增量密度加权近似支持向量机

鲁淑霞 崔芳芳 忽丽莎
(河北省机器学习重点实验室 河北大学数学与计算机学院 保定 071002)

摘 要 近似支持向量机(PSVM)是一个正则化最小二乘问题,有解析解,但是它失去了支持向量机(SVM)的稀疏性,使得所有的训练样例都成为支持向量。为了有效地控制近似支持向量机的稀疏性,提出了增量密度加权近似支持向量机(IDWPSVM),它在训练集中选取最基本的支持向量。实验表明,IDWPSVM方法与SVM,PSVM和DWPSVM方法相比,其精度相似,收敛速度快,可有效地控制近似支持向量机的稀疏性。

关键词 近似支持向量机,密度加权,增量,稀疏性
中图法分类号 TP181 **文献标识码** A

Incremental Density Weighted Proximal Support Vector Machine

LU Shu-xia CUI Fang-fang HU Li-sha
(Key Lab of Machine Learning, Hebei Province, College of Mathematics and Computer, Hebei University, Baoding 071002, China)

Abstract The proximal support vector machines (PSVM) is a regularized least-squares problem, which has an analytic solution. However, the PSVM lacks sparseness, and all training dates become support vectors. This paper focused on effectively controling the sparseness of the PSVM. An incremental density weighted proximal support vector machine (IDWPSVM) was proposed, which selects the basis support vectors in the training set. The experiment results show that the accuracy of the IDWPSVM can is similar with the SVM, PSVM and DWPSVM methods, and convergence speed is faster. The IDWPSVM can effectively control the sparseness of the PSVM.

Keywords Proximal support vector machine, Density weight, Increment, Sparseness

1 引言

标准的支持向量机(SVM)[1]的求解属于二次规划问题,对于大规模数据,SVM的训练时间较长。为了减少时间的复杂度,许多学者提出了SVM的各种变形,如最小二乘支持向量机(Least Square Support Vector Machines, LSSVM)[2,3]、近似支持向量机(Proximal Support Vector Machines, PSVM)[4]、约简支持向量机(Reduced Support Vector Machines, RSVM)[5]。文献[6]给出了多类近似支持向量机;文献[7]提出了一种改进的近似支持向量机。文献[8]是PSVM的增量版本,适合解决大样本和高维数据问题。

标准的PSVM方法是Mangasarian等人基于优化理论提出的,它将SVM的二次规划问题转化为求解线性方程组的问题,可以得到解析解,适合解决大样本问题,但是它使所有训练样例都成为了支持向量,失去了标准SVM的稀疏特性。因此需要对PSVM模型进行稀疏化。文献[9]提出了去除部分样本的方法;文献[10]去掉具有最小引入误差的样本,获得了好的稀疏性能;文献[11]用剪枝法控制稀疏性。为了有效地控制PSVM方法的稀疏性,本文在文献[12]的基础上,提出了增量密度加权近似支持向量机方法,其从训练集合中选取使得目标函数值最小的点作为支持向量,进而选取所期望的支持向量的个数,很好地控制了PSVM的稀疏性。

2 密度加权近似支持向量机

2.1 标准的PSVM

标准的PSVM通过拟合两类数据,得到两个平行的间隔面,这两个间隔面通过类中心,使得样本的聚类误差尽可能小,且两个间隔面的间隔最大。

标准的PSVM的优化问题:

$$\min_{(\omega,b,\xi)\in R^{n+1+m}} \upsilon \| \xi \|^2+\frac{1}{2}(\omega^{\mathrm{T}}\omega+b^2) \tag{1}$$

$$\text{s.t. } D(A\omega-eb)+\xi=e$$

式中,矩阵$A\in R^{m\times n}$表示训练集合,A_+表示训练集合中正类样本组成的集合,A_-表示训练集合中负类样本组成的集合;$\xi\in R^m$是误差项;$\omega\in R^n$是法向量;υ为正则化常数;$e\in R^m$为全1向量;D是一个$m\times m$的对角矩阵,对角线元素取+1时,对应的点为点集A_+中的元素,对角线元素取−1时,对应的点为点集A_-中的元素。

