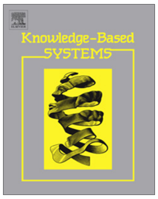
Knowledge-Based Systems 52 (2013) 223–235

Contents lists available at ScienceDirect

Knowledge-Based Systems

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



A fast algorithm for kernel 1-norm support vector machines

Li Zhang [a,*], Wei-Da Zhou [b]

[a] *Research Center of Machine Learning and Data Analysis, School of Computer Science and Technology, Soochow University, Suzhou, 215006 Jiangsu, China*
[b] *AI Speech Ltd., Suzhou, 215123 Jiangsu, China*

ARTICLE INFO

Article history:
Received 22 January 2013
Received in revised form 30 July 2013
Accepted 2 August 2013
Available online 14 August 2013

Keywords:
1-Norm SVM
Linear programming
Column generation
Newton algorithm
Kernel function

ABSTRACT

This paper presents a fast algorithm called Column Generation Newton (CGN) for kernel 1-norm support vector machines (SVMs). CGN combines the Column Generation (CG) algorithm and the Newton Linear Programming SVM (NLPSVM) method. NLPSVM was proposed for solving 1-norm SVM, and CG is frequently used in large-scale integer and linear programming algorithms. In each iteration of the kernel 1-norm SVM, NLPSVM has a time complexity of $O(\ell^3)$, where ℓ is the sample number, and CG has a time complexity between $O(\ell^3)$ and $O(n'^3)$, where n' is the number of columns of the coefficient matrix in the subproblem. CGN uses CG to generate a sequence of subproblems containing only active constraints and then NLPSVM to solve each subproblem. Since the subproblem in each iteration only consists of n' unbound constraints, CGN thus has a time complexity of $O(n'^3)$, which is smaller than that of NLPSVM and CG. Also, CGN is faster than CG when the solution to 1-norm SVM is sparse. A theorem is given to show a finite step convergence of CGN. Experimental results on the Ringnorm and UCI data sets demonstrate the efficiency of CGN to solve the kernel 1-norm SVM.

1. Introduction

In recent years, 1-norm Support Vector Machine (SVM) has attracted substantial attentions for its good sparsity [1–4]. 1-Norm SVM uses 1-norm regularization to replace 2-norm regularization in the standard SVM and approaches a more sparse model representation [5–9]. In fact, the optimization problem of 1-norm SVM is a linear program. Many methods have been proposed for solving 1-norm SVM. But, none of them is efficient enough for solving the kernel 1-norm SVM.

Since 1-norm SVM can be posed as a linear program, it can be solved by general mathematical programming methods, such as the simplex method and the interior-point method. However, these optimization methods are usually time consuming for problems in which the sample dimensionality is much less than the number of training samples.

Some specific algorithms have proposed for solving the linear program of 1-norm SVM. Fung et al. [10] presented the Newton Linear Programming SVM (NLPSVM) method to solve 1-norm SVM by minimizing the exterior penalty function for the dual problem of 1-norm SVM. In NLPSVM, each iteration has a time complexity of $O(min(m,n)^3)$ when applying the Sherman–Morrison–Woodbury identity equation [10], where m and n are the number of rows and columns of the sample matrix or the kernel gram matrix, respectively. It is quite efficient when NLPSVM is used to solve the linear 1-norm SVM. But in the case of using kernel functions, m could be equal to n, which implies that NLPSVM is still time consuming when solving the kernel 1-norm SVM. Demiriz et al. [11] introduced CG to the linear programming boosting, which can also be applied to solve 1-norm SVM by a simple generalization. In CG, the time complexity of each iteration is usually between $O(m^3)$ and $O(n^3)$ when using the simplex method [12] to directly solve the subproblem, where m and n are the number of rows and columns of the active constraint coefficient matrix in the subproblem for each iteration, respectively. More specifically, for 1-norm SVM, the complexity of the last iteration of CG is cubic of the SV number. In the standard SVM, the idea of CG is widely applied to speed up the optimization of quadratic programming [13–16].

In order to obtain a fast algorithm for the kernel 1-norm SVM, we propose a Column Generation Newton (CGN) method. This method is a combination of CG and NLPSVM. For the kernel 1-norm SVM, the coefficient matrix is a matrix with ℓ rows and ℓ columns, where ℓ is the sample number. Thus, NLPSVM would have a time complexity of $O(\ell^3)$. Let n' be the number of active constraints. Then CG thus has a time complexity between $O(\ell^3)$ and $O(n'^3)$.

In CGN, CG is used to construct a sequence of subproblems containing only active constraints, and NLPSVM is exploited to solve each subproblem. This means that CGN has a time complexity of $O(n'^3)$ in each intermediate iteration and a time complexity which is cubic of the number of ESVs in the last iteration. Since the complexity of CGN is much smaller than that of NLPSVM, it is expected that CGN is faster than NLPSVM when applied to the

[*] Corresponding author. Tel.: +86 15295659028.
E-mail address: lizhang.ml@gmail.com (L. Zhang).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.08.008

kernel case. Furthermore, CGN is faster than CG when the number of ESVs is smaller than that of SVs. As pointed out in [9], SVs is usually a small part of training samples and ESVs is only a small subset of SVs.

