中华博士园地

这是本刊特为海内外正在就读和学成立业的博士、博士后青年学者们开辟的一片科普园地. 深学浅著是一门德识、慧学、素质修养的学问. 你们的新知识、新调研、新观察、新目光、新展望,能够用尽可能深入浅出、通俗流畅的语言,汇报给祖国人民、家乡父老子弟乡亲们吗? 中华博士园地,乃耕耘忠孝之地,科教兴国、民族昌盛之地. 要用慈母听得懂的语言,写出你们的心声!

中图法分类号: TP391.4 文章编号: 1006-8961(2002)06-0618-06

支持向量机的研究现状

柳回春 马树元
(北京理工大学机械工程及自动化学院, 北京 100081)

0 理论背景

基于数据的机器学习是现代智能技术中十分重要的一个方面. 机器学习的目的是根据给定的训练样本求对某系统输入输出之间依赖关系的估计,使它能够对未知输出作出尽可能准确的预测. 机器学习一般地可以表示为[1]:变量 y 与 x 存在一定的未知依赖关系,即遵循某一未知的联合概率 $F(x,y)$,(x 和 y 之间的确定性关系可以看作是其特例),机器学习问题就是根据 l 个独立同分布观测样本

$$(x_1,y_1),(x_2,y_2),\cdots,(x_l,y_l) \tag{1}$$

在一组函数 $\{f(x,w)\}$ 中,求一个最优的函数 $f(x,w_0)$,用以对 x 和 y 之间的依赖关系进行估计,使期望风险最小,即:

$$\min R(w)=\int L(y,f(x,w))\mathrm{d}F(x,y) \tag{2}$$

其中,预测函数集 $\{f(x,w)\}$ 可以表示任何函数集合,w 为函数的广义参数,$L(y,f(x,w))$ 为用 $f(x,w)$ 对 y 进行预测而造成的损失,不同类型的学习问题有不同形式的损失函数.

统计模式识别的传统方法都是在样本数目足够多的前提下进行研究的,所提出的各种方法只有在样本数趋于无穷大时其性能才有理论上的保证. 而在实际的应用中,样本数目通常是有限的,于是,人们采用了所谓的经验风险最小化(Empirical Risk Minimization, ERM)准则,即用样本定义经验风险

$$R_{\mathrm{emp}}(w)=\frac{1}{l}\sum_{i=1}^{l}L(y_i,f(x_i,w)) \tag{3}$$

机器学习就是要设计学习算法,使 $R_{\mathrm{emp}}(w)$ 最小化,作为对式(2)的估计. 多年来,人们将大部分注意力集中到如何更好地最小化经验风险上,但是,从期望风险最小化到经验风险最小化并没有可靠的理

柳回春 1975 年生,博士生. 主要研究方向为模式识别,图象处理,支持向量机,神经网络等.

论依据[2]. 首先 $R_{emp}(w)$和 $R(w)$都是 w 的函数,概率论中的大数定理只说明了在一定条件下,当样本数趋于无穷大时,$R_{emp}(w)$将在概率意义上趋近于 $R(w)$,并没有保证使 $R_{emp}(w)$最小的 w^* 与使 $R(w)$最小的 w^{**} 是同一个点,更不能保证 $R_{emp}(w^*)$能够趋近于 $R(w^{**})$;其次,即使有办法使这些条件在样本数无穷大时得到保证,也无法认定在这些前提下得到的经验风险最小化方法在样本数有限时仍能得到好的结果. Vapnik 等人早在 20 世纪 60 年代就开始研究有限样本情况下的机器学习问题,但直到 90 年代以前,也没有提出能够将其理论付诸实现的较好办法,直到 90 年代中,有限样本情况下的机器学习理论研究才逐渐成熟起来,形成了一个较完善的理论体系——统计学习理论(Statistical learning theory,简称 SLT)[3]. 它能将很多现有方法纳入其中,可望解决许多原来难以解决的问题,如学习能力和推广能力的统一. 1992 年~1995 年,Vapnik 等人又在统计学习理论的基础上,发展出了一种新的通用的学习方法——支持向量机(Support vector machine,简称 SVM),其在解决小样本、非线性及高维模式识别问题中表现出许多特有的优势,并且能够推广到函数逼近和概率密度估计等其他机器学习问题中. 目前,SVM 算法在模式识别、回归估计、概率密度函数估计等方面都有应用. 例如,在模式识别方面,对于手写体识别[4]、语音识别、人脸识别、文本分类等问题,SVM 算法在精度上已经超过传统的学习算法或与之不相上下. 统计学习理论和支持向量机已经成为国际上机器学习领域新的研究热点.

