文章编号: 1001-0920 (2014) 04-0707-06 **DOI:** 10.13195/j.kzyjc.2012.1891

求解广义支持向量机的光滑型算法

倪 铁[1a], 李永利[2], 邵良杉[1b]

(1. 辽宁工程技术大学 a. 工商管理学院，b. 系统工程研究所，辽宁 葫芦岛 125105;
2. 中国中煤能源股份有限公司 鄂尔多斯分公司，内蒙古 鄂尔多斯 017200)

摘 要: 提出一种求解支持向量机 (SVMs) 的光滑型算法. 该算法基于其对偶优化模型的 KKT 系统, 提出一类新的光滑函数族, 将其 KKT 系统重构为一个光滑方程组, 并采用光滑型算法进行求解. 在适当的条件下, 该算法是全局收敛和局部超线性收敛的. 多个算例表明该算法非常有效, 具有广阔的应用前景.

关键词: 支持向量机；光滑函数；光滑算法；KKT 系统

中图分类号: TP181 **文献标志码:** A

Smoothing type algorithm for solving generalized support vector machines

NI Tie[1a], *LI Yong-li*[2], *SHAO Liang-shan*[1b]

(1a. College of Business Administration，1b. Institute of Systems Engineering，Liaoning Technical University，Huludao 125105，China；2. China Coal Energy Company Limited Ordos Branch，Ordos 017200，China．Correspondent：NI Tie，E-mail：nitie_4108@yahoo.com.cn)

Abstract: A smoothing-type algorithm for solving the generalized support vector machines(SVMs) is proposed. Based on the KKT system of the dual optimization model for SVMs, a new class of smoothing functions is proposed, and the KKT system is reformulated as a system of parameterized smooth equations and solved by using the smoothing type algorithm. Under some reasonable conditions, the proposed algorithm is shown to be globally convergent and locally superlinearly convergent. Numerical examples show that the proposed algorithm is promising and has a bright future of applications.

Key words: support vector machine；smoothing function；smoothing algorithm；KKT system

0 引 言

支持向量机 (SVMs) 是在 Cort 和 Vapnik[1-2] 提出的统计学习理论基础上发展而来的, 是一种新的机器学习方法. 该方法一经提出, 就受到了国内外学者的广泛关注, 并做了大量的研究工作. 由于其具有很好的泛化和推广能力, 能够较好地解决模型选择与欠学习、过学习以及非线性问题, 避免局部最优解, 有效地克服了“维数灾难”, 已成功地应用于人脸识别、模式分类、回归分析和手写字体识别等诸多领域.

设计一个有效的算法求解支持向量机模型一直是备受关注的问题之一. 国内外专家学者在支持向量机模型的算法研究方面做了大量的工作, 并取得了很好的结果, 例如: 内点算法[3-4]、半光滑算法[5]、光滑算法[6]、积极集算法[7]等.

光滑型算法已成功地用来求解很多优化问题[8-13], 其计算效果非常好. 这类算法与经典的路径跟踪内点方法有着密切的关系. 两种算法都属于一大类迭代算法, 迭代思想完全一致. 但光滑型算法又不同于内点方法. 二者之间显著的区别在于: 内点算法的初始点必须为可行内点, 迭代过程的中间迭代点也必须为可行内点, 这些要求给实际计算带来不小的挑战; 然而, 光滑型算法的初始点可以任意选取, 并且不要求中间的迭代点为内点, 而且光滑型算法在每一次迭代时只需求解一个线性方程组 (称之为一步光滑型算法). 因此, 与内点算法相比, 光滑型算法更具有通用性、适应性, 更便于数值计算.

本文提出一种求解支持向量机模型的光滑型算法. 该算法与已有的光滑型算法的重要差别在于: 本

收稿日期: 2012-12-20；**修回日期:** 2013-07-03.

基金项目: 国家自然科学基金项目(70971059).

作者简介: 倪铁(1980−), 男, 讲师, 博士, 从事最优化理论及其算法应用的研究；李永利(1962−), 男, 硕士, 从事煤矿发展战略、瓦斯灾害预警模型的研究.

文设计了一个新的带有双参数的光滑函数族, 这类光滑函数包含若干已有的光滑函数, 并且拥有已有光滑函数具备的性质; 另外, 在步长的获取方面, 采用了非单调线搜索技术, 有别于传统的单调线搜索. 在后面的数值计算部分发现, 采用非单调线搜索技术的光滑型算法要远远好于带有单调线搜索的光滑型算法.

