基于活跃集的支持向量机切平面法

肖 锋,周 杰
(清华大学自动化系,北京 100084)

摘 要: 切平面法作为求解非光滑凸优化问题的典型方法,在支持向量机问题的求解中得到了广泛的应用.但是该算法在求解过程中往往会出现不稳定的情况.针对这一不稳定性,前人提出了优化切平面法,通过在切平面法中加入线搜索环节来确保目标函数单调下降.但是优化切平面法的运算复杂度比较高,不适合训练数据量大、对训练速度要求高的应用.本文提出了一种基于活跃集的优化切平面法,在计算目标函数和进行线搜索时,只单独处理活跃集内的样本,将其它样本当作一个整体来进行处理.相对于传统的优化切平面法,本文方法只需在一部分样本上计算目标函数和进行线搜索,从而可以在不损失求解精度的前提下节省求解时间.

关键词: 切平面法; 支持向量机; 优化切平面法; 活跃集
中图分类号: TN911.23 **文献标识码:** A **文章编号:** 0372-2112 (2013) 04-0757-06
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2013.04.022

Active Set Based Cutting Plane Algorithm for SVM

XIAO Feng, ZHOU Jie
(*Department of Automation, Tsinghua University, Beijing* 100084, *China*)

Abstract: As a typical method for solving non-smooth convex optimization problems, cutting plane method is widely used in solving support vector machine problems. However, this algorithm suffers from the instability problem. To ease this instability, researchers proposed an optimized cutting plane algorithm which incorporated a line search stage. However, the computational complexity of such algorithm is too high for applications where the number of training samples is large. In this paper we propose an active set based optimized cutting plane algorithm to reduce the computation complexity of the original algorithm. When computing the objective function and performing line search, only those samples which fall in the active set are considered. Compared to optimized cutting plane algorithm, the proposed algorithm needs to calculate the objective function and perform line search only for a small fraction of the samples, leading to a significant drop in computational complexity without losing accuracy.

Key words: cutting plane algorithm; support vector machine; optimized cutting plane algorithm; active set

1 引言

支持向量机是机器学习领域中的一个重要研究方向[1,2].在支持向量机中,优化的目标函数包括两项,分别是经验风险损失函数和正则化项.这两项通过正则化系数联系起来,构成了结构化的风险损失函数.在对大规模数据集进行训练时,由于每个样本都会引入一个约束,因此支持向量机问题的约束往往比较多,使得设计一些特殊的优化算法成为必需.序贯最小化方法[3]、随机梯度下降方法[4]、对偶坐标下降方法[5]、以及切平面法[6]是比较常用的优化方法.其中切平面法采用一系列超平面来逼近待优化的函数,通过求解多面体约束的约化问题来近似求解原优化问题.这一算法能够用统一的框架来处理两类和多类分类问题、线性和非线性支持向量机问题,在近年得到了特别的重视[6~8].文献[4]中作者比较了切平面法和随机梯度下降方法,认为在一些问题上,切平面法的效率要高于随机梯度下降算法.文献[7]对切平面法的应用范围及其收敛判据进行了系统的分析.

传统的切平面法存在一定的不稳定性,目标函数会发生振荡,很多文献也分析了这种不稳定性[9~11].切平面法的不稳定性会显著提高计算的复杂度,因此消除这一不稳定性是很有必要的.在优化领域的研究中[10],一般通过对目标函数进行线搜索来解决.文献[12]针对线搜索这一环节来对切平面法进行优化,仅需对次梯度的间断点进行排序.相对于传统的切平面法,该算法用比较简单的线搜索消除了算法的不稳定性,减少了迭代次数.但是在样本较多时,对次梯度的间断点进行排序仍然会花费较多时间.

收稿日期:2012-05-11;修回日期:2012-12-19
基金项目:国家自然科学基金(No.61225008,No.61020106004,No.61021063,No.61005023)

本文提出活跃集的概念,对优化切平面法进行改进,以大幅降低其训练复杂度.活跃集包含所有可能在下次迭代中越过分类边缘的样本,活跃集外的样本的目标函数是参数的线性函数.当参数没有明显的振荡时,就能够避免大量不必要的计算.

