International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:4, 2010

Genetic Algorithms and Kernel Matrix-based Criteria Combined Approach to Perform Feature and Model Selection for Support Vector Machines

A. Perolini

Abstract**—**Feature and model selection are in the center of attention of many researches because of their impact on classifiers' performance. Both selections are usually performed separately but recent developments suggest using a combined GA-SVM approach to perform them simultaneously. This approach improves the performance of the classifier identifying the best subset of variables and the optimal parameters' values. Although GA-SVM is an effective method it is computationally expensive, thus a rough method can be considered. The paper investigates a joined approach of Genetic Algorithm and kernel matrix criteria to perform simultaneously feature and model selection for SVM classification problem. The purpose of this research is to improve the classification performance of SVM through an efficient approach, the Kernel Matrix Genetic Algorithm method (KMGA).

Keywords**—**Feature and model selection, Genetic Algorithms, Support Vector Machines, kernel matrix.

I. Introduction

THIS paper tackles feature and model selection problem for classification task through SVM. Due to the performance of SVM depends on the choice of parameters and variables [1]-[5] selecting the most predictive features and the right values of kernel parameters will improve classifiers performance. International literature suggests several approaches to improve the discrimination ability of classification methods focusing the attention on separate [6]-[14] or joined [1], [2], [11], [15]-[20] feature and model selection approaches. Most of the proposed methods belong to one of the two problems thus for each method two important shortcomings arise. How to select features? How to choose parameters' values?

Evolutionary Algorithms answer these questions in an efficient way. In fact, to reach high accuracy classification methods need to estimate parameters values and to select the relevant variables. The main issue of these approaches is that they are applied separately. This means that some assumptions have to be introduced before employing them. Anyway, in this context a general method like Genetic Algorithm can be helpful. Evolutionary Algorithms (and GA) took place in the Artificial Intelligence field and rapidly develop due to their simplicity and effectiveness. GAs are based on the principle of creatures' evolution: different operators reproduce the biological functions that allow individuals to evolve accordingly to a specific criterion (i.e. a fitness function). A whole population evolves, the best individuals that survive the "natural" selection will contribute to the development of the next generation until an optimal solution is reached or stopping rules end the search. The idea of Genetic Algorithm applied to Machine Learning problems to solve feature and model selection for SVM classification task is recent [1], [21]-[24].

In this paper an efficient approximated method (Kernel matrix Genetic Algorithm) is proposed to perform feature and model selection. Three kernel matrix-based criteria are analyzed to discover whether they can lead the evolutionary process to find the best subset of features and the best kernel's parameters for SVM classification problem.

The paper is organized as follows. Section II gives a brief introduction to Feature Selection and Model Selection problems for SVM providing some remarks about their combined approach. Section III describes the Genetic Algorithm and the SVM methods explaining the details of the GA-SVM joined approach. Section IV provides the detail of kernel matrix approach and section V the experiments conducted on four datasets.

II. Feature and Model Selection for SVM

A. Feature Selection

In machine learning problems feature selection is used to identify predictive variables and to reduce the dimension of the dataset removing irrelevant or highly correlated ones. This process has strong impacts on classification task because it allows to improve classifiers' predictive and generalization ability and to reduce the computational time [2], [8], [10]-[12].

Many researchers [1], [2], [8], [10]-[12], [17], [24] investigate the factors that influence feature selection process through several approaches. Weston et al. [8] propose an optimization approach based on gradient descent. Moreover they underline the implications of irrelevant features in

A. Perolini is with the Dipartimento di Ingegneria Gestionale, Politecnico di Milano, P.za Leonardo da Vinci 32, Milano 20133 Italy (phone: 390223993998; e-mail: alessandro.perolini@polimi.it).

classification process proving the negative effects of these variables on SVM classification performance. Reunanen [12] explains the limits of traditional feature selection methods investigating two approaches: Sequential Forward Selection (SFS) and Sequential Forward Floating Selection (SFFS). Comparing state-of-art methods Guyon and Elisseeff [11] discuss the problem of feature selection highlighting drawbacks and advantages. Rakotomamonjy [2] extends the SVM-RFE method proposed by [10] analyzing three SVM bounds. Cheng et al. [17] focus their research on feature selection with nonlinear relation between variables proposing the Relevance Feature Vector Machine (RFVM). Further researches suggest the use of evolutionary approaches. Fröhlich et al. [1] describe a feature selection method based on GA and SVM generalization error bounds. Variables are evaluated estimating an expected generalization error of the SVM. The main advantage of this procedure is its speed. In fact, the computation of bounds is faster than training n times a SVM. Tan et al. [24] suggest a hybrid method between GA and SVM with an improvement based on correlation filter approach to overcome the limitations of unfeasibility for large size problems and high risk of local optima falling of established methods (hill climbing and best-first search).

Despite the results of these techniques an important remark has to be made: no model selection is performed. The value of parameters is defined a priori even if the set of variables changes.

B. Model Selection

The performance of SVM strictly depends on the right choice of the parameters: an adequate choice assures good results in terms of accuracy and generalization ability while a non-correct choice may have a harmful impact impairing the prediction. If the classification problem is simple (i.e. the instances are linearly separable) only the parameter that controls the generalization ability of the model has to be chosen. But if the complexity of the problem increases nonlinear classification is required so the number of parameters arises[1] and the identification of the parameters values becomes hard [14].

Parameters tuning is a non trivial process, it has no theoretical foundations thus different methods were proposed to fill this gap. They can be grouped into five areas: Exhaustive Research, Random Search, Grid Search, Optimization-based methods and GA-based methods.

The most known – but not feasible – approach is the Exhaustive Research which provides an accurate investigation of the parameters values. This method is extremely simple and effective but intractable for application purposes [9].

The Random Search is as simple and effective as Exhaustive search. Starting from the variables search space some points (i.e. solutions) are randomly picked and evaluated, the best one is chosen and the predictive model is then applied. This procedure must be repeated many times to ensure a good investigation of the parameters' values. The random choice shows two main drawbacks represented by the random process extraction and the number of repetitions. Moreover, unless specific criteria are employed to select potential solutions, the control on the process is very shrunken.

