Variables Selection for Multiclass SVM Using the Multiclass Radius Margin Bound

Fatima Zahra Aazi*, Abdeljalil Elouardighi, Boujemâa Achchab, Rafik Abdesselam, Yann Guermeur

LM^{2}CE Laboratory, FSJES, Hassan 1st University, Settat, Morocco, and ERIC Laboratory, Lumière Lyon 2 University, Lyon, France. faazi@mail.univ-lyon2.fr
LM^{2}CE Laboratory, FSJES, Hassan 1st University, Settat, Morocco. jalilardighi@yahoo.fr
LM^{2}CE, LAMSAD Laboratories, EST Berrechid, Hassan 1st University, Morocco. achchab@estb.ac.ma
COACTIS Laboratory, ISH, Lumière Lyon 2 University, Lyon, France. Rafik.Abdesselam@univ-lyon2.fr
LORIA-CNRS, Campus Scientifique, Vandoeuvre-lès-Nancy cedex, France. Yann.Guermeur@loria.fr

ABSTRACT

Support vector machines (SVM) are considered as a powerful tool for classification which demonstrate great performances in various fields. Presented for the first time for binary problems, SVMs have been extended in several ways to multiclass case with good results in practice. However, the existence of noise or redundant variables can reduce their performances, where the need for a selection of variables.
In this work, we are interested in determining the relevant explanatory variables for an SVM model in the case of multiclass discrimination (MSVM). The criterion proposed here consist in determining such variables using one of the upper bounds of generalization error specific to MSVM models known as radius margin bound [1].
A score derived from this bound will establish the order of relevance of variables, then, the selection of optimal subset will be done using forward method. The experiments are conducted on simulated and real data, and some results are compared with those of other methods of variable selection by MSVM.

KEYWORDS

Discrimination, Multiclass Support Vectors Machines (MSVM), Variables Selection, Multiclass Radius-Margin Bound, Hard Margin MSVM Models.

1 INTRODUCTION

In a classification problem, the relevant variables are not known a priori. The importance of selection is justified by the possibility of existence of correlated, noise and / or redundant variables which usually give significant error rates. Indeed, the variables selection essentially allows to improve the performances of forecasting or classification models by using only the variables that are important for the studied problem, reduce time and cost calculation and facilitate the understanding of the process generating information.
There are generally three categories of methods for variables selection [2-4]: Filter, Wrapper and Embedded. In the first category, the selection is made a priori before the estimation of forecasting model, it consists in testing each variable independently of others and then order them according to a given criterion. The Wrapper methods select variables after developing model and thus take into account the influence of variables on the performances of the model. The last category (Embedded) incorporates the selection of variables during the learning process.
In the context of SVM, binary or multiclass, the developed models do not allow an automatic selection of variables and use all available ones.
In binary case, several approaches were been proposed to show the possibility of variable selection with SVM, these approaches can be grouped into two categories. The first, containing Embedded methods, consist in modifying the optimization program of SVM, so as to integrate the selection in the classification process. The second derives criteria from SVM to do selection (Wrapper approaches).
Within the first category, several new forms of SVM were been proposed, the L_0SVM [5], L_1SVM [6,7], combination of L_0 and L_1 SVMs [8] and the infinite norm SVM [9] are examples of these forms. Similarly, by deriving criteria from SVM, various approaches were presented, including the recursive feature elimination algorithm SVM-RFE of Guyon et al. [10] using the margin as selection criterion and Rakotomamonjy's approach [11], considered as

extension of SVM-RFE, using the upper bounds of generalization error specifics to SVM.

In multi-class case, as extension of the approaches of the first category, Wang and Shen [12,13] replaced the L_2-penalty in MSVM model of Lee et al. ($MSVM_{LLW}$) [14] by L_1-penalty (L_1MSVM). Similarly, Zhang et al. [15] proposed a sup norm penalty which is more efficient and easier to implement than that given by the L_1MSVM solution. Other methods were also been proposed in this context [16,17].

Moreover, and as extension of SVM-RFE, several techniques were presented, based either on a decomposition method, selecting variables for each pair of classes and then extend the results to multiclass case [18,19,20] or on a direct approach, considering all classes simultaneously [21,22].

However, in spite of the significant number of proposed extensions to multiclass case and their good performances compared to some existing techniques, no method is best or optimal [22,23] and the issue is still relevant. For this reason, and in order to contribute in this framework, we propose this article.

Indeed, studying the various extensions, we note that although the theoretical bases and good performances of Rakotomamonjy's approach [11] in selecting variables in binary case, no study, to our knowledge, has used an upper bound of generalization error proper to multiclass case to select the sub-optimal set of variables.

In this article, we propose a new method for ranking and selecting relevant variables in multiclass case, based on the upper bound of generalization error called radius margin bound (RM) [1]. This bound is specific to multiclass case and only applicable to hard margin $MSVM_{LLW}$ model [14] i.e. without training error and to $MSVM^2$ model of guermeur et al. [1]. In this work we will use the first model.

The RM bound is proposed for model selection [1]. The contribution of this paper is to use it for model and variables selection.

The proposed method consists of three steps: firstly, we choose the parameters of hard margin $MSVM_{LLW}$ model minimizing the multiclass RM bound in presence of all variables (model selection), then, we classify variables in order of relevance, and finally, proceeding by forward method, we choose the optimal subset minimizing the testing error, calculated on a sample or by cross-validation.

