超球体单类支持向量机的SMO训练算法

徐 图 罗 瑜 何大可
（西南交通大学信息科学与技术学院 成都610031）

摘 要 由于One-class支持向量机能用于无监督学习，被广泛用于信息安全、图像识别等领域中。而超球体One-class支持向量机能生成一个合适的球体，将训练样本包含其中，故更适合于呈球形分布的样本学习。但由于超球体One-class支持向量机没有一种快速训练算法，使其在应用中受到限制。SMO算法成功地训练了标准SVM，其训练思想也可用于超球体One-class支持向量机的训练。本文提出了超球体One-class支持向量机的SMO训练算法，并对其空间和时间复杂度进行了分析。实验表明，这种算法能迅速、有效地训练超球体One-class支持向量机。

关键词 无监督学习，超球体One-class支持向量机，SMO训练算法

SMO Training Algorithm for Hyper-sphere One-class SVM

XU Tu LUO Yu HE Da-ke
(School of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, China)

Abstract One-Class SVM, as an unsupervised learning algorithm, is used widely in the areas of information security and image recognition etc. Moreover, Hyper-Sphere One-Class SVM can product a right sphere including the training examples, so it is fit to learn the examples with sphere-shaped distribution. However, Hyper-Sphere One-Class SVM is limited in real applications because it lacks a fast training algorithm. Training standard SVM successfully, the idea of SMO algorithm can be used to train Hyper-Sphere One-Class SVM too. The SMO algorithm for Hyper-Sphere One-Class SVM is proposed, the space and time complexity degrees are also analyzed in this paper. As shown in our numeric experiments, the new algorithm can train Hyper-Sphere One-Class SVM precisely and efficiently.

Keywords Unsupervised learning, Hyper-sphere one-class SVM, SMO algorithm

1 引言

用一个半径尽可能小的超球体将训练样本包含其中的思想最早见于文献[1]，Schölkopf等人将其用于VC维的估计。1999年，Tax等人将这种思想用于数据域描述(Support Vector Data Description, SVDD)[2]，也就是通常说的单类分类。单类分类就是根据训练样本建立“自我”区域。对于一个测试数据，如果在“自我”区域内，则认为是“内点”，否则是“野点”。这类学习机，称为单类支持向量机(One-Class SVM, OC-SVM)。1999年末，Schölkopf在前人工作的基础上，提出2种OC-SVM算法[3]，分别是基于超平面和基于超球体的OC-SVM，并提出了基于超平面OC-SVM的SMO训练算法。此后，基于超平面的OC-SVM被大量地运用于信息安全、图像处理、模式识别等领域。而基于超球体的OC-SVM，由于缺乏相应的训练算法，其应用受到了限制。

序贯最优算法(SMO)是Platt于1999年提出的用于标准SVM训练的快速优化算法[4]。本文根据SMO算法的思想，提出了超球体OC-SVM的SMO快速训练算法，并分析了该算法的空间和时间复杂度。

2 超球体One-class SVM算法

根据文献[2]，超球体OC-SVM算法的思想为寻找一个超球体，使其半径尽可能小的同时，包含的训练样本数尽可能多。其目标函数为

$$\min_R R^2+C\sum_{i=1}^{l}\zeta_i \tag{1}$$

$$s.t.\ \|\Phi(x_i)-a\|^2\leqslant R^2+\zeta_i \tag{2}$$

$$\zeta_i\geqslant 0 \tag{3}$$

其中R为球体半径，a为球心，ζ为松弛变量，l为训练样本的个数。C为正则化参数，控制对错分样本的惩罚程度，实现对球的大小和所包含的样本数之间的折衷。同SVM类似，当样本点非线性可分时，使用非线性映射$\Phi(x_i)$，将样本点映射到高维特征空间，在那里样本是线性可分的，并选择满足Mercer条件的核函数$K(x,y)=\langle\Phi(x),\Phi(y)\rangle$，其中$\langle\Phi(x),\Phi(y)\rangle$表示$\Phi(x)$与$\Phi(y)$的内积。