Another simple method to tune the parameters is the Grid Search. Parameters values are scanned in a pre-defined space: the research area is split in small blocks, the intersections of the grid are evaluated and the "global optimum" is located. The finer is the grid the greater will be the quality of the search. The main trouble is represented by the trade-off between efficiency and effectiveness. An application to a small space guarantees low consumption of time but hinder the research of the optimal solution. On the contrary, though a finer or a wider investigation is preferred the computational time increases a lot. Hence Grid Search is suitable for investigating very few parameters [9], [13].

Optimization based methods search for the best parameters' values solving an optimization problem. During the last decade many Optimization-based methods were suggested [6]-[7], [9], [14]. These techniques are fast and efficient providing good parameters' values that increase the predictive ability of classifiers. Furthermore, some of them take advantages by the employed classification algorithm, like the approach described by Chapelle et al. [9]. However these methods introduce restrictive assumptions like differentiability of kernel functions or functions' approximation (see [13] and [14]). Friedrichs and Igel [13] prove that Optimization methods based on the Gradient Descent suffer from three main problems related to kernel functions, score function and approximation induced by restrictive assumptions. The first is the differentiability of the kernel function and the second the differentiability of the score function respect to regularization and kernel parameters. Moreover the researchers assert that "Iterative gradient-based algorithms, which usually rely on smoothed approximations of a score function, do not ensure that the search direction points exactly to an optimum of the original, often discontinuous generalization performance measure". Fröhlich and Zell [14] analyzing the research of Chapelle and Vapnik [7] and Chapelle et al. [9] provide similar remarks of Friedrichs and Igel [13] about the use of radius-margin and span bounds.

As discussed before for feature selection impacts on SVM classification performance, similar remarks can be gleaned for model selection. Whether a linear classification problem is considered or a nonlinear model is required, the choice of the parameters improves or reduces the ability of the classifier to discriminate instances.

C. Feature and Model Selection

Many papers focused their attention on the role of feature and model selection [1], [2], [11], [15]-[20]. But these methods are, in general, "single target" approaches in which feature selection or model selection are performed separately. In other words they do not involve both procedures at the same time. Filters[2] feature reduction do not consider classifier's methods. In fact, they use general criteria to estimate the predictive ability, the relevance and the

[1] The number of parameters depends on the type of kernel.

[2] Filter, wrapper and embedded approaches are considered as in [25] and [11].

redundancy of variables. On the contrary, wrapper and embedded approaches use SVM but with fixed parameters. This hypothesis is strong [1]-[5] because parameters' values are optimal for a problem with fixed variables and instances. Changing just a small part of the problem may compromise the optimality of the solution (i.e. parameters' values and selected variables may be different). Single target model selection methods have an opposite assumption, since they use a fixed set of variables and then look for the best parameters' values.

Besides the studies that investigate feature and model selection separately a new heuristic based research area faced these problems. In this context Evolutionary Algorithms can settle at the same time the problem, of selecting relevant features and choosing the best parameters values [21]-[23].

III. GA-SVM Method

SVM shows good performance but to reach better results adequate values of the parameters have to be chosen [1], [3]-[5] and relevant variables have to be selected [2], [8], [11]-[12], [15]. The GA and SVM process answers the problem of performing together feature and model selection. In fact, SVM can be employed in the "GA cycle" using its results in the fitness function computation. The next paragraphs describe the GA process, the SVM methods and the embedded method (GA-SVM).

A. Genetic Algorithm

Genetic Algorithm is a heuristic search method based on principles of natural selection [26], [27]. The GA approach makes a population evolve using an "indicator" that leads the evolution. Three operators control the evolutionary process making the chromosomes reproduce and mutate. The selection operator extracts the best chromosomes from the population; the crossover operator manages the exchange of genes between chromosomes; the mutation operator modifies the genes applying small variations to their value. This procedure (Fig. 1) emulates the natural selection and forces the individuals (chromosomes) to converge toward an optimal solution.

The GA evolves assessing individuals according to a specific function called fitness function. During the evolutionary process the fitness function is computed and the best chromosomes are at first selected and then put in the matting pool for reproduction. The next generation is created combining the genes of the past one and some random changes are introduced to make the worst individuals grow correctly.

B. Support Vector Machine

The SVM approach is a learning machine, originally developed by Vapnik [28], [29], based on the Structural Risk Minimization principle. In this paragraph the classification problem is described in order to introduce the recent development of GA and SVM approach.

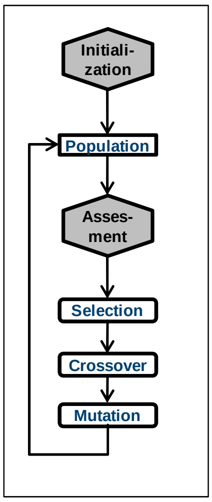

Fig. 1 Genetic Algorithm cycle

Given a dataset of m points $(x_i, y_i), i \in M = \{1,2,\dots,m\}$ in $\mathbb{R}^{n+1}$ where x_i is an n -dimensional vector and y_i is a scalar that represents the class of the i-th instance where $y_i \in \{-1,+1\}$. The SVM problem for linear classification can be formulated as follows

$$\begin{aligned}&\min_{\alpha} \frac{1}{2}\sum_{i=1}^{m}\sum_{i=1}^{m} y_i y_j \alpha_i \alpha_j x_i x_j - \sum_{i=1}^{m} \alpha_i \\ &s.t. \qquad \sum_{i=1}^{m} y_i \alpha_i = 0 \\ &\qquad 0 \le \alpha_i \le C \quad i = 1,\dots,m\end{aligned} \tag{1}$$

where α_i are the Lagrange multipliers.

