta)_\ell \geq \lambda_\ell$, where $\mu(\zeta)_\ell$ are different performance measures, forced to take values above thresholds λ_ℓ, and ζ_i are new binary variables that check whether sample i is counted as correctly classified. See Benítez-Peña et al. for the details. Its (partial) dual formulation is

$$\min_{\alpha, \beta, \xi, \zeta} \quad \sum_{i,j \in I} \alpha_i y_i \alpha_j y_j K(x_i, x_j) + C \sum_{i \in I} \xi_i$$
$$s.t. \quad y_i(\sum_{j \in I} \alpha_j y_j K(x_j, x_i) + \beta) \geq 1 - \xi_i, \quad i \in I$$
$$\sum_{i \in I} \alpha_i y_i = 0$$
$$0 \leq \alpha_i \leq C/2 \quad i \in I \tag{9}$$
$$0 \leq \xi_i \leq M_1(1 - \zeta_i) \quad i \in I$$
$$\mu(\zeta)_\ell \geq \lambda_\ell \quad \ell \in L$$
$$\zeta_i \in \{0, 1\} \quad i \in I.$$

As before, this is similar to the standard partial dual formulation of the SVM with general kernel and constraints in the performance measures, as in (8). For more information about how formulation (9) is obtained, the reader is referred to the Appendix. Note that, while mathematical optimization problems addressed in the statistical literature are, traditionally, as (2) or (5), nonlinear programs in continuous variables, our approach involves integer variables, which define harder optimization problems. However, Integer Programming has shown to be rather competitive thanks to the impressive advances in (nonlinear) Integer Programming solvers, as demonstrated in recent papers addressing different topics in data analysis (Bertsimas et al., 2016a; Bertsimas et al., 2014; Carrizosa et al., 2011; Carrizosa et al., 2016; Carrizosa et al., 2017a; Carrizosa et al., 2017b).

The remainder of the paper is structured as follows. In Section 2 we present the new FS methodology for SVM. For either linear or nonlinear kernels, we reduce the optimization problem to solving a standard linear integer program plus, eventually, a quadratic convex problem. The performance of our FS approach is empirically tested under different experiments described in Section 3. The results of those experiments are shown in Section 4. Comparisons between the use of linear and radial kernels, and between the standard linear SVM with and without embedded FS are also provided. The paper ends with conclusions and possible extensions in Section 5.

2. Cost-sensitive Feature Selection

In this section we present a novel linear formulation for SVM where classification costs are modeled via certain constraints, and where, in addition, a FS approach is embedded in such a way that only the relevant features are considered.

In order to cope with classification costs, first we recall some performance measures, namely,

- TPR (True Positive Rate): $P(\mathbf{w}^\top X + \beta > 0 | Y = +1)$
- TNR (True Negative Rate): $P(\mathbf{w}^\top X + \beta < 0 | Y = -1)$
- Acc (Accuracy): $P(Y(\mathbf{w}^\top X + \beta) > 0)$,

where $P(\mathbf{w}^\top X + \beta > 0 | Y = +1)$ and $P(\mathbf{w}^\top X + \beta < 0 | Y = -1)$ denote, respectively, the probability of classify correctly a positive or

negative labeled instance, and $P(Y(\mathbf{w}^\top X + \beta) > 0)$ is probability of classify correctly a given instance.

The objective is to perform classification using a reduced set of features, in such a way that certain constraints over the performance, such as $TPR \geq \lambda_1$ or $TNR \geq \lambda_{-1}$ (for threshold values λ_1, $\lambda_{-1} \in [0, 1]$), are fulfilled.

Note that the pair (X, Y) is a random vector with unknown distribution from which a sample $\{(x_i, y_i)\}_{i \in I}$ is generated. This implies that TPR and TNR should be estimated from sample data. This leads to the empirical constraints $\widehat{TPR} \geq \lambda_1^*$ and $\widehat{TNR} \geq \lambda_{-1}^*$, for $\lambda_1^* \geq \lambda_1$ and $\lambda_{-1}^* \geq \lambda_{-1}$, where the performance measures are replaced by their sample estimates. Two possible choices, which shall be explored in this work, are

$$
\begin{aligned}
\lambda_1^* &= \lambda_1 \\
&\text{and} \\
\lambda_{-1}^* &= \lambda_{-1},
\end{aligned}
\tag{10}
$$

or the more conservative approach based on Hoeffding inequality,

$$
\begin{aligned}
\lambda_1^* &= \lambda_1 + \sqrt{\frac{-\log \alpha}{2|I_+|}} \\
&\text{and} \\
\lambda_{-1}^* &= \lambda_{-1} + \sqrt{\frac{-\log \alpha}{2|I_-|}},
\end{aligned}
\tag{11}
$$

where α is the significance level for the hypothesis test whose null hypothesis is either $TPR \leq \lambda_1$ or $TNR \leq \lambda_{-1}$. See Benítez-Peña et al. for more details.

Note that it is straightforward to extend our results to the case in which measurement costs are associated with the features, as in e.g. Carrizosa et al. (2008), and then the minimum-cost feature set is sought instead.

2.1. The cost-sensitive FS procedure

Assume that we have a linear kernel, i.e., the kernel K in (5) is given by $K(x, x') = x^\top x'$, and thus the SVM with all features is obtained by solving (2). We state the Feature Selection problem as a Mixed Integer Linear Program. Consider an auxiliary variable ζ_i that in case of being equal to 1, the instance i is counted as correctly classified. Hence, estimates of TPR and TNR from sample I have lower bounds $\widehat{TPR} \geq \sum_{i \in I} \zeta_i (1 + y_i) / \sum_{i \in I} (1 + y_i)$ and $\widehat{TNR} \geq \sum_{i \in I} \zeta_i (1 - y_i) / \sum_{i \in I} (1 - y_i)$, respectively. Associated with each feature k, $1 \leq k \leq N$, we define the variable z_k taking the value 1 if feature k is selected for classifying, and 0 otherwise. Hence, the optimization problem that defines a linear classifier (hyperplane) taking into account the classification rates and in which a cost-based FS procedure is integrated is given by

$$
\begin{aligned}
\min_{\mathbf{w}, \beta, z, \zeta} \quad & \sum_{k=1}^{N} c_k z_k \\
s.t. \quad & y_i(\mathbf{w}^\top x_i + \beta) \geq 1 - M_2(1 - \zeta_i), && \forall i \in I \\
& \sum_{i \in I} \zeta_i (1 - y_i) \geq \lambda_{-1}^* \sum_{i \in I}(1 - y_i) \\
& \sum_{i \in I} \zeta_i (1 + y_i) \geq \lambda_1^* \sum_{i \in I}(1 + y_i) && (P1) \\
& |w_k| \leq M_3 z_k && \forall k \in 1, \dots, N \\
& \zeta_i \in \{0, 1\} && \forall i \in I \\
& z_k \in \{0, 1\} && \forall k \in 1, \dots, N
\end{aligned}
$$

where M_2 and M_3 are sufficiently large numbers. Also, c_k is the cost associated to the kth feature, so we perform the FS by reducing the overall cost of the features. The case $c_k = 1 \quad \forall k$, is the standard FS in which the number of features selected is minimized.