The rest of the paper is organized as follows. Section 2 gives a brief review of 1-norm SVM, the NLPSVM method, and the CG method. Section 3 presents the CGN algorithm for 1-norm SVM and some schemes to speed up CGN. Experimental results on the Ringnorm and UCI data sets are presented in Section 4. Section 5 concludes the paper.

2. Related work

Other than CG and NLPSVM, several other algorithms exist for 1-norm SVM. Similar to CG and NLPSVM, these methods also have high computational complexity for the kernel 1-norm SVM. In [17], three decomposition techniques for LP of 1-norm SVMs were proposed. Bradley et al. [18] proposed the Linear Programming Chunking (LPChunking) algorithm to solve the linear 1-norm SVM. LPChunking can be viewed as a block column generation method. In [19], a general technique was proposed for generalizing almost all available 2-norm SVM algorithms to the corresponding soft version by using 1-norm regularization. The soft 1-norm SVMs constructed by the proposed technique have the same convergence and an almost identically computational cost to that of the corresponding hard ones. Below, a brief review on 1-norm SVM, the NLPSVM method, and the CG method is given.

2.1. 1-Norm SVM

Consider a training sample set of two classes $\boldsymbol{X} = \{(\mathbf{x}_1,y_1), (\mathbf{x}_2,y_2), \ldots, (\mathbf{x}_\ell,y_\ell)\}$, where $\mathbf{x}_i \in \mathbb{R}^d, y_i \in \{-1,+1\}$ is the corresponding label for $\mathbf{x}_i$, d is the dimensionality of samples, and ℓ is the number of training samples. Although different LP forms exist for 1-norm SVM, they can be shown as equivalent when parameters are appropriately selected. Here, we use the LP formulation proposed in [9], which is a variant of the LP form presented in [10]. In [9], the primal problem of 1-norm SVM is described as:

$$\begin{aligned} \min_{\boldsymbol{\alpha}^+,\boldsymbol{\alpha}^-,\beta^+,\beta^-,\boldsymbol{\gamma}} \quad & \sum_{j=1}^{d}\left(\alpha_j^+ + \alpha_j^-\right) + \sigma(\beta^+ + \beta^-) + C\sum_{i=1}^{\ell}\gamma_i \\ \textbf{s.t.} \quad & y_i\left[\mathbf{x}_i^T(\boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-) + (\beta^+ - \beta^-)\right] \geqslant 1 - \gamma_i \\ & \alpha_j^+,\ \alpha_j^- \geqslant 0, j=1,\ldots,d;\ \ \beta^+,\ \beta^- \geqslant 0,\ \gamma_i \geqslant 0,\ i=1,\ldots,\ell \end{aligned} \tag{1}$$

where C is a positive penalty factor, σ is a small positive constant to ensure a unique solution, $\boldsymbol{\alpha}^+ = \left[\alpha_1^+,\ldots,\alpha_d^+\right]^T$, $\boldsymbol{\alpha}^- = \left[\alpha_1^-,\ldots,\alpha_d^-\right]^T$, $\beta^+ \in \mathbb{R}$ and $\beta^- \in \mathbb{R}$ are model coefficients for 1-norm SVM, and $\boldsymbol{\gamma} = [\gamma_1,\ldots,\gamma_\ell]^T \in \mathbb{R}^\ell$ is a loss vector. The decision function of 1-norm SVM for classification takes the form:

$$f(\mathbf{x}) = sign\left((\boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-)^T\mathbf{x} + (\beta^+ - \beta^-)\right) \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^d$ is a sample and $sign(\cdot)$ is a sign function. It has been showed that most of entries in both $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ take zeros for the sparsity of 1-norm SVM.

Rewrite (1) in its matrix form, we have

$$\begin{aligned} \min_{\mathbf{u}} \quad & \mathbf{c}^T\mathbf{u} \\ \textbf{s.t.} \quad & \mathbf{A}\mathbf{u} \geqslant \mathbf{b} \\ & \mathbf{u} \geqslant \mathbf{0} \end{aligned} \tag{3}$$

where $\mathbf{c} = [\mathbf{1}^T,\mathbf{1}^T,\sigma,\sigma,C\mathbf{1}^T]^T$, the variable vector $\mathbf{u} = [\boldsymbol{\alpha}^{+T},\boldsymbol{\alpha}^{-T},\beta^+,\beta^-,\boldsymbol{\gamma}^T]^T$, $\mathbf{b} = \mathbf{1}$, the constraint coefficient matrix $\mathbf{A} = [\mathbf{D}_y\mathbf{X}, -\mathbf{D}_y\mathbf{X}, \mathbf{y}, -\mathbf{y}, \mathbf{I}]$, $\mathbf{X} \in \mathbb{R}^{\ell\times d}$ is the sample matrix in which each row vector is a training sample, $\mathbf{y} = [y_1,y_2,\ldots,y_\ell]^T$, $\mathbf{D}_y$ is the diagonal matrix with the diagonal line of $\mathbf{y}$, and $\mathbf{1}$ and $\mathbf{I}$ are the column vector of all ones and the identity matrix with proper sizes, respectively. The dual problem of (7) can be expressed as follows:

$$\begin{aligned} \max_{\mathbf{v}} \quad & \mathbf{b}^T\mathbf{v} \\ \text{s.t.} \quad & \mathbf{A}^T\mathbf{v} \leqslant \mathbf{c} \\ & \mathbf{v} \geqslant \mathbf{0} \end{aligned} \tag{4}$$

