最小 VC维分类器的一种实现方法

魏慧荣, 许建华

(南京师范大学 数学与计算机科学学院, 江苏 南京 210097)

[摘要] 通常在支持向量机算法中核函数参数是事先选定好的, 而最小 VC维分类器的非线性约束规划问题中包含 RBF核的参数, 在算法执行中可以自适应地确定. 综合复形调优法、罚函数法及梯度法, 提出了一种最小 VC维分类器的实现方法. 该实现方法在保证分类器有较高分类性能的前提下, 可以以较快的速度处理较大的样本集. 针对 4个基准数据集, 将该方法与 SVM^{light}算法进行了比较, 试验结果表明该最小 VC维分类器的实现方法在分类精度与计算时间上都有一定的优势.

[关键词] 支持向量机, 核函数参数, 分解算法, 复形调优法

[中图分类号] O235　[文献标识码] A　[文章编号] 1672-1292(2008)01-0075-05

An Implementation Method for Minimal VC Dimensional Classifier

Wei Huirong　Xu Jianhua

(School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China)

Abstract In support of vector machine the parameters of kernel function have to be adjusted in advance. However, the constrained nonlinear programming of minimal VC dimensional classifier involves the parameter of RBF kernel, which could be determined adaptively. In this paper, a fast implementation method based on the complex optimization method, penalty function method and gradient descent method is designed to solve such a nonlinear problem. The method has not only good performance of classification, but a high speed to deal with large data sets. The experimental results on four benchmark data sets demonstrate that our algorithm runs faster and obtains higher precision than the famous SVM^{light} algorithm does.

Key words support vector machines, kernel functional parameter, decompositional method, complex optimization method

在统计学习理论[1-3]中, 评价分类器性能的好坏是通过推广能力的界来体现的, 推广能力的界由经验风险和置信范围两部分组成, 这两部分都依赖于 VC维. VC维越大, 经验风险越小, 置信范围越大. 因此, 在设计分类器时要保证推广能力的界最小, 这也正是结构风险最小化原则的基本思想.

在模式分类中, Δ间隔超平面的 VC维为 $\min([R^2/\Delta^2], n)+1$[1], 其中 R为能够包围所有训练样本的球体中最小球体的半径, Δ为超平面到离它最近样本点的间距, n为空间维数. 若核函数及其参数事先确定, 对于给定的训练集, R是确定的. 在硬间隔支持向量机(SVM)中, 经验风险为零, 间隔 $\Delta=\frac{1}{\|\boldsymbol{w}\|_2}$, 其中 $\boldsymbol{w}$是训练集的权向量. 要保证推广能力的界最小, 相当于保证 $\|\boldsymbol{w}\|_2^2$最小. 若核函数及其参数不确定, 则 R不确定, 在经验风险为零的情况下, 保证分类器推广能力的界最小, 就是要最小化 $R^2\|\boldsymbol{w}\|^2$.

在 SVM中, 核参数可以事先通过 k重交叉验证法[4,5] 结合参数网格扫描法、半径间隔界法[6,7] 等方法来确定. 在 k重交叉验证法中, 将 N个样本分为 k $(2\leqslant k\leqslant N)$ 个子集, 其中的 $(k-1)$ 个子集构成训练集, 剩余的一个子集用于测试, 重复 k次后计算出错误率或正确率. 通过网格扫描法确定算法的参数值, 利用 k重检验法计算错误率, 将最小错误率对应的参数作为最佳参数. 这是目前应用最广泛的技术, 其缺点是计

收稿日期: 2007-04-02
基金项目: 江苏省自然科学基金(BK2004142)资助项目.
作者简介: 魏慧荣(1979-), 女, 硕士研究生, 研究方向: 模式识别. E-mail: weihuirong@gmail.com
通讯联系人: 许建华(1962-), 高级工程师, 博士, 研究方向: 模式识别、机器学习等. E-mail: xujianhua@njnu.edu.cn

算量大. 在半径间隔法中, 将 $R^2\|\boldsymbol{w}\|^2$ 作为目标函数, 并通过改变核函数将软间隔 SVM 转化成硬间隔问题, 用梯度法来逐步搜索最佳参数值. 但是, 对于每一组参数值需要分别求解两个二次规划问题得到目标函数中的两个项, 所以同样存在计算量较大的缺点.

针对上述方法都有计算量过大的问题, 文[8]提出了最小 VC维分类器, 其非线性约束规划问题中包含 RBF核的参数, 所以最小 VC维分类器中核函数的参数不需要事先确定, 而是在求解分类器的过程中自动确定, 但是文[8]中的实验是基于 Matlab实现的, 因此计算速度也比较慢.

