Intelligent Data Analysis 19 (2015) S117–S133
DOI 10.3233/IDA-150773
IOS Press

Robust feature selection for multiclass Support Vector Machines using second-order cone programming

Julio López[a] and Sebastián Maldonado[b,*]
[a]*Facultad de Ingeniería, Universidad Diego Portales, Ejército 441, Santiago, Chile*
[b]*Universidad de los Andes, Las Condes, Santiago, Chile*

Abstract. This work addresses the issue of high dimensionality for linear multiclass Support Vector Machines (SVMs) using second-order cone programming (SOCP) formulations. These formulations provide a robust and efficient framework for classification, while an adequate feature selection process may improve predictive performance. We extend the ideas of SOCP-SVM from binary to multiclass classification, while a sequential backward elimination algorithm is proposed for variable selection, defining a contribution measure to determine the feature relevance. Experimental results with multiclass microarray datasets demonstrate the effectiveness of a low-dimensional data representation in terms of performance.

Keywords: Feature selection, multiclass classification, second-order cone programming, Support Vector Machines

1. Introduction

A Support Vector Machine (SVM) is one of the standard tools for machine learning and classification. Based on the structural risk minimization principle [29], SVM for binary classification attempts to find the separating hyperplane which has the greatest distance to the nearest training data point of each class. For multiclass classification, a series of binary classifiers can be constructed, or the problem can also be tackled directly by solving a single multiclass SVM [11,33]. SVM has proved to be very effective in business analytics applications, such as churn prediction [30] and credit scoring [10]. For the latter case, multiclass SVM can be used to deal with two types of defaulters: those who cannot pay because of cash flow problems, and those that lack of willingness to pay [10].

Second-order cone programming (SOCP) formulations have been proposed as an alternative optimization scheme for SVMs [2,6,23], providing robust SVM classifiers. The formulation correctly classifies objects belonging to a given class up to a rate η, even for the worst-case distribution of the data, using the information of the mean and covariance of the training patterns. While SOCP-SVM has been applied successfully for binary classification, it has not yet been formalized for multiclass classification in this context, to the best of our knowledge. The only reference appearing in the literature in the context of SOCP for multiclass classification is Zhong and Fukushima [36]. This method studies the problem of

*Corresponding author: Sebastián Maldonado, Universidad de los Andes, Mons. Álvaro del Portillo 12455, Las Condes, Santiago, Chile. Tel.: +56 2 26181874; E-mail: smaldonado@uandes.cl.

classification with noisy data (i.e. instances with measurement errors), which is a completely different approach from the ones reported in this paper.

A plethora of feature selection methods has been proposed for binary SVM, but only a few have been extended to multiclass classification. This work uses SOCP formulations for multiclass SVMs to assess the relevance of the attributes, proposing a backward elimination algorithm to reduce the dimensionality of the problem and improve classification performance.

The paper is structured as follows. Section 2 presents the SOCP-SVM formulation for binary classification, which we extend to multiclass in our proposal. Section 3 introduces Support Vector Machines for multiclass classification. Recent developments for multiclass feature selection using SVMs are reviewed in Section 4. The proposed feature selection approach is presented in Section 5. Section 6 provides experimental results using real-world datasets. A summary of this paper can be found in Section 7, where we also provide its main conclusions and address future developments.

2. Second order cone programming SVMs

Let us consider a set of tuples $(\mathbf{x}_i, y_i)$ of training points $\mathbf{x}_i \in \Re^{|\mathcal{S}|}$, where $\mathcal{S}$ represents the full set of variables and $|\mathcal{S}|$ its cardinality, and their respective labels $y_i \in \{-1, +1\}$, $i = 1, \ldots, m$. Suppose that $\mathbf{X}_1$ and $\mathbf{X}_2$ are random vectors that generate the samples of the positive and negative classes respectively, with means and covariance matrices given by $(\boldsymbol{\mu}_i, \Sigma_i)$ for $i = 1, 2$, where $\Sigma_i \in \Re^{|\mathcal{S}|\times|\mathcal{S}|}$ are symmetric positive semidefinite matrices. Let us denote a family of distributions which have a common mean and covariance by $\mathbf{X} \sim (\boldsymbol{\mu}, \Sigma)$.

In order to construct a maximum margin linear classifier, such that the probability of false-negative and false-positive errors does not exceed $1 - \eta_1$ and $1 - \eta_2$ respectively, with $\eta_1,\ \eta_2 \in (0, 1)$, Nath and Bhattacharyya [23] suggested considering the following quadratic chance-constrained programming problem:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{s.t.} \quad & \text{Prob}\{\mathbf{w}^\top \cdot \mathbf{X}_1 - b \geqslant 1\} \geqslant \eta_1, \\ & \text{Prob}\{\mathbf{w}^\top \cdot \mathbf{X}_2 - b \leqslant -1\} \geqslant \eta_2. \end{aligned} \tag{1}$$

In other words, the model requires that the random variable $\mathbf{X}_i$ lies on the correct side of the hyperplane, with a probability greater than η_i for $i = 1, 2$. In this case, we want to be able to classify each training pattern correctly, up to the rate η_i, even for the *worst data distribution*, considering $\mathbf{X}_i \sim (\boldsymbol{\mu}_i, \Sigma_i)$. For this purpose, the probability constraints in (1) are replaced with their *robust* counterparts:

$$\inf_{\mathbf{X}_1 \sim (\boldsymbol{\mu}_1, \Sigma_1)} \text{Prob}\{\mathbf{w}^\top \cdot \mathbf{X}_1 - b \geqslant 1\} \geqslant \eta_1, \quad \inf_{\mathbf{X}_2 \sim (\boldsymbol{\mu}_2, \Sigma_2)} \text{Prob}\{\mathbf{w}^\top \cdot \mathbf{X}_2 - b \leqslant -1\} \geqslant \eta_2.$$

Applying the multivariate Chebyshev inequality [20, Lemma 1], these constraints are equivalents to

$$\mathbf{w}^\top \cdot \boldsymbol{\mu}_1 - b \geqslant 1 + \kappa_1 \sqrt{\mathbf{w}^\top \cdot \Sigma_1 \mathbf{w}}, \quad b - \mathbf{w}^\top \cdot \boldsymbol{\mu}_2 \geqslant 1 + \kappa_2 \sqrt{\mathbf{w}^\top \cdot \Sigma_2 \mathbf{w}},$$

where $\kappa_i = \sqrt{\frac{\eta_i}{1-\eta_i}}$, for $i = 1, 2$. Hence, this leads to the following deterministic problem:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \cdot \boldsymbol{\mu}_1 - b \geqslant 1 + \kappa_1 \|S_1^\top \mathbf{w}\|, \\ & b - \mathbf{w}^\top \cdot \boldsymbol{\mu}_2 \geqslant 1 + \kappa_2 \|S_2^\top \mathbf{w}\|. \end{aligned} \tag{2}$$

Finally, Eq. (2) can be cast into a convex problem with a linear objective function and three second-order cone (SOC) constraints by introducing a new variable t and an additional constraint $\|\mathbf{w}\| \leqslant t$. The solutions for both problems are essentially the same but linear SOCP formulations are required by some SOCP solvers, such as SeDuMi Toolbox for Matlab [28]. These linear SOCP formulations can be solved efficiently by interior point methods [1,2].

3. Multiclass Support Vector Machines

In this section we describe the formulation of SVMs for multiclass classification, considering the two most common variations: One-versus-All and One-versus-One. The second-order cone programming formulations for both approaches are provided in Section 5, together with the proposed feature selection algorithm.