1　"VC 维"和"推广性的界"

为了研究学习过程一致收敛的速度和推广性,统计学习理论定义了一系列有关函数集学习性能的指标,其中最重要的是 Vapnik 和 Chervonenkis 提出的 VC 维[3](VC 就是取 Vapnik 和 Chervonenkis 名字的首字而成). VC 维是统计学习理论中的一个核心概念,它是目前为止,对函数集学习性能最好的描述指标. 一个函数集的 VC 维可以理解为由其分类函数能正确给以所有可能二值标识的最大训练样本数,也就是说,如果存在 h 个样本的样本集能够被函数集打散,而不存在有 $h+1$ 个样本的样本集能够被函数集打散,则函数集的 VC 维就是 h. 如果对于任意的样本数,总能找到一个样本集能够被这个函数集打散,则函数集的 VC 维就是无穷大. 应当指出,这里是存在 h 个样本的样本集能够被函数集打散,不是指任意 h 个样本的样本集能够被函数集打散. 是函数集的 VC 维(而不是其自由参数个数)影响了学习机器的推广性能. 这给我们克服所谓的"维数灾难"创造了一个很好的机会:以一个包含很多参数但却有较小的 VC 维的函数集为基础实现较好的推广性.

统计学习理论系统地研究了各种类型函数集的经验风险和实际风险之间的关系,即推广性的界[3]. 关于两类分类问题有如下结论:对指示函数集中的所有函数(包括使经验风险最小化的函数),经验风险 $R_{emp}(w)$和实际风险 $R(w)$之间至少以概率 $1-\eta$ 满足下列关系:

$$R(w) \leq R_{emp}(w) + \varphi(l/h) \tag{4}$$

其中,h 是函数集的 VC 维,l 是样本数,η 是满足 $0<\eta<1$ 的参数,$\varphi(l/h)$称作置信范围(confidence interval),也有人把它叫做 VC 信任(VC confidence). 置信范围不但受置信水平 $1-\eta$ 的影响,而且更是函数集的 VC 维和训练样本数目的函数,且随着它们比值的增加而单调减少. 因为这个界限反映了根据经验风险最小化原则得到的机器学习的推广能力,所以称它为推广性的界.

可以看出,置信界限反映了真实风险和经验风险差值的上确界,它和学习机器的 VC 维 h 及训练样本数 l 有关. 因此,要想得到期望风险最小值,除了控制经验风险最小外,还要控制函数集的置信界限,而置信界限随着函数集 VC 维的增长而增大. 在有限训练样本下,学习机器的复杂性越高,VC 维越高,则置信界限越大,也就会导致真实风险与经验风险之间可能的差别越大. 在神经网络中存在一个过学习现象,其原因首先是学习样本不充分,其次是学习机器的复杂性过高,导致它的 VC 维过大,以至于在经验风险最小的情况下,函数集的置信界限很大,所以其推广能力仍然很差.

经验风险最小化原则在样本有限时,不尽合理,一般需要同时最小化经验风险和置信界限. 统计学习理论提出了一种新的策略,即把函数集构造为一个函数子集序列,使各个子集按照 VC 维的大小(亦即 φ 的大小)排列;在每个子集中寻找最小经验风险,在子集间折衷考虑经验风险和置信界限,取得实际风险的最小. 这种思想称为结构风险最小化(Structural Risk Minimization,SRM),即 SRM 准则.

