基于类间最近邻支持向量信息测度排序的快速分类算法研究

胡正平[1),2)] 张 晔[1)]

[1)](哈尔滨工业大学通信电子工程系图像信息处理研究所,哈尔滨 150001)

[2)](燕山大学通信电子工程系,秦皇岛 066004)

摘 要 提出了基于特征空间中最近邻类间支持向量信息测度排序的快速支持向量机分类算法,对于训练样本首先进行最近邻类间支持向量信息测度升序排列处理;然后根据排序的结果选择最优的训练样本子空间,在选择的样本子空间内采用乘性规则直接求取 Lagrange 因子,而不是传统的二次优化方法;最后加入附加剩余样本进行交叉验证处理,直到算法满足收敛性准则。各种分类实验结果表明,该算法具有非常良好的性能,特别是在训练样本庞大,支持向量数量较多的情况下,能够较大幅度地减少计算复杂度,提高分类速度。

关键词 支持向量机 核函数 乘性规则

中图法分类号:TP391.4 **文献标识码**: A **文章编号**: 1006-8961(2005)06-0758-04

A Fast Support Vector Classification Algorithm Based on the Sort of Nearest Neighborhood Information Measure

HU Zheng-ping[1),2)], ZHANG Ye[1)]

[1)](*Department of Communication and Electronic Engineering, Institute of Image Information Processing, Harbin Institute of Technology, Harbin* 150001)

[2)](*Department of Communication and Electronic Engineering, Yanshan University, Qinhuangdao* 066004)

Abstract To improve the training speed performance of large-scale support vector machine(SVM), a fast algorithm is proposed in this paper by exploiting the geometric distribution of support vector in feature space. A support vector information measure definition based on the nearest inter-classes distance is set up and a sort process is presented. Then a reduced number of sample subspace is extracted for support vector training. In addition, instead of the traditional quadratic programming, multiplicative update is used to solve Lagrange multiplier in optimizing the solution of support vector. The samples of rest are used for cross validating till the algorithm is convergence. Experimental results demonstrate that this method has better performance and has overcome the flaw of standard SVM. This algorithm could greatly reduce the computational load and increase the speed of training, especially in the case of large number of training samples.

Keywords support vector machines, kernel function, multiplicative update

1 引 言

支持向量机(support vector classifier,SVC)因为其优越的性能成为近年研究的热点,它在逼近与分类应用领域取得了极大的成功。支持向量机建立在统计学习理论的 VC 维概念以及结构风险最小原理的基础上,根据有限样本信息在模型复杂度与经验风险之间进行折衷,以获得最好的推广能力。

经典 SVM 训练算法都是把原问题转化为对偶的二次规划问题进行求解。对偶优化求解存在着计算量大,速度慢,参数选择不具有自适应性的问题。近年来人们针对 SVM 方法本身的特点提出了许多改进的算法。例如 SMO(sequential minimal optimization)[1] 算

基金项目:国家自然科学基金项目(60272073)

收稿日期:2004-09-02;**改回日期**:2004-12-28

第一作者简介:胡正平(1970~),男,讲师。1996 年于燕山大学获无线电专业学士学位,1999 年于燕山大学获电路与系统专业硕士学位,现为哈尔滨工业大学图像信息研究所博士研究生。目前的研究方向为统计学习理论、医学图像分析与处理。E-mail: tnpochw@263.net

法、加速分解方法[2]和 SVM-Light 方法[3]。这些方法使用了所有的训练样本,共同的思路就是将原问题分解为若干子问题,按照某种迭代策略,反复求解子问题,直到收敛到原问题的最优解。这些方法本身都是针对训练过程,而不涉及分类过程,因此分类速度没有明显提高。RSVM(reduced support vector machines)方法通过随机选择训练样本子集,减少了训练规模,提高了训练速度[4]。该方法的缺陷在于随机选择训练样本的数目不同而在不同程度上影响所得分类器的速度与性能,算法本身缺乏平稳性。LS-SVM (least square support vector machine,最小二乘支持向量机)将二次规划问题转化为线性方程组的求解,大大简化了计算的复杂度[5]。该方法的不足之处在于:优化参数更小,因此所得支持向量更多而影响分类速度,对于非平衡数据分类准确度下降明显。文献[6]提出利用最近邻规则提高支持向量分类器的精度,并取得了较好的结果;文献[7]提出了利用支持向量几何特征选择训练子集,进而加快训练速度的算法,此方法选择数据点到另一类中心距离作为信息测度准则,这种方法只适合于处理类内数据聚合度大的球形分布训练样本,对于条形分布等情况由于信息测度的误差导致算法的性能下降。针对这个问题,本文提出了一种新的信息测度准则,训练数据点到另一类数据点之间的最小距离被利用建立支持向量信息测度进行排序操作,然后根据排序结果选择合适的训练子集,减少训练规模提高训练速度。学习过程中采用乘性规则[8]解决二次优化问题,该方法不需要像 SVM 那样仔细选择每一步的学习速率,也不需要选择工作子集,它是一种简洁的直接优化方法。实验结果表明该方法在基本上不降低分类精度的情况下,能够较好地解决支持向量机训练的速度问题,特别是在训练样本庞大的情况下,效果显著。