(1)-(3)式的Lagrange函数为

$$L(R,a,\zeta_i,\alpha_i,\beta_i)=R^2+C\sum_{i=1}^{l}\zeta_i-\sum_{i=1}^{l}\alpha_i(R^2+\zeta_i-\|\Phi(x_i)-a\|^2)-\sum_i^{l}\beta_i\zeta_i \tag{4}$$

(1)-(3)式的对偶问题为

$$W(a)=\min_{\alpha}\sum_i\sum_j\alpha_i\alpha_j k(x_i,x_j)-\sum_{i=1}^{l}a_j(x_i,x_j) \tag{5}$$

其中α_i满足$\sum_{i=1}^{l}\alpha_i=1$ (6)

式(5)为二次规划(Quadratic Programming, QP)问题。求解上述QP问题，可得Lagrange乘子α_i，即可确定球心和半径。球面上的点对球体的确定起关键作用，称其为支持向

徐 图 博士生，主要研究方向为机器学习、智能网络安全；**罗 瑜** 博士生，主要研究方向为并行计算及模式识别；**何大可** 博士生导师，教授，主要研究方向为密码学、信息安全、并行计算。

量(SVs)。

3 超球体 OC-SVM 的 SMO 训练算法

根据最优化原理，当所有的 α_i 满足目标函数的 Karush-Kuhn-Tucker(KKT)条件，则可认为是原方程的一个解。因此，首先给出判别 α_i 是否违反 KKT 条件的方法。

定理 1 对 $\forall \alpha_i$，若满足问题(1)-(3)的 KKT 条件，等价于满足下列条件：

(1) 当 $\alpha_i=0$ 时，$d_i^2\leqslant R^2$；当 $0<\alpha_i<C$ 时，$d_i^2=R^2$；当 $\alpha_i=C$ 时，$d_i^2\leqslant R^2$；

(2) $\sum\limits_{i=1}^{l}\alpha_i=1$

其中 α_i 为 Lagrange 乘子，d_i^2 为样本点 x_i 到球心 a 的距离的平方，R^2 为球体半径的平方。

证明：由(4)式可得 QP 问题(1)-(3)式的 KKT 条件为

(a) $\dfrac{\partial L}{\partial R}=0\Rightarrow\sum\limits_{i}^{l}\alpha_i=1$；

(b) $\dfrac{\partial L}{\partial a}=0\Rightarrow a=\sum\limits_{i=1}^{l}\alpha_i$；

(c) $\dfrac{\partial L}{\partial \zeta_i}=0\Rightarrow C-\alpha_i-\beta_i=0$；

(d) $\alpha_i(R^2+\zeta_i-d_i^2)=0$；

(e) 由(2)，$R^2+\zeta_i-d_i^2\geqslant 0$；

(f) $\beta_i\zeta_i=0$；

(g) 由(3)，$\zeta_i\geqslant 0$；

(h) $\alpha_i\geqslant 0,\beta_i\geqslant 0$。

当 $\alpha_i=0$ 时，由(c)知 $\beta_i=C$，由(f)得 $\zeta_i=0$，由(e)有 $d_i^2\leqslant R^2$；

当 $0\leqslant\alpha_i<C$ 时，由(c)得 $\beta_i>0$，再由(f)知 $\zeta_i=0$，由(d)得 $d_i^2=R^2$；