3.1. One-versus-All Support Vector Machine

This is the simplest and probably the earliest implementation for multiclass SVMs [8]. This approach is called the One-versus-All (OvA), and constructs K binary SVM classifiers, K being the total number of classes, where each one separates a particular class from the remaining training patterns. The k-th SVM classifier is trained with all the training examples of the k-th class as positive labels, while the remaining instances are used with negative labels. Formally, for m training points of the form $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$, where $\mathbf{x}_i \in \Re^{|\mathcal{S}|}$ is a feature vector representing the i-th sample, and $y_i \in \{1, 2, \ldots, K\}$ is the class label of $\mathbf{x}_i$, the k-th SVM solves the following problem:

$$\begin{aligned} \min_{\mathbf{w}_k, b_k, \boldsymbol{\xi}^k} \quad & \frac{1}{2}\|\mathbf{w}_k\|^2 + C\sum_{i=1}^{m} \xi_i^k \\ \text{s.t.} \quad & \tilde{y}_i(\mathbf{w}_k^\top \cdot \mathbf{x}_i + b_k) \geqslant 1 - \xi_i^k, \\ & \xi_i^k \geqslant 0, \quad i = 1, \ldots, m, \end{aligned} \tag{3}$$

where $\tilde{y}_i^k = 1$ if $y_i = k$ and $\tilde{y}_i^k = -1$ otherwise. The decision function associated with this problem is given by $f_k(\mathbf{x}) = \mathbf{w}_k^\top \cdot \mathbf{x} + b_k$. Then, a sample $\mathbf{x}$ will be classified in the class which attains the greatest value of $f_k(\mathbf{x})$, that is, $\mathbf{x}$ is in the k^*th class when $f_{k^*}(\mathbf{x}) = \max\{f_k(\mathbf{x}) : k = 1, \ldots, K\}$. In the (exceptional) case when this maximum is attained in more than one class sample $\mathbf{x}$ is (by convention) classified in the class associated with the lowest index k^*.

Note that in the binary case (i.e. $K = 2$), Problem (3) reduces to the classical SVM problem [29].

3.2. One-versus-One Support Vector Machines

Another important SVM-based multiclass classification method is known as One-versus-One (OvO) Support Vector Machine [19]. This method constructs $K(K-1)/2$ binary SVM classifiers, one for every pair of classes. For training data from the k-th and the l-th classes, $k \neq l$ ($k < l$), OvO-SVM solves the

following binary classification problem:

$$\begin{aligned} \min_{\mathbf{w}_{kl}, b_{kl}, \boldsymbol{\xi}^{kl}} \quad & \frac{1}{2}\|\mathbf{w}_{kl}\|^2 + C\sum_i \xi_i^{kl} \\ \text{s.t.} \quad & \mathbf{w}_{kl}^\top \cdot \mathbf{x}_i + b_{kl} \geqslant 1 - \xi_i^{kl}, \text{ if} y_i = k, \\ & -(\mathbf{w}_{kl}^\top \cdot \mathbf{x}_i + b_{kl}) \geqslant 1 - \xi_i^{kl}, \text{ if} y_i = l, \\ & \xi_i^{kl} \geqslant 0, \quad i = 1, \ldots, m_k + m_l, \end{aligned} \tag{4}$$

where m_k denotes the number of elements of the class k. The decision function associated with this problem is given by $f_{kl}(\mathbf{x}) = \mathbf{w}_{kl}^\top \cdot \mathbf{x} + b_{kl}$.

Classification of new instances is performed by a max-wins voting strategy [14], in which every classifier assigns each data point to one of the two classes, increasing the vote for the assigned class by one. Finally, the class with the maximum number of votes determines the classification of each instance.

4. Feature selection for multiclass classification

Three main types of feature selection approaches have been proposed in the literature: filter, wrapper, and embedded methods [15]. The first scheme (*filter methods*) uses statistical properties of the features to filter out the irrelevant ones, assessing the correlation between predictors and labels. One common filter method is the Fisher Criterion Score, which is based on Fisher's Linear Discriminant Analysis (LDA). For each attribute $j \in \mathcal{S}$, the multiclass version for the Fisher Score follows [13,35]:

$$F(j) = \frac{\sum_{k=1}^{K} \frac{n_k}{K-1} (\bar{x}_{kj} - \hat{x}_j)}{\sigma^2}, \tag{5}$$

where $\bar{x}_{kj} = \frac{1}{n_k}\sum_{i \in C_k} x_{ij}$ is the average value of variable j in class C_k, n_k denotes the number of elements of class C_k, and $\hat{x}_j = \frac{1}{K}\sum_{k=1}^{K} \bar{x}_{kj}$, their respective means along the different classes. The variance of the whole data set σ^2 (see [35]) is given by:

$$\sigma^2 = \frac{\sum_{k=1}^{K} n_k (n_k - 1)\, \sigma_k^2}{|\mathcal{S}| - K}, \tag{6}$$

where

$$\sigma_k^2 = \frac{\sum_{i \in C_k} (x_{ij} - \bar{x}_{kj})^2}{n_k - 1} \tag{7}$$

denotes the variance of variable j in the k-th class. Another filter approach is Cho's measure, defined as follows [12,35]:

$$Cho(j) = \frac{mean(j) \cdot std(j)}{std(\hat{x}_j)}, \tag{8}$$

where

$$\textit{mean}(j) = \frac{\sum_{i=1}^{n} w_i x_{ij}}{\sum_{i=1}^{n} w_i}$$

represents a weighted mean for attribute j, and where w_i is $\frac{1}{n_k}$ if the instance i belongs to class k, and zero otherwise. $\textit{std}(\hat{x}_j)$ is the standard deviation of $\bar{x}_{kj}$, and $\textit{std}(j)$ is the weighted standard deviation around $\textit{mean}(j)$, defined as:

$$\textit{std}(j) = \sqrt{\frac{\sum_{i=1}^{|\mathcal{S}|} \left(x_{ij} - \textit{mean}(j)\right)^2}{|\mathcal{S}| - \frac{1}{|\mathcal{S}|}\sum_{i=1}^{|\mathcal{S}|} w_i}} \tag{9}$$

Another filter approach designed for gene selection, called GS1, was proposed by Yang et al. [35]. For a given attribute $j \in \mathcal{S}$, the following contribution measure is computed:

$$GS1(j) = \sqrt{\frac{1}{K}\sum_{k=1}^{K} \bar{x}_{kj}^{K}} + \sqrt{\frac{1}{K}\sum_{k=1}^{K} \left(\bar{x}_{kj} - \hat{x}_j\right)^2}, \tag{10}$$

where the first component can be interpreted as the means of intra-class variations, and the second represents the intra-class deviations.

Wrapper methods aim at scoring different feature subsets according to their predictive power. Since the exhaustive search for an optimal subset of features is an NP-hard problem [3], several heuristic approaches have been suggested, such as a greedy search or genetic algorithms [15].

Embedded methods attempt to find an optimal subset of variables in the process of model construction. These methods depend directly on the nature of the classification strategy used. In general, embedded methods present important advantages in terms of variable and model interaction, accurately capturing the dependencies between variables, and being computationally less demanding than wrapper methods [15].

One popular embedded method, which is relevant for the remainder of this work, is known as Recursive Feature Elimination (RFE-SVM) [16]. The goal of this approach is to find a subset of r variables from $\mathcal{S}$, eliminating those whose removal leads to the largest margin of class separation. This can be achieved using a backward elimination approach, based on the components of the weight vector $\mathbf{w}$.

