文章编号:1000-1506(2003)02-0001-08

基 于 核 函 数 的 学 习 算 法

田 盛 丰
(北方交通大学 计算机与信息技术学院,北京 100044)

摘　要:论述基于核函数的方法,包括支持向量机的分类、聚类与回归算法,基于核的 Fisher 判别分析、感知机和主成分分析,介绍学习算法、简化算法和多类分类等研究课题,及基于核函数方法的应用.

关键词:支持向量机;基于核函数的方法;机器学习
中图分类号:O234;TP18　　文献标识码:A

Kernel-Based Learning Algorithms

TIAN Sheng-feng
(School of Computer and Information Technology, Northern Jiaotong University, Beijing 100044, China)

Abstract: This paper describes the kernel-based methods including the classification, clustering and regression of support vector machines, Fisher discriminant analysis, perceptron and principal component analysis based on kernel. The research topics including learning algorithms, simplification algorithms and multi-classese classification, and the applications of kernel-based methods are discussed.

Key words: support vector machine; kernel-based method; machine learning

支持向量机是一种学习机制,可用于模式识别和回归估计[1,2].支持向量机的概念是前苏联学者 Vapnik 等人在 1974 年提出的,直到最近几年才受到重视,并开始成为人工智能界的一个研究热点.该项研究属于机器学习、模式识别和人工神经网络等多个学科,由于它与这些学科现有的理论和方法相比,有明显的优越性,因此有重大的潜在应用价值.支持向量机的优越性表现在:①支持向量机是根据结构风险最小化原则,尽量提高学习机的泛化能力,即由有限的训练样本得到的小的误差能够保证对独立的测试集仍保持小的误差.②支持向量机算法是一个凸优化问题,因此局部最优解一定是全局最优解.这些特点是其它学习算法,如人工神经网络学习算法所不及的.

支持向量机的理论基础是统计学习理论[3].传统的学习机器采用经验风险最小化准则,即使训练集的误差尽量小.缺点是在训练集有限时不能保证小的期望风险,即训练集的误差小并不能保证在测试集上误差小,这就是泛化能力问题.统计学习理论就是研究有限样本的学习问题,采用结构风险最小化准则,既考虑减小训练集的误差,也兼顾减小学习机的复杂性,称为 VC 维(VC 维的概念是由 Vapnik 和 Cheyvonenkis 提出的,定义为能被一个函数集合分开最大数目的训练例子数),从而保证好的泛化能力.

支持向量机分类器的基本原理是使用一个非线性变换将一个线性不可分的空间映射到一个高维的线性可分的空间,并建立一个具有极小 VC 维数的分类器.该分类器仅由大量样本中的极少量支持向量确定,且具有最大的边界宽度.支持向量机算法的技巧在于不直接计算复杂的非线性变换,而是计算非线性变换的点积,即核函数,从而大大简化了计算.通过把核函数引入到一些学习算法,可以方便地把线性算法转换为非线性算法,我们将其与支持向量机一起称为基于核函数的方法.

收稿日期:2002-09-05
作者简介:田盛丰(1944—),男,河北定兴人,教授,博士生导师.email:sftian@center.njtu.edu.cn

本文介绍基于核函数方法的功能与应用及主要的研究方向.

1 支持向量机的算法分析

1.1 分类问题

对两类问题,设样本集为$(x_1,y_1),\cdots,(x_n,y_n)\in \mathbf{R}^d\times\{1,-1\}$,应使两类样本到分类超平面 $w\cdot x+b=0$ 的最小距离最大,即最大化分类间隔(Margin).可证分类间隔为 $2/\|w\|$,使分类间隔最大等价于 $\|w\|^2$ 最小.Vapnik 指出,$\|w\|^2$ 最小就相当于使 VC 维上界最小.故学习问题最小化目标函数为[1,2]

$$R(w,\xi)=\frac{1}{2}w\cdot w+C\sum_{i=1}^{n}\xi_i \tag{1}$$

约束条件为

$$y_i[w\cdot x+b]\geqslant 1-\xi_i,\xi_i\geqslant 0,i=1,\cdots,n.$$

其中,ξ_i 为松弛项,在线性不可分的情况下,允许一定的错分.可见,目标函数的第一项减小 VC 维,第二项减小经验风险,可得到最小的期望风险.在线性可分的情况下,经验风险为 0,VC 维得到最小化.在线性不可分的情况下,折中考虑了经验风险和 VC 维的最小化.