1 支持向量机模型

1.1 线性支持向量机模型

考虑二分类问题. $m \times n$ 矩阵 A 的行向量 $A_i\ (i = 1, 2, \cdots, m)$ 表示 $\boldsymbol{R}^n$ 空间的 m 个点, $m \times m$ 对角阵 $D = \mathrm{diag}(d_1, d_2, \cdots, d_m)$, 其中 $d_i = +1$ 或 -1. 这样, 对于训练样本为 $\{(A_i, d_i)\}_{i=1}^m \subset \boldsymbol{R}^n \times \{+1, -1\}$ 的二分类问题, 由统计学理论, 经典的软间隔支持向量机模型如下:

$$\begin{aligned}&\min_{w,\gamma,\xi}\ \frac{1}{2}\|w\|^2 + \frac{\upsilon}{2}\|\xi\|^2,\\ &\text{s.t. }\ D(Aw - \gamma e) + \xi \geqslant e,\end{aligned} \tag{1}$$

其中 $e \in R^m$ 是全 1 向量. 容易看出, 模型 (1) 是关于变量 (w, γ, ξ) 的凸规划.

由文献 [14-15] 易知, 模型 (1) 的对偶问题为

$$\begin{aligned}&\min_x\ \frac{1}{2}x^{\mathrm{T}}DAA^{\mathrm{T}}Dx + \frac{1}{2\upsilon}x^{\mathrm{T}}x - e^{\mathrm{T}}x;\\ &\text{s.t. }\ e^{\mathrm{T}}Dx = 0,\ x \geqslant 0.\end{aligned} \tag{2}$$

不难看出, 模型 (2) 是关于变量 x 的严格凸规划, 其中 $\upsilon > 0$.

1.2 非线性支持向量机模型

通过选择任意的核函数 $K(A, A^{\mathrm{T}})$, 文献 [16, 6] 建立了下面的非线性支持向量机模型:

$$\begin{aligned}&\min_{u,\gamma,\xi}\ f(u) + \frac{\upsilon}{2}\|\xi\|^2,\\ &\text{s.t. }\ D(K(A, A^{\mathrm{T}})Du - \gamma e) + \xi \geqslant e,\end{aligned} \tag{3}$$

其中 f 是 $\boldsymbol{R}^n$ 上的某一凸函数. 相应地, 分离曲面为

$$K(x^{\mathrm{T}}, A^{\mathrm{T}})Du = \gamma. \tag{4}$$

这样, 模型 (2) 可以推广为

$$\begin{aligned}&\min_x\ \frac{1}{2}x^{\mathrm{T}}DK(A, A^{\mathrm{T}})Dx + \frac{1}{2\upsilon}x^{\mathrm{T}}x - e^{\mathrm{T}}x;\\ &\text{s.t. }\ e^{\mathrm{T}}Dx = 0,\ x \geqslant 0.\end{aligned} \tag{5}$$

事实上, 如果令 $K(x^{\mathrm{T}}, A^{\mathrm{T}}) = x^{\mathrm{T}}A^{\mathrm{T}}$, $w = A^{\mathrm{T}}Du$, 则式 (4) 退化为 $w^{\mathrm{T}}x = \gamma$. 而且, 若令 $f(u) = \frac{1}{2}u^{\mathrm{T}}DAA^{\mathrm{T}} \times Du$, $K(A, A^{\mathrm{T}}) = AA^{\mathrm{T}}$, 则模型 (3) 变为模型 (1), 模型 (5) 变为模型 (2).

2 光滑重构

下面设计求解模型 (5) 的光滑型算法.

因为模型 (5) 是一严格凸规划, 所以由约束最优化模型的一阶必要条件可得

$$\begin{aligned}&x \geqslant 0,\\ &\Big(\frac{1}{\upsilon}I + DK(A, A^{\mathrm{T}})D\Big)x - De\gamma - e \geqslant 0,\\ &x^{\mathrm{T}}\Big(\Big(\frac{1}{\upsilon}I + DK(A, A^{\mathrm{T}})D\Big)x - De\gamma - e\Big) = 0,\\ &e^{\mathrm{T}}Dx = 0.\end{aligned} \tag{6}$$

式 (6) 即为模型 (5) 的 Karush-Kuhn-Tucker(KKT)条件. 令

$$y = \Big(\frac{1}{\upsilon}I + DK(A, A^{\mathrm{T}})D\Big)x - De\gamma - e,$$

则系统 (6) 可简化为

$$x \geqslant 0,\ y \geqslant 0,\ x^{\mathrm{T}}y = 0,\ e^{\mathrm{T}}Dx = 0. \tag{7}$$

由于系统 (6) 中含有互补方程, 对其求解变得十分困难. 由文献 [17] 可知, 互补方程等价于 CP- 函数. 对于 CP- 函数, 有如下定义.