2 切平面法与优化切平面法

2.1 切平面法

假定样本集合$\{(\boldsymbol{x}_1,y_1),\cdots,(\boldsymbol{x}_m,y_m)\}$的维度为 n,样本数为 m.支持向量学习问题需要求解如式(1)所示的约束优化问题:

$$\boldsymbol{\omega}=\arg\min_{\boldsymbol{\omega}}F(\boldsymbol{\omega})=\arg\min_{\boldsymbol{\omega}}\frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega}+CR(\boldsymbol{\omega}) \quad (1)$$

其中 $R(\boldsymbol{\omega})=\sum_{i=1}^{m}\max\{0,1-y_i\langle\boldsymbol{\omega},\boldsymbol{x}_i\rangle\}/m$.

本文将平行于分类面的超平面$\langle\boldsymbol{\omega},\boldsymbol{x}\rangle=\pm1$称为分类边缘.在支持向量学习问题中,一个特定样本在跨越其对应的分类边缘时,对目标函数的贡献会发生突变.切平面法[6~8]通过一系列的超平面来切割可行域得到原优化问题的约化问题,如式(2)所示:

$$(\boldsymbol{\omega},\xi_t)=\arg\min_{(\boldsymbol{\omega},\xi)}F_t(\boldsymbol{\omega})=\arg\min_{(\boldsymbol{\omega},\xi)}\{\frac{1}{2}\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{\omega}+C\xi\} \quad (2)$$

$$\text{s.t. }\xi\geqslant\langle\boldsymbol{a}_i,\boldsymbol{\omega}\rangle+b_i,\xi\geqslant0,i=1,\cdots,t$$

其中$\langle\boldsymbol{a}_i,\boldsymbol{\omega}\rangle+b_i,i=1,\cdots,t$是一系列的超平面,$\boldsymbol{\omega}$为待求的参数向量,$\boldsymbol{a}_i$和$b_i$为第$i$个切平面的系数,$t$为超平面的个数,$F_t(\boldsymbol{\omega})$为约化问题的目标函数.

该约化问题的对偶问题仍是一个二次规划问题,其变量数目等于超平面的个数.在特征维数较高时,超平面的个数一般比特征维数要小.所以对于这一优化问题,求解其对偶问题效率更高.其对偶问题如式(3)所示:

$$\alpha_t=\arg\max D_t(\alpha)=\sum_{i=1}^{t}\alpha_ib_i-\frac{1}{2}\|\sum_{i=1}^{t}\boldsymbol{a}_i\alpha_i\|^2 \quad (3)$$

$$\text{s.t. }\sum_{i=1}^{t}\alpha_i\leqslant C,\alpha_i\geqslant0,i=1,\cdots,t$$

得到对偶问题的解α_t之后,主问题的解$\{\boldsymbol{\omega}_t,\xi_t\}$可由式(4)确定:

$$\boldsymbol{\omega}_t=-\sum_{i=1}^{t}[\alpha_t]_i\boldsymbol{a}_i,\qquad \xi_t=\max(\langle\boldsymbol{\omega}_t,\boldsymbol{a}_i\rangle+b_i) \quad (4)$$

将主问题的解$\{\boldsymbol{\omega}_t,\xi_t\}$代入$F_t(\boldsymbol{\omega})$的表达式(2)中即可得到约化问题的目标函数值$F_t(\boldsymbol{\omega}_t)$.

在用切平面法求解优化问题时,一般采用原问题和约化问题的差距$(F(\boldsymbol{\omega}_t^b)-F_t(\boldsymbol{\omega}_t))$或相对差距$(F(\boldsymbol{\omega}_t^b)-F_t(\boldsymbol{\omega}_t))/F(\boldsymbol{\omega}_t^b)$来度量问题解的与最优解的距离[7,8,12].其中F和F_t的形式分别如式(1)和式(2)所示.我们用ε表示差距,用ε表示相对差距.本文采用相对差距小于给定值作为停止条件,比如$\varepsilon=(F(\boldsymbol{\omega}_t^b)-F_t(\boldsymbol{\omega}_t))/F(\boldsymbol{\omega}_t^b)\leqslant1\mathrm{e}-5$等.