基于上述原因, 本文提出了最小 VC维分类器的快速实现方法. 该方法综合采用复形调优法[9]、罚函数法和梯度法[10-12]这几种优化算法, 其中罚函数法和梯度法用来确定初始可行解, 复形调优法用于求解非线性约束极小化问题. 为了说明该方法的性能, 针对 4组基准数据集, 本方法与 SVM$^{\text{light}}$ 进行了比较, 结果说明该方法在分类精度和计算速度上都有一定的优势.

1 最小 VC维分类器

本文针对文[8]给出的最小化 VC维分类器来寻找一种实现方法. 下面简要地给出最小 VC维分类器的推导过程:

现有两类训练样本集:

$$\{(x_1, y_1), \cdots, (x_i, y_i), \cdots, (x_l, y_l)\}, \text{其中 } x_i \in \mathbf{R}^n,\ y_i \in \{+1, -1\}.$$

在特征空间中, 包围所有样本的球体半径的平方为

$$R^2 = \min_a \max_{x_i}(k(x_i, x_i) + k(\boldsymbol{a}, \boldsymbol{a}) - 2k(x_i, \boldsymbol{a})). \tag{1}$$

式中, $\boldsymbol{a}$ 是原始空间中对应于特征空间中球中心的向量, $k(x_i, x_j)$ 是核函数. 在原始空间中,

$$R_0^2 = \min_{a_0}\max_{x_i}(\|x_i - \boldsymbol{a}_0\|_2^2). \tag{2}$$

式中 $\boldsymbol{a}_0$ 为原始空间中包含所有训练集的球中心向量, 若 $\boldsymbol{a}_0$ 固定, 则

$$R_0^2 \leqslant \max_{x_i}(\|x_i - \boldsymbol{a}_0\|_2^2) = d^2. \tag{3}$$

文[8]中定义 $\boldsymbol{a}_0 = (a_1, a_2, \cdots, a_n)^{\mathrm{T}}$, 其中

$$a_j = (\max_i(x_i^j) + \min_i(x_i^j))/2,\ i = 1, \cdots, l. \tag{4}$$

本文只考虑 RBF核函数, 即

$$k(x, y) = \mathrm{e}^{-\gamma\|x-y\|_2^2}, \quad \gamma \geqslant 0 \tag{5}$$

式中, γ 是 RBF核函数参数, 因此式(1)又可写成

$$R^2 = \min_a\max_{x_i}(2(1 - \mathrm{e}^{-\gamma\|x_i-\boldsymbol{a}\|_2^2})). \tag{6}$$

在此, 假设样本集的中心向量 $\boldsymbol{a}_0$ 为特征空间中球体的中心对应于原始空间的向量, 即 $\boldsymbol{a} = \boldsymbol{a}_0$, 则

$$R^2 \leqslant \max_{x_i}(2(1 - \mathrm{e}^{-\gamma\|x_i-\boldsymbol{a}_0\|_2^2})) = 2(1 - \mathrm{e}^{-\gamma\max_{x_i}\|x_i-\boldsymbol{a}_0\|_2^2}) = 2(1 - \mathrm{e}^{-\gamma d^2}). \tag{7}$$

在特征空间中, 权向量可以表示为

$$\boldsymbol{w} = \sum_{i=1}^{l}\alpha_i\Phi(x_i), \quad \alpha_i \in \mathbf{R},\ i = 1, 2, \cdots, l \tag{8}$$

式中, α_i 是样本 x_i 所对应的加权系数, $\Phi(x_i)$ 是输入空间中的样本 x_i 在特征空间中的像. 则

$$\|\boldsymbol{w}\|_2^2 = \sum_{i,j=1}^{l}\alpha_i\alpha_j\Phi(x_i)\cdot\Phi(x_j). \tag{9}$$

根据核函数定义 $k(x_i, x_j) = \Phi(x_i)\cdot\Phi(x_j)$, 有

$$\|\boldsymbol{w}\|_2^2 = \sum_{i,j=1}^{l}\alpha_i\alpha_j\mathrm{e}^{-\gamma\|x_i-x_j\|_2^2}. \tag{10}$$

综合式(7)和(10)得

$$R^2\|\boldsymbol{w}\|_2^2 \leqslant 2(1 - \mathrm{e}^{-\gamma d^2})\sum_{i,j=1}^{l}\alpha_i\alpha_j\mathrm{e}^{-\gamma\|x_i-x_j\|_2^2}. \tag{11}$$

