Support Vector Machines with Constraints for Sparsity in the Primal Parameters

Vanessa Gómez-Verdejo, Manel Martínez-Ramón, *Senior Member, IEEE*, Jerónimo Arenas-García, *Member, IEEE*, Miguel Lázaro-Gredilla, *Member, IEEE*, and Harold Molina-Bulla, *Member, IEEE*

***Abstract*— This paper introduces a new support vector machine (SVM) formulation to obtain sparse solutions in the primal SVM parameters, providing a new method for feature selection based on SVMs. This new approach includes additional constraints to the classical ones that drop the weights associated to those features that are likely to be irrelevant. A ν-SVM formulation has been used, where ν indicates the fraction of features to be considered. This paper presents two versions of the proposed sparse classifier, a 2-norm SVM and a 1-norm SVM, the latter having a reduced computational burden with respect to the first one. Additionally, an explanation is provided about how the presented approach can be readily extended to multiclass classification or to problems where groups of features, rather than isolated features, need to be selected. The algorithms have been tested in a variety of synthetic and real data sets and they have been compared against other state of the art SVM-based linear feature selection methods, such as 1-norm SVM and doubly regularized SVM. The results show the good feature selection ability of the approaches.**

***Index Terms*— Feature group selection, feature selection, margin maximization, multiclass classification, support vector machines.**

I. Introduction

SUPPORT vector machines (SVMs) [1], [2] are considered the state-of-art in machine learning due to their well known good performance in a wide range of applications [3]–[5]. The SVM criterion minimizes a loss term, called hinge loss, plus an additional quadratic penalization term which regularizes the solution [6]. This hinge loss minimization allows SVMs to approximate Bayes' rule without estimating the conditional class probability [7] and makes it converge to a maximum margin solution [8], thus endowing SVMs with good generalization properties.

In spite of the generally good performance of SVMs, in many practical situations, useless, redundant, or noisy features can degrade the attained solution. The reason for this is that the SVM solution is based on a combination of all input features, including the irrelevant ones. As it is stated in the bet-on-sparsity principle [9], this situation is undesired and it would be preferable to obtain a solution consisting only of the relevant features. That way, more accurate and interpretable solutions can be achieved.

To achieve this goal, a feature selection process [10], [11] is usually applied. Classical feature selection techniques, such as filtering [12] or wrapping [13], [14] approaches, are used as an independent preprocessing step before the training of the final classification (or regression) machine. More recent feature selection methods combine the feature selection process with the final predictor training. For instance, in [15]–[17] an objective function that combines an accuracy prediction term with a term associated to the sparsity in the number of selected variables is employed. In [18]–[20] the SVM prediction output is considered as a linear combination of kernel functions and then, the prediction accuracy is evaluated as a function of the used and discarded features. This method, known as recursive feature elimination (RFE), has been widely employed for SVM classification, however, recent works [21] have shown that RFE is not consistent with maximum margin solutions.

In contrast to the approaches that include an explicit feature selection strategy (either independent or combined with the classification step), classifiers directly providing sparse solutions are usually preferred. Following this point of view, the LASSO method was proposed in [15]. LASSO includes a 1-norm regularization term in the optimization problem. Since this norm has a singularity at the origin, some coefficients of the solution vector are shrunk to zero, what provides sparse solutions. Since then, many researchers have focused their work on minimizing 1-norm penalized functions [22]–[24]. In fact [25] points out the need and usefulness of linear sparse solutions in problems like functional magnetic resonance imaging.

In [26], the classical SVM formulation is modified by replacing the quadratic penalization term with a 1-norm penalty, what leads to solutions with sparse coefficients. Although this SVM formulation can only be used for feature selection in linear classification problems, this approach has nevertheless been successfully used in a large number of applications, such as computational biology [27], [28], drug-design [17] or gene microarrays classification [29], among others.

Although 1-norm SVMs retain most of the desired properties of classical SVMs, such as margin maximization, they may fail to provide good solutions in certain situations. As it is

EQ:1

Manuscript received June 22, 2010; revised April 11, 2011; accepted XXXX XX, XXXX. This work was supported in part by the Ministry of Science and Innovation (Spanish Goverment), under Grant TEC2008-02473.

V. Gómez-Verdejo, M. Martínez-Ramón, J. Arenas-García, and H. Molina-Bulla are with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Madrid 28911, Spain (e-mail: vanessa@tsc.uc3m.es; manel@tsc.uc3m.es; jarenas@tsc.uc3m.es; hmolina@tsc.uc3m.es).

M. Lázaro-Gredilla is with the Department of Communication Engineering, Universidad de Cantabria, Santander 39005, Spain (e-mail: miguellg@gtas.dicom.unican.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2011.2148727

illustrated in [9], when most of the input features are relevant for the classification task at hand, classical 2-norm SVMs usually outperform their 1-norm counterparts. Furthermore, as it is pointed out in [30] and [31], the 1-norm SVM presents two additional limitations: first, when there are highly correlated variables, it usually removes some of them, and, second, the maximum number of selected features is limited by the number of available training data. Trying to overcome these drawbacks, elastic nets [32] and their particularization to SVMs by means of the doubly regularized support vector machine (Dr-SVM) [30], [31] are proposed, this new approach generalizes the LASSO and 1-norm SVM methods by keeping the 2-norm regularization term and including an additional 1-norm penalty term to force sparsity. Despite common improved performance of Dr-SVM, both 1-norm and Dr-SVMs are not suitable methods when the underlying model is truly sparse, since they are not able to remove all unnecessary variables from the final classifier, this problem was already remarked for 1-norm SVMs in [33] and, in the experimental section of this paper, we will illustrate it for Dr-SVM.

An additional limitation of 1-norm SVM and Dr-SVM, is that they are not well suited to multiclass classification or to problems where features have to be selected or removed using predefined groups. One possible solution could consist in adding a group LASSO [34] or an ∞-norm [35] penalization term into the SVM formulation. However, both options result in a more complex SVM formulation, which cannot be solved with standard linear programming (LP) or quadratic programming (QP) solvers.

In this paper, a new SVM formulation for the linear case is presented that directly forces sparse solutions. Rather than modifying the objective function, additional constraints are included in the minimization task in order to identify irrelevant features and to drop their associated weights to values lower than a small parameter ε. This constant can be adjusted during the optimization problem resolution by predefining the number of relevant features to be kept in the final solution using a ν-SVM formulation [36]. We will show that these additional constraints can be incorporated to force sparsity in both 2-norm and 1-norm SVM formulations. Our approach allows to overcome the limitations of 1-norm SVMs and Dr-SVMs in different ways. First, by properly adjusting parameter ν, the algorithm is able to remove all irrelevant features from the final model. Second, the proposed formulation can be applied to the selection of isolated features or predefined feature groups where needed. Finally, as it will be shown in the experiments section, more accurate solutions are usually achieved, particularly, when using the new constraints together with the 2-norm SVM.

The rest of this paper is organized as follows. In the next section, we introduce our approach to force feature selection in SVM classifiers, explaining how it can be applied both to 2-norm and 1-norm formulations. Section III presents some extensions of the method to address the selection of features in predefined groups of variables, as well as for multiclass classification problems. Section IV presents extensive simulation work to illustrate the performance of our approach, and its advantages with respect to previous proposals for feature selection in SVMs. Finally, Section V presents the main conclusion of our work, and identifies some lines for future research.

II. SVM WITH EXPLICIT CONSTRAINTS FOR FEATURE SELECTION

A. Problem Overview

In this paper, we consider classification problems where the representation of the input data contains some features, which are irrelevant for the task at hand. This may happen as a consequence of redundancy between the input variables or, simply, because some of the input features do not carry any valuable information for the classification. In a standard machine learning setup, we are given a set of N training labeled data, $\mathcal{S} = \{\mathbf{x}^{(l)}, y^{(l)}\}$, $l = 1, \ldots, N$, where $\mathbf{x}^{(l)} \in \Re^d$ are the input vectors and $y^{(l)}$ are used to encode class membership, from which we have to learn both the subset of relevant input variables and the classification function itself.

Linear classifiers obtain their outputs according to a thresholded version of the estimator

$$\hat{y} = \mathbf{w}^T\mathbf{x} + b \tag{1}$$

where $\hat{y}$ is the output of the classifier for input vector $\mathbf{x}$, $\mathbf{w}$ is the vector that defines the classifier, and b is a bias term. For the SVM case, the Representer's Theorem [1], [2] states that the solution vector will lie in the subspace spanned by all training vectors $\{\mathbf{x}^{(l)}\}$. When irrelevant features are present in the data we can carry out a pre-processing stage to select the most informative variables or, alternatively, discard the variables x_i whose associated weight w_i is exactly zero after the optimization of the classifier. However, since noise is normally present in the data, none of the components of $\mathbf{w}$ will be exactly zero unless sparsity is included as an optimization criterion during the training of the classifier.

A standard way to impose sparsity in $\mathbf{w}$ is to include a regularization term in the cost function, based on the 1-norm of $\mathbf{w}$, i.e., $\|\mathbf{w}\|_1 = \sum_i |w_i|$. This regularizer presents singularity points whenever any of the components of $\mathbf{w}$ is zero, what tends to nullify some of the solution weights, thus favoring sparse solutions. However, this mechanism does not necessarily imply that all weight components associated to irrelevant variables will become zero [33].

Rather than modifying the structural risk term in the SVM functional, in this paper, we propose a new approach to impose sparsity in the solution by introducing a set of additional constraints for the optimization problem. We will see that our method is able to automatically identify all irrelevant features, thus constituting an effective mechanism for implementing SVMs that incorporate a feature selection approach. Furthermore, since the 2-norm regularization term can still be used, this usually results in a better performance when the true underlying solution is non sparse.

B. 2-Norm SVMs with Sparsity Constraints

Classical SVMs are based on the minimization of a functional that includes two terms. The first term is the squared norm of the weight vector $\mathbf{w}$, which is inversely proportional to the margin of classification [1], thus, this term is related to the structural risk of the classifier and to its generalization capabilities. The second term in the objective functional, which is known as the empirical risk term, is a sum of errors over the training data. In other words, the linear SVM problem can be stated as

$$\begin{aligned} \min \quad & \|\mathbf{w}\|^2 + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} \\ \text{s.t.} \quad & y^{(l)}\left(\mathbf{w}^T\mathbf{x}^{(l)} + b\right) \geq 1 - \xi^{(l)}; \qquad \forall l \\ & \xi^{(l)} \geq 0; \qquad \forall l \end{aligned} \tag{2}$$

where slack variables $\xi^{(l)}$ are introduced to allow some of the training patterns to be misclassified or to lie inside the classifier margin, and where C is a constant that controls the trade-off between the structural and empirical risk terms.

As it is well known, this optimization method provides a sparse solution in the sense that $\mathbf{w}$ is a linear combination of only a subset of the training data [the so-called support vectors (SVs)]. However, if feature selection is pursued during the optimization, a solution sparse in the parameters $\mathbf{w}$ is needed. In order to obtain such a solution, we will introduce some additional constraints in the optimization problem.

We start by rewriting each of the weight components, w_i, $i = 1, \ldots, d$, as $w_i = u_i - v_i$, with $u_i, v_i \geq 0$. As we will explain later, our optimization problem will implicitly enforce that at least one of the two terms in the subtraction, u_i or v_i, is zero, depending on whether the optimal weight is positive ($u_i > 0$ and $v_i = 0$), negative ($u_i = 0$ and $v_i > 0$) or zero ($u_i = v_i = 0$). Therefore, the square norm of the weight vector is given, in terms of these new variables, by

$$\|\mathbf{w}\|_2^2 = \sum_{i=1}^{d} u_i^2 + v_i^2. \tag{3}$$

Furthermore, in order to obtain a sparse solution in $\mathbf{w}$, we introduce some additional constraints to upper bound the absolute value of weight components by a small constant ε, i.e., $|w_i| = u_i + v_i < \varepsilon$. Introducing (3) and the new constraints into (2), we get the following modified SVM formulation:

$$\begin{aligned} \min \quad & \sum_{i=1}^{d}(u_i^2 + v_i^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + \frac{C'}{d}\sum_{i=1}^{d}\gamma_i \\ \text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \qquad \forall l \\ & \xi^{(l)} \geq 0; \qquad \forall l \\ & u_i + v_i \leq \varepsilon + \gamma_i; \qquad \forall i \\ & u_i, v_i \geq 0; \qquad \forall i \\ & \gamma_i \geq 0; \qquad \forall i. \end{aligned} \tag{4}$$

Although the above optimization problem has not explicitly included, the constraint $u_i v_i = 0$, (4) is indirectly forcing that either u_i or v_i is equal to 0. Note that among all possible pairs of values (u_i, v_i) that are able to provide a certain value w_i, the pair which minimizes $\sum_{i=1}^{d}(u_i^2 + v_i^2)$ has to fix either u_i or v_i to 0, for instance, for positive w_i and according to its definition in terms of u_i and v_i, minimization of the functional in (4) will lead to $v_i = 0$ and $u_i = w_i$. The opposite situation will occur for $w_i < 0$.

Note that in our redefinition of the problem we have introduced new slack variables γ_i and those slack variables associated with relevant features will be greater than zero after the functional optimization. Thus, these constants need to be introduced in the objective functional weighted with a trade-off parameter C'. The above minimization problem can be directly solved in the primal over the variables u_i, v_i, b, γ_i, and $\xi^{(l)}$, using standard QP algorithm.

We can now get some insight into the sparsity mechanism that has been adopted. If irrelevant features are present in the input representation space, most classification schemes would still assign them a non zero weight w_i due to the noise present in the data. However, if a w_i value greater than ε were assigned in our scheme, γ_i would be strictly positive, increasing the value of the functional. Thus, on the one hand irrelevant features that do not significantly decrease the empirical error term will simply be assigned weights smaller, in absolute terms, than ε. On the other hand, components w_i which are necessary to define the SVM solution will have values larger than ε. It is straightforward to use the values of slacks γ_i after the optimization to check whether a variable has been removed or incorporated into the classification model.

This new SVM with sparsity constraints performs feature selection on the input variables, so we will hereafter refer to it as sparse primal support vector machine (SP-SVM).

At first sight, one could think that the sparsity constraints in (4) are equivalent to a 1-norm penalty term and thus algorithm (4) is equivalent to Dr-SVM. Nevertheless, these constraints have been introduced here through an ε-insensitive cost function. As we will analyze along this paper, this new formulation provides two advantages: 1) the sparsity of the model can be easily adjusted by the user through a ν SVM formulation, and 2) extensions of this model to group feature selection and multiclass problems are straightforwardly derived.

The computational cost of (4) is larger than that of 1-norm or Dr-SVMs due to the new constrains. However, an efficient implementation of the problem, which exploits the sparse formulation of these constrains, it results in a very moderate computational increase.

Finally, it is important to point out that a major limitation of problem (4), as well as 1-norm and Dr-SVM algorithms, is their linear formulation. Note that their non linear extension would provide a non linear boundary with a kernel selection mechanism, instead of an automatic feature selection criterion.

C. 2-Norm ν-SP-SVM

In this section, we introduce a modification of the SP-SVM formulation in (4) to automatically adjust the value of ε, following the ν-SVM that was introduced in [36]. In this formulation of the SVM, ε is traded off against model complexity and slack variables through a constant $\nu \in (0, 1]$. Then, the optimization problem to solve is given by

$$\min \quad \sum_{i=1}^{d}(u_i^2 + v_i^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + C'\left[\nu\varepsilon + \frac{1}{d}\sum_{i=1}^{d}\gamma_i\right]$$

$$
\begin{aligned}
\text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \quad \forall l \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& u_i + v_i \leq \varepsilon + \gamma_i; \quad \forall i \\
& u_i, v_i \geq 0; \quad \forall i \\
& \gamma_i \geq 0; \quad \forall i \\
& \varepsilon \geq 0.
\end{aligned} \tag{5}
$$

As above, this optimization problem can be directly solved in the primal, with respect to variables u_i, v_i, b, γ_i, $\xi^{(l)}$, and ε.

It is well known [36] that, when the standard ν support vector regression is applied resulting a non zero ε, ν is an upper bound on the fraction of errors and a lower bound on the fraction of SVs. Note that in (5), if the dual formulation of the problem was used and we let $\{\beta_i\}_{i=1}^{d}$ be the dual variables associated to the sparsity constraints, the following equalities had to be verified:

$$
\sum_{i=1}^{d} \beta_i \leq \frac{C'}{d}\nu
$$

$$
0 \leq \beta_i \leq \frac{C'}{d}
$$

what forces ν to be an upper bound of the number of dual variables β_i taking a value of C'/d, that is, ν is an upper bound over the number of slack variables γ_i different from 0. This leads to a useful result for the proposed ν-SP-SVM: ν is an upper bound on the fraction of components of $\mathbf{w}$ whose absolute value is less than ε. In other words, parameter ν can be used to control the sparsity of the solution, setting *a priori* the maximum number of features that can be selected by the 2-norm ν-SP-SVM.

D. 1-Norm ν-SP-SVM

Using the 1-norm of $\mathbf{w}$ in the structural risk term of classical SVMs leads to LP problems, which have a reduced computational burden when compared to the QP formulation required for 2-norm SVMs. Similar benefits can be obtained for the SP-SVM proposed in the previous sections. Note that the constraints that were imposed in order to force sparsity do not affect the regularizer for $\mathbf{w}$ in any way, thus, in order to extend either (4) or (5) to the 1-norm case, it is sufficient to replace the structural risk term accordingly. For instance, for the ν-SP-SVM in its 1-norm version this leads to

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{d}(u_i + v_i) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + C'\left[\nu\varepsilon + \frac{1}{d}\sum_{i=1}^{d}\gamma_i\right] \\
\text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \quad \forall l \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& u_i + v_i \leq \varepsilon + \gamma_i; \quad \forall i \\
& u_i, v_i \geq 0; \quad \forall i \\
& \gamma_i \geq 0; \quad \forall i \\
& \varepsilon \geq 0.
\end{aligned} \tag{6}
$$

Using LP optimization tools, this problem can be solved in a more efficient way than with QP optimizers, obtaining the values of u_i, v_i, and b that define the solution. As with the 2-norm formulation, the selected features will be those whose corresponding slacks γ_i are greater than zero.

III. SP-SVM Extensions

In this section, we consider two different extensions of our SVM with feature selection. First, we will consider the joint selection (or removal) of features that are assigned to predefined groups, second, we will study how the SP-SVM can be extended to multi-class problems. During our derivations in this section, we will only consider the ν-SP-SVM formulation with 2-norm for the regularization term, although it would be straightforward to apply similar extensions to the standard SP-SVM or 1-norm ν-SP-SVM.

A. ν-SP-SVM with Feature Selection Over Predefined Groups

In some practical situations, variables can appear grouped together in predefined sets that can be jointly relevant or irrelevant. Then, the feature selection process must be applied over these sets rather than over the isolated features. This is for instance the case when encoding categorical variables with binary words. Either all binary variables corresponding to the same categorical feature should be selected or removed together.