2.2 优化切平面法以及近似优化切平面法

与切平面法不同,优化切平面法[12]需要保留两个参数序列$\boldsymbol{\omega}_t^b$和$\boldsymbol{\omega}_t$.其中$\boldsymbol{\omega}_t^b$是到当前为止原问题的最优解,$\boldsymbol{\omega}_t$是约化问题的最优解.在求解约化问题(2)之后,还需如式(5)所示进行线搜索得到最优步长λ_t^*以及$\boldsymbol{\omega}_t^b$:

$$\begin{cases}\lambda_t^*=\arg\min_{\lambda\geqslant0}F(\boldsymbol{\omega}_{t-1}^b(1-\lambda)+\boldsymbol{\omega}_t\lambda)\\ \boldsymbol{\omega}_t^b=\boldsymbol{\omega}_{t-1}^b(1-\lambda_t^*)+\boldsymbol{\omega}_t\lambda_t^*\end{cases} \quad (5)$$

得到$\boldsymbol{\omega}_t^b$之后,需要按照式(6)在$\boldsymbol{\omega}_t^b$附近选择一个点$\boldsymbol{\omega}_t^c$:

$$\boldsymbol{\omega}_t^c=\boldsymbol{\omega}_t^b(1-\mu)+\boldsymbol{\omega}_t\mu,\mu=0.05 \quad (6)$$

式(6)中的参数μ可以取(0,1)之间的任意值.我们的实验表明,选择$\mu=0.05$整体效果最好.切平面如式(7)所示($\boldsymbol{a}$和b即为式(2)中切平面的系数)

$$\boldsymbol{a}=-\sum_{i:y_i\langle\omega_t^c,x_i\rangle\leqslant1}y_ix_i/m,\qquad b=R(\omega_t^c)-\langle\omega_t^c,\boldsymbol{a}\rangle \quad (7)$$

文献[13]改进了文献[12]中的精确线搜索,提出了基于三点近似线搜索的切平面法.

2.3 收敛性及切平面的合并

切平面法和优化切平面法的收敛性由如下定理保证:

定理 1[12] 假定$\|\partial R(\boldsymbol{\omega})\|\leqslant G$对所有$\boldsymbol{\omega}\in\boldsymbol{\Omega}$均成立,其中$\boldsymbol{\Omega}$是包含了之前迭代所有的$\boldsymbol{\omega}$的一个区域,则$\varepsilon_t-\varepsilon_{t+1}\geqslant\varepsilon_t\min\{1,\varepsilon_t/4C^2G^2\}/2$.

其证明可参考文献[8]或[12].主要思路是将原命题的证明转化为证明$F_{t+1}(\boldsymbol{\omega}_{t+1})-F_t(\boldsymbol{\omega}_t)$不小于$\boldsymbol{\varepsilon}_t\min\{1,\boldsymbol{\varepsilon}_t/4C^2G^2\}/2$.在切平面集的大小$N=2$时,可以证明$F_{t+1}(\boldsymbol{\omega}_{t+1})-F_t(\boldsymbol{\omega}_t)$的下界就是上式等号的右端项.当$N>2$时也自然满足收敛性条件.但是切平面集合的容量太小时,偏差ε_t的下降会接近下界.

在切平面法中,每次增长都会加入一个切平面,带来巨大的存储和计算负担.因此一般会对切平面集合进行合并.我们采用比较简单的切平面合并策略,即固定切平面集合的大小N.在切平面数量达到N以后,每次迭代将最早加入的两个向量合并成一个向量.本文将切平面集的容量设定为$N=20$.

2.4 优化切平面法的分析

本文主要在线搜索和计算目标函数这两个环节对优化切平面法进行改进.

在优化切平面法中,线搜索的范围为从$\boldsymbol{\omega}_{t-1}^b$出发并经过$\boldsymbol{\omega}_t$的射线,也即式(5)中的$\lambda\geqslant0$.通过对优化切平面法的分析和实验我们发现这一范围存在一定的冗

余.首先,实验表明,除开始几个循环外,最优步长往往比较小或者变化幅度比较小.其次,优化切平面法中原问题是对精确的目标函数寻优,而对偶问题是对目标函数的线性近似寻优.因此原问题的解应该占更大的权重,最优步长 λ_t^* 应该比较小.由于上述原因,最优步长的搜索范围可以适当减小.

此外,最优步长的大小还对目标函数的计算量造成影响.如果最优步长比较小,那么连续的两次迭代中分类边缘的变化就比较小.虽然最优步长只有在线搜索之后才能得到,但是如果最优步长一般都比较小且波动不大,那么可以依照之前几次迭代的最优步长来给出一个上界.因此,没有跨过分类边缘的样本可以被当作一个整体来处理,其梯度在参数变化时可视为不变,其函数值可视为参数的线性函数.