The linear problem (1) can be extended to nonlinear case using kernel functions. A kernel function (2) maps data from an input space to a high dimensional feature space in which a linear separation can be performed [30].

$$\Phi : \mathbb{R}^n \to \mathbb{R}^k \tag{2}$$

Kernel functions do not physically map data, they just perform an implicit mapping – kernel trick – of the variables in the feature space through inner product. Thus changing from a linear classifier (1) to a non linear one (4) is done replacing the dot product with a kernel function. This operation does not impair the advantages of the standard SVM and allows maintaining its properties.

$$\begin{aligned}&\min_{\alpha} \frac{1}{2}\sum_{i=1}^{m}\sum_{i=1}^{m} y_i y_j \alpha_i \alpha_j \phi(x_i)\phi(x_j) - \sum_{i=1}^{m} \alpha_i \\ &s.t. \qquad \sum_{i=1}^{m} y_i \alpha_i = 0 \\ &\qquad 0 \le \alpha_i \le C \qquad i = 1,\dots,m\end{aligned} \tag{3}$$

where $\phi(x_i)$ is defined $\mathbb{R}^n \to \mathbb{R}^k,\ k \gg n$.

$$\begin{aligned}&\min_{\alpha}\frac{1}{2}\sum_{i=1}^{m}\sum_{i=1}^{m} y_i y_j \alpha_i \alpha_j K(x_i,x_j)-\sum_{i=1}^{m}\alpha_i\\ &s.t. \qquad \sum_{i=1}^{m} y_i\alpha_i = 0 \\ &\qquad 0\le \alpha_i \le C \qquad i=1,\dots,m\end{aligned} \tag{4}$$

Where the kernel matrix, also known as Gram matrix, is a positive semi-definite matrix which respects the Mercer's condition[3]. It is defined as

$$K=\{\langle\phi(x_i),\phi(x_j)\rangle\}_{i=1,\dots,m;\ j=1,\dots,m} \tag{5}$$

C. GA-SVM Details

The GA approach performs feature and model selection making a population evolve through a leading SVM performance indicator, usually the accuracy or the error of the classification process. A set of solutions (i.e. a population) grows according to that fitness function which summarizes chromosome's skill in a single value. Every generation the best chromosomes that survive the artificial selection are selected for reproduction[4]. Through their genes they will contribute to next generations' solutions until the maximum number of generations is reached or other stopping criteria end the search.

The GA-SVM process can be – ideally – divided in two phases: the population evolution performed by the GA and the individuals' evaluation performed by the SVM. Referring to Fig. 1 GA works like a carrier that brings the solutions to the SVM which – second phase – assesses them. The performance reached by the classifier defines the fitness value of each individual that settles the population.

The population involved in the evolutionary process can be described by its internal frame and by its size. The structure of the chromosome (i.e. a single solution), depicted in Fig. 2, is composed of two parts: the first part represents the variables, while the second part the parameters. Each gene represents a single feature (gray squares) and single parameters (light blue squares). The number of parameters (l) hinges on the chosen kernel functions. The population size is influenced by the initialization policy and the nature of the investigated problem. In this paper individuals' genes are randomly generated covering the entire search space: for the variables' part a subset of features is selected while for the parameters' part l values are extracted from pre-defined parameters' ranges. The nature of the problem plays a key role in dimension estimation. In fact, to guarantee a good search, the number of individuals must be adequate to the investigation space. In simple terms, more variables and more parameters entail a larger search space hence a larger population. Nevertheless further factors influence the population size like the amount of time at disposal and how fine the research is.

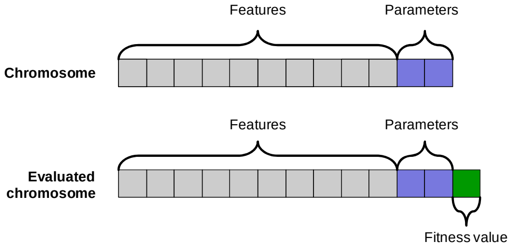

Fig. 2 Chromosome's structure

In the common GA-SVM approach the fitness value is represented by the accuracy or the error of the SVM. Even if different fitness functions based on SVM performance indicators were suggested the most used criterion remains the accuracy. To compute the fitness function two main techniques are employed: the k-folds Cross Validation and the Leave One Out Cross Validation. But, because of the required time [31], the first one is usually preferred.

To reproduce the natural evolution tournament selection, p-points crossover and mutation operators were applied. The tournament selection allows increasing the pressure on the population forcing it to converge. Crossover and mutation operators are employed with a good probability of reproduction and low mutation occurrences.

IV. Kernel Matrix Genetic Algorithm

The common GA evolves according to the accuracy of the trained SVM thus to complete the evolution many SVM must be trained. Even if this technique ensures the best results the whole process is hard in terms of complexity and its application computational expensive thus speed and employed resources are very important. In order to deal with the high computational cost of GA-SVM method a new solution is proposed, the Kernel Matrix Genetic Algorithm.

Since SVM method uses the kernel matrix (5) to define the optimization problem (4), it is interesting to directly investigate the goodness of the kernel. In this context kernel matrix-based criteria [32]-[34] are employed to evaluate the Gram matrix in order to estimate the performance of the classifier. Compared to GA-SVM approach they estimate SVM accuracy without solving an optimization problem thus the evolutionary time can be reduced. Cristianini et al. [32] were among the first researchers to propose a kernel matrix evaluation criterion for SVM. They suggest the Kernel Target Alignment (KTA) criterion that assesses the goodness of a kernel computing the alignment between the kernel matrix and the target matrix. Nguyen and Ho [33] widen this idea and suggest a less restrictive criterion the FSM. Moreover, Jia and Liao [34] advise to use a modification of FSM, the FCMC criterion, that catches the inner distance of a class and the inner distance between two classes. These criteria, if compared to GA-SVM – that requires a complete classification process –, are faster to be computed so a complete evolution can be done in a shorter time. This will

[3] See [28].

[4] A high fitness value facilitates – but doesn't assure – that a chromosome survives the (artificial) selection.

positively impact on the GA process, which requires a great number of assessments.

A. Kernel Matrix Criteria

Referring to KMGA problem three kernel based criteria are investigated. Some notations have to be introduced before explaining the criteria. Without loss of generality, consider a binary classification problem and a dataset as defined in the Support Vector Machine paragraph. The training set can be sorted arranging +1 class at first rows[5]. The number of elements that belong to +1 class is n_+ (from y_1 to y_{m+}) whereas those one that belong to –1 class are n_- (from y_{m+1} to y_m)[6]. Therefore the vector of classes becomes as follows

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_{m+} \\ y_{m+1} \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} +1 \\ \vdots \\ +1 \\ -1 \\ \vdots \\ -1 \end{bmatrix} \tag{6}$$

B. Kernel Target Alignment

Cristianini et al. [32] introduce the Kernel Target Alignment (KTA) in order to estimate the performance of a SVM. This criterion esteems the goodness of a kernel comparing the kernel matrix with the target matrix. The higher the resemblance between the kernel matrix and the target matrix is the better the classification performance of a SVM classifier will be. The similarity measure is computed through the Frobenius inner product defined as:

$$\langle K, T \rangle_F = \sum_{i=1}^{m} \sum_{j=1}^{m} k_{ij} t_{ij} \tag{7}$$

Where K is the kernel matrix and T is the target matrix defined as:

$$T = y \cdot y' \tag{8}$$

KTA is defined as the normalized Frobenius inner product:

$$A(K, y) = \frac{\langle K, T \rangle_F}{\sqrt{\langle K, K \rangle_F \langle T, T \rangle_F}} \tag{9}$$

and it range[7] is [0,1]. A high value indicates a good kernel matrix so high expected accuracy of the classifier, while a low value reveals low expected accuracy thus a bad kernel.