Let us assume that the input features are structured in $G < d$ disjoint groups, i.e., each input feature belongs to exactly one group. Let us also denote by S_g the indexes of the g-th group of variables, with $g = 1, \ldots, G$. Then, we can modify (5) by replacing the constraints over the absolute values of each individual weight (i.e., $u_i + v_i \leq \varepsilon + \gamma_i$) by alternative constraints each one consisting of the sum of absolute values of all weights corresponding to the variables belonging to the same group

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{d}(u_i^2 + v_i^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + C'\left[\nu\varepsilon + \frac{1}{G}\sum_{g=1}^{G}\gamma_g\right] \\
\text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \quad \forall l \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& \sum_{i \in S_g} u_i + v_i \leq \varepsilon + \gamma_g; \quad \forall g \\
& u_i, v_i \geq 0; \quad \forall i \\
& \gamma_g \geq 0; \quad \forall g \\
& \varepsilon \geq 0
\end{aligned} \tag{7}
$$

where γ_g are slacks associated to each group and γ_g values greater than 0 after optimization indicate, which groups have been selected and included in the classification model. Now, parameter ν can be used to *a priori* establish the maximum number of groups that should be selected by the algorithm, thus providing a control mechanism for adjusting the degree of sparsity desired for the solution.

Finally, it is important to point out some advantages of this formulation with regard to other reference methods.

1) The standard formulation of 1-norm SVMs [26] cannot be used for feature selection in the setup that we have studied here. This is due to the fact that standard 1-norm SVM directly introduces term $\|\mathbf{w}\|_1$ in the objective function to force sparsity, making it impossible to force all coefficients of the same group to shrink to zero at the same time.
2) Forcing sparsity over groups with a group LASSO penalty term [34] precludes the standard SVM formulation, since it turns it out into a non linear convex optimization problem. Feature selection over groups only implies a modification of the introduced constraints due to the fact that our approach forces sparsity by means of additional constraints; therefore, standard LP or QP optimizers can be used to solve the problem.
3) Furthermore, if 1-norm were used to penalize weights coefficients in the functional of (7), not only groups selection would be implemented, but also sparsity within the groups would be favored.

B. Multiclass ν-SP-SVM

Here, we present the extension to multiclass classification problems by following the SVM multiclass approach from [37]. Let us consider a classification problem with K classes. Then, in this case we have $y^{(l)} \in \{1, \ldots, K\}$. Accordingly, the classification function for a linear classifier is given by

$$\hat{y} = \arg \max_{k=1,\ldots,K} \mathbf{w}_k^T \mathbf{x} + b_k \tag{8}$$

i.e., K different outputs associated to each class are computed, and then the pattern is classified according to the largest output. The set of vectors and bias terms $\{\mathbf{w}_k, b_k\}$, $k = 1, \ldots, K$, which define the classifier can be obtained as the solution to the following optimization problem:

$$\begin{aligned} \min \quad & \sum_{k=1}^{K} \|\mathbf{w}_k\|^2 + \frac{C}{N} \sum_{l=1}^{N} \xi^{(l)} \\ \text{s.t.} \quad & \left[\mathbf{w}_{y^{(l)}}^T \mathbf{x}^{(l)} + b_{y^{(l)}}\right] - \left[\mathbf{w}_m^T \mathbf{x}^{(l)} + b_m\right] \geq 2 - \xi^{(l)}; \\ & \forall l; \quad m \neq y^{(l)} \\ & \xi^{(l)} \geq 0 \qquad \forall l. \end{aligned} \tag{9}$$

As with the binary SVM, the objective function consists of the sum of two terms that are related to the structural and empirical risks. The constraints for the minimization try to force that, for each training sample, the largest output of the system is obtained for the correct class. Otherwise, slack variable $\xi^{(l)}$ will take a value equal to the distance between the largest output and the output associated to the actual class of the pattern [37].

We can now introduce sparsity constraints to allow feature selection during the training of the multiclass SVM. A straightforward extension of our strategy for the binary case would lead to

$$\begin{aligned} \min \quad & \sum_{k=1}^{K} \sum_{i=1}^{d} (u_{k,i}^2 + v_{k,i}^2) + \frac{C}{N} \sum_{l=1}^{N} \xi^{(l)} \\ & \quad + C' \left[\nu\varepsilon + \frac{1}{Kd} \sum_{k=1}^{K} \sum_{i=1}^{d} \gamma_{k,i}\right] \\ \text{s.t.} \quad & \left[\sum_{i=1}^{d} (u_{y^{(l)},i} - v_{y^{(l)},i}) x_i^{(l)} + b_{y^{(l)}}\right] \\ & - \left[\sum_{i=1}^{d} (u_{m,i} - v_{m,i}) x_i^{(l)} + b_m\right] \geq 2 - \xi^{(l)}; \ \forall l; \ m \neq y^{(l)} \\ & \xi^{(l)} \geq 0; \qquad \forall l \\ & u_{k,i} + v_{k,i} \leq \varepsilon + \gamma_{k,i}; \qquad \forall i; \ \forall k \\ & u_{k,i}, v_{k,i} \geq 0; \qquad \forall i; \ \forall k \\ & \gamma_{k,i} \geq 0; \qquad \forall i; \ \forall k \\ & \varepsilon \geq 0 \end{aligned} \tag{10}$$

where we have defined $\mathbf{w}_k = \mathbf{u}_k - \mathbf{v}_k$, and $u_{k,i}$ and $v_{k,i}$ are the i-th components of $\mathbf{u}_k$ and $\mathbf{v}_k$, respectively.

The above formulation would result in vectors $\mathbf{w}_k$ with different sparsity distributions. It should be noted, however, that in order to perform a true feature selection, it would be necessary that the irrelevant features are removed from all $\mathbf{w}_k$ at the same time. In other words, to discard a feature x_i from the final classification model, it is necessary that such a feature is simultaneously ignored for the computation of all K system outputs. In order to do so, we can use an approach similar to that in Section III-A, including in a single constraint all weights $u_{k,i}$ and $v_{k,i}$ associated to the same feature. Proceeding in this way, (10) is changed into

$$\begin{aligned} \min \quad & \sum_{k=1}^{K} \sum_{i=1}^{d} (u_{k,i}^2 + v_{k,i}^2) + \frac{C}{N} \sum_{l=1}^{N} \xi^{(l)} \\ & \quad + C' \left[\nu\varepsilon + \frac{1}{d} \sum_{i=1}^{d} \gamma_i\right] \\ \text{s.t.} \quad & \left[\sum_{i=1}^{d} (u_{y^{(l)},i} - v_{y^{(l)},i}) x_i^{(l)} + b_{y^{(l)}}\right] \\ & - \left[\sum_{i=1}^{d} (u_{m,i} - v_{m,i}) x_i^{(l)} + b_m\right] \geq 2 - \xi^{(l)}; \ \ \forall l; \ m \neq y^{(l)} \\ & \xi^{(l)} \geq 0; \qquad \forall l \\ & \sum_{k=1}^{K} u_{k,i} + v_{k,i} \leq \varepsilon + \gamma_i; \qquad \forall i \\ & u_{k,i}, v_{k,i} \geq 0; \qquad \forall i; \ \forall k \\ & \gamma_i \geq 0; \qquad \forall i \\ & \varepsilon \geq 0. \end{aligned} \tag{11}$$

The above problem can be solved using QP optimizers. At the solution, those features with an associated $\gamma_i > 0$ will be selected, while all the rest are excluded from the classifier.

TABLE I

CE RATES AND NUMBER OF FEATURES PROVIDED IN THE ORANGE DATA PROBLEM BY THE DIFFERENT METHODS UNDER STUDY: STANDARD 2 AND 1-NORM SVMS, DR-SVM AND 2 AND 1-NORM ν-SP-SVMS. PARAMETERS q AND p INDICATE THE NUMBER OF RANDOM FEATURES INCLUDED IN THE DATA SET AND THE TOTAL NUMBER OF FEATURES IN THE EXPANDED INPUT SPACE, RESPECTIVELY

q, p		Standard SVM		Dr-SVM	ν-SP-SVM	
		2-norm	1-norm		2-norm	1-norm
0, 5	CE	7.87(±2.15)	7.30(±1.18)	7.30(±1.08)	6.89(±1.08)	6.89(±1.07)
	# feat.	–	4.46(±0.93)	4.75(±0.63)	2.66(±0.94)	2.67(±0.91)
2, 14	CE	10.56(±2.50)	8.16(±1.18)	8.42(±1.39)	6.78(±1.16)	6.81(±1.15)
	# feat.	–	6.34(±3.40)	7.46(±3.30)	2.45(±1.28)	2.27(±0.88)
4, 27	CE	13.83(±2.88)	8.71(±1.39)	8.84(±1.60)	6.88(±1.28)	6.91(±1.36)
	# feat.	–	6.49(±4.65)	9.79(±3.26)	2.48(±1.35)	2.27(±0.87)
6, 44	CE	15.89(±3.01)	8.75(±1.34)	9.19(±1.61)	6.64(±1.23)	6.74(±1.34)
	# feat.	–	6.41(±4.93)	13.56(±3.79)	2.36(±1.65)	2.44(±1.47)
8, 65	CE	18.81(±2.92)	8.93(±1.49)	10.05(±2.07)	6.76(±1.37)	6.85(±1.47)
	# feat.	–	6.22(±4.21)	18.63(±5.02)	2.27(±1.21)	2.38(±1.42)
12, 119	CE	23.59(±2.83)	8.80(±1.16)	11.11(±2.94)	6.64(±1.24)	6.70(±1.22)
	# feat.	–	7.60(±3.04)	25.44(±8.41)	2.15(±1.27)	2.21(±1.32)
16, 189	CE	27.18(±2.65)	8.98(±1.40)	12.86(±3.54)	6.84(±1.30)	6.97(±1.34)
	# feat.	–	10.00(±4.65)	34.81(±8.49)	2.53(±2.10)	2.56(±1.80)

As before, parameter ν can be used to control the maximum number of features to be selected by the multiclass ν-SP-SVM.

Similarly to what we explained for the group selection case, imposing sparsity through additional constraints is key in order to perform a common feature selection for all classification problems, and approaches relying on the introduction of 1-norm penalties in the objective function would either fail to select the same features for all classification tasks, or preclude the use of standard LP or QP optimizers.

IV. EXPERIMENTS

In this section, we will test the performance of the proposed 2 and 1-norm ν-SP-SVM algorithms. For this purpose, we will analyze both the provided classification error (CE) rate and the number of selected features compared to those of standard 2 and 1-norm SVMs, as well as the Dr-SVM from [30].

In all experiments, free SVM parameters have been optimized through a cross validation (CV) process. Parameter C of standard SVMs has been logarithmically swept with 10 values from 10^{-2}N to 10^{6}N, N being the number of training data. Parameter C of ν-SP-SVMs has been explored with 5 values in the same range. For each value of C, C' has been swept in the set of values: $\{0.01C, 0.1C, C, 10C, 100C\}$. In order to evaluate the influence of ν in the number of selected features, we have considered the overall set of values $\nu = i/d$, $1 \leq i \leq d$, where d is the data dimension, when ν-SP-SVM is applied over a predefined feature group, parameter d is replaced by the number of groups G. As for Dr-SVM parameters, λ_1 and λ_2, they have been selected among the set of values $\{0.01, 0.1, 1, 10, 100\}$.

In the following discussions, both results evaluating the evolution of the CE and the number of features when ν value is explored, and results achieved when ν value is cross validated, will be analyzed. Additionally, we will include the CE achieved by a new SVM retrained with only the subset of features selected by the ν-SP-SVM methods, in this way, we will check whether the fact of pruning the weights associated to irrelevant features degrades the final model performance.

The MOSEK library[1] has been used as optimizer for all algorithms under study.

A. Orange Data Model

As a first simulation problem, we have considered the "orange data" model, which has been previously employed in [29] to test the standard 1-norm SVM performance. In this problem, two standard normal independent random variables x_1, x_2 are generated. Negative class elements of data $[x_1, x_2]^T$ satisfy inequality $4.5 \leq x_1^2 + x_2^2 \leq 8$, whereas positive elements are distributed along all space $\mathbb{R}^2$. Thus, negative class surrounds almost all positive class patterns, like the skin of an orange. Additionally, to check the feature selection ability of the different algorithms, q random independent standard Gaussian inputs have been included in the model. Finally, this input space has been expanded with a second degree polynomial function, i.e., $\{\sqrt{2}x_j, \sqrt{2}x_j x_k, x_j^2, j, k = 1, 2, \ldots, 2+q\}$ to create a new data set with p new input features.[2]

In the experiments, the number of added random features, q, has been fixed to 0, 2, 4, 6, 8, 12, and 16 generating an expanded input space of 5, 14, 27, 44, 65, 119, and 189 features. To design the different SVM classifiers, independent and balanced training, validation and test data sets have been generated with 100, 500, and 1000 data, respectively, and each simulation has been repeated 200 times. In this experiment,

[1]MOSEK ApS, Denmark. Available at http://www.mosek.com. The MOSEK Optimization Tools version 6.0 (Revision 61). User's manual and reference, 2010.

[2]Note that the Bayes boundary is given by $x_1^2+x_2^2 = 4.5$, therefore, from the overall set of p new features, only terms x_1^2 and x_2^2 are useful.

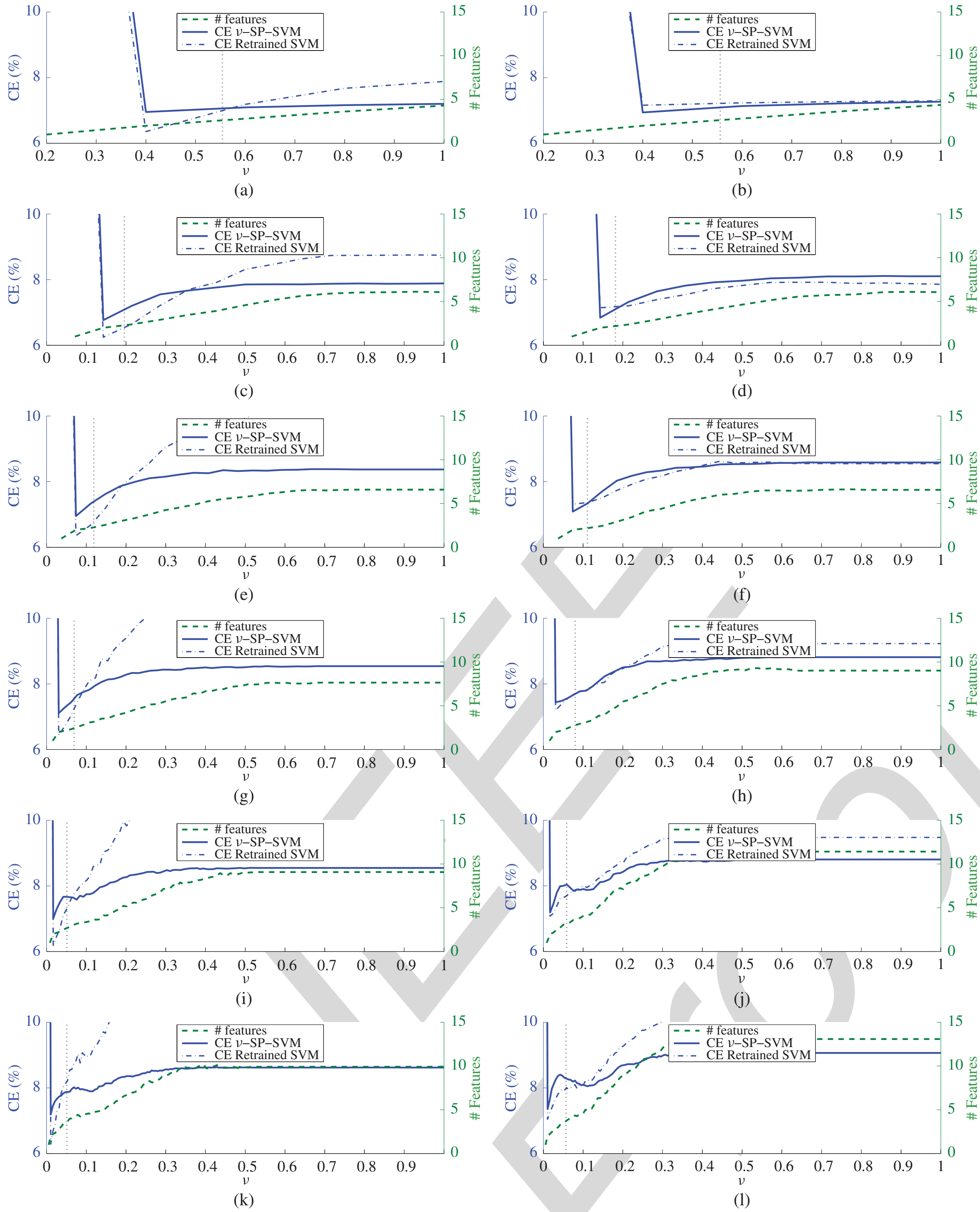


Fig. 1. Evolution of the averaged CE and the averaged number of selected features in ν-SP-SVM methods as a function of ν for orange data set. Dash-dotted line shows the averaged CE of an SVM retrained with the features selected by ν-SP-SVM. Dotted vertical line marks the averaged cross-validated ν value. (a) 2 norm ν-SP-SVM ($q = 0$). (b) 1 norm ν-SP-SVM ($q = 0$). (c) 2 norm μ-SP-SVM ($q = 2$). (d) 1 norm ν-SP-SVM ($q = 2$). (e) 2 norm ν-SP-SVM ($q = 4$). (f) 1 norm ν-SP-SVM ($q = 4$). (g) 2 norm ν-SP-SVM ($q = 8$). (h) 1 norm ν-SP-SVM ($q = 8$). (i) 2 norm ν-SP-SVM ($q = 12$). (j) 1 norm ν-SP-SVM ($q = 12$). (k) 2 norm ν-SP-SVM ($q = 16$). (l) 1 norm ν-SP-SVM ($q = 16$).

different SVM free parameters (C, C', and ν) have been optimized using the validation set.

The MATLAB code that implements the proposed ν-SP-SVM algorithms and a demo, which allows us to replicate the results shown in this section can be downloaded from http://www.tsc.uc3m.es/ hmolina/paper_nu-SP-SVM/.

Table I presents the averaged CE rates achieved by the different SVM methods under study and the number of features in their models. These results show the following.

1) Classical SVM methods rise the CE rate and the number of features in the model when q is increased, as it is expected, standard 1-norm SVM and Dr-SVM provide

sparser solutions than standard 2-norm SVM, even if some noisy features are included in the final model. Note that Dr-SVM, which penalizes with L1 and L2 norms, retains more useless features than 1-norm SVM and, although its performance improves 2-norm SVM, it is not as accurate as 1-norm SVM.

2) The proposed ν-SP-SVM approaches keep the classification error rates around 7%, independently of q and, in most cases, they only employ the useful features: note that the average number of selected features is always very close to 2. However, standard 2-norm SVM uses all original features and standard 1-norm SVM and Dr-SVM tend to include some useless features.

3) When 2-norm and 1-norm ν-SP-SVM results are compared to each other, we do not observe relevant differences, since they present similar CEs and similar number of features.

Fig. 1 depicts the evolution of the averaged classification error and the averaged number of selected features as a function of parameter ν in the orange problem, for each value of ν, parameters C and C' have been adjusted by the validation process. A dotted vertical line indicates the working point of the results from Table I, when ν was also selected in the validation process. Additionally, this figure includes the averaged CE rate, which could be achieved by retraining a new standard SVM with the set of features selected by ν-SP-SVMs. This figure shows the following behaviors of the proposed methods.

1) As it was expected, ν plays a crucial role to obtain a reduced number of features and an accurate solution. Fixing $\nu = 1$, the provided results would be similar to the standard 1-norm SVM, however, reducing ν both performance improvements and reductions in the number of model parameters could be achieved, mainly if ν was close to $2/d$.

2) The role of ν as upper bound on the number of selected features is clearly seen. When ν is close to 1, the proposed ν-SP-SVM methods do not include all original features in their models, since most noisy features are removed. For instance, when $q = 8$, 12, or 16, there are 65, 119, and 189 original features, but ν-SP-SVMs employ less than 10, 12, or 14 features.