为求解这个优化问题,引入拉格朗日系数

$$L(w,b,\xi,\alpha,\gamma)=\frac{1}{2}w\cdot w+C\sum_{i=1}^{n}\xi_i-\sum_{i=1}^{n}\alpha_i[y_i(w\cdot x_i+b)-1+\xi_i]-\sum_{i=1}^{n}\gamma_i\xi_i \tag{2}$$

其中,$\alpha_i\geqslant 0,\gamma_i\geqslant 0$.

函数 $L(w,b,\xi,\alpha,\gamma)$应对 α_i 和 γ_i 最大化,且对 w,b 和 ξ_i 最小化.函数 $L(w,b,\xi,\alpha,\gamma)$的极值应满足条件

$$\frac{\partial}{\partial w}L(w,b,\xi,\alpha,\gamma)=0,\quad \frac{\partial}{\partial b}L(w,b,\xi,\alpha,\gamma)=0,\quad \frac{\partial}{\partial \xi_i}L(w,b,\xi,\alpha,\gamma)=0 \tag{3}$$

从而得到

$$\sum_{i=1}^{n}\alpha_i y_i=0,\quad w=\sum_{i=1}^{n}\alpha_i y_i x_i,\quad C-\alpha_i-\gamma_i=0 \tag{4}$$

将式(4)代入式(2),可以得到优化问题的对偶形式,最小化函数

$$W(\alpha)=\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j(x_i\cdot x_j)-\sum_{i=1}^{n}\alpha_i \tag{5}$$

其约束为

$$\sum_{i=1}^{n}\alpha_i y_i=0,\quad 0\leqslant\alpha_i\leqslant C.$$

这是一个典型的二次优化问题,已有高效的算法求解.

判别函数为

$$f(x)=\operatorname{sign}[\sum_{\text{支持向量}}\alpha_i y_i(x,x_i)+b] \tag{6}$$

对于非线性分类,首先使用一非线性映射函数 Φ 把数据从原空间 $\mathbf{R}^d$ 映射到一个高维特征空间Ω,再在高维特征空间 Ω 建立优化超平面.高维特征空间 Ω 的维数可能是非常高的,支持向量机算法巧妙地解决了这个问题.观察到在线性情况只用到了原空间的点积运算,在非线性空间也只考虑在高维特征空间 Ω 的点积运算 $\Phi(x)\cdot\Phi(y)=K(x,y)$,不必明确知道 $\Phi(x)$是什么函数.$K(x,y)$称为核函数,上述公式中只需将(x,y)变换成 $K(x,y)$即可.

核函数 $K(x,y)$的选取应使其为特征空间的一个点积,即 $\Phi(x)\cdot\Phi(y)=K(x,y)$.已经证明,对称函数 $K(x,y)$只要满足 Mercer 条件即可满足要求.常用的核函数有

(1)多项式核函数

$$K(x,y)=(x\cdot y+1)^d,\quad d=1,2,\cdots \tag{7}$$

(2)RBF(Radial Basis Function)核函数

$$K(x,y)=\exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right) \tag{8}$$

(3)Sigmoid 核函数

$$K(x,y)=\tanh[b(x\cdot y)-c] \tag{9}$$

式中,b、c 为常数.

1.2 回归问题

若考虑样本$(x_1,y_1),\cdots,(x_n,y_n)\in \mathbf{R}^d\times R$,回归函数设为[1]

$$f(\boldsymbol{x}) = \boldsymbol{w}\cdot\boldsymbol{\Phi}(\boldsymbol{x}) + b \tag{10}$$

优化问题最小化函数为

$$R(\boldsymbol{w},\boldsymbol{\xi},\boldsymbol{\xi}^*) = \frac{1}{2}\boldsymbol{w}\cdot\boldsymbol{w} + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \tag{11}$$

条件为 $f(\boldsymbol{x}_i) - y_i \leqslant \xi_i^* + \varepsilon$, $y_i - f(\boldsymbol{x}_i)\leqslant \xi_i + \varepsilon$, ξ_i、$\xi_i^* \geqslant 0$. 式(11)中第一项使函数更为平坦,从而提高泛化能力,第二项则为减小经验风险,常数 C 对两者做出折中. ε 为一正常数,$f(\boldsymbol{x}_i)$与 y_i 的差别小于ε时不计入误差,大于 ε 时误差计为$|f(\boldsymbol{x}_i) - y_i| - \varepsilon$.

引入拉格朗日函数可以得到优化问题的对偶形式,最大化函数为

$$W(\boldsymbol{\alpha},\boldsymbol{\alpha}^*) = -\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(\boldsymbol{x}_i\cdot\boldsymbol{x}_j) + \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)y_i - \sum_{i=1}^{n}(\alpha_i + \alpha_i^*)\varepsilon \tag{12}$$

其约束为

$$\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0,\quad 0\leqslant \alpha_i,\quad \alpha_i^* \leqslant C.$$

这也是一个二次优化问题,求解这个问题得到回归函数

$$f(\boldsymbol{x}) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(\boldsymbol{x},\boldsymbol{x}_i) + b \tag{13}$$

1.3 单类分析

单类分析(Single-class classification)可用于数据描述(SVDD, Support vector data description)、异常点检测(Novelty Detection)和聚类分析(Clustering),属于非监督学习的范围,主要有两种不同的方法.