定义 1 函数 $\phi : \boldsymbol{R}^2 \to \boldsymbol{R}$ 称为 CP- 函数, 如果 $\phi(a, b) = 0 \Leftrightarrow ab = 0,\ a \geqslant 0,\ b \geqslant 0$.

常用的 CP- 函数有

$$\begin{aligned}&\varphi_{FB}(a, b) = \sqrt{a^2 + b^2} - a - b,\\ &\phi_{\min}(a, b) = \min\{a, b\}.\end{aligned} \tag{8}$$

由式 (8) 可知, 求解系统 (6) 可重构为求解下面的非光滑方程组:

$$F(x, y, \gamma) = \begin{bmatrix} y - \Big(\frac{1}{\upsilon}I + DK(A, A^{\mathrm{T}})D\Big)x + De\gamma + e \\ \varPhi(x, y) \\ e^{\mathrm{T}}Dx \end{bmatrix} = 0, \tag{9}$$

其中 $\varPhi(x, y) = (\varphi_{FB/\min}(x_1, y_1), \cdots, \varphi_{FB/\min}(x_m, y_m))^{\mathrm{T}}$. 因为函数 F 不可微, 所以某些经典的迭代算法 (如牛顿型算法) 不能用来求解非光滑方程组 (9). 为了克服这一缺陷, 首要的任务是将函数 F 光滑化. 进一步, 将 CP- 函数 $\varPhi(x, y)$ 光滑化.

光滑型算法中的一个关键环节是如何构造一个性态良好的光滑函数. 一般而言, 光滑函数是依赖于某个 CP- 函数的具体形式而构造的. 根据函数 $\min(\cdot, \cdot)$ 的结构, 对于任意的 $(a, b) \in \boldsymbol{R}^2$, 引入参数 $\mu \in \boldsymbol{R}$, 本文提出一种新的对称双扰动的带有双参数的光滑函数族, 即

$$\begin{aligned}&\varphi_{\kappa p}(\mu, a, b) =\\ &[\mu a + (1 + \kappa\mu)b] + [(1 + \kappa\mu)a + \mu b] -\\ &\sqrt[p]{(1 + \kappa\mu - \mu)^p|a - b|^p + 4\mu^p}.\end{aligned} \tag{10}$$

其中: κ 是任意的有限正实数, $p \in [2, +\infty)$.

下面的引理给出了光滑函数族 $\varphi_{\kappa p}$ 的基本性质.

引理 1 假设 $\kappa \in \boldsymbol{R}_{++}$ 是一任意有限参数, $p \in [2, +\infty)$ 是一任意且固定的参数, 光滑函数族 $\varphi_{\kappa p}$ 由

式 (10) 定义, 则有:

1) 对于任意的 $(\mu,a,b)\in\boldsymbol{R}^3$, 当 $\mu\neq 0$ 时, 函数 $\varphi_{\kappa p}$ 连续可微;

2) $\varphi_{\kappa p}(0,a,b)=0$ 当且仅当 $a\geqslant 0, b\geqslant 0, ab=0$.

对于任意 $(\mu,a,b)\in\boldsymbol{R}^3$ 且 $\mu\neq 0$, 函数 $\varphi_{\kappa p}$ 是一类非常广的光滑函数族, 它包含以下特殊的光滑函数:

1) 当 $\kappa=0, p=2$ 时, 光滑函数族 (10) 变为

$$\varphi_{02}(\mu,a,b)=$$
$$(1+\mu)(a+b)-\sqrt{(1-\mu)^2(a-b)^2+4\mu^2}.$$

这是 Huang 等[8]提出的一个光滑函数, 它是 CHKS 光滑函数[18-20]的一种正则化形式; Huang 等[9]将其拓展为欧氏若当代数意义下的光滑函数, 用于求解对称锥上的优化问题.