Several extensions of the RFE-SVM have been proposed for multiclass classification. A method called OvA-RFE has been proposed, in which all k hyperplanes are estimated first, and RFE-SVM is then performed for each decision function independently [25,37]. After k feature subsets are selected, the final subset is obtained by simply combining all k subsets. Another strategy is to combine the weights obtained by all k functions in a single contribution measure for each attribute j, commonly using the Euclidean norm [37]. Formally, for a set of weight vectors obtained by using any multiclass strategy ($\mathbf{w}_k$ for OvA-SVM or $\mathbf{w}_{kl}$ for OvO-SVM), a contribution metric c_j can be computed as $c_j(\mathbf{w}) = \sum_k w_{kj}^2$ for OvA-SVM, or $c_j(\mathbf{w}) = \sum_k \sum_l w_{klj}^2$ for OvO-SVM. The backward algorithm for multiclass RFE-SVM is presented in Algorithm 1.

In Algorithm 1 one single feature is eliminated at each iteration, which would be inefficient when there is a large number of irrelevant features. On the other hand, removing too many features at once increases the risk of losing relevant features [15]. The authors found that a good compromise between

Algorithm 1 Recursive Feature Elimination SVM for Multiclass

1. **repeat**
2. $\mathbf{w} \leftarrow$ SVM Training (multiclass formulation).
3. Eliminate feature j with smallest value of $c_j(\mathbf{w})$.
4. **until** r variables remain.

speed and feature quality was to remove 50% of the current features at every iteration for microarray datasets.

An alternative embedded method is the minimization of the cardinality of the non-zero components of the weight vector, also known as the "zero norm": $\|\mathbf{w}\|_0 = |\{j : w_j \neq 0\}|$. Note that $\|\cdot\|_0$ is not a norm because the triangle inequality does not hold [9]. Weston [32] proposed an approach for "zero norm" minimization (l_0-SVM) by scaling the variables iteratively, multiplying them by componentwise addition of the absolute values of $\mathbf{w}_k$ obtained from the OvA-SVM formulation, until convergence. Variables can be ranked by removing those features whose weights become zero during the iterative algorithm and computing the order of removal. The l_0-SVM Algorithm for multiclass follows:

Algorithm 2 l_0-SVM for Multiclass

1. set $\mathbf{z} = \mathbf{e}$
2. **repeat**
3. $\mathbf{w}_k \leftarrow$ SVM Training, solve :

$$\begin{aligned} \min_{\mathbf{w}_k, b_k} \quad & \sum_{k=1}^{K} ||\mathbf{w}_k||^2 \\ \text{s.t.} \quad & y_{ki} \cdot ({\mathbf{w}_k}^\top \cdot \mathbf{x}_i * \mathbf{z} + b_k) \geqslant 1, \quad i = 1, \ldots, m;\ k = 1, \ldots, K; \end{aligned} \tag{11}$$

4. $\mathbf{z} \leftarrow \mathbf{z} * \left(\sum_{k=1}^{K} |\bar{w}_{k1}|, \ldots, \sum_{k=1}^{K} |\bar{w}_{k|\mathcal{S}|}| \right)$.
5. **until convergence**

where $*$ denotes the componentwise vector product operator, which is defined as $\mathbf{a} * \mathbf{b} = (a_1 b_1, \ldots, a_n b_n)$, $\mathbf{z}$ are the scaling factors and $\bar{\mathbf{w}}_k \in \Re^{|\mathcal{S}|}$ represents the solution of Formulation (11), which is equivalent to the linear OvA-SVM where the matrix $\mathbf{X}$ has been scaled by the factor $\mathbf{z}$.

To the best of our knowledge, no feature selection approach based on the SOCP formulation for SVM has been proposed so far for multiclass classification, although one study presented by Bhattacharyya [6] uses second-order cones to perform embedded feature selection for binary classification. In that work, the author extends the ideas of l_1 Support Vector Machine (l_1-SVM) to second-order cones, by minimizing the l_1 norm instead of the Euclidean norm used in Formulation (2).

5. Multiclass RFE-SOCP, a novel feature selection approach

We propose a family of embedded methods for backward feature selection using multiclass SOCP Support Vector Machines, which were inspired by the backward elimination procedure of RFE-SVM [15] presented in the previous Section. The rationale behind our approach is that we eliminate those features whose removal has less impact for the final solution.

Section 5.1 presents the RFE algorithm for One-versus-All classification, while Section 5.2 describe the model for the One-versus-One SVM, considering second-order cone programming formulations.

5.1. One versus All RFE-SOCP Support Vector Machines

The proposed approach is introduced in two steps. The OvA-SOCP-SVM formulation for multiclass classification is presented first, while the algorithm for backward elimination is described subsequently.

5.1.1. OvA-SOCP Support Vector Machines

Based on the method OvA-SVM, we can formulate a version of OvA-SOCP-SVM. Let $\mathbf{X}_k$ be a random vector variable that generates samples of class k, with mean and covariance matrix given by $(\boldsymbol{\mu}_k, \Sigma_k)$; and let $\mathbf{X}_k^c$ be the random vector that generates samples of the remaining classes, having $(\boldsymbol{\mu}_k^c, \Sigma_k^c)$, where $\Sigma_k, \Sigma_k^c \in \Re^{|\mathcal{S}|\times|\mathcal{S}|}$ are symmetric positive semidefinite matrices. Then, for each $k = 1, \ldots, K$, we consider the following quadratic chance-constrained programming problem:

$$\begin{aligned} \min_{\mathbf{w}_k, b_k} \ & \frac{1}{2}\|\mathbf{w}_k\|^2 \\ \text{s.t.} \ & \inf_{\mathbf{X}_k \sim (\boldsymbol{\mu}_k, \Sigma_k)} \text{Prob}\{\mathbf{w}_k{}^\top \cdot \mathbf{X}_k \geqslant b_k + 1\} \geqslant \eta_k, \\ & \inf_{\mathbf{X}_k^c \sim (\boldsymbol{\mu}_k^c, \Sigma_k^c)} \text{Prob}\{\mathbf{w}_k{}^\top \cdot \mathbf{X}_k^c \leqslant b_k - 1\} \geqslant \eta_k^c, \end{aligned} \tag{12}$$

where $\eta_k, \eta_k^c \in (0, 1)$. Thanks to an appropriate application of the multivariate Chebyshev inequality [20, Lemma 1], the Eq. (12) can be stated as the following quadratic SOCP problem for each $k = 1, \ldots, K$:

$$\begin{aligned} \min_{\mathbf{w}_k, b_k} \ & \frac{1}{2}\|\mathbf{w}_k\|^2 \\ \text{s.t.} \ & \mathbf{w}_k^\top \cdot \boldsymbol{\mu}_k - b_k \geqslant 1 + \kappa_k \|S_k^\top \mathbf{w}_k\|, \\ & b_k - \mathbf{w}_k^\top \cdot \boldsymbol{\mu}_k^c \geqslant 1 + \kappa_k^c \|S_k^{c\top} \mathbf{w}_k\|, \end{aligned} \tag{13}$$

with $\kappa_k = \sqrt{\frac{\eta_k}{1-\eta_k}}$ (resp. $\kappa_k^c = \sqrt{\frac{\eta_k^c}{1-\eta_k^c}}$).

The decision function is similar to the one used for OvA-SVM, that is, a new data point $\mathbf{x}$ belongs to the class k^* iff $k^* = \arg\max_{k=1,\ldots,K}\{\mathbf{w}_k^\top \cdot \mathbf{x} - b_k\}$.

Note that in the binary case (i.e. $K = 2$), Problems (12) and (13) reduce to the formulations proposed by [23] (cf. Eqs (1) and (2)). In Bosch et al. [7], we used the OvA version for SOCP-SVM to classify fish schools, although the method was not formalized in the form of Eqs (12) and (13).

