Neurocomputing 117 (2013) 98–106

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A PSO and pattern search based memetic algorithm for SVMs parameters optimization

Yukun Bao *, Zhongyi Hu, Tao Xiong

Department of Management Science and Information Systems, School of Management, Huazhong University of Science and Technology, Wuhan 430074, PR China

ARTICLE INFO

Article history:
Received 3 October 2012
Received in revised form
15 November 2012
Accepted 16 January 2013
Communicated by W.S. Hong
Available online 27 February 2013

Keywords:
Parameters optimization
Support vector machines
Memetic algorithms
Particle swarm optimization
Pattern search

ABSTRACT

Addressing the issue of SVMs parameters optimization, this study proposes an efficient memetic algorithm based on particle swarm optimization algorithm (PSO) and pattern search (PS). In the proposed memetic algorithm, PSO is responsible for exploration of the search space and the detection of the potential regions with optimum solutions, while pattern search (PS) is used to produce an effective exploitation on the potential regions obtained by PSO. Moreover, a novel probabilistic selection strategy is proposed to select the appropriate individuals among the current population to undergo local refinement, keeping a well balance between exploration and exploitation. Experimental results confirm that the local refinement with PS and our proposed selection strategy are effective, and finally demonstrate the effectiveness and robustness of the proposed PSO-PS based MA for SVMs parameters optimization.

1. Introduction

Support vector machines (SVMs), first presented by Vapnik [1] based on statistical learning theory (SLT) and structural risk minimization principle (SRM), solve the classification problem by maximizing the margin between the separating hyper-plane and the data. SVMs implement the structural risk minimization principle (SRM), which seeks to minimize an upper bound of the generalization error by using penalty parameter C as trade-off between training error and model complexity. The use of kernel tricks enables the SVMs to have the ability of dealing with nonlinear features in high dimensional feature space. Due to the excellent generalization performance, SVMs have been widely used in various areas, such as pattern recognition, text categorization, fault diagnosis and so on. However, the generalization ability of SVMs highly depends on the adequate setting of parameters, such as penalty coefficient C and kernel parameters. Therefore, the selection of the optimal parameters is of critical importance to obtain a good performance in handling learning task with SVMs.

In this study, we mainly concentrate on the parameters optimization of SVMs, which has gained great attentions in the past several years. The most popular and universal method is grid search, which conducts an exhaustive search on the parameters space with the validation error (such as fivefold cross-validation error) minimized. Obviously, although it can be easily parallelized and seems safe [2], its computational cost scales exponentially with the number of parameters and the number of the sampling points for each parameter [3]. Besides, the performance of grid search is sensitive to the setting of the grid range and coarseness for each parameter, which are not easy to set without prior knowledge. Instead of minimizing the validation error, another streams of studies focused on minimizing the approximated bounds on generalization performance by numerical optimization methods [4–12]. The numerical optimization methods are generally more efficient than grid search for their fast convergence rate. However, this kind of methods can only be suitable for the cases that error bounds are differentiable and continuous with respect to the parameters in SVMs. It is also worth noting that the numerical optimization methods, such as gradient descent, may get stuck in local optima and highly depend on the starting points. Furthermore, experimental evidence showed that several established bounds methods could not compete with traditional fivefold cross-validation method [6], which indicates inevitably gap between the approximation bounds and real error [13]. Recently, evolutionary algorithms such as genetic algorithm (GA), particle swarm optimization (PSO), ant colony optimization (ACO) and simulated annealing algorithm (SA) have been employed to optimize the SVMs parameters for their better global search abilities against numerical optimization methods [14–19]. However, GAs and other evolutionary algorithms (EAs) are not

* Corresponding author. Tel.: +86 27 87558579; fax: + 86 27 87556437.
E-mail addresses: yukunbao@hust.edu.cn, y.bao@ieee.org, yukunbao@mail.hust.edu.cn (Y. Bao).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.01.027

guaranteed to find the global optimum solution to a problem, though they are generally good at finding “acceptable good” or near-optimal solutions to problems. Another drawback of EAs is that they are not well suited to perform finely tuned search, but on the other hand, they are good at exploring the solution space since they search from a set of designs and not from a single design.

Memetic algorithms (MAs), first coined by Moscato [20,21], have been regarded as an promising framework that combines the evolutionary algorithms (EAs) with problem-specific local searcher (LS), where the latter is often referred to as a meme defined as a unit of cultural evolution that is capable of local refinements. From an optimization point of view, MAs are hybrid EAs that combine global and local search using an EA to perform exploration while the local search method performs exploitation. This has the ability to exploit the complementary advantages of EAs (generality, robustness, global search efficiency), and problem-specific local search (exploiting application-specific problem structure, rapid convergence toward local minima) [22]. Up to date, MAs have been recognized as a powerful algorithmic paradigm for evolutionary computing in a wide variety of areas [23–26]. In particular, the relative advantage of MAs over EAs is quite consistent on complex search spaces.