2 支持向量机

支持向量机理论是从线性可分情况下的最优分类面发展而来的. 基本思想可用图 1 的两维情况来说明.

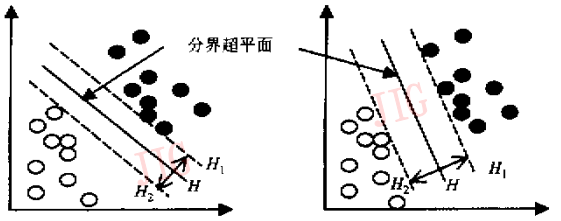

图 1 SVM 分界面的比较

图中实心点和空心点分别代表两类样本,H 为分类线,H_1、H_2 分别为过各类中离分类线最近的样本且平行于分类线的直线,它们之间的距离叫做分类间隔(margin). 所谓最优分类线就是要求分类线不但能将两类正确分开(训练错误率为 0),而且使分类间隔最大. 分类线方程为 $\boldsymbol{x}\cdot\boldsymbol{w}+b=0$,可以对它进行归一化,使得对线性可分的样本集 $(\boldsymbol{x}_i,y_i)$,$i=1,\cdots,n$,$\boldsymbol{x}\in\mathbf{R}^{(d)}$,$y\in\{+1,-1\}$,满足

$$y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1\geq 0,i=1,\cdots,n \tag{5}$$

此时分类间隔等于 $2/\|\boldsymbol{w}\|$,使间隔最大等价于使 $\|\boldsymbol{w}\|^2/2$ 最小. 满足式(5),且使 $\|\boldsymbol{w}\|^2/2$ 最小的分类面就叫做最优分类面,H_1、H_2 上的训练样本点就称作支持向量. 利用 Lagrange 优化方法可以把上述最优分类面问题转化为其对偶问题

$$\begin{aligned}\max_{\alpha}W(\alpha)&=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j\boldsymbol{x}_i\cdot\boldsymbol{x}_j\\&=\Lambda\cdot\boldsymbol{I}-\frac{1}{2}\Lambda\cdot\boldsymbol{D}\cdot\Lambda\end{aligned} \tag{6}$$

满足约束

$$\sum_{i=1}^{l}y_i\alpha_i=0\quad\alpha_i\geq 0\quad i=1,2,\cdots,l$$

$$\alpha=\{\alpha_1,\cdots,\alpha_l\} \tag{7}$$

其中,$\Lambda=(\alpha_1,\alpha_2,\cdots,\alpha_l)$,$\boldsymbol{I}=(1,\cdots,1)$,$\boldsymbol{D}$ 是 $l\times l$ 的对称矩阵,各个单元为

$$\boldsymbol{D}_{ij}=y_i y_j\boldsymbol{x}_i\cdot\boldsymbol{x}_j \tag{8}$$

α_i 为原问题中,与每个约束条件式(7)对应的 Lagrange 乘子. 这是一个不等式约束下,二次函数寻优的问题,存在唯一解. 容易证明,解中将只有一部分(通常是少部分)α_i 不为零,其对应的样本就是支持向量. 解上述问题后得到的最优分类函数是

$$\begin{aligned}f(\boldsymbol{x})&=\operatorname{sgn}\{(\boldsymbol{w}\cdot\boldsymbol{x})+b\}\\&=\operatorname{sgn}\Big\{\sum_{i=1}^{n}\alpha_i^*y_i(\boldsymbol{x}_i\cdot\boldsymbol{x})+b^*\Big\}\end{aligned} \tag{9}$$

式中的求和实际上只对支持向量进行. α_i^* 为 α_i 的最优解,b^* 是分类阈值,可以用任一个支持向量(满足式(5)中的等号)求得,或通过两类中任意一对支持向量取中值求得.