5.1.2. RFE algorithm for OvA SOCP-SVM

Following the notation used by Song et al. [26], the proposed approach constructs K classifiers and determines a subset $\mathcal{I}$ of features to be eliminated at each iteration. The output of the method is an ordered vector of variables $\mathcal{S}^\dagger$, which can be used to construct different feature subsets, as we describe in the experimental section. Similar to RFE-SVM [15], we consider removing between 10% and 50% of the available features at every iteration. The Recursive Feature Elimination algorithm for One-versus-All SOCP-SVM (OvA-RFE-SOCP) is presented in Algorithm 3.

Algorithm 3 RFE Algorithm for OvA-SOCP-SVM (OvA-RFE-SOCP)

Input: The original set of features ($\mathcal{S}$)
Output: An ordered vector of features $\mathcal{S}^\dagger$

1. $\mathcal{S}^\dagger \leftarrow \emptyset$
2. **repeat**
3. $\mathbf{w}_k \leftarrow \text{OvA SOCP} - \text{SVM Training:}$

$$\begin{aligned} \min_{\mathbf{w}_k, b_k} \quad & \frac{1}{2}\|\mathbf{w}_k\|^2 \\ \text{s.t.} \quad & \mathbf{w}_k^\top \cdot \boldsymbol{\mu}_k - b_k \geqslant 1 + \kappa_k \|S_k^\top \mathbf{w}_k\|, \\ & b_k - \mathbf{w}_k^\top \cdot \boldsymbol{\mu}_k^c \geqslant 1 + \kappa_k^c \|S_k^{c\top} \mathbf{w}_k\|, \end{aligned} \tag{14}$$

4. $\mathcal{I} \leftarrow \text{argmin}_{\mathcal{I}} \sum_{j \in \mathcal{I}} c_j = \sum_k w_{kj}^2,\ \mathcal{I} \subset \mathcal{S}$
5. $\mathcal{S} \leftarrow \mathcal{S} \setminus \mathcal{I}$
6. $\mathcal{S}^\dagger \leftarrow (\mathcal{S}^\dagger, \mathcal{I})$
7. **until** $\mathcal{S} = \emptyset$

5.2. One-versus-One RFE-SOCP Support Vector Machines

We introduce this method in two steps. The One versus One SOCP-SVM formulation is presented first, while the feature selection algorithm is described subsequently.

5.2.1. OvO-SOCP Support Vector Machines

Similar to OvA-SOCP-SVM, let us consider $\mathbf{X}_k$ a random vector variable that generates samples of class k, with mean and covariance matrix given by $(\boldsymbol{\mu}_k, \Sigma_k)$ for $k = 1, \ldots, K$, where $\Sigma_k \in \Re^{|\mathcal{S}|\times|\mathcal{S}|}$ are symmetric positive semidefinite matrices.

Then, we can formulate a version of OvO-SOCP-SVM based on the idea of OvO-SVM. More precisely, for training data from the k-th and the l-th classes ($k < l$), we solve the following quadratic chance-constrained programming problem:

$$\begin{aligned} \min_{\mathbf{w}_{kl}, b_{kl}} \quad & \frac{1}{2}\|\mathbf{w}_{kl}\|^2 \\ \text{s.t.} \quad & \inf_{\mathbf{X}_k \sim (\boldsymbol{\mu}_k, \Sigma_k)} \text{Prob}\{\mathbf{w}_{kl}{}^\top \cdot \mathbf{X}_k \geqslant b_{kl} + 1\} \geqslant \eta_{kl}, \\ & \inf_{\mathbf{X}_l \sim (\boldsymbol{\mu}_l, \Sigma_l)} \text{Prob}\{\mathbf{w}_{kl}{}^\top \cdot \mathbf{X}_l \leqslant b_{kl} - 1\} \geqslant \eta_{lk}, \end{aligned} \tag{15}$$

where $\eta_{kl}, \eta_{lk} \in (0, 1)$.

Again, thanks to an appropriate application of the multivariate Chebyshev inequality, Formulation (15) can be stated as the following quadratic SOCP problem:

$$\begin{aligned} \min_{\mathbf{w}_{kl}, b_{kl}} \quad & \frac{1}{2}\|\mathbf{w}_{kl}\|^2 \\ \text{s.t.} \quad & \mathbf{w}_{kl}^\top \cdot \boldsymbol{\mu}_k - b_{kl} \geqslant 1 + \kappa_{kl} \|S_k^\top \mathbf{w}_{kl}\|, \\ & b_{kl} - \mathbf{w}_{kl}^\top \cdot \boldsymbol{\mu}_l \geqslant 1 + \kappa_{lk} \|S_l{}^\top \mathbf{w}_{kl}\|, \end{aligned} \tag{16}$$

with $\kappa_{kl} = \sqrt{\frac{\eta_{kl}}{1-\eta_{kl}}}$ (resp. $\kappa_{lk} = \sqrt{\frac{\eta_{lk}}{1-\eta_{lk}}}$). Equivalently to OvO-SVM, this method constructs $K(K-1)/2$ binary classifiers, one for each pair of classes.

The decision function is given by $f_{kl}(\mathbf{x}) = \mathbf{w}_{kl}^{\top} \cdot \mathbf{x} - b_{kl}$, and the prediction of a new point $\mathbf{x}$ is done by the Max-Wins voting strategy.

5.2.2. RFE algorithm for OvO SOCP-SVM

Similar to the feature selection algorithm for OvA-SOCP-SVM, we construct the respective classifiers iteratively, compute the contribution measure based on the weight vectors, and eliminate a subset of irrelevant attributes $\mathcal{I}$, according to the proposed contribution measure. From the algorithm we obtain an ordered vector of features, $\mathcal{S}^{\dagger}$, and the classifiers for each training step. The Recursive Feature Elimination algorithm for One versus One SOCP-SVM (OvO-RFE-SOCP) is presented in Algorithm 4.

Algorithm 4 RFE Algorithm for OvO-SOCP-SVM (OvO-RFE-SOCP)

Input: The original set of features $(\mathcal{S})$
Output: An ordered vector of features $\mathcal{S}^{\dagger}$

1. $\mathcal{S}^{\dagger} \leftarrow \emptyset$
2. **repeat**
3. $\mathbf{w}_k \leftarrow \mathrm{OvO\ SOCP-SVM\ Training:}$

$$\begin{aligned} \min_{\mathbf{w}_{kl}, b_{kl}} \quad & \frac{1}{2}\|\mathbf{w}_{kl}\|^2 \\ \text{s.t.} \quad & \mathbf{w}_{kl}^{\top} \cdot \boldsymbol{\mu}_k - b_{kl} \geqslant 1 + \kappa_{kl}\|S_k^{\top}\mathbf{w}_{kl}\|, \\ & b_{kl} - \mathbf{w}_{kl}^{\top} \cdot \boldsymbol{\mu}_l \geqslant 1 + \kappa_{lk}\|S_l{}^{\top}\mathbf{w}_{kl}\|, \end{aligned} \tag{17}$$

4. $\mathcal{I} \leftarrow \mathrm{argmin}_{\mathcal{I}} \sum_{j \in \mathcal{I}} c_j = \sum_k \sum_l w_{klj}^2,\ \mathcal{I} \subset \mathcal{S}$
5. $\mathcal{S} \leftarrow \mathcal{S} \setminus \mathcal{I}$
6. $\mathcal{S}^{\dagger} \leftarrow (\mathcal{S}^{\dagger}, \mathcal{I})$
7. **until** $\mathcal{S} = \emptyset$

6. Experimental results

We applied the proposed and alternative feature selection approaches to four microarray data sets for multiclass classification. These sets have already been used as benchmarks in feature selection (see, for example, [27] or [35]).

We provide a description of the microarray data sets in Section 6.1, while Section 6.2 presents a summary of the performance obtained for all the proposed and alternative approaches. Finally, an empirical analysis regarding the influence of the different parameters is presented in Section 6.3.