3　基于活跃集的优化切平面法

基于上一节的分析,在本节提出基于活跃集的优化切平面法.为简化分析,假设所有样本已经过归一化处理,即 $\| \boldsymbol{x} \| = 1$.首先定义一个活跃集 X_t^a,它包括了第 $t+1$ 次迭代时可能会跨过分类边缘的样本,其具体表达式稍后引入,用 $\overline{X_t^a}$ 表示其余集.记全体样本的集合为 X,初始化时认为 $X_0^a = X$,即每个样本都有可能跨过分类边缘.用 D_t^i 来表示第 t 次迭代时样本点 i 到其对应的分类边缘的距离,D_t^i 的表达式如式(8)所示:

$$D_t^i = \begin{cases} |1 - y_i\langle \boldsymbol{\omega}_t^b, \boldsymbol{x}_i\rangle| & \text{若 } i \in X_{t-1}^a \\ D_{t-1}^i - \lambda_t^* \| \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b \| & \text{若 } i \notin X_{t-1}^a \end{cases} \tag{8}$$

对于属于 X_{t-1}^a 的样本,D_t^i 就是该样本点到其对应的分类边缘的距离,即 $|1 - y_i\langle \boldsymbol{\omega}_t^b, \boldsymbol{x}_i\rangle|$.而对于不属于 X_{t-1}^a 的样本,则其对应的离分类边缘的距离只能用上一迭代的值进行估计,式(8)将其设定为用 D_{t-1}^i 减去本次迭代距离变化的上界.由于 D_t^i 与 $\boldsymbol{\omega}$ 成正比,因此在相邻两次迭代中,其距离的变化的上界是 $\max |\langle \boldsymbol{\omega}_t^b - \boldsymbol{\omega}_{t-1}^b, \boldsymbol{x}_i\rangle| \leq \lambda_t^* \| \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b \|$.

假设第 t 次迭代的最优步长为 λ_t^*,我们要求下一次迭代的距离变化不能超过 $(\max\{\lambda_{t-3}^*, \cdots, \lambda_t^*\} + \mu) \| \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b \|$,其中 $\mu = 0.05$.当第 $t+1$ 次迭代时求得 $\boldsymbol{\omega}_{t+1}$ 后,我们即可得到相应的步长上界 Λ_{t+1},如式(9)所示:

$$\Lambda_{t+1} = (\max\{\lambda_{t-3}^*, \cdots, \lambda_t^*\} + \mu) \frac{\| \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b \|}{\| \boldsymbol{\omega}_{t+1} - \boldsymbol{\omega}_t^b \|} \tag{9}$$

由于第 $t+1$ 次迭代的步长还没有指定,因此这一要求一定可以满足.但是需要设定一个较合理的上界,以便可以在较多的迭代中取到最优的步长.上界的形式利用了过去几次迭代的距离变化的最大值的信息.在最大步长上加上 μ 是为了不影响切平面的计算.

X_t^a 的表达式如式(10)所示,它包括了所有到分类边缘的距离小于下一迭代距离变化的上界的样本:

$$X_t^a = \{ i \mid D_t^i < (\max\{\lambda_{t-3}^*, \cdots, \lambda_t^*\} + \mu) \| \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b \| \} \tag{10}$$

在少数情况下,第 $t+1$ 步的最优步长会超过指定上界 Λ_{t+1},从而 $\overline{X_t^a}$ 中的某些样本也有可能越过分类边缘,进而影响到算法的精度.在这种情况下,我们放弃采用最优步长,而是将第 $t+1$ 步的步长设定为满足指定上界的值.通过放弃一部分线搜索的最优性,可以保证目标函数和切平面的计算不受影响.