2) 当 $\kappa=1, p=2$ 时, 光滑函数族 (10) 变为

$$\varphi_{12}(\mu,a,b)=$$
$$[\mu a+(1+\mu)b]+[(1+\mu)a+\mu b]-$$
$$\sqrt{(a-b)^2+4\mu^2}.$$

这类光滑函数在文献 [21] 中用到.

3) 当 $p=2$ 时, 光滑函数族 (10) 变为

$$\varphi_{\kappa 2}(\mu,a,b)=$$
$$[\mu a+(1+\kappa\mu)b]+[(1+\kappa\mu)a+\mu b]-$$
$$\sqrt{(1+\kappa\mu-\mu)^2(a-b)^2+4\mu^2}.$$

这是 Ni 等[22]提出的一类光滑函数族.

对于任意的 $z:=(\mu,x,y,\gamma)\in\boldsymbol{R}^{2m+2}$, 其中 “:=” 表示 “被定义为”. 令

$$G(z):=\begin{bmatrix}\mu\\ F(z)\end{bmatrix}. \tag{11}$$

其中

$$F(z)=\begin{bmatrix} y-\Big(\dfrac{1}{v}I+DK(A,A^{\mathrm T})D\Big)x+De\gamma+e\\ \varPhi(x,y,\mu)\\ e^{\mathrm T}Dx\end{bmatrix}, \tag{12}$$

$$\varPhi(x,y,\mu):=\begin{bmatrix}\varphi_{\kappa p}(\mu,x_1,y_1)\\ \vdots\\ \varphi_{\kappa p}(\mu,x_n,y_n)\end{bmatrix}. \tag{13}$$

不难得出 $\varphi_{\kappa p}(0,a,b)=0\Leftrightarrow a\geqslant 0, b\geqslant 0, ab=0$. 这样, 由式 (11)～(13) 可得

$$G(z)=0\ \Leftrightarrow\ \mu=0 \text{ 和 } (x,\gamma) \text{ 是系统 (6) 的解}. \tag{14}$$

因此, 本文提出的光滑型算法是通过求解光滑方程组 $G(z)=0$ 逼近 KKT 系统 (6). 显然, 当 $\mu\downarrow 0$ 时, 式 (11) 的解趋近于 KKT 系统 (6) 的解, 即用一个光滑函数 $G(\mu,w)=0$ 近似代替 $G(0,w)=0$, 然后用牛顿型方法求解 $G(z)=0$, 进而求得 KKT 系统 (6) 的近似解.

3 算法设计与可行性分析

3.1 算法设计

对于任意的 $z=(\mu,x,y,\gamma)\in\boldsymbol{R}^{2m+2}$, 定义

$$\varPsi(z):=\|G(z)\|^2.$$

算法 1 基于非单调线搜索的光滑型算法.

Step 0 选取 $\delta\in(0,1)$, $\sigma\in(0,1/2)$, 以及任意的非负实数 κ. 选取 $\mu_0=\bar\mu>0$ 和 $(x^0,y^0,\gamma_0)\in\boldsymbol{R}^{2m+1}$. 置 $z^0:=(\mu_0,x^0,y^0,\gamma_0)$. 选取 $\vartheta\in(0,1)$ 使得 $\vartheta\bar\mu<1$. 置 $e_0:=(1,0,\cdots,0)\in\boldsymbol{R}^{2m+2}$, $D_0:=\varPsi(z^0)$, 以及 $P_0:=1$. 令 $\eta_{\min}$ 和 $\eta_{\max}$ 是两个常数且 $0\leqslant\eta_{\min}<\eta_{\max}<1$. 选取 $\eta_0\in[\eta_{\min},\eta_{\max}]$. 记 $\beta(z^0):=\vartheta\min\{1,\varPsi(z^0)\}$, $k:=0$.

Step 1 如果 $\|G(z^k)\|=0$, 则终止算法.