(1)超平面法. 超平面法由 Schölkopf 等人于 1999 年提出[4]. 方法是在特征空间中计算一个超平面使之与原点的距离尽量大且使尽量多的训练例位于超平面的另一侧.

学习问题最小化目标函数为

$$R(\boldsymbol{w},\boldsymbol{\xi},\rho) = \frac{1}{2}\boldsymbol{w}\cdot\boldsymbol{w} + \frac{1}{\upsilon n}\sum_{i=1}^{n}\xi_i - \rho \tag{14}$$

条件为

$$\boldsymbol{w}\cdot\boldsymbol{\Phi}(\boldsymbol{x}_i)\geqslant \rho - \xi_i,\quad \xi_i\geqslant 0\qquad 0\leqslant i\leqslant n,$$

其中,$\rho\in\mathbf{R}$, $\upsilon\in(0,1)$.

决策函数为

$$f(\boldsymbol{x}) = \operatorname{sign}(\boldsymbol{w}\cdot\boldsymbol{\Phi}(\boldsymbol{x}_i) - \rho) \tag{15}$$

对大多数训练样本,函数的值为正 对奇异点,函数的值为负.

对偶形式最小化目标函数为

$$W(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) \tag{16}$$

其约束为

$$\sum_{i=1}^{n}\alpha_i = 1,\quad 0\leqslant \alpha_i \leqslant 1/\upsilon n.$$

判决函数为核的展开式

$$f(\boldsymbol{x}) = \operatorname{sign}\Big[\sum_{i=1}^{n}\alpha_i K(\boldsymbol{x}_i,\boldsymbol{x}) - \rho\Big] \tag{17}$$

由于在鞍点式(14)中非零的 α_i 所对应的条件等号成立,因此根据相应的 $\boldsymbol{x}_i$ 可求出ρ,即

$$\rho = \boldsymbol{w}\cdot\boldsymbol{\Phi}(\boldsymbol{x}_i) = \sum_{j=1}^{n}\alpha_j K(\boldsymbol{x}_j,\boldsymbol{x}_i) \tag{18}$$

(2)超球法. 超球法由 Tax 和 Duin 于 1999 年提出[5,6]. 方法是在特征空间计算一个超球使之围住大部分的训练例.

学习问题最小化目标函数为

$$L(R,a,\xi) = R^2 + C\sum_{i=1}^{n}\xi_i \tag{19}$$

条件为

$$\|\boldsymbol{\Phi}(\boldsymbol{x}_i) - a\|^2 \leqslant R^2 + \xi_i\qquad 0\leqslant i\leqslant n,$$

其中,R 和 a 分别为特征空间中超球的半径和球心.

对偶形式最小化目标函数为

$$W(\boldsymbol{\alpha}) = \sum_{i,j=1}^{n} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) - \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_i) \tag{20}$$

其约束为 $\sum_{i=1}^{n} \alpha_i = 1, \quad 0 \leqslant \alpha_i \leqslant C.$

且有

$$\boldsymbol{a} = \sum_{i=1}^{n} \alpha_i \boldsymbol{\Phi}(\boldsymbol{x}_i) \tag{21}$$

在特征空间,样本与球心的距离平方为

$$R^2 = \| \boldsymbol{\Phi}(\boldsymbol{x}) - \boldsymbol{a} \|^2 = K(\boldsymbol{x}, \boldsymbol{x}) - 2\sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) + \sum_{i,j=1}^{n} \alpha_i \alpha_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{22}$$

球的半径由少量球面上的支持向量决定 $R = \{R(\boldsymbol{x}_i) \mid \boldsymbol{x}_i$ 为支持向量$\}$.

对于单类分析,球内的点为类内的点,球外的点为奇异点. 对于聚类分析,对于给定的两点,若两点连线上所有的点均在球内,则两点为同类,否则为异类.

2 基于核函数的方法

将核函数的思想应用到其它线性学习机,可以得到非线性学习机的效果,称为基于核函数的方法(Kernel-based approaches).