2.2 密度加权近似支持向量机

标准PSVM的间隔面经过类中心,一般情况下,距离类中心较近点周围的样例个数较多,密度较大,而类中心附近的点是重要的点。标准的PSVM在分类时,给样例赋予了相同

到稿日期:2012-01-05 返修日期:2012-06-08 本文受国家自然科学基金项目(611700040),河北省自然科学基金项目(F2011201063,F2010000323)资助。

鲁淑霞(1966—),女,博士,教授,主要研究方向为机器学习与计算智能、支持向量机,E-mail:cmclusx@126.com;崔芳芳(1984—),女,硕士,主要研究方向为支持向量机;忽丽莎(1986—),女,硕士,主要研究方向为支持向量机。

的惩罚因子，而通常情况下，每个样例的重要程度是不一样的，因此将密度加权加入到目标函数的二范数误差项中。定义样本$\{x_k, y_k\}$处的密度指标为：

$$M_k=\sum_{j=1}^{N}\exp\left(-\frac{\| x_k-x_j \|^2}{\gamma_a/2}\right) \tag{2}$$

式中，γ_a 表示其邻域半径，一般取值为 0.05～0.2 之间，N 表示 x_k 的 γ_a 邻域中点的个数。如果$\{x_k, y_k\}$和$\{x_j, y_j\}$之间的距离越近，密度指标 M_k 的贡献就越大，如果一个样本相邻的其它样本点很多，则该样本具有较高的密度指标值。

密度加权近似支持向量机(Density Weighted Proximal Support Vector Machines，DWPSVM)[13]的目标函数为：

$$\min_{(\omega,b,\xi)\in R^{n+1+m}} \| M^{1/2}\xi \|^2+\frac{1}{2}(\omega^{\mathrm{T}}\omega+b^2) \tag{3}$$

s. t. $D(A\omega-eb)=e-\xi$

$M\in R^{m\times m}$是一个对角矩阵：

$$\begin{cases} M_{ij}=M_i, & i=j \\ M_{ij}=0, & i\neq j \end{cases} \tag{4}$$

2.2.1 线性 DWPSVM

对于线性的 DWPSVM，我们可以将目标函数转换为如下的 Lagrange 函数：

$$L(\omega,b,\xi,\alpha)=\| M^{1/2}\xi \|^2+\frac{1}{2}(\omega^{\mathrm{T}}\omega+b^2)-\alpha^{\mathrm{T}}[D(A\omega-eb)+\xi-e]$$

式中，$\alpha\in R^m$ 是 Lagrange 乘子。根据 Karush-Kuhu-Tucher (KKT)条件，Lagrange 函数关于 ω, b, ξ 求导可以得到：

$$\omega=A^{\mathrm{T}}D\alpha, b=-e^{\mathrm{T}}D\alpha, \xi=\frac{1}{2}M^{-1}\alpha$$

因此可以得到 α 的表达式：

$$\alpha=\left(\frac{1}{2}M^{-1}+HH^{\mathrm{T}}\right)^{-1}e \tag{5}$$

式中，$H=D[A-e]$。

则线性 DWPSVM 的决策函数为：

$$x^{\mathrm{T}}\omega-b\begin{cases} >0, & \text{那么 } x\in A_+ \\ <0, & \text{那么 } x\in A_- \\ =0, & \text{那么 } x\in A_+ \text{或 } A_- \end{cases} \tag{6}$$

2.2.2 非线性 DWPSVM

当非线性可分时，引入核函数 K，用式 $\omega=A^{\mathrm{T}}D\alpha$ 代替目标函数中的 ω，用非线性核 $K(A, A^{\mathrm{T}})$来代替 AA^{T}，于是得到非线性 DWPSVM 的目标函数：

$$\min_{(\omega,b,\xi)\in R^{n+1+m}} \| M^{1/2}\xi \|^2+\frac{1}{2}(\alpha^{\mathrm{T}}\alpha+b^2) \tag{7}$$

s. t. $D(K(A, A^{\mathrm{T}})D\alpha-eb)=e-\xi$

令 $K=K(A, A^{\mathrm{T}})$，可以得到拉格朗日函数。根据 KKT 条件，得到相应的拉格朗日乘子 η：

$$\begin{aligned}\eta&=[(M^{1/2})^{-1}+D(KK^{\mathrm{T}}+ee^{\mathrm{T}})D]^{-1}e\\&=[(M^{1/2})^{-1}+HH^{\mathrm{T}}]^{-1}e\end{aligned} \tag{8}$$