The rest of the paper is organized as follow: section 2 presents the $MSVM_{LLW}$ model and the RM upper bound of generalization error. The proposed procedure for variable ranking and optimal subset' selection is given in section 3. The data used, results of experiments and comparisons are presented in Section 4, followed by a general conclusion and perspectives of this work.

2 $MSVM_{LLW}$ MODEL AND RM BOUND

In the framework of multiclass SVM (MSVM), we are interested in q categories classification problems ($2 < q < \infty$). The goal is to estimate q decision function $f_k(x)$ and classify observations according to the classification rule $\Phi_f(x) = \arg\max f_k(x)$ with k=1, 2,…,q.

The estimation of the decision functions is done using a set of pairs of independent and identically distributed observations $\{(x_i,y_i), i=1,\ldots,n\}$ called training set, where x is the description of an object belonging to the descriptions space X described by 'p' variables and Y the set of categories 'y' identified by their indices [1, q].

Several approaches were proposed in the context of MSVM, belonging either to the category of decomposition or direct methods [24].

In this work, we will test the performances of the RM bound to perform variables selection for a hard margin $MSVM_{LLW}$ model. This section will briefly present the properties of this model and describes the RM bound.

2.1 The $MSVM_{LLW}$ Model

As all direct approaches [1,25,26], the $MSVM_{LLW}$ model solves the multiclass problem directly without decomposition, estimating 'q' decision functions simultaneously by solving one optimization program. It is considered as the most theoretically based of MSVM models as is the only one that implements asymptotically the Bayes decision rule [14].

The optimization problem is to solve, subject to the constraint $\sum_{k=1}^{q} f_k = 0$, the objective function of the form:

$\min_f \quad \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{q} I(y_i \neq k)[f_k(x_i) + 1]_+ + \lambda \sum_{k=1}^{q} \sum_{j=1}^{p} w_{kj}^2,$ (1)

The first term $I(y_i \neq k)[f_k(x_i) + 1]_+$ represents the loss function, which measures the difference between estimations and reality. This term can also be written as $C \sum_{i=1}^{n} \sum_{k \neq y_i} \xi_{ik}$, with ξ_{ik} the slack variables and C the weight of these variables.
The second term $\lambda \sum_{k=1}^{q} \sum_{j=1}^{p} w_{kj}^2$ with $\lambda \in R$ determined by cross-validation, and $\sum_{k=1}^{q} \sum_{j=1}^{p} w_{kj}^2$ measures the ability or the complexity of the hypothesis space, and also equal to the inverse of k separators' margins to maximize.

$f_k(x) = \langle w_k, \Phi(x_i) \rangle + b_k$, $1 \leq i \leq n$,
(w_k, b_k) the parameters of k^{th} separator to estimate.
$\Phi(x_i)$ the nonlinear transformation of x_i from original to feature space if data are not linearly separable. If not, $\Phi(x_i) = x_i$.

Problem solving is done using the Lagrangian, and the nonlinear transformation of data will be replaced by a kernel function.

2.2 Multiclass Radius Margin Bound

The RM upper bound of the generalization error that we will use is a direct extension of the two-class radius margin bound [27]. Used for model selection, it is considered as the easiest and the most popular of generalization error's upper bounds.
Guermeur et al. [1] demonstrate that the number of errors denoted L_m, resulting from the application of leave-one-out cross-validation procedure (LOOCV) for a hard margin q-category $MSVM_{LLW}$ trained on d_m, is upper bounded as follows:

$$L_m \leq \frac{(q-1)^3}{q} D_m^2 \sum_{i=1}^{m} max_{1 \leq k \leq Q} \alpha_{ik}^* \quad (2)$$

with,
m : the size of training sample,
q : the number of categories,
D_m : the diameter of the smallest sphere containing the dataset in original or feature space.
α_{ik}^* : the Lagrange parameters resulting from the resolution of the optimization program (1).

Since the value given by LOOCV is an almost unbiased estimator of the generalization error, a variable is considered as relevant according to its influence on this error by measuring its contribution to minimize the second term of (2) which is the RM bound.

3 THE PROPOSED PROCEDURE FOR VARIABLE RANKING AND OPTIMAL SUBSET' SELECTION

RM bound is generally used for model selection; it means to choose the optimal parameters of MSVM model. These parameters to optimize are: C, representing the weight of training errors ξ, and σ the parameter of the kernel function if we decide to change the data space.
Note that a large value of C means a big weight of errors and thus get closer to a hard margin learning, and, conversely, a small value reflect acceptance of errors and therefore a soft margin learning.
The idea in this article is to use the RM bound to model and variables selection. The proposed procedure is based on a score called zero-order score proposed for two class problems [11], whose value will rank variables in order of relevance.
The zero-order score of a variable is the value of the RM bound when this variable is removed. The variable whose suppression greatly increases the value of the bound and therefore, contributes to the minimization of the generalization error is considered as most relevant.
The RM bound (2) depends on three factors: the number of categories q, the diameter D_m of the smallest sphere containing data and Lagrange parameters α_{ik}^*.
The first element 'q' is constant and independent of the number of variables, in contrast to the two other parameters D_m and α_{ik}^*. Indeed, an object is represented by its coordinates in space, so its position changes necessarily by removing a variable and thus the diameter of the sphere. Similarly, when removing a variable, data which are inputs to estimate the model change, and therefore α_{ik}^*, model's outputs, change too. So, the research of relevant variables will be based on the product:

$$D_m^2 \sum_{i=1}^{m} max_{1 \leq k \leq q} \alpha_{ik}^* \quad (3)$$

Once the order of relevance of variables established, we proceed by forward method, as described in [28], incorporating a variable at a time in decreasing order of relevance and we choose the subset giving the minimum error rate calculated on a test sample or by cross-validation.