在第 t 次迭代时,我们将目标函数转成以 λ 为变量的函数,如式(11)所示:

$$f(\lambda) = A_0\lambda^2/2 + B_0\lambda + C_0 + \sum_{i \in X_{t-1}^a} \max\{0, \lambda B_i + C_i\} \tag{11}$$

其中 $A_0 = \| \boldsymbol{\omega}_{t-1}^b - \boldsymbol{\omega}_t \|^2$,

$B_0 = \langle \boldsymbol{\omega}_{t-1}^b + \boldsymbol{a}_x, \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b \rangle$,

$C_0 = \| \boldsymbol{\omega}_{t-1}^b \|^2/2 + \langle \boldsymbol{\omega}_{t-1}^b, \boldsymbol{a}_X \rangle$,

$B_i = Cy_i\langle \boldsymbol{\omega}_{t-1}^b - \boldsymbol{\omega}_t, \boldsymbol{x}_i \rangle / m$,

$C_i = C(1 - y_i\langle \boldsymbol{\omega}_{t-1}^b, \boldsymbol{x}_i \rangle)/m, i \in X_{t-1}^a$.

对式(11)求导可得函数 f 在 λ 处的次梯度为 $\partial f(\lambda) = A_0\lambda + B_0 + \sum_{i \in X_{t-1}^a} \partial f_i(\lambda) + \langle \boldsymbol{\omega}_t - \boldsymbol{\omega}_{t-1}^b, \boldsymbol{a}_x \rangle$.其中 $\boldsymbol{a}_X$ 为 $\overline{X_{t-1}^a}$ 中样本的总的次梯度.化简可得次梯度 $\partial f(0)$ 的最大值 $\max(\partial f(0))$ 和 $\partial f(\Lambda_t)$ 的最小值 $\min(\partial f(\Lambda_t))$.

由于 $F(\boldsymbol{\omega})$ 是 $\boldsymbol{\omega}$ 的凸函数,$\boldsymbol{\omega}$ 是 λ 的线性函数,而凸函数的复合函数还是凸函数,因此 $f(\lambda)$ 仍是 λ 的凸函数.在 $f(\lambda)$ 的最小值处有 $0 \in \partial f(\lambda)$.若 $\max(\partial f(0)) \leq 0$ 且 $\min(\partial f(\Lambda_t)) \geq 0$,则 $\lambda_t^* \in (0, \Lambda_t)$,说明最小值确实在预估范围内.若 $\max(\partial f(0)) > 0$,则说明该线搜索方向不是下降方向,在这种情况下不再进行线搜索,直接将 λ_t^* 置为 0.若 $\min(\partial f(\Lambda_t) < 0$,则说明预估范围偏小.为确保步长不越界,不再进行线搜索,将步长指定为 $0.9\Lambda_t$,同时在 Λ_t 处计算切平面.通过以上分析可以得到需要进行线搜索的集合 X_t^s,其表达式如式(12)所示:

$$X_t^s = \begin{cases} \{ i \mid 0 < \lambda_i < \Lambda_t \} & \text{若 } \max(\partial f(0)) \leq 0 \text{ 且 } \min(\partial f(\Lambda_t)) \geq 0 \\ \Phi & \text{若 } \min(\partial f(\Lambda_t)) < 0 \text{ 或 } \max(\partial f(0)) > 0 \end{cases} \tag{12}$$

线搜索通过对集合 X_t^s 中的 λ_i 值进行排序,然后增加 λ 的值直到 $0 \in \partial f(\lambda)$ 为止,主要的时间开销是对集合 X_t^s 进行排序.因此本文对线搜索的加速也主要体现在减少了 X_t^s 所包含的样本上.

在通过线搜索得到最优的 λ_t^* 之后,需要计算出切

平面并插入到切平面集合中去.对$\overline{X_{t-1}^a}$中的样本,可以沿用上一次迭代的总的切平面(也即总的次梯度)$\boldsymbol{a}_X$.

基于活跃集的切平面法算法流程如算法 1 所示.

4 计算复杂度分析

本文不对迭代次数进行优化,因此我们可以只对每步的计算量进行分析.每步花费时间较多的环节包括求解二次规划,线搜索,目标函数计算,添加新的切平面,权重向量计算等五个环节.求解二次规划采用的是已有的算法,此处没有列出.假设样本数为 l,特征维度为 m,每个样本平均非零元为 $\bar{m}$,切平面集的容量按照前述定义为 N,X_t^a 和 X_t^s 的元素个数分别为 p 和 q.则优化切平面法的各个步骤的计算复杂度如下:线搜索的计算量为 $l\log(l)$次比较,目标函数的计算量为 $l\bar{m}$ 次乘法.由于切平面法中的切平面不是稀疏的,因此计算新的切平面的复杂度为 $l\bar{m}$ 次加法,将切平面加入集合中的复杂度为 Nm 次乘法.计算权重向量的复杂度为 Nm 次乘法和 Nm 次加法.本文提出的算法在其中三个环节会减小计算量,见表 1.其中括号外是文献[12]的算法,括号内是本文算法的计算量,计算量未改进的没有重复标出.在添加新的切平面的计算中,由于切平面的一部分可以沿用上一次迭代的值,所以加法的计算量减少了,但由于这一部分减少的是加法运算,因此下文也不再单列出来分析.线搜索和目标函数的计算量减小得比较明显.由于 N 为固定值,因此 Nm 相对于 $l\bar{m}$ 一般来说比较小.

