支持向量机的训练算法综述

王书舟，伞 冶
（哈尔滨工业大学 控制与仿真中心，黑龙江 哈尔滨 150001）

摘 要：支持向量机（SVM）是在统计学习理论基础上发展起来的新方法，其训练算法本质上是一个二次规划的求解问题.首先简要概述了SVM的基本原理，然后对SVM训练算法的国内外研究现状进行综述，重点分析SVM的缩减算法和具有线性收敛性质的算法，对这些算法的性能进行比较，并且对SVM的扩展算法也进行简单介绍.最后对该领域存在的问题和发展趋势进行了展望.

关键词：统计学习理论；支持向量机；训练算法
中图分类号：TP391.9 **文献标识码**：A **文章编号**：1673-4785(2008)06-0467-09

A survey on training algorithms for support vector machine

WANG Shu-zhou, SAN Ye
(Control & Simulation Centre, Harbin Institute of Technology, Harbin 150001, China)

Abstract: Support vector machines (SVMs) use new methods that originated in statistical learning theory. Training of an SVM can be formulated as a quadratic programming problem. The principles of SVM have been summarized briefly in this paper. The latest developments in SVM training algorithms in domestic and overseas research were reviewed, especially reduction algorithms and algorithms with linear convergence properties. The performance of these algorithms was then compared, and a brief introduction to a proposed extension of them was given. Finally some problems and potential directions for future research are discussed.

Keywords: statistical learning theory; support vector machine; training algorithms

支持向量机（support vector machine，SVM）是近年发展起来的一种通用机器学习新方法.它不但具有坚实的理论基础、简洁的数学形式、直观的几何解释，而且能够较好地解决小样本、非线性、维数灾和局部极小等问题[1-2]，因此在模式分类[3-4]、回归问题[5-6]等很多领域得到了广泛的应用.

训练SVM的算法归结为求解一个受约束的凸二次优化问题.对于小规模的二次优化问题，利用牛顿法、内点法等成熟的经典最优化算法就可以很好地求解.但这些算法通常需要利用整个Hessian矩阵，内存占用过多，从而导致训练时间过长.当训练集很大，特别是支持向量数目也很大时，求解二次优化问题的经典方法不再可行.因此设计适用于大量样本的训练算法成为SVM研究的重要内容.目前针对SVM本身的特点提出了许多算法，本文对这些算法进行综述，包括块算法、分解算法、序贯最小优化算法、增量与在线训练算法、缩减算法、具有线性收敛性质的算法，以及SVM的扩展算法.

收稿日期：2008-06-30.
基金项目：国家自然科学基金资助项目(60474069).
通信作者：王书舟. E-mail：seek2000@163.com.

1 支持向量机

SVM最初是在模式分类中提出的，其基本思想是：通过非线性变换$\phi(\cdot)$将输入空间映射到一个高维特征空间，在这个特征空间中求取最大间隔分类超平面$f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\phi(\boldsymbol{x})+b$，其中$\boldsymbol{w}$、$b$分别是这个超平面的权值和阈值.特征空间的维数可能是非常高的，通常导致计算非常复杂.SVM算法通过核函数$K(\boldsymbol{x},\boldsymbol{y})$巧妙地解决了这个问题.SVM不直接计算复杂的非线性变换$\phi(\cdot)$，而是计算非线性变换$\phi(\cdot)$的内积$K(\boldsymbol{x},\boldsymbol{y})$，即核函数$K(\boldsymbol{x},\boldsymbol{y})=\phi(\boldsymbol{x})\cdot\phi(\boldsymbol{y})$，从而大大简化了计算.核函数$K(\boldsymbol{x},\boldsymbol{y})$的利用是由于在原空间和高维特征空间只用到了内积运算

的缘故.

类似用于模式分类的SVM,回归SVM(support vector regression,SVR)的基本思想是,通过非线性变换$\phi(\cdot)$,将输入空间映射到一个高维特征空间,并在这个特征空间用线性函数$f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\phi(\boldsymbol{x})+b$拟合样本数据,同时保证能得到较好的泛化能力. 设$\boldsymbol{x}_i\in R^n, y_i\in R, i=1,\cdots,l, l$为观测样本,$R^n$代表输入空间,SVR问题可以表示为线性约束二次规划的优化问题:

$$\min\left[\frac{1}{2}\|\boldsymbol{w}\|^2+\frac{C}{l}\sum_{i=1}^{l}(\zeta_i+\zeta_i^*)\right],$$

$$\text{s.t.}\begin{cases}((\boldsymbol{w}\cdot\boldsymbol{x}_i)+b)-y_i\leqslant\varepsilon+\zeta_i,\\ y_i-((\boldsymbol{w}\cdot\boldsymbol{x}_i)+b)\leqslant\varepsilon+\zeta_i^*,\\ \zeta_i,\zeta_i^*\geqslant 0.\end{cases}\tag{1}$$

其中$C>0$是函数复杂度和损失误差之间的一个平衡量. 由优化问题(1)的Lagrange函数相对于变量$\boldsymbol{w}$、b、ζ_i、ζ_i^*的偏导数为0,可得优化问题(1)的对偶问题:

$$\min\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)K(\boldsymbol{x}_i,\boldsymbol{x}_j)+\varepsilon\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)-\sum_{i=1}^{l}y_i(\alpha_i-\alpha_i^*).\tag{2}$$

$$\text{s.t.}\begin{cases}\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)=0,\\ 0\leqslant\alpha_i,\alpha_i^*\leqslant l, i=1,\cdots,l.\end{cases}$$

这个约束最优化问题的解是核函数的线性组合,具有如下的形式:

$$f(\boldsymbol{x})=\sum_{i=1}^{l}(\alpha_i-\alpha_i^*)K(\boldsymbol{x}_i,\boldsymbol{x})+b\tag{3}$$

这个用作函数回归的学习机器就是支持向量机,支持向量就是使表达式(3)中系数$(\alpha_i-\alpha_i^*)$不为零的训练样本. 在使用SVM解决实际问题时,首先需要把它转化为一个可以用SVM求解的数学模型,这一工作称为模型选择,它包括训练集的选择、SVM类型的选择、核函数的选择、SVM中自由参数的选择等. 对已经进行了模型选择的最优化问题(1)或(2),其求解方法就是SVM的训练算法.