当 $\alpha_i=C$ 时，由(d)有 $R^2+\zeta_i-d_i^2=0$，则由(g)得 $d_i^2\geqslant R^2$。

由此可知条件(1)成立。

条件(2)可由式(6)获得。 □

在实际应用中，并不要求上述条件完全满足，而是在一定精度 e 下满足即可。

SMO 算法的思想是对违反 KKT 条件的 Lagrange 乘子两两地进行优化，直到所有的 Lagrange 乘子均满足 KKT 条件为止。

设 α_1 和 α_2 是不满足 KTT 条件的两个乘子，对它们进行优化时，将其他乘子暂视为常数，并记 α_1,α_2 经优化后的值为 λ_1,λ_2。

为保证 λ_1,λ_2 在可行区域内，现讨论 λ_1,λ_2 的取值范围。

定理 2 λ_1,λ_2 的可行区域为 $0\leqslant\lambda_1\leqslant C,L\leqslant\lambda_2\leqslant H$，其中 $L=\max(0,\alpha_1,+\alpha_2-C)$，$H=\min(\alpha_1+\alpha_2,C)$

证明：设 $h=\sum\limits_{i=3}^{l}\alpha_i$，由(6)式得

$\alpha_1+\alpha_2=\lambda_1+\lambda_2=1-h$

于是

$$\lambda_2=\alpha_1+\alpha_2-\lambda_1 \tag{7}$$

由于 $0\leqslant\lambda_1,\lambda_2\leqslant C$，当 $\lambda_1=0$ 时，由(7)式知 λ_2 取得最大值，故 $H=\min(\alpha_1+\alpha_2,C)$；当 $\lambda_1=C$ 时，λ_2 取得最小值，故 $L=\max(0,\alpha_1+\alpha_2-C)$。 □

定理 3 α_1,α_2 经优化后得的值为 $\lambda_1=\alpha_1+\alpha_2-\lambda_2$，$\lambda_2=\begin{cases}H, & \bar{\lambda}_2\geqslant H\\ \bar{\lambda}_2, & L<\bar{\lambda}_2<H\\ L, & \bar{\lambda}_2\leqslant L\end{cases}$，其中 $\bar{\lambda}_2=\alpha_2+\dfrac{d_2^2-d_1^2}{2\eta}$，$\eta=K(x_1,x_1)-2K(x_1,x_2)+K(x_2,x_2)$，$d_i^2=\|\Phi(x_i)-\alpha\|^2=K_{ii}-2\sum\limits_{j=1}^{l_m}\alpha_jK_{ij}+\alpha^2$

此处，K_{ij} 是 $K(x_i,x_j)$ 的简写。

证明：将式(5)改写为关于 λ_1,λ_2 的目标函数：

$$W(\lambda_1,\lambda_2)=\lambda_1^2K_{11}+2\lambda_1\lambda_2K_{12}+\lambda_2^2K_{22}+2\lambda_1\sum_{j=3}^{l}\alpha_jK_{1j}+2\lambda_2\sum_{j=3}^{l}\alpha_jK_{2j}-\lambda_1K_{11}-\lambda_2K_{22}+Q_1-Q_2 \tag{8}$$

其中，$Q_1=\sum\limits_{i=3}^{l}\sum\limits_{j=3}^{l}\alpha_i\alpha_jK_{ijk}$，$Q_2=\sum\limits_{i=3}^{l}\alpha_iK_{ii}$，而 Q_1,Q_2 均不依赖于 λ_1,λ_2，故可视为常数。

$$令\ \alpha_1+\alpha_2=\lambda_1+\lambda_2=A \tag{9}$$

则有 $\lambda_1=A-\lambda_2$，并记 $q_i=\sum\limits_{j=3}^{l}\alpha_jK_{ij}$，将其代入(8)式，得

$$W(\lambda_2)=(A-\lambda_2)^2K_{11}+2(A-\lambda_2)\lambda_2K_{12}+\lambda_2^2K_{22}+2(A_{q_1}-\lambda_{2q_1}+\lambda_{2q2})-(A-\lambda_2)K_{11}-\lambda_2K_{22}+Q_1-Q_2$$

$$\frac{\partial W}{\partial\lambda_2}=2(\lambda_2-A)K_{11}+2(A-\lambda_2)K_{12}-2\lambda_2K_{12}+2\lambda_2K_{22}+2(q_2-q_1)+K_{11}-K_{22} \tag{10}$$