算法 1 基于活跃集的优化切面算法

1 $t\leftarrow 1, \boldsymbol{\omega}_0^b \leftarrow \bar{0}, X_0^a = X$

2 While 终止条件未满足 do

3 计算约化问题求得 $\boldsymbol{\omega}_t$

4 若切面集合中的切面数大于 N,则将最早加入的两个切平面合并.

5 计算 X_{t-1}^a 中的样本的目标函数值,并计算$\partial f(0)$和$\partial f(\lambda)$.若 $\max(\partial f(0))>0$,则 $\lambda_t^*=0$;若 $\min(\partial f(\Lambda_t))<0$,则 $\lambda_t^*=0.9\Lambda_t$;若 $\max(\partial f(0))\leqslant 0$ 且 $\min(\partial f(\Lambda_t))\geqslant 0$ 按照式(12)计算 X_t^s.
在 X_t^s 上进行线搜索得到λ_t^* 和$\boldsymbol{\omega}_t^b$.

6 若$(1-\mu)\lambda_t^*+\mu<\Lambda_t$,则按照式(6)计算 $\boldsymbol{\omega}_t^c$;否则
$\omega_t^c=(1-\Lambda_t)\omega_{t-1}^b+\Lambda_t\boldsymbol{\omega}_t$.

7 在 $\boldsymbol{\omega}_t^c$ 处计算新的切平面并添加到切平面集合中.X_{t-1}^a的样本对切平面的贡献由式(7)得到,$\overline{X_{t-1}^a}$对切平面的贡献沿用上一次迭代的值.

8 计算 D_t^i,X_t^a 和$\overline{X_t^a}$.对$\overline{X_t^a}$中的样本,按照式(7)计算总的次梯度 $\boldsymbol{a}_X$.

9 若$(F(\boldsymbol{\omega}_t^b)-F_t(\boldsymbol{\omega}_t))/F(\boldsymbol{\omega}_t^b)\leqslant\varepsilon$,则终止.

10 $t\leftarrow t+1$

11 end while

12 return $\boldsymbol{\omega}_t^b$

表 1 各个步骤的计算量

	乘法	加法	比较
线搜索	$l(q)$	0	$l\log l(q\log q)$
目标函数	$l\bar{m}(p\bar{m})$	$l\bar{m}(p\bar{m})$	0
添加切平面	Nm	$l\bar{m}+Nm(p\bar{m}+Nm)$	0
计算权重向量	Nm	Nm	0

5 实验结果与分析

我们在机器学习领域内常用的数据集上进行了实验,并将所提算法与文献[12]和文献[13]的算法进行了比较.实验所用到的数据集为 CCAT[14], COV1, NEWS20[15]和 REAL-SIM(这些数据集可以从文献[16]和文献[17]下载).对于没有给出测试集的数据集,本文按照 9:1 的比例来划分训练集和测试集.对于标签分布不均匀的数据库,采用随机抽取的方法来保证得到的训练集和测试集的标签分布是基本相同的.实验在 CPU 为 Intel Xeon 5620,内存为 32GB,系统为 ubuntu11.10 的服务器上完成.各个数据库的信息如表 2 所示.

表 2 实验所用数据库的信息

	CCAT	COV1	REAL-SIM	NEWS20
训练样本数	781,265	522,912	65,078	17,959
测试样本数	23,149	58,100	7,231	1,955
特征维度	47,152	54	20,958	1,355,191
非零特征数	59155144	6246016	3340340	8205212
平均非零特征数	75.7	11.94	50.84	456.89

我们比较了本文算法和文献[12,13]中的算法优化效率.图 1 列出了采用不同算法时相对偏差随时间的变化情况,其中各个算法的切平面集限制为最多包括 20 个切平面.从图中可以看出,本文提出的算法在大多数数据集上都取得了最好的效果.