在线性不可分的情况下,引入非负松弛变量集合 $\xi=(\xi_1,\xi_2,\cdots,\xi_l)$,这样将式(5)的线性约束条件转化为:

$$y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)\geq 1-\xi_i,i=1,\cdots,l \tag{10}$$

当样本 $\boldsymbol{x}_i$ 满足不等式(5)时,ξ_i 为零,否则 $\xi_i>0$,表示此样本为造成线性不可分的点. 利用 Lagrange 乘子法及对偶原理对式(10)进行处理,可得到线性不可分条件下的对偶问题

$$\begin{aligned}\max_{\alpha}W(\alpha)&=\sum_{i=1}^{l}\alpha_i-\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j x_i\cdot x_j\\&=\Lambda\cdot\boldsymbol{I}-\frac{1}{2}\Lambda\cdot\boldsymbol{D}\cdot\Lambda\end{aligned} \tag{11}$$

满足约束

$$\sum_{i=1}^{l}y_i\alpha_i=0,0\leq\alpha_i\leq C,i=1,2,\cdots,l \tag{12}$$

其中,C 为大于零的常数.

在对这类约束优化问题的求解和分析中,库恩-塔克条件(Karush-Kuhn-Tucker,KKT)[5]起着重要的作用,KKT 条件为

$$\begin{cases}\text{若 }\alpha_i=0,\text{则 }\xi_i=0,y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)\geq 1;\\ \text{若 }0<\alpha_i<C,\text{则 }\xi_i=0,y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)=1;\\ \text{若 }\alpha_i=C,\text{则 }\xi_i\geq 0,y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i+b)\leq 1\end{cases} \tag{13}$$

KKT 条件是最优解应满足的充要条件,所以目前提出的一些算法几乎都是以是否违反 KKT 条件作为迭代策略的准则.

以上都是在线性分界超平面的基础上进行的讨论,在很多问题中需要将其推广到非线性分类超平面中. SVM 的非线性特性可以这样来实现,把输入样本 $\boldsymbol{x}$ 映射到高维特征空间(可能是无穷维)$\boldsymbol{H}$ 中,并在 $\boldsymbol{H}$ 中使用线性分类器来完成分类,即将 $\boldsymbol{x}$ 做变换 $\Phi:\mathbf{R}^{(d)}\rightarrow\boldsymbol{H}$,前面的分析同样适用. 当在特征空间 $\boldsymbol{H}$ 中构造最优超平面时,训练算法仅使用空间中的点积,即仅仅使用 $\Phi(\boldsymbol{x}_i)\cdot\Phi(\boldsymbol{x}_j)$,而没有单独的 $\Phi(\boldsymbol{x}_i)$ 出现. 因此,如果能够找到一个函数 K 使得 $K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\Phi(\boldsymbol{x}_i)\cdot\Phi(\boldsymbol{x}_j)$,那么,在高维空间实际上

只需进行内积运算,而这种内积运算是可以用原空间中的函数来实现的,甚至没有必要知道 Φ 的形式.根据泛函的有关理论,只要一种核函数 $K(\boldsymbol{x}_i,\boldsymbol{x}_j)$ 满足 Mercer 条件,它就对应某一变换空间中的内积.常用的核函数有多项式、径向基函数、Sigmoid 函数、样条(spline)函数核和 Fourier 核等.因此,在最优分类面中,采用适当的内积函数 $K(\boldsymbol{x}_i,\boldsymbol{x}_j)$ 就可以实现某一非线性变换后的线性分类,而计算复杂度却没有增加.这一特点为算法可能导致的"维数灾难"问题提供了解决方法:在构造判别函数时,不是对输入空间的样本作非线性变换,而后再在特征空间中求解;而是先在输入空间比较向量(例如求点积或是某种距离),然后再对结果作非线性变换,这样,大的工作量将在输入空间而不是在高维特征空间中完成.