Let us discuss the rationality of the formulation $(P1)$. The overall cost associated with the features used for classifying is to be minimized in the objective. The first constraint identifies which individuals are counted as correctly classified, since, as soon as $\zeta_i = 1$,

the score $\Psi(x_i)$ is forced to be $\Psi(x_i) \geq 1$ (if $y_i = 1$) or $\Psi(x_i) \leq -1$ (if $y_i = -1$). Furthermore, the constant $\sum_{i \in I}(1 - y_i)$ is equal to two times the cardinality of the set $\{i \in I : y_i = -1\}$, whereas $\sum_{i \in I} \zeta_i (1 - y_i)$ yields two times the number of individuals counted as correctly classified in the class -1. Hence, the second and third constraints force respectively the fraction of individuals with label $y_i = -1$ (respectively, $y_i = 1$) counted as correctly classified to be at least λ_{-1}^* (respectively, at least λ_1^*). Finally, the fourth constraint forces to select those features k with $z_k = 1$. Note that, if very demanding classification rates are imposed, problem $(P1)$ may be infeasible. The solver will return this message, advising thus the user to lower the threshold values λ_1, λ_{-1}.

Solving $(P1)$ identifies the features to be used in the classification. However, an SVM classifier has not been built yet, since the margin has not been maximized. The next section shall address such problem by using the SVM either with the linear kernel or with an arbitrary one.

We should stress that the Feature Selection is based on the linear kernel, yielding the tractable linear integer optimization problem $(P1)$. Extension of our FS approach to nonlinear kernels are formally straightforward, but the resulting nonconvex mixed integer nonlinear problems are not tractable, even for low dimensions. For this reason, we perform the FS by assuming a linear kernel, and then, once the features are selected, the classifier is built using an arbitrary kernel, as detailed in Section 2.2.

Of course more flexibility is gained if, in a preprocessing step, data x are embedded in a higher dimensional space through a nonlinear mapping ϕ, and thus the original x is replaced by $\phi(x)$ in $(P1)$.

2.2. Cost-sensitive sparse SVMs: linear vs. arbitrary kernels

Here we explain how the sparse SVM is built. Let us first consider the case of the classifier with linear kernel. Hence, the sparse SVM that controls the classification rates is formulated as

$$
\begin{aligned}
\min_{w, \beta, \xi} \quad & \sum_{j=1}^{N} w_j^2 z_j + C \sum_{i \in I} \xi_i \\
s.t. \quad & y_i\left(\sum_{j=1}^{N} w_j z_j x_{ij} + \beta\right) \geq 1 - \xi_i, && \forall i \in I \\
& 0 \leq \xi_i \leq M_1(1 - \zeta_i) && \forall i \in I && (P2) \\
& \zeta_i \in \{0, 1\} && \forall i \in I \\
& \sum_{i \in I} \zeta_i (1 - y_i) \geq \lambda_{-1}^* \sum_{i \in I}(1 - y_i) \\
& \sum_{i \in I} \zeta_i (1 + y_i) \geq \lambda_1^* \sum_{i \in I}(1 + y_i).
\end{aligned}
$$

Note that $(P2)$ is defined similarly as a standard linear SVM optimization problem. The slight difference is that in $(P2)$ only the variables selected by the FS approach described in Section 2.1 are considered. This means that the values of z in $(P2)$ are those obtained in problem $(P1)$. Note too that the constraints concerning the performance measures are also added here.

Now, assume the SVM classifier has the form (3), and an arbitrary kernel function $K(x, x') = \phi(x)^\top \phi(x')$ is used instead of the linear one. See e.g. Carrizosa and Romero-Morales (2013), Cristianini and Shawe-Taylor (2000) and Vapnik (1995, 1998) for details. Although formally similar, the case of an arbitrary kernel K implies that, if an FS procedure as $(P1)$ is desired, nonlinear constraints are involved and thus the optimization problem is much harder to solve. For this reason, instead of coping with such hard problem, we propose an alternative strategy: first, $(P1)$ is solved (as before), and then the SVM classifier (with the selected kernel) is built, using only the features selected in the problem described in Section 2.1. In what follows we focus on the radial kernel, even though one can consider any arbitrary kernel K. First, we define the binary variables z identifying the features which are selected for classifying. The choice of the features, identified with the vec-

---

Algorithm 1 Pseudocode for general kernel approach.

---

1 **for** $kf = 1,\dots,folds$ **do**

2   Split data (D) into "$folds$" subsets, $D = \{D_1,\dots,D_{folds}\}$

3   Set $Validation = D_{kf}$ and set $I = D - \{D_{kf}\}$

4   **for** *each pair* $(C, gamma)$ **do**

5   **for** $kf2 = 1,\dots, folds2$ **do**

6   split $D' = D - \{D_{kf}\}$ into "$folds2$" subsets,

                $D' = \{D'_1,\dots,D'_{folds2}\}$

7   Set $Validation' = D'_{kf2}$ and set $I' = D' - \{D_{kf2}\}$

8   Run $(P1)$ over I, and select the relevant features.

9   Run $(P2)$ or $(P3)$ over I with the corresponding modified kernel.

10   Validate over $Validation'$, getting the accuracy $(acc[kf2])$

11   **end**

12   Calculate the average accuracies $(\sum_{kf2} acc[kf2])/folds2$

13   **if** $acc[kf2] \geq bestacc$ **then**

14   Set $bestacc = acc[kf2]$, $bestgamma = gamma$ and $bestC = C$

15   **end**

16   **end**

17   Run $(P1)$ over I, and select the relevant features.

18   Run $(P2)$ or $(P3)$ with the corresponding modified kernel and the parameters $bestgamma$ and $bestC$, using I.

19   Validate over $Validation$, getting the accuracy $(acc2[kf])$, and the correct classification probabilities $(TPR[kf], TNR[kf])$ as well as the number of features selected $Z[kf] = \sum_{k=1}^{N} z[k]$.