这里 $H=D[K-e]$，由于 K 是一个方阵，转置以后维数不会降低，因此式(8)用 Sherman-Morrison-Woodbury[14]求解不再适用。由于数据集的规模很大，因此求逆矩阵的规模也很大，很耗费时间，这里使用约简支持向量机(Reduced Support Vector Machine，RSVM)[5]。

则非线性 DWPSVM 的决策函数为：

$$(K(x^{\mathrm{T}}, A^{\mathrm{T}})K(A, A^{\mathrm{T}})^{\mathrm{T}}+e^{\mathrm{T}})D\eta\begin{cases} >0, & \text{那么 } x\in A_+ \\ <0, & \text{那么 } x\in A_- \\ =0, & \text{那么 } x\in A_+ \text{或 } A_- \end{cases} \tag{9}$$

3 增量密度加权近似支持向量机

由于 PSVM 将标准 SVM 的不等式约束改成了等式约束，因此它失去了 SVM 的稀疏特性。标准 PSVM 在分类时，给样例赋予了相同的惩罚因子，而通常情况下，每个样例的重要程度是不同的，因此给出了密度加权近似支持向量机。为了控制密度加权近似支持向量机的稀疏性，给出了增量密度加权近似支持向量机(IDWPSVM)。

稀疏性问题可以通过减少支持向量技术来缓和。如果 $A_2\in R^{p\times n}$是训练集合 $A\in R^{m\times n}$ 的子矩阵($p<m$，p 表示支持向量的个数，即支持向量的个数比训练点的个数少)，核矩阵为 $K=K(A, A_2^{\mathrm{T}})$，$K\in R^{m\times p}$，则通过选取支持向量降低训练分类器的复杂度。

为了提高 DWPSVM 的精度，从训练集合 A 中任意选取子集合构成 A_2，然而并不是所有的自由选取的矩阵 A_2 都具有相同的性能。如果一个给定的分类器含有 p 个支持向量，那么可以找到一个含有 p 个训练点的最优的子集合 A_2^*，使得相应的 DWPSVM 具有很好的性能，下面给出一种求集合 A_2^* 的较好的方法。

IDWPSVM 方法是从一定数量的数据点中选取一个点，使得目标函数值最小，然后将这个点作为支持向量，这个过程一直迭代，直到最终达到所预期的支持向量的个数。

IDWPSVM 训练算法：

初始化：训练数据集 A，支持向量 $A_2=[]$，类标 D，高斯核参数 σ，所需的支持向量个数 p，每次迭代的次数 q。

步骤 1 计算密度权矩阵 $M_k=\sum_{j=1}^{N}\exp\left(-\frac{\| x_k-x_j \|^2}{\gamma_a/2}\right)$。

步骤 2 (选取所需要的支持向量)

for i=1 to p

(1)从 A 中任意选出 q 行(不包括 A_2 中已经含有的行)，记作 B_1，$B_2, \cdots, B_q$。

(2)for j=1 to q

(i)$K=K(A, [A_2; B_j])$;

(ii) 用更新后的 K 求解优化问题(3)，得到 $\xi, \omega^{(j)}, b^{(j)}$;

(iii) $z_j=\| M^{1/2}\xi \|^2+\frac{1}{2}((\omega^{(j)})^{\mathrm{T}}(\omega^{(j)})+(b^{(j)})^2)$。

(3)$j^*=\arg\min_j(z_j)$。

(4) 更新结果：$A_2=[A_2; B_{j^*}]$，$\omega=\omega^{(j^*)}$，$b=b^{(j^*)}$。

输出：支持向量 A_2，法向量 ω，偏置 b。

步骤 3 根据输出的支持向量 A_2，法向量 ω，偏置 b，训练分类器，然后对样例进行预测。

其中算法(i) $K=K(A, [A_2; B_j])$中的';'表示将 B_j 行加入到 A_2 中。

PSVM 的间隔面通过类中心，所有的数据点都是支持向量，它的稀疏性是隐含的、不可控制的。IDWPSVM 用于寻找与 PSVM 相似的超平面，控制 PSVM 的稀疏性。在迭代的过程中，其找到所预期的支持向量的个数，控制了稀疏性，同时更好地增进了与 PSVM 间隔面的相似性。由于 IDWPSVM 是从一定的数据点中选取一个点，当这个点所对应的目标函数值最小时，将这个点作为支持向量，因此当数据集合很大