3　支持向量机目前的研究状况

由于支持向量机坚实的理论基础和它在很多领域表现出的良好的推广性能,目前,国际上正在广泛开展对支持向量机方法的研究.许多关于 SVM 方法的研究,包括算法本身的改进和算法的实际应用,都陆续提了出来.以下是其中主要的研究热点.

3.1　改进训练算法

由于 SVM 对偶问题的求解过程相当于解一个线性约束的二项规划问题(QP),需要计算和存储核函数矩阵,其大小与训练样本数的平方相关,因此,随着样本数目的增多,所需要的内存也就增大,例如,当样本数目超过4 000时,存储核函数矩阵需要多达 128M 内存;其次,SVM 在二次型寻优过程中要进行大量的矩阵运算,多数情况下,寻优算法是占用算法时间的主要部分.通常,训练算法改进的思路是把要求解的问题分成许多子问题,然后通过反复求解子问题来求得最终的解,方法有以下几种:

(1) 块处理算法[6](chunking algorithm)　它的思想是将样本集分成工作样本集和测试样本集,每次对工作样本集利用二项规划求得最优解,剔除其中的非支持向量,并用训练结果对剩余样本进行检验,将不符合训练结果(一般是指违反 KKT 条件)的样本(或其中的一部分)与本次结果的支持向量合并,成为一个新的工作样本集,然后重新训练.如此重复下去,直到获得最优结果.其依据是去掉 Lagrange 乘子等于零的训练样本不会影响原问题的解.块算法的一个前提是:支持向量的数目比较少.然而如果支持向量的数目本身就比较多,那么随着训练迭代次数的增加,工作样本数也越来越大,就会导致算法无法实施.

(2) 固定工作样本集算法　它使样本数目固定在足以包含所有的支持向量,且算法速度在计算机可以容忍的限度内.迭代过程中只是将剩余样本中部分"情况最糟的样本"与工作样本集中的样本进行等量交换.即使支持向量的个数超过工作样本集的大小,也不改变工作样本集的规模.文献[7]介绍了一种具体的算法,将样本集分为 B 和 N 两个集合,集合 B 作为子问题的工作样本集进行 SVM 训练,集合 N 中所有样本的 Lagrange 乘子均置为零.显然,如果把集合 B 中,对应 Lagrange 乘子为零的样本 i(即 $\alpha_i=0, i\in B$)与集合 N 中的样本 j(即 $\alpha_j=0, j\in N$)交换,不会改变子问题与原问题的可行性(即仍旧满足约束条件).于是可以按照以下步骤迭代求解:①选择集合 B,构造子问题;②求子问题最优解 $\alpha_i, i\in B$ 及 b,并置 $\alpha_j=0, j\in N$;③计算 $g(\boldsymbol{x}_j)y_j, j\in N$,找出其中 $g(\boldsymbol{x}_j)y_j<1$ 的样本 j,(其中 $g(\boldsymbol{x}_j)=\sum_{p=1}^{l}\alpha_p y\quad K(\boldsymbol{x}_j,\boldsymbol{x}_p)+b$),与 B 中满足 $\alpha_i=0$的样本 i 交换,构成新的子问题.需要指出:如果集合 B 不足以包括所有的支持向量,该算法没有提出改变 B 的大小的策略,那么有可能得不到结果.

(3) SMO 算法[8]　SMO 是固定工作样本集算法的一个极端情况,其工作样本数目为 2.需要两个样本,是因为等式线性约束的存在使得同时至少有两个 Lagrange 乘子发生变化.由于只有两个变量,而且应用等式约束可以将其中一个用另一个表示出来,所以迭代过程中,每一步子问题的最优解都可以直接用解析的方法求出来,这样,算法就避开了复杂的数值求解优化问题的过程;此外,算法还设计了一个两层嵌套循环,分别选择进入工作样本集的样本,这种启发式策略大大加快了算法的收敛速度.标准样本集的实验结果证明,SMO 在速度方面表现出良好性能.