2.1 基于核的 Fisher 判别分析

这是由 Mika 等人于 1999 年提出的方法[7]. 设两类 d 维样本分别为 $\boldsymbol{x}^1 = \{\boldsymbol{x}_1^1, \cdots, \boldsymbol{x}_{n1}^1\}$, $\boldsymbol{x}^2 = \{\boldsymbol{x}_1^2, \cdots, \boldsymbol{x}_{n2}^2\}$, $n = n1 + n2$. Fisher 判别分析的原理是将 d 维 $\boldsymbol{x}$ 空间的样本映射成一维空间点集,这个一维空间的方向就是相对于 Fisher 准则 $J(\boldsymbol{w})$ 为最大的 $\boldsymbol{w}$

$$J(\boldsymbol{w}) = \frac{\boldsymbol{w}^{\mathrm{T}} \boldsymbol{S}_{\mathrm{B}} \boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}} \boldsymbol{S}_{\mathrm{w}} \boldsymbol{w}} \tag{23}$$

式中,$\boldsymbol{S}_{\mathrm{B}} = (M_1 - M_2)(M_1 - M_2)^{\mathrm{T}}$ 为样本类间离散度矩阵;$\boldsymbol{S}_{\mathrm{w}} = \sum_{i=1}^{2} \sum_{x \in x^i} (x - M_i)(x - M_i)^{\mathrm{T}}$ 为样本类内离散度矩阵. 其中,M_1 和 M_2 分别为两类的均值.

与上面相同,令 $\boldsymbol{w} = \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i$,

代入 $J(\boldsymbol{w})$,并用核函数 $K(\boldsymbol{x}, \boldsymbol{x}_i) = \boldsymbol{\Phi}(\boldsymbol{x}) \cdot \boldsymbol{\Phi}(\boldsymbol{x}_i)$ 代替点积 $\boldsymbol{x} \cdot \boldsymbol{x}_i$,则有

$$J(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^{\mathrm{T}} M \boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}} N \boldsymbol{\alpha}} \tag{24}$$

式中,$M = (M_1 - M_2)(M_1 - M_2)^{\mathrm{T}}$;$(M_i)_j = \frac{1}{n_i} \sum_{k=1}^{n_i} K(\boldsymbol{x}_j, \boldsymbol{x}_k^i)$, $i = 1,2$, $j = 1, \cdots, n$;

$N = \sum_{i=1}^{2} K^i (\boldsymbol{I} - h^i)(K^i)^{\mathrm{T}}$;$(K^i)_{pq} = K(\boldsymbol{x}_p, \boldsymbol{x}_q^i)$, $i = 1,2$, $p = 1, \cdots, n$, $q = 1, \cdots, n_i$.

其中,$\boldsymbol{I}$ 为单位矩阵,h^i 的所有项均为 $1/n_i$, $i = 1,2$;可得 $\boldsymbol{\alpha} = N^{-1}(M_1 - M_2)$,为保证 N^{-1} 的计算,往往将 N 代以 $N + \mu \boldsymbol{I}$,其中 μ 为正整数. 映射可表示为

$$\boldsymbol{w} \cdot \boldsymbol{\Phi}(\boldsymbol{x}) = \sum_{i=1}^{n} \alpha_i (\boldsymbol{x}_i, \boldsymbol{x}) \tag{25}$$

2.2 基于核的感知机(Perceptron based on kernel)

这是 J. Xu 等人于 2001 年提出的方法[8]. 传统的单层神经网络(即感知机),在线性可分情况下是一个性能优良的分类器,其判别函数可定义为

$$g(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b \tag{26}$$

根据支持向量机的理论,向量 $\boldsymbol{w}$ 可表示为所有训练样本的线性组合

$$\boldsymbol{w} = \sum_{i=1}^{n} \alpha_i \boldsymbol{x}_i \tag{27}$$

因此 $g(x)$ 可表示为

$$g(x) = \sum_{i=1}^{n} \alpha_i < x, x_i > + b \tag{28}$$

用核函数 $K(x, x_i) = \Phi(x) \cdot \Phi(x_i)$ 代替点积 $x \cdot x_i$，可得

$$g(x) = \sum_{i=1}^{n} \alpha_i K(x, x_i) + b \tag{29}$$

式(29)即为基于核的感知机，在 $\Phi(x)$ 所处的特征空间 F，分类器是线性的，但在原空间的分类器则成为非线性的. 学习算法仍可采用梯度下降法.