The proposed procedure for ranking and selecting relevant variables, for a hard margin $MSVM_{LLW}$ model, follows the three following steps:

Step 1: Choice of the model's parameters

In this step, we choose the parameters of the hard margin $MSVM_{LLW}$ model which minimize the RM bound, and therefore the generalization error, in presence of all variables. These parameters that will be used in the next step to rank variables.

To do this, we first work in original space using a linear kernel (K = 1). In this case, there will be only the parameter C to determine, as described in the function F1 bellow.

If we are unable to work without training error or if the required time is very important, we proceed to change the data space and work with a Gaussian kernel. In this case (F2), we have to set the values of the two parameters C and σ.

The algorithmic description of this step is as follows:

Table 1. Parameters and variables definition for variable selection by RM bound algorithm

- n : number of observations
- p : number of variables
- k : kernel function
- k=1 : linear kernel
- k=2 : gaussien kernel
- D_m : diameter of the smallest sphere enclosing data
- E : training Error
- t : learning time
- s : maximum learning time allowed by user
- σ_m : maximal value of Gaussian kernel's parameter (determined by user)
- TER: testing error rate
- TDB: testing data base
- C_{min} and C_{max} : minimal and maximal value of C chosen by user
- σ_{min} and σ_{max} : minimal and maximal value of σ chosen by user
- C^*, σ^*: optimal values of C and σ (chosen in step 1).

F1: Parameters determination K=1

Inputs
Training examples $X \epsilon R^p$, with
$X= [x_1, x_2, .., x_n]^T$
Class labels $Y= [y_1, y_2, \ldots y_n]^T$
Initialize
k=1
Compute D_m with hard margin SVDD algorithm
Loop
for $C= C_i$ with i=1 to c **do**
$[E, t, \sum_{i=1}^{m} max_{1 \le k \le Q} \alpha_{ik}^*]^T =$
Train $MSVM_{LLW}(X,Y,C)$
end for
Output
Choose $C^* = C_i$ with E=0 , T<s and argmin (RM)
if C^* exist **then**
goto F3
else
goto F2
end if

F2: Parameters determination K=2

Inputs
Training examples $X \epsilon R^p$, with
$X= [x_1, x_2, , x_n]^T$
Class labels $Y= [y_1, y_2, \ldots y_n]^T$
a. Initialize
K=2
C
b. Loop
for σ = 1 à σ_m **do**
$[E, t, \sum_{i=1}^{m} \max_{1 \le k \le Q} \alpha_{ik}^*]^T =$
Train $MSVM_{LLW}(X,Y,C,\sigma)$
Compute D_m [σ]
Compute $RM[\sigma] = D_m^2 \sum_{i=1}^{m} \max_{1 \le k \le Q} \alpha_{ik}^*$
if E=0 and t < s **then**
Increment σ
else
goto a
end if
end for
Outputs
$C^* = C$
$\sigma^* = argmin(RM[\sigma])$

A definition of the parameters and the variables of the 4 algorithm's blocks (F1, F2, F3, F4) is given in table 1.

Step 2: Variables ranking

In this step, we rank variables in order of relevance according to the values of their zero-order scores. For this, we re-estimate, removing each time a variable, the $MSVM_{LLW}$ model and we compute the value of the product $D_m^2 \sum_{i=1}^{m} max_{1\le k\le q}\ \alpha_{ik}^*$. (F3).

F3: Variables ranking
Inputs
Training examples $X \in R^p$, with
$X= [x_1, x_2, .., x_n]^T$
Class labels $Y= [y_1, y_2, \ldots y_n]^T$
Loop
for j=1 to p **do**
remove X[i,j]
$\sum_{i=1}^{m} \max_{1\le k\le Q} \alpha_{ik}^* =$Train $MSVM_{LLW}$(X,Y,C*,σ*)
Compute Dm
Compute RM[j]=$D_m^2 \ \sum_{i=1}^{m} \max_{1\le k\le Q}\ \alpha_{ik}^*$
end for
Variable Ranking
Classify RM[j] by decreasing order and recuperate j
Classify the matrix X by the new order of j

Step 3: Choice of the optimal sub set of variables

After having the order of relevance of variables, the last step is to choose the optimal subset. For this, we construct, by forward method, a sequence of models, with the first one contains the first relevant variable, the second one contains the first two relevant variables and so on until all variables are integrated in descending order of relevance. Then we calculate the testing error rates. The model giving the minimum error rate is chosen as the best model with the optimal number of variables (F4).

F4: Selection of optimal subset
Initialize
j=1
M= X[i,1]
Loop
While (j≤ p) **do**
for C = C_{min} to C_{max} **do**
for σ_{min} to σ_{max} **do**
TER= $MSVM_{LLW}$ (TDB)
end for
end for
R[j] = argmin (TER)
Increment j
M ⟵ M + X[i,j]
end while
Output
Tmin ⟵ argmin(R[j])
Recuperate j

Note that for the first two stages, model selection and variables ranking, we must work without training error, as these two steps are based on the RM bound which is applicable to a hard margin $MSVM_{LLW}$ model. By contrast, it is not mandatory to do so in step 3, because we no longer use the RM bound, so we do simulations with combinations of values of C and σ until we find the values that minimize the testing error. Also, we insist on the idea that the biggest contribution of this article is in giving the order of relevance of variables which was not been done on multiclass case with direct approaches of MSVM before, this means that we can change the third step and use another method to select the optimal subset from the order given in the second step, here we use forward method but backward method or other procedures can be used.

