Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition 40 (2007) 2543–2549
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/pr

A simple decomposition algorithm for support vector machines with polynomial-time convergence

Hong Qiao[a], Yan-Guo Wang[a], Bo Zhang[b,c,*]

[a]Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China
[b]Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing 100080, China
[c]Department of Mathematical Sciences, Coventry University, Coventry CV1 5FB, UK

Received 22 December 2005; received in revised form 21 November 2006; accepted 27 December 2006

Abstract

Support vector machines (SVMs) are a new and important tool in data classification. Recently much attention has been devoted to large scale data classifications where decomposition methods for SVMs play an important role.

So far, several decomposition algorithms for SVMs have been proposed and applied in practice. The algorithms proposed recently and based on rate certifying pair/set provide very attractive features compared with many other decomposition algorithms. They converge not only with finite termination but also in polynomial time. However, it is difficult to reach a good balance between low computational cost and fast convergence.

In this paper, we propose a new simple decomposition algorithm based on a new philosophy on working set selection. It has been proven that the working set selected by the new algorithm is a rate certifying set. Further, compared with the existing algorithms based on rate certifying pair/set, our algorithm provides a very good feature in combination of lower computational complexity and faster convergence.

Keywords: Support vector machines; Decomposition methods; Convergence; Statistical learning theory; Pattern recognition

1. Introduction

Support vector machines (SVMs) are a new classification method (see, e.g. Refs. [1–3]), which is widely used in data mining and machine learning. Suppose there are m training samples $\{(x_1, y_1), \ldots, (x_m, y_m)\}$, where x_i is an n-dimensional vector and $y_i \in \{-1, 1\}$. SVMs find a hyperplane $(w\cdot x)+b=0$ in a mapped sample space, which classifies the samples into proper classes, that is, $\mathrm{sgn}((w \cdot x_i) + b) = y_i$, where w is an m-dimensional vector and the minimum distance between the two classes gets its largest value. The above problem can be represented with its dual representation as the convex quadratic optimization problem:

$$\begin{cases} \min\limits_{\alpha} & f(\alpha) = \frac{1}{2}\alpha^\top Q\alpha - \mathrm{e}^\top\alpha \\ \text{s.t.} & y^\top\alpha = 0, \\ & 0 \leqslant \alpha_i \leqslant C, \quad i = 1, \ldots, m, \end{cases} \tag{1.1}$$

* Corresponding author. Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing 100080, China. Tel.: +86 10 6265 1358.

E-mail addresses: hong.qiao@mail.ia.ac.cn (H. Qiao), b.zhang@amt.ac.cn (Y. Wang), b.zhang@coventry.ac.uk (B. Zhang).

where C is a positive number, α is an m-dimensional vector, e is an m-dimensional vector with elements 1, and Q is a symmetric and positive semi-definite matrix with elements $q_{ij} = y_i y_j \cdot K(x_i, x_j)$. Here $K(x_i, x_j)$ denotes the kernel function used in SVMs classifiers.

In practical application, the size of the training samples is very huge, and solving the problem (1.1) on a large data set is challenging. So far, the main technique of solving the problem (1.1) is to decompose the original optimization problem into a series of sub-optimization problems, each of which optimizes the objective function $f(\alpha)$ with only a small number of elements in α varying, so that the size of the sub-optimization problems is fit for the computing memory. This is called as the decomposition method which was proposed in Refs. [5,7,8,10–12]. The basic procedure of this method is as follows:

Algorithm 1.1 (*Decomposition method*).

(1) Let α^0 be the initial solution. Set $k = 0$.

0031-3203/$30.00
doi:10.1016/j.patcog.2006.12.024

(2) If α^k is an optimal solution of Eq. (1.1), then stop. Otherwise, find a working set $B \subset \{1, \ldots, m\}$ with the size $|B| \leqslant q$, and set $N = \{1, \ldots, m\} \backslash B$.
(3) Solve the following subproblem for the variable α_B:

$$\begin{cases} \min & \frac{1}{2}\alpha_B^\top Q_{BB}\alpha_B - (e_B - Q_{BN}\alpha_N^k)^\top \alpha_B \\ \text{s.t.} & y_B^\top \alpha_B = -y_N^\top \alpha_N^k, \\ & 0 \leqslant (\alpha_B)_i \leqslant C, \quad i = 1, \ldots, q, \end{cases} \tag{1.2}$$

where α_B denotes the sub-vector of α projected onto the working set B, α_N^k denotes the sub-vector of α^k projected onto the non-working set N, y_B and y_N are defined similarly, and e_B is a $|B|$-dimensional vector with elements 1, and $\begin{bmatrix} Q_{BB} & Q_{BN} \\ Q_{NB} & Q_{NN} \end{bmatrix}$ is a permutation of the matrix Q.
(4) Set α_B^{k+1} to be the optimal solution of Eq. (1.2) and let $\alpha_N^{k+1} = \alpha_N^k$. Set $k \leftarrow k + 1$ and go to Step (2).