2.3 非线性主成分分析(Kernel Principal Component Analysis)

PCA 用于以较少的维数描述数据，同时最大限度地保持数据的结构[9]. 给定数据集合 $\{x_1, x_2, \cdots, x_n\}$，$x \in \mathbf{R}^d$，目标是找到向量 w 使投影量 $w \cdot x$ 具有最大的偏差，相当于 1 类 Fisher 判别函数问题，即

$$\max_{w} \sum_{i=1}^{n} (w, x_i)^2 \tag{30}$$

为使 $\| w \|$ 尽量小，优化问题表示最小化目标函数为

$$R(w, e) = \frac{1}{2} w \cdot w - \frac{\gamma}{2} \sum_{i=1}^{n} e_i^2 \tag{31}$$

条件为 $e_i = w \cdot x_i, i = 1, \cdots, n$. 引入拉格朗日函数求解后得到

$$\frac{1}{\gamma} \alpha_i - \sum_{j=1}^{n} \alpha_j x_j \cdot x_i = 0.$$

定义 $\lambda = 1/\gamma$，得到对偶的对称特征值问题

$$\begin{bmatrix} x_1 \cdot x_1 & \cdots & x_1 \cdot x_n \\ \vdots & & \vdots \\ x_n \cdot x_1 & \cdots & x_n \cdot x_n \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} = \lambda \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix} \tag{32}$$

为得到最大的偏差，应选择最大特征值对应的特征向量，投影量为

$$z(x) = w \cdot x = \sum_{i=1}^{n} \alpha_i x_i \cdot x \tag{33}$$

引入非线性变换 $\Phi(x)$[9,10]，将投影放在高维特征空间进行，则对偶问题变成

$$C\alpha = \lambda\alpha \tag{34}$$

式中，$C_{ij} = \Phi(x_i) \cdot \Phi(x_j) = K(x_i, x_j)$，投影量变成

$$z(x) = w \cdot \Phi(x) = \sum_{i=1}^{n} \alpha_i K(x_i \cdot x) \tag{35}$$

3 主要的研究问题

近年来对支持向量机和其它基于核函数方法的研究集中在以下几个方面：高效的学习算法、支持向量机的简化、多类分类、新型的支持向量机、核函数的性质与选择等. 本文对前面 3 个方面加以讨论.

3.1 学习算法

支持向量机学习算法是典型的二次规划(QP)算法，常用内点算法求解，学习效率是很高的. 对于大规模样本集的应用，内点算法占用内存过多，因此发展了很多新的算法.

分解法是由 Osuna 等人于 1997 年为人脸检测而提出的[11]. Joachims 在 1998 年使之系统化，构造了软件 $\text{SVM}^{\text{light}}$，后来又有人不断改进[12]. 分解法的主要思想是把整个数据分成两个子集合 B 和 N，其中 B 为工作集合，保持固定大小. 每次循环中，将 α_N 固定，改变工作集合 B 中的 α_B：每次循环选择使 $W(\alpha)$ 获得最大程度的极小化的工作集合 B，对 B 求解 QP 子问题，直至满足最优条件为止.

Platt 于 1998 年提出串行算法[13]，他将工作集合的大小取为 2，即每次循环中仅取出 2 个 α_i 进行优化，因此可在保持约束 $\sum \alpha_i y_i = 0$ 的条件下用分析的方法将目标函数极小化. 在支持向量很少的情况下，若将所有 α_i 初始化为 0，则仅有少数 α_i 被调整，因此效率较高. 算法由两个步骤组成：选择一对 α_i，将它们进行优化.

Bradley 等人于 1998 年提出用线性规划算法求解支持向量机问题,即线性支持向量机(LSVM),再结合 Chunking 方法解决大规模数据问题[14].

Suykens 等人于 1999 年提出最小平方支持向量机(Least Squares Support Vector Machine, LS-SVM),在目标函数中使用平方误差[15].对分类问题,最小化目标函数为

$$R(\boldsymbol{w},\xi) = \frac{1}{2}\boldsymbol{w}\cdot\boldsymbol{w} + \frac{C}{2}\sum_{i=1}^{n}\xi_i^2 \tag{36}$$

约束条件为 $y_i[\boldsymbol{w}\cdot\boldsymbol{\Phi}(\boldsymbol{x}_i)+b] = 1-\xi_i \quad i=1,\cdots,n.$

这是一个等式约束的二次优化问题,其对偶问题相当于求解线性方程组

$$\begin{bmatrix} 0 & \boldsymbol{y}^{\mathrm{T}} \\ \boldsymbol{y} & \boldsymbol{H} \end{bmatrix}\begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \boldsymbol{e} \end{bmatrix} \tag{37}$$

式中,$\boldsymbol{y}=[y_1,\cdots,y_n]^{\mathrm{T}}$,$\boldsymbol{e}=[1,\cdots,1]^{\mathrm{T}}$,$\boldsymbol{H}_{ij}=y_iy_j\boldsymbol{K}_{ij}+1/C$.

LS-SVM 求解线性方程组相当于求解无约束的二次优化问题,因此速度很快.缺点是几乎所有样本都是支持向量($\alpha_i = C\xi_i$).