从上面的推导可以得到, 可分情况下最小 VC维分类器的优化模型:

$$\min_{\alpha,\beta,\gamma} 2(1-e^{-\gamma d^2})\sum_{i,j=1}^{l}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2},$$

$$\text{s.t. } y_j\left(\sum_{i=1}^{l}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta\right)\geqslant 1,\quad \gamma\geqslant 0,\quad j=1,2,\cdots,l. \tag{12}$$

求解优化问题(12), 可以得到非线性分类器 $f(x)=\sum_{i=1}^{l}\alpha_i e^{-\gamma\|x_i-x\|_2^2}+\beta$中的参数 α, β和 γ. 我们称此分类器为最小 VC维分类器.

2 最小 VC维分类器的实现方法

当训练样本集较小时, 可用复形调优法结合罚函数法和梯度法直接解问题(12). 但当训练样本集较大时, 直接求解的计算时间会非常长. 因此, 为了能够处理较大样本集, 本文用分解算法来实现最小 VC维分类器. 这是受支持向量机分解算法[13-15]的启发, 将较大的问题分解成一系列小的子问题进行求解. 本文先将较大样本集分成 B和 N 两个集合, B 称为工作集, 其大小固定为 q. 然后仅对工作集 B 进行优化处理, 集合 N 中样本对应的变量暂时保持不变.

下面将问题(12)基于集合 B 和集合 N 表示为:

$$\begin{aligned}\min\ & 2(1-e^{-\gamma d^2})\left(\sum_{i\in B,j\in B}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2}+2\sum_{i\in B,j\in N}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2}+\sum_{i\in N,j\in N}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2}\right)\\ \text{s.t. } & y_j\left(\sum_{i\in B}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\sum_{i\in N}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta\right)\geqslant 1,\quad j\in B,\\ & y_j\left(\sum_{i\in B}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\sum_{i\in N}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta\right)\geqslant 1,\quad j\in N,\\ & \gamma\geqslant 0.\end{aligned} \tag{13}$$

由于每次的优化都是基于集合 B的, 因此, 在优化问题中仅留下和集合 B 相关的项, 即将目标函数中和集合 B 无关的项 $\sum_{i\in N,j\in N}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2}$, 及约束条件中对集合 N 中样本的约束项 $y_j(\sum_{i\in B}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\sum_{i\in N}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta)\geqslant 1$, $j\in N$ 去掉, 得到简化后的优化模型:

$$\begin{aligned}\min\ & 2(1-e^{-\gamma d^2})\left(\sum_{i\in B,j\in B}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2}+2\sum_{i\in B,j\in N}\alpha_i\alpha_j e^{-\gamma\|x_i-x_j\|_2^2}\right),\\ \text{s.t. } & y_j\left(\sum_{i\in B}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\sum_{i\in N}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta\right)\geqslant 1,\quad j\in B,\\ & \gamma\geqslant 0.\end{aligned} \tag{14}$$

由于该算法中核参数 γ待定, 因此, 简化后会失掉一些有用信息, 但为了处理较大数据集, 这样处理是有必要的. 而且试验结果证明这样做是可行的.

对子问题(14)进行求解, 主要有两步: 利用罚函数梯度法确定初始可行解, 据此再用复形调优法实现对子问题的优化. 求解的总体流程如下:

(1) 随机选择 q个样本放入工作集 B 中;

(2) 用 2.1, 2.2所述的求解子问题方法对 B进行优化处理;

(3) 当样本集中被错分的样本数占总样本数的比率大于某个给定的阀值 ρ时, 更新工作集 B: 将 N中被错分的一个样本取代 B 中任意一个被正确分类的样本, 重复(2). 否则, 算法结束.