The key issue of the decomposition method is how to select the working set in each iteration. It is not easy to select the working set B to guarantee a fast convergence of the method. Many different selection rules have been proposed in decomposition methods for SVMs (see, e.g. Refs. [4–12]). For example, the SVM^{light} algorithm [5], which is a very important and widely used algorithm, selects $q/2$ pairs of α_i as the working set which violate the Karush–Kuhn–Tucker (KKT) conditions of the problem (1.1) the most, where $q \geqslant 2$ is an even number. The convergence of the SVM^{light} algorithm was proved in Ref. [8] under the condition that $\min_I(\min(eig(Q_{II}))) > 0$, where Q_{II} is any $|I| \times |I|$ sub-matrix of Q with $|I| \leqslant q$ and $\min(eig(Q_{II}))$ is the smallest eigenvalue of Q_{II}. The sequential minimum optimization (SMO) algorithm [11] is another popular decomposition algorithm which is similar to SVM^{light} but restricts the size of the working set to be two, so that the sub-optimization problem can be solved analytically. The generalized SMO algorithm proposed in Refs. [6,7] selects one the so-called τ-violating pair as the working set. The finite termination of the generalized SMO algorithm was shown in Ref. [6] without the condition that $\min_I(\min(eig(Q_{II}))) > 0$.

Recently, a polynomial time decomposition algorithm was proposed in Ref. [4] which introduced the concept of an v-rate certifying pair. It was proved in Ref. [4] that, if each working set includes at least one v-rate certifying pair then it is guaranteed that the algorithm approaches within ε of optimality after $O(1/(\varepsilon v^2))$ iterations. Further, an algorithm was given in Ref. [4] that finds an $1/m^2$-rate certifying pair in $O(m \log m)$ steps. Thus, the decomposition algorithm in Ref. [4] is within ε of optimality after $O(m^4/\varepsilon)$ iterations. In Ref. [9], the above result of Ref. [4] was generalized and improved by introducing the general notion of an v-rate certifying r-set which is an extension of the concept of the v-rate certifying pairs to the case of general convex quadratic optimization problems with $r - 1$ equality constraints. A polynomial time decomposition algorithm was also provided in Ref. [9] to find an v-rate certifying r-set. For the special case of SVMs the decomposition algorithm in Ref. [9] finds an $1/m$-rate certifying 2-set (that is, an $1/m$-rate certifying pair) in $O(m \log m)$ steps. So the decomposition algorithm in Ref. [9] is within ε of optimality after $O(m^2/\varepsilon)$ iterations, which improves on the bound obtained in Ref. [4] by factor m^2. It should be remarked that the algorithm given in Ref. [9] in selecting a rate certifying set requires to solve a linear programming problem and therefore is of higher computational cost. Note that the convergence rate of the SVM decomposition algorithms in Refs. [4,9] is given only in terms of ε and the problem parameters (like, for example, the number m of training samples and the number of equality constraints), whereas the dependence on the size of the working size is not clarified.

In this paper we propose a new simple polynomial time decomposition algorithm based on a new philosophy, which selects a working set of size $q \leqslant q_m$, where q and q_m are even numbers and q_m is the maximal size of the working set. A main feature of the algorithm is that the algorithm does not need to solve a linear programming problem in selecting the working set but guarantees that the working set selected is an $q/(4m)$-rate certifying set. So the algorithm is within ε of optimality after $O(m^2/(\varepsilon q))$ iterations. Compared with the algorithms in Refs. [4,9] our algorithm is of both lower computational cost and fast convergence rate.

The remaining part of the paper is organized as follows. Section 2 gives a detailed analysis of some existing working set selection algorithms. The new simple algorithm is given in Section 3 along with the proof of its polynomial-time convergence. The conclusions are given in Section 4.

2. Some existing working set selection algorithms

2.1. Maximally KKT-violating pairs

The most prominent approaches to working set selections are the selection of the maximally KKT-violating pairs as implemented for example in SVM^{light} [5]. In fact, the working set selection used in SVM^{light} is based on the feasible direction method. It finds the optimization directions through solving a linear optimization problem, that is, only the components of α^k with non-zero d_i are included in the working set, where k is the number of iteration and d is an optimal solution of the following problem:

$$\begin{cases} \min_d & \nabla f(\alpha^k)^\top d \\ \text{s.t.} & y^\top d = 0, \\ & -1 \leqslant d_i \leqslant 1, \quad i = 1, \ldots, m, \\ & d_i \geqslant 0 \text{ if } (\alpha^k)_i = 0; \quad d_i \leqslant 0 \text{ if } (\alpha^k)_i = C, \\ & |\{d_i | d_i \neq 0\}| \leqslant q, \end{cases} \tag{2.1}$$

where d_i is shown to be 0, -1 or 1. However, the implementation of the above optimization problem is equivalent to that derived from the maximal violation of the KKT conditions of the original problem (1.1) with the size of the working set being restricted to be even. The KKT conditions for the problem (1.1) can be written as

$$F_i(\alpha) \geqslant F_j(\alpha) \quad \forall i \in I_{up}(\alpha) \cup I_{mid}(\alpha), \\ j \in I_{low}(\alpha) \cup I_{mid}(\alpha), \tag{2.2}$$