Step 2 由

$$G(z^k)+G'(z^k)\Delta z^k=\mu_0\beta(z^k)e_0, \tag{15}$$

计算 $\Delta z^k:=(\Delta\mu_k,\Delta x^k,\Delta y^k,\Delta\gamma_k)\in\boldsymbol{R}^{2m+2}$. 其中

$$G'(z^k)=\begin{bmatrix}1 & 0_{1\times m} & 0_{1\times m} & 0\\ 0_{m\times 1} & M_1 & E & De\\ M_2 & M_3 & M_4 & 0_{m\times 1}\\ 0 & e^{\mathrm T}D & 0_{1\times m} & 0\end{bmatrix},$$

E 为 $m\times m$ 单位矩阵, 而

$$M_1:=-\frac{1}{v}E-DK(A,A^{\mathrm T})D,$$
$$M_2:=(\theta+1)(x^k+y^k)-d_\mu,$$
$$M_3:=(1+(\kappa+1)\mu_k)E-D_x,$$
$$M_4:=(1+(\kappa+1)\mu_k)E-D_y.$$

这里

$$d_\mu:=\mathrm{vec}\{(d_\mu)_i\},$$
$$(d_\mu)_i:=$$
$$\frac{(\kappa-1)(1+\kappa\mu-\mu)^{p-1}|x_i-y_i|^p+4p\mu^{p-1}}{(\sqrt[p]{(1+\kappa\mu-\mu)^p|x_i-y_i|^p+4p\mu^p})^{p-1}};$$
$$D_x:=\mathrm{diag}\{(D_x)_i\},$$
$$(D_x)_i:=$$
$$\frac{\mathrm{sgn}(x_i-y_i)(1+\kappa\mu-\mu)^p|x_i-y_i|^{p-1}}{(\sqrt[p]{(1+\kappa\mu-\mu)^p|x_i-y_i|^p+4p\mu^p})^{p-1}};$$
$$(D_y)_i:=$$
$$\frac{-\mathrm{sgn}(x_i-y_i)(1+\kappa\mu-\mu)^p|x_i-y_i|^{p-1}}{(\sqrt[p]{(1+\kappa\mu-\mu)^p|x_i-y_i|^p+4p\mu^p})^{p-1}}.$$

Step 3 令 ε_k 是 $1,\delta,\delta^2,\cdots$ 中使得下式成立的最大值:

$$\varPsi(z^k+\varepsilon_k\Delta z^k)\leqslant[1-2\sigma(1-\vartheta\mu_0)\varepsilon_k]D_k. \tag{16}$$

Step 4 记 $z^{k+1}=z^k+\alpha_k\Delta z^k$, $k:=k+1$, 且

$$\beta(z^k):=\min\{\gamma,\gamma\varPsi(z^k),\beta(z^{k-1})\}.$$

Step 5　选取 $\eta_k \in [\eta_{\min}, \eta_{\max}]$, 记

$$P_k := \eta_{k-1}Q_{k-1} + 1,$$

$$D_k := \frac{\eta_{k-1}P_{K-1}D_{k-1} + \Psi(z^k)}{P_k}.$$

返回 Step 1.

3.2　算法分析

光滑型算法具有以下优良特性:

1) 与文献 [3-4] 提出的内点法相比, 算法 1 可以起始于任意一点, 避免了内点法等优化算法对初始点敏感的问题.

2) 在每步迭代中算法 1 只需求解一个线性方程组.

3) 算法 1 不依赖于中心路径的邻域, 因而不需要为保持迭代点位于中心路径的某个邻域而进行必要的计算.

4) 线搜索 (16) 是一个非单调线搜索. 这种非单调线搜索技术是由 Zhang 等[23]提出的. 如果对于所有的 k, 有 $\eta_k = 0$ 成立, 则 $D_K = \Psi(z^k)$, 线搜索 (16) 归结为单调 Armijo 线搜索. 这是牛顿型算法中运用较为广泛的搜索方法. 在这种情况下, 类似的算法框架已得到广泛的研究[8-13, 21]. 从后面的数值实验过程可以发现, 算法 1 的非单调版本比算法 1 的单调版本更加有效, 因此在数值实验中略去了单调算法的数值结果. 另外, 每一个被测试的问题都可以在迭代步数非常少的情况下成功求解, 计算速度也非常快.

5) 算法 1 是适定的, 即牛顿方程 (15) 是可解的, 线搜索 (16) 是适定的.

6) 在一定的条件下, 可以证明该算法是全局线收敛和局部超线性收敛.

4　数值实验

这里所有的实验都是在配置为 CPU 2.99 GHz, RAM 1.96 GB 的计算机上完成的, 并且所有代码都是用 Matlab 编写的.