2 支持向量机的基本算法

2.1 块算法与分解算法

块算法(chunking algorithm)[2]最早是由Boser等人提出来的. 它的出发点是,删除矩阵中对应于Lagrange乘子为零的行和列不会对最终结果产生影响. 对于给定的样本,块算法的目标就是通过某种迭代方式逐步排除非支持向量. 块算法将矩阵的规模从训练样本数的平方减少到具有非零Lagrange乘子的样本数的平方,从而降低了训练过程对存储容量的要求.

Osuna等人提出的分解算法(decomposition algorithm)[2],是目前有效解决大规模问题的主要方法. 分解算法将二次规划问题分解成一系列规模较小的二次规划子问题,进行迭代求解. 在每次迭代中,选取拉格朗日乘子分量的一个子集做为工作集,利用传统优化算法求解一个二次规划的子问题. Joachims在上述分解算法的基础上做了几点重要改进. 第一,采用类似Zoutendijk可行方向法的策略确定工作集B,即求解一个线性规划问题,得到可行下降方向,把该方向中的非零分量作为本次迭代的工作集. 该线性规划存在高效算法,其核心是一个排序问题. 第二,提出shrinking方法,估计出有界支持向量和非支持向量,有效地减小QP问题的规模. 最后,利用KernelCache来减少矩阵中元素的计算次数. Joachims利用这些方法实现的SVMlight,是目前设计SVM分类器的重要软件.

Lin[7-8]和Takahashi[9]等人分析并证明了分解算法的收敛性. Lin[10]对v-SVM和多类SVM的停机准则进行了分析,并针对多类SVM证明了分解算法的渐进收敛性. Hu[11]得到了鲁棒SVM的KKT条件和分解算法,利用预选取技术提高分解算法的收敛速度. Dong[12]引进并行优化算法来快速剔除非支持向量,用对角块矩阵逼近原核矩阵,从而原始问题分解为易于求解的子问题. Qiao[13]提出一种工作集选择规则,使分解算法具有多项式收敛的性质.

块算法的目标是找出所有的支持向量,因而最终需要存储相应的核函数矩阵. 对于支持向量数很大的问题,块算法十分复杂. 与块算法不同,分解算法的目的不是找出所有的支持向量,而是每次只针对很小的训练子集来求解,即使支持向量的个数超过工作集的大小,也不改变工作集的规模. 各种分解算法的区别在于工作集的大小和工作集生成原则的不同. 工作集的选择对于分解算法的收敛与否和收敛速度至关重要,因此开发新的工作集选择算法是提高分解算法性能的重要途径.

除了分解算法,还有Huber近似算法[14]、多拉格朗日乘子协同优化算法[15]、剪枝算法[16]等SVM的求解方法.

2.2 序贯最小优化算法

由Platt提出的序列最小优化(sequential mini-

mal optimization,SMO)算法[17]是分解算法工作集的个数等于 2 的特殊情形,即 SMO 把一个大的优化问题分解成一系列只含两个变量的优化问题. 两个变量的最优化问题可以解析求解,因而不需要迭代地求解二次规划问题. 对分类 SMO 算法,Keerthi 等人[18]修正了优化条件,并针对经验方法提出两个改进措施,以保证算法收敛和减少迭代次数. 随后 Keerthi 等人[19]提出了广义 SMO(generalized SMO,GSMO)算法,利用违反对的概念确定工作集,指出前面两种改进都是 GSMO 的特例,并证明,$\forall t>0$,以 τ 违反对为工作集,则 GSMO 算法有限终止,得到优化问题的 τ 近似优化解. Lin[20]对 SMO 算法的渐进收敛性进行了证明.

起初 SMO 算法主要用于模式分类问题,后来 Smola[21]等人进行类比扩展,提出了一种训练回归 SVM 的 SMO 算法. 这种算法选取两对变量 α_i、$\hat{\alpha}_i$、α_j、$\hat{\alpha}_j$,在每个迭代步,类似 Platt 的 SMO 的策略,按照所选变量不同取值的 4 种情况,对 QP 子问题进行解析求解. Shevade[22]指出 Smola 的更新阈值 b 方法并不是很有效,提出利用双阈值方法改进,同时还提出了一种按照违反 KKT 优化条件的程度选择变量的新方法. Flake 等人[23]针对 SVM 回归问题,指出通过设置 $\beta_i \triangleq \alpha_i - \hat{\alpha}_i, i=1,\cdots,l$,$2l$ 变量的 QP 问题可以转化为 l 个变量的 QP 问题,并改进了经验方法以提高缓存的利用效率. Michael[24]针对不需要计算偏置项 b 的情况,分别对 SMO 的分类和回归算法进行了改进. Keerthi[25]提出了用于最小二乘 SVM 的 SMO 算法. Zeng[26]等人提出了基于 SMO 算法的稀疏最小二乘剪枝算法. Norikazu[27-29]对 SMO 的中止条件进了严格的证明. Cao[30-31]提出了训练 SVM 的并行 SMO 算法. Chen[32]等人对 SMO 类型的分解算法进行了研究. Bo[33]对最小二乘 SVM 的 SMO 算法的工作集的选择进行了研究. 此外还有其他针对 SMO 的分类和回归的改进算法[34-37].