2　算法研究与实现

2.1　算法思想

对于 SVM 分类器而言,训练的目的是找到所有处于类边界上的样本,即支持向量。由于支持向量决定最优超平面的形式,也就决定了分类函数的形式。其准确度决定分类的精度,其数量影响分类器的速度。所以提高 SVM 的速度主要从两方面进行研究:一是提高支持向量训练的速度;二是减少支持向量的数目或者别的方法,提高分类的检测速度。本文的研究属于前者,即在大样本情况下,如何快速找到支持向量集合是本文工作的出发点。支持向量具有位于两类样本相互接近的边界的几何特性,本文正是利用该几何特征提取训练样本属于支持向量的测度,然后根据该测度进行训练子集选择,通过降低训练样本数目降低运算复杂度。

2.2　基于最近邻类间距离的训练子集选择方法

从图 1 可以看出,由于支持向量大都集中在超平面附近并且它们相互之间比较接近,因此可以采取训练数据点到对方类数据点之间的最小距离作为一种可能属于支持向量的测度,即说明不同数据点所提供的最优超平面的信息量不同,可见根据该测度选择合适的训练子集方法是合理的。

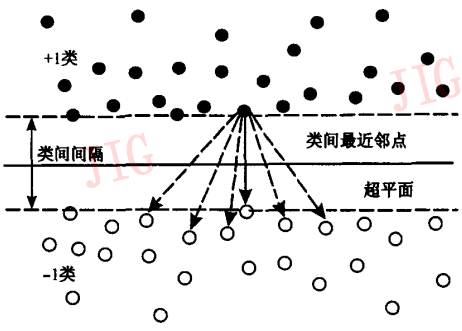

图 1　最近邻类间距离支持向量测度示意图

Fig. 1　The measurement demostration of nearest neighbor support vector

正样本(+1 类)点 x_i 到负样本(-1 类)的最小的距离定义为

$$D_i = D(x_i, y_j) = \min_{y_j \in \{-1\text{类}\}} \|\varphi(x_i) - \varphi(y_j)\|$$

$$= \min_{y_j \in \{-1\text{类}\}} (K(x_i,x_i) + K(y_j,y_j) - 2K(x_i,y_j))^{1/2} \quad (1)$$

这里 $x_i, i = 1,2,\cdots,n_+$ 是 +1 类中的训练样本点,$y_j, j = 1,2,\cdots,n_-$ 是 -1 类中的训练样本点,$K(\cdot)$为核函数。

负样本(-1 类)点 y_j 到正样本(+1 类)的最小距离定义为

$$D_j = \min_{x_i \in \{+1\text{类}\}} (K(x_i,x_i) + K(y_j,y_j) - 2K(x_i,y_j))^{1/2} \quad (2)$$

从式(1)、式(2)可见,距离计算公式是相同的(这里也有别的距离定义)。

使用上面定义的距离测度,训练子集排序过程如下:

(1) 利用式(1)计算每一正样本点 x_i 到所有负样本(-1 类)的最小距离,并记录最小距离对应的

负样本下标。

(2)利用上面计算的距离对所有的正训练样本进行升序操作。

(3)同样地也得到了所有负样本的升序排列。

2.3 支持向量机二次规划目标函数

设 $x_i\{i=1,\cdots,p\}$,这里 x_i 为 N 维特征矢量,p 为训练样本数。$y_j\{+1,-1\}$ 类标记。$g(x)$ 为一非线性映射函数,将 x(N 维)映射到 l 维空间,w 为 l 维矢量,b 为标量。如果训练数据线性不可分,引入松弛变量 $\xi_i(>0)$。一阶范数软间隔支持向量机优化问题对应的拉格朗日函数为

$$\max L=\frac{1}{2}\langle w,w\rangle+C\sum_i\xi_i-\sum_i\beta_i\xi_i-\sum_i\alpha_i[y_i(\langle g(x_i),w\rangle+b)-1+\xi_i] \quad (3)$$