时，选取支持向量的这个过程所需要的时间就会相对增加，运行时间的复杂度正比于所选支持向量的个数。

4 实验结果与分析

为了验证所提出的增量密度加权近似支持向量机的性能，在 UCI[15] 机器学习数据库中的 IRIS 和 CMC 数据集合上进行了实验。图 1 和图 2 是在 IRIS 数据集上所做的实验，其中图 1 选取的是 IRIS 数据集的前 3 个属性，图 2 选取的是 IRIS 数据集属性中的 2 个属性。IRIS 数据为 3 类，人为地将其分为两类，'+'是一类，中间部分 '.' 和右下角的'.'是一类，用'o'和 '◇'表示用 IDWPSVM 方法提取出来的支持向量。图 3 和图 4 是 CMC 数据集上所做的实验，图 3 是选取数据集的前 3 个属性所做的实验，图 4 是选取数据集的 2 个属性所做的实验。其中'.'表示训练数据点，'◇'表示用 IDWPSVM 方法从训练集合中选取的支持向量。如图所示 IDWPSVM 方法所选取的支持向量的个数比数据集的个数要少很多，它能够根据样本的分布情况保留使得目标函数值最小的点，也即对分类器起关键作用的点作为支持向量，这在一定程度上会缩短训练分类器的时间。从图 2 可以看出，用 IDWPSVM 提取出来的支持向量可以训练出分类器，有效地将 IRIS 数据集合分类。

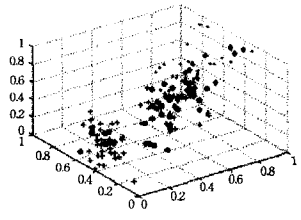

图 1 IRIS 数据图像(3 个属性)

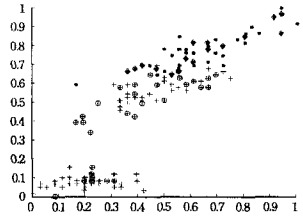

图 2 IRIS 数据图像(2 个属性)

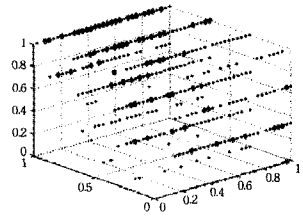

图 3 CMC 数据图像(3 个属性)

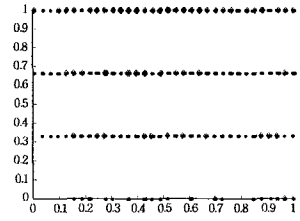

图 4 CMC 数据图像(2 个属性)

为了进一步验证 IDWPSVM 的性能，在 UCI 机器学习数据库中的 5 个数据集上进行了试验，并与其它的方法进行了比较。这里所选取的 Pima 数据是 2 类，Cmc，Yeast，Sat，Spambase 数据集都是多类数据，人为地将多类数据分为 2 类。表 1 给出了所选数据集合样例的个数、维数，以及分成 2 类以后的正负样例的个数，正类用 P 表示，负类用 N 表示。

表 1 实验数据

数据集合	样例个数	维数	正负样例的个数	
Pima	768	9	P:268	N:500
Cmc	1473	10	P:629	N:844
Yeast	1484	9	P:1299	N:185
Sat	4435	37	P:3397	N:1038
Spambase	4601	58	P:1813	N:2788

表 2 是在 UCI 数据库中 Yeast，Pima 数据集上所做的实验。根据 IDWPSVM 的方法，从训练集合中分别选取 50，100，200，400 个支持向量，根据这些支持向量训练出分类超平面，然后对数据集合进行分类，从结果可以看出，随着所选取的支持向量个数的增加，分类器的分类性能逐渐提高。

表 2 针对不同支持向量个数，IDWPSVM 方法的精度

数据集	支持向量个数			
	50	100	200	400
Yeast	49.59	80.58	93.39	94.57
Pima	64.18	76.87	77.61	78.12