在整个数值实验中, 算法 1 中初始点的选择方式如下:

$$x^0 = \mathrm{randn}(m, 1) \in \boldsymbol{R}^m,\ y^0 = \mathrm{randn}(m, 1) \in \boldsymbol{R}^m,$$
$$\gamma^0 = \mathrm{randn}(1) \in \boldsymbol{R},\ \mu_0 = 1.0.$$

算法 1 中参数为 $\bar{\mu} = 1.0, \vartheta = 0.5, \delta = 0.5, \sigma = 0.25$. 参数 υ, κ 的选取在表 1 中给定. 终止规则为 $\|G(z^k)\| \leqslant 10^{-6}$, 其中函数 G 由式 (11) 定义. 选取数据库 Irvine machine learning repository[24]和 University of California at Irvine (UCI) 中的测试问题对算法 1 进行测试, 相关数值计算结果见表 1 ～ 表 5. 在测试结果表中, 问题的名称与文献 [24] 中的问题名称完全一致. 其中: Dataset size 表示数据库中相应测试问题名称和问题规模; IT 表示算法迭代的次数; $\|G\|$ 表示算法终止时 $\|G(z^k)\|$ 的值; CPU 表示算法的迭代时间, 单位: s.

不难发现, 模型 (2) 的解与 KKT 系统 (6) 的解满足如下关系:

$$\begin{cases} w = A^{\mathrm{T}}Dx, \\ \xi = x/\upsilon, \\ b = \text{约束方程 } e^{\mathrm{T}}Dx = \\ \qquad 0 \text{ 对应的乘子向量}. \end{cases} \tag{17}$$

由该式可得模型 (1) 的分类超平面, 并且测试其分类的准确性. 在测试结果表中, Training correctness 表示分类超平面对训练数据分类的正确性; Testing correctness 表示分类超平面对测试数据分类的正确性. 在这里, 训练数据选取原始数据集的 90%, 其余 10% 的数据用于检测, 并且用 Training correctness 来判断测试的性能. 表中的每一个结果都是经过 10 次测试的平均值.

由数值计算结果可知, 算法 1 显示出良好的效率、准确性和灵活性. 首先, 比较表 1、表 2 和表 3 的计算结果不难看出, 当算法 1 和原问题取不同的参数时, 测试不同问题的准确性变化很大. 由这种变化得到一个启示: 为了得到更好的测试结果准确性, 对于不同的测试问题, 可以选择不同的参数, 以得到更好的测试结果. 同时注意到, 对于某些问题 (例如:

表 1　基于线性核函数的计算结果

Dataset size $m \times n$	(υ, k, p)	Training correctness/ %	Testing correctness/ %	$\|G\|$	IT	CPU
Ionosphere (351×34)	(100, 0.5, 2)	85.06	83.10	5.214 0e-09	13	1.017 9e+00
Pima Indians (768×8)	(1, 0 ,2)	64.40	71.43	4.871 1e-09	8	6.190 1e+01
BUPA Liver (345×6)	(1, 0, 2.5)	57.42	62.86	2.109 1e-07	10	9.592 5e-01
Hepatitis (80×20)	(1, 1, 2.5)	81.25	93.75	6.621 9e-07	11	4.448 7e-02
Leukemia (72×7 129)	(100, 1, 20)	68.42	73.33	3.300 5e-09	7	3.357 8e-02
Postoperative (87×9)	(100, 0, 2.5)	71.01	77.78	1.093 5e-09	10	8.788 3e-02

表 2 基于高斯非线性核函数的计算结果

Dataset size $m \times n$	(v, k, p, λ)	Training correctness/ %	Testing correctness %	$\|G\|$	IT	CPU
Ionosphere (351×34)	(100, 0.5, 2, 0.1)	85.21	80.00	6.900 9e-08	9	8.883 2e-01
Pima Indians (768×8)	(1, 0, 2, 0.1)	64.40	71.43	4.871 1e-09	8	6.190 1e+01
BUPA Liver (345×6)	(1, 0, 2.5, 0.1)	57.74	60.00	2.120 2e-12	8	1.890 4e+00
Hepatitis (80×20)	(1, 0.5, 2.5, 0.000 01)	83.33	87.50	1.366 3e-08	8	4.642 3e-02
Leukemia (72×7 129)	(10, 0, 20, 0.001)	64.06	75.00	5.732 4e-10	7	5.667 4e-02
Postoperative (87×9)	(1, 0, 2.5, 0.1)	71.79	77.78	1.111 5e-07	7	4.613 8e-02