计算 $\dfrac{\partial^2 W}{\partial\lambda_2^2}=K_{11}-2K_{12}+K_{22}=\eta$，当 $x_1\neq x_2$ 时，$\eta>0$，$W(\lambda_2)$ 有唯一极小值，令 $\dfrac{\partial W}{\partial\lambda_2}=0$，则由(10)式得

$$\lambda_2=\frac{2(A-1)K_{11}-2AK_{12}+K_{22}+2(q_1-q_2)}{2(K_{11}-2K_{12}+K_{22})} \tag{11}$$

$$由\ d_i^2=\|\Phi(x_i)-a\|^2=K_{ii}-2\sum_{j=1}^{l}\alpha_jK_{ij}+\alpha^2=K_{ii}-2\alpha_1K_{1i}-2\alpha_2K_{2i}-2q_i+\alpha^2 \tag{12}$$

则 $2q_i=K_{ii}-2\alpha_iK_{1i}-2\alpha_2K_{2i}-d_i^2+\alpha^2,i=1,2$。且 $A=\alpha_1+\alpha_2$，代入(11)式得 $\lambda_2=\alpha_2+\dfrac{d_2^2-d_1^2}{2\eta}$，又由(9)式得 $\lambda_1=\alpha_1+\alpha_2-\lambda_2$，根据定理 2 有 $L\leqslant\lambda_2\leqslant H$，可知定理成立。 □

为将训练集分为不同子集，称满足 $\alpha_i=0$ 或 $\alpha_i=C$ 的点为边界上的点；称满足 $0<\alpha_i<C$ 的点为界内的点，并按照定理 1 判断某点是否违反 KKT 条件。

综上所述，超球体 One-class SVM 的训练过程如下：

step 1 对于 l 个训练样本，初始化 $\alpha_i=1/l,\alpha=0,R=0$，并根据(12)式计算 d_i^2；

step 2 搜索所有界内的点，寻找第 1 个违反 KKT 条件的点 x_1，若找到，转 step 4；

step 3 搜索所有边界上的点，寻找第 1 个违反 KKT 条件的点 x_1，若没找到，转 step 7；

step 4 为使目标函数最大限度的下降，选择使 $|d_1^2-d_2^2|$ 取最大值的点为 x_2，按定理 3 优化 α_1 和 α_2；

step 5 根据优化后的值 λ_1,λ_2 更新 α^2，其中 $\alpha^2=\sum\limits_{i}\sum\limits_{j}\alpha_i\alpha_jK_{ij}$，然后更新 d_1^2,d_2^2，并计算 $R^2=(d_1^2+d_2^2)/2$；

step 6 返回 step 2；

step 7 算法收敛，训练结束。

4 SMO 算法的空间和时间复杂度分析

4.1 空间复杂度

SMO 算法是无需存储核矩阵的[4]，但为了加快训练过程，从式(12)可以看出，存储 K_{ii} 能减少重复计算。这样，

SMO 算法的空间复杂度为 $O(l)$，l 为样本个数。

4.2 时间复杂度

在搜索第一个乘子时，要分别搜索边界上和界内的点，设在某次搜索时有 n_B 个边界点，n_{NB} 个界内点，计算一次 d_i^2 需要计算 l 次核函数。又 $n_B+n_{NB}=l$，设样本的维数为 m，于是搜索第 1 个乘子的时间复杂度为 $f_1=O(\frac{1}{2}l^2m)$。搜索第 2 个乘子需要遍历所有的样本，故其时间复杂度为 $f_2=O(l^2m)$，f_1+f_2 是进行一对乘子优化时的时间复杂度。若优化过程共进行了 t 次迭代，算法收敛，则总的时间复杂度为 $f=t\cdot(f_1+f_2)=O(\frac{3}{2}l^2m\cdot t)$。