6.1. Datasets and experimental settings

GLIOMA data set: The GLIOMA data set contains 50 instances described by 4433 genes in four classes: *cancer glioblastomas* (14 samples), *non-cancer glioblastomas* (14 samples), *cancer oligodendrogliomas* (7 samples), and *non-cancer oligodendrogliomas* (15 samples) [24,35]. We studied the per-

formance of the following subsets of features: 20, 50, 100, 250, 1000, 2000, 4433 (i.e. no features removed).

SRBCT data set: The SRBCT data set contains 83 samples in four classes: *Ewing family of tumors* (29 samples), *Burkitt lymphoma* (11 samples), *neuroblastoma* (18 samples), and *rhabdomyosarcoma* (25 samples) [18]. Each sample contains 2308 genes. We studied the performance of the following subsets of features: 20, 50, 100, 250, 1000, 2308 (i.e. no features removed).

LUNG data set: The LUNG data set contains 203 samples described by 3312 genes in five classes: *adenocarcinomas* (139 samples), *squamous cell lung carcinomas* (21 samples), *pulmonary carcinoids* (20 samples), *small-cell lung carcinomas* (6 samples), and *normal lung* (17 samples) [5,35]. We studied the performance of the following subsets of features: 20, 50, 100, 250, 1000, 2000, 3312 (i.e. no features removed).

MLL data set: The MLL data set contains 72 samples described by 8685 genes in three classes: *acute lymphoblastic leukemia* (ALL, 24 samples), *acute myeloid leukemia* (AML, 28 samples) and *mixed-lineage leukemia gene*(MLL, 20 samples) [4,35]. We studied the performance of the following subsets of features: 20, 50, 100, 250, 1000, 2000, 4000, 8685 (i.e. no features removed).

The following model selection procedure was performed: training and test subsets were constructed using leave-one-out (LOO) cross-validation, which have been frequently used for assessing feature selection and classification in microarray data [22,34]. The filter methods Fisher Score, Cho's test, and Yang's GS1 were used to construct a ranking of features over the training set, and then we trained both standard SVM and SOCP-SVM as classifiers for a fixed number of ranked features. Similarly, the RFE-SVM and l_0-SVM methods were used for feature ranking based on standard SVM, and then we trained both presented classification approaches, namely standard SVM and SOCP-SVM, for a fixed number of ranked features. The proposed methods, OvO-RFE-SOCP and OvA-RFE-SOCP, were trained as proposed in Section 5, and therefore are only suitable with SOCP-SVM as the classifier.

The goal of the empirical framework is to assess the performance of our proposal compared to alternative feature selection approaches in high dimensional datasets, controlling by the nature of the classifier (standard SVM and SOCP-SVM). We limit ourselves to linear SVM classifiers. For this work we studied balanced accuracy as the main performance metric to assess predictive performance.

A grid search was performed to study the influence of the parameters C for soft-margin models, and η for SOCP approaches. In this case, we considered $\eta_{kl} = \eta$ (One-versus-One classification) and $\eta_k = \eta_k^c = \eta$ (One-versus-All classification), and studied the following values of $\eta \in \{0.2, 0.4, 0.6, 0.8\}$. For standard SVM approaches, we used the following set of values for parameter C:

$$C \in \{2^{-7}, 2^{-6}, 2^{-5}, 2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^{0}, 2^{1}, 2^{2}, 2^{3}, 2^{4}, 2^{5}, 2^{6}, 2^{7}\}.$$

For the above procedure, we used the Spider Toolbox for Matlab [31] for standard SVM approaches, and the SeDuMi Matlab Toolbox for linear SOCP-based classifiers [28].

6.2. Classification performance summary

Table 1 summarizes the results obtained from the model selection procedure for each feature selection approach and for all four data sets. We first averaged all the different subsets of attributes to obtain a mean performance for each method, and then we selected the best combination of hyperparameters C (standard SVM) and η (SOCP-SVM). Table 1 presents the best performance considering four different base classifiers: OvA-SVM, OvO-SVM, OvA-SOCP-SVM, and OvO-SOCP-SVM, with the only exception being the proposed RFE-SOCP method, which was designed only for OvA-SOCP-SVM and OvO-SOCP-SVM.

Table 1
Performance summary for different feature selection approaches. All datasets

		GLIOMA	LUNG	MLL	SRBCT
Fisher	OvA SVM	70.8	86.8	95.2	99.2
	OvO SVM	70.8	89.0	94.6	98.8
	OvA SOCP	70.3	90.8	96.4	*99.5*
	OvO SOCP	73.5	89.3	96.4	99.3
Cho	OvA SVM	62.9	83.8	93.7	91.7
	OvO SVM	62.4	81.0	93.5	90.7
	OvA SOCP	68.3	88.8	94.8	92.6
	OvO SOCP	70.3	90.8	94.0	92.5
GS1	OvA SVM	67.9	85.6	95.4	98.0
	OvO SVM	67.6	84.2	94.8	98.7
	OvA SOCP	72.3	89.6	95.8	98.8
	OvO SOCP	73.2	89.6	95.8	98.8
l0	OvA SVM	66.1	90.4	91.1	98.4
	OvO SVM	67.4	88.1	96.2	97.8
	OvA SOCP	67.1	92.7	91.9	98.1
	OvO SOCP	70.0	91.1	*96.6*	98.3
SVM-RFE	OvA SVM	69.9	78.2	87.7	81.5
	OvO SVM	69.1	77.2	91.2	86.1
	OvA SOCP	73.1	84.1	90.6	86.1
	OvO SOCP	71.6	85.2	92.4	88.0
RFE-SOCP	OvA	70.3	*94.0*	*96.6*	*99.5*
	OvO	*76.5*	92.6	96.5	99.4

In Table 1 we observe that the best predictive results were achieved with the proposed approach. For the first two data datasets the gain is significant compared to the alternative approaches, while results are relatively similar between the latter two approaches, and good results can be achieved with all approaches. Another interesting result is that SOCP-SVM in both versions tend to be better base classifiers than standard SVM predictors, while the difference between OvO and OvA classifiers is not significant. According to these results, we can conclude that the gain in terms of performance of the proposed method is due to the use of SOCP-SVM classifiers and the embedded feature selection process based on SOCP-SVM.

For the next experiments we studied the performance for the different subsets of attributes presented in the description of the data set. For each feature selection approach we selected the one with the best overall performance for all hyperparameters and base classifiers. Figures 1 to 4 display the results of these experiments for each dataset.

For the GLIOMA dataset (Fig. 1), we observe a linear decay in the performance of most methods, while our approach works consistently well for the different subsets of attributes, achieving best performance while using 1000 variables.

For the LUNG dataset (Fig. 2), results are more stable and follow similar curves. Again, the proposed method performs better along the different subsets, achieving its peak with 20 attributes. Cho's test and RFE have the lowest performance among the alternative approaches. The performance is strongly affected when selecting 10 attributes in all cases.

For the MLL dataset (Fig. 3), several approaches behave equally well, such as the proposed RFE-SOCP, Fisher, and l_0-SVM. The RFE-SVM method has the best single performance with 250 attributes, but then its performance decreases rapidly while the best approaches are consistently good for all the different subsets of features.