表 3 基于多项式非线性核函数的计算结果

Dataset size $m \times n$	(v, k, p, d)	Training correctness/ %	Testing correctness/ %	$\|G\|$	IT	CPU
Ionosphere (351×34)	(1, 1.5, 2, 1)	70.16	75.00	7.892 7e-07	13	1.461 2e+00
Pima Indians (768×8	(1, 0, 2, 1)	64.40	71.43	2.022 0e-08	10	8.267 7e+00)
BUPA Liver (345×6)	(1, 0, 2.5, 1)	57.10	65.71	2.573 6e-09	8	7.815 8e+01
Hepatitis (80×20)	(1, 1.5, 2, 1)	81.94	100	1.088 7e-10	12	5.776 5e-02
Leukemia (72×7 129)	(10, 0, 20, 1)	64.06	75.00	5.732 4e-10	7	5.667 4e-02
Postoperative (87×9)	(1, 0, 2.5, 1)	71.79	77.78	1.111 5e-07	7	4.613 8e-02

表 4 基于线性核函数的计算结果

Dataset size (Training, Testing)	(v, k, p)	Training correctness/ %	Testing correctness/ %	$\|G\|$	IT	CPU
(1 605, 30 957)	(1, 0.5, 2)	75.59	75.93	1.556 790e-07	11	12.79
(2 265, 30 297)	(1, 0.5, 2)	77.13	75.82	9.643 350e-08	11	29.14
(3 185, 29 377)	(1, 0.5, 2)	75.0	76.01	6.409 713e-07	11	64.93
(4 781, 27 781)	(1, 0.5, 2)	75.57	75.98	2.628 788e-07	12	210.88
(6 414, 26 148)	(1, 0.5, 2)	76.19	75.85	4.187 640e-07	11	448.69
(8 140, 24 422)	(1, 0.5, 2)	75.82	75.95	6.378 069e-07	11	878.32

表 5 基于多项式非线性核函数的计算结果

Dataset size (Training, Testing)	(v, k, p, d)	Training correctness/ %	Testing correctness/ %	$\|G\|$	IT	CPU
(1 605, 30 957)	(1, 0.5, 2, 1)	75.32	75.95	4.636 330e-08	11	11.22
(2 265, 30 297)	(1, 0.5, 2, 1)	76.25	75.89	1.056 170e-07	13	30.42
(3 185, 29 377)	(1, 0.5, 2, 1)	74.69	76.05	1.069 380e-07	13	75.38
(4 781, 27 781)	(1, 0.5, 2, 1)	75.19	76.04	2.945 196e-07	12	207.06
(6 414, 26 148)	(1, 0.5, 2, 1)	75.29	76.07	4.341 544e-07	11	448.55
(8 140, 24 422)	(1, 0.5, 2, 1)	75.82	75.95	6.377 990e-07	12	1 003.05

Ionosphere、PimaIndians、Hepatitis), 算法 1 计算的数值效果是最好的. 通过观察表 4 和表 5 不难发现, University of California at Irvine (UCI) 中 Adult 问题的规模比 Irvine machine learning repository 中的问题大很多, 而且, 算法 1 在求解规模较大的问题上也是很有效的. 这些数值结果表明了本文提出的一类新光滑函数族的重要性和有效性. 对于同样的测试问题, 通过选择适当的光滑函数可能会得到更好的数值计算结果. 而且, 当选择不同的核函数时, 即高斯核函数

$$K(A_i, A_j) = \exp(-\lambda\|A_i - A_j\|^2),$$
$$i, j = 1, 2, \cdots, m,$$

多项式核函数

$$K(A_i, A_j) = (A_i * A_j^{\mathrm{T}} + 1)^d,\ i, j = 1, 2, \cdots, m,$$

求解不同测试问题的数值结果可能会得到进一步改进, 具体计算结果见表 2 和表 3. 这种现象也表明了算法 1 在计算时对各种核函数的通用性和有效性. 因此, 在选择其他核函数进行求解时, 算法 1 可能会更好.

5 结 论

本文提出了一类新的光滑函数族,它包含了文献中 3 个具体的光滑函数并将其作为特殊情况. 针对支持向量机 (SVMs) 的对偶优化模型, 通过使用这类光滑函数族将对偶优化模型的 KKT 系统重构为一光滑方程组, 提出了一个带有非单调线搜索技术的光滑型算法进行求解, 从而找到支持向量机的解. 最后对所提出的光滑型算法进行了数值计算, 所得结果表明了该算法的有效性.