3) Finally, it is important to point out that the model performance is not degraded by pruning the coefficients associated to irrelevant features (those whose slack variables γ_i are zero). If we compare the solutions provided by ν-SP-SVM models with a new standard SVM trained with the selected set of features, slight performance improvements could be achieved; but, when any noisy feature is included in the model, the retrained SVM tends to overfit, whereas proposed ν-SP-SVM models provide accurate solutions.

B. Benchmark Data Sets

To test the performance of the proposed ν-SP-SVM classifiers over real data sets, 8 benchmark binary classification problems have been selected from the universal communications identifier (UCI) repository [38]: *Abalone*, *Credit*, *Hand*, *Image*, *Ionosphere*, *Pima*, *Spam*, and Wisconsin Diagnostic Breast Cancer (*Wdbc*). These problems have been chosen because of their diversity in the number of data and dimensions. The main characteristics of these problems are summarized in Table II. To adjust the free parameters of the different models, the parameter ranges described in the introduction of the experimental section have been swept by applying a five-fold CV process.

TABLE II

CHARACTERISTICS OF THE BINARY DATA SETS: NUMBER OF FEATURES AND NUMBER OF DATA BELONGING TO EACH CLASS IN TRAINING AND TEST SETS

Problem	# Features (d)	# Train samples (n_1/n_{-1})	# Test samples (n_1/n_{-1})
Abalone	8	1238/1269	843/827
Credit	15	215/268	92/115
Hand	62	1923/1900	906/891
Image	18	821/1027	169/293
Ionosphere	34	150/84	75/42
Pima	8	188/350	80/150
Spam	57	1218/1847	595/941
Wdbc	30	238/141	119/71

For this benchmark analysis we have also included, as an additional reference method, the RFE method from [39]. This algorithm carries out a feature selection process by iteratively removing the feature with less weight in the SVM solution. To fairly compare this method with proposed ν-SP-SVM methods, we have implemented the linear version of the RFE algorithm, additionally, the final feature subset of the RFE method is selected with a CV process (note that the RFE method obtains a different feature subset in each iteration) and a new SVM has been trained using only the selected features.

Table III shows the results achieved by the different SVM algorithms under study averaged over 50 runs with randomly selected training/validation sets. As it can be observed, standard 1-norm SVM fails to remove irrelevant features in some problems. For instance, in *Abalone*, *Pima*, and *Spam* almost all original features are retained. Dr-SVM is worse than the standard 1-norm SVM in this regard, and hardly removes any feature in the considered problems (with the exception of *Credit*).

In contrast, it is possible to perform effective feature selection with the proposed ν-SP-SVMs without incurring in any significant degradation in classification performance. In particular, Table III shows a 25% model complexity reduction in *Image*, *Spam*, and *Wdbc* when ν-SP-SVM, as opposed to its standard counterpart, is used. This percentage is even better for other problems, reaching 33.3% in *Abalone* and *Hand* and 50% in *Ionosphere*.

When we compare the proposed ν-SP-SVM approaches with the RFE method, we observe that the automatic feature selection carried out by our proposals is competitive with standard feature selection procedures which have to, first, select the feature subset and, second, train the classifier. According to Table III, results are quite similar for most problems. However,

TABLE III

CE AND NUMBER OF SELECTED FEATURES PROVIDED BY STANDARD 2 AND 1-NORM SVMs, DR-SVM, THE RFE METHOD AND THE 2 AND 1-NORM ν-SP-SVMs IN THE BINARY CLASSIFICATION PROBLEMS

		Standard SVM		Dr-SVM	RFE	ν-SP-SVM	
		2-norm	1-norm			2-norm	1-norm
Abalone	CE	21.10(±0.89)	20.51(±0.11)	20.60(±0.14)	20.90(±0.58)	20.90(±0.37)	20.85(±0.34)
	# feat.	8.00(±0.00)	7.96(±0.20)	8.00(±0.00)	4.34(±2.18)	5.36(±2.11)	5.80(±1.87)
Credit	CE	10.65(±0.10)	11.07(±0.13)	11.07(±0.13)	10.99(±0.21)	10.68(±0.15)	11.02(±0.19)
	# feat.	15.00(±0.00)	1.16(±0.55)	2.08(±3.36)	4.32(±4.83)	7.16(±3.15)	1.36(±0.78)
Hand	CE	9.17(±0.18)	9.24(±0.10)	9.20(±0.12)	9.43(±0.22)	9.15(±0.22)	9.29(±0.21)
	# feat.	62.00(±0.00)	55.68(±4.20)	55.56(±4.08)	34.82(±6.04)	45.72(±4.96)	42.06(±5.67)
Image	CE	14.94(±0.95)	12.94(±0.18)	13.11(±0.23)	14.05(±1.07)	13.18(±0.43)	12.98(±0.19)
	# feat.	18.00(±0.00)	13.96(±0.20)	17.24(±0.77)	16.06(±1.49)	14.38(±2.58)	13.52(±1.03)
Ionosphere	CE	11.93(±2.02)	11.73(±2.35)	12.38(±0.85)	13.76(±2.12)	11.79(±1.92)	12.27(±1.08)
	# feat.	33.00(±0.00)	24.42(±7.47)	30.92(±3.29)	13.96(±5.13)	18.32(±6.55)	17.44(±3.90)
Pima	CE	23.63(±0.71)	23.29(±0.22)	23.35(±0.31)	23.78(±1.03)	23.36(±0.33)	23.00(±0.20)
	# feat.	8.00(±0.00)	7.44(±0.50)	7.76(±0.43)	5.26(±2.04)	6.34(±1.14)	6.72(±1.05)
Spam	CE	6.88(±0.17)	7.15(±0.09)	7.03(±0.06)	6.78(±0.21)	6.99(±0.24)	7.09(±0.15)
	# feat.	57.00(±0.00)	54.52(±1.79)	56.22(±0.79)	44.68(±3.03)	44.88(±3.21)	42.88(±3.28)
Wdbc	CE	2.97(±0.92)	4.31(±0.68)	3.19(±0.51)	3.43(±0.57)	3.28(±0.53)	3.77(±0.75)
	# feat.	30.00(±0.00)	18.52(±3.25)	27.38(±3.17)	21.80(±3.59)	22.64(±2.27)	13.80(±2.70)

in the case of *Image*, both ν-SP-SVM proposals outperform the RFE method, and for *Credit* and *Wdbc*, the 1-norm ν-SP-SVM approach achieves the best accuracy-complexity trade-off. On the other hand, in problems such as *Ionosphere* or *Hand*, RFE presents a lower number of features, although this advantage is achieved at the expense of a CE increase.

Figs. 2 and 3 show the evolution of the classification error and the number of selected features as a function of ν in the different data sets. A dashed line depicts the CE achieved by new standard SVMs retrained with the set of features selected by the proposed ν-SP-SVM models and a dotted vertical line points out the ν value selected in the validation process. These figures remark the clear trade-off between the model complexity and the final CE. In problems such as *Credit*, *Image*, *Ionosphere*, and *Wdbc*, when the 1-norm ν-SP-SVM is applied, we could directly have fixed $\nu = 1$, and most useless features would have been removed. However, an adequate selection of ν is crucial to obtain an accurate solution. The validation process has carried out a conservative selection of parameter ν, if, during the validation process, a slight performance degradation had been allowed, a additional features would have been removed, in fact, for all the problems under study but *Credit*, lower values of ν would have resulted in a lower number of features, while keeping similar error rates. Finally, it is important to note that the retraining procedure does not show any clear improvement, since although in some cases the final CE is slightly improved, in other cases it is similar or, even, slightly worse.

C. High Dimensional Datasets

The aim of this section is to test the performance of the proposed methods when we are dealing with a large number of input features. For this purpose, the Dexter dataset [40] has been considered. The goal of this problem is to classify texts about "corporate acquisitions" into two categories. The data set has 20 000 features, from which 9947 variables correspond to a "bag-of-words" representation of several texts and the remaining 10 053 features are noisy features added to complicate the classification task. The different data set partitions are balanced with 300 training data, 300 validation patterns and 2000 test samples.

Due to the large number of input features, the CV of all possible ν values in the ν-SP-SVM methods is not reasonable. For this reason, we have followed this strategy.

1) We have first trained the proposed methods with $\nu = 1$, what provides a first approximation to the number of useful features. In this case, 1-norm ν-SP-SVM achieves a $CE = 8.1\%$ with only 150 features and 2-norm ν-SP-SVM a $CE = 6\%$ with 3976 variables.
2) According to above number of selected features, the maximum value of ν, worthy of being explored, has been fixed. For instance, in 1-norm ν-SP-SVM this value has been fixed to 0.01 (150 is less than the 1% of 20 000) and in 2-norm ν-SP-SVM has been set to 0.2 (3976 is close to the 20% of 20 000).
3) Then, a range of 10 linearly spaced ν values has been defined. In particular, ranges $\{0.1\%, 0.2\%, \ldots, 1\%\}$ and $\{2\%, 4\%, \ldots, 20\%\}$ have been explored by each ν-SP-SVM model.
4) Finally, the optimum ν value has been selected as the one with minimum validation error.

As a result of this procedure, 1-norm ν-SP-SVM has selected a ν value of 0.004, achieving a $CE = 7.75\%$ with only 79 features, whereas 2-norm ν-SP-SVM has used a final ν value of 0.1 providing a CE of 6.4% with 1487 features. Reference methods, 2-norm, 1-norm, and Dr-SVMs, have

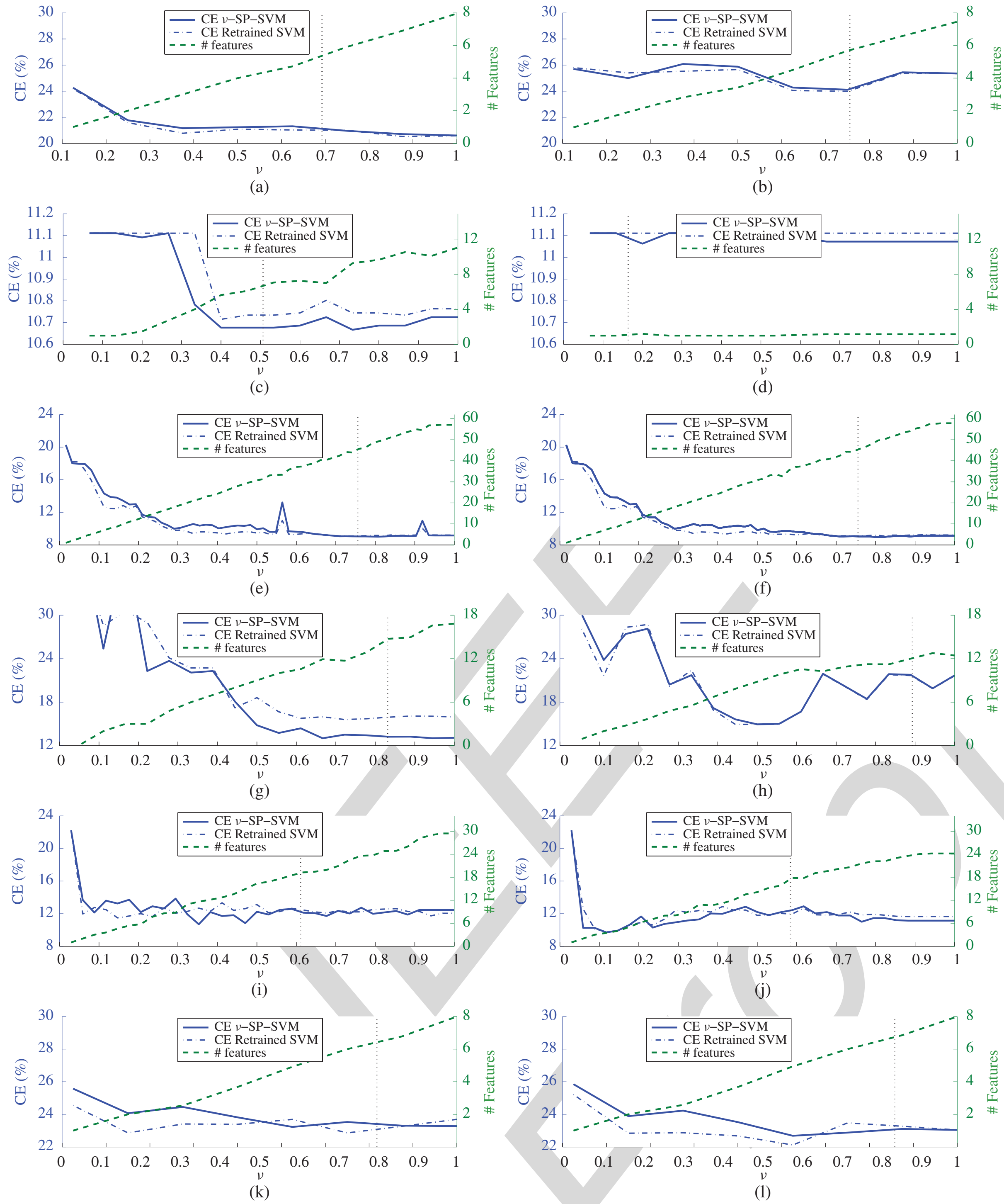


Fig. 2. Evolution of CE and the number of selected features in ν-SP-SVMs as a function of ν for data sets: *Abalone*, *Credit*, *Hand*, *Image Ionosphere*, and *Pima*. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Abalone*. (b) 1-norm ν-SP-SVM *Abalone*. (c) 2-norm ν-SP-SVM *Credit*. (d) 1-norm ν-SP-SVM *Credit*. (e) 2-norm ν-SP-SVM *Hand*. (f) 1-norm ν-SP-SVM *Hand*. (g) 2-norm ν-SP-SVM *Image*. (h) 1-norm ν-SP-SVM *Image*. (i) 2-norm ν-SP-SVM *Ionosphere*. (j) 1-norm ν-SP-SVM *Ionosphere*. (k) 2-norm ν-SP-SVM *Pima*. (l) 1-norm ν-SP-SVM *Pima*.

presented CEs of 6.45%, 8.10% and 6.05%, respectively, and they have used 7142, 159, and 5750 features (see Table IV).

These results show that 1-norm ν-SP-SVM outperforms standard 1-norm SVM by achieving a lower CE with half the number of features. Regarding 2-norm ν-SP-SVM and standard 2-norm SVM, they present similar error rates, but the latter is using 35% of the features instead of 7.43% used by 2-norm ν-SP-SVM. Finally, Dr-SVM provides the lowest CE, but the number of selected features (5750) is much higher than the 1487 of the 2-norm ν-SP-SVM.

Besides, it is important to point out that 1-norm-based algorithms (standard 1-norm SVM and 1-norm ν-SP-SVM)

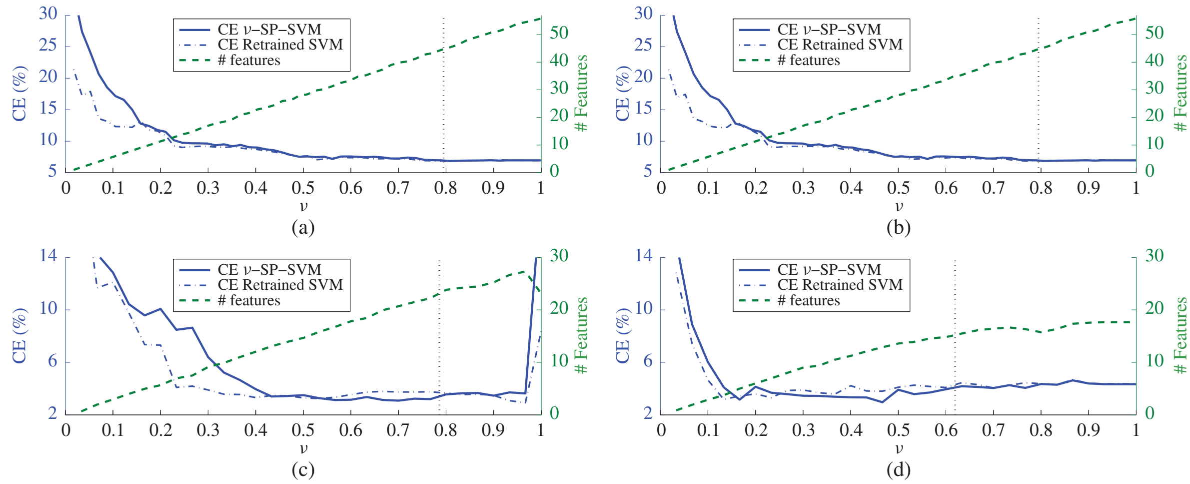


Fig. 3. Evolution of CE and the number of selected features in ν-SP-SVMs as a function of ν for data sets: *Spam* and *Wdbc*. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Spam*. (b) 1-norm ν-SP-SVM *Spam*. (c) 2-norm ν-SP-SVM *Wdbc*. (d) 1-norm ν-SP-SVM *Wdbc*.

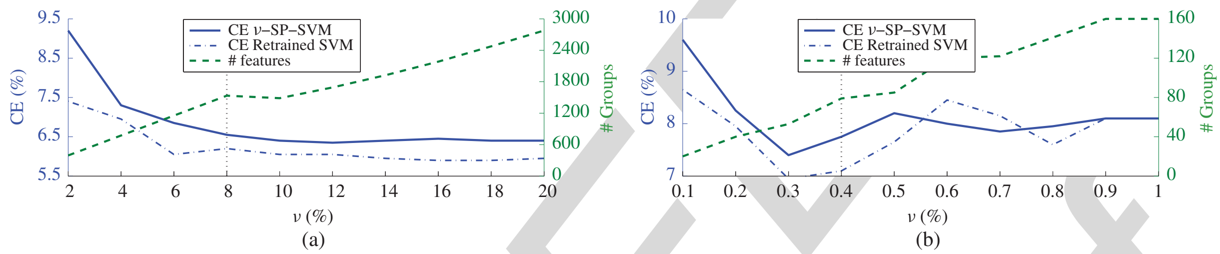


Fig. 4. CE and the number of selected features in ν-SP-SVM algorithms as a function of ν in *Dexter* data set. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM. (b) 1-norm ν-SP-SVM.

AQ:1

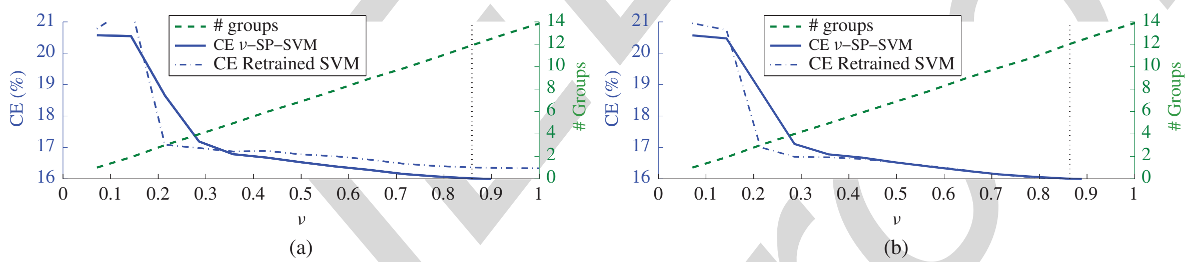


Fig. 5. Evolution of CE and the number of selected features in ν-SP-SVMs as a function of ν for data sets: *Spam* and *Wdbc*. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Spam*. (b) 1-norm ν-SP-SVM *Spam*. (c) 2-norm ν-SP-SVM *Wdbc*. (d) 1-norm ν-SP-SVM *Wdbc*.

have selected a few number of features, prompting a performance degradation. This effect is due to the fact that the maximum number of features that can be selected is always upper bounded by the number of training data [30], [32]. For this reason, these approaches are working with few hundreds of features instead of selecting thousands as the 2-norm-based methods.