2.1 基于罚函数法和梯度法确定初始可行解

复形调优法要求初始解必须是可行解. 本文使用罚函数法和梯度法来确定初始可行解的. 可行解就是找到一个向量 $(\alpha_1,\cdots\alpha_l,\beta,\gamma)$ 满足:

$$\begin{aligned}& y_j\left(\sum_{i=1}^{l}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta\right)-1\geqslant 0,\quad j=1,\cdots,l.\\ & \gamma\geqslant 0,\end{aligned} \tag{15}$$

令

$$g_j(\alpha,\beta,\gamma)=y_j\left(\sum_{i=1}^{l}\alpha_i e^{-\gamma\|x_i-x_j\|_2^2}+\beta\right)-1, \tag{16}$$

取

$$\Psi_j(\alpha,\beta,\gamma)=\mu_j g_j(\alpha,\beta,\gamma),\quad j=1,\cdots,l. \tag{17}$$

其中

$$\mu_j=\max_j\{0,-g_j(\alpha,\beta,\gamma)\},\quad j=1,\cdots,l. \tag{18}$$

将(16),(18)代入(17)得:

$$\phi_j(\alpha,\beta,\gamma)=\mu_j\Big[y_j\Big(\sum_{i=1}^{l}\alpha_i e^{-r\|x_i-x_j\|_2^2}\Big)+\beta-1\Big]. \tag{19}$$

因此,得到目标函数:

$$F(\alpha,\beta,\gamma)=\sum_{j=1}^{l}\Psi_j(\alpha,\beta,\gamma). \tag{20}$$

综合式(19)、(20),给出确定初始值的模型:

$$\min\sum_{j=1}^{l}\mu_j\Big[y_j\Big(\sum_{i=1}^{l}\alpha_i e^{-r\|x_i-x_j\|_2^2}+\beta\Big)-1\Big],\quad j=1,\cdots l.$$

$$\gamma\geqslant 0. \tag{21}$$

用梯度法[13]解(21)优化问题,得到初始值.

2.2　复形调优法解优化问题

复形调优法的基本思想是:在可行域上任取 $2m$ 个点作为顶点构成初始复形,其中 m 为参数个数,比较复形各顶点处的目标函数值,不断去掉复形的最高点,代之以目标函数值较小的点作为新的顶点构成新的复形,使顶点的目标函数值逐步下降,顶点逐步逼近约束问题的最优点.

在子问题(14)中有 $(l+2)$ 个参数需要确定,复形调优法优化子问题(14)的步骤为:

(1) 首先给定初始复形中的第一个顶点坐标:$x_{i=0}=(\alpha_{00},\alpha_{10},\cdots,\cdots,\alpha_{l0},\beta_{00},\gamma_{00})$,反射系数 $\mu=1.3$,精确度 $\varepsilon=1.0e-03$.该顶点坐标是用上述确定初始可行解的方法得到的.

(2) 用上述确定初始可行解的方法,在 $l+2$ 维变量空间中再确定出初始复形的其余 $2(l+2)-1$ 个顶点,并计算各个顶点处的目标函数值.

(3) 找出目标函数最大时对应的顶点 x_R(最坏点)和剩余 $2(l+2)-1$ 个顶点对应目标函数最大的顶点 x_G;

(4) 计算最坏点 x_R 的对称点:$x_T=(1+\mu)x_F-\mu x_R$,其中 $x_F=\dfrac{1}{2(l+2)-1}\sum\limits_{\substack{i=0\\ i\neq R}}^{2(l+2)-1}x_{(i)}$.

(5) 确定一个新的顶点替代最坏点 x_R 以构成新的复形:如果 $f_T>f_G$,修改 x_T:$x_T=\dfrac{(x_F+x_T)}{2}$,然后检查 x_T 是否满足所有约束条件,若不满足则用确定初始可行解的方法使 x_T 成为可行解,重复步骤(5),直到 $f_T\leqslant f_G$.此时令 $x_R=x_T$,$f_R=f_T$.

(6) 重复步骤(3)~(5),直到复形中各顶点距离小于预先给定的精度要求为止.

3　实验结果及分析

实验所用数据取自 http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets,数据的具体情况见表 1.实验所用硬件平台:Athlon 1GHz内存 256M,软件平台:MFC6.0.

SVM^{light}中 γ的最佳值是通过扫描确定的.γ取得最佳值的标准是测试正确率达到最高.针对于样本集 a1a.txt,a2a.txt,a3a.txt,γ的扫描区间设定为从 0.1~0.5,步长为 0.01,分别找到 γ 的最佳值为:0.25,0.26,0.35;对于样本集 splice.txt,γ的扫描区间设定为从 0.001~0.05,步长为 0.001,找到 γ的最佳值为

表 1　实验数据的概况

Table 1　Experiment data

数据集		特征数	样本数
训练集	a1a.txt	123	1 605
	a2a.txt	123	2 265
	a3a.txt	123	3 185
	splice.txt	60	1 000
测试集	a1a.t	123	30 956
	a2a.t	123	30 296
	a3a.t	123	29 376
	splice.t	60	2 175

0.036 SVM^{light}中参数 C 是所有样本到原点的平均核距离的平方的倒数(缺省值). 实验结果见表 2

在得到 γ的最佳值的同时, 也得到了 SVM^{light}对 4组数据的训练次数、训练总时间、训练正确率和测试正确率. 在 SVM^{light}中工作集大小取其默认值 10. 本文提出的最小 VC维分类器的实现方法 VCDC设定工作集大小为 6, 并对同样的基准数据集进行训练和测试并得到相应结果, 如表 3所示.