式(3)中第 1 项控制分类器的复杂度(VC 维),第 2 项控制错误分类的经验误差,C 为常数,$\alpha_i,\beta_i\geqslant 0$。对偶表示可以通过求对应于 w,ξ,b 偏导,置 0 并代入式(3)得到。

$$\max L=\sum_i\alpha_i-\frac{1}{2}\sum_{i,j}y_iy_j\alpha_i\alpha_j\langle g(x_i),g(x_j)\rangle \quad (4)$$

引入核函数,SVM 目标函数变为

$$\max L=\sum_i\alpha_i-\frac{1}{2}\sum_{i,j}y_iy_j\alpha_i\alpha_jK(x_i,x_j) \quad (5)$$

其中,$\sum_i\alpha_iy_i=1$ 且 $0\leqslant\alpha_i\leqslant C$。

2.4 相乘性更新规则求解二次规划问题

上面式(5)的非负二次规划优化问题可以归纳为下面的标准形式:

$$\min F(v)=\frac{1}{2}v^{\mathrm{T}}Av+b^{\mathrm{T}}v \quad (6)$$

约束条件为:$v=\{v_i,i=1,\cdots,N\}$,$v_i\geqslant 0$,况且矩阵 A 为对称和半正定的。$F(v)$ 为凸二次优化问题,由于非负的限制条件,可以构造一迭代方法求解全局最优。在式(6)中设 A^+ 和 A^- 表示如下定义的非负矩阵:

$$A^+_{i,j}=\begin{cases}A_{i,j} & A_{i,j}>0\\ 0 & \text{其他}\end{cases}$$
$$A^-_{i,j}=\begin{cases}|A_{i,j}| & A_{i,j}<0\\ 0 & \text{其他}\end{cases} \quad (7)$$

于是有 $A=A^+-A^-$,根据上面定义的非负矩阵,迭代规则定义为

$$v_i\leftarrow v_i\left[\frac{-b_i+\sqrt{b_i^2+4(A^+v)_i(A^-v)_i}}{2(A^+v)_i}\right] \quad (8)$$

通过分析式(8)右式可见整个过程不会脱离非负限制。并且文献[8]中证明了上面迭代规则是收敛的,它可以单调下降到全局最小。

对于式(5)来说,令 $b_i=-1$,$A_{i,j}=y_iy_jK(x_i,x_j)$,则式(5)求解的迭代更新规则为

$$v_i\leftarrow v_i\left[\frac{-1+\sqrt{1+4(A^+v)_i(A^-v)_i}}{2(A^+v)_i}\right] \quad (9)$$

利用式(9)相乘性迭代更新规则求解上面式(5)的二次优化问题,避免了常规 SVM 求解中的矩阵运算,优化速度得到较大提高,算法实现简单明了。

2.5 系统实现以及算法描述

从图 2 的系统实现原理框图可以看出,本文构建的快算法主要包括下列步骤:

(1)选择合适的核函数以及核参数。

(2)根据支持向量建立的信息测度式(1)进行排序处理。

(3)选择合适的样本子集(从小到大)进行训练,得到 SVC 分类器。

(4)交叉验证,判断是否满足要求。如果不满足,增加样本子集,返回步骤 3。直到算法收敛到事先的规定。

(5)返回步骤 1,多次试验找到最佳核参数。

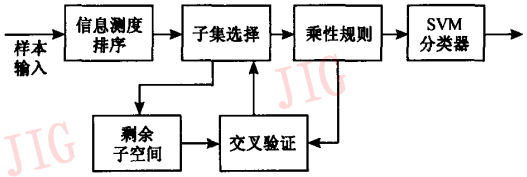

图 2 系统实现原理框图

Fig. 2 The realization frame of system principle

3 实验结果与分析

实验 1 高斯分布样本点分类实验

随机产生 2 类 3 维样本的数据点,每类样本数目 8 000 点,满足多元正态分布,样本点分布参数如表 1 所示。

表 1 合成样本点参数分布

Tab. 1 The parameter distributions of synthetic samples

类别	均值 1	均值 2	均值 3	标准差
1	0.2	1.0	2.5	1
2	1.0	1.8	0.5	1

实验时,每一类别中选择 6 000 点作为训练样

本,2 000 点作为测试样本。同时将本文方法与LS-SVM等算法进行对比,实验中采用高斯核函数(0.5),错误惩罚平衡因子 C 等于 5。对比实验结果如表 2 所示。由于样本的分布情形的不同导致本文方法与中心距离法[7]的差异。

表 2　对比实验结果

Tab.2　Comparison results of experiments

方法	分类正确率(%)	支持向量数目	训练时间(s)
LS-SVM	81.43	591	1 279
SMO	81.76	497	3 672
文献[7]方法	81.62	403	1 028
本文方法	82.90	405	1 007