Mangasarian 和 Musicant 于 1999 年提出 SOR 算法(Successive Overrelaxation)[16],该算法利用变型的目标函数,即最小化形式为

$$R(\boldsymbol{w},\xi) = \frac{1}{2}\boldsymbol{w}\cdot\boldsymbol{w} + \frac{1}{2}b^2 + C\sum_{i=1}^{n}\xi_i \tag{38}$$

约束条件为 $y_i[\boldsymbol{w}\cdot\boldsymbol{x}+b]\geqslant 1-\xi_i \quad i=1,\cdots,n;\quad \xi_i\geqslant 0 \quad i=1,\cdots,n.$

对偶形式为 $$\min 1/2\,\boldsymbol{\alpha}^{\mathrm{T}}(Q+\boldsymbol{y}\boldsymbol{y}^{\mathrm{T}})\boldsymbol{\alpha} - \boldsymbol{e}^{\mathrm{T}}\boldsymbol{\alpha} \tag{39}$$

约束条件为 $0\leqslant\alpha_i\leqslant C \quad i=1,\cdots,n.$

其中,$Q_{ij}=y_iy_j$,构成无等式约束的二次优化问题,因此每次循环仅优化 1 个点即可,可以处理上千万个样本的大规模问题.据称该算法对小规模问题快于 SVM$^{\mathrm{light}}$,相当或快于 SMO.

Lee 和 Mangasarian 于 1999 年提出 SSVM(Smooth Support Vector Machine)[17].该算法利用变型的目标函数,并利用函数

$$p(\boldsymbol{x},a) = \boldsymbol{x} + a^{-1}\log(1+\mathrm{e}^{-a\boldsymbol{x}}) \qquad a>0 \tag{40}$$

消去不等式约束,从而将优化问题转化为对任意函数的无约束优化问题.一般可取 $a=5$,利用 Newton-Armijo 算法求解.据称对大规模问题,SSVM 相当于或快于 SVM$^{\mathrm{light}}$,SOR 和 SMO.

Keerthi 等人于 1999 年提出最近点法(Nearest Point Algorithm)[18],把分类问题转化为计算两个凸多边形的最近点,但该算法不能求解回归问题.

Yang 等人于 2000 年提出了一种几何方法[19],通过求解一组线性规划问题寻找‘卫向量’(Guard vector),支持向量的一个小的超集,再求解以‘卫向量’构成的小型 QP 问题找出支持向量,算法较传统 QP 方法效率更高,且需要更少的内存.

Pavlov 等人于 2000 年提出 Boost-SMO 算法[20],该算法训练一个分类器序列,下一个分类器主要考虑上一个分类器的错分样本.一般情况下虽不能达到最佳,但速度比 SMO 快 3~10 倍.

Pavlov 等人于 2000 年提出 Squashing 算法[21],利用聚类方法构造带权的压缩数据集合,样本数减少 50~100 倍,从而提高了效率.

3.2 支持向量机的简化

支持向量机计算的复杂性依赖于支持向量的数目,因此减少支持向量的数目成为一个重要的研究课题.Burges 于 1996 年提出了一种简化技术[22],设

$$\boldsymbol{w} = \sum_{i=1}^{n}\alpha_iy_i\boldsymbol{\Phi}(\boldsymbol{x}_i) \tag{41}$$

寻找 $$\boldsymbol{w}' = \sum_{i=1}^{m}\beta_i\boldsymbol{\Phi}(\boldsymbol{z}_i) \qquad m<n \tag{42}$$

使 $\|\boldsymbol{w}-\boldsymbol{w}'\|^2$ 最小.算法的目标是:①确定最优系数 β;②确定简化集 $\{\boldsymbol{z}_i\}$.

Downs 等人于 2001 年提出了一种精确的简化算法[23],利用某些支持向量在特征空间线性相关的特

点进行简化.以分类问题为例,设支持向量 x_k 在特征空间与其它支持向量线性相关,即

$$K(\boldsymbol{x},\boldsymbol{x}_k)=\sum_{i=1,i\neq k}^{n}c_iK(\boldsymbol{x},\boldsymbol{x}_i) \tag{43}$$

式中,c_i 为常数.因此决策面可表示为

$$f(\boldsymbol{x})=\sum_{i=1,i\neq k}^{n}\alpha_iy_iK(\boldsymbol{x},\boldsymbol{x}_i)+\alpha_ky_k\sum_{i=1,i\neq k}^{n}c_iK(\boldsymbol{x},\boldsymbol{x}_i)+b \tag{44}$$

定义 $\alpha_ky_kc_i=\alpha_iy_i\gamma_i$,故上式可写成

$$f(\boldsymbol{x})=\sum_{i=1,i\neq k}^{n}\beta_iy_iK(\boldsymbol{x},\boldsymbol{x}_i)+b \tag{45}$$

式中,$\beta_i=\alpha_i(1+\gamma_i)$.通过在特征空间发现线性相关,减少了支持向量数目.