Finally, Fig. 4 shows the evolution of the CE and the number of features in the model for the explored range of ν values. At first glance, it can be seen that, in the explored range of ν, values larger than 8% in 2-norm ν SP-SVM and 0.3% for 1-norm ν SP-SVM are able to provide accurate results with a low number of features, even lower than 1-norm, 2-norm, and Dr-SVM methods. This figure also shows the CE achieved when the SVM is retrained with the selected set of features, suggesting that, in problems where the number of removed features is high, the retraining process is able to provide an additional advantage in terms of CE reduction.

TABLE IV
CE AND NUMBER OF SELECTED FEATURES PROVIDED BY DIFFERENT METHODS UNDER STUDY IN DEXTER DATA SETS

		Standard SVM		Dr-SVM	ν-SP-SVM	
		2-norm	1-norm		2-norm	1-norm
Dexter	CE	6.45	8.10	6.05	6.4	7.75
	# feat.	7142	159	5750	1487	79

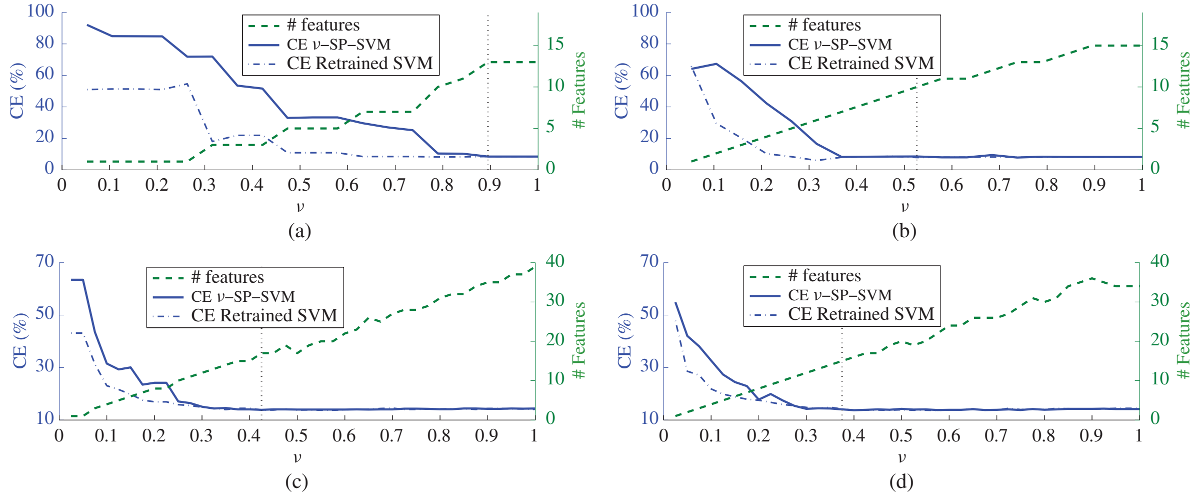


Fig. 6. Evolution of the CE and the number of selected features in ν-SP-SVM algorithms as a function of ν in multiclass problems. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model and dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Segmentation*. (b) 1-norm ν-SP-SVM *Segmentation*. (c) 2-norm ν-SP-SVM *Wave*. (d) 1-norm ν-SP-SVM *Wave*.

TABLE V

PREDEFINED FEATURE GROUPS IN THE PROBLEM ADULT. CATEGORICAL FEATURES ARE CODIFIED WITH DUMMY VARIABLES

# group	Original feature	Categorical / continous	# of categories	# of features in each group
1	age	continuous	–	1
2	workclass	categorical	8	3
3	fnlwgt	continuous	–	1
4	education	categorical	16	4
5	education-num	continuous	–	1
6	marital-status	categorical	7	3
7	occupation	categorical	14	4
8	relationship	categorical	6	3
9	race	categorical	5	3
10	sex	categorical	2	1
11	capital-gain	continuous	–	1
12	capital-loss	continuous	–	1
13	hours-per-week	continuous	–	1
14	native-country	categorical	41	6

D. Selecting Feature Groups with ν-SP-SVM

To analyze the performance of the proposed methods when features need to be selected according to predefined sets, instead of selecting isolated features, we have chosen the dataset Adult from [38]. The aim of this problem is to determine whether a person earns over 50K a year from several demographic characteristics from 14 original features, of which six are continuous and eight are categorical. Each categorical feature has been coded with dummy variables, using N indicatrix variables (0 or 1) to codify their 2^N possible values, in this way, each data is finally represented by 33 features belonging to 14 groups as it is described in Table V. Then, when a group selection approach is applied, the dummy variables representing to the same categorical feature will be either all selected or all removed from the final model. Note that only when all variables from a certain group are removed it is possible to skip the capture of the associated categorical variable.

This binary data set has 30 162 training samples and 15 060 data to test the model. To train the different SVMs, we have randomly selected a 10% of the original training data set, therefore, 3016 data have been used to train the different methods. A 5-fold CV process has been applied to adjust the free parameters of the different methods and their performances have been evaluated over whole test data. The different SVMs have been trained 100 times, with different randomly selected training data, and their averaged results have been studied.

As result, standard 2 and 1-norm SVMs present an averaged CE of 16.33(±0.3)% and 15.97(±0.2)% employing 14 and 13.9 ± 0.3 groups, respectively, whereas Dr-SVM presents the same performance (both in CE and number of selected features) as 1-norm SVMs. This result is a consequence of standard 2-norm SVM having selected all groups and 1-norm SVM and Dr-SVM having seldom discarded group 10, this group is associated to original feature *sex* and codified with only one dummy variable.

To compare these results with the proposed methods, Fig. 5 depicts the values of the CE and the number of selected groups as a function of parameter ν in ν-SP-SVMs. It can be seen that if ν is cross validated (see dotted vertical line), ν-SP-SVMs present CE close to 16% with 12 groups, since groups 3 and 10 are usually removed. However, if we had wanted to select a lower number of groups, ν could have been fixed around 0.3, keeping the CE lower than 17% and selecting just the 4 most relevant groups: Groups associated to original features *education-num*, *relationship*, and *capital-gain* are always chosen and additionally, either group 4 (*education*) or 7 (*occupation*) is included in the model. Thus, this example illustrates the convenience of the ν formulation of SP-SVM for allowing a more flexible selection of the number of variables to be incorporated in the model.

Again, a retraining process (dash-dotted line in Fig. 5) provides a small improvement, since for most ν values, ν-SP-SVMs, and retrained SVMs achieve similar CEs.

TABLE VI
CE AND NUMBER OF SELECTED FEATURES PROVIDED BY DIFFERENT METHODS UNDER STUDY IN MULTICLASS DATA SETS

		Classical SVMs		Dr-SVM	Sparse SVMs	
		2-norm	1-norm		2-norm	1-norm
Segmentation	CE	9.05	9.00	8.24	8.43	8.52
	# feat.	18.00	13.00	15	13.00	10.00
Wave	CE	13.87	14.33	14.20	13.87	14.07
	# feat.	40.00	38.00	30.00	17.00	15.00

E. Multiclass Problems

In this section, we will test the performance of the ν-SP-SVMs over multiclass datasets *Segmentation* and *Wave* from the UCI repository [38]. The purpose of *Segmentation* problem is to classify hand-segmented images represented by 19 features in 7 categories: *brickface*, *sky*, *foliage*, *cement*, *window*, *path*, and *grass*. The data set has 210 and 2100 training and test data, respectively. *Wave* problem consists of 3 classes of waves to be identified from 40 features, whose latter 19 ones are all noise, the data set has 3500 training samples and 1500 test data. As in the previous sections, the free parameters of the different methods have been adjusted with a 5 fold CV process.

To train the different classifiers, proposed ν-SP-SVM methods have solved problem (10), either in its 2-norm or in its 1-norm version, whereas reference methods have directly used the multiclass problem defined by (9) with their corresponding penalization terms. Table VI presents the results achieved by both standard and proposed SVMs. As it can be observed, ν-SP-SVMs achieve lower error rates with lower number of features. In *Segmentation*, CE is reduced in a 0.5%, with respect to 1-norm and 2-norm SVMs, using only 13 and 10 features, whereas Dr-SVM achieves a slightly lower CE using 15 features. In *Wave*, the advantages of the proposed SVM classifiers are clearer, since the number of features in the model is half the number for the reference methods and the CE is similar in the 2-norm models, slightly reduced in the 1-norm methods and Dr-SVMs are outperformed by both ν-SP-SVMs.

When the evolution of CE and the number of features are analyzed as a function of ν (see Fig. 6), the trade-off between these parameters is again observed. Besides, retrained SVMs provide a significant CE reduction in *Segmentation* problem.

V. CONCLUSION

This paper introduced a method for feature selection based on a new formulation of linear SVMs that includes constraints additional to the classical ones. These constraints drop the weights associated to those features that are likely to be irrelevant. In order to predefine an upper bound for the number of relevant features, a ν-SVM formulation has been used, where ν is a parameter that indicates the fraction of features to be considered. This parameter is swept in an efficient way in order to find the optimal number of features over a validation set of data. This paper presented two versions of the formulation, the first one being an SVM with a 2-norm regularization term. The second one uses a 1-norm regularization, that has a reduced computational burden with respect to the first one. Besides, this new SVM formulation allows us to easily apply the feature selection process over predefined feature sets. This, in turn, is useful to introduce a straightforward, yet efficient way to extend the algorithms to multiclass problems.

Experiments showed that the introduced methods present advantages not only in terms of CE, but also in the ability of reducing the model complexity by adequately removing features during the training process, not as a preprocessing stage. Also, these experiments showed that the algorithms are efficient when applied to the task of feature group selection and to multiclass problems.

Future research includes nonlinear versions of the algorithm in order to take into account the nonlinear relationships between features. Applications can also include extensions to regression problems as well as linear model selection for signal processing tasks, such as filter design or plant modeling, in situations where optimal models are known to be sparse.

REFERENCES

[1] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. Cambridge, MA: MIT Press, 2001.

[2] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[3] D. J. Sebald and J. A. Bucklew, "Support vector machine techniques for nonlinear equalization," *IEEE Trans. Signal Process.*, vol. 48, no. 11, pp. 3217–3226, Nov. 2000.

[4] M. M. Ramon, N. Xu, and C. Christodoulou, "Beamforming using support vector machines," *IEEE Antennas Wireless Propag. Lett.*, vol. 4, pp. 439–442, 2005. AQ:2

[5] Z. Shi and M. Han, "Support vector echo-state machine for chaotic time-series prediction," *IEEE Trans. Neural Netw.*, vol. 18, no. 2, pp. 359–372, Mar. 2007.

[6] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*. New York: Springer-Verlag, 2001.

[7] Y. Lin, "Support vector machines and the Bayes rule in classification," *Data Min. Knowl. Discov.*, vol. 6, no. 3, pp. 259–275, 2002.

[8] S. Rosset, J. Zhu, and T. Hastie, "Margin maximizing loss functions," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, pp. 1237–1246.

[9] J. Friedman, T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "Consistency in boosting: Discussion," *Ann. Stat.*, vol. 32, no. 1, pp. 102–107, Feb. 2004.

[10] H. Liu and H. Motoda, *Feature Selection for Knowledge Discover and Data Mining*. Norwell, MA: Kluwer, 1998.

[11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, nos. 7–8, pp. 1157–1182, Oct.–Nov. 2003.

[12] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proc. 20th Int. Conf. Mach. Learn.*, Washington D.C., 2003, pp. 1–8.
[13] R. Kohavi and G. John, "Wrappers for feature selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, 1997.
[14] L. Wang, N. Zhou, and F. Chu, "A general wrapper approach to selection of class-dependent features," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1267–1278, Jul. 2008.
[15] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Royal Stat. Soc., Ser. B*, vol. 58, no. 1, pp. 267–288, 1994.
[16] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for SVMs," in *Advances in Neural Information Processing Systems 13*, T. L. T. Dietterich and V. Tresp, Eds. Cambridge, MA: MIT Press, 2000.
[17] J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song, "Dimensionality reduction via sparse support vector machines," *J. Mach. Learn. Res.*, vol. 3, pp. 1229–1243, Mar. 2003.
[18] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.
[19] A. Rakotomamonjy, "Variable selection using SVM based criteria," *J. Mach. Learn. Res.*, vol. 3, nos. 7–8, pp. 1357–1370, 2003.
[20] J. Weston, F. Perez-Cruz, O. Bousquet, O. Chapelle, A. Elisseeff, and B. Scholkopf, "Feature selection and transduction for prediction of molecular bioactivity for drug design," *Bioinformatics*, vol. 19, no. 6, pp. 764–771, 2003.
[21] Y. Aksu, D. J. Miller, G. Kesidis, and Q. X. Yang, "Margin-maximizing feature elimination methods for linear and nonlinear kernel-based discriminant functions," *IEEE Trans. Neural Netw.*, vol. 21, no. 5, pp. 701–717, May 2010.
[22] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical LASSO," *Biostat*, vol. 9, no. 3, pp. 432–441, 2008.
[23] C. Z. J. Huang and S. Ma, "Adaptive LASSO for sparse high-dimensional regression models," *Stat. Sinica*, vol. 18, no. 374, pp. 1603–1618, 2008.
[24] N. Meinshausen and B. Yu, "LASSO-type recovery of sparse representations for high-dimensional data," *Ann. Stat.*, vol. 37, no. 1, pp. 246–270, 2009.
[25] Y. Li, P. Namburi, Z. Yu, C. Guan, J. Feng, and Z. Gu, "Voxel selection in fMRI data analysis based on sparse representation," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 10, pp. 2439–2451, Oct. 2009.
[26] P. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 82–90.
[27] L. R. Grate, C. Bhattacharyya, M. I. Jordan, and I. S. Mian, "Simultaneous relevant feature identification and classification in high-dimensional spaces," in *Algorithms in Bioinformatics* (Lecture Notes in Computer Science), vol. 2452. New York: Springer-Verlag, 2002, pp. 1–9.
[28] G. M. Fung and O. L. Mangasarian, "A feature selection Newton method for support vector machine classification," *Comput. Optim. Appl.*, vol. 28, no. 2, pp. 185–202, 2004.
[29] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani, "1-norm support vector machines," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, pp. 49–56.
[30] L. Wang, J. Zhu, and H. Zou, "The doubly regularized support vector machine," *Stat. Sinica*, vol. 16, pp. 589–615, 2006. AQ:3
[31] J. Zhu and H. Zou, "Variable selection for the linear support vector machine," in *Trends in Neural Computation* (Studies in Computational Intelligence), vol. 35, K. Chen and L. Wang, Eds. Berlin, Germany: Springer-Verlag, 2007, pp. 35–59.
[32] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. Royal Stat. Soc.*, vol. 67, no. 2, pp. 301–320, Apr. 2005.
[33] H. Zou, "An improved 1-norm support vector machine for simultaneous classification and variable selection," in *Proc. 11th Int. Conf. Artif. Intell. Stat.*, 2007, pp. 1–7.
[34] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Royal Stat. Soc.: Ser. B (Stat. Methodol.)*, vol. 68, no. 1, pp. 49–67, Feb. 2006.
[35] H. Zou and M. Yuan, "The F_∞-norm support vector machine," *Stat. Sinica*, vol. 18, pp. 379–398, 2008. AQ:4
[36] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural Comput.*, vol. 12, no. 5, pp. 1207–1245, May 2000.
[37] J. Arenas-García and F. Pérez-Cruz, "Multi-class support vector machines: A new approach," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. 2. Hong Kong, Apr. 2003, pp. 781–784.
[38] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository* [Online]. Available: http://archive.ics.uci.edu/ml
[39] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.
[40] I. Guyon, "Feature selection challenge," in *Proc. Neural Inf. Process. Syst. Workshop Feature Extract.*, Dec. 2003, pp. 1–8.

Vanessa Gómez-Verdejo was born in Madrid, Spain, in 1979. She received the telecommunication engineering degree from the Universidad Politécnica de Madrid, Madrid, in 2002. She received the Ph.D. degree from the Universidad Carlos III de Madrid, Madrid, in 2007.

She is currently a Visiting Professor in the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. She has co-authored around 20 papers, including journal and conference contributions. She has participated in several research and development projects with public funding and companies, which have provided her with an extensive experience in solving real-world problems. Her current research interests include machine learning algorithms and methods of feature selection for support vector machine classifiers.

Manel Martínez-Ramón (M'00–SM'04) received the degree in telecommunications engineering from the Universitat Politécnica de Catalunya, Barcelona, Spain, in 1994, and the Ph.D. degree in telecommunications engineering from the Universidad Carlos III de Madrid, Madrid, Spain, in 1999. AQ:5

He is with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. He has co-authored more than 20 papers in international journals and 40 conference papers on his areas of expertise. He has written a book on applications of support vector machines to antennas and electromagnetics and co-authored several book chapters. His current research interests include applications of the statistical learning to signal processing with emphasis in communications and brain imaging.

Jerónimo Arenas-García (S'00–M'04) received the degree (with honors) in telecommunication engineering from the Universidad Politécnica de Madrid, Madrid, Spain, in 2000, and the Ph.D. degree (with honors) in telecommunication technologies from the Universidad Carlos III de Madrid, Madrid, in 2004. AQ:6

He held a post-doctoral position at the Technical University of Denmark, Lyngby, Denmark, before he returned to the Universidad Carlos III de Madrid, where he is currently an Associate Professor of digital signal and information processing with the Department of Signal Theory and Communications. His current research interests include statistical learning, particularly in adaptive algorithms, advanced machine learning techniques, and their applications, for instance in remote sensing data, and multimedia information processing.

Dr. Arenas-García is a current member of the IEEE Machine Learning for Signal Processing Technical Committee.

AQ:7

Miguel Lázaro-Gredilla (M'11) received the degree (with honors) in telecommunication engineering from the University of Cantabria, Santander, Spain, in 2004, and the Ph.D. degree (with honors) from the University Carlos III de Madrid, Madrid, Spain, in 2010, where he has taught several undergraduate courses.

He is currently a Research Associate at the University of Cantabria, after two short stays at the University of Cambridge, Cambridge, U.K., and University of Manchester, Manchester, U.K. His current research interests include Gaussian processes and Bayesian models.

AQ:8

Harold Molina-Bulla (S'90–M'99) received the degree in electronic engineering from Pontificia Universidad Javeriana, Bogotá, Colombia, in 1994, and the Ph.D. degree (with honors) in telecommunication technologies from the Universidad Carlos III de Madrid, Madrid, Spain, in 1999.

He is currently a Visiting Professor of electronic communications with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. His current research interests include high performance computing for signal processing, advanced machine learning techniques, and their applications.

EDITOR QUERY

EQ:1 = Please provide the accepted date for this article.

AUTHOR QUERIES

AQ:1	=	Please provide the images for Fig. 5(c) and (d).
AQ:2	=	Please provide the issue no and publication month for the IEEE ref. [4].
AQ:3 & 4	=	Please provide the issue no or publication month for the refs. [30] and [35].
AQ:5, 6, 7, & 8	=	Please specify the degree details.