In order to compare KTA with the other criteria an estimation of the classifier's error is provided:

$$KTA_{err} = 1 - A(K, y) \tag{10}$$

C. Feature Space-based Kernel Matrix Evaluation Measure

Nguyen and Ho [33] suggest using a surrogate of KTA that overcomes its drawbacks. They agree on the simplicity and efficiency of KTA but advise about the conditions of its applicability. Nguyen and Ho argue that KTA is only a sufficient condition but not a necessary one therefore a kernel matrix can be good even if KTA assumes low values. To overcome this limitation and to provide a more general criterion they relax some hypotheses introducing a new indicator called Feature Space-based kernel matrix evaluation Measure (FSM). In addition, they build the FSM to make it invariant to linear operators in the feature space and to preserve the proprieties of efficiency and error bound.

FSM is based on the within class variance and the relative positions of class centers hence, in order to compute the FSM, three elements are required: the kernel function (ϕ) and the centers of positive and negative class instances (11).

$$\phi_+ = \frac{\sum_{i=1}^{n_+} \phi(x_i)}{n_+} \qquad \phi_- = \frac{\sum_{i=n_++1}^{n} \phi(x_i)}{n_-} \tag{11}$$

Nguyen and Ho define the FSM as the ratio of the total within class variance in the direction between the class centers to the distance between the class centers.

$$FSM(K, y) = \frac{var}{\|\phi_- - \phi_+\|} \tag{12}$$

Where K is the kernel matrix, y is the vector of classes and var is

$$var = \sqrt{\frac{\sum_{i=1}^{n_+} \langle \phi(x_i) - \phi_+, e \rangle^2}{n_+ - 1}} + \sqrt{\frac{\sum_{i=n_++1}^{n} \langle \phi(x_i) - \phi_-, e \rangle^2}{n_- - 1}} \tag{13}$$

Where e is the unit vector in the direction between class centers, defined as

$$e = \frac{\phi_- - \phi_+}{\|\phi_- - \phi_+\|} \tag{14}$$

The estimation of the SVM performance is given by the FSM_{err}:

$$FSM_{err} = \frac{FSM(K, y)^2}{1 + FSM(K, y)^2} \tag{15}$$

FSM_{err} assumes values between 0 and 1 where low values entail a low expected error rate.

[5] This procedure is required by FSM and FCMC not by KTA.

[6] The number of the elements of both classes is $n_+ + n_- = n$.

[7] Nguyen and Ho [33] referring to Crisitanini et al. [32] advise that KTA range is $-1 \leq A(K, y) \leq 1$.

D. Feature Distance based Combinatorial Kernel Matrix Evaluation Criterion

Jia and Liao [34], inspired by the research activities of Nguyen and Ho [33] and Kandola et al. [35], extend the remarks made by Cristianini et al. [32] about the KTA introducing a new measure for combinatorial kernel matrix: the Feature distance based Combinatorial kernel Matrix evaluation Criterion (FCMC).

To explain FCMC criterion some introductory elements have to be defined. The (linear) combination of kernel, defined in (16), is composed of h kernels (K) of weights (p).

$$K(p) = \sum_{h=1}^{k} p_h \, K_h = \{\langle \phi_p(x_i), \phi_p(x_j) \rangle\}_{i,j=1,\dots,m} \qquad (16)$$

Where $p_h \geq 0$.

The function ϕ_p projects the subset of the training algorithm $\{x_i\}_{y=+1}$ and $\{x_i\}_{y=-1}$ into ϕ_p^+ and ϕ_p^- respectively. Where ϕ_p^+ e ϕ_p^- represent the centers of the classes.

$$\begin{aligned} \phi_p^+ &= \frac{1}{m_+} \sum_{i=1}^{n_+} \phi_p(x_i) \\ \phi_p^- &= \frac{1}{m_-} \sum_{i=n_+ + 1}^{n} \phi_p(x_i) \end{aligned} \qquad (17)$$

The distance between the two classes is

$$d_{out} = \|\eta\|^2 \qquad (18)$$

Where $\eta = \phi_p^+ + \phi_p^-$.

The sum of the distances between features and their centers within a class (d_{in}^+ and d_{in}^-) are

$$\begin{aligned} d_{in}^+ &= \sum_{i=1}^{n_+} \langle \phi_p(x_i) - \phi_p^+, \eta \rangle^2 \\ d_{in}^- &= \sum_{i=n_+ + 1}^{n} \langle \phi_p(x_i) - \phi_p^-, \eta \rangle^2 \end{aligned} \qquad (19)$$

Now the FCMC measure can be defined

$$FCMC(X, K(p), T) = \frac{d_{in}^+ + d_{in}^-}{d_{out}} \qquad (20)$$

Where X is the data matrix, $K(p)$ is the combination of kernels and T is the target matrix as defined in (8).