20 **end**

21 Calculate and display the average performance measures:

    $(\sum_{kf} acc2[k2])/folds$, $(\sum_{kf} TPR[kf])/folds$, $(\sum_{kf} TNR[kf])/folds$ and $(\sum_{kf} Z[kf])/folds$

---

tor z, leads to the kernel K_z, defined as

$$K_z(x, x') = exp\left(-\gamma\left(\sum_{k=1}^{N} z_k(x^{(k)} - x'^{(k)})^2\right)\right),$$

where $x^{(k)}$ denotes the kth component of vector x.

For z (and thus K_z) fixed, the aim is to solve (4), but replacing the terms $\mathbf{w}^\top\mathbf{w}$ and $\mathbf{w}^\top\phi(x_i)$, respectively, by the expressions (6) and (7), apart from adding the constraints related to the performance measurements, as described in Benítez-Peña et al.. Therefore, the cost-sensitive sparse SVM with an arbitrary kernel K is

defined (once z is fixed) as

$$\min_{\alpha,\xi,\beta,\zeta} \quad \sum_{i,j\in I} \alpha_i y_i \alpha_j y_j K_z(x_i, x_j) + C\sum_{i\in I} \xi_i$$

$$s.t. \quad y_i\left(\sum_{j\in I} \alpha_j y_j K_z(x_i, x_j) + \beta\right) \geq 1 - \xi_i, \quad \forall i \in I$$

$$0 \leq \xi_i \leq M_1(1 - \zeta_i) \quad \forall i \in I$$

$$\sum_{i\in I} \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C/2 \quad \forall i \in I \quad (P3)$$

$$\sum_{i\in I} \zeta_i(1 - y_i) \geq \lambda^*_{-1}\sum_{i\in I}(1 - y_i)$$

$$\sum_{i\in I} \zeta_i(1 + y_i) \geq \lambda^*_1\sum_{i\in I}(1 + y_i)$$

$$\zeta_i \in \{0, 1\} \quad \forall i \in I$$

Table 1
Details concerning the implementation of the CSVM for the considered datasets.

Name	$|\Omega|$	V	$|\Omega_+|$ (%)
wisconsin	569	30	357 (62.7%)
votes	435	32	267 (61.4%)
nursery	12,960	19	4320 (33.3%)
Australian	690	34	383 (55.5%)
careval	1728	15	1210 (70.023%)
leukemia	72	7728	47 (65.278%)
gastrointestinal	76	698	55 (72.368%)

Let us discuss the formulation (*P*3). The set of features is fixed through *z*. The objective function, the first, third and fourth constraints are the usual ones in SVM. The second constraint together with the fifth, sixth and seventh constraints force some samples to be correctly classified, as in (*P*1).

3. Experiment description

In this section, a description of the experiments to be carried out in Section 4 of the cost-sensitive sparse SVM with linear kernel (problem (*P*2)) are compared to those under the radial kernel (problem (*P*3)), where, as described in the previous section, the variables *z* in both (*P*2) and (*P*3) are the solutions of the FS problem formulated by (*P*1). Also, the solutions under the sparse methodology will be tested against the standard linear SVM. Although it would be natural to compare the solutions of (*P*3) with the solutions of a standard radial SVM, this comparison is not straightforward since (*P*1) may become infeasible when the performance measures obtained with the radial SVM are higher than those under the linear SVM. For simplicity we assume all measurement costs equal to 1, and then our aim is to minimize the number of features used.

Next, a description of how the experiments have been carried out is given. In order to solve problems (*P*1), (*P*2) and (*P*3), the solver Gurobi (Gurobi Optimization, Inc., 2016), and its Python language interface (Python Core Team, 2015), are used. In order to estimate the performance of these FS procedures, a 10-fold cross-validation (CV), Kohavi (1995), is used, and out of samples accuracies are reported. However, for those datasets that have less than 100 instances, a Leave-One-Out procedure is carried out, in order to have a good size in the training sample. Also, depending on whether the linear or the radial kernel is considered, a parameter *C* or a pair of parameters (*C*, γ) must be tuned. Hence, in either the first or in the second case, $C \in \{2^{-5}, 2^{-4}, \ldots, 2^4, 2^5\}$, $\gamma \in \{2^{-5}, 2^{-4}, \ldots, 2^4, 2^5\}$ are considered. Problems in integer variables are hard to solve to optimality. However, excellent solutions are obtained in reasonable time. A time limit of 300 s is set, giving the solver enough time for finding (sub)optimal solutions. Parameters M_1, M_2 and M_3 are set as 100. Moreover, parameters tuning is done by another 10-fold CV (respectively, another Leave-One-Out), and the best set of parameters selected is the one with highest accuracy in average (or, in the case of unbalanced data, with the highest geometric mean between the TPR and the TNR).

For a better understanding, the whole procedure is summarized in Algorithm 1.

4. Numerical results

Here, the experimental results are presented. We have chosen the datasets wisconsin (Breast Cancer Wisconsin (Diagnostic) Data Set), votes (Congressional Voting Records Data Set), nursery (Nursery Data Set), Australian (Statlog (Australian Credit Approval) Data Set), careval (Car Evaluation Data Set) and gastrointestinal (Gastrointestinal Lesions in Regular

Colonoscopy Data Set), all well referenced and described with detail in Lichman (2013), and leukemia (Leukemia data), described in Golub et al. (1999). First, a brief data description is given in Section 4.1. Then, results under the linear kernel approach will be presented and discussed in Section 4.2. Finally, the case of the radial kernel will be analyzed in Section 4.3.

Note that the main idea of a FS approach is to reduce the number of features or, more generally, the overall associated costs, in such a way that the performance is not severely affected. As we can control the proportion of samples well classified, this is not a problematic issue. In fact, experiments are done so that new performance measurements will not be 0.025 points lower than the originals ones, i.e., those obtained under the standard version of the SVM with linear kernel. Using the notation as in Benítez-Peña et al., *TNR* and *TPR* are the true negative and true positive rates, and TNR_0 and TPR_0 are their obtained values under the standard SVM with linear kernel on a validation sample, $TNR \geq \lambda_{-1} = \min\{1, TNR_0 - 0.025\}$ and $TPR \geq \lambda_1 = \min\{1, TPR_0 - 0.025\}$ are desired. For both linear and radial cases we have considered the two possible selection of the thresholds, defined by (10) and (11).