SMO 算法是分解算法中选取工作集为 2 的特殊情形,SMO 将工作集的规模减到最小,一个直接的后果就是迭代次数的增加. 与分解算法相比,尽管它可能需要更多的迭代步,但是由于每步只需要很少的计算量,SMO 算法通常表现出整体快速收敛的性质. 另外该算法还具有不需要储存核矩阵、没有矩阵运算、容易实现等重要特点. 但是分解算法、SMO 算法,都是只能够离线应用的训练算法.

2.3　增量及在线训练算法

如果学习机的样本是随着时间序列获得的,或是在线采集的,就必须使用增量式学习算法或在线学习算法[38-39]. 增量学习是机器学习系统在处理新增样本时,能够只对原学习结果中与新样本有关的部分进行增加、修改或删除操作,与之无关的部分则不被触及. 增量训练算法的一个突出特点是支持向量机的学习不是一次离线进行的,而是一个数据逐一加入反复优化的过程. 文献[40]的算法改进是基于 Cauwenberghs 提出的用于模式识别的增量减量式学习方法,这种算法的目的是防止训练过程中有用支持向量的丢失. 考虑了增加或减少一个训练样本对拉格朗日系数和 SVM 的影响. 在减少一个样本时,给出了模型选择算法留一法的形象解释. 增量型支持向量机训练算法可以用于实时在线训练,如 Ma[41]提出了用于函数回归问题的增量型在线 SVM 训练算法. 序贯训练算法是样本数据序贯加入的训练方法,也是在线训练方法的一种. 人们提出一种 Kernel Adatron 算法[42]对 SVM 进行分类的序贯训练,这种方法的基本思想是借助感知机中的 Adatron 算法的原理来改变拉格朗日系数,具体来说通过序贯加入的样本的预测误差来修改 SVM 样本的系数,其本质上是一个爬山的寻优算法,通过反复的修改序贯加入样本的系数,使训练过程最终收敛. 后来将上述算法改进,使其不但适合分类而且也适合回归. Vojislav[42]证明了 Kernel Adatron 与 SMO 算法的等价性. 用于回归的 Kernel Adatron 和 SMO 的系数更新公式分别为

$$\begin{cases}\alpha_i \leftarrow \alpha_i - \alpha_i^* - \eta_i(E_i+\varepsilon),\\ \alpha_i^* \leftarrow \alpha_i^* - \alpha_i + \eta_i(E_i-\varepsilon);\end{cases} \tag{4}$$

$$\begin{cases}\alpha_i \leftarrow \alpha_i - \alpha_i^* - \dfrac{(E_i+\varepsilon)}{K(\boldsymbol{x}_i,\boldsymbol{x}_i)},\\ \alpha_i^* \leftarrow \alpha_i^* - \alpha_i + \dfrac{(E_i-\varepsilon)}{K(\boldsymbol{x}_i,\boldsymbol{x}_i)}.\end{cases} \tag{5}$$

取 $\eta_i = 1/K(\boldsymbol{x}_i,\boldsymbol{x}_i)$,则容易看出式(4)与(5)算法的等价性. Yaakov[43]在 Kernel Adatron 算法的基础上,结合支持向量的精确缩减,并把它们用于在线学习的框架,提出一种稀疏在线贪婪(sparse online greedy,SOG)SVR 算法.

Smola[44]提出了一种在再生核 Hibert 空间采用随机梯度下降的在线算法,可以用于支持向量机的分类、回归和新奇性检测. Smale[45]提出在再生核空间和一般的 Hibert 空间的在线算法,给出了随机梯度算法的一般形式和可能的收敛界. Ying[46]提出了一种在再生核 Hibert 空间中,基于一般凸正则损失

函数的在线分类算法. Bianchi[47-48]基于独立同分布的训练数据,给出对任意在线训练方法所取得具有的最小风险的假设.

增量减量算法[39-41]、基于 Kernel Adatron 的算法[42-43]、在线算法[44-48]都可用于在线学习,但是增量减量算法比较复杂,需要存储核矩阵,因此训练速度较慢. 而基于 Kernel Adatron 的算法原理简单,每次更新的运算量小,需要的内存也不大. 然而这种训练算法随着每次新样本的加入,都会引起其他样本系数的改变,需要不断地进行反复优化. 在线算法[47-48]可以采用基于随机梯度下降的方法,算法比较简单,而且对风险的界和收敛性等都有较严格的理论分析和证明. 另外,这些在线学习方法的缺点是需要考虑所有的历史数据,无法控制支持向量的个数. 只有考虑到 SVM 解的稀疏性,才有可能减小计算量,缩短计算时间.

2.4 支持向量缩减算法

支持向量的数量的下界与训练样本数成线性关系[49],这表明支持向量的数量至少随训练样本的加入而线性地增加. 对具有大量训练样本的优化问题,支持向量的数量是影响在线训练和预测速度的主要因素. 因此,减少预测函数中包含支持向量的数量,成为提高 SVM 在线训练速度和预测速度的目标. 文献[50]提出一种内积矩阵分解的算法,来提高 SVM 的分类速度. 内积矩阵分解算法的思想是通过优化的方式,适当地变换特征空间中的内积矩阵,并进一步变换分类函数的形式,既减少分类函数中支持向量的数量,又能将全部支持向量的信息保留在分类函数中,达到不损失分类精度,又提高分类速度的目的. Wu[51]通过在标准支持向量上附加更多的限制条件,直接得到稀疏大间隔分类器. Nguyen[52-53]提出了一种 bottom-up 的方法,其缩减过程迭代地选择属于同一类的两个最近的支持向量,然后用一个新向量代替. 新向量的构建只需要计算一个单变量函数在(0,1)内的惟一最大值点,因此可以有效减少支持向量的数量. Keerthi[54]提出用贪婪算法选择基向量,来逼近原支持向量,从而减小分类函数的复杂度. Li[55]基于向量相关准则和贪婪算法的思想,提出了一种特征向量选择的自适应缩减方法. Sumeet[56]利用 Span 概念进行支持向量的启发式缩减,并提出了一种支持向量机在线训练算法.