实验 2　MNIST 手写数字识别实验

利用网上下载的 MNIST 手写数字数据库进行仿真实验,从含有 60 000 个训练样本的库中选择了 3 000 个训练样本("3"和"5"),从含有 10 000 个测试样本的库中顺序选择 500 个作为测试样本,采用量化 PCA(principal component analysis)投影矢量作为训练特征,对比实验结果如表 3 所示。

表 3　对比实验结果

Tab.3　Comparison results of experiments

方法	分类正确率(%)	支持向量数目	训练时间(s)
LS-SVM	98.59	101	745
SMO	98.77	87	2 016
文献[7]方法	98.74	85	703
本文方法	98.73	83	691

从上面的对比实验可以看出,本文方法无论对于合成数据还是真实数据在保持分类精度的情况下,通过减少训练样本大大减少了计算量,加快了算法的速度。

对于实验 1,最小近邻类间距离支持向量信息测度与中心距离法[7]相比,具有更加广泛的使用范围,而且这种度量所得到信息更加准确,进而得到更加准确的有效样本子空间,所以算法性能略有改进。

对于实验 2,采用最近邻类间支持向量信息测度排序的方法选择合适的训练样本子空间一方面大大降低了 SVM 训练中有效样本的数目,大大降低了核矩阵的大小,使得 SVM 优化的难度得到降低,进而使得实际的计算更加接近凸二次规划理想的极值点,因而算法性能得到提高。

4　结　论

本文利用支持向量分布的几何特征以及新的乘性法则提出了一种快速训练算法,该方法根据样本数据点提供的支持向量信息量的不同,选择有效的训练子集作为训练样本,由于训练样本数目的减少,使得计算量减少而提高训练速度。为克服传统二次优化问题的不足,采用乘性规则迭代求解二次规划的优化问题,思路简单明了并且进一步降低了算法的运算量。虽然算法的运算量得到降低,但是由于分类函数的支持向量数目变化不大,使得整个算法的分类精度基本上没有损失。针对合成数据以及真实数据的实验结果表明本文提出的方法是合理有效的。同时值得指出的是本文提出的算法通过选择相同的训练样本子集可以解决非平衡样本的分类问题;也可以通过选择合适的训练样本子集解决训练样本中出现出格点(outlier)的分类问题,提高算法在有噪环境下的鲁棒性。这将是下一步研究的问题。

参考文献(References)

1　Platt J C. Fast algorithm for training support vector machines using Sequential minimimal optimization [A]. In: Advance in kernel methods-support vector learning[M], Scholkopf B, Burges C J C, Smola A J, editors, Combridge, MA, USA: MIT Press, 1999: 185 ~ 208.

2　Hu W J, Song Q. An accelerated decomposition algorithm for robust support vector machines [J]. IEEE Transactions on Circuits and Systems-II: Express Briefs, 2004, **51**(5):234 ~ 240.

3　Joachims T. Making large-scale support vector machine learning practical[A]. In: Advance in kernel methods-support vector learning [M], Scholkopf B, Burges C J C, Smola A J, editors, Cambridge, MA, USA: MIT Press, 1999:169 ~ 184.

4　Lin Kuan-Ming, Lin Chih-Jen. A study of reduced support vector machines [J]. IEEE Transactions on Neural Networks, 2003, **14**(6):1449 ~ 1459.

5　Suykens J A K, Vandewalle J. Least squares support vector machine classifiers[J]. Neural Processing Letters, 1999, **9**(3): 293 ~ 300.

6　Li Hong-Lian, Wang Chun-hua, Yuan Bao-zong. An improved SVM: NN-SVM [J]. Chinese Journal of Computers, 2003, **26** (8): 1015 ~ 1020. [李红莲,王春花,袁保宗. 一种改进的支持向量机 NN-SVM[J]. 计算机学报, 2003,**26**(8):1015 ~ 1020.]

7　Jiao Licheng, Zhang L, Zhou W D. Pre-extracting support vectors for support vector machine[J]. Acta Electronica Sinica, 2001,**29**(3): 383 ~ 386. [焦李成,张莉,周伟达. 支持矢量预选取的中心距离比值法[J],电子学报,2001,**29**(3):383 ~ 386.]

8　Sha F, Saul L K, Lee D D. Multiplicative updates for nonnegative quadratic programming in support vector machines[EB/OL]. http://www.cs.cmu.edu/groups/NIPS/NIPS2002/NIPS2002 preproceedings/papers/AA71.html, 2002-07-15.