4 RESULTS OF EXPERIMENTS AND COMPARISONS

In this section, we present the tests showing the ability of the score based on the RM bound to rank the variables and, therefore, to select the optimal subset. Five datasets are considered, including three simulated databases and two real sets.
For all data sets, several simulations are conducted to find the parameters of the $MSVM_{LLW}$ model minimizing the RM bound (step 1) and to select the optimal subset (step 3).
Simulations and results have been obtained using the MSVMpack of Lauer et al. [29] allowing to train the $MSVM_{LLW}$ model and giving the parameters α_{ik}^* as output.
The diameter D_m of the smallest sphere containing data has been calculated using the hard margin SVDD algorithm [30].

4.1 Simulated Data

The used data are linearly separable in original or features space. For each case, n_1 observations are generated as training set and n_2 observations as testing set. Each observation is described by p variables (x^1, x^2, ..., x^p) with 2 are relevant and the others are noise variables.
The 2 relevant variables are generated from a mixture Gaussian as follows: for each class k, 2

variables are generated independently from N(μ_k ,$\sigma^2 I_2$), with $\sigma = \sqrt{2}$ and for k=1,2,..q:

$$\mu_k = 2(\cos([2k-1]\,\pi/q), \sin([2k-1]\,\pi/q)), \quad (4)$$

The remaining variables are independent and identically distributed generated from N(0 ,1).

4.1.1 Example 1

The data of this first example are those described by Zhang et al. [15], with n_1= 250 observations, n_2= 50000, q= 5 equally weighted classes (each class has the same number of observations) and p=10 variables with (x^1, x^2) are relevant and (x^3, x^4,…, x^{10}) are 8 noise variables.
To estimate the parameters of the hard margin $MSVM_{LLW}$ model in the presence of all variables, we first worked with a linear kernel.
The results show that this kernel did not allow to train model without error. We then tried a Gaussian kernel which gave a zero training error.
The model estimation using a Gaussian kernel requires setting the values of parameters C and σ. For C, high values are used in order to penalize errors and therefore obtain a hard margin model. Simulations showed that the value C=1000 allows to work without error for different values of σ.
To set the value of σ, we conducted several simulations to select the value that, keeping zero training error, minimizes the generalization error via its upper bound RM= $\frac{(q-1)^3}{q} D_m^2 \sum_{i=1}^{m} max_{1\le k\le q}\, \alpha_{ik}^*$. The term $\frac{(q-1)^3}{q}$ being constant, we choose the value that minimizes the product $D_m^2 \sum_{i=1}^{m} max_{1\le k\le q}\, \alpha_{ik}^*$.
The simulations' results are described in Table 2.

Table 2. Values of $D_m^2 \sum_{i=1}^{m} max_{1\le k\le Q}\, \alpha_{ik}^*$ in terms of the values of σ

σ	$\sum_{i=1}^{m} max_{1\le k\le q}\, \alpha_{ik}^*$	D_m	$D_m^2 * \sum_{i=1}^{m} max_{1\le k\le q}\, \alpha_{ik}^*$
0.5	312.4004	15.805	78 037.995
1.0	315.7785	14.027	62 131.820
1.5	413.478	9.180	34 841.310
2.0	771.132	5.984	**27 609.610**
2.5	1862.703	4.214	33 081.605
3.0	4315.325	3.220	44 741.290
4.0	19595.13	2.223	96 799.942
5.0	63546.20	1.780	201 390.617
6.0	114203.68	1.542	271 576.351
7.0	144505.89	1.398	282 480.114

The minimum of the upper bound is reached for σ = 2. The model will therefore be estimated with C=1000 and σ = 2.
The second step is to test the ability of the zero-order score to give the order of relevance of the variables.
To do this, we calculated, each time removing a variable, the $\sum_{i=1}^{m} max_{1\le k\le Q}\, \alpha_{ik}^*$ and the diameter D_m of the smallest sphere enclosing data in the feature space, using the parameters chosen in step 1. The results are reported in table 3.

Table 3. Zero-order score of the 10 variables

Removed Variable	$\sum_{i=1}^{m} max_{1\le k\le Q}\, \alpha_{ik}^*$	D_m	Zero-order scores (D_m^2 * $\sum_{i=1}^{m} max_{1\le k\le Q}\, \alpha_{ik}^*$).
x^1	2010.701	4.841	**47 116.914**
x^2	1814.651	4.854	**42 759.475**
x^3	1158.179	5.580	36 063.672
x^4	1192.931	5.571	37 026.283
x^5	1137.751	5.504	34 470.632
x^6	1113.298	5.564	34 468.209
x^7	1173.368	5.504	35 543.780
x^8	1023.852	5.504	31 013.183
x^9	1070.409	5.554	33 023.519
x^{10}	1140.087	5.633	36 181.583

The most relevant variable is the one that maximizes the value of the zero-order score which is equal to the value of the product $D_m^2 \sum_{i=1}^{m} max_{1\le k\le Q}\, \alpha_{ik}^*$ when the variable is removed. The order of relevance of variables obtained according to table 3 is as follows:

x^1, x^2, x^4, x^{10}, x^3, x^7, x^5, x^6, x^9, x^8

The proposed score has successfully classify the first two variables which are most relevant in the two first ranges.
After ranking variables, we have estimated models to select the one that minimizes testing error and gives optimal sub-set of variables. For this, we built 10 databases: the first contains the first relevant variable, the second contains the first two relevant variables and so on until integrating all variables one by one in the decreasing order of relevance.
Training and testing errors obtained according to used variables on a test sample of 50000 observations with C = 10 and σ = 2 are shown in Table 4.