Support Vector Machines with Constraints for Sparsity in the Primal Parameters

Vanessa Gómez-Verdejo, Manel Martínez-Ramón, *Senior Member, IEEE*, Jerónimo Arenas-García, *Member, IEEE*, Miguel Lázaro-Gredilla, *Member, IEEE*, and Harold Molina-Bulla, *Member, IEEE*

***Abstract*— This paper introduces a new support vector machine (SVM) formulation to obtain sparse solutions in the primal SVM parameters, providing a new method for feature selection based on SVMs. This new approach includes additional constraints to the classical ones that drop the weights associated to those features that are likely to be irrelevant. A ν-SVM formulation has been used, where ν indicates the fraction of features to be considered. This paper presents two versions of the proposed sparse classifier, a 2-norm SVM and a 1-norm SVM, the latter having a reduced computational burden with respect to the first one. Additionally, an explanation is provided about how the presented approach can be readily extended to multiclass classification or to problems where groups of features, rather than isolated features, need to be selected. The algorithms have been tested in a variety of synthetic and real data sets and they have been compared against other state of the art SVM-based linear feature selection methods, such as 1-norm SVM and doubly regularized SVM. The results show the good feature selection ability of the approaches.**

***Index Terms*— Feature group selection, feature selection, margin maximization, multiclass classification, support vector machines.**

I. Introduction

SUPPORT vector machines (SVMs) [1], [2] are considered the state-of-art in machine learning due to their well known good performance in a wide range of applications [3]–[5]. The SVM criterion minimizes a loss term, called hinge loss, plus an additional quadratic penalization term which regularizes the solution [6]. This hinge loss minimization allows SVMs to approximate Bayes' rule without estimating the conditional class probability [7] and makes it converge to a maximum margin solution [8], thus endowing SVMs with good generalization properties.

In spite of the generally good performance of SVMs, in many practical situations, useless, redundant, or noisy features can degrade the attained solution. The reason for this is that the SVM solution is based on a combination of all input features, including the irrelevant ones. As it is stated in the bet-on-sparsity principle [9], this situation is undesired and it would be preferable to obtain a solution consisting only of the relevant features. That way, more accurate and interpretable solutions can be achieved.

To achieve this goal, a feature selection process [10], [11] is usually applied. Classical feature selection techniques, such as filtering [12] or wrapping [13], [14] approaches, are used as an independent preprocessing step before the training of the final classification (or regression) machine. More recent feature selection methods combine the feature selection process with the final predictor training. For instance, in [15]–[17] an objective function that combines an accuracy prediction term with a term associated to the sparsity in the number of selected variables is employed. In [18]–[20] the SVM prediction output is considered as a linear combination of kernel functions and then, the prediction accuracy is evaluated as a function of the used and discarded features. This method, known as recursive feature elimination (RFE), has been widely employed for SVM classification, however, recent works [21] have shown that RFE is not consistent with maximum margin solutions.

In contrast to the approaches that include an explicit feature selection strategy (either independent or combined with the classification step), classifiers directly providing sparse solutions are usually preferred. Following this point of view, the LASSO method was proposed in [15]. LASSO includes a 1-norm regularization term in the optimization problem. Since this norm has a singularity at the origin, some coefficients of the solution vector are shrunk to zero, what provides sparse solutions. Since then, many researchers have focused their work on minimizing 1-norm penalized functions [22]–[24]. In fact [25] points out the need and usefulness of linear sparse solutions in problems like functional magnetic resonance imaging.

In [26], the classical SVM formulation is modified by replacing the quadratic penalization term with a 1-norm penalty, what leads to solutions with sparse coefficients. Although this SVM formulation can only be used for feature selection in linear classification problems, this approach has nevertheless been successfully used in a large number of applications, such as computational biology [27], [28], drug-design [17] or gene microarrays classification [29], among others.

Although 1-norm SVMs retain most of the desired properties of classical SVMs, such as margin maximization, they may fail to provide good solutions in certain situations. As it is

EQ:1 Manuscript received June 22, 2010; revised April 11, 2011; accepted XXXX XX, XXXX. This work was supported in part by the Ministry of Science and Innovation (Spanish Goverment), under Grant TEC2008-02473.

V. Gómez-Verdejo, M. Martínez-Ramón, J. Arenas-García, and H. Molina-Bulla are with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Madrid 28911, Spain (e-mail: vanessa@tsc.uc3m.es; manel@tsc.uc3m.es; jarenas@tsc.uc3m.es; hmolina@tsc.uc3m.es).

M. Lázaro-Gredilla is with the Department of Communication Engineering, Universidad de Cantabria, Santander 39005, Spain (e-mail: miguellg@gtas.dicom.unican.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2011.2148727

illustrated in [9], when most of the input features are relevant for the classification task at hand, classical 2-norm SVMs usually outperform their 1-norm counterparts. Furthermore, as it is pointed out in [30] and [31], the 1-norm SVM presents two additional limitations: first, when there are highly correlated variables, it usually removes some of them, and, second, the maximum number of selected features is limited by the number of available training data. Trying to overcome these drawbacks, elastic nets [32] and their particularization to SVMs by means of the doubly regularized support vector machine (Dr-SVM) [30], [31] are proposed, this new approach generalizes the LASSO and 1-norm SVM methods by keeping the 2-norm regularization term and including an additional 1-norm penalty term to force sparsity. Despite common improved performance of Dr-SVM, both 1-norm and Dr-SVMs are not suitable methods when the underlying model is truly sparse, since they are not able to remove all unnecessary variables from the final classifier, this problem was already remarked for 1-norm SVMs in [33] and, in the experimental section of this paper, we will illustrate it for Dr-SVM.

An additional limitation of 1-norm SVM and Dr-SVM, is that they are not well suited to multiclass classification or to problems where features have to be selected or removed using predefined groups. One possible solution could consist in adding a group LASSO [34] or an ∞-norm [35] penalization term into the SVM formulation. However, both options result in a more complex SVM formulation, which cannot be solved with standard linear programming (LP) or quadratic programming (QP) solvers.

In this paper, a new SVM formulation for the linear case is presented that directly forces sparse solutions. Rather than modifying the objective function, additional constraints are included in the minimization task in order to identify irrelevant features and to drop their associated weights to values lower than a small parameter ε. This constant can be adjusted during the optimization problem resolution by predefining the number of relevant features to be kept in the final solution using a ν-SVM formulation [36]. We will show that these additional constraints can be incorporated to force sparsity in both 2-norm and 1-norm SVM formulations. Our approach allows to overcome the limitations of 1-norm SVMs and Dr-SVMs in different ways. First, by properly adjusting parameter ν, the algorithm is able to remove all irrelevant features from the final model. Second, the proposed formulation can be applied to the selection of isolated features or predefined feature groups where needed. Finally, as it will be shown in the experiments section, more accurate solutions are usually achieved, particularly, when using the new constraints together with the 2-norm SVM.

The rest of this paper is organized as follows. In the next section, we introduce our approach to force feature selection in SVM classifiers, explaining how it can be applied both to 2-norm and 1-norm formulations. Section III presents some extensions of the method to address the selection of features in predefined groups of variables, as well as for multiclass classification problems. Section IV presents extensive simulation work to illustrate the performance of our approach, and its advantages with respect to previous proposals for feature selection in SVMs. Finally, Section V presents the main conclusion of our work, and identifies some lines for future research.

II. SVM with Explicit Constraints for Feature Selection

A. Problem Overview

In this paper, we consider classification problems where the representation of the input data contains some features, which are irrelevant for the task at hand. This may happen as a consequence of redundancy between the input variables or, simply, because some of the input features do not carry any valuable information for the classification. In a standard machine learning setup, we are given a set of N training labeled data, $\mathcal{S} = \{\mathbf{x}^{(l)}, y^{(l)}\}$, $l = 1, \ldots, N$, where $\mathbf{x}^{(l)} \in \Re^d$ are the input vectors and $y^{(l)}$ are used to encode class membership, from which we have to learn both the subset of relevant input variables and the classification function itself.

Linear classifiers obtain their outputs according to a thresholded version of the estimator

$$\hat{y} = \mathbf{w}^T \mathbf{x} + b \qquad (1)$$

where $\hat{y}$ is the output of the classifier for input vector $\mathbf{x}$, $\mathbf{w}$ is the vector that defines the classifier, and b is a bias term. For the SVM case, the Representer's Theorem [1], [2] states that the solution vector will lie in the subspace spanned by all training vectors $\{\mathbf{x}^{(l)}\}$. When irrelevant features are present in the data we can carry out a pre-processing stage to select the most informative variables or, alternatively, discard the variables x_i whose associated weight w_i is exactly zero after the optimization of the classifier. However, since noise is normally present in the data, none of the components of $\mathbf{w}$ will be exactly zero unless sparsity is included as an optimization criterion during the training of the classifier.

A standard way to impose sparsity in $\mathbf{w}$ is to include a regularization term in the cost function, based on the 1-norm of $\mathbf{w}$, i.e., $\|\mathbf{w}\|_1 = \sum_i |w_i|$. This regularizer presents singularity points whenever any of the components of $\mathbf{w}$ is zero, what tends to nullify some of the solution weights, thus favoring sparse solutions. However, this mechanism does not necessarily imply that all weight components associated to irrelevant variables will become zero [33].

Rather than modifying the structural risk term in the SVM functional, in this paper, we propose a new approach to impose sparsity in the solution by introducing a set of additional constraints for the optimization problem. We will see that our method is able to automatically identify all irrelevant features, thus constituting an effective mechanism for implementing SVMs that incorporate a feature selection approach. Furthermore, since the 2-norm regularization term can still be used, this usually results in a better performance when the true underlying solution is non sparse.

B. 2-Norm SVMs with Sparsity Constraints

Classical SVMs are based on the minimization of a functional that includes two terms. The first term is the squared norm of the weight vector $\mathbf{w}$, which is inversely proportional to the margin of classification [1], thus, this term is related to the structural risk of the classifier and to its generalization capabilities. The second term in the objective functional, which is known as the empirical risk term, is a sum of errors over the training data. In other words, the linear SVM problem can be stated as

$$\begin{aligned}\min \quad & \|\mathbf{w}\|^2 + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} \\ \text{s.t.} \quad & y^{(l)}\left(\mathbf{w}^T\mathbf{x}^{(l)} + b\right) \geq 1 - \xi^{(l)}; \qquad \forall l \\ & \xi^{(l)} \geq 0; \qquad \forall l\end{aligned} \tag{2}$$

where slack variables $\xi^{(l)}$ are introduced to allow some of the training patterns to be misclassified or to lie inside the classifier margin, and where C is a constant that controls the trade-off between the structural and empirical risk terms.

As it is well known, this optimization method provides a sparse solution in the sense that $\mathbf{w}$ is a linear combination of only a subset of the training data [the so-called support vectors (SVs)]. However, if feature selection is pursued during the optimization, a solution sparse in the parameters $\mathbf{w}$ is needed. In order to obtain such a solution, we will introduce some additional constraints in the optimization problem.

We start by rewriting each of the weight components, w_i, $i = 1, \ldots, d$, as $w_i = u_i - v_i$, with $u_i, v_i \geq 0$. As we will explain later, our optimization problem will implicitly enforce that at least one of the two terms in the subtraction, u_i or v_i, is zero, depending on whether the optimal weight is positive ($u_i > 0$ and $v_i = 0$), negative ($u_i = 0$ and $v_i > 0$) or zero ($u_i = v_i = 0$). Therefore, the square norm of the weight vector is given, in terms of these new variables, by

$$\|\mathbf{w}\|_2^2 = \sum_{i=1}^{d} u_i^2 + v_i^2. \tag{3}$$

Furthermore, in order to obtain a sparse solution in $\mathbf{w}$, we introduce some additional constraints to upper bound the absolute value of weight components by a small constant ε, i.e., $|w_i| = u_i + v_i < \varepsilon$. Introducing (3) and the new constraints into (2), we get the following modified SVM formulation:

$$\begin{aligned}\min \quad & \sum_{i=1}^{d}(u_i^2 + v_i^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + \frac{C'}{d}\sum_{i=1}^{d}\gamma_i \\ \text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \qquad \forall l \\ & \xi^{(l)} \geq 0; \qquad \forall l \\ & u_i + v_i \leq \varepsilon + \gamma_i; \qquad \forall i \\ & u_i, v_i \geq 0; \qquad \forall i \\ & \gamma_i \geq 0; \qquad \forall i.\end{aligned} \tag{4}$$

Although the above optimization problem has not explicitly included, the constraint $u_i v_i = 0$, (4) is indirectly forcing that either u_i or v_i is equal to 0. Note that among all possible pairs of values (u_i, v_i) that are able to provide a certain value w_i, the pair which minimizes $\sum_{i=1}^{d}(u_i^2 + v_i^2)$ has to fix either u_i or v_i to 0, for instance, for positive w_i and according to its definition in terms of u_i and v_i, minimization of the functional in (4) will lead to $v_i = 0$ and $u_i = w_i$. The opposite situation will occur for $w_i < 0$.

Note that in our redefinition of the problem we have introduced new slack variables γ_i and those slack variables associated with relevant features will be greater than zero after the functional optimization. Thus, these constants need to be introduced in the objective functional weighted with a trade-off parameter C'. The above minimization problem can be directly solved in the primal over the variables u_i, v_i, b, γ_i, and $\xi^{(l)}$, using standard QP algorithm.

We can now get some insight into the sparsity mechanism that has been adopted. If irrelevant features are present in the input representation space, most classification schemes would still assign them a non zero weight w_i due to the noise present in the data. However, if a w_i value greater than ε were assigned in our scheme, γ_i would be strictly positive, increasing the value of the functional. Thus, on the one hand irrelevant features that do not significantly decrease the empirical error term will simply be assigned weights smaller, in absolute terms, than ε. On the other hand, components w_i which are necessary to define the SVM solution will have values larger than ε. It is straightforward to use the values of slacks γ_i after the optimization to check whether a variable has been removed or incorporated into the classification model.

This new SVM with sparsity constraints performs feature selection on the input variables, so we will hereafter refer to it as sparse primal support vector machine (SP-SVM).

At first sight, one could think that the sparsity constraints in (4) are equivalent to a 1-norm penalty term and thus algorithm (4) is equivalent to Dr-SVM. Nevertheless, these constraints have been introduced here through an ε-insensitive cost function. As we will analyze along this paper, this new formulation provides two advantages: 1) the sparsity of the model can be easily adjusted by the user through a ν SVM formulation, and 2) extensions of this model to group feature selection and multiclass problems are straightforwardly derived.

The computational cost of (4) is larger than that of 1-norm or Dr-SVMs due to the new constrains. However, an efficient implementation of the problem, which exploits the sparse formulation of these constrains, it results in a very moderate computational increase.

Finally, it is important to point out that a major limitation of problem (4), as well as 1-norm and Dr-SVM algorithms, is their linear formulation. Note that their non linear extension would provide a non linear boundary with a kernel selection mechanism, instead of an automatic feature selection criterion.

C. 2-Norm ν-SP-SVM

In this section, we introduce a modification of the SP-SVM formulation in (4) to automatically adjust the value of ε, following the ν-SVM that was introduced in [36]. In this formulation of the SVM, ε is traded off against model complexity and slack variables through a constant $\nu \in (0, 1]$. Then, the optimization problem to solve is given by

$$\min \quad \sum_{i=1}^{d}(u_i^2 + v_i^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + C'\left[\nu\varepsilon + \frac{1}{d}\sum_{i=1}^{d}\gamma_i\right]$$

$$
\begin{aligned}
\text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \quad \forall l \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& u_i + v_i \leq \varepsilon + \gamma_i; \quad \forall i \\
& u_i, v_i \geq 0; \quad \forall i \\
& \gamma_i \geq 0; \quad \forall i \\
& \varepsilon \geq 0.
\end{aligned} \tag{5}
$$

As above, this optimization problem can be directly solved in the primal, with respect to variables u_i, v_i, b, γ_i, $\xi^{(l)}$, and ε.

It is well known [36] that, when the standard ν support vector regression is applied resulting a non zero ε, ν is an upper bound on the fraction of errors and a lower bound on the fraction of SVs. Note that in (5), if the dual formulation of the problem was used and we let $\{\beta_i\}_{i=1}^{d}$ be the dual variables associated to the sparsity constraints, the following equalities had to be verified:

$$\sum_{i=1}^{d}\beta_i \leq \frac{C'}{d}\nu$$

$$0 \leq \beta_i \leq \frac{C'}{d}$$

what forces ν to be an upper bound of the number of dual variables β_i taking a value of C'/d, that is, ν is an upper bound over the number of slack variables γ_i different from 0. This leads to a useful result for the proposed ν-SP-SVM: ν is an upper bound on the fraction of components of $\mathbf{w}$ whose absolute value is less than ε. In other words, parameter ν can be used to control the sparsity of the solution, setting *a priori* the maximum number of features that can be selected by the 2-norm ν-SP-SVM.

D. 1-Norm ν-SP-SVM

Using the 1-norm of $\mathbf{w}$ in the structural risk term of classical SVMs leads to LP problems, which have a reduced computational burden when compared to the QP formulation required for 2-norm SVMs. Similar benefits can be obtained for the SP-SVM proposed in the previous sections. Note that the constraints that were imposed in order to force sparsity do not affect the regularizer for $\mathbf{w}$ in any way, thus, in order to extend either (4) or (5) to the 1-norm case, it is sufficient to replace the structural risk term accordingly. For instance, for the ν-SP-SVM in its 1-norm version this leads to

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{d}(u_i + v_i) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + C'\left[\nu\varepsilon + \frac{1}{d}\sum_{i=1}^{d}\gamma_i\right] \\
\text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \quad \forall l \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& u_i + v_i \leq \varepsilon + \gamma_i; \quad \forall i \\
& u_i, v_i \geq 0; \quad \forall i \\
& \gamma_i \geq 0; \quad \forall i \\
& \varepsilon \geq 0.
\end{aligned} \tag{6}
$$

Using LP optimization tools, this problem can be solved in a more efficient way than with QP optimizers, obtaining the values of u_i, v_i, and b that define the solution. As with the 2-norm formulation, the selected features will be those whose corresponding slacks γ_i are greater than zero.

III. SP-SVM Extensions

In this section, we consider two different extensions of our SVM with feature selection. First, we will consider the joint selection (or removal) of features that are assigned to predefined groups, second, we will study how the SP-SVM can be extended to multi-class problems. During our derivations in this section, we will only consider the ν-SP-SVM formulation with 2-norm for the regularization term, although it would be straightforward to apply similar extensions to the standard SP-SVM or 1-norm ν-SP-SVM.

A. ν-SP-SVM with Feature Selection Over Predefined Groups

In some practical situations, variables can appear grouped together in predefined sets that can be jointly relevant or irrelevant. Then, the feature selection process must be applied over these sets rather than over the isolated features. This is for instance the case when encoding categorical variables with binary words. Either all binary variables corresponding to the same categorical feature should be selected or removed together.

Let us assume that the input features are structured in $G < d$ disjoint groups, i.e., each input feature belongs to exactly one group. Let us also denote by S_g the indexes of the g-th group of variables, with $g = 1, \ldots, G$. Then, we can modify (5) by replacing the constraints over the absolute values of each individual weight (i.e., $u_i + v_i \leq \varepsilon + \gamma_i$) by alternative constraints each one consisting of the sum of absolute values of all weights corresponding to the variables belonging to the same group

$$
\begin{aligned}
\min \quad & \sum_{i=1}^{d}(u_i^2 + v_i^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} + C'\left[\nu\varepsilon + \frac{1}{G}\sum_{g=1}^{G}\gamma_g\right] \\
\text{s.t.} \quad & y^{(l)}\left[\sum_{i=1}^{d}(u_i - v_i)x_i^{(l)} + b\right] \geq 1 - \xi^{(l)}; \quad \forall l \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& \sum_{i \in S_g} u_i + v_i \leq \varepsilon + \gamma_g; \quad \forall g \\
& u_i, v_i \geq 0; \quad \forall i \\
& \gamma_g \geq 0; \quad \forall g \\
& \varepsilon \geq 0
\end{aligned} \tag{7}
$$

where γ_g are slacks associated to each group and γ_g values greater than 0 after optimization indicate, which groups have been selected and included in the classification model. Now, parameter ν can be used to *a priori* establish the maximum number of groups that should be selected by the algorithm, thus providing a control mechanism for adjusting the degree of sparsity desired for the solution.