We stress that the purpose of this experimental section is to show how we can control TPR or TNR without a severe deterioration of overall classification rates, hopefully with a strong decrease in the number or cost of the features selected. This is the reason why we are comparing the performance of our approach with respect to the performance of the standard SVM. In a real application, the thresholds λ_1, λ_{-1} are to be given by the user, either based or not on SVM classification rates.

4.1. Data description

The performance of these novel approaches is illustrated using six real-life datasets from the UCI Repository, (Lichman, 2013), as well as the leukemia dataset (Golub et al., 1999). The positive label will be assigned to the majority class in 2-class datasets. In addition, multiclass datasets are transformed into 2-class ones, by giving positive label to the largest class and negative labels to the remaining samples. Categorical variables are transformed into dummy variables, i.e., if a categorical variable with ν levels is present, it will be replaced by $\nu - 1$ binary variables. Also, if there exist missing values, they are replaced by the median. A description of the datasets can be found in Table 1. Such table is split in 4 columns. The first shows the name of the dataset (the actual names of the datasets are presented at the beginning of this section). The total number of samples of the dataset is given in the second column. The number of variables considered, and the number (and percentage) of positive samples in the dataset, are given in the last two columns.

4.2. Results under the cost-sensitive sparse SVM with linear kernel

Two types of results will be shown, corresponding to the choices (10) and (11) of the thresholds. As a summary, it will turn out that (10) yields sparser classifiers, while (11), which is a more conservative choice, usually yields less sparse classifiers but with better accuracies.

The choice of threshold parameters in (10) leads to results summarized in Table 2. The first column of Table 2 gives the name of the dataset used (the abbreviation we have chosen for the dataset). Then, the second and third columns show, respectively, the performance measures for the standard SVM (using the linear kernel) and the proposed cost-sensitive sparse methodology. Such columns are split into two subcolumns: the first one shows the average values and the second one the standard deviations. The last column

Table 2
Performance measures under the cost-sensitive sparse SVM with linear kernel and $\lambda_1^* = \lambda_1$, $\lambda_{-1}^* = \lambda_{-1}$.

Name		SVM		FS		Feature reduction
		Mean	Std	Mean	Std	
wisconsin	Acc	0.975	0.021	0.947	0.025	30 → 2 (0 Std)
	TPR	0.992	0.013	0.973	0.031	
	TNR	0.943	0.051	0.905	0.063	
votes	Acc	0.954	0.033	0.949	0.036	32 → 2 (0 Std)
	TPR	0.955	0.038	0.928	0.059	
	TNR	0.947	0.059	0.979	0.036	
nursery	Acc	1	0	1	0	19 → 1 (0 Std)
	TPR	1	0	1	0	
	TNR	1	0	1	0	
Australian	Acc	0.848	0.051	0.855	0.057	34 → 1 (0 Std)
	TPR	0.798	0.083	0.801	0.087	
	TNR	0.912	0.05	0.926	0.041	
careval	Acc	0.956	0.017	0.946	0.019	15 → 9 (0 Std)
	TPR	0.96	0.022	0.963	0.017	
	TNR	0.948	0.024	0.907	0.04	
leukemia	Acc	0.972	0.164	0.875	0.331	7128 → 3.139 (1.205 Std)
	TPR	0.979	0.196	0.896	0.305	
	TNR	0.96	0.144	0.833	0.373	
gastrointestinal	Acc	0.895	0.307	0.829	0.379	698 → 1 (0 Std)
	TPR	0.929	0.258	0.839	0.367	
	TNR	0.8	0.4	0.8	0.4	

Table 3
Performance measures under the cost-sensitive sparse SVM with linear kernel and $\lambda_1^* = \lambda_1 + \sqrt{-\log\alpha/(2|I_1|)}$, $\lambda_{-1}^* = \lambda_{-1} + \sqrt{-\log\alpha/(2|I_{-1}|)}$.

Name		SVM		FS		Feature reduction
		Mean	Std	Mean	Std	
wisconsin	Acc	0.975	0.021	0.965	0.023	30 → 6.2 (0.919 Std)
	TPR	0.992	0.013	0.975	0.023	
	TNR	0.943	0.051	0.947	0.048	
votes	Acc	0.954	0.033	0.954	0.033	32 → 9.3 (1.16 Std)
	TPR	0.955	0.038	0.96	0.034	
	TNR	0.947	0.059	0.945	0.052	
nursery	Acc	1	0	1	0	19 → 1 (0 Std)
	TPR	1	0	1	0	
	TNR	1	0	1	0	
Australian	Acc	0.848	0.051	0.837	0.057	34 → 5.5 (1.78 Std)
	TPR	0.769	0.083	0.772	0.074	
	TNR	0.912	0.05	0.924	0.053	
careval	Acc	0.956	0.017	0.954	0.018	15 → 11 (0 Std)
	TPR	0.96	0.022	0.962	0.018	
	TNR	0.948	0.024	0.935	0.039	
leukemia	Acc	0.972	0.164	0.944	0.229	7128 → 2 (0 Std)
	TPR	0.979	0.196	0.957	0.202	
	TNR	0.96	0.144	0.92	0.272	
gastrointestinal	Acc	0.895	0.307	0.842	0.365	698 → 3.105 (0.552 Std)
	TPR	0.929	0.258	0.927	0.26	
	TNR	0.8	0.4	0.619	0.486	

reports the feature reduction, by indicating the original and selected (average) number of variables. From the table, it can be concluded that the approach with a linear kernel works well in general. In the case of wisconsin, the TPR has desirable values, since it only differentiates -0.019 points from the original. However, in the case of the accuracy and TNR, the loss is bigger than 0.025 points. This is due mainly to two reasons: first, the constraints are imposed on the training sample, while the performance is calculated using a test sample. Second, since the thresholds are considered as $\lambda_1^* = \lambda_1$, $\lambda_{-1}^* = \lambda_{-1}$, this implies we are not much restrictive as if $\lambda_1^* > \lambda_1$ ($\lambda_{-1}^* > \lambda_{-1}$) were required. Nevertheless, the new TNR value is only 0.038 points smaller than the original, and the reduction of features is significant since only two variables out of 30 are used. Also, in votes the features are significantly reduced and the most affected performance measure is the TPR, which decreases 0.027 points, making the accuracy smaller. However, the value on the TNR is increased. As happened with wisconsin,

the loss is due mainly to the two facts previously mentioned. For nursery, an amazing reduction to only one feature is achieved, in addition getting a perfect classification. This is explained as follows. As commented in Section 4.1, multiclass datasets are transformed into 2-class ones, and this is the case, obtaining the classes "not_recom" and "others", which are the positive and negative classes, respectively. In addition, one of the (categorical) features in the data (which is the one selected by our procedure) completely determines the class. In Australian, the total number of variables is also reduced to only one, having similar performance measures values as in the standard SVM. In fact, we obtain here even better results than under the original linear SVM. If the variable selected with the algorithm is studied, one can observe that it is a binary variable X, where the contingency table together with the class variable is given in Table 4. Hence this variable is by itself a good predictor, as the FS procedure pointed out. In the case of careval, we got the smallest reduction in the number

Table 4
Contingency table of the feature selected in `Australian`.