Table 4. Training and testing errors

Used Variables	Training error %	Testing error %
(x^1)	56.80	61.18
(x^1, x^2)	34.00	**40.10**
(x^1, x^2, x^4)	29.60	41.33
(x^1, x^2, x^4, x^{10})	24.80	42.88
(x^1, x^2, x^4, x^{10}, x^3)	17.60	44.78
(x^1, x^2, x^4, x^{10}, x^3, x^7)	10.80	46.64
(x^1, x^2, x^4, x^{10}, x^3, x^7, x^5)	4.00	48.26
(x^1, x^2, x^4, x^{10}, x^3, x^7, x^5, x^6)	1.60	49.75
(x^1, x^2, x^4, x^{10}, x^3, x^7, x^5, x^6, x^9)	0.00	51.48
(x^1, x^2, x^4, x^{10}, x^3, x^7, x^5, x^6, x^9, x^8)	0.00	50.39

the minimum testing error 40.10% obtained using the subset of the first two relevant variables is lower than that obtained using all variables (50.39%).

- ***Comparison with the results of other methods of selection by MSVM***

Zhang et al. compared in [15] the results of 6 MSVM methods (Table 5), on the first data set used above, in terms of testing errors. L_2 method represents the model $MSVM_{LLW}$ i.e. a classification method without variable selection, while the 5 other methods include selection in the classification process. Since our first data set was generated according to the same procedure, we compare our result in Table 4 to those in Table 5 [15].

Table 5 shows that for the six methods, the best testing error obtained, on 250 training observations and 50000 testing ones, is 45.3% using the Supnorm method. With our procedure, we obtained a better error rate: 40.1% using the first two relevant variables (Table 4).

Table 5. Testing error rates of 6 methods for the first dataset

Method	testing error rates
L2	0.454 (0.034)
L1	0.558 (0.022)
Adapt-L1	0.553 (0.020)
Supnorm	**0.453 (0.020)**
Adapt-supI	0.455 (0.024)
Adapt-supII	0.457 (0.046)
Bayes	0.0387 (---)

4.1.2 Example 2

In the simulation example in section 4.1.1, the relevant variables are the first two ones. The goal of this section is to make sure that the position of variables and the number of classes do not affect the performances of the score. For this, we apply our selection procedure to a second data set with n_1= 300 observations, n_2 = 30000 observations, q=3 equally weighted classes and p=10 variables with x^3 and x^9 are both relevant and the other 8 are noise variables.

The numerous simulations have allowed to choose the type of kernel (Gaussian) and to set the values of parameters C and σ (C = 1000 and σ = 2) that allow to work without training error, require a reduced calculation time and minimize the value of the RM bound. Table 6 presents the order of relevance of variables according to the values of zero-order score and reveals that the score has successfully classified x^3 and x^9,which are the most relevant, in the first two rows.

Table 6. Ranking of variables according to their zero-order scores

Removed Variable	$\sum_{i=1}^{m} max_{1\le k\le Q} \alpha_{ik}^*$	D_m	Zero-order scores	Ranking
x^1	1 020.699	5.126	26 817.172	10
x^2	993.915	5.199	26 867.323	9
x^3	2 760.869	5.884	**95 581.726**	**1**
x^4	1 006.519	5.836	34 282.656	5
x^5	974.648	5.905	33 989.801	6
x^6	959.529	5.839	32 709.998	8
x^7	975.099	5.931	34 302.062	4
x^8	979.815	5.927	34 419.439	3
x^9	2 240.179	5.866	**77 078.090**	**2**
x^{10}	987.095	5.842	33 686.352	7

Table 7, gives the testing error rates obtained in terms of used variables, in decreasing order of relevance, on a testing sample of 30000 observations with an appropriate combinations

Table 7. Training and testing errors in terms of used variables

Used Variables	Training error %	Testing error %
(x^3)	38.33	66.48
(x^3, x^9)	16.00	66.78
(x^3, x^9, x^8)	11.33	**57.71**
(x^3, x^9, x^8, x^7)	6.33	61.34
(x^3, x^9, x^8, x^7, x^4)	1.00	60.31
(x^3, x^9, x^8, x^7, x^4, x^5)	0.00	64.70
(x^3, x^9, x^8, x^7, x^4, x^5, x^{10})	0.00	62.61
(x^3, x^9, x^8, x^7, x^4, x^5, x^{10}, x^6)	0.00	63.23
(x^3, x^9, x^8, x^7, x^4, x^5, x^{10}, x^6, x^2)	0.00	64.24
(x^3, x^9, x^8, x^7, x^4, x^5, x^{10}, x^6, x^2, x^1)	0.00	65.68

of C and σ. (several simulations with different combinations of C and σ were conducted, the

combination giving the minimum testing rates was chosen).
The results show that the minimal testing error of 57.71% obtained using the third subset of variables is better than that obtained using all available ones: 65.68%.