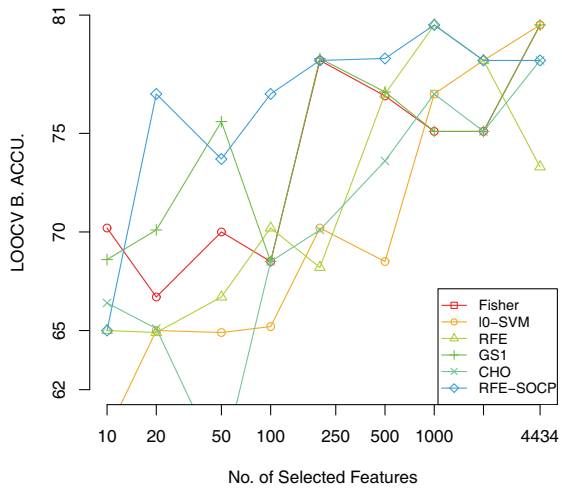


Fig. 1. Performance versus the number of ranked variables for different feature selection approaches. GLIOMA dataset. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-150773)

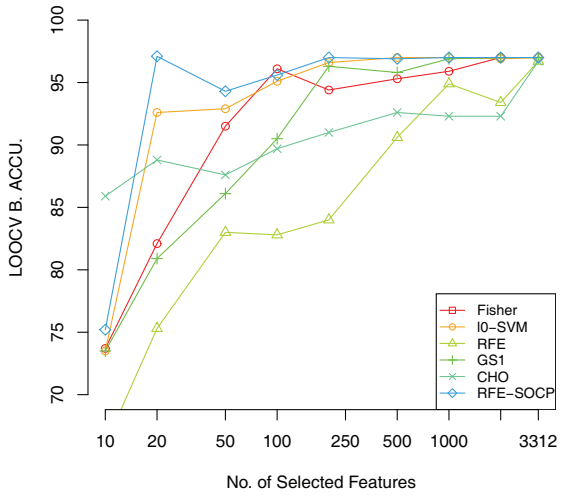


Fig. 2. Performance versus the number of ranked variables for different feature selection approaches. LUNG dataset. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-150773)

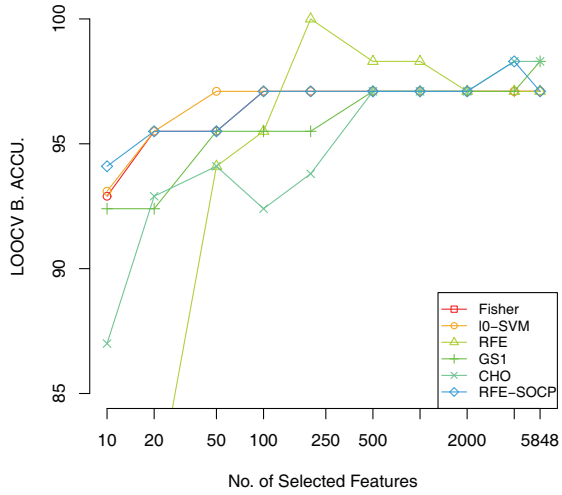


Fig. 3. Performance versus the number of ranked variables for different feature selection approaches. MLL dataset. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-150773)

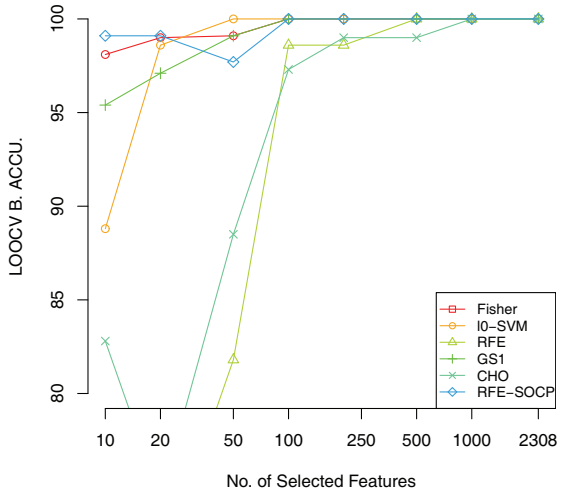
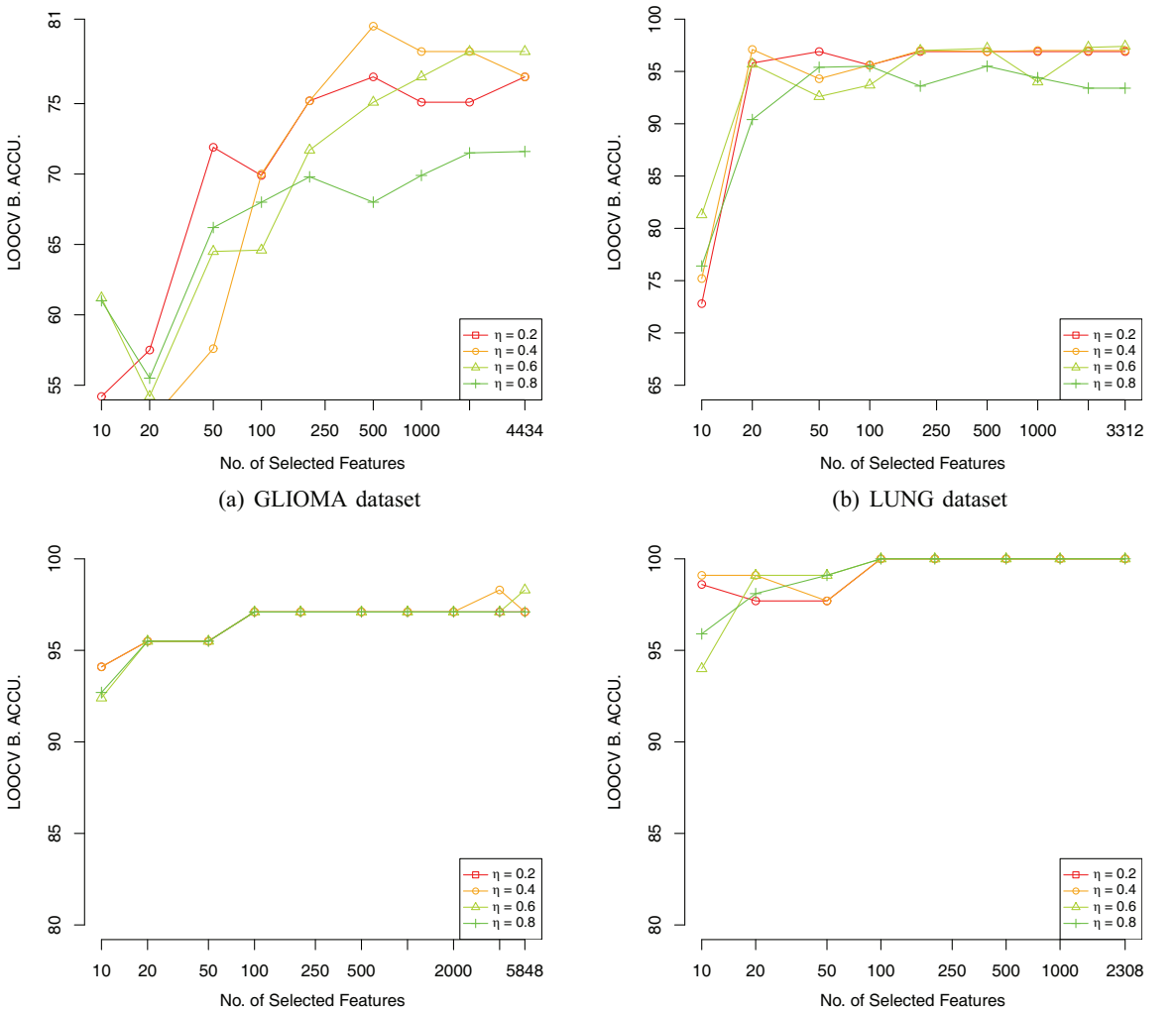


Fig. 4. Performance versus the number of ranked variables for different feature selection approaches. SRBCT dataset. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-150773)

Finally, for the SRBCT dataset (Fig. 4), the best approaches, Fisher and RFE-SOCP, are again consistently good and near 100% along the different subsets. Cho's test and RFE have the lowest performance among the alternative approaches.