	$X = 0$	$X = 1$
Class $+$	306	77
Class $-$	23	284

of variables selected, maintaining the performance measures values above the imposed thresholds. On the other hand, in the case of `leukemia`, the number of variables is significantly reduced. However, since the number of instances is small, the performance measurements are affected by this reduction of features. Also, for `gastrointestinal`, the results are similar to what happened for `leukemia`, but the TNR has not been affected at all.

Consider next the results shown by Table 3, for the case where we are restrictive regarding the performance values, that is, when $\lambda_1^* = \lambda_1 + \sqrt{-\log\alpha/(2|I_1|)}$ and $\lambda_{-1}^* = \lambda_{-1} + \sqrt{-\log\alpha/(2|I_{-1}|)}$. From the table, it can be seen how this approach tends to work better concerning the performance measures, but achieves less sparse solutions. For example, if we focus on `wisconsin`, the TNR, the TPR and the accuracy as well, obtain the desired performance requirements. However, only a reduction of variables of one fifth is obtained. In the case of `votes`, an analogous result is obtained for the performance measures and only a reduction in one third of the variables is achieved. The same pattern as before is observed for `nursery`. For `Australian`, we obtain even an improvement in all the three performance measures considered, reducing the number of features to one fifth. In addition, we get again in `careval` the smallest reduction in the number of variables selected, maintaining the performance measures values above the thresholds imposed as before, but using a larger number of features. On the contrary, and surprisingly, we have obtained for `leukemia` even a bigger reduction in the number of features and better results using Hoeffding inequality. However, `gastrointestinal` dataset goes with the flow and the number of features is increased when using the mentioned inequality. Nevertheless, the TPR has not been affected now whereas the TNR has significantly decreased.

4.3. Results under the cost-sensitive sparse SVM with radial kernel

The analogous results to those in Section 4.2 are presented here, for the case of the radial kernel. However, only `wisconsin`, `votes` and `Australian` datasets are used here. As shown by Tables 5 and 6 and similarly as occurred in Section 4.2, the use of the threshold values obtained by the Hoeffding inequality (as in (11)) tends to yield a lower level of sparsity, but also, a higher predictive power in general (particularly, when achieving the desired bounds). Concerning the performance measures, it can be de-

duced from Tables 5 and 6 that this approach works well in general, especially when using Hoeffding. Finally, it should be noted how the reduction in the number of features is quite notable for some datasets, as before.

4.4. Comparison with other methodologies

The cost-sensitive FS procedure presented here can be compared in a certain way with some other benchmark methodologies. However, the authors are not aware of FS methods for SVM controlling, as we do, TPR or TNR. Among the different FS techniques that can be applied to SVM we can find, for example, the following ones: *Filter methods* (they are based on measures like Pearson's Correlation, Linear Discriminant Analysis or Chi-Square), *Wrapper methods* (Forward Selection, Backward Elimination, Recursive Feature elimination, ...) and *Embedded methods* (such as the presented in the introduction section).

In order to make a comparison with another state-of-the-art method, we have selected the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018). The results can be seen in Tables 7 and 8. In Table 7 we can see the results for the standard SVM, the results of our FS approach when $\lambda_1^* = \lambda_1$ and $\lambda_{-1}^* = \lambda_{-1}$, and the results of the state-of-the-art method when the maximum number of features selected is the same as the obtained for our methodology. In Table 8, the results for the standard SVM are reported together with the results of our FS approach when $\lambda_1^* = \lambda_1 + \sqrt{-\log\alpha/(2|I_1|)}$ and $\lambda_{-1}^* = \lambda_{-1} + \sqrt{-\log\alpha/(2|I_{-1}|)}$, as well as the results of the state-of-the-art method when the maximum number of features selected is the same as the obtained with our methodology.

We can observe that, except for `gastrointestinal` dataset (when using Hoeffding inequality), where we obtain better results than the comparative, similar results are obtained for our method and the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018) in terms of accuracy, while our methodology is cost-sensitive and we can control the performance measurements. As an illustration, in Table 9 we have collected all the results for dataset `australian` when applying the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018) and varying the number of features from 1 (minimum) to 34 (maximum). There, we can see how the maximum TPR obtained is 0.8007, so with our methodology, and maybe at the expense of increasing misclassification rates in the another class, we can improve the accuracy rates in the target class. The results obtained when either TPR or TNR are varied are summarized in Table 10.

From these experimental results we can conclude that, indeed, our method is able not only to reduce the number of features but it also controls the performance measures. If we observe the cases where $(\lambda_1^*, \lambda_{-1}^*)$ is (0.85,0.5) or (0.85, 0.55), we see how we have strongly increased the value in the TPR, although the TNR has decreased a lot. A similar behavior is observed for the pair (0.9,0.5).

Table 5
Performance measures under the cost-sensitive sparse SVM with radial kernel and $\lambda_1^* = \lambda_1$, $\lambda_{-1}^* = \lambda_{-1}$.

Name		SVM		FS		Feature reduction
		Mean	Std	Mean	Std	
`wisconsin`	Acc	0.975	0.021	0.956	0.012	30 → 2 (0 Std)
	TPR	0.992	0.013	0.988	0.016	
	TNR	0.943	0.051	0.893	0.051	
`votes`	Acc	0.954	0.033	0.947	0.034	32 → 2 (0 Std)
	TPR	0.955	0.038	0.928	0.059	
	TNR	0.947	0.059	0.974	0.036	
`nursery`	Acc	1	0	1	0	19 → 1 (0 Std)
	TPR	1	0	1	0	
	TNR	1	0	1	0	

Table 6

Performance measures under the cost-sensitive sparse SVM with radial kernel and $\lambda_1^* = \lambda_1 + \sqrt{-\log\alpha/(2|I_1|)}$, $\lambda_{-1}^* = \lambda_{-1} + \sqrt{-\log\alpha/(2|I_{-1}|)}$.