Since the parameters space of SVMs is often considered complex, it is of interest to justify the use of MAs for SVMs parameters optimization. There have been, if any, few works related to MAs reported in the literature of SVMs parameters optimization. In this study, by combining particle swarm optimization algorithm (PSO) and pattern search (PS), an efficient PSO-PS based memetic algorithm (MA) is proposed to optimize the parameters of SVMs. In the proposed PSO-PS based MA, PSO is responsible for exploration of the search space and the detection of the potential regions with optimum solution, while a direct search algorithm, pattern search (PS) in this case, is used to produce an effective exploitation on the potential regions obtained by PSO. Furthermore, the problem of selecting promising individuals to experience local refinement is also addressed and thus a novel probabilistic selection strategy is proposed to keep a balance between exploration and exploitation. The performance of proposed PSO-PS based MA for parameters optimization in SVMs is justified on several benchmarks against selected established counterparts. Experimental results and comparisons demonstrate the effectiveness of the proposed PSO-PS based MA for parameters optimization of SVMs.

The rest of the study is organized as follows. Section 2 presents a brief review on SVMs, Section 3 elaborates on the PSO-PS based MA proposed in this study, the results with discussions are reported in Section 4, and finally, we conclude this study in Section 5.

2. Support vector machines

Consider a binary classification problem involving a set of training dataset $\{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2)...(\mathbf{x}_n,y_n)\} \subset R^n \times R$, where $\mathbf{x}_i$ is input space, $y_i \in \{-1,1\}$ is the labels of the input space $\mathbf{x}_i$ and n denotes the number of the data items in the training set. Based on the structured risk minimization (SRM) principle [1], SVMs aim to generate an optimal hyper-plane to separate the two classes by minimizing the regularized training error

$$\begin{aligned} &\min \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{n}\xi_i \\ &\quad s.t. \quad y_i(\langle \mathbf{w},\mathbf{x}_i \rangle + b) \geq 1-\xi_i, \quad i=1,2.....n \\ &\quad \xi_i \geq 0, \quad i=1,2.....n \end{aligned} \tag{1}$$

where $\langle , \rangle$ denotes the inner product; w is the weight vector, which controls the smoothness of the model; b is a parameter of bias; ξ_i is a non-negative slack variable which defines the permitted misclassification error. In the regularized training error given by Eq. (1), the first term, $1/2||\boldsymbol{w}||^2$, is the regularization term to be used as a measure of flatness or complexity of the function. The second term $\sum_{i=1}^{n}\xi_i$ is the empirical risk. Hence, C is referred to as the penalty coefficient and it specifies the trade-off between the empirical risk and the regularization term.

According to Wolfe's Dual theorem and the saddle-point condition, the dual optimization problem of the above primal one is obtained as the following quadratic programming form

$$\begin{aligned} &\max \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j \langle \mathbf{x}_i,\mathbf{x}_j \rangle \\ &\quad s.t. \quad \sum_{i=1}^{n}\alpha_i y_i = 0, \quad 0 \leq \alpha_i \leq C, \quad i=1,2.....n \end{aligned} \tag{2}$$

where $(\alpha_i)_{i \in n}$ are nonnegative Lagrangian multipliers that can be obtained by solving the convex quadratic programming problem stated above.

Finally, by solving Eq. (2) and using the trick of kernel function, the decision function can be defined as the following explicit form

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{n} y_i\alpha_i K(\mathbf{x}_i,\mathbf{x}) + b\right) \tag{3}$$

Here, $K(\mathbf{x}_i,\mathbf{x})$ is defined as kernel function. The value of the kernel function is equivalent to the inner product of two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ in the high-dimensional feature space $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$, that is, $K(\mathbf{x}_i,\mathbf{x}_j) = \langle \phi(\mathbf{x}_i),\phi(\mathbf{x}_j) \rangle$. The elegance of using the kernel function is that one can deal with feature spaces of arbitrary dimensionality without having to compute the map $\phi(\mathbf{x})$ explicitly. Any function that satisfies Mercer's condition [27] can be used as the kernel function. The typical examples of kernel function are as follows:

Linear kernel: $K(\mathbf{x}_i,\mathbf{x}_j) = \langle \mathbf{x}_i,\mathbf{x}_j \rangle$
Polynomial kernel: $K(\mathbf{x}_i,\mathbf{x}_j) = (\gamma\langle \mathbf{x}_i,\mathbf{x}_j \rangle + r)^d$, $\gamma > 0$.
Radial basis function (RBF) kernel: $K(\mathbf{x}_i,\mathbf{x}_j) = \exp(-\gamma||\mathbf{x}_i - \mathbf{x}_j||^2)$, $\gamma > 0$.
Sigmoid kernel: $K(\mathbf{x}_i,\mathbf{x}_j) = \tanh(\gamma\langle \mathbf{x}_i,\mathbf{x}_j \rangle + r)$.

where,γ, r and d are kernel parameters. The kernel parameter should be carefully chosen as it implicitly defines the structure of the high dimensional feature space $\phi(\mathbf{x})$ and thus controls the complexity of the final solution [28]. Generally, among these kernel functions, RBF kernel is strongly recommended and widely used for its performance and complexity [2] and thus SVMs with RBF kernel function is the one studied in this study.

Overall, SVMs are a powerful classifier with strong theoretical foundations and excellent generalization performance. Note that before implementing the SVMs with RBF kernel, there are two parameters (penalty parameter C and RBF kernel parameterγ) have to set. Previous studies show that these two parameters play an important role in the success of SVMs. In this study, addressing the selection of these two parameters, a PSO-PS based memetic algorithm (MA) is proposed and justified within the context of SVMs with RBF kernel function.

3. Proposed memetic algorithms

Memetic algorithms (MAs), first coined by Moscato [20,21], have come to light as an union of population-based stochastic global search and local based cultural evolution, which are inspired