在求解 DWPSVM 的过程中，选用的核函数是高斯核函数，核参数的选取采用的是网格搜索法，即将参数值的可行区间划分为一系列的小区间，然后求出对应各参数值的组合，从而求得该区间内最小目标值及其对应的参数值。表 3 给出了用网格搜索法得到的各个数据集合所对应的高斯核函数的参数。从数据集合中选取 70%的样例作为训练样例，30%的样例作为测试样例来测试分类器的性能。

表 3 高斯核参数 σ

数据集	Pima	Cmc	Yeast	Sat	Spambase
σ	50	10	45	8.5	0.5

表 4 列出了 SVM、标准 PSVM、DWPSVM、IDWPSVM 的分类结果，选用高斯核函数，核参数的选择采用的是网格搜索法。从表 4 的实验结果可以看出 4 种分类方法的分类性能，IDWPSVM 的精度比其他 3 种方法所得到的精度稍高，或者与其他 3 种方法的精度相似，但是在训练分类器的时间上，IDWPSVM 比其他 3 种方法所用的时间都要少。因此所提出的增量密度加权近似近似支持向量机所用的训练时间较少，具有较好的分类性能，同时它还可以得到所预期的支持向量的个数，从而很好地控制了近似支持向量机的稀疏性。

表 4 分类结果

数据集		SVM	PSVM	DWPSVM	IDWPSVM
Pima	时间(s)	3.460	0.046	0.343	0.045
	精度(%)	67.54	80.97	81.72	78.12
Cmc	时间(s)	53.09	0.063	5.094	0.051
	精度(%)	68.08	66.81	67.23	69.16
Yeast	时间(s)	5.806	0.141	3.359	0.041
	精度(%)	85.545	85.54	86.57	94.21
Sat	时间(s)	124.92	4.390	83.109	3.276
	精度(%)	97.22	96.79	96.79	97.26
Spambase	时间(s)	211.69	4.891	91.781	3.472
	精度(%)	80.87	99.93	99.95	99.84

结束语 针对 PSVM 失去稀疏性的问题，给出了一种增量密度加权近似支持向量机的方法。该方法从训练集合中选取预先指定的 p 个点，这些点在一定的点集范围内使得目标函数值最小，将这些点作为分类器的最基本的支持向量，从而有效地控制了 DWPSVM 的稀疏性，提高了 DWPSVM 的分类性能。实验结果表明，随着支持向量的个数 p 的增加，所训练出来的分类器的分类性能逐渐增强，但是运行时间的复杂度正比于所选支持向量的个数。

参 考 文 献

[1] Vapnik V N. The Nature If Statistical Learning Theory [M]. USA, New York: Springer-Verlag, 1995: 273-297

[2] Suykens J A K, Vandewalle J. Least squares support vector machine classifiers [J]. Neural Processing Letters, 1998, 9: 293-300

(下转第 207 页)

从图8—图10中可以看出，对于UCI上的3个数据集，本文提出的AISS算法的聚类准确性都是比较高的。聚类性能都可以随着约束对的增加而逐渐增强，CCL算法则有一定的波动性。对于IRIS数据集和Wine数据集，当约束对较少时，AISS算法的聚类正确率就可以达到90%，这可能是由于这两个数据集的各类样本点在较大程度上分布在某一流形上，使用流形距离度量时，能让同类中的样本点距离更近，这也符合大多数聚类样本的性质。

综合以上实验可以看到，本文所提出的算法对于人工数据集及真实数据集均有很好的聚类效果。

结束语 本文提出一种基于近邻流形距离的人工免疫半监督聚类算法，其将成对约束信息加入到样本度量矩阵中；通过近邻流形距离得到新的度量矩阵，将聚类问题转化为一个组合优化模型；采用能收敛到全局最优解的克隆选择算法求解模型，以获得最优的聚类划分；在人工数据集和标准数据集上的仿真实验结果表明，本文所提算法具有良好的聚类性能。

参 考 文 献

[1] Demiriz A, Benneit K P. Semi-Supervised clustering using genetic algorithm [C]// ANNIE'99. New York: ASME Press, 1999: 809-814

[2] Xing E P, Ng A Y, Jordan M. Distance metric learning, with application to clustering with side-information [C]// The 16th Annual Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2003: 505-512