Name		SVM		FS		Feature reduction
		Mean	Std	Mean	Std	
wisconsin	Acc	0.975	0.021	0.947	0.03	30 → 6.2 (0.919 Std)
	TPR	0.992	0.013	0.967	0.039	
	TNR	0.943	0.051	0.907	0.02	
votes	Acc	0.954	0.033	0.949	0.03	32 → 9.3 (1.16 Std)
	TPR	0.955	0.038	0.959	0.034	
	TNR	0.947	0.059	0.939	0.043	
nursery	Acc	1	0	1	0	19 → 1 (0 Std)
	TPR	1	0	1	0	
	TNR	1	0	1	0	

Table 7

Performance measures under the cost-sensitive sparse SVM with linear kernel and $\lambda_1^* = \lambda_1$, $\lambda_{-1}^* = \lambda_{-1}$ and comparative with the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018).

Name		SVM		FS		Compar.	
		Mean	Std	Mean	Std	Mean	Std
wisconsin	Acc	0.975	0.021	0.947	0.025	0.954	0.021
	TPR	0.992	0.013	0.973	0.031	0.977	0.025
	TNR	0.943	0.051	0.905	0.063	0.911	0.056
votes	Acc	0.954	0.033	0.949	0.036	0.956	0.026
	TPR	0.955	0.038	0.928	0.059	0.949	0.039
	TNR	0.947	0.059	0.979	0.036	0.969	0.034
nursery	Acc	1	0	1	0	1	0
	TPR	1	0	1	0	1	0
	TNR	1	0	1	0	1	0
Australian	Acc	0.848	0.051	0.855	0.057	0.855	0.054
	TPR	0.798	0.083	0.801	0.087	0.801	0.082
	TNR	0.912	0.05	0.926	0.041	0.925	0.039
careval	Acc	0.956	0.017	0.946	0.019	0.949	0.016
	TPR	0.96	0.022	0.963	0.017	0.967	0.012
	TNR	0.948	0.024	0.907	0.04	0.91	0.043
leukemia	Acc	0.972	0.164	0.875	0.331	0.653	0.471
	TPR	0.979	0.196	0.896	0.305	0.66	0.474
	TNR	0.96	0.144	0.833	0.373	0.68	0.466
gastrointestinal	Acc	0.895	0.307	0.829	0.379	0.857	0.35
	TPR	0.929	0.258	0.839	0.367	0.9	0.3
	TNR	0.8	0.4	0.8	0.4	0.75	0.433

Table 8

Performance measures under the cost-sensitive sparse SVM with linear kernel and $\lambda_1^* = \lambda_1 + \sqrt{-\log\alpha/(2|I_1|)}$, $\lambda_{-1}^* = \lambda_{-1} + \sqrt{-\log\alpha/(2|I_{-1}|)}$ and comparative with the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018).

Name		SVM		FS		Compar.	
		Mean	Std	Mean	Std	Mean	Std
wisconsin	Acc	0.975	0.021	0.965	0.023	0.967	0.018
	TPR	0.992	0.013	0.975	0.023	0.989	0.017
	TNR	0.943	0.051	0.947	0.048	0.926	0.033
votes	Acc	0.954	0.033	0.954	0.033	0.954	0.033
	TPR	0.955	0.038	0.96	0.034	0.948	0.036
	TNR	0.947	0.059	0.945	0.052	0.961	0.035
nursery	Acc	1	0	1	0	1	0
	TPR	1	0	1	0	1	0
	TNR	1	0	1	0	1	0
Australian	Acc	0.848	0.051	0.837	0.057	0.851	0.053
	TPR	0.798	0.083	0.772	0.074	0.798	0.081
	TNR	0.912	0.05	0.924	0.053	0.919	0.046
careval	Acc	0.956	0.017	0.954	0.018	0.954	0.017
	TPR	0.96	0.022	0.962	0.018	0.97	0.016
	TNR	0.948	0.024	0.935	0.039	0.917	0.027
leukemia	Acc	0.972	0.164	0.944	0.229	0.932	0.252
	TPR	0.979	0.196	0.957	0.202	0.938	0.242
	TNR	0.96	0.144	0.92	0.272	0.917	0.276
gastrointestinal	Acc	0.895	0.307	0.842	0.365	0.714	0.452
	TPR	0.929	0.258	0.927	0.26	0.75	0.433
	TNR	0.8	0.4	0.619	0.486	0.625	0.484

Table 9

Performance measures using the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018), varying the maximum number of features from 1 (minimum) to 34 (maximum) in `Australian` dataset.

Australian							
# Feat.	Acc	TPR	TNR	# Feat.	Acc	TPR	TNR
1	0.8551	0.8007	0.9248	18	0.8464	0.7954	0.9121
2	0.8551	0.8007	0.9248	19	0.8464	0.7954	0.9121
3	0.8551	0.8007	0.9248	20	0.8464	0.7954	0.9121
4	0.8551	0.8007	0.9248	21	0.8464	0.7954	0.9121
5	0.8507	0.798	0.9186	22	0.8578	0.7953	0.9154
6	0.8507	0.798	0.9186	23	0.8464	0.7954	0.9121
7	0.8507	0.798	0.9186	24	0.8464	0.7954	0.9121
8	0.8507	0.798	0.9186	25	0.8464	0.7954	0.9121
9	0.8507	0.798	0.9186	26	0.8449	0.7929	0.9121
10	0.8507	0.798	0.9186	27	0.8478	0.7981	0.9121
11	0.8507	0.798	0.9186	28	0.8478	0.7954	0.9153
12	0.8478	0.798	0.9121	30	0.8493	0.798	0.9153
13	0.8478	0.798	0.9121	31	0.8478	0.7954	0.9153
14	0.8478	0.798	0.9121	32	0.8478	0.7981	0.9121
15	0.8478	0.798	0.9121	33	0.8478	0.7981	0.9121
16	0.8464	0.7954	0.9121	34	0.8478	0.7981	0.9121
17	0.8478	0.798	0.9121				

Table 10

Performance measures using the method in Chan et al. (2007), Ghaddar and Naoum-Sawaya (2018), varying the maximum number of features from 1 (minimum) to 34 (maximum) in `Australian` dataset.