支持向量缩减方法可以用于在线学习,但是这些方法的缺点是需要考虑所有的历史数据,没有遗忘机制,无法控制支持向量的个数. 当样本增加较多时,需要保存的样本也很多. Crammer[57]引入一个称为 Budget 的量,他以 Rosenblatt 的感知机为基础,增加一个样本的插入和删除过程,从而达到控制支持向量个数的目的. 令 I 为样本索引的集合,$|I|$ 表示集合所含元素的个数,则此方法可以描述如下:

初始化: 设置 ε, n, $\alpha_i = 0, \boldsymbol{w}_0 = \boldsymbol{0}, I_0$ 为空

For $t = 1, 2, \cdots, T$

取得一个新样本 $\boldsymbol{x} \in R^n$, 及其标签 y_t, 并做预测:

$\hat{y} = \text{sign}\ (y_t(\boldsymbol{x}_t \cdot \boldsymbol{w}_{t-1}))$.

If $y_t(\boldsymbol{x}_t \cdot \boldsymbol{w}_{t-1}) \leqslant \varepsilon$

1) If $|I_t| = n$, 删除一个样本:

a. 令 $i = \text{argmax}_{j \in I_t}\{y_j(\boldsymbol{w}_{t-1} - \alpha_j y_j \boldsymbol{x}_j)\}$.

b. 更新: $\boldsymbol{w}_{t-1} \leftarrow \boldsymbol{w}_{t-1} - \alpha_i y_i \boldsymbol{x}_i$.

c. 删除第 i 个样本: $I \leftarrow I / \{i\}$.

End

2) 插入新样本: $I_t \leftarrow I_{t-1} \cup \{t\}$.

3) 令 $\alpha_t = 1$.

4) 计算 $\boldsymbol{w}_t \leftarrow \boldsymbol{w}_{t-1} + y_t \alpha_t \boldsymbol{x}$.

End

End

输出 $f(\boldsymbol{x}) = \text{sign}(\boldsymbol{w}^{\mathrm{T}} \cdot x)$.

Weston[58]指出上述以间隔的大小作为删除样本的度量准则,具有噪声敏感的缺点,进而提出一种改进算法,以选定的样本子集的误差作为度量准则,具有更好的鲁棒性. Dekel[59]通过收缩系数 ϕ_t 选取,删除的是最旧的样本,得到了具有严格限制的支持向量和相关错误界. Dekel[60]用插值模代替标准 SVM 中的任意模,实现对支持向量个数的控制. 并证明了: p 插值范数和 ∞ 范数,近似等价于限制为样本前 t 个绝对值最大的元素的 p 插值范数; 特别地,当 $p = 1$ 时, $1 - \infty$ 插值范数等价于对样本 t 个绝对值最大的元素进行绝对值求和. 在此文献中还改进标准的 SMO 算法以适应这种框架下的求解.

上述缩减算法大多考虑了 SVM 解的稀疏性,但却又回到了更复杂的解决方法上,如通过核函数构成的矩阵的逆计算所有支持向量,计算新的优化问题等,实际上没能做到减少运算量.

2.5 具有线性收敛性质的算法

训练 SVM 的常用方法,如分解方法、SMO 方法等,结果收敛所需要的时间对样本数 n 来说通常是超线性的. 因此这些方法在应用于大数据集时失去有效性. Joachims[61]对线性支持向量机提出了 SVM-Perf 方法,这种方法在每一迭代步计算当前解的梯度并把它加到优化问题中,使训练 SVM 所用的时间

与样本数呈线性关系. SVMPerf 以精度 ε 在时间 $O(md/(\lambda\varepsilon^2))$ 内求得解. 这个界被 Shwartz[60] 提出的 Pegasos 方法改进为以精度 ε,以置信度 $1-\delta$,在时间 $O(1/(\lambda\delta\varepsilon))$ 内得到解. Pegasos 实质上是执行随机次梯度下降,把解返回并投影到以 $1/\sqrt{\lambda}$ 半径的 L_2 球面上.

Smola[63] 针对正则风险最小化问题,提出了一种统一框架的全局收敛方法,可以应用于任何导致凸优化问题的正则风险最小化问题. 其基本思想是:在正则化函数中,正则项不变,用经验风险的下界,即经验风险的一次泰勒逼近,代替经验风险,进行最优化求解. Smola 指出 SVMPerf 是这种方法的一个特例,并给出了更紧的收敛界,即对 ε 精度,对一般的凸优化问题在 $O(1/\varepsilon)$ 步内收敛,对连续可微问题在 $O(\lg(1/\varepsilon))$ 步内收敛. 这个方法的另一个重要特点是,它可以自动利用优化问题的光滑性,它的一个改进是无需求解二次规划问题而拥有同样的收敛速度.

这种方法一个关键技术点是次梯度. 次梯度是凸函数梯度的一个推广,其中凸函数可以是非光滑的. 假设 $\boldsymbol{w}$ 是凸函数 F 的一个函数值有限点,那么次梯度是在 $\boldsymbol{w}$ 点 F 的任何正切支撑超平面的标准法向量. 确切地,$\boldsymbol{\mu}$ 被称为 F 在 $\boldsymbol{w}$ 点的次梯度,当且仅当

$$\forall \boldsymbol{w}', F(\boldsymbol{w}') \geqslant F(\boldsymbol{w}) + <\boldsymbol{w}'-\boldsymbol{w}, \boldsymbol{\mu}>,$$