3.3　多类问题

支持向量机处理的基本问题是两类分类,多类问题可在两类分类的基础上进行.常用的方法有两种:一对其余$(1-v-r)$和一对一$(1-v-1)$.对于 k 类分类,一对其余的方法是为每一类建立一个分类器,训练集以该类为一类,以其余各类为另一类,共需建立 k 个分类器.一对一的方法为每一对类别建立一个分类器,共需建立 $k(k-1)/2$ 个分类器.

Weston 和 Watkins 于 1998 年提出了一次优化的方法[24].设 n 个 k 类样本表示为:$(x_1,y_1),\cdots,(x_n,y_n)$,优化问题表示为最小化

$$\frac{1}{2}\sum_{m=1}^{k}\boldsymbol{w}_m\cdot\boldsymbol{w}_m+C\sum_{i=1}^{n}\sum_{m\neq y_i}^{k}\xi_i^m \tag{46}$$

约束为

$$\boldsymbol{w}_{yi}\cdot\boldsymbol{\Phi}(\boldsymbol{x}_i)+b_{yi}\geqslant\boldsymbol{w}_i\cdot\boldsymbol{\Phi}(\boldsymbol{x}_i)+b_m+2-\xi_i^m$$

$$\xi_i^m\geqslant 0,i=1,\cdots,n\qquad m=\{1,\cdots,k\}\setminus y_i.$$

判决函数为

$$f(\boldsymbol{x})=\arg\max_m[\boldsymbol{w}_m\cdot\boldsymbol{\Phi}(\boldsymbol{x})+b_m]\qquad m=1,\cdots,k \tag{47}$$

注意,当 $k=2$ 时,若取 $\boldsymbol{w}_1=-\boldsymbol{w}_2,b_1=-b_2$,则上式与二类问题的表示完全等价.

4　支持向量机的应用

综上所述,支持向量机具有很强的泛化能力且没有局部最优问题.此外,优化问题的规模只与样本数有关,而与样本维数无关,即有利于维数高而样本少的问题,如人脸识别问题,而且不需要使用先验知识建造机器的结构.支持向量机还巧妙地处理了非线性问题.因此,支持向量机和其它基于核的方法获得了很多应用. 支持向量机最基本的应用是模式识别,如人脸检测[11]、文本识别[25,26]、物体识别[27]、说话人识别、文字识别、DNA 分析[28]等.

时间序列预测问题可转化为回归问题[29],如给定 m,问题可表示为

$$(\boldsymbol{x}_i,y_i):=((\boldsymbol{x}_{i-m},\cdots,\boldsymbol{x}_i),\boldsymbol{x}_{i+1}).$$

时间序列的预测用途很广,如可用于股市预测等方面.

基于核的 PCA 作为一种非线性特征提取的工具也可用于图像处理,完成去噪、压缩等功能[30].

参考文献:

[1] Cristianini N, Shawe-Taylor J. An Introduction to Support Vector machines[M]. Cambridge, UK: Cambridge University Press, 2000.

[2] Müller K-R, Mika S, Rätsch G, et al. An Introduction to Kernel-Based Learning Algorithms[J]. IEEE Transactions on Neural Networks, 2001,12(2):181-201.

[3] Vapnik V N. The Nature of Statistical Learning Theory[M]. NY: Springer, 1995.

[4] Schölkopf B, Platt J C, Shawe-Taylor J, et al. Estimating the Support of a High-dimensional Distribution[R]. Technical Report MSR-TR-99-87, Microsoft Research, 1999.

[5] Tax D, Duin R. Data Domain Description by Support Vectors[A]. In: Verleysen M. Ed. D. Proc. ESANN[C]. Brussels: Facto Press, 1999, 251-256.