Australian					
λ_1^*	λ_{-1}^*	Acc	TPR	TNR	Aver. # Feat. selected
0.85	0.5	0.738	0.94	0.484	1
0.85	0.55	0.738	0.94	0.484	1
0.85	0.575	0.812	0.854	0.754	1.7
0.85	0.6	0.855	0.801	0.92	2
0.9	0.5	0.757	0.896	0.582	1.333

However, when using (0.85,0.575) a different trade-off is found, whereas for (0.85,0.6) we recover the original results.

5. Concluding remarks

In this paper we have proposed a Feature Selection procedure for Support Vector Machines that yields a novel, sparse, SVM. Contrary to existing Feature Selection approaches, we take explicitly into account that misclassification costs may be rather different in the two groups, and thus, instead of seeking the classifier maximizing the margin, we seek the most sparse classifier that attains certain true positive and true negative rates on the dataset. For both SVM with linear and radial kernel, the problem is written in a straightforward manner, solving first a mixed integer linear problem and then their standard SVM formulations, considering only the features obtained in the first problem as well as the performance constraints. The reported numerical results show that the novel approaches lead to comparable or better performance rates, in addition to an important reduction in the number of variables.

Several extensions of the approach presented in this paper are possible. In our opinion, they deserve further study. First, several classification and regression procedures based on optimization problems, such as Support Vector Regression, logistic regression or distance-weighted discrimination, are amenable to address, as done here, an integrated FS and classification or regression. The optimization problems obtained in this way have a structure which should be exploited to make the approach competitive and including cost-sensitivity in the FS procedure. Second, even within SVM, it should be observed that SVM is a tool for binary classification. For multiclass datasets, SVM classification is performed by solving a series of SVM problems, see Cristianini and Shawe-Taylor (2000),

Wang and Shen (2007). When some classes are hard to identify, the basic multiclass strategies may yield discouraging results. Performing simultaneously Feature Selection and class fusion, as in Guo (2010), is an interesting nontrivial extension of our approach. To do this, problems (P1), (P2) and (P3) need to be conveniently modified.

Appendix A

In this section we describe step by step how formulation (9) is built from Eq. (8). Hence, let us suppose first that we have the model

$$\min_{\boldsymbol{w},\beta,\xi} \quad \boldsymbol{w}^\top \boldsymbol{w} + C\sum_{i\in I}\xi_i$$
$$s.t. \quad y_i(\boldsymbol{w}^\top x_i + \beta) \geq 1 - \xi_i, \quad i \in I$$
$$0 \leq \xi_i \leq M_1(1 - \zeta_i) \quad i \in I$$
$$\mu(\zeta)_\ell \geq \lambda_\ell \quad \ell \in L$$
$$\zeta_i \in \{0,1\} \quad i \in I.$$

This one can be rewritten as

$$\min_\zeta \qquad \min_{\boldsymbol{w},\beta,\xi} \quad \boldsymbol{w}^\top \boldsymbol{w} + C\sum_{i\in I}\xi_i$$
$$s.t. \quad \zeta_i \in \{0,1\} \quad i \in I \quad s.t. \quad y_i(\boldsymbol{w}^\top x_i + \beta) \geq 1 - \xi_i, \quad i \in I$$
$$\mu(\zeta)_\ell \geq \lambda_\ell \quad \ell \in L \qquad 0 \leq \xi_i \leq M_1(1 - \zeta_i) \quad i \in I$$

If we assume that the binary variables ζ fixed, the Karush–Kuhn–Tucker (KKT) conditions for the inner problem are

$$w = \sum_{i\in I}\alpha_i y_i x_i$$
$$0 = \sum_{i\in I}\alpha_i y_i$$
$$0 \leq \alpha_i \leq C/2 \quad i \in I.$$

Substituting these expressions into the last optimization problem, the partial dual of such problem can be calculated, obtaining

$$\min_\zeta \qquad \min_{\alpha,\beta,\xi} \quad \left(\sum_{i\in I}\alpha_i y_i x_i\right)^\top \left(\sum_{i\in I}\alpha_i y_i x_i\right) + C\sum_{i\in I}\xi_i$$
$$s.t. \quad z_j \in \{0,1\} \quad j \in J \quad s.t. \quad y_i\left(\left(\sum_{i\in I}\alpha_i y_i x_i\right)^\top x_i + \beta\right)$$
$$\geq 1 - \xi_i \quad i \in I$$
$$\mu(\zeta)_\ell \geq \lambda_\ell \quad \ell \in L \qquad 0 \leq \xi_i \leq M_1(1 - \zeta_i) \quad i \in I$$
$$\sum_{i\in I}\alpha_i y_i = 0$$
$$0 \leq \alpha_i \leq C/2 \quad i \in I$$

As a last step, the kernel trick is used and the final formulation (9) is obtained.

References

Aytug, H., 2015. Feature selection for support vector machines using generalized benders decomposition. Eur. J. Oper. Res. 244 (1), 210–218.

Bartlett, P.L., Jordan, M.I., McAuliffe, J.D., 2006. Convexity, classification, and risk bounds. J. Am. Stat. Assoc. 101 (473), 138–156.

Ben-Tal, A., Bhadra, S., Bhattacharyya, C., Saketha Nath, J., 2011. Chance constrained uncertain classification via robust optimization. Math. Program. 127 (1), 145–173.

Benítez-Peña, S., Blanquero, R., Carrizosa, E., Ramírez-Cobo, P., 2017. On Support Vector Machines Under a Multiple-Cost Scenario. Working Paper, https://www.researchgate.net/publication/321765946_On_Support_Vector_Machines_under_a_multiple-cost_scenario.

Bertolazzi, P., Felici, G., Festa, P., Fiscon, G., Weitschek, E., 2016. Integer programming models for feature selection: new extensions and a randomized solution algorithm. Eur. J. Oper. Res. 250 (2), 389–399.

Bertsimas, D., King, A., Mazumder, R., et al., 2016a. Best subset selection via a modern optimization lens. Ann. Stat. 44 (2), 813–852.

Bertsimas, D., Mazumder, R., et al., 2014. Least quantile regression via modern optimization. Ann. Stat. 42 (6), 2494–2525.

Bertsimas, D., O'Hair, A.K., Pulleyblank, W.R., 2016b. The Analytics Edge. Dynamic Ideas, Massachusetts.

Boţ, R.I., Lorenz, N., 2011. Optimization problems in statistical learning: duality and optimality conditions. Eur. J. Oper. Res. 213 (2), 395–404.

Bradley, P.S., Fayyad, U.M., Mangasarian, O.L., 1999. Mathematical programming for data mining: formulations and challenges. INFORMS J. Comput. 11 (3), 217–238.