F 在一点 $\boldsymbol{w}$ 的所有次梯度的集合称为 F 在这点的次微分,表示为 $\partial_{\boldsymbol{w}}F(\boldsymbol{w})$. 若这个集合非空,那么称 F 在 $\boldsymbol{w}$ 点次可微. 若这个集合只有一个元素,称 F 在 $\boldsymbol{w}$ 点可微. 用 $\boldsymbol{x}\in\boldsymbol{X}, y\in Y$ 分别表示训练样本的输入模式和输出标签,$l(\boldsymbol{x},y,\boldsymbol{w})$ 是凸损失函数且 $\boldsymbol{w}\in\boldsymbol{W}$,$\boldsymbol{W}$ 是再生核希尔伯特空间. 对给定的一组观测样本 $\boldsymbol{x}_i\in R^n, y_i\in R, i=1,\cdots,l$,正则风险最小化函数可表示为

$$J(\boldsymbol{w}) := R_{\text{emp}}(\boldsymbol{w}) + \lambda\Omega(\boldsymbol{w}),$$

$$R_{\text{emp}}(\boldsymbol{w}) := \frac{1}{m}\sum_{i=a}^{m} l(\boldsymbol{x}_i, y_i, \boldsymbol{w}).$$

$\Omega(\boldsymbol{w})$ 是光滑的凸正则项,$\lambda>0$ 是正则系数,:= 表示按定义等于.

令 $\boldsymbol{w}_i\in\boldsymbol{W}$ 表示在每次迭代中 $\boldsymbol{w}$ 的取值,并且令 $\boldsymbol{a}_t\in\boldsymbol{W}$,$b_t\in R$,$\boldsymbol{w}_0=\boldsymbol{0}$,$\boldsymbol{a}_0=\boldsymbol{0}$,$b_0=0$,则 $R_{\text{emp}}[\boldsymbol{w}_t]$ 的泰勒展开系数为

$$\boldsymbol{a}_{t+1} := \partial_{\boldsymbol{w}} R_{\text{emp}}(\boldsymbol{w}_i),$$

$$b_{i+1} := R_{\text{emp}}(\boldsymbol{w}_t) - <\boldsymbol{a}_{t+1}, \boldsymbol{w}_t>.$$

因为一次泰勒逼近是逼近函数的下界,则

$$R_{\text{emp}}(\boldsymbol{w}) \geqslant \max_t <\boldsymbol{a}_t, \boldsymbol{w}>.$$

定义 R_{emp} 和 J 的下界为

$$R_t(\boldsymbol{w}) := \max_{t'\leqslant t} <\boldsymbol{a}_{t'}, \boldsymbol{w}> + b_{t'},$$

$$J_t(\boldsymbol{w}) := \lambda\Omega(\boldsymbol{w}) + R_t(\boldsymbol{w}).$$

且对所有 $t'\leqslant t$ 构造 $R_{t'}\leqslant R_t\leqslant R_{\text{emp}}$,$J_{t'}\leqslant J_t\leqslant J$,通过定义:

$$\boldsymbol{w}^* = \operatorname{argmin}_{\boldsymbol{w}} J(\boldsymbol{w}), \boldsymbol{w}_t := \operatorname{argmin}_{\boldsymbol{w}} J_t(\boldsymbol{w}),$$

$$\gamma_t := J_{t+1}(\boldsymbol{w}_t) - J(\boldsymbol{w}_t), \varepsilon_t t := \min_{t'\leqslant t} J_{t'+1}(\boldsymbol{w}_{t'}) - J_t(\boldsymbol{w}_t).$$

则可得到如下结论,对所有 $t'\leqslant t$ 有如下关系成立:

$$J_{t'}(\boldsymbol{w}_{t'}) \leqslant J_t(\boldsymbol{w}_t) \leqslant J(\boldsymbol{w}^*) \leqslant J(\boldsymbol{w}_t) = J_{t+1}(\boldsymbol{w}_t).$$

进而,ε_t 是单调减的且

$$\varepsilon_t - \varepsilon_{t+1} \geqslant J_{t+1}(\boldsymbol{w}_{t+1}) - J_t(\boldsymbol{w}_t) \geqslant 0.$$

另外 ε_t 为到最优点距离的上界:

$$\gamma_t \geqslant \varepsilon_t \geqslant \min_{t'\leqslant t} J(\boldsymbol{w}_{t'}) - J(\boldsymbol{w}^*)$$

在主空间支持向量机的线性算法可简单描述如下:

初始化: $t=0, \boldsymbol{w}_0=\boldsymbol{0}, \boldsymbol{a}_0=\boldsymbol{0}, b_0=0, J_0(\boldsymbol{w})=\lambda\Omega(\boldsymbol{w})$

While　$\varepsilon_t\leqslant\varepsilon$

　　取得最小值 $\boldsymbol{w}_t := \operatorname{argmin}_{\boldsymbol{w}} J_t(\boldsymbol{w})$

　　计算次梯度 $\boldsymbol{a}_{t+1}$ 和偏移量 b_{t+1}

　　$t\leftarrow t+1$

End

此外,Smola[63] 还给出了在对偶空间 SVM 的线性算法,并由经验数据得出结论:在对偶空间执行精确的线性搜索,则在主空间的目标函数有更快的收敛速度.

3　支持向量机的扩展算法

随着对 SVM 研究的深入,人们提出了一些 SVM 的扩展算法. 这些扩展算法主要是通过增加函数项、变量或系数等方法使公式变形,产生出有某一方面优势,或者有一定应用范围的算法,如 v-SVM[64-65],广义 SVM(generalized SVM, GSVM)[66-67],最小二乘 SVM(least-square SVM, LS-SVM)[68-69] 等. v-SVM 算法中用参数 v 取代 C,v 是间隔错误样本的个数所占总样本点数的份额的上界,也是支持向量的个数所占总样本点数的份额的下界. 参数 v 对于各种噪声具有较好的鲁棒性,易于选择,并且可用以控制支持向量的数目和误差. 广义 SVM 直接以优化系数和核矩阵构造出一个不等式约束的非线性优化问题,其对偶形式与标准 SVM 对偶形式等价. 但广义 SVM 并不是直接求解此优化问题或其对偶形式,而是构

造出若干特例:光滑 SVM、近似 SVM、简化 SVM 等. LS-SVM 算法主要是为了解决计算复杂性问题,它采用二次损失函数,并用等式约束来代替标准 SVM 算法中的不等式约束,把标准 SVM 算法的二次规划问题转变成了线性方程组来求解.