4.1.3 Example 3

After having successfully ranked the 2 relevant variables in presence of 8 noise ones in sections 4.1.1 and 4.1.2, we proceed, here, to measure the effect of increasing the number of irrelevant variables on the performances of the score.
For this, we generate a third dataset with the same characteristics as the first one, except in terms of the number of noise variables which moves from 8 to 98 variables.
To set the values of C and σ, we conduct several simulations using 3 values of C: 10 000, 1 000 and 100 and 17 values of σ to choose the best combination: (C = 100 and σ = 10).
The results of variables ranking based on the values of zero-order scores are presented in Figure 1.

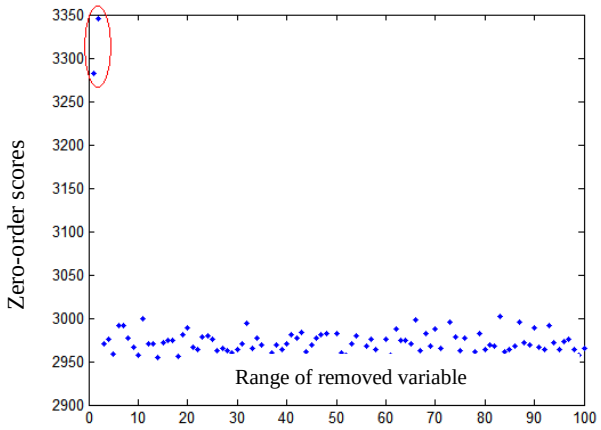

Figure 1. Zero-order scores of the 100 variables

According to Figure 1, the maximal values of zero-order score are those of the first two variables that we know they are the most relevant. Thus, the proposed score has been able to classify the first two variables in the first two positions in large dimensions.

4.2 Real Data

4.2.1 Example 1

In this example, we apply our selection procedure to the real data set 'lung cancer' which is composed of 56 variables, 32 observations and 3 classes. These data, available at UCI repository, were presented for the first time by Hong et al. [31] and have been used in the context of variable selection by multiclass SVM by LI et al. [32].
Using their proposed method, Li et al obtained, an average testing error rate of 45.8%, while, the application of Optimal Brain Damage method on this dataset gave an average testing error rate of 44.15%.
Our main goals here are: firstly, show the performances of our procedure to select the optimal subset and give a better testing rate relative to the case where all variables are conserved, and secondly, compare the obtained testing error rate to those of the two methods described above.
For experiments, we use the 32 observations as a training set and we calculate testing error rates by LOOCV.
Figure 2 gives the variables ranking according to their zero order scores calculated using a linear kernel with C = 1000.

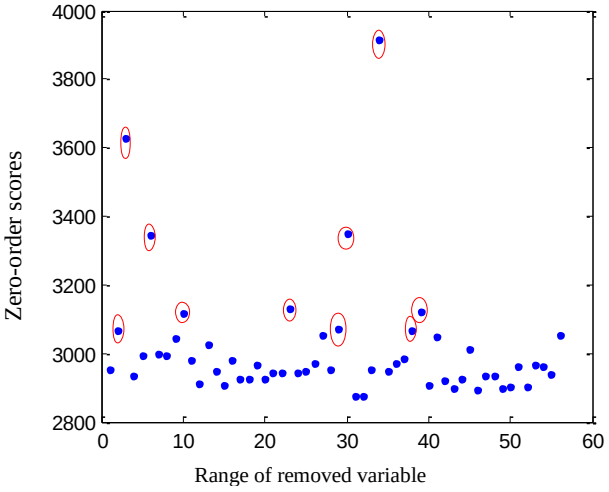

Figure 2. Zero-order scores of variables

The first 10 variables, in order of relevance, according to Figure 2 are:

$$x^{34}, x^{3}, x^{30}, x^{6}, x^{23}, x^{39}, x^{10}, x^{29}, x^{38}, x^{2}$$

To select optimal model, we build 56 databases. For each one, in order to minimize error rates, we try different values of C. Figure 3 shows the results of estimation of testing error rates by LOOVC according to values of C and the number of used variables in decreasing order of relevance.

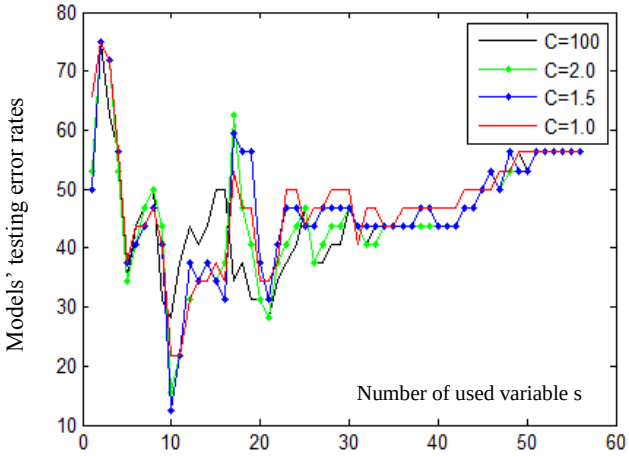

Figure 3. Testing error rates of the 56 models

The best testing error rate is 12.5% obtained using the first 10 variables in order of relevance with C=1.5. This rate is much better than that obtained in presence of all variables i.e. 56.25%. So we can confirm the effectiveness of the proposed procedure for ranking and selecting the optimal subset.
Furthermore, comparing this result to those of two variables selection methods by MSVM described before, we find that the achieved rate of 12.5% is much better.