参考文献(References)

[1] Cort C, Vapnik V N. Support vector networks[J]. Machine Learning, 1995, 20(3): 273-297.

[2] Vapnik V N. Statistical learning theory[M]. New York: John Wiley and Sons, 1998: 1-490.

[3] Ferris M C, Munson T S. Interior point methods for massive support vector machines[J]. SIAM J on Optimization, 2003, 13(3): 783-804.

[4] Woodsend K. Using interior point methods for largescale support vector machine training[D]. Scotland: School of Mathematics, University of Edinburgh, 2009.

[5] Ferris M C, Munson T S. Semismooth support vector machines[J]. Mathematical Programming, 2004, 101(1): 185-204.

[6] Lee Y J, Mangasarian O L. SSVM: A smooth support vector machine for classification[J]. Computational Optimization and Applications, 2001, 20(1): 5-22.

[7] Mangasarian O L, Musicant D R. Active set support vector machine classification[C]. Advances in Neural Information Processing Systems 13. Vancouver: MIT Press, 2001: 577-583.

[8] Huang Z H, Han J, Chen Z. A predictor-correor smoothing Newton algorithm, based on a new smoothing function, for solving the nonlinear complementarity problem with a P_0 function[J]. J of Optimization Theory and Applications, 2003, 117(1): 39-68.

[9] Huang Z H, Ni T. Smoothing algorithms for complementarity problems over symmetric cones[J]. Computational Optimization and Applications, 2010, 45(3): 557-579.

[10] Huang Z H, Qi L, Sun D. Sub-quadratic convergence of a smoothing Newton algorithm for the P_0 and monotone LCP[J]. Mathematical Programming, 2004, 99(3): 423-441.

[11] Huang Z H, Sun D, Zhao G Y. A smoothing Newton-type algorithm of stronger convergence for the quadratically constrained convex quadratic programming[J]. Computationa Optimization and Applications, 2006, 35(23): 199-237.

[12] Huang Z H, Xu S W. Convergence properties of a noninterior-point smoothing algorithm for the P_* NCP[J]. J of Industrial and Management Optimization, 2007, 3(3): 569-584.

[13] Huang Z H, Zhang Y, Wu W. A smoothing-type algorithm for solving system of inequalities[J]. J of Computational and Applied Mathematics, 2008, 220(1/2): 355-363.

[14] Wolfe P. A duality theorem for nonlinear programming[J]. Quarterly of Applied Mathematics, 1961, 19(3): 239-244.

[15] Mangasarian O L. Nonlinear programming[M]. New York: McGraw-Hill, 1969: 1-220.

[16] Mangasarian O L. Generalized support vector machines[C]. Advances in Large Margin Classifiers. Cambridge: MIT Press, 2000: 135-146.

[17] Facchinei F, Pang J S. Finite-dimensional ariationnal inequalities and complementarity problems[M]. New York: Springer Verlag, 2003: 1-728.

[18] Chen B, Harket P T. A non-interior-point continuation method for linear complementarity problem[J]. SIAM J on Matrix Analysic and Applications, 1993, 14(4): 1168-1190.

[19] Kanzow C. Some noninterior continuation methods for linear complementarity problems[J]. SIAM J on Matrix Analysis and Applications, 1996, 17(4): 851-868.

[20] Smale S. Algorithms for solving equations[C]. Proc of Int Congress of Mathematicians. Providence: American Mathematics Society, 1987: 172-195.

[21] Ni T, Liu X H, Gu W Z. Convergence of a smoothing Newton algorithm for linear programming over symmetric cones[R]. Tianjin: School of Science, Tianjin University, 2009.

[22] Ni T, Gu W Z. Smoothing newton algorithm for symmetric cone comlpementarity problems based on a one-parametric class of smoothing functions[J]. J of Applied Mathematics and Computing, 2011, 35(1/2): 73-92.

[23] Zhang H C, Hager W W. A nonmonotone line search technique and its application to unconstrained optimization[J]. SIAM J on Optimization, 2004, 14(4): 1043-1056.

[24] Murphy P M, Aha D W. UCI repository of machine learning databases[EB/OL]. [2013-07-01]. www.ics.uci.edu/∼mlearn/MLRepository.html.

(责任编辑: 李君玲)