5 数值实验

为了检验本文提出的 SMO 算法对超球体 One-Class SVM 进行训练的效果，进行了数值实验。实验数据分别采用 UCI 数据库中的 Pen-based Handwritten Digits 数据集和 Optical Recognition of Handwritten Digits 数据集[5]。两数据集均包含 10 类数据，第 1 个数据集中共包括 7494 个训练数据和 3498 个测试数据，每个数据为 16 维向量；第 2 个数据集中共包括 3823 个训练数据和 1797 个测试数据，每个数据为 64 维向量。分别为每类数据建立一个超球体 One-Class SVM 进行训练，并用相应的测试数据进行检验。实验环境为 Inter Xeon 3G，1G RAM，Windows Server 2003，编译环境为 C++ Builder 6.0。

实验中使用高斯核函数为

$K(x,y)=\exp(-\|x-y\|^2/2\sigma^2)$

训练精度取 $e=0.1$，$C=0.8$。取不同的核参数 σ，记录最好的结果。设总检测率为各类被正确分类样本之和占总样本数的百分率。表 3 示出了在第 1 个数据集上取 $\sigma=50$ 的检测结果，其总检测率为 93.39%。表 4 示出了在第 2 个数据集上取 $\sigma=15$ 的检测结果，其总检测率为 95.26%。时间单位为秒。

表 1 Pen-based Handwritten Digits 数据集的构成

数据类型	1类	2类	3类	4类	5类	6类	7类	8类	9类	10类	总和
训练集	780	779	780	719	780	720	720	778	719	719	7494
测试集	363	364	364	336	364	335	336	364	336	336	3498

表 2 Optical Recognition of Handwritten Digits 数据集的构成

数据类型	1类	2类	3类	4类	5类	6类	7类	8类	9类	10类	总和
训练集	376	389	380	389	387	376	377	387	380	382	3823
测试集	178	182	177	183	181	182	181	179	174	180	1797

表 3 在两个数据集上的训练结果

	Pen-based Handwritten Digits		Optical Recognition of Handwritten Digits	
类别	训练时间(s)	检测率(%)	训练时间(s)	检测率(%)
1	3.487	95.31	2.262	96.06
2	3.484	89.01	2.340	98.35
3	3.485	95.05	2.286	95.48
4	3.215	98.21	2.352	88.52
5	3.482	91.75	2.328	98.34
6	3.220	88.65	2.259	96.7
7	3.228	97.02	2.268	100
8	3.479	84.06	2.463	96.64
9	3.205	99.7	2.271	87.93
10	3.251	96.13	2.298	94.44

实验表明，高斯核最适合于超球体 One-Class SVM。超球体 One-Class SVM 对参数 C 并不敏感；参数 σ 能影响训练速度：σ 越大，速度越慢。表 3 说明本文提出的 SMO 算法能对超球体 One-Class SVM 进行快速有效的训练，并获得较高的检测精度。

结束语 超球体 One-Class SVM 由于一直缺乏有效的训练算法，使其在应用中受到限制。本文根据标准 SVM 的 SMO 训练算法思想，推导出了超球体 One-Class SVM 的 SMO 训练算法，并分析这种算法的空间和时间复杂度，最后用实验证明了这种算法的有效性。

参考文献

[1] Schölkopf B, Burges C, Vapnik V. Extracting support data for a given task[C]// Fayyad U M, Uthurusamy R, eds. Proceedings, First International Conference on Knowledge Discovery & Data Mining. Menlo Park, CA: AAAI Press, 1995

[2] Tax D M J, Duin R P W. Data domain description by support vectors[C]// Verleysen M, ed. Proceedings ESANN, Brussels, 1999: 251 - 256

[3] Schölkopf B, Platt J, Shawe-Taylor J A S, et al. Estimating the support of a highdimensional distribution[J]. Neural Computation, 1990, 13: 7

[4] Platt J. Fast training of support vector machines using sequential minimal optimization [M]// Schölkopf B, Burges C, Smola A, eds. Advances in Kernel Methods—Support Vector Learning. Cambridge, MA: MIT Press, 1999: 185-208

[5] http://www.ics.uci.edu/-mlearn/MLRepository.html