3.2　提高测试速度

从式(9)中可以看到,SVM 判决函数的计算量和支持向量的数目成正比.对于大训练集合,其支持向量的数目会达到几千个,这就使 SVM 对实验样本的测试判决速度变慢,因此,提高 SVM 的测试速度是另一个研究热点.

在式(9)中,对所有 N_s 个支持向量求和,计算量很大,如果可以减少求和的数目,使其达到 $M(M \ll N_s)$个,则可以大大提高速度. 判决函数$d(\boldsymbol{x})$的近似式如下:

$$d(\boldsymbol{x}) = \sum_{i=1}^{M} \gamma_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b \tag{14}$$

据此,Osuna 提出 3 种方案[9]:

(1) 对 $d(\boldsymbol{x})$在特征空间中进行拟合,得到一个近似的判决函数 $d'(\boldsymbol{x})$:

$$d'(\boldsymbol{x}) = \sum_{i=1}^{l'} \gamma_i K(\boldsymbol{x}, \boldsymbol{z}_i) + b \tag{15}$$

其中,$l' \ll N_s$,$\boldsymbol{z}_i$ 是特征空间中的合成向量,不一定是样本点,γ_i 是权重.

(2) 判决函数 $d(\boldsymbol{x})$在特征空间中近似为

$$d'(\boldsymbol{x}) = \sum_{i=1}^{l'} \gamma_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b \tag{16}$$

其中,$l' \ll N_s$,γ_i 是与支持向量 $\boldsymbol{x}_i$ 对应的权重.

(3) 对原问题重新定义,得到新的判决函数为

$$d(\boldsymbol{x}) = \sum_{i=1}^{l'} \gamma_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b \tag{17}$$

其中,$l' \ll N_s$,$\boldsymbol{x}_i$ 是某些训练样本点,但不一定是支持向量,γ_i 是权重,y_i 是样本点的类别标号.

其中,第 1 个方案已经被 Burges 论证,并实现[10],其基本思想是:在 SVM 的高维特征空间中,运用比原来少得多的精简集合(Reduced Set,RS)向量来拟合原来所有 SV 所构成的分界超平面,可以在损失极少信息的基础上,大大提高测试速度.

3.3 核函数的构造、改进以及相应参数的调整

SVM 的核函数有多项式核函数、径向基函数等. 尽管一些实验结果表明核函数的具体形式对分类效果的影响不大,但是核函数的形式以及其参数的确决定了分类器的类型和复杂程度,它显然应该作为控制分类器性能的手段.

最常用的模型参数选择方法是最小化"留一法(LOO)"错误率. 其步骤是:

(1) 对于 $j=1,2,\cdots,l$ 个训练样本,每次取出其中一个样本 i,而对其余的 $l-1$ 个训练样本求解式(11);

(2) 对 $l-1$ 个样本的训练结果,采用式(9)对第 i 个样本进行测试;

(3) 反复重复上述步骤.

LOO 的错误率是:

$$L(\boldsymbol{x}_1, y_1, \cdots, \boldsymbol{x}_l, y_l) = 1 - \sum_{j=1}^{l} \frac{\left| \operatorname{sgn}\left\{ \sum_{i=1}^{l-1} \alpha_i y_i K(\boldsymbol{x}_i \cdot \boldsymbol{x}_j) + b^* \right\} - y_i \right|}{2l} \tag{18}$$

可以看出,用 LOO 估计错误率的方法来调整参数需要的训练量很大,因此有人提出用估计错误率上限的方法来调整 SVM 的参数. 文献[11]中提出了几种估计错误率上限的方法,其中有逐个确认估计、留一法上限估计、用支持向量数与训练样本数的比值估计上限、Jaakkola-Haussler 上限、Opper-Winther 上限、Radius-margin 上限、Span 上限.