此外还有加权 SVM(weighted SVM)[70-71]模糊 SVM(fuzzy SVM)[72-73]、鲁棒 SVM(robust SVM)[74-75],积极学习 SVM(active SVM)[76]、中心 SVM(center SVM)[77]、并行 SVM(parallel SVM)[78]、多类 SVM(multi-class SVM)[79]、几何 SVM(geometric SVM)[80]、转导半监督 SVM(transductive semi-supervised SVM)[81-82]等.

4 结束语

统计学习理论系统地研究了机器学习问题,尤其是在有限样本情况下的统计学习问题. 这一理论框架下产生的 SVM 是一种通用的机器学习新方法,在理论和实际应用中表现出很多优越的性能. SVM 算法的理论与应用均取得了长足的进步,但在处理有大量训练数据的实际应用中,仍然存在计算速度和存储容量等问题. 该领域需要进一步发展和完善,研究方向包括:

1) 更高效的算法. 训练算法的收敛速度和计算所需内存是 SVM 发展的瓶颈,设计更快、更小的高效算法一直是 SVM 算法研究的主要目标.

2) 更符合实际情况的假设. 现有方法基本上是基于数据的独立同分布假设,而很多实际情况,如非线性动态系统的建模,显然不满足这个条件. 开发非独立同分布假设情况下的算法,将具有更大的实用价值.

3) 统一框架的建立. SVM 的各种在线算法之间,以及 SVM 与 Logistic 回归、条件随机域、Lsso 类方法估计、p 范数方法、稀疏表示等方法之间的内在联系是怎样的,如何建立统一的模型和研究理论体系也是值得研究的方向.

当前的 SVM 缩减方法基本上是采用一步优化策略的贪婪算法,这不但增加了算法的计算量,而且难以得到整体最优解. 基于基寻踪原理的稀疏表示,是一种采用 p 范数作为度量函数的全局竞争优化算法,在一定条件下稀疏表示获得的解是最稀疏的解,同时也是待逼近函数的本质驱动源. 另外,次梯度在训练过程中表现出良好的收敛速度,且不要求目标函数可微. 因此,基于次梯度和稀疏表示的方法可望在进一步提高 SVM 训练速度方面发挥重要作用. 随着理论的不断发展和完善,SVM 在模式识别、回归分析、生物信息技术、医学研究以及其他一些领域,必将得到更加广泛的应用.

参考文献:

[1] VAPNIK V. The nature of statistical learning theory[M]. New York:Springer Verlag,2000.

[2] CRISTIANINI N, SHAWE T J. An introduction to support vector machine [M]. New York: Cambridge University Press, 2000.

[3] DOUMPOS M, ZOPOUNIDIS C. Additive support vector machines for pattern classification[J]. IEEE Trans on Systems, Man, and Cybernetics:Part B, 2007, 37(3): 540-550.

[4] JAYADEVA, KHEMCHANDANI R, CHANDRA S. Twin support vector machines for pattern classification[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2007, 29(5): 905-910.

[5] WU Zhili, LI Chunhung, JOSEPH K, et al. Location estimation via support vector regression[J]. IEEE Trans on Mobile Computing, 2007, 6(3): 311-321.

[6] HAO Peiyi, CHIANG J H. Fuzzy regression analysis by support vector learning approach[J]. IEEE Trans on Fuzzy Systems, 2008, 16(2): 428-441.

[7] CHANG C C, HSU C W, LIN C J. The analysis of decomposition methods for support vector machines[J]. IEEE Trans on Neural Networks, 2000, 11 (4): 1003-1008.

[8] LIN C J. On the convergence of the decomposition method for support vector machines[J]. IEEE Trans on Neural Networks, 2001, 12(6): 1288-1298.

[9] NORIKAZU T, TETSUO N. Global convergence of decomposition learning methods for support vector machines[J]. IEEE Trans on Neural Networks, 2006, 17 (6): 1362-1369.

[10] LIN C J. A Formal analysis of stopping criteria of decomposition methods for support vector machines[J]. IEEE Trans on Neural Networks, 2002, 13(5): 1045-1052.

[11] HU W J, SONG Q. An accelerated decomposition algorithm for robust support vector machines[J]. IEEE Trans on Circuits and Systems, 2004, 51(5): 234-240.

[12] DONG Jianxiong, ADAM K, CHING Y S. Fast SVM training algorithm with decomposition on very large data sets[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2005, 27(4): 603-618.

[13] QIAO Hong, WANG Yanguo, ZHANG Bo. A simple decomposition algorithm for support vector machines with polynomial-time convergence[J]. Pattern Recognition, 2007, 40(9): 2543-2549.

[14] 周水生, 詹海生, 周利华. 训练支持向量机的 Huber 近

似算法[J]. 计算机学报, 2005, 28(10): 1664-1670.
ZHOU Shuisheng, ZHAN Haisheng, ZHOU Lihua. A Huber approximation method for training the support vector machines[J]. Chinese Journal of Computers, 2005, 28(10): 1664-1670.

[15]业　宁, 孙瑞祥,董逸生. 多拉格朗日乘子协同优化的SVM快速学习算法研究[J]. 计算机研究与发展, 2006, 43(3): 442-448.
YE Ning, SUN Ruixiang, DONG Yisheng. SVM fast training algorithm research based on multi-lagrange multiplier [J]. Journal of Computer Research and Development, 2006, 43(3): 442-448.

[16]杨晓伟, 路　节, 张广全. 一种高效的最小二乘支持向量机分类器剪枝算法[J]. 计算机研究与发展, 2007, 44(7): 1128-1136.
YANG Xiaowei , LU Jie , ZHANG Guangquan. An effective pruning algorithm for least squares support vector machine classifier[J]. Journal of Computer Research and Development, 2007, 44(7): 1128-1136.