Jia and Liao compare their measure with KTA and prove that for small values of FCMC the SVM reaches a low error rate. FCMC criterion assumes values in a positive range. Jia and Liao do not provide an indicator for the error, so in order to compare it to KTA_{err} and FSM_{err} the $FCMC_{err}$ has to be introduced. The formulation is the same of FSM_{err}:

$$FCMC_{err} = \frac{FCMC(X, K(p), T)^2}{1 + FCMC(X, K(p), T)^2} \qquad (21)$$

V. Experiments

The target of the experiments is to estimate the goodness of kernel matrix-based criteria in selecting features and kernel parameters' values. To do this the KMGA is applied to four kernels (linear, polynomial, RBF and sigmoid) on four binary datasets (Australian credit approval, Diabete Indian, Heart and Ionosphere).

$$\text{Linear}: \; K(x_i, x_j) = x_i \cdot x_j \qquad (22)$$

$$\text{RBF}: \; K(x_i, x_j) = \exp\left(-\frac{\|\mathrm{x_i} - \mathrm{x_j}\|^2}{2\sigma^2}\right) \qquad (23)$$

$$\text{Polynomial}: \; K(x_i, x_j) = \left(\sigma(x_i \cdot x_j)\right)^{\mathrm{d}} \qquad (24)$$

$$\text{Sigmoid}: \; K(x_i, x_j) = \left(\sigma(x_i \cdot x_j) + \theta\right) \qquad (25)$$

KMGA consists of three kernel matrix-based criteria: FCMC, FSM and KTA. These methods are compared against the GA-SVM using the classification error on training and test set and the number of retained features as performance indicators. Kernel matrix criteria provide an estimation of the classifier's error and can be employed as chromosomes' fitness value during the evolutionary process. They speed up the GA evolution but introduce a restriction. They do not consider the parameter that controls the complexity of the SVM (the so called *C* parameter). Thus, in order to evaluate the information provided by these criteria the value of the *C* parameter will be fixed at 1.

In the experiments' description paragraph the details of the processes are provided while the results of the KMGA are shown in the last three paragraphs. The terms "kernel performance" will be referred to SVM performance obtained using a specific kernel and criterion.

A. Description

Experiments were conducted using a simple GA method written in MATLAB code and libsvm [36] to perform SVM classification method. The analyzed datasets have been taken from the UCI machine learning Repository [37]. All the datasets were normalized, shuffled and split in training set (70%) and test set (30%), except the Ionosphere dataset for which the original partitions were used. Categorical variables were converted in binary ones. Only binary classification problems were considered and no cost matrices were employed. The starting population was randomly chosen: each individual was composed of a subset of the whole variables and parameters' values were uniformly drawn from a pre-defined range. The population's size depended on the dimension of the dataset and on the kernel used. The number of generations was chosen according to the size of the problem and to the parameters' search space. The crossover and mutation operator probability were set to 0.8 and to 0.04 respectively. The selection operation was carried out by tournament selection with 40% of chromosomes involved in the matches. The elitist strategy was applied, the percentage of elite chromosomes was 4%. The evolution of the GA is performed using the classification error for the GA-SVM and the estimation of the error computed, as described in (10), (15) and (21), for KMGA approaches. The stopping criterion of the GA was the number of generations.

The Australian dataset shows 5% of missing values, replaced by the mode and the mean for categorical and numerical attributes respectively. The Diabete Indian and Heart-Statlog datasets were used without employing the cost matrix.

The experiments were conducted ten times each and, in order to reduce the computational time, the GA-SVM was applied using 10-folds cross validation instead of Leave-One-Out Cross Validation.

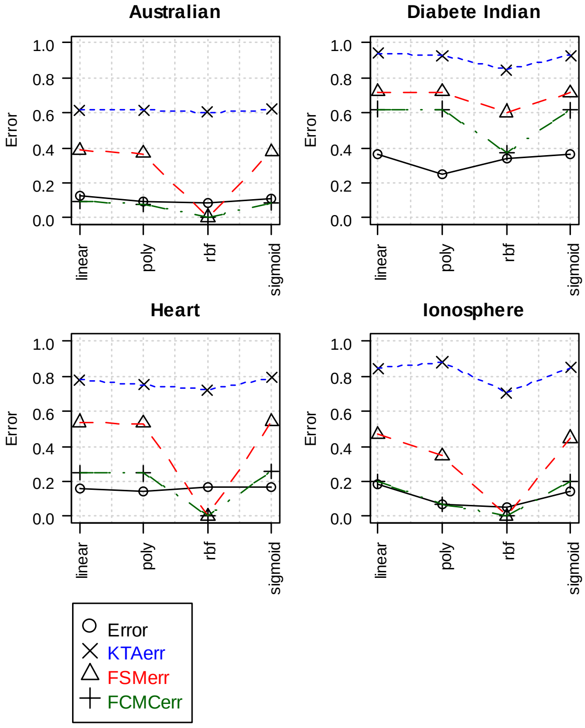

Fig. 3 Mean cross validation errors and estimated error on training set

B. *Experiments' Results on Training Set*

Fig. 3 shows the results of the experiments on the training set comparing the CV error computed by the GA-SVM to the estimated errors of the KMGA. The kernel matrix criteria estimate correctly the CV error two times: for Australian and Ionosphere datasets. In the second case all the criteria follow the CV error, except for the KTA that shows a higher value for the polynomial kernel.

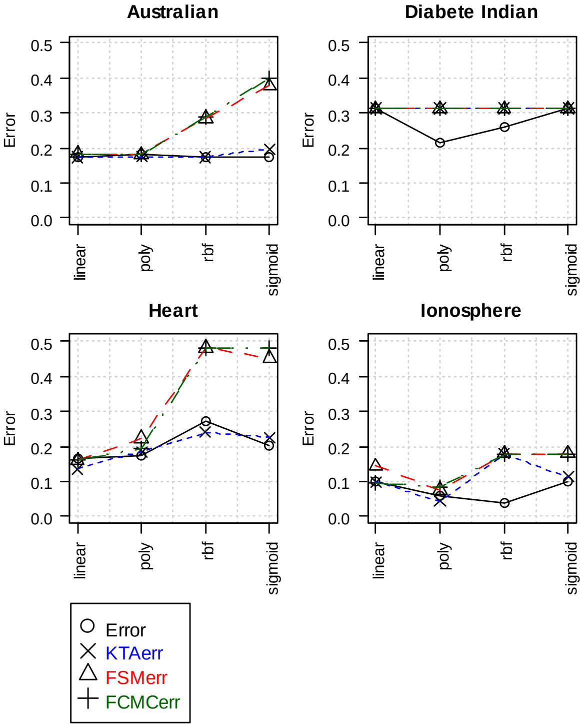

Fig. 4 Mean classification errors on test set

It is interesting to analyze two extremes in the kernel matrix-based criteria behavior. Considering the Australian dataset the lowest error is obtained by the RBF kernel even if the other three kernels show similar performances. KTA assumes steady values but shows the minimum for the RBF kernel; FCMC and FSM follow the CV error but sometimes they overestimate it. In the Diabete Indian dataset kernel matrix criteria reach the same results performing well for the RBF kernel estimation but fail identifying the lowest CV error.