表 2 SVM^{light}中 γ值扫描参数

Table 2 Scan parameter γ in SVM^{light}

训练集	γ的扫描区间	扫描步长	γ的最佳值	C
a1a. txt	0.1~0.5	0.01	0.25	3.8632
a2a. txt	0.1~0.5	0.01	0.26	3.8634
a3a. txt	0.1~0.5	0.01	0.35	3.8641
splice. txt	0.001~0.05	0.001	0.036	0.5065

表 3 实验结果

Table 3 The result of experiment

训练集	工作集大小	γ值	训练次数	训练总时间 /s	训练正确率 /%	测试正确率 /%	分类器
a1a. txt	10	0.25	50	158	88.35	82.91	SVM^{light}
a2a. txt	10	0.26	50	310.5	87.46	82.98	
a3a. txt	10	0.35	50	683	88.04	82.47	
Splice. txt	10	0.036	50	131.5	99	81.89	
a1a. txt	6	0.65	1	47.5	93.58	81.65	VCDC
a2a. txt	6	1.24	1	28	97.88	81.08	
a3a. txt	6	0.55	1	82.3	89.23	82.57	
Splice. txt	6	0.024	1	22.2	87.70	83.59	

实验结果显示, 本文提出的最小 VC维分类器的快速实现方法在和 SVM^{light}算法有相当的训练和测试正确率的情况下, 所需的训练时间更短, 并同时得到了 γ的最佳值.

4 结论

通常在支持向量机算法中核函数参数必须事先通过 k重交叉验证法[4,5]结合参数网格扫描法、半径间隔界法[6,7]等方法来确定. 而这些方法都存在计算量较大的缺点. 文[8]提出了最小 VC分类器, 其 RBF核的参数可以在求解分类器的过程中自动确定, 但其实验是基于 Matlab实现的, 因此计算速度也比较慢. 由于上述原因, 本文综合了罚函数法、梯度法和复形调优法 3种优化算法, 提出了最小 VC维分类器的快速实现方法. 针对给定的 4组基准数据集, 该算法在分类精度上和计算时间上都较好于 SVM^{light}算法. 本文主要针对可分情况下的最小 VC维分类器进行处理, 不可分情况下的最小 VC维分类器将在下一步的工作中处理. 本文仅基于 4组基准数据进行测试来验证 VCDC算法的有效性, 还需测试更多的基准数据来验证 VCDC算法的推广能力.

[参考文献] (References)

[1] Vapnik V N. 统计学习理论的本质[M]. 张学工, 译. 北京: 清华大学出版社, 1999.
Vapnik V N. The Nature of Statistical Learning Theory[M]. Zhang Xuegong Translated. Beijing: Tsinghua University Press, 1999. (in Chinese)

[2] Vapnik V N. An overview of statistical learning theory[J]. IEEE Transactions on Neural Networks, 1999, 10(5): 988-999.

[3] 张学工. 关于统计学习理论与支持向量机[J]. 自动化学报, 2001, 26(1): 32-42.
Zhang Xuegong. About statistical learning theory and support vector machines[J]. Acta Automatica Sinica, 2001, 26(1): 32-42. (in Chinese)

[4] Wahba G, Lin Y, Zhang H. Margin-like quantities and generalized approximate cross validation for support vector machines[C] // Neural Networks for Signal Processing IX (Proceedings of the 1999 IEEE Signal Processing Society Workshop). Madison, USA: IEEE Computer Society Press, 1999: 12-20.

[5] Wahba G, Lin X, Gao F, et al. The bias-variance tradeoff and the randomized GACV[C] // Advances in Information Processing Systems 11. Cambridge MA: MIT Press, 1999: 620-626.

[6] Chapelle O, Vapnik V, Bousquet O. Choosing multiple parameters for support vector machines[J]. Machine Learning, 2002, 46(3): 131-159.

[7] Chung K M, Kao W C, Sun C L, et al. Radius margin bounds for support vector machines with the RBF kernel[J]. Neural Computation, 2003, 15(11): 2643-2681.