6.3. Influence of the hyperparameters and discussion

In this subsection we report the performance of the proposed feature selection methodologies by performing sensitivity analysis of the relevant parameters, characterizing their influence on the final solu-



(a) GLIOMA dataset (b) LUNG dataset

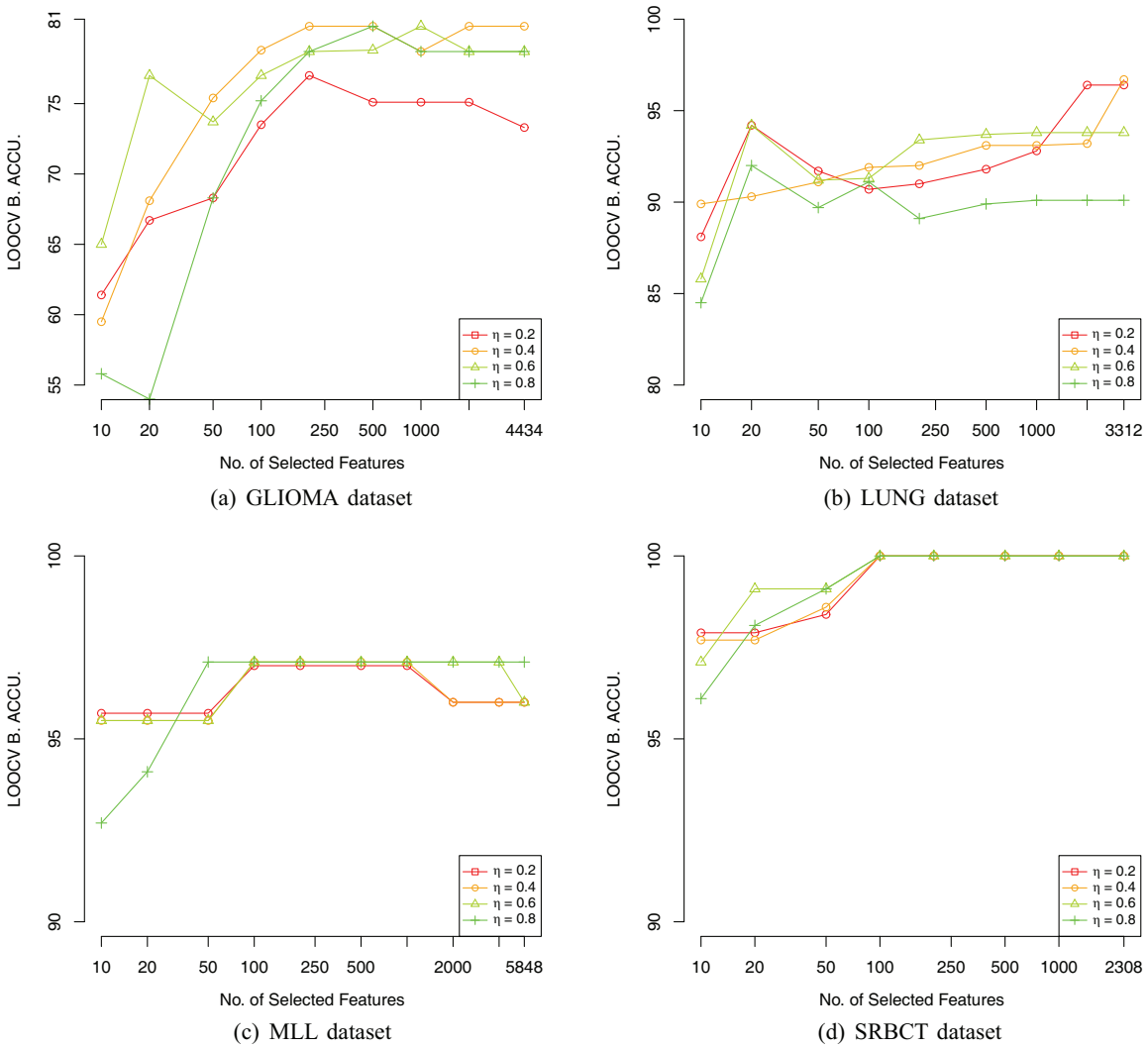
(c) MLL dataset (d) SRBCT dataset

Fig. 5. Performance versus the number of ranked variables for different η values. OvA-RFE-SOCP approach. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-150773)

tion. Our goal was to assess whether the results are stable along different values of the parameter η. If this is the case, a less rigorous validation strategy can be used. In contrast, a high variance in the performance would require more exhaustive model selection in order to find the best combination of parameters.

Figures 5 and 6 present the performance for the different η values for the OvA-RFE-SOCP and OvO-RFE-SOCP approaches respectively. For each figure, four graphs are presented, one for each data set. Each graph represents the performance of the respective approach for an increasing number of ranked features and for $\eta \in \{0.2, 0.4, 0.6, 0.8\}$.

In the previous figures we observe a low variance along the different values of η. All the curves have relatively similar shapes, concluding that the model is robust in the context of the hyperparameter settings.



(a) GLIOMA dataset (b) LUNG dataset

(c) MLL dataset (d) SRBCT dataset

Fig. 6. Performance versus the number of ranked variables for different η values. OvO-RFE-SOCP approach. (Colours are visible in the online version of the article; http://dx.doi.org/10.3233/IDA-150773)

7. Conclusions

In this paper we present a backward elimination strategy for multiclass classification and feature selection using the robust SVM formulation with second-order cones. A comparison with other feature selection and classification approaches in high dimensional applications showed the advantages of the proposed strategy:

- It outperforms other feature selection techniques in terms of classification performance, based on the ability of SOCP-SVM to generalize better by assuming the worst distribution of the data, while embedding a feature selection strategy in this classifier.
- It extends the work of Guyon et al. [16] on multiclass feature selection and classification for standard SVM to SOCP-SVM, proposing a backward elimination approach that considers the use of the

Euclidean norm in the SOCP-SVM formulation, which has proven to be as effective and precise as standard SVM, outperforming it in some cases [21].

– Our approach can be extended to kernel functions for nonlinear feature selection and classification, giving flexibility to the model construction. The backward elimination procedure proposed in this work defines a contribution measure c_j (step 4 of Algorithms 3 and 4) based on the weights of the multiclass SOCP-SVM formulations. This measure can be adapted to kernel-based formulation by following the procedure suggested in Guyon et al. [15] for kernel-based RFE-SVM for binary classification.

In our experiments we assessed the alternative feature selection approaches considering four different classifiers: OvO and OvA-SVM, and the proposed OvO and OvA-SOCP-SVM. From the point of view of classifiers, we observed consistently better results using robust SOCP classification, while no significant differences were found between OvO and OvA classifiers. This is a powerful conclusion, since several works that compare the effectiveness of both approaches have been developed in the last decade [17]. In the experimental section we also discussed the influence of the hyperparameters associated with our proposal, namely the η parameter, concluding that the proposed methods are not strongly dependent on the setting of this parameter, presenting stable results along the different values of η.

There are several opportunities for future research in multiclass feature selection. First, feature selection can be seen as an optimization problem via feature penalization. The literature offers interesting approaches based on l_1 and l_0 penalization, which can be extended to multiclass SOCP-SVM. For instance, the extension of the work presented by Bhattacharyya [6] on embedded feature selection for binary SOCP-SVM via l_1 penalization to multiclass classification represents an interesting research opportunity. Secondly, there is a pressing need for more efficient implementations than SeDuMI Matlab toolbox for SOCP-SVM. Finally, SOCP-SVM presents interesting properties for classification on highly imbalanced data sets, a very relevant topic in pattern recognition given the vast application field. Since the parameter η controls the Type I and Type II errors, a differentiated value of this parameter may help to construct better classification functions that consider the costs of both types of errors. This is particularly interesting in multiclass classification, where it is relatively common for one or several categories to be rarities in the dataset, and the classifier will favor the better-represented classes and produce classifiers with poorly balanced performance [35].