Some general remarks can be made. The RBF kernel shows the best performance overall, the CV error and the kernel matrix criteria tend to estimate its performance more than other kernels. KTA shows less variations, it seems the more conservative criterion. FSM and FCMC values, if compared with KTA, are closer to CV error but tend to overestimate the SVM performance. In fact, three times out of four they show lower values of error. FCMC criterion follows the GA-SVM error better then KTA and FSM.

C. *Experiments Results on Test Set*

This paragraph provides the results on test set using the retained variables and the parameters' values suggested by GA-SVM and KMGA methods. The term "solution" will be referred to the combination of selected features and parameters' values.

With the purpose of investigating the ability of KMGA to discover good solutions classifiers' results on test set are reported in Fig. 4. The errors computed by the GA-SVM (circle points) are considered as benchmarks for KMGA performed through kernel matrix criteria. As it can be seen from Fig. 4 KMGA approaches are able to perform as well as GA-SVM one. A deeper analysis clarifies the results:

- KTA provides the best estimation performance. Moreover, it gives the most reliable results: the error of the classifier computed with the solution provided by KTA evolution is at least small and at best better than the GA-SVM method.
- FSM and FCMC provide good predictions of classifier's performance, especially for linear and polynomial kernels. Furthermore, they show the same behavior.
- Kernel matrix criteria were not able to find a good solution for the Diabete Indian dataset. This could be due to the number of selected features, 64% for GA-SVM and 35% for KMGA.

TABLE I
SUMMARY OF THE EXPERIMENTS ON AUSTRALIAN AND DIABETE INDIAN DATASETS. BOLD VALUES REPRESENT THE BEST SOLUTIONS FOR EACH KERNEL

			Error		Attributes				
Data	Ker	Crite-rion	CV	Test	Tot #	% retained	Mean±sd	Min	Max
Australian	Linear	Error	0,128	**0,174**	42	56%	24±2	19	27
		FCMC	0,093	0,184	42	69%	29±0	29	29
		FSM	0,387	0,184	42	67%	28±0	28	28
		KTA	0,615	**0,174**	42	14%	6±0	6	6
	Poly	Error	0,092	0,181	42	40%	17±3	14	25
		FCMC	0,078	0,181	42	73%	31±2	29	37
		FSM	0,368	0,181	42	76%	32±2	30	36
		KTA	0,615	**0,176**	42	10%	4±0	4	5
	Rbf	Error	0,090	**0,173**	42	45%	19±3	15	23
		FCMC	0,000	0,287	42	70%	30±3	26	36
		FSM	0,004	0,285	42	70%	29±3	25	34
		KTA	0,606	0,174	42	17%	7±1	6	9
	Sigmoid	Error	0,112	**0,173**	42	54%	23±3	19	26
		FCMC	0,086	0,401	42	47%	20±2	17	25
		FSM	0,380	0,378	42	51%	22±1	20	24
		KTA	0,622	0,198	42	24%	10±2	6	13
Diabete Indian	Linear	Error	0,364	**0,313**	8	48%	4±2	2	7
		FCMC	0,622	**0,313**	8	38%	3±0	3	3
		FSM	0,719	**0,313**	8	38%	3±0	3	3
		KTA	0,940	**0,313**	8	13%	1±0	1	1
	Poly	Error	0,247	**0,214**	8	66%	5±0	5	6
		FCMC	0,622	0,313	8	38%	3±0	3	3
		FSM	0,719	0,313	8	38%	3±0	3	3
		KTA	0,926	0,313	8	14%	1±0	1	2
	Rbf	Error	0,339	**0,260**	8	91%	7±0	7	8
		FCMC	0,374	0,313	8	41%	3±0	3	4
		FSM	0,602	0,313	8	43%	3±1	3	4
		KTA	0,846	0,313	8	91%	7±1	6	8
	Sigmoid	Error	0,364	**0,313**	8	50%	4±2	1	6
		FCMC	0,614	**0,313**	8	38%	3±0	3	3
		FSM	0,716	**0,313**	8	38%	3±0	3	3
		KTA	0,926	**0,313**	8	14%	1±0	1	2

Some additional observations can be made. The Ionosphere dataset shows a gap between kernel matrix criteria estimations and GA-SVM test error for RBF kernel. FSM and FCMC have some problems in estimating RBF and sigmoid kernels. Comparing to GA-SVM method, KTA guarantees similar results with very few attributes. Moreover, even if KTA is an approximated indicator, it performs better in the Heart dataset for linear and RBF kernels and in the Ionosphere dataset for the polynomial kernel.

TABLE II
SUMMARY OF THE EXPERIMENTS OF HEART AND IONOSPHERE DATASETS. BOLD VALUES REPRESENT THE BEST SOLUTIONS FOR EACH KERNEL