4.2.2 Example 2

The second example on which we test the performances of our approach is the children cancer data set classifing the small round blue cell tumors (SRBCTs) into four classes, namely neuroblastoma (NB), rhabdomyosarcoma (RMS), non-Hodgkin lymphoma (NHL), et Ewing (EWS) using cDNA gene expression profiles.(http://research.nhgri.nih.gov/microarray/Supplement/).
The data set includes 83 observations and 2308 variables. It was presented for the first time by Khan et al. [33] and has been used in the context of variable selection by multiclass SVM by Zhang et al. [15].
After model selection, we proceed to rank features in order of relevance, Figure 4 gives the zero order scores of the 2308 variables.
To estimate the testing error rates, we used 10 combinations of C and σ (C: 100, 10, 1 and σ: 1, 2, 3, 2.5) to select the best combination(s) giving the minimum testing error rate.
The results of the two best simulations in terms of testing error rates obtained by LOOCV according to the values of C, σ and the number

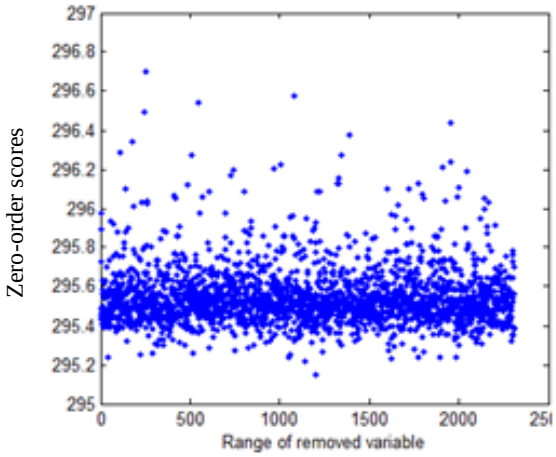

Figure 4. Zero-order scores of variables

of used variables in decreasing order of relevance are presented in Figure 5.

From Figure 5, we note that the proposed approach based on RM bound gave a zero error rate for the two simulations using the first 9 variables.

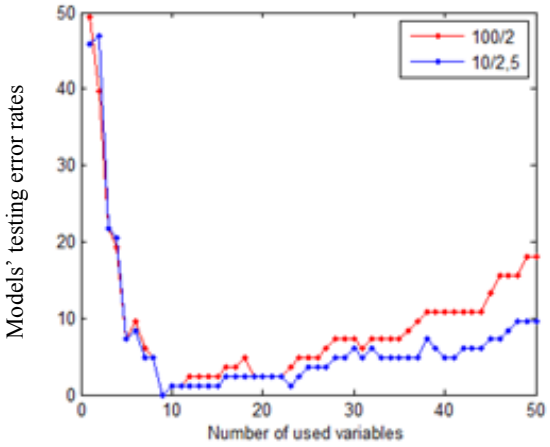

Figure 5. Testing error rates of the first 50 models

Comparing these results with those of previous studies on this dataset, we can see that they are far better in terms of the number of variables needed. Indeed, selecting variables for multiclass SVM using adaptive sup-norm regularization, Zhang et al. [15] obtained zero error rate, on a test sample, using 47 variables and using the L1 norm, 62 variables were required. Thus, using neural networks, get a zero error rate was possible but using the first 96 genes.

5 CONCLUSION AND PERSPECTIVES

The results of studies on variable selection by multiclass SVM show the effectiveness of using

this technique to reduce dimensions and improve classification's accuracy.

In this paper, we have proposed a new approach, based on the radius margin upper bound of generalization error, to give the order of relevance of variables and the optimal subset. As a result, the proposed method gives the correct order of relevance of variables for the simulated data, and significantly reduces the error rate for all used data sets.

In fact, one of the advantages of our method is that it uses the $MSVM_{LLW}$ model which is the most theoretically based of MSVM models and as a wrapper approach, selecting variables after the estimation of the model, takes into account the influence of each variable on the performances of the estimated model.

A constraint for the application of our procedure, since it uses a hard- margin model, is that the data must be linearly separable in original or feature space. As future work, and for non-linearly separable data, we will test the performances of the RM bound using $MSVM^2$ model of Guermeur et al. which is considered as a variant of $MSVM_{LLW}$ model with soft margin.

Another constraint consists in required computation time which is relatively large in very large dimensions, given the need to re-estimate the prediction model several times to calculate α. To deal with this problem, we propose to remove a set of variables each time, instead of removing one by one.

6 ACKNOWLEDGMENTS

We would like to acknowledge support for this work from CNRST-Ministry of Higher Education, Scientific Research And Executive Training of Morocco and Rhône-Alpes Region/France by according to AAZI F.Z, respectively, a research scholarship and accueil-DOC 2012 and 2013 scholarships.

We wish to especially thank Hassan 1st University and express our gratitude for its financial support and finally, the authors would like to thank Professor F. Lauer for his valuable help and for providing them with additional information on his work.

7 REFERENCES

[1] Y. Guermeur and E. Monfrini, "A Quadratic Loss Multi-Class SVM for which a Radius–Margin Bound Applies". Informatica, Vol. 22, No. 1, pp.73–96. 2011.

[2] A. Blum and P. Langley, "Selection of relevant features and examples in machine learning". Artificial Intelligence, 97(1-2) :pp. 245-271. 1997.