Bradley, P.S., Mangasarian, O.L., Street, W.N., 1998. Feature selection via mathematical programming. INFORMS J. Comput. 10 (2), 209–217.

Carrizosa, E., Martín-Barragán, B., Romero-Morales, D., 2008. Multi-group support vector machines with measurement costs: a biobjective approach. Discrete Appl. Math. 156 (6), 950–966.

Carrizosa, E., Martín-Barragán, B., Romero-Morales, D., 2011. Detecting relevant variables and interactions in supervised classification. Eur. J. Oper. Res. 213 (1), 260–269.

Carrizosa, E., Nogales-Gómez, A., Romero-Morales, D., 2016. Strongly agree or strongly disagree?: rating features in support vector machines. Inf. Sci. 329, 256–273.

Carrizosa, E., Nogales-Gómez, A., Romero-Morales, D., 2017a. Clustering categories in support vector machines. Omega 66, 28–37.

Carrizosa, E., Olivares-Nadal, A.V., Ramírez-Cobo, P., 2017b. A sparsity-controlled vector autoregressive model. Biostatistics 18 (2), 244–259. doi:10.1093/biostatistics/kxw042.

Carrizosa, E., Romero-Morales, D., 2013. Supervised classification and mathematical optimization. Comput. Oper. Res. 40 (1), 150–165.

Chan, A.B., Vasconcelos, N., Lanckriet, G.R.G., 2007. Direct convex relaxations of sparse SVM. In: Proceedings of the Twenty-Fourth International Conference on Machine Learning. ACM, New York, NY, USA, pp. 145–153. doi:10.1145/1273496.1273515.

Corne, D., Dhaenens, C., Jourdan, L., 2012. Synergies between operations research and data mining: the emerging use of multi-objective approaches. Eur. J. Oper. Res. 221 (3), 469–479.

Cristianini, N., Shawe-Taylor, J., 2000. An Introduction to Support Vector Machines and other Kernel-based Learning Methods. Cambridge University Press.

Fung, G.M., Mangasarian, O.L., 2004. A feature selection Newton method for support vector machine classification. Comput. Optim. Appl. 28 (2), 185–202. doi:10.1023/B:COAP.0000026884.66338.df.

Ghaddar, B., Naoum-Sawaya, J., 2018. High dimensional data classification and feature selection using support vector machines. Eur. J. Oper. Res. 265 (3), 993–1004. https://doi.org/10.1016/j.ejor.2017.08.040.

Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S., 1999. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286 (5439), 531–537. doi:10.1126/science.286.5439.531. http://science.sciencemag.org/content/286/5439/531.full.pdf.

Guo, J., 2010. Simultaneous variable selection and class fusion for high-dimensional linear discriminant analysis. Biostatistics 11 (4), 599. doi:10.1093/biostatistics/kxq023.

Gurobi OptimizationInc., 2016. Gurobi Optimizer Reference Manual. Gurobi Optimization, Inc..

Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. J. Mach. Learn. Res. 3 (Mar), 1157–1182.

Kohavi, R., 1995. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the 1995 International Joint Conference on Artificial Intelligence (IJCAI), 14. Stanford, CA, pp. 1137–1143.

Le Thi, H.A., Le, H.M., Pham Dinh, T., 2015. Feature selection in machine learning: an exact penalty approach using a difference of convex function algorithm. Mach. Learn. 101 (1), 163–186.

Lichman, M., 2013. UCI Machine Learning Repository, institution: University of California, Irvine, School of Information and Computer Sciences. http://archive.ics.uci.edu/ml

Maldonado, S., Pérez, J., Bravo, C., 2017. Cost-based feature selection for support vector machines: an application in credit scoring. Eur. J. Oper. Res. 261 (2), 656–665. https://doi.org/10.1016/j.ejor.2017.02.037.

Maldonado, S., Weber, R., 2009. A wrapper method for feature selection using support vector machines. Inf. Sci. 179 (13), 2208–2217.

Maldonado, S., Weber, R., 2009. A wrapper method for feature selection using support vector machines. Inf. Sci. 179 (13), 2208–2217. Special Section on High Order Fuzzy Sets

Maldonado, S., Weber, R., Basak, J., 2011. Simultaneous feature selection and classification using kernel-penalized support vector machines. Inf. Sci. 181 (1), 115–128.

Maldonado, S., Weber, R., Basak, J., 2011. Simultaneous feature selection and classification using kernel-penalized support vector machines. Inf. Sci. 181 (1), 115–128. https://doi.org/10.1016/j.ins.2010.08.047.

Marron, J.S., Todd, M.J., Ahn, J., 2007. Distance-weighted discrimination. J. Am. Stat. Assoc. 102 (480), 1267–1271.

Meisel, S., Mattfeld, D., 2010. Synergies of operations research and data mining. Eur. J. Oper. Res. 206 (1), 1–10.

Panagopoulos, O.P., Pappu, V., Xanthopoulos, P., Pardalos, P.M., 2016. Constrained subspace classifier for high dimensional datasets. Omega 59, 40–46.

Plastria, F., Carrizosa, E., 2012. Minmax-distance approximation and separation problems: geometrical properties. Math. Program. 132 (1), 153–177.

Provost, F., Fawcett, T., 2013. Data Science for Business: What You Need to Know About Data Mining and Data-Analytic thinking, first O'Reilly Media, Inc..

Python Core Team, 2015. Python: A Dynamic, Open Source Programming Language. Python Software Foundation..

Richtárik, P., Takáč, M., 2016. Parallel coordinate descent methods for big data optimization. Math. Program. 156 (1), 433–484.

Sánchez, B.N., Wu, M., Song, P.X.K., Wang, W., 2016. Study design in high-dimensional classification analysis. Biostatistics 17 (4), 722. doi:10.1093/biostatistics/kxw018.

Shen, X., Tseng, G.C., Zhang, X., Wong, W.H., 2003. On ψ-learning. J. Am. Stat. Assoc. 98 (463), 724–734.

Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer-Verlag, New York, NY, USA.

Vapnik, V., 1998. Statistical Learning Theory, 1. Wiley, New York.

Wang, L., Shen, X., 2007. On $L1$-norm multiclass support vector machines. J. Am. Stat. Assoc. 102 (478), 583–594. doi:10.1198/016214506000001383.

Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., Vapnik, V., 2001. Feature selection for SVMs. In: Leen, T.K., Dietterich, T.G., Tresp, V. (Eds.), Advances in Neural Information Processing Systems, 13. MIT Press, pp. 668–674.