			Error		Attributes				
Data	Ker	Crite-rion	CV	Test	Tot #	% retained	Mean±sd	Min	Max
Heart (Statlog)	Linear	Error	0,161	0,165	25	42%	11±2	7	15
		FCMC	0,250	0,161	25	56%	14±0	14	14
		FSM	0,535	0,161	25	56%	14±0	14	14
		KTA	0,779	**0,136**	25	40%	10±0	10	10
	Poly	Error	0,141	**0,174**	25	48%	12±3	7	15
		FCMC	0,245	0,195	25	66%	16±3	13	24
		FSM	0,530	0,224	25	82%	21±4	15	24
		KTA	0,752	0,185	25	68%	17±0	17	17
	RBF	Error	0,167	0,272	25	44%	11±2	7	13
		FCMC	0,000	0,482	25	66%	16±3	12	19
		FSM	0,000	0,482	25	60%	15±4	6	20
		KTA	0,723	**0,240**	25	32%	8±1	7	9
	Sigmoid	Error	0,163	**0,201**	25	50%	12±2	9	17
		FCMC	0,256	0,482	25	58%	14±2	12	18
		FSM	0,539	0,452	25	56%	14±2	10	17
		KTA	0,793	0,225	25	31%	8±2	6	11
Ionosphere	Linear	Error	0,180	0,098	34	54%	18±2	13	20
		FCMC	0,198	**0,097**	34	42%	14±0	14	15
		FSM	0,468	0,143	34	41%	14±0	14	14
		KTA	0,844	0,099	34	15%	5±0	5	5
	Poly	Error	0,069	0,059	34	46%	16±2	12	18
		FCMC	0,068	0,084	34	47%	16±0	15	16
		FSM	0,345	0,076	34	43%	15±1	13	16
		KTA	0,880	**0,043**	34	56%	19±1	18	20
	RBF	Error	0,055	**0,038**	34	51%	17±3	14	21
		FCMC	0,000	0,179	34	79%	27±5	18	33
		FSM	0,000	0,179	34	78%	27±7	13	33
		KTA	0,704	0,179	34	28%	10±4	4	15
	Sigmoid	Error	0,145	**0,101**	34	52%	18±3	13	21
		FCMC	0,197	0,179	34	42%	14±1	14	15
		FSM	0,446	0,179	34	41%	14±0	13	15
		KTA	0,848	0,113	34	17%	6±1	5	7

D. Experiments' Results – Number of Retained Features

TABLE and TABLE summarize the performance obtained by the GA-SVM and the KMGA approaches. In this paragraph the focus is on the number of retained features.

The analysis of Australian dataset shows that FCMC and FSM have the highest number of retained feature, KTA the lowest and GA-SVM is placed "in the middle". This trend is confirmed also for Diabete Indian and Heart while the Ionosphere dataset gives unclear results. Among the kernel matrix criteria the evolution performed by the KTA greatly reduces the number of features of the final solution (low values of percentage). Only in two cases, for Heart and

TABLE III
HEART DATASET COMPUTATIONAL TIME (*S*). VALUES ARE EXPRESSED AS MEAN±SD AND PERCENTAGE OF REQUIRED TIME OVER THE GA-SVM ERROR TIME

	linear		**poly**		**rbf**		**sigmoid**	
Error	610±85	100%	1786±460	100%	1618±113	100%	3305±197	100%
FCMC	202±3	33%	177±3	10%	615±17	38%	642±93	19%
FSM	200±5	33%	182±12	10%	611±45	38%	656±21	20%
KTA	190±1	31%	186±5	10%	496±18	31%	560±15	17%

Ionosphere datasets, KTA selects more attributes than other approaches both with polynomial kernel. Furthermore, KTA shows stable results (low standard deviation). On the contrary, FSM and FCMC show an average percentage of selected attributes higher than KTA and GA-SVM and less stable results.

E. Experiments' Results – Computational Time

In order to compare, in term of requested time, the GA-SVM and the KMGA the results on Heart dataset are provided. To make the comparison reliable only the maximum number of generations was used as stopping rule.

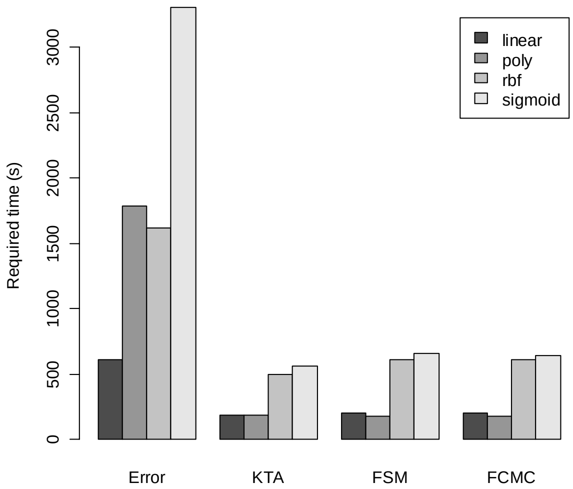

Fig. 5 Heart dataset computational time

Fig. 5 summarizes the time expenditure to perform a full evolution. The KMGA takes at worst 38% time and at best 10% of the GA-SVM, proving the efficiency of the joined kernel matrix and GA approach. Three important remarks can be deduced from Fig. 5 and Table III:

- the computational effort required by KMGA methods is similar;
- the evolution driven by KTA is usually faster than the ones led out by FSM and FCMC because of the number of selected features;
- KMGA shows the best performance with the polynomial kernel. Kernel matrix-based criteria perform the estimation ten times faster than through GA-SVM.

VI. CONCLUSION

This paper presents the KMGA, a joined GA and Kernel Matrix criteria approach to perform simultaneously feature and model selection to improve the classification performance of SVM.

To select variables and to tune SVM and kernel parameters literature suggests using the GA-SVM approach. Even if it proves its effectiveness, it is computationally expensive because training a classifier means solving an optimization problem. Thus, in order to reduce the efforts a kernel matrix approach can be considered. Furthermore, since the SVM depends on the kernel to map data into a high dimensional space, the kernel matrix becomes a "natural" proxy of SVM classification ability. In this context KMGA approach provides an efficient method that overcomes the GA-SVM drawback of the required computational time.

Experiments confirm that KMGA improves the SVM classification performance. In fact, the error on the test set obtained by the best solution estimated with kernel matrix criteria is better – or close to – the GA-SVM error. Among the kernel matrix criteria KTA shows the best performance and, even if its estimations on the training set are not promising, on the test set its errors are at least equal to GA-SVM and for three times better than that. FSM and FCMC show a different behavior: despite some difficulties with RBF and sigmoid kernels they perform well on linear and polynomial ones reaching similar results on the test set. In addition, analyzing kernel matrix criteria effects on selected variables it is not possible to define a unique impact on the evolutionary process because the number of retained variables differs both in datasets and kernel's types.

KMGA approach is able to compete with *state-of-the-art* methods, like GA-SVM, providing an effective tool to identify the best subset of features and the optimal kernel's parameters and to reduce the whole computational time.

REFERENCES

[1] H. Fröhlich, O. Chapelle and B. Schölkopf, "Feature selection for support vector machines by means of genetic algorithms," in *Proceedings: 15th IEEE International Conference on Tools with Artificial Intelligence,* 2003, pp. 142-148.