[3] I. Guyon, A. Elisseeff, and L. Kaelbling. "An Introduction to Variable and Feature Selection". Journal of Machine Learning Research, 3(7-8) :pp. 1157-1182. 2003.

[4] R. Kohavi and G. John. "Wrapper for feature subset selection", Artificial Intelligence 97(1–2): pp. 273–324. 1997.

[5] J. Weston, A. Elisseeff, B. Schlkopf and P. Kaelbling. "Use of the zero-norm with linear models and kernel methods". Journal of Machine Learning Research, 3:pp. 1439-1461. 2003.

[6] P.S. Bradley and O.L. Mangasarian. "Feature selection via concave minimization and support vector machines". Machine Learning Proceedings of the Fifteenth International Conference (ICML 98), pp. 82-90. Morgan Kaufmann. 1998.

[7] J. Zhu, S. Rosset, T. Hastie and R. Tibshirani. "1-norm support vector machines". Neural Information Processing Systems. MIT Press. 2003.

[8] Y. Liu and Y. Wu. "Variable selection via a combination of the L_0 and L_1 penalties". Journal of Computation and Graphical Statistics. 2007.

[9] H. Zou and M. Yuan. "The f∞-norm support vector machine". Statistica Sinica. 2008.

[10] I. Guyon, J.Weston, S. Barnhill, and V. N. Vapnik. "Gene selection for cancer classification using support vector machines". Machine Learning, 46(1-3) : pp. 389-422, 2002.

[11] A. Rakotomamonjy, "Variable Selection Using SVM-based Criteria". Journal of Machine Learning Research vol. 3, pp. 1357-1370. 2003.

[12] L. Wang and X. Shen, "Multi-category support vector machines, feature selection and solution path". Statistica Sinica 16. pp. 617-633. 2006.

[13] L. Wang and X. Shen, "On l1-norm multi-class support vector machines: methodology and theory". Journal of the American Statistical Association. pp.583–594. 2007.

[14] Y. Lee, Y. Lin, and G. Wahba. "Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data". Journal of the American Statistical Association, 99(465):pp. 67-81, 2004.

[15] H.H. Zhang, Y.Liu, Y.Wu and J. Zhu, "Variable selection for the multicategory SVM via adaptive sup-norm regularization". Electronic Journal of Statistics, Vol. 2, pp. 149–167. 2008.

[16] J. Guo, "Class-specific Variable Selection for Multicategory Support Vector Machines", Statistics and its interface. 2011.

[17] J-T. LI and Y-M. JIA, "Huberized Multiclass Support Vector Machine for Microarray Classification". Acta Automatica Sinica, Vol. 36, No. 3. 2010.

[18] M-D. Shieh and C-C. Yang, "Multiclass SVM-RFE for product form feature selection", Expert Systems with Applications, vol 35. pp. 531–541. 2008.

[19] X-W. Chen , X. Zeng, and D.V. Alphen, "Multi-class feature selection for texture classification". Pattern Recognition Letters, 27. pp. 1685–1691. 2006.

[20] Y. Mao, X. Zhou, D. Pi, Y. Sun, and S.T.C. Wong, "Multiclass Cancer Classification by Using Fuzzy Support Vector Machine and Binary Decision Tree With Gene Selection". Journal of Biomedicine and Biotechnology. 2. pp 160–171. 2005.
[21] O. Chapelle and S. Keerthiy. "Multi-Class Feature Selection with Support Vector Machines". 2008.
[22] X. Zhou and D.P. Tuck. "MSVM-RFE: extensions of SVM-RFE for multiclass gene selection on DNA microarray data". Bioinformatics Vol. 23 No. 9, pp. 1106–1114. 2007.
[23] P.M. Granitto and A. Burgos, "Feature selection on wide multiclass problems using OVA-RFE". Inteligencia Artificial vol 44. pp 27-34. 2009.
[24] Y. Guermeur. "SVM multiclasses, théorie et applications". Habilitation à diriger des recherches, Université Nancy 1, 2007.
[25] J. Weston and C. Watkins. "Multi-class support vector machines". Technical Report CSD-TR-98-04, Royal Holloway, University of London, Department of Computer Science. 1998.
[26] K. Crammer and Y. Singer. "On the algorithmic implementation of multiclass kernel based vector machines". Journal of Machine Learning Research, 2: pp. 265_292, 2001.
[27] V.N. Vapnik. "The Nature of Statistical Learning Theory". Springer-Verlag, New York, 1995.
[28] A. Ben Ishak, "Sélection des variables par les machines à vecteurs supports pour la discrimination binaire et multiclasse en grande dimension". Université de la Méditerranée (Aix-Marseille II)/ Université de Tunis, 2007.
[29] F. Lauer and Y. Guermeur, "MSVMpack: a Multi-Class Support Vector Machine package". Journal of Machine Learning Research vol. 12, pp. 2293-2296. 2011.
[30] D M.J. Tax and R. P.W. Duin, "Support Vector Data Description", Machine Learning, 54, pp. 45–66, 2004
[31] Z.Q. Hong and J.Y. Yang. "Optimal discriminant plane for a small number of samples and design method of classifier on the plane". Pattern recognition .1991.
[32] G-Z Li, J. Yang, G-P Liu and L. Xue. "Feature Selection for Multi-Class Problems Using Support Vector machines". Lecture Notes on Artificial Intelligence. 3173 (PRICAI2004), pp. 292-300. Springer. 2004.
[33] 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. pp. 673-679. 2001.