Finally, it is important to point out some advantages of this formulation with regard to other reference methods.

1) The standard formulation of 1-norm SVMs [26] cannot be used for feature selection in the setup that we have studied here. This is due to the fact that standard 1-norm SVM directly introduces term $\|\mathbf{w}\|_1$ in the objective function to force sparsity, making it impossible to force all coefficients of the same group to shrink to zero at the same time.
2) Forcing sparsity over groups with a group LASSO penalty term [34] precludes the standard SVM formulation, since it turns it out into a non linear convex optimization problem. Feature selection over groups only implies a modification of the introduced constraints due to the fact that our approach forces sparsity by means of additional constraints; therefore, standard LP or QP optimizers can be used to solve the problem.
3) Furthermore, if 1-norm were used to penalize weights coefficients in the functional of (7), not only groups selection would be implemented, but also sparsity within the groups would be favored.

B. Multiclass ν-SP-SVM

Here, we present the extension to multiclass classification problems by following the SVM multiclass approach from [37]. Let us consider a classification problem with K classes. Then, in this case we have $y^{(l)} \in \{1, \ldots, K\}$. Accordingly, the classification function for a linear classifier is given by

$$\hat{y} = \arg \max_{k=1,\ldots,K} \mathbf{w}_k^T \mathbf{x} + b_k \tag{8}$$

i.e., K different outputs associated to each class are computed, and then the pattern is classified according to the largest output. The set of vectors and bias terms $\{\mathbf{w}_k, b_k\}$, $k = 1, \ldots, K$, which define the classifier can be obtained as the solution to the following optimization problem:

$$\begin{aligned}
\min \quad & \sum_{k=1}^{K} \|\mathbf{w}_k\|^2 + \frac{C}{N}\sum_{l=1}^{N} \xi^{(l)} \\
\text{s.t.} \quad & \left[\mathbf{w}_{y^{(l)}}^T \mathbf{x}^{(l)} + b_{y^{(l)}}\right] - \left[\mathbf{w}_m^T \mathbf{x}^{(l)} + b_m\right] \geq 2 - \xi^{(l)}; \\
& \forall l; \quad m \neq y^{(l)} \\
& \xi^{(l)} \geq 0 \quad \forall l.
\end{aligned} \tag{9}$$

As with the binary SVM, the objective function consists of the sum of two terms that are related to the structural and empirical risks. The constraints for the minimization try to force that, for each training sample, the largest output of the system is obtained for the correct class. Otherwise, slack variable $\xi^{(l)}$ will take a value equal to the distance between the largest output and the output associated to the actual class of the pattern [37].

We can now introduce sparsity constraints to allow feature selection during the training of the multiclass SVM. A straightforward extension of our strategy for the binary case would lead to

$$\begin{aligned}
\min \quad & \sum_{k=1}^{K}\sum_{i=1}^{d}(u_{k,i}^2 + v_{k,i}^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} \\
& + C'\left[\nu\varepsilon + \frac{1}{Kd}\sum_{k=1}^{K}\sum_{i=1}^{d}\gamma_{k,i}\right] \\
\text{s.t.} \quad & \left[\sum_{i=1}^{d}(u_{y^{(l)},i} - v_{y^{(l)},i})x_i^{(l)} + b_{y^{(l)}}\right] \\
& - \left[\sum_{i=1}^{d}(u_{m,i} - v_{m,i})x_i^{(l)} + b_m\right] \geq 2 - \xi^{(l)}; \ \forall l; \ m \neq y^{(l)} \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& u_{k,i} + v_{k,i} \leq \varepsilon + \gamma_{k,i}; \quad \forall i; \ \forall k \\
& u_{k,i}, v_{k,i} \geq 0; \quad \forall i; \ \forall k \\
& \gamma_{k,i} \geq 0; \quad \forall i; \ \forall k \\
& \varepsilon \geq 0
\end{aligned} \tag{10}$$

where we have defined $\mathbf{w}_k = \mathbf{u}_k - \mathbf{v}_k$, and $u_{k,i}$ and $v_{k,i}$ are the i-th components of $\mathbf{u}_k$ and $\mathbf{v}_k$, respectively.

The above formulation would result in vectors $\mathbf{w}_k$ with different sparsity distributions. It should be noted, however, that in order to perform a true feature selection, it would be necessary that the irrelevant features are removed from all $\mathbf{w}_k$ at the same time. In other words, to discard a feature x_i from the final classification model, it is necessary that such a feature is simultaneously ignored for the computation of all K system outputs. In order to do so, we can use an approach similar to that in Section III-A, including in a single constraint all weights $u_{k,i}$ and $v_{k,i}$ associated to the same feature. Proceeding in this way, (10) is changed into

$$\begin{aligned}
\min \quad & \sum_{k=1}^{K}\sum_{i=1}^{d}(u_{k,i}^2 + v_{k,i}^2) + \frac{C}{N}\sum_{l=1}^{N}\xi^{(l)} \\
& + C'\left[\nu\varepsilon + \frac{1}{d}\sum_{i=1}^{d}\gamma_{i}\right] \\
\text{s.t.} \quad & \left[\sum_{i=1}^{d}(u_{y^{(l)},i} - v_{y^{(l)},i})x_i^{(l)} + b_{y^{(l)}}\right] \\
& - \left[\sum_{i=1}^{d}(u_{m,i} - v_{m,i})x_i^{(l)} + b_m\right] \geq 2 - \xi^{(l)}; \ \forall l; \ m \neq y^{(l)} \\
& \xi^{(l)} \geq 0; \quad \forall l \\
& \sum_{k=1}^{K} u_{k,i} + v_{k,i} \leq \varepsilon + \gamma_{i}; \quad \forall i \\
& u_{k,i}, v_{k,i} \geq 0; \quad \forall i; \ \forall k \\
& \gamma_{i} \geq 0; \quad \forall i \\
& \varepsilon \geq 0.
\end{aligned} \tag{11}$$

The above problem can be solved using QP optimizers. At the solution, those features with an associated $\gamma_i > 0$ will be selected, while all the rest are excluded from the classifier.

TABLE I

CE Rates and Number of Features Provided in the Orange Data Problem by the Different Methods Under Study: Standard 2 and 1-Norm SVMs, Dr-SVM and 2 and 1-Norm ν-SP-SVMs. Parameters q and p Indicate the Number of Random Features Included in the Data Set and the Total Number of Features in the Expanded Input Space, Respectively

$q,\ p$		Standard SVM		Dr-SVM	ν-SP-SVM	
		2-norm	1-norm		2-norm	1-norm
0, 5	CE	7.87(±2.15)	7.30(±1.18)	7.30(±1.08)	6.89(±1.08)	6.89(±1.07)
	# feat.	–	4.46(±0.93)	4.75(±0.63)	2.66(±0.94)	2.67(±0.91)
2, 14	CE	10.56(±2.50)	8.16(±1.18)	8.42(±1.39)	6.78(±1.16)	6.81(±1.15)
	# feat.	–	6.34(±3.40)	7.46(±3.30)	2.45(±1.28)	2.27(±0.88)
4, 27	CE	13.83(±2.88)	8.71(±1.39)	8.84(±1.60)	6.88(±1.28)	6.91(±1.36)
	# feat.	–	6.49(±4.65)	9.79(±3.26)	2.48(±1.35)	2.27(±0.87)
6, 44	CE	15.89(±3.01)	8.75(±1.34)	9.19(±1.61)	6.64(±1.23)	6.74(±1.34)
	# feat.	–	6.41(±4.93)	13.56(±3.79)	2.36(±1.65)	2.44(±1.47)
8, 65	CE	18.81(±2.92)	8.93(±1.49)	10.05(±2.07)	6.76(±1.37)	6.85(±1.47)
	# feat.	–	6.22(±4.21)	18.63(±5.02)	2.27(±1.21)	2.38(±1.42)
12, 119	CE	23.59(±2.83)	8.80(±1.16)	11.11(±2.94)	6.64(±1.24)	6.70(±1.22)
	# feat.	–	7.60(±3.04)	25.44(±8.41)	2.15(±1.27)	2.21(±1.32)
16, 189	CE	27.18(±2.65)	8.98(±1.40)	12.86(±3.54)	6.84(±1.30)	6.97(±1.34)
	# feat.	–	10.00(±4.65)	34.81(±8.49)	2.53(±2.10)	2.56(±1.80)

As before, parameter ν can be used to control the maximum number of features to be selected by the multiclass ν-SP-SVM.

Similarly to what we explained for the group selection case, imposing sparsity through additional constraints is key in order to perform a common feature selection for all classification problems, and approaches relying on the introduction of 1-norm penalties in the objective function would either fail to select the same features for all classification tasks, or preclude the use of standard LP or QP optimizers.

IV. Experiments

In this section, we will test the performance of the proposed 2 and 1-norm ν-SP-SVM algorithms. For this purpose, we will analyze both the provided classification error (CE) rate and the number of selected features compared to those of standard 2 and 1-norm SVMs, as well as the Dr-SVM from [30].

In all experiments, free SVM parameters have been optimized through a cross validation (CV) process. Parameter C of standard SVMs has been logarithmically swept with 10 values from 10^{-2}N to 10^{6}N, N being the number of training data. Parameter C of ν-SP-SVMs has been explored with 5 values in the same range. For each value of C, C' has been swept in the set of values: $\{0.01C, 0.1C, C, 10C, 100C\}$. In order to evaluate the influence of ν in the number of selected features, we have considered the overall set of values $\nu = i/d$, $1 \leq i \leq d$, where d is the data dimension, when ν-SP-SVM is applied over a predefined feature group, parameter d is replaced by the number of groups G. As for Dr-SVM parameters, λ_1 and λ_2, they have been selected among the set of values $\{0.01, 0.1, 1, 10, 100\}$.

In the following discussions, both results evaluating the evolution of the CE and the number of features when ν value is explored, and results achieved when ν value is cross validated, will be analyzed. Additionally, we will include the CE achieved by a new SVM retrained with only the subset of features selected by the ν-SP-SVM methods, in this way, we will check whether the fact of pruning the weights associated to irrelevant features degrades the final model performance.

The MOSEK library[1] has been used as optimizer for all algorithms under study.

A. Orange Data Model

As a first simulation problem, we have considered the "orange data" model, which has been previously employed in [29] to test the standard 1-norm SVM performance. In this problem, two standard normal independent random variables x_1, x_2 are generated. Negative class elements of data $[x_1, x_2]^T$ satisfy inequality $4.5 \leq x_1^2 + x_2^2 \leq 8$, whereas positive elements are distributed along all space $\mathbb{R}^2$. Thus, negative class surrounds almost all positive class patterns, like the skin of an orange. Additionally, to check the feature selection ability of the different algorithms, q random independent standard Gaussian inputs have been included in the model. Finally, this input space has been expanded with a second degree polynomial function, i.e., $\{\sqrt{2}x_j, \sqrt{2}x_j x_k, x_j^2, j, k = 1, 2, \ldots, 2 + q\}$ to create a new data set with p new input features.[2]

In the experiments, the number of added random features, q, has been fixed to 0, 2, 4, 6, 8, 12, and 16 generating an expanded input space of 5, 14, 27, 44, 65, 119, and 189 features. To design the different SVM classifiers, independent and balanced training, validation and test data sets have been generated with 100, 500, and 1000 data, respectively, and each simulation has been repeated 200 times. In this experiment,

[1] MOSEK ApS, Denmark. Available at http://www.mosek.com. The MOSEK Optimization Tools version 6.0 (Revision 61). User's manual and reference, 2010.

[2] Note that the Bayes boundary is given by $x_1^2 + x_2^2 = 4.5$, therefore, from the overall set of p new features, only terms x_1^2 and x_2^2 are useful.

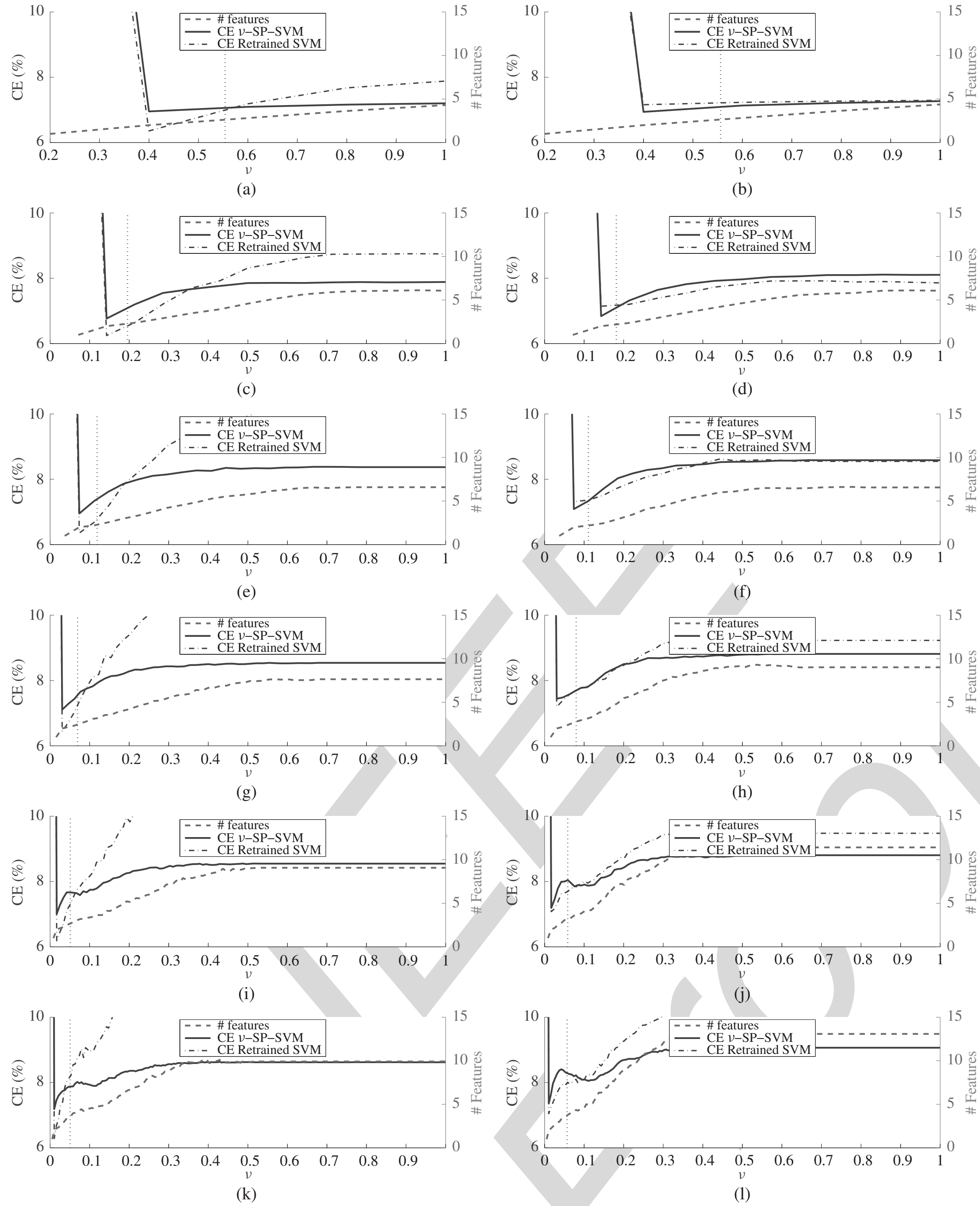


Fig. 1. Evolution of the averaged CE and the averaged number of selected features in ν-SP-SVM methods as a function of ν for orange data set. Dash-dotted line shows the averaged CE of an SVM retrained with the features selected by ν-SP-SVM. Dotted vertical line marks the averaged cross-validated ν value. (a) 2 norm ν-SP-SVM ($q = 0$). (b) 1 norm ν-SP-SVM ($q = 0$). (c) 2 norm μ-SP-SVM ($q = 2$). (d) 1 norm ν-SP-SVM ($q = 2$). (e) 2 norm ν-SP-SVM ($q = 4$). (f) 1 norm ν-SP-SVM ($q = 4$). (g) 2 norm ν-SP-SVM ($q = 8$). (h) 1 norm ν-SP-SVM ($q = 8$). (i) 2 norm ν-SP-SVM ($q = 12$). (j) 1 norm ν-SP-SVM ($q = 12$). (k) 2 norm ν-SP-SVM ($q = 16$). (l) 1 norm ν-SP-SVM ($q = 16$).

different SVM free parameters (C, C', and ν) have been optimized using the validation set.

The MATLAB code that implements the proposed ν-SP-SVM algorithms and a demo, which allows us to replicate the results shown in this section can be downloaded from http://www.tsc.uc3m.es/ hmolina/paper_nu-SP-SVM/.

Table I presents the averaged CE rates achieved by the different SVM methods under study and the number of features in their models. These results show the following.

1) Classical SVM methods rise the CE rate and the number of features in the model when q is increased, as it is expected, standard 1-norm SVM and Dr-SVM provide

sparser solutions than standard 2-norm SVM, even if some noisy features are included in the final model. Note that Dr-SVM, which penalizes with L1 and L2 norms, retains more useless features than 1-norm SVM and, although its performance improves 2-norm SVM, it is not as accurate as 1-norm SVM.

2) The proposed ν-SP-SVM approaches keep the classification error rates around 7%, independently of q and, in most cases, they only employ the useful features: note that the average number of selected features is always very close to 2. However, standard 2-norm SVM uses all original features and standard 1-norm SVM and Dr-SVM tend to include some useless features.

3) When 2-norm and 1-norm ν-SP-SVM results are compared to each other, we do not observe relevant differences, since they present similar CEs and similar number of features.

Fig. 1 depicts the evolution of the averaged classification error and the averaged number of selected features as a function of parameter ν in the orange problem, for each value of ν, parameters C and C' have been adjusted by the validation process. A dotted vertical line indicates the working point of the results from Table I, when ν was also selected in the validation process. Additionally, this figure includes the averaged CE rate, which could be achieved by retraining a new standard SVM with the set of features selected by ν-SP-SVMs. This figure shows the following behaviors of the proposed methods.

1) As it was expected, ν plays a crucial role to obtain a reduced number of features and an accurate solution. Fixing $\nu = 1$, the provided results would be similar to the standard 1-norm SVM, however, reducing ν both performance improvements and reductions in the number of model parameters could be achieved, mainly if ν was close to $2/d$.

2) The role of ν as upper bound on the number of selected features is clearly seen. When ν is close to 1, the proposed ν-SP-SVM methods do not include all original features in their models, since most noisy features are removed. For instance, when $q = 8$, 12, or 16, there are 65, 119, and 189 original features, but ν-SP-SVMs employ less than 10, 12, or 14 features.

3) Finally, it is important to point out that the model performance is not degraded by pruning the coefficients associated to irrelevant features (those whose slack variables γ_i are zero). If we compare the solutions provided by ν-SP-SVM models with a new standard SVM trained with the selected set of features, slight performance improvements could be achieved; but, when any noisy feature is included in the model, the retrained SVM tends to overfit, whereas proposed ν-SP-SVM models provide accurate solutions.