[17]PLATT J C. Fast training of support vector machines using sequential minimal optimization[C]//Advances in Kernel Methods - Support Vector Learning. Cambridge, MA: MIT Press, 1999: 185-208.

[18]KEERTHI S, SHEVADE S, BHATTCHARYYA C, et al. Improvements to Platt's SMO algorithm for SVM classifier design[J]. Neural Computation, 2001, 13(3): 637-649.

[19]KEERTHI S, GILBERT E. Convergence of a generalized SMO algorithm for SVM classifier design[J]. Machine Learning, 2002, 46(1/2/3): 351-360.

[20]LIN C J. Asymptotic convergence of an SMO algorithm without any assumptions[J]. IEEE Trans on Neural Networks, 2002, 13(1): 248-250.

[21]SMOLA A, SCHÖLKOPF B. A tutorial on support vector regressions[J]. Statistics and Computing, 2004, 14(8): 199-222.

[22]SHEVADE S K, KEERTHI S S, BHATTACHARYYA C. Improvements to SMO algorithm for SVM regression[J]. IEEE Trans on Neural Networks, 2000, 11(5): 1188-1193.

[23]FLAKE G W, LAWRENCE S. Efficient SVM regression training with SMO[J]. Machine Learning, 2002, 46(1/2/3): 271-290.

[24]VOGT M. SMO algorithms for support vector machines without bias term[R]. Darmstadt, Germany: Institute of Automatic Control Laboratory for Control Systems and Process Automation, Technische Univ. Darmstadt, 2002.

[25]KEERTHI S S, SHEVADE S K. SMO algorithm for least squares SVM formulations[J]. Neural Computation, 2003, 15(2): 487-507.

[26]ZENG Xiangyan, CHEN Xuewen. SMO-based pruning methods for sparse least squares support vector machines [J]. IEEE Trans on Neural Networks, 2005, 16(6): 1541-1546.

[27]NORIKAZU T, TETSUO N. Rigorous proof of termination of SMO algorithm for support vector machines[J]. IEEE Trans on Neural Networks, 2005, 16(3): 774-776.

[28]GUO J, TAKAHASHI N, NISHI T. A novel sequential minimal optimization algorithm for support vector regression [C]//Proceedings of the 13th International Conference on Neural Information Processing. Hong Kong, China, 2006, 4234: 827-836.

[29]TAKAHASHI N, GUO J, NISHI T. Global convergence of SMO algorithm for support vector regression[J]. IEEE Trans on Neural Networks, 2008, 19(6): 971-982.

[30]CAO L J, KEERTHI S S, ONG C J. Developing parallel sequential minimal optimization for fast training support vector machine[J]. Neurocomputing, 2006, 70(3): 93-104.

[31]CAO L J, KEERTHI S S, ONG C J, et al. Parallel sequential minimal optimization for the training of support vector machines[J]. IEEE Trans on Neural Networks, 2006, 17(4): 1039-1049.

[32]CHEN P H, FAN R E, LIN C J. A study on SMO-type decomposition methods for support vector machines[J]. IEEE Trans on Neural Networks, 2006, 17(4): 893-908.

[33]BO Liefeng, JIAO Licheng, WANG Ling. Working set selection using functional gain for LS-SVM[J]. IEEE Trans on Neural Networks, 2007, 18(5): 1541-1544.

[34]孙　剑, 郑南宁, 张志华. 一种训练支撑向量机的改进贯序最小优化算法[J]. 软件学报, 2002, 13(10): 2007-2012.
SUN Jian, ZHENG Nanning, ZHANG Zhihua. An improved sequential minimization optimization algorithm for support vector machine training[J]. Journal of Software, 2002, 13(10): 2007-2012.

[35]李建民,张　钹,林福宗. 序贯最小优化的改进算法[J]. 软件学报, 2003,14(5): 918-924.
LI Jianmin, ZHANG Bo, LIN Fuzong. An improvement algorithm to sequential minimal optimization[J]. Journal of Software, 2003, 14(5): 918-924.

[36]张浩然, 韩正之. 回归支持向量机的改进序列最小优化学习算法[J]. 软件学报, 2003, 14(12): 2006-2013.
ZHANG Haoran, HAN Zhengzhi. An improved sequential minimal optimization learning algorithm for regression support vector machine[J]. Journal of Software, 2003, 14(12): 2006-2013.

[37]朱齐丹,张　智,邢卓异. 支持向量机改进序列最小优

化学习算法[J]. 哈尔滨工程大学学报,2007,28(2):183-188.
ZHU Qidan, ZHANG Zhi, XING Zhuoyi. Improved SMO learning method of support vector machine [J]. Journal of Harbin Engineering University,2007,28(2):183-188.

[38]张浩然, 汪晓东. 回归最小二乘支持向量机的增量和在线式学习算法[J]. 计算机学报, 2006, 29(3):400-406.
ZHANG Haoran, WANG Xiaodong. Incremental and online learning algorithm for regression least squares support vector machine[J]. Chinese Journal of Computers, 2006, 29(3): 400-406.

[39]杨 静,张健沛,刘大昕. 基于多支持向量机分类器的增量学习算法研究[J]. 哈尔滨工程大学学报,2006,27(1):103-106.
YANG Jing, ZHANG Jianpei, LIU Daxin. Research on incremental learning algorithm with multiple support vector machine classifiers [J]. Journal of Harbin Engineering University,2006,27(1):103-106.

[40]汪 辉. 增量型支持向量机回归训练算法及在控制中的应用[D]. 杭州: 浙江大学, 2006.
WANG Hui. Incremental support vector machine regression training algorithm and its application in control [D]. Hangzhu: zhejiang University,2006.

[41]MA J H, THEILER J, PERKINS S. Accurate on-line support vector regression[J]. Neural Computation, 2003, 15(11): 2683-2703.