同时,还实现了根据错误率上限的估计来调整 SVM 参数的算法,其具体思路是:初始化 SVM 的核函数参数,求解式(11),然后利用求得的结果估计错误率上限,再利用对错误率上限的梯度下降法调整核函数参数,反复执行上述步骤,直到得到最小的错误率上限. Chapelle 等人于 2000 年使用基于矩阵的二项规划方法实现了利用 LOO 上限来调整 SVM 参数的方法[12]. 应当指出的是,即使对于只有几千个样本的中型问题,也很难在内存中容下整个核函数矩阵,并进行相应的矩阵操作,这种方法有时会带来问题,因此,应当找到一种合适的迭代策略来解决这个问题.

3.4 利用 SVM 解决多分类的问题

由于支持向量机是针对两分类问题提出的,因此,存在一个如何将其推广到多分类问题上,特别是对极大类别分类的问题上. 目前有以下几种方案:

(1) 一对多(One class Versus all Others,OVO) 其基本想法是把某一种类别的样本当作一个类别,剩余其他类别的样本当作另一个类别,这样就变成了一个两分类问题. 这种分类方案的缺点是训练样本数目大,训练困难.

(2) 一对一(One class Versus Another class,OVA)[13] 其做法是在多类别中,任意抽取两类进行两两配对,转化成两类问题进行训练学习. 识别时,对所构成的多个 SVM 进行综合判断,一般可采用投票方式来完成多类识别. 这种分类方案存在一个明显的缺点,就是子分类器过多,测试时需要对每两类都进行比较,导致测试速度慢. DAG 方案[14]在训练阶段也是采用一对一的配对训练方式,它的优点在于,对训练结果的推广性进行了分析,另外,它的测试速度也比一对一的方案要快.

(3) SVM 决策树(Decision Tree Method,

DTM)[15]　它通常和二叉决策树结合起来,构成多类别的识别器.这种方法的缺点是如果在某个节点上发生了分类错误,将会把错误延续下去,该节点后续下一级节点上的分类就失去了意义.

(4) Multi-Class SVM　它直接在目标函数上进行改进,建立 k 分类支持向量机[16].由二分类支持向量机进行推广得到目标函数:

$$\min_{\boldsymbol{w},\xi,b}\varphi(\boldsymbol{w},\xi)=\frac{1}{2}\sum_{m=1}^{k}\|\boldsymbol{w}_m\|^2+C\sum_{i=1}^{l}\sum_{m\neq y_i}\xi_i^m \quad (19)$$

满足约束:

$$\boldsymbol{w}_{y_i}^{\mathrm{T}}\varphi(\boldsymbol{x}_i)+b_{y_i}\geq\boldsymbol{w}_m^{\mathrm{T}}\varphi(\boldsymbol{x}_i)+b_m+2-\xi_i^m \quad (20)$$

$$\xi_i^m\geq 0,i=1,\cdots,l\quad m\in\{1,\cdots,k\}\backslash y_i \quad (21)$$

这样目标决策函数为

$$f(\boldsymbol{x})=\underset{n=1,\cdots,k}{\operatorname{argmax}}\left[(\boldsymbol{w}_n^{\mathrm{T}}\varphi(\boldsymbol{x})+b_n\right] \quad (22)$$

利用 Lagrange 优化方法同样可以把上述最优分类面问题转化为其对偶问题,得到的函数变量数为 kl.这种方法因为变量数目过多,所以只在小型问题的求解中才能使用.

另外,SVM 的研究点还有如何把 SVM 中的核函数内积思想应用到其他方面. Schölkopf 等学者首先把核函数的概念引入到 PCA 中[17],用核函数实现非线性主元分析,它是传统主元分析(Principal Component Analysis,PCA)方法的推广.对经典的 SVM 算法的改进也是一个研究点. Schölkopf 提出了一类新的支持向量算法[18],它运用参数 γ 来控制支持向量的数目及误差,使新的 γ-SVR 回归算法更加实用,并把 γ-SVR 的思想运用到了 γ-SV 的分类问题中.他还提出了 SVR 的一种新算法,从 ε-SVR 到 γ-SVR,具有更好的适应性及鲁棒性.此外,有人研究如何处理样本集合中的一些特殊点或远点,尤其是样本集中的一些离散点,进一步提高 SVM 识别器的泛化能力.