B. Benchmark Data Sets

To test the performance of the proposed ν-SP-SVM classifiers over real data sets, 8 benchmark binary classification problems have been selected from the universal communications identifier (UCI) repository [38]: *Abalone*, *Credit*, *Hand*, *Image*, *Ionosphere*, *Pima*, *Spam*, and Wisconsin Diagnostic Breast Cancer (*Wdbc*). These problems have been chosen because of their diversity in the number of data and dimensions. The main characteristics of these problems are summarized in Table II. To adjust the free parameters of the different models, the parameter ranges described in the introduction of the experimental section have been swept by applying a five-fold CV process.

TABLE II
CHARACTERISTICS OF THE BINARY DATA SETS: NUMBER OF FEATURES AND NUMBER OF DATA BELONGING TO EACH CLASS IN TRAINING AND TEST SETS

Problem	# Features (d)	# Train samples (n_1/n_{-1})	# Test samples (n_1/n_{-1})
Abalone	8	1238/1269	843/827
Credit	15	215/268	92/115
Hand	62	1923/1900	906/891
Image	18	821/1027	169/293
Ionosphere	34	150/84	75/42
Pima	8	188/350	80/150
Spam	57	1218/1847	595/941
Wdbc	30	238/141	119/71

For this benchmark analysis we have also included, as an additional reference method, the RFE method from [39]. This algorithm carries out a feature selection process by iteratively removing the feature with less weight in the SVM solution. To fairly compare this method with proposed ν-SP-SVM methods, we have implemented the linear version of the RFE algorithm, additionally, the final feature subset of the RFE method is selected with a CV process (note that the RFE method obtains a different feature subset in each iteration) and a new SVM has been trained using only the selected features.

Table III shows the results achieved by the different SVM algorithms under study averaged over 50 runs with randomly selected training/validation sets. As it can be observed, standard 1-norm SVM fails to remove irrelevant features in some problems. For instance, in *Abalone*, *Pima*, and *Spam* almost all original features are retained. Dr-SVM is worse than the standard 1-norm SVM in this regard, and hardly removes any feature in the considered problems (with the exception of *Credit*).

In contrast, it is possible to perform effective feature selection with the proposed ν-SP-SVMs without incurring in any significant degradation in classification performance. In particular, Table III shows a 25% model complexity reduction in *Image*, *Spam*, and *Wdbc* when ν-SP-SVM, as opposed to its standard counterpart, is used. This percentage is even better for other problems, reaching 33.3% in *Abalone* and *Hand* and 50% in *Ionosphere*.

When we compare the proposed ν-SP-SVM approaches with the RFE method, we observe that the automatic feature selection carried out by our proposals is competitive with standard feature selection procedures which have to, first, select the feature subset and, second, train the classifier. According to Table III, results are quite similar for most problems. However,

TABLE III
CE AND NUMBER OF SELECTED FEATURES PROVIDED BY STANDARD 2 AND 1-NORM SVMs, DR-SVM, THE RFE METHOD AND THE 2 AND 1-NORM ν-SP-SVMs IN THE BINARY CLASSIFICATION PROBLEMS

		Standard SVM		Dr-SVM	RFE	ν-SP-SVM	
		2-norm	1-norm			2-norm	1-norm
Abalone	CE	21.10(±0.89)	20.51(±0.11)	20.60(±0.14)	20.90(±0.58)	20.90(±0.37)	20.85(±0.34)
	# feat.	8.00(±0.00)	7.96(±0.20)	8.00(±0.00)	4.34(±2.18)	5.36(±2.11)	5.80(±1.87)
Credit	CE	10.65(±0.10)	11.07(±0.13)	11.07(±0.13)	10.99(±0.21)	10.68(±0.15)	11.02(±0.19)
	# feat.	15.00(±0.00)	1.16(±0.55)	2.08(±3.36)	4.32(±4.83)	7.16(±3.15)	1.36(±0.78)
Hand	CE	9.17(±0.18)	9.24(±0.10)	9.20(±0.12)	9.43(±0.22)	9.15(±0.22)	9.29(±0.21)
	# feat.	62.00(±0.00)	55.68(±4.20)	55.56(±4.08)	34.82(±6.04)	45.72(±4.96)	42.06(±5.67)
Image	CE	14.94(±0.95)	12.94(±0.18)	13.11(±0.23)	14.05(±1.07)	13.18(±0.43)	12.98(±0.19)
	# feat.	18.00(±0.00)	13.96(±0.20)	17.24(±0.77)	16.06(±1.49)	14.38(±2.58)	13.52(±1.03)
Ionosphere	CE	11.93(±2.02)	11.73(±2.35)	12.38(±0.85)	13.76(±2.12)	11.79(±1.92)	12.27(±1.08)
	# feat.	33.00(±0.00)	24.42(±7.47)	30.92(±3.29)	13.96(±5.13)	18.32(±6.55)	17.44(±3.90)
Pima	CE	23.63(±0.71)	23.29(±0.22)	23.35(±0.31)	23.78(±1.03)	23.36(±0.33)	23.00(±0.20)
	# feat.	8.00(±0.00)	7.44(±0.50)	7.76(±0.43)	5.26(±2.04)	6.34(±1.14)	6.72(±1.05)
Spam	CE	6.88(±0.17)	7.15(±0.09)	7.03(±0.06)	6.78(±0.21)	6.99(±0.24)	7.09(±0.15)
	# feat.	57.00(±0.00)	54.52(±1.79)	56.22(±0.79)	44.68(±3.03)	44.88(±3.21)	42.88(±3.28)
Wdbc	CE	2.97(±0.92)	4.31(±0.68)	3.19(±0.51)	3.43(±0.57)	3.28(±0.53)	3.77(±0.75)
	# feat.	30.00(±0.00)	18.52(±3.25)	27.38(±3.17)	21.80(±3.59)	22.64(±2.27)	13.80(±2.70)

in the case of *Image*, both ν-SP-SVM proposals outperform the RFE method, and for *Credit* and *Wdbc*, the 1-norm ν-SP-SVM approach achieves the best accuracy-complexity trade-off. On the other hand, in problems such as *Ionosphere* or *Hand*, RFE presents a lower number of features, although this advantage is achieved at the expense of a CE increase.

Figs. 2 and 3 show the evolution of the classification error and the number of selected features as a function of ν in the different data sets. A dashed line depicts the CE achieved by new standard SVMs retrained with the set of features selected by the proposed ν-SP-SVM models and a dotted vertical line points out the ν value selected in the validation process. These figures remark the clear trade-off between the model complexity and the final CE. In problems such as *Credit*, *Image*, *Ionosphere*, and *Wdbc*, when the 1-norm ν-SP-SVM is applied, we could directly have fixed $\nu = 1$, and most useless features would have been removed. However, an adequate selection of ν is crucial to obtain an accurate solution. The validation process has carried out a conservative selection of parameter ν, if, during the validation process, a slight performance degradation had been allowed, a additional features would have been removed, in fact, for all the problems under study but *Credit*, lower values of ν would have resulted in a lower number of features, while keeping similar error rates. Finally, it is important to note that the retraining procedure does not show any clear improvement, since although in some cases the final CE is slightly improved, in other cases it is similar or, even, slightly worse.

C. High Dimensional Datasets

The aim of this section is to test the performance of the proposed methods when we are dealing with a large number of input features. For this purpose, the Dexter dataset [40] has been considered. The goal of this problem is to classify texts about "corporate acquisitions" into two categories. The data set has 20 000 features, from which 9947 variables correspond to a "bag-of-words" representation of several texts and the remaining 10 053 features are noisy features added to complicate the classification task. The different data set partitions are balanced with 300 training data, 300 validation patterns and 2000 test samples.

Due to the large number of input features, the CV of all possible ν values in the ν-SP-SVM methods is not reasonable. For this reason, we have followed this strategy.

1) We have first trained the proposed methods with $\nu = 1$, what provides a first approximation to the number of useful features. In this case, 1-norm ν-SP-SVM achieves a $CE = 8.1\%$ with only 150 features and 2-norm ν-SP-SVM a $CE = 6\%$ with 3976 variables.
2) According to above number of selected features, the maximum value of ν, worthy of being explored, has been fixed. For instance, in 1-norm ν-SP-SVM this value has been fixed to 0.01 (150 is less than the 1% of 20 000) and in 2-norm ν-SP-SVM has been set to 0.2 (3976 is close to the 20% of 20 000).
3) Then, a range of 10 linearly spaced ν values has been defined. In particular, ranges $\{0.1\%, 0.2\%, \ldots, 1\%\}$ and $\{2\%, 4\%, \ldots, 20\%\}$ have been explored by each ν-SP-SVM model.
4) Finally, the optimum ν value has been selected as the one with minimum validation error.

As a result of this procedure, 1-norm ν-SP-SVM has selected a ν value of 0.004, achieving a $CE = 7.75\%$ with only 79 features, whereas 2-norm ν-SP-SVM has used a final ν value of 0.1 providing a CE of 6.4% with 1487 features. Reference methods, 2-norm, 1-norm, and Dr-SVMs, have

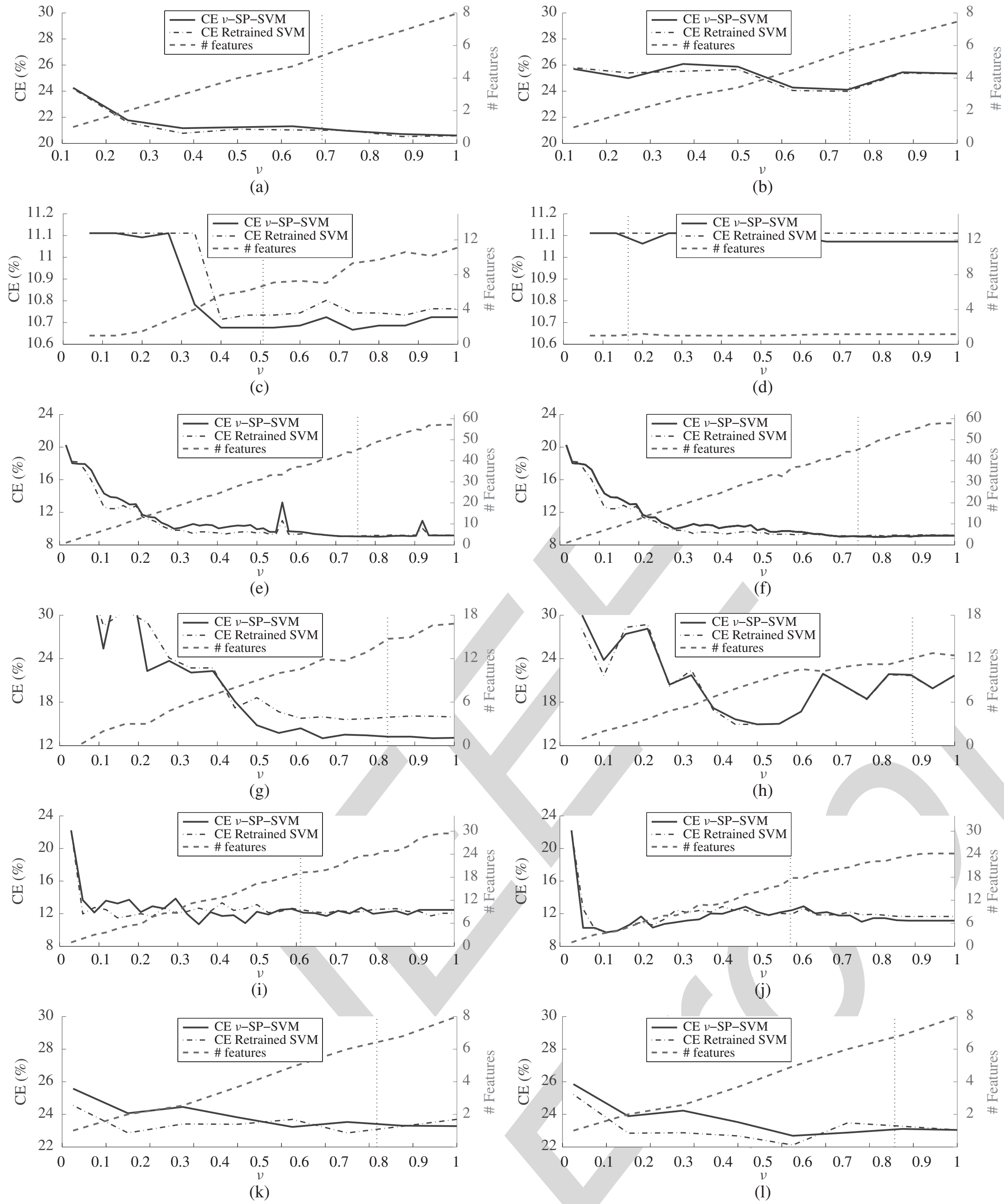


Fig. 2. Evolution of CE and the number of selected features in ν-SP-SVMs as a function of ν for data sets: *Abalone*, *Credit*, *Hand*, *Image Ionosphere*, and *Pima*. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Abalone*. (b) 1-norm ν-SP-SVM *Abalone*. (c) 2-norm ν-SP-SVM *Credit*. (d) 1-norm ν-SP-SVM *Credit*. (e) 2-norm ν-SP-SVM *Hand*. (f) 1-norm ν-SP-SVM *Hand*. (g) 2-norm ν-SP-SVM *Image*. (h) 1-norm ν-SP-SVM *Image*. (i) 2-norm ν-SP-SVM *Ionosphere*. (j) 1-norm ν-SP-SVM *Ionosphere*. (k) 2-norm ν-SP-SVM *Pima*. (l) 1-norm ν-SP-SVM *Pima*.

presented CEs of 6.45%, 8.10% and 6.05%, respectively, and they have used 7142, 159, and 5750 features (see Table IV).

These results show that 1-norm ν-SP-SVM outperforms standard 1-norm SVM by achieving a lower CE with half the number of features. Regarding 2-norm ν-SP-SVM and standard 2-norm SVM, they present similar error rates, but the latter is using 35% of the features instead of 7.43% used by 2-norm ν-SP-SVM. Finally, Dr-SVM provides the lowest CE, but the number of selected features (5750) is much higher than the 1487 of the 2-norm ν-SP-SVM.

Besides, it is important to point out that 1-norm-based algorithms (standard 1-norm SVM and 1-norm ν-SP-SVM)

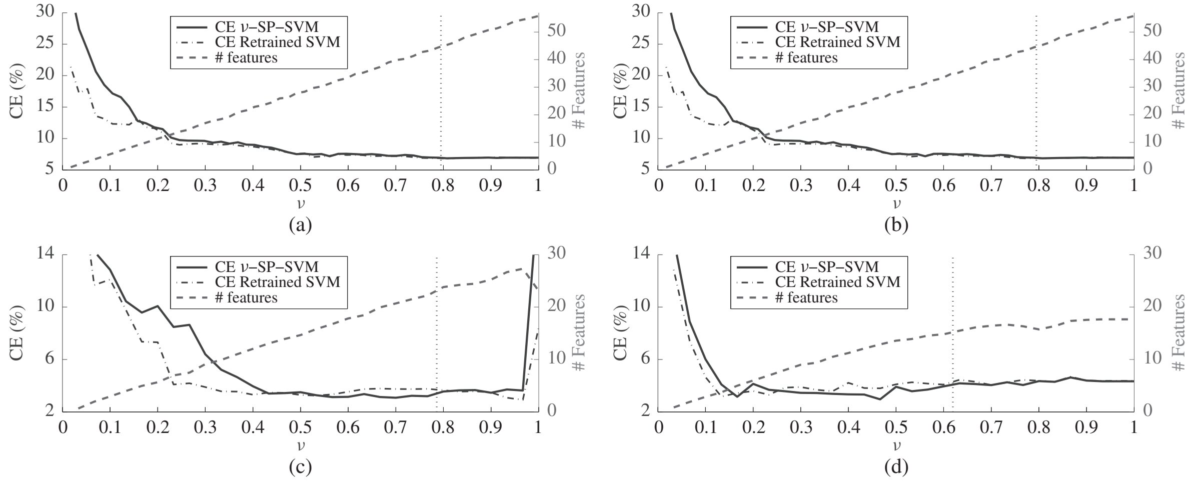


Fig. 3. Evolution of CE and the number of selected features in ν-SP-SVMs as a function of ν for data sets: *Spam* and *Wdbc*. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Spam*. (b) 1-norm ν-SP-SVM *Spam*. (c) 2-norm ν-SP-SVM *Wdbc*. (d) 1-norm ν-SP-SVM *Wdbc*.

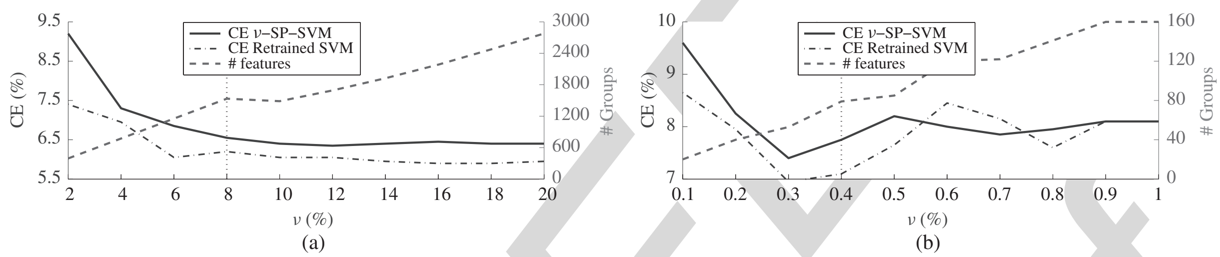


Fig. 4. CE and the number of selected features in ν-SP-SVM algorithms as a function of ν in *Dexter* data set. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM. (b) 1-norm ν-SP-SVM.

AQ:1

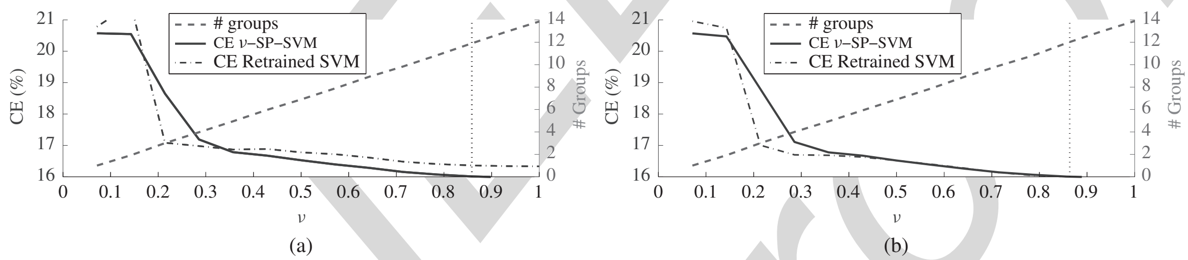


Fig. 5. Evolution of CE and the number of selected features in ν-SP-SVMs as a function of ν for data sets: *Spam* and *Wdbc*. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model. Dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Spam*. (b) 1-norm ν-SP-SVM *Spam*. (c) 2-norm ν-SP-SVM *Wdbc*. (d) 1-norm ν-SP-SVM *Wdbc*.

have selected a few number of features, prompting a performance degradation. This effect is due to the fact that the maximum number of features that can be selected is always upper bounded by the number of training data [30], [32]. For this reason, these approaches are working with few hundreds of features instead of selecting thousands as the 2-norm-based methods.