In order to generalize the linear 1-norm SVM to the nonlinear 1-norm SVM, a group of nonlinear mapping functions $\phi_j(\mathbf{x}), j = 1, \ldots, D$ are introduced to map the sample $\mathbf{x}$ into a feature space: $\mathbf{x} \to \Phi(\mathbf{x}) = [\phi_1(\mathbf{x}),\phi_2(\mathbf{x}), \ldots, \phi_D(\mathbf{x})]^T$. Currently the most popular mapping functions are still the Mercer kernel functions [20,21],

$$\Phi(\mathbf{x}) = [k(\mathbf{x}_1,\mathbf{x}), k(\mathbf{x}_2,\mathbf{x}), \ldots, k(\mathbf{x}_\ell,\mathbf{x})]^T$$

where $k(\mathbf{x}_i,\mathbf{x})$ is a Mercer kernel function, such as the Radial Basis Function (RBF) kernel $k(\mathbf{x}_i,\mathbf{x}) = \exp\left(-\|\mathbf{x}_i - \mathbf{x}\|_2^2/2p^2\right)$ with p being the RBF kernel parameter. Therefore, the nonlinear 1-norm SVM is often referred to as the kernel 1-norm SVM. The linear programming of the kernel 1-norm SVM has the same form as that of the linear 1-norm SVM except that the sample matrix $\mathbf{X}$ is replaced by the kernel Gram matrix $\mathbf{K} \in \mathbb{R}^{\ell\times\ell}$ in the constraint coefficient matrix $\mathbf{A}$, i.e. $\mathbf{A} = [\mathbf{D}_y\mathbf{K}, -\mathbf{D}_y\mathbf{K}, \mathbf{y}, -\mathbf{y}, \mathbf{I}]$, where $K_{ij} = k(\mathbf{x}_i,\mathbf{x}_j)$. To unify the expression of $\mathbf{A}$, we let $\mathbf{A} = [\mathbf{A}_s, -\mathbf{A}_s, \mathbf{y}, -\mathbf{y}, \mathbf{I}]$, where $\mathbf{A}_s = \mathbf{D}_y\mathbf{X}$ in the case of the linear 1-norm SVM and $\mathbf{A}_s = \mathbf{D}_y\mathbf{K}$ in the case of the kernel 1-norm SVM.

Now we give some notations borrowed from [9], which are important to analyze the complexity of the algorithms of CG and NLPSVM. Assume that 1-norm SVM (7) has an optimal solution $\mathbf{u}^*$, the set of Exact Support Vectors (ESVs) is defined as $ESV = \{\mathbf{x}_i|y_if(\mathbf{x}_i) = 1, i = 1, \ldots, \ell\}$, the set of Saturated Support Vectors (SSVs) is defined as $SSV = \{\mathbf{x}_i|y_if(\mathbf{x}_i) < 1, i = 1, \ldots, \ell\}$, the set of Non-Support Vectors (NSVs) is defined as $NSV = \{\mathbf{x}_i|y_if(\mathbf{x}_i) > 1, i = 1, \ldots, \ell\}$, and the set of Support Vectors (SVs) is thus the union of ESVs and SSVs: i.e., $SV = ESV \cup SSV$.

The following theorem is about the sparsity of 1-norm SVM [9], which indicates that 1-norm SVM has a better sparsity than the standard SVM and motivates our CGN method.

Theorem 1. *Let $\mathbf{u}^*$ be an optimal solution of the primal problem (7). The number of nonzero coefficients for 1-norm SVM is upper bounded by*

$$|NC| \leqslant |ESV|,$$

where $NC = \left\{\alpha_j^*|\alpha_j^* = \alpha_j^{+*} - \alpha_j^{-*} \neq 0, j = 1,\ldots,d(orD)\right\}$. If the LP of 1-norm SVM is non-degenerate with respect to $\mathbf{u}^*$, the following equality

$$|NC| = |ESV| - 1$$

holds.

In the standard SVM, the number of nonzero coefficients equals to the number of SVs. The SSVs set is usually nonempty, especially in some difficult recognition problems which have much more SSVs than ESVs. Accordingly, 1-norm SVM usually has a better sparsity than the standard SVM. By the Complementary Slackness Theorem [12] and Theorem 1, it is straightforward that there are at least $|ESV|$ active constraints among $\mathbf{A}_s^T\mathbf{v} \leqslant \mathbf{1}$ and $-\mathbf{A}_s^T\mathbf{v} \leqslant \mathbf{1}$ in the dual problem (4).

2.2. NLPSVM

In NLPSVM [10,22], the dual problem of 1-norm SVM can be transformed into a convex unconstrained optimization problem