4　结束语

总之,支持向量机是一种基于统计学习理论的新的机器学习方法,它可以用于模式识别、回归分析和函数拟合等问题中,并且有一套坚实的理论基础.遗憾的是,虽然支持向量机在理论上有很突出的优势,但与其理论研究相比,应用研究尚相对比较滞后,目前只有比较有限的实验研究报道,且多属仿真和对比实验.支持向量机的应用研究应该是一个大有作为的方向.我们相信支持向量机是一个值得大力研究的领域,对它的研究将对机器学习等学科领域产生重要影响.

参 考 文 献

1　张学工. 关于统计学习理论与支持向量机[J]. 自动化学报, 2000,26(1):32～43.

2　边肇祺,张学工等. 模式识别[M]. 北京:清华大学出版社, 2000:286～294.

3　Vapnik V N. The nature of statistical learning theory[M]. NY: Springer-Verlag, 1995.

4　Bottou L, Cortes C, Denker J *et al*. Comparison of classifier methods: A case study in handwritten digit recognition[A]. In: 12th IAPR[C], IEEE Computer Society Press, Los Alamos, California, 1994:77～83.

5　Fletcher R. Practical methods of optimization[M](2nd edition). New York:John Wiley and Sons Inc, 1987.

6　Boser Bernhard E, Guyon Isabelle M, Vapnik Vladimir N. A training algorithm for optimal margin classifiers [EB/OL]. http://www.cs.rhul.ac.uk/colt/nips2000/vincent.ps.

7　Edgar Osuna *et al*. Training support vector machines:An application to face detection [EB/OL]. http://www.citeseer.nj.nec.com/osuna97training.html.

8　Platt J. Sequential minimal optimization: A fast algorithm for training support vector machines[A]. In: Advances in Kernel Methods-Support Vector learning[C]. Massachusetts: The MIT Press, 1999:185～208.

9　Osuna Edgar, Girosi Federico. Reducing the run-time complexity of support Vector Machines [EB/OL]. http://www.citeseer.nj.nec.com/osuna98reducing.html.

10　Burges C. Simplified support vector decision rules[A]. In: Proceedings of the 13th International Conference on Machine Learning[C], CA:Morgan Kaufmann, 1996:71～77.

11　Oliver Chapelle, Vladimir Vapnik. Choosing multiple parameters for support vector machines[EB/OL]. http://www.citeseer.nj.nec.com/chapelle01choosing.html.

12　Chapelle O, Vapnik V, Bonsquet O *et al*. Choosing kernel parameters for support vector machines[EB/OL]. http://www.citeseer.nj.nec.com/chapelle01choosing.html.

13　Krebel Ulrich H G. Pairwise classification and support vector machines[A]. In: Schölkopf Bernhard(edi.). Advances in Kernel Methods:Support Vector Learning[C]. Massachusetts, The MIT Press, 1999:255～268.

14　Boser Bernhard E, Guyon Isabelle M, Vapnik Vladimir N. A training for optimal margin classifiers[A]. Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT'92)[C], Pittsburgh:ACM, 1992,5:144～152.

15　Bennett K, Blue J. A support vector machine approach to decision trees[R]. Rensselaer Polytechnic Institute, Troy, NY:R.P.I Math Report, 1997:97～100.

16　Vapnik V. Statistical learning theory[M]. New York: Wiley, 1998.

17　Schölkopf B, Smola A, Müller K. Kernel principal component analysis[EB/OL]. http://www.citeseer.nj.nec.com/25296.html.

18　Schölkopf B, Mika S, Smola A *et al*. Kernel PCA pattern reconstruction via approximate pre-images[A]. In: L. Niklasson and M. Boyoudian den and T. Ziemke (eds.). Perspectives in Neural Computing[C]. Berlin:springer Verlag, 1998.