[42]VOJISLAV K, MICHAEL V, HUANG Teming. On the equality of kernel AdaTron and sequential minimal optimization in classification and regression tasks and alike algorithms for kernel machines[C]//European Symposium on Artificial Neural Networks. Bruges, Belgium: D-side Publications,2003: 215-222.

[43]YAAKOV E, SHIE M, RON M. Sparse online greedy support vector regression [C]//Machine learning: ECMI 2002. Berlin: Springer-Verlag,2002:84-96.

[44]KIVINEN J, SMOLA A J, WILLIAMSON R C. Online learning with kernels[J]. IEEE Trans on Signal Processing, 2004, 52(8): 2165-2176.

[45]SMALE S, YAO Y. Online learning algorithms[J]. Foundations of Computational Mathematics, 2006, 6(3): 145-170.

[46]YING Yiming, ZHOU Dingxuan, Online regularized classification algorithms[J]. IEEE Trans on Information Theory, 2006, 52(11): 4775-4788.

[47]BIANCHI N C, CONCONI A, GENTILE C. On the generalization ability of on-line learning algorithms[J]. IEEE Trans on Information Theory, 2004, 50(9): 2050-2057.

[48]BIANCHI N C, GENTILE C. Improved risk tail bounds for on-line algorithms[J]. IEEE Trans on Information Theory, 2008, 54(1): 386-390.

[49]STEINWART I. Sparseness of support vector machines [J]. Journal of Machine Learn Research, 2003, 4(3): 1071-1105.

[50]刘向东, 陈兆乾. 一种快速支持向量机分类算法的研究[J]. 计算机研究与发展, 2004, 41(8): 1327-1332.
LIU Xiangdong, CHEN Zhaoqian. A fast classification algorithm of support vector machines[J]. Journal of Computer Research and Development, 2004, 41(8): 1327-1332.

[51]WU M, SCHÖLKOPF B, BAKIR G. A direct method for building sparse kernel learning algorithms[J]. Journal of Machine Learning Research, 2006, 7(4): 603-624.

[52]NGUYEN D, HO T. An efficient method for simplifying support vector machines[C]//International Conference on Machine Learning. Bonn, Germany, 2005: 617-624.

[53]NGUYEN D. HO T. A bottom-up method for simplifying support vector solutions[J]. IEEE Trans on Neural Networks, 2006, 17(3): 792-796.

[54]KEERTHI S S, CHAPELLE O, DECOSTE D. Building support vector machines with reduced classifier complexity [J]. Journal of Machine Learning Research, 2006, 7(7): 1493-1515.

[55]LI Qing, JIAO Licheng, HAO Yingjuan. Adaptive simplification of solution for support vector machine[J]. Pattern Recognition,2007, 40(3): 972-980.

[56]SUMEET A, SARADHI V V, HARISH K. Kernel-based online machine learning and support vector reduction[J]. Neurocomputing, 2008, 71 (9): 1230-1237.

[57]CRAMMER K, KANDOLA J, SINGER Y. Online classification on a budget[C]//Advances in Neural Information Processing Systems. Whistler, Canada,2003: 225-232.

[58]WESTON J, BORDES A, BOTTOU L. Online (and offline) on an even tighter budget[C]//Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics. Barbados, 2005:413-420.

[59]DEKEL O, SHWARTZ S S, SINGER Y. The forgetron: A kernel-based perceptron on a fixed budget[C]//Advances in Neural Information Processing Systems. Vancouver, Canada,2005: 259-266.

[60]OFER D, YORAM S. Support vector machines on a budget [C]//Advances in Neural Information Processing Systems. Whistler, Canada,2006: 345-352.

[61]JOACHIMS T. Training linear SVMs in linear time[C]//International Conference on Knowledge Discovery and Data Mining(KDD). New York, USA, 2006: 217-226.

[62]SHWARTZ S S, SINGER Y, SREBRO N. Pegasos: Primal estimated sub-gradient solver for SVM[C]//Proceed-

ings of the Twenty-Fourth International Conference on Machine Learning(ICML2007). Corvalis,USA,2007: 807-814.

[63] SMOLA A J,VISHWANATHAN SV N,QUOC V L. Bundle methods for machine learning[C]//Advances in Neural Information Processing Systems. Vancour, Canada, 2008.

[64] CHEN PH,LIN CJ, SCHLKOPF B. A tutorial on v-support vector machines [J]. Applied Stochastic Models in Business and Industry,2005,21(2):111-136.

[65] KAZUSHI I. Effects of kernel function on v-support vector machines in extreme cases[J]. IEEE Trans on Neural Networks,2006,17(1):1-9.

[66] MANGASARIAN O L,WILD E W. Multisurface proximal support vector machine classification via generalized eigenvalues[J]. IEEE Trans on Pattern Analysis and Machine Intelligence,2006,28(1):69-74.

[67] LEE Y J,HUANG S Y. Reduced support vector machines: a statistical theory[J]. IEEE Trans on Neural Networks,2007,18(1): 1-13.

[68] ANTHONY K,PHILIPPE D W. Comments on"pruning error minimization in least squares support vector machines"[J]. IEEE Trans on Neural Network,2007,18(2): 606-609.

[69] JIAO Licheng,BO Liefeng,WANG Ling. Fast sparse approximation for least squares support vector machine[J]. IEEE Trans on Neural Network,2007,18(3):685-697.

[70] WANG Defeng,DANIEL S Y,ERIC C C T,Weighted mahalanobis distance kernels for support vector machines [J]. IEEE Trans on Neural Networks,2007,18(5): 1453-1462.

[71] DU Shuxin,CHEN Shengtan. Weighted support vector machine for classifications[C]//IEEE International Conference on Systems,Man and Cybernetics. Hawaii,USA, 2005,4: 3866-3871.

[72] CAWLEY G C. An empirical evaluation of the fuzzy kemel perceptron