我们还比较了在不同的 C 值下本文算法的迭代次数、X_t^a 中的样本数的平均值和 X_t^s 中的样本数的平均值.以文献[12]的算法的对应量进行归一化,得到如图 2 所示的柱状图.终止条件选为 $\varepsilon/C\leq 1e-5$.从图 2 可以看出,本文算法计算目标函数和线搜索的样本数量比优化切平面法有了较大的减小.此外,虽然在一些迭代中我们没有采用最优的步长,但是达到指定精度时本文算法和优化切平面法需要的迭代次数比较接近.在部分数据集上还比优化切平面法要少一些,这说明局部的最优步长不一定是全局最优的.

由于求解支持向量机的主要目的是要用得到的模型来进行分类,因此在测试集上的性能也是很重要的指标.表 3 统计了为达到给定精度 ε,本文算法和文献[12]的算法计算目标函数和线搜索所花的时间以及测

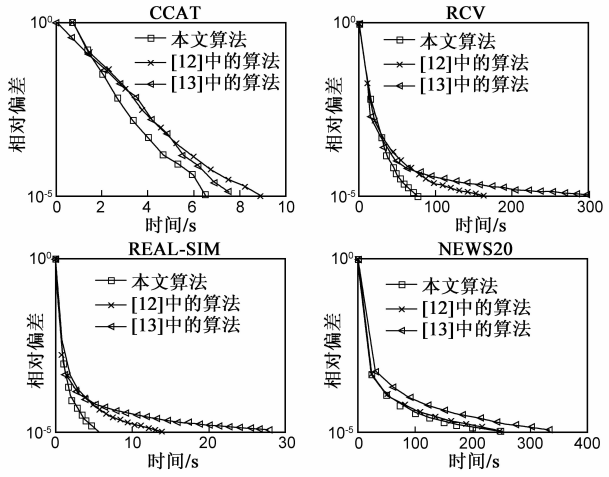

图1　不同的算法中优化目标的相对偏差随时间的变化

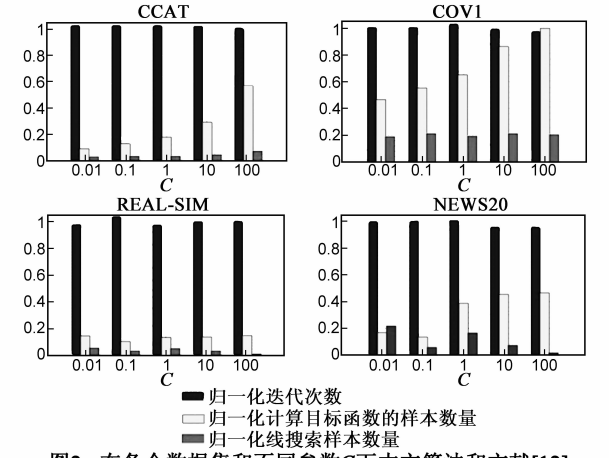

图2　在各个数据集和不同参数C下本文算法和文献[12]算法的对比

试错误率.正则化系数 C 取为 1.T1 为计算目标函数花费的时间,T2 为线搜索花费的时间,err 为测试错误率,T1 和 T2 均以秒为单位,err 用百分比表示,从实验结果上来看,本文算法和文献[12]的算法得到的模型在测试集上的错误率基本没有差异,而计算目标函数和线搜索的时间都比文献[12]的算法有大幅度的减少.

表 3　达到不同求解精度时各步骤花费的时间(s)和测试错误率(%)

数据集 \ 精度		1e-2		1e-3		1e-4		1e-5	
		[12]算法	本文算法	[12]算法	本文算法	[12]算法	本文算法	[12]算法	本文算法
CCAT	T1	7.5	7.4	14.7	12.7	33.3	17.6	93.9	23.9
	T2	2.4	0.6	4.8	0.7	10.6	0.7	29.2	0.8
	err	5.06	5.05	5.04	5.05	5.03	5.03	5.04	5.04
COV1	T1	0.7	0.7	1.0	0.9	1.5	1.2	2.1	1.6
	T2	1.3	0.7	1.9	0.7	2.9	0.8	4.0	0.8
	err	32.9	32.9	32.6	32.6	32.4	32.5	32.2	32.2
REAL-SIM	T1	0.2	0.2	0.6	0.4	2.0	0.7	7.3	1.3
	T2	0.1	0.02	0.3	0.02	0.9	0.04	2.8	0.1
	err	2.34	2.34	2.35	2.35	2.35	2.35	2.37	2.37
NEWS20	T1	1.2	1.1	4.1	2.6	14.8	6.9	67.6	31.0
	T2	0.04	0.01	0.1	0.02	0.4	0.1	1.7	0.3
	err	2.71	2.71	2.66	2.66	2.66	2.66	2.66	2.66