(下转第 87页)

38(3): 99- 101. (in Chinese)
[4] 王涛, 李伟生. 最短路径子图[J]. 北方交通大学学报, 2004, 28(4): 46- 49.
Wang Tao, Li Weisheng. The shortest path subgraph[J]. Journal of Northern Jiaotong University, 2004, 28(4): 46- 49. (in Chinese)
[5] 孙强, 杨宗源. 求受顶点数限制的最短路径问题的一个算法[J]. 计算机工程, 2002, 28(9): 73- 74.
Sun Qiang, Yang Zongyuan. A new algorithm for vertices-constrained shortest path[J]. Computer Engineering, 2002, 28(9): 73- 74. (in Chinese)
[6] 刘玉海, 肖江阳. 一种新型最短路径搜索算法的研究[J]. 计算机工程与应用, 2001, 37(17): 109- 110.
Liu Yuhai, Xiao Jiangyang. Research on an new shortest algorithm[J]. Computer Engineering and Applications, 2001, 37(17): 109- 110. (in Chinese)
[7] 安红岩, 胡光岷. 网络最短路径的动态算法[J]. 计算机工程与应用, 2003, 39(1): 173- 174.
An Hongyan, Hu Guangmin. A new dynamic algorithm for network mininun distance[J]. Computer Engineering and Applications, 2003, 39(1): 173- 174. (in Chinese)
[8] 杨云, 孙向军. 一种启发式遗传算法及其在最短路径求取中的应用[J]. 计算机工程与应用, 2003, 39(1): 12- 14.
Yang Yun, Sun Xiangjun. An algorithm based on illumination and it' s application in shortest path algorithm[J]. Computer Engineering and Applications, 2003, 39(1): 12- 14. (in Chinese)
[9] 毕军, 付梦印, 张宇河. 一种改进的蚁群算法求解最短路径问题[J]. 计算机工程与应用, 2003, 39(1): 107- 109.
Bi Jun, Fu Mengyin, Zhang Yuhe. An improved ant colony algorithm for the shortest path problem[J]. Computer Engineering and Applications, 2003, 39(1): 107- 109. (in Chinese)
[10] 姜启源. 数学模型[M]. 北京: 高等教育出版社, 1993.
Jiang Qiyuan. Mathematical Model[M]. Beijing: Higher Education Press, 1993. (in Chinese)
[11] 高尚. 套汇问题研究[J]. 数学的实践与认识, 2005, 35(10): 36- 40.
Gao Shang. Research on arbitrage problem[J]. Mathematics in Practice and Theory, 2005, 35(10): 36- 40. (in Chinese)

[责任编辑: 严海琳]

(上接第 79页)

[8] Xu J H. Designing nonlinear classifiers through minimizing VC dimension bound[J]. Lecture Notes in Computer Sciences, 2005, 3496: 900- 905.
[9] 徐士良. 常用算法程序集(C语言描述) [M]. 北京: 清华大学出版社, 2004.
Xu Shiliang. Set Procedures Commonly Used Algorithm (C Description Language) [M]. Beijing: Tsinghua University Press, 2004. (in Chinese)
[10] 袁亚湘, 孙文瑜. 最优化理论与方法[M]. 北京: 科学出版社, 2003.
Yuan Yaxiang, Sun Wenyu. Optimization Theory and Methods[M]. Beijing: Science Press, 2003. (in Chinese)
[11] Himmelblau D M. 实用非线性规划[M]. 北京: 科学出版社, 1981.
Himmelblau D M. Practical Nonlinear Programming[M]. Beijing: Science Press, 1981. (in Chinese)
[12] 刘宝光编著. 非线性规划[M]. 北京: 北京理工大学出版社, 1988.
Liu Baoguang. Nonlinear Programming[M]. Beijing: Beijing Institute of Technology Press, 1988. (in Chinese)
[13] Osuna E, Freund R, Girosi F. An improved training algorithm for support vector machines[C] // IEEE Workshop on Neural Networks for Signal Processing. Amelea Island, FL: IEEE, 1997: 276- 285.
[14] Joachims T. Making large-scale SVM learning practical[C] // Scholkopf B, Burges C J C, Smola A J. Advances in Kernel Methods-Support Vector Learning. Cambridge MA: MIT Press, 1998: 41- 56.
[15] Platt J C. Sequential minimal optimization: a fast algorithm for training support vector machines[R]. Microsoft Research Technical Report MSR- TR- 98- 14, 1998.

[责任编辑: 严海琳]