[6] Ben-Hur A, Horn D, Siegelmann H T, et al. Support Vector Clustering[J]. Journal of Machine Learning Research, 2001, 2:

125－137.
[7] Mika S, Rätsch G, Weston B, et al. Fisher Discriminant Analysis with Kernels[A]. Neural Networks for Signal Processing IX[C]. New York：IEEE Press, 1999, 41－48.
[8] Xu J, Zhang X, Ci Y. Kernel Neuron and its Training Algorithm[A]. In：ICONIP2001[C], 2001.
[9] Suykens J A K, Gestel T V, Vandewalle J, et al. A Support Vector Machine formulation to PCA Analysis and its Kernel version[R]. ESAT-SCD-SISTA Technical Report 2002－68, Belgium：Katholieke Universiteit Leuven, 2002.
[10] Schölkopf B, Smola A, Müller K-R. Nonlinear Component Analysis as a Kernel Eigenvalue Problem[J]. Neural Computation, 1998, 10：1299－1319.
[11] Osuna E, Freund R, Girosi F. Training Support Vector Machines, An Application to Face Detection[A]. In：Proc. of CVPR'97[C], Puerto Rico, 1997.
[12] Joachims T. Making Large-scale svm Learning Practical[A]. In：Schölkopf B. et al. Eds. Advances in Kernel Method-support Vector Learning[C]. Cambridge, MA：MIT Press, 1998.
[13] Platt J C. Sequential Minimal Optimization：a Fast Algorithm for Training Support Vector Machines[R]. Technical Report MSR-TR-98-14, Microsoft Research,1998.
[14] Bradley P S, Mangasarian O L. Massive Data Discrimination Via Linear Support Vector Machines[R]. Technical Report 98－05, Madison, WI：Univ. of Winsconsin, 1998.
[15] Suykens J A K, Vandewalle J. Least Square Support Vector Machine Classifiers[J]. Neural Processing Letters, 1999, 9(3)：293－300.
[16] Mangasarian O L, Musicant D R. Successive Overrelaxation for Support Vector Machines[J]. IEEE Tran. On Neural Networks, 1999, 10：1032－1037.
[17] Lee Y T, Mangasarian O L. SSVM：A Smooth Support Vector Machine for Classification[R]. Technical Report 99－03, Madison：Univ. of Winsconsin, Data Mining Institute, 1999.
[18] Keerthi S S, Shevade S K. Bhattacharyya C, et al. A Fast Iterative Nearest Point Algorithm to Support Vector Machine Classifier Design[R]. Technical Report TR-ISL-99-03, India：India Institute of Science, 1999.
[19] Yang M, Ahuja N. A Geometric Approach to Train Support Vector Machines[A]. In：Proc. of the 2000 IEEE Conf. on Computer Vision and Pattern Recognition[C]. CVPR2000, 2000. 430－437.
[20] Pavlov D, Mao J, Dom B. Scaling-up Support Vector Machines Using Boosting Algorithm[A]. In：Proceedings of the International Conference on Pattern Recognition[C]. ICPR-2000,2000.
[21] Pavlov D, Chudova D, Smyth P. Towards Scalable Support Vector Machines Using Squashing[A]. In：Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery in Databases[C]. KDD-2000, 2000.
[22] Burges, C J C. Simplified Support Vector Decision Rules[A]. In：Saitta, L. ed. Proceedings of 13th International Conference on Machine Learning[C]. San Mateo, CA：Morgan Kaufmann Publishers, Inc., 1996. 71－77.
[23] Downs T, Gates K E, Masters A. Exact Simplification of Support Vector Solutions[J]. Journal of Machine Learning Research, 2001, 2：293－297.
[24] Weston J, Watkins C. Multi-class Support Vector Machines[R]. Technical Report CSD-TR-98-04, London：Royal Holloway University of London, 1998.
[25] Joachims T. Transductive Inference for Text Classification Using Support Vector Machines[A]. In：Int. Conf. on Machine Learning(ICML)[C]. Blad,Slowenien, 1999.
[26] Joachims T. Text Categorization with Support Vector Machines：Learning with Many Relevant Features[A]. In：Proc. of the European Conf. on Machine Learning[C]. Berlin,1998, 137－142.
[27] Roobaert D. Improving the Generalization of Linear Support Vector machines：An Application to 3D Object Recognition with Cluttered Background[A]. In：Proc. SVM Workshop at the 16th Int J[C]. Conf. on AI, Stockholm, Sweden, 1999.
[28] Zien A., Rätsch G, Mika S, et al. Engineering Support Vector Machine Kernels That Recognize Translation Initiation Sites in DNA[J]. Bioinformatics, 2000, 16：799－807.
[29] Müller K-R, Smola A, Rätsch G, et al. Predicting time Series with Support vector machines[A]. In：Schölkopf B, Burges C J C, Smola A J. Eds. Advances in Kernel Methods-Support Vector Learning[C]. MA：MIT Press, 1999, 243－254.
[30] Mika S, Schölkopf B, Smola A, et al. Kernel PCA and De-noising in Feature Spaces[A]. In：Kearns M S, Solla S A, Cohn D A. Eds. Advances in Neural Information Processing Systems II[C]. Cambridge, MA：MIT Press, 1999, 536－542.