6　结论

本文提出了一种基于活跃集的用于支持向量学习的优化切平面方法.在求解过程中,预先估计不会越过分类边缘的样本集合.这样就可以采用统一的线性函数来近似不会越过分类边缘的样本的目标函数,用常向量来近似其梯度,从而减少了优化切平面法中的两个关键步骤(计算目标函数和线搜索)需要处理的样本的数量.与优化切平面法相比,在相同的优化指标和相近的模型精度下,目标函数的计算量最多可以减小 88.0%,线搜索的计算量可以减小 46% – 97%.

参考文献

[1] Vladimir Vapnik. The Nature of Statistical Learning Theory [M]. New York: Springer, 2000.

[2] 王国胜,钟义信.支持向量机的若干新进展[J].电子学报,2001,29(10):1397 – 1400.
Guosheng Wang, Yixin Zhong. Some new developments on support vector machine[J]. Acta Electronica Sinica, 2001, 29(10):1397 – 1400.(in Chinese)

[3] Schölkopf B, Burges C J C, Smola A J, eds. Advances in Kernel Methods[M]. Cambridge, MA, USA: MIT Press, 1999.

[4] Shalev-Shwartz S, Singer Y, Srebro N, et al. Pegasos: primal estimated sub-gradient solver for SVM[J]. Mathematical Programming, 2011, 127(1):3 – 30.

[5] Fan R, Chang K, Hsieh C, et al. LIBLINEAR: a library for large linear classification[J]. The Journal of Machine Learning Research, 2008, (9):1871 – 1874.

[6] Joachims T. Training linear SVMs in linear time[A]. Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining [C]. Philadelphia: ACM, 2006. 217 – 226.

[7] Joachims T, Finley T, Yu C. Cutting-plane training of structural SVMs[J]. Machine Learning, 2009, 77(1):27 – 59.

[8] Teo C, Vishwanthan S, Smola A, et al. Bundle methods for regularized risk minimization[J]. The Journal of Machine Learning Research, 2010, (11):311 – 365.

[9] Nemirovski A, Yudin D. Problem Complexity and Method Efficiency in Optimization[M]. New York: Wiley, 1983.

[10] Hiriart-Urruty J, Lemaréchal C. Convex Analysis and Minimization Algorithms: Fundamentals[M]. New York: Springer-Verlag, 1996.

[11] Bertsekas D. Nonlinear Programming[M]. Belmont, Mass: Athena Scientific, 1999.

[12] Franc V, Sonnenburg S. Optimized cutting plane algorithm for large-scale risk minimization[J]. The Journal of Machine Learning Research, 2009, (10): 2157 – 2192.

[13] Arnosti N, Kalita J. Cutting Plane Training for Linear Support Vector Machines[J]. IEEE Transactions on Knowledge and Data Engineering, 2011, 99: 1 – 6.

[14] Lewis D, Yang Y, Rose T, et al. RCV1: A new benchmark collection for text categorization research[J]. The Journal of Machine Learning Research, 2004, (5): 361 – 397.

[15] Keerthi S, DeCoste D. A modified finite Newton method for fast solution of large scale linear SVMs[J]. Journal of Machine Learning Research, 2006, 6(1): 341.

[16] LIBSVM dataset[DB/OL]. http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/, 2012 – 12 – 10.

[17] SGD dataset[DB/OL]. ttp://leon.bottou.org/projects/sgd, 2012 – 12 – 10.

作者简介

肖 锋 男,1982 年生于湖北宜昌,清华大学自动化系博士研究生.主要研究方向包括姿态估计、图像处理、图片分类.

周 杰 男,1968 年生于河南信阳,清华大学自动化系教授.主要研究方向包括模式识别、计算机视觉、多媒体信息处理、信息服务科学与技术.