[3] Bilenko M, Basu S, Mooney R J. Integrating Constraints and Metric Learning in Semi-Supervised Clus tering [C]// The 21st International Conference on Machine Learning. Banff, Canada, 2004: 81-88

[4] Basu S, Bilenko M, Mooney R J. A Probabilistic Framework for Semi-Supervised Clustering[C]// The 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2004: 59-68

[5] Yin X S, Chen S C, Hu E L, et al. Semi-supervised clustering with metric learning: An adaptive kernel method [J]. Pattern Recognition, 2010, 43(4): 1320-1333

[6] Mahdieh S B, Saeed B S. Kernel-based metric learning for semi-supervised clustering [J]. Neurocomputing, 2010, 73(7-9): 1352-1361

[7] Zhang H X, Lu J. Semi-supervised fuzzy clustering: A kernel-based approach [J]. Knowledge-based systems, 2009, 22(6): 477-481

[8] 冯晓磊，于洪涛. 基于流形距离的半监督近邻传播聚类算法[J]. 计算机应用研究，2011，28(10)：3656-3664

[9] 公茂果，焦李成，马文萍，等. 基于流形距离的人工免疫无监督分类与识别算法[J]. 自动化学报，2008，34(3)：367-375

[10] 焦李成，杜海峰，刘芳，等. 免疫优化计算、学习与识别[M]. 北京：科学出版社，2006：92-99

[11] Klein D, Kamvar S D, Manning C D. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering[C] // Proc of the 19th International Conference on Machine Learning. Sydney, 2002: 307-314

[12] Bilenko M, Basu S, Mooney R J. Integrating constraints and metric learning in semi-supervised clustering[C]//Proc of the 21st International Conference on Machine Learning. Banff: ACM Press, 2004: 81-88

(上接第196页)

[3] Rubio G, Pomares H, Rojas I. A heuristic method for parameter selection in LS_SVM: Application to time series prediction [J]. International Journal of Forecasting, 2011, 27: 725-739

[4] Fung G, Mangasarian O. Proximal support vector machines: knowledge discovery and data mining[C]// Proceedings of the Knowledge Discovery and Data Mining Conference. San Francisco, 2001: 77-86

[5] Lee Y J, Mangasarian O L. RSVM: Reduced support vector machines[C]// Proceedings of the First SIAM International Conference on Data Mining. Chicago, 2001: 5-7

[6] Glenn M, Fung O L. Mangasarian. Multicategory proximal support vector classifiers [J]. Machine Learning, 2005, 59: 77-97

[7] 陶晓燕，姬红兵，董淑福. 改进的PSVM及其在非平衡数据分类中的应用[J]. 计算机工程，2007，33(24)：191-193

[8] Zhu Zhen-feng, Zhu Xing-quan, Guo Yue-fei, et al. Inverse matrix-free incremental proximal support vector machine [J]. Decision Support Systems, 2012, 53(3): 395-405

[9] Jak S, Debrabanter J, Lukas L, et al. Weighted least squares support vector machines: robustness and sparse approximation [J]. Neurocomputing, 2002, 48(1-4): 85-105

[10] Dekruif B J, Devries T J A. Pruning error minimization in least squares support vector machines [J]. IEEE Transactions on Neural Networks, 2003, 14(3): 696-702

[11] Hoegaerts L, Suykens J A K, Vandewalle J, et al. A comparison of pruning algorithms for sparse support vector machines[C]// Proc. of the 11th International Conference on Neural Information Processing (ICONIP). New York: Springer Verlag, 2004: 1247-1253

[12] Renjifo C, Barsic D, Carmen C. Improving radial basis function kernel classification through incremental learning and automatic parameter selection [J]. Neurocomputing, 2008, 72: 3-14

[13] 王熙照，崔芳芳，鲁淑霞. 密度加权近似支持向量机[J]. 计算机科学，2012，39(1)：182-184

[14] Golub G H, Van Loan C C. Matrix Computations [M]. Baltimore, USA: The John Hopkins University Press, 1996

[15] Murphy M. UCI-Benchmark Repository of Artificial and Real Data Set [DB/OL]. http://www.ics.uci.edu/, 2005-04-01