Finally, Fig. 4 shows the evolution of the CE and the number of features in the model for the explored range of ν values. At first glance, it can be seen that, in the explored range of ν, values larger than 8% in 2-norm ν SP-SVM and 0.3% for 1-norm ν SP-SVM are able to provide accurate results with a low number of features, even lower than 1-norm, 2-norm, and Dr-SVM methods. This figure also shows the CE achieved when the SVM is retrained with the selected set of features, suggesting that, in problems where the number of removed features is high, the retraining process is able to provide an additional advantage in terms of CE reduction.

TABLE IV
CE AND NUMBER OF SELECTED FEATURES PROVIDED BY DIFFERENT METHODS UNDER STUDY IN DEXTER DATA SETS

		Standard SVM		Dr-SVM	ν-SP-SVM	
		2-norm	1-norm		2-norm	1-norm
Dexter	CE	6.45	8.10	6.05	6.4	7.75
	# feat.	7142	159	5750	1487	79

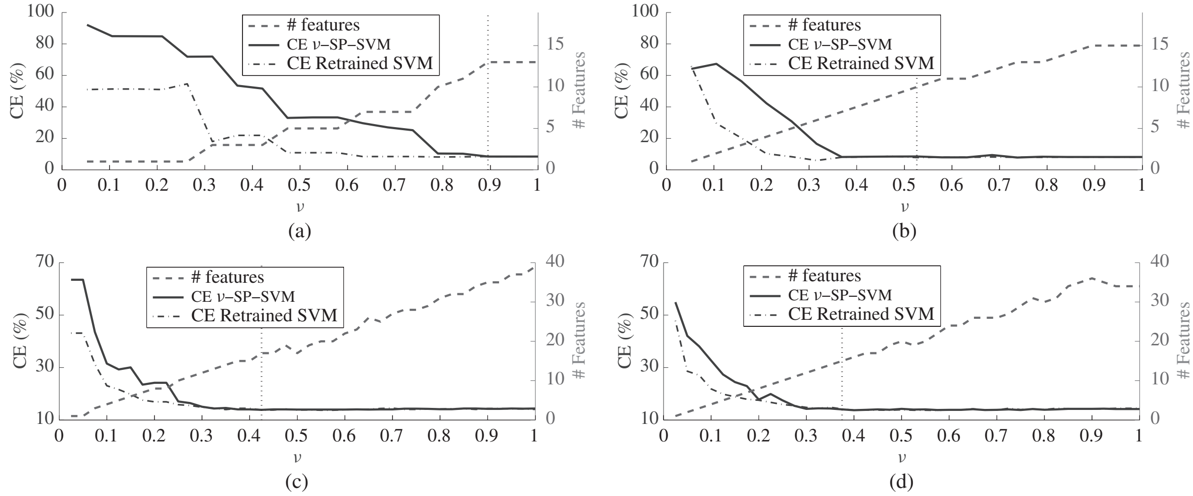


Fig. 6. Evolution of the CE and the number of selected features in ν-SP-SVM algorithms as a function of ν in multiclass problems. Dash-dotted line shows the CE of an SVM which has been retrained with the features selected by ν-SP-SVM model and dotted vertical line marks the cross-validated ν value. (a) 2-norm ν-SP-SVM *Segmentation*. (b) 1-norm ν-SP-SVM *Segmentation*. (c) 2-norm ν-SP-SVM *Wave*. (d) 1-norm ν-SP-SVM *Wave*.

TABLE V

PREDEFINED FEATURE GROUPS IN THE PROBLEM ADULT. CATEGORICAL FEATURES ARE CODIFIED WITH DUMMY VARIABLES

# group	Original feature	Categorical / continous	# of categories	# of features in each group
1	age	continuous	–	1
2	workclass	categorical	8	3
3	fnlwgt	continuous	–	1
4	education	categorical	16	4
5	education-num	continuous	–	1
6	marital-status	categorical	7	3
7	occupation	categorical	14	4
8	relationship	categorical	6	3
9	race	categorical	5	3
10	sex	categorical	2	1
11	capital-gain	continuous	–	1
12	capital-loss	continuous	–	1
13	hours-per-week	continuous	–	1
14	native-country	categorical	41	6

D. Selecting Feature Groups with ν-SP-SVM

To analyze the performance of the proposed methods when features need to be selected according to predefined sets, instead of selecting isolated features, we have chosen the dataset Adult from [38]. The aim of this problem is to determine whether a person earns over 50K a year from several demographic characteristics from 14 original features, of which six are continuous and eight are categorical. Each categorical feature has been coded with dummy variables, using N indicatrix variables (0 or 1) to codify their 2^N possible values, in this way, each data is finally represented by 33 features belonging to 14 groups as it is described in Table V. Then, when a group selection approach is applied, the dummy variables representing to the same categorical feature will be either all selected or all removed from the final model. Note that only when all variables from a certain group are removed it is possible to skip the capture of the associated categorical variable.

This binary data set has 30 162 training samples and 15 060 data to test the model. To train the different SVMs, we have randomly selected a 10% of the original training data set, therefore, 3016 data have been used to train the different methods. A 5-fold CV process has been applied to adjust the free parameters of the different methods and their performances have been evaluated over whole test data. The different SVMs have been trained 100 times, with different randomly selected training data, and their averaged results have been studied.

As result, standard 2 and 1-norm SVMs present an averaged CE of 16.33($\pm$0.3)% and 15.97($\pm$0.2)% employing 14 and 13.9 ± 0.3 groups, respectively, whereas Dr-SVM presents the same performance (both in CE and number of selected features) as 1-norm SVMs. This result is a consequence of standard 2-norm SVM having selected all groups and 1-norm SVM and Dr-SVM having seldom discarded group 10, this group is associated to original feature *sex* and codified with only one dummy variable.

To compare these results with the proposed methods, Fig. 5 depicts the values of the CE and the number of selected groups as a function of parameter ν in ν-SP-SVMs. It can be seen that if ν is cross validated (see dotted vertical line), ν-SP-SVMs present CE close to 16% with 12 groups, since groups 3 and 10 are usually removed. However, if we had wanted to select a lower number of groups, ν could have been fixed around 0.3, keeping the CE lower than 17% and selecting just the 4 most relevant groups: Groups associated to original features *education-num*, *relationship*, and *capital-gain* are always chosen and additionally, either group 4 (*education*) or 7 (*occupation*) is included in the model. Thus, this example illustrates the convenience of the ν formulation of SP-SVM for allowing a more flexible selection of the number of variables to be incorporated in the model.

Again, a retraining process (dash-dotted line in Fig. 5) provides a small improvement, since for most ν values, ν-SP-SVMs, and retrained SVMs achieve similar CEs.

TABLE VI
CE AND NUMBER OF SELECTED FEATURES PROVIDED BY DIFFERENT METHODS UNDER STUDY IN MULTICLASS DATA SETS

		Classical SVMs		Dr-SVM	Sparse SVMs	
		2-norm	1-norm		2-norm	1-norm
Segmentation	CE	9.05	9.00	8.24	8.43	8.52
	# feat.	18.00	13.00	15	13.00	10.00
Wave	CE	13.87	14.33	14.20	13.87	14.07
	# feat.	40.00	38.00	30.00	17.00	15.00

E. Multiclass Problems

In this section, we will test the performance of the ν-SP-SVMs over multiclass datasets *Segmentation* and *Wave* from the UCI repository [38]. The purpose of *Segmentation* problem is to classify hand-segmented images represented by 19 features in 7 categories: *brickface*, *sky*, *foliage*, *cement*, *window*, *path*, and *grass*. The data set has 210 and 2100 training and test data, respectively. *Wave* problem consists of 3 classes of waves to be identified from 40 features, whose latter 19 ones are all noise, the data set has 3500 training samples and 1500 test data. As in the previous sections, the free parameters of the different methods have been adjusted with a 5 fold CV process.

To train the different classifiers, proposed ν-SP-SVM methods have solved problem (10), either in its 2-norm or in its 1-norm version, whereas reference methods have directly used the multiclass problem defined by (9) with their corresponding penalization terms. Table VI presents the results achieved by both standard and proposed SVMs. As it can be observed, ν-SP-SVMs achieve lower error rates with lower number of features. In *Segmentation*, CE is reduced in a 0.5%, with respect to 1-norm and 2-norm SVMs, using only 13 and 10 features, whereas Dr-SVM achieves a slightly lower CE using 15 features. In *Wave*, the advantages of the proposed SVM classifiers are clearer, since the number of features in the model is half the number for the reference methods and the CE is similar in the 2-norm models, slightly reduced in the 1-norm methods and Dr-SVMs are outperformed by both ν-SP-SVMs.

When the evolution of CE and the number of features are analyzed as a function of ν (see Fig. 6), the trade-off between these parameters is again observed. Besides, retrained SVMs provide a significant CE reduction in *Segmentation* problem.

V. CONCLUSION

This paper introduced a method for feature selection based on a new formulation of linear SVMs that includes constraints additional to the classical ones. These constraints drop the weights associated to those features that are likely to be irrelevant. In order to predefine an upper bound for the number of relevant features, a ν-SVM formulation has been used, where ν is a parameter that indicates the fraction of features to be considered. This parameter is swept in an efficient way in order to find the optimal number of features over a validation set of data. This paper presented two versions of the formulation, the first one being an SVM with a 2-norm regularization term. The second one uses a 1-norm regularization, that has a reduced computational burden with respect to the first one. Besides, this new SVM formulation allows us to easily apply the feature selection process over predefined feature sets. This, in turn, is useful to introduce a straightforward, yet efficient way to extend the algorithms to multiclass problems.

Experiments showed that the introduced methods present advantages not only in terms of CE, but also in the ability of reducing the model complexity by adequately removing features during the training process, not as a preprocessing stage. Also, these experiments showed that the algorithms are efficient when applied to the task of feature group selection and to multiclass problems.

Future research includes nonlinear versions of the algorithm in order to take into account the nonlinear relationships between features. Applications can also include extensions to regression problems as well as linear model selection for signal processing tasks, such as filter design or plant modeling, in situations where optimal models are known to be sparse.

REFERENCES

[1] B. Schölkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. Cambridge, MA: MIT Press, 2001.

[2] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[3] D. J. Sebald and J. A. Bucklew, "Support vector machine techniques for nonlinear equalization," *IEEE Trans. Signal Process.*, vol. 48, no. 11, pp. 3217–3226, Nov. 2000.

[4] M. M. Ramon, N. Xu, and C. Christodoulou, "Beamforming using support vector machines," *IEEE Antennas Wireless Propag. Lett.*, vol. 4, pp. 439–442, 2005. AQ:2

[5] Z. Shi and M. Han, "Support vector echo-state machine for chaotic time-series prediction," *IEEE Trans. Neural Netw.*, vol. 18, no. 2, pp. 359–372, Mar. 2007.

[6] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*. New York: Springer-Verlag, 2001.

[7] Y. Lin, "Support vector machines and the Bayes rule in classification," *Data Min. Knowl. Discov.*, vol. 6, no. 3, pp. 259–275, 2002.

[8] S. Rosset, J. Zhu, and T. Hastie, "Margin maximizing loss functions," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, pp. 1237–1246.

[9] J. Friedman, T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "Consistency in boosting: Discussion," *Ann. Stat.*, vol. 32, no. 1, pp. 102–107, Feb. 2004.

[10] H. Liu and H. Motoda, *Feature Selection for Knowledge Discover and Data Mining*. Norwell, MA: Kluwer, 1998.

[11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, nos. 7–8, pp. 1157–1182, Oct.–Nov. 2003.

[12] L. Yu and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," in *Proc. 20th Int. Conf. Mach. Learn.*, Washington D.C., 2003, pp. 1–8.

[13] R. Kohavi and G. John, "Wrappers for feature selection," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 273–324, 1997.

[14] L. Wang, N. Zhou, and F. Chu, "A general wrapper approach to selection of class-dependent features," *IEEE Trans. Neural Netw.*, vol. 19, no. 7, pp. 1267–1278, Jul. 2008.

[15] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Royal Stat. Soc., Ser. B*, vol. 58, no. 1, pp. 267–288, 1994.

[16] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for SVMs," in *Advances in Neural Information Processing Systems 13*, T. L. T. Dietterich and V. Tresp, Eds. Cambridge, MA: MIT Press, 2000.

[17] J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song, "Dimensionality reduction via sparse support vector machines," *J. Mach. Learn. Res.*, vol. 3, pp. 1229–1243, Mar. 2003.

[18] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[19] A. Rakotomamonjy, "Variable selection using SVM based criteria," *J. Mach. Learn. Res.*, vol. 3, nos. 7–8, pp. 1357–1370, 2003.

[20] J. Weston, F. Perez-Cruz, O. Bousquet, O. Chapelle, A. Elisseeff, and B. Scholkopf, "Feature selection and transduction for prediction of molecular bioactivity for drug design," *Bioinformatics*, vol. 19, no. 6, pp. 764–771, 2003.

[21] Y. Aksu, D. J. Miller, G. Kesidis, and Q. X. Yang, "Margin-maximizing feature elimination methods for linear and nonlinear kernel-based discriminant functions," *IEEE Trans. Neural Netw.*, vol. 21, no. 5, pp. 701–717, May 2010.

[22] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical LASSO," *Biostat*, vol. 9, no. 3, pp. 432–441, 2008.

[23] C. Z. J. Huang and S. Ma, "Adaptive LASSO for sparse high-dimensional regression models," *Stat. Sinica*, vol. 18, no. 374, pp. 1603–1618, 2008.

[24] N. Meinshausen and B. Yu, "LASSO-type recovery of sparse representations for high-dimensional data," *Ann. Stat.*, vol. 37, no. 1, pp. 246–270, 2009.

[25] Y. Li, P. Namburi, Z. Yu, C. Guan, J. Feng, and Z. Gu, "Voxel selection in fMRI data analysis based on sparse representation," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 10, pp. 2439–2451, Oct. 2009.

[26] P. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 82–90.

[27] L. R. Grate, C. Bhattacharyya, M. I. Jordan, and I. S. Mian, "Simultaneous relevant feature identification and classification in high-dimensional spaces," in *Algorithms in Bioinformatics* (Lecture Notes in Computer Science), vol. 2452. New York: Springer-Verlag, 2002, pp. 1–9.

[28] G. M. Fung and O. L. Mangasarian, "A feature selection Newton method for support vector machine classification," *Comput. Optim. Appl.*, vol. 28, no. 2, pp. 185–202, 2004.

[29] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani, "1-norm support vector machines," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, pp. 49–56.

[30] L. Wang, J. Zhu, and H. Zou, "The doubly regularized support vector machine," *Stat. Sinica*, vol. 16, pp. 589–615, 2006. AQ:3

[31] J. Zhu and H. Zou, "Variable selection for the linear support vector machine," in *Trends in Neural Computation* (Studies in Computational Intelligence), vol. 35, K. Chen and L. Wang, Eds. Berlin, Germany: Springer-Verlag, 2007, pp. 35–59.

[32] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. Royal Stat. Soc.*, vol. 67, no. 2, pp. 301–320, Apr. 2005.

[33] H. Zou, "An improved 1-norm support vector machine for simultaneous classification and variable selection," in *Proc. 11th Int. Conf. Artif. Intell. Stat.*, 2007, pp. 1–7.

[34] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *J. Royal Stat. Soc.: Ser. B (Stat. Methodol.)*, vol. 68, no. 1, pp. 49–67, Feb. 2006.

[35] H. Zou and M. Yuan, "The F_∞-norm support vector machine," *Stat. Sinica*, vol. 18, pp. 379–398, 2008. AQ:4

[36] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural Comput.*, vol. 12, no. 5, pp. 1207–1245, May 2000.

[37] J. Arenas-García and F. Pérez-Cruz, "Multi-class support vector machines: A new approach," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, vol. 2. Hong Kong, Apr. 2003, pp. 781–784.

[38] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository* [Online]. Available: http://archive.ics.uci.edu/ml

[39] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[40] I. Guyon, "Feature selection challenge," in *Proc. Neural Inf. Process. Syst. Workshop Feature Extract.*, Dec. 2003, pp. 1–8.

Vanessa Gómez-Verdejo was born in Madrid, Spain, in 1979. She received the telecommunication engineering degree from the Universidad Politécnica de Madrid, Madrid, in 2002. She received the Ph.D. degree from the Universidad Carlos III de Madrid, Madrid, in 2007.

She is currently a Visiting Professor in the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. She has co-authored around 20 papers, including journal and conference contributions. She has participated in several research and development projects with public funding and companies, which have provided her with an extensive experience in solving real-world problems. Her current research interests include machine learning algorithms and methods of feature selection for support vector machine classifiers.

Manel Martínez-Ramón (M'00–SM'04) received the degree in telecommunications engineering from the Universitat Politécnica de Catalunya, Barcelona, Spain, in 1994, and the Ph.D. degree in telecommunications engineering from the Universidad Carlos III de Madrid, Madrid, Spain, in 1999. AQ:5

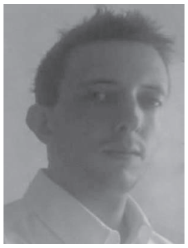
He is with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. He has co-authored more than 20 papers in international journals and 40 conference papers on his areas of expertise. He has written a book on applications of support vector machines to antennas and electromagnetics and co-authored several book chapters. His current research interests include applications of the statistical learning to signal processing with emphasis in communications and brain imaging.

Jerónimo Arenas-García (S'00–M'04) received the degree (with honors) in telecommunication engineering from the Universidad Politécnica de Madrid, Madrid, Spain, in 2000, and the Ph.D. degree (with honors) in telecommunication technologies from the Universidad Carlos III de Madrid, Madrid, in 2004. AQ:6

He held a post-doctoral position at the Technical University of Denmark, Lyngby, Denmark, before he returned to the Universidad Carlos III de Madrid, where he is currently an Associate Professor of digital signal and information processing with the Department of Signal Theory and Communications. His current research interests include statistical learning, particularly in adaptive algorithms, advanced machine learning techniques, and their applications, for instance in remote sensing data, and multimedia information processing.

Dr. Arenas-García is a current member of the IEEE Machine Learning for Signal Processing Technical Committee.

AQ:7



Miguel Lázaro-Gredilla (M'11) received the degree (with honors) in telecommunication engineering from the University of Cantabria, Santander, Spain, in 2004, and the Ph.D. degree (with honors) from the University Carlos III de Madrid, Madrid, Spain, in 2010, where he has taught several undergraduate courses.

He is currently a Research Associate at the University of Cantabria, after two short stays at the University of Cambridge, Cambridge, U.K., and University of Manchester, Manchester, U.K. His current research interests include Gaussian processes and Bayesian models.

AQ:8

Harold Molina-Bulla (S'90–M'99) received the degree in electronic engineering from Pontificia Universidad Javeriana, Bogotá, Colombia, in 1994, and the Ph.D. degree (with honors) in telecommunication technologies from the Universidad Carlos III de Madrid, Madrid, Spain, in 1999.

He is currently a Visiting Professor of electronic communications with the Department of Signal Theory and Communications, Universidad Carlos III de Madrid. His current research interests include high performance computing for signal processing, advanced machine learning techniques, and their applications.

EDITOR QUERY

EQ:1 = Please provide the accepted date for this article.

AUTHOR QUERIES

AQ:1	=	Please provide the images for Fig. 5(c) and (d).
AQ:2	=	Please provide the issue no and publication month for the IEEE ref. [4].
AQ:3 & 4	=	Please provide the issue no or publication month for the refs. [30] and [35].
AQ:5, 6, 7, & 8	=	Please specify the degree details.