Acknowledgements

The first author was funded by FONDECYT project 11110188 and by CONICYT Anillo ACT1106, while the second author was supported by FONDECYT projects 11121196 and 1140831. The work reported in this paper has been partially funded by the Complex Engineering Systems Institute (ICM: P-05-004-F, CONICYT: FB016).

References

[1] F. Alizadeh and D. Goldfarb, Second-order cone programming, *Mathematical Programming* **95** (2003), 3–51.

[2] F. Alvarez, J. López and H. Ramírez C., Interior proximal algorithm with variable metric for second-order cone programming: applications to structural optimization and support vector machines, *Optimization Methods Software* **25**(6) (2010), 859–881.

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

[4] S.A. Armstrong, J.E. Staunton, L.B. Silverman, R. Pieters, M.L. den Boer, M.D. Minden, S.E. Sallan, E.S. Lander, T.R. Golub and S.J. Korsmeyer, Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia, *Nature Genetics* **30** (2002), 41–47.
[5] A. Bhattacharjee, W.G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E.J. Mark, E.S. Lander, W. Wong, B.E. Johnson, T.R. Golub, D.J. Sugarbaker and M. Meyerson, Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses, in: *Proceedings of the National Academy of Sciences of USA 98*, (2001), 13790–13795.
[6] C. Bhattacharyya, Second order cone programming formulations for feature selection, *Journal of Machine Learning Research* **5** (2004), 1417–1433.
[7] P. Bosch, J. López, H. Ramírez C. and H. Robotham, Support vector machine under uncertainty: An application for hydroacoustic classification of fish-schools in Chile, *Expert Systems with Applications* **40**(10) (2013), 4029–4034.
[8] L. Bottou, C. Cortes, J.S. Denker, H. Drucker, I. Guyon, L.D. Jackel, Y. LeCun, U.A. Muller, E. Sackinger, P. Simard and V. Vapnik, Comparison of classifier methods: a case study in handwritten digit recognition, in: *Proceedings of International Conference on Pattern Recognition* **2** (1994), 77–82.
[9] P. Bradley and O. Mangasarian, Feature selection via concave minimization and support vector machines, in: *Machine Learning Proceedings of the Fifteenth International Conference (ICML'98) 82–90, San Francisco, California, Morgan Kaufmann*, (1998).
[10] C. Bravo, L.C. Thomas and R. Weber, Improving credit scoring by differentiating defaulter behaviour, *Journal of the Operational Research Society* **66** (2014), 771–781.
[11] E.J. Bredensteiner and K.P. Bennett, Multicategory classification by support vector machines, *Computational Optimizations and Applications* **12** (1999), 53–79.
[12] J.H. Cho, D. Lee, J.H. Park and I.B. Lee, New gene selection for classification of cancer subtype considering within-class variation, *FEBS Letters* **551** (2003), 3–7.
[13] S. Dudoit, J. Fridlyand and T.P. Speed, Comparison of discrimination methods for the classification of tumors using gene expression data, *Journal of the American Statistical Association* **97** (2002), 77–87.
[14] J.H. Friedman, Another approach to polychotomous classification, Technical report, Department of Statistics, Stanford University, 1996.
[15] I. Guyon, S. Gunn, M. Nikravesh and L.A. Zadeh, *Feature Extraction, Foundations and Applications*, Springer, Berlin, 2006.
[16] I. Guyon, J. Weston, S. Barnhill and V. Vapnik, Gene selection for cancer classification using support vector machines, *Machine Learning* **46**(1–3) (2002), 389–422.
[17] C.W. Hsu and C.J. Lin, A comparison of methods for multiclass support vector machines, *Neural Networks, IEEE Transactions on* **13**(2) (2002), 415–425.
[18] J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson and P.S. Meltzer, Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, *Nature Medicine* **7** (2001), 673–679.
[19] U.H.G. Kressel, Advances in kernel methods, chapter Pairwise classification and support vector machines, MIT Press, Cambridge, MA, USA, 1999, pp. 255–268.
[20] G. Lanckriet, L.E. Ghaoui, C. Bhattacharyya and M. Jordan, A robust minimax approach to classification, *Journal of Machine Learning Research* **3** (2003), 555–582.
[21] S. Maldonado and J. López, Imbalanced data classification using second-order cone programming support vector machines, *Pattern Recognition* **47**(5) (2014), 2070–2079.
[22] S. Maldonado, R. Weber and F. Famili, Feature selection for high-dimensional class-imbalanced data sets using support vector machines, *Information Sciences* **286** (2014), 228–246.
[23] S. Nath and C. Bhattacharyya, Maximum margin classifiers with specified false positive and false negative error rates, in: *Proceedings of the SIAM International Conference on Data Mining*, (2007).
[24] C.L. Nutt, D.R. Mani, R.A. Betensky, P. Tamayo, J.G. Cairncross, C. Ladd, U. Pohl, C. Hartmann, M.E. McLaughlin, T.T. Batchelor, P.M. Black, A. von Deimling, S.L. Pomeroy, T.R. Golub and D.N. Louis, Gene expression-based classification of malignant gliomas correlates better with survival than histological classification, *Cancer Research* **63** (2003), 1602–1607.
[25] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C.H. Yeang, M.L. Angelo, C. Ladd, M.L. Reich, E. Latulippe, J.P. Mesirov, T. Poggio, W. Gerald, M. Loda, E.S. Lander and T.R. Golub, Multiclass cancer diagnosis using tumor gene expression signatures., *Proceedings of the National Academy of Sciences* **98**(26) (2001), 15149–15154.
[26] L. Song, A. Smola, A. Gretton, J. Bedo and K. Borgwardt, Feature selection via dependence maximization, *Journal of Machine Learning Research* **13** (2012), 1393–1434.
[27] S. Student and K. Fujarewicz, Stable feature selection and classification algorithms for multiclass microarray data, *Biology Direct* **7**(33) (2012).
[28] J. Sturm, Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones, *Optimization Methods and Soft-*

ware **11**(12) (1999), 625–653, special issue on Interior Point Methods (CD supplement with software).
[29] V. Vapnik, *Statistical Learning Theory*, John Wiley and Sons, 1998.
[30] T. Verbraken, W. Verbeke and B. Baesens, A novel profit maximizing metric for measuring classification performance of customer churn prediction models, *IEEE Transactions on Knowledge and Data Engineering* **25**(5) (2012), 961–973.
[31] J. Weston, A. Elisseeff, G. BakIr and F. Sinz, The spider machine learning toolbox, 2005.
[32] J. Weston, A. Elisseeff, B. Schölkopf and M. Tipping, The use of zero-norm with linear models and kernel methods, *Journal of Machine Learning Research* **3** (2003), 1439–1461.
[33] J. Weston and C. Watkins, Multi-class support vector machines, in: *Proceedings of the Seventh European Symposium on Artificial Neural Networks* (1999).
[34] E. Xing, M. Jordan and R. Karp, Feature selection for high-dimensional genomic microarray data, in: *Proceedings of the Eighteenth International Conference on Machine Learning* (2001), 601–608.
[35] K. Yang, Z. Cai, J. Li and G. Lin, A stable gene selection in microarray data analysis, *BMC Bioinformatics* **7** (2006), 228.
[36] P. Zhong and M. Fukushima, Second-order cone programming formulations for robust multiclass classification, *Neural Computation* **19** (2007), 258–282.
[37] X. Zhou and D.P. Tuck, Msvm-rfe: extensions of svm-rfe for multiclass gene selection on dna microarray data, *Bioinformatics* **23**(9) (2007), 1106–1114.