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

[3] C. Huang and C. Wang, "A GA-based feature selection and parameters optimization for support vector machines," *Expert Systems with Applications,* vol. 31, pp. 231-240, /8. 2006.

[4] K. Y. Chan, H. L. Zhu, C. C. Lau and S. H. Ling, "Gene signature selection for cancer prediction using an integrated approach of genetic algorithm and support vector machine," in *2008 IEEE Congress on Evolutionary Computation, CEC 2008,* 2008, pp. 217-224.

[5] I. Mejía-Guevara and Á. Kuri-Morales, "Genetic support vector classification and feature selection," in *7th Mexican International Conference on Artificial Intelligence, MICAI 2008,* 2008, pp. 75-81.

[6] Y. Bengio, "Gradient-based optimization of hyperparameters," *Neural computation,* vol. 12, pp. 1889-1900, 2000.

[7] O. Chapelle and V. Vapnik, "Model Selection for Support Vector Machines," 2000.

[8] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil and V. Vapnik, "Feature selection for SVMs," in *Advances in Neural Information Processing Systems 13,* 2000, pp. 668-674.

[9] O. Chapelle, V. Vapnik, O. Bousquet and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Machine Learning,* vol. 46, pp. 131-159, 2002.

[10] I. Guyon, J. Weston, S. Barnhill and V. Vapnik, "Gene Selection for Cancer Classification using Support Vector Machines," *Machine Learning,* vol. 46, pp. 389-422, 2002.

[11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research,* vol. 3, pp. 1157-1182, 2003.

[12] J. Reunanen, "Overfitting in making comparisons between variable selection methods," *Journal of Machine Learning Research.,* vol. 3, pp. 1371-1382, 2003.

[13] F. Friedrichs and C. Igel, "Evolutionary tuning of multiple SVM parameters," *Neurocomputing,* vol. 64, pp. 107-117, 2005.

[14] H. Frohlich and A. Zell, "Efficient parameter selection for support vector machines in classification and regression via model-based global optimization," in *International Joint Conference on Neural Networks, IJCNN 2005, July 31, 2005 - August 4,* 2005, pp. 1431-1436.

[15] K. Kira and L. A. Rendell, "Feature selection problem: Traditional methods and a new algorithm," in *Proceedings Tenth National Conference on Artificial Intelligence - AAAI-92,* 1992, pp. 129-134.

[16] Y. Chen, Y. Li, X. Cheng and L. Guo, "Survey and Taxonomy of Feature Selection Algorithms in Intrusion Detection System," *In: H. Lipmaa, M. Yung and D. Lin, Editors, Inscrypt 2006 4318, LNCS 2006, pp. 153–167.*

[17] H. Cheng, H. Chen, G. Jiang and K. Yoshihira, "Nonlinear feature selection by relevance feature vector machine," in proceedings MLDM '07: Proceedings of the 5th international conference on Machine Learning and Data Mining in Pattern Recognition, pp.144-159, 2007.

[18] C. Park, J. -. Koo, P. T. Kim and J. W. Lee, "Stepwise feature selection using generalized logistic loss," *Computational Statistics and Data Analysis,* vol. 52, pp. 3709-3718, 2008.

[19] K. Shen, C. Ong, X. Li and E. P. V. Wilder-Smith, "Novel multi-class feature selection methods using sensitivity analysis of posterior probabilities," in *2008 Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, SMC 2008,* 2008, pp. 1116-1121.

[20] P. Maji, "F-Information measures for efficient selection of discriminative genes from microarray data," *IEEE Transactions on Biomedical Engineering.,* vol. 56, pp. 1063-1069, 2009.

[21] H. Huang and F. Chang, "ESVM: Evolutionary support vector machine for automatic feature selection and classification of microarray data," *BioSystems,* vol. 90, pp. 516-528, 2007.

[22] P. L. Braga, A. L. I. Oliveira and S. R. L. Meira, "A GA-based feature selection and parameters optimization for support vector regression applied to software effort estimation," in *23rd Annual ACM Symposium on Applied Computing, SAC'08,* 2008, pp. 1788-1792.

[23] E. Avci, "Selecting of the optimal feature subset and kernel parameters in digital modulation classification by using hybrid genetic algorithm-support vector machines: HGASVM," *Expert Systems with Applications,* vol. 36, pp. 1391-1402, 2009.

[24] K. C. Tan, E. J. Teoh, Q. Yu and K. C. Goh, "A hybrid evolutionary algorithm for attribute selection in data mining," *Expert Systems with Applications,* vol. 36, pp. 8616-8630, 2009.

[25] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence,* vol. 97, pp. 273-324, 1997.

[26] J. H. Holland, *Adaptation in Natural and Artificial Systems.* Ann Arbor, MI, USA: University of Michigan Press, 1975.

[27] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning.* Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc, 1989.

[28] V. Vapnik, *The Nature of Statistical Learning Theory.* Springer-Verlag New York, Inc, 1995.

[29] V. N. Vapnik, *Statistical Learning Theory.* Wiley, New York, 1998.

[30] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods.* Cambridge University Press, 2000.

[31] K. Duan, S. S. Keerthi and A. N. Poo, "Evaluation of simple performance measures for tuning SVM hyperparameters," *Neurocomputing,* vol. 51, pp. 41-59, 2003.

[32] N. Cristianini, J. Kandola, A. Elisseeff and J. Shawe-Taylor, "On kernel-target alignment," in *Advances in Neural Information Processing Systems 14,* 2002, pp. 367-373.

[33] C. H. Nguyen and T. B. Ho, "An efficient kernel matrix evaluation measure," *Pattern Recognition,* vol. 41, pp. 3366-3372, 2008.

[34] L. Jia and S. Liao, "Combinatorial kernel matrix model selection using feature distances," in *International Conference on Intelligent Computation Technology and Automation, ICICTA 2008,* 2008, pp. 40-43.

[35] J. Kandola, J. Shawe-Taylor and N. Cristianini, "Optimizing kernel alignment over combinations of kernel," Department of Computer Science,Royal Holloway, University of London, UK, 2002.

[36] C. Chang and C. Lin, *LIBSVM: A Library for Support Vector Machines,* 2001.

[37] A. Asuncion and D. J. Newman, "UCI Machine Learning Repository," 2007. http://archive.ics.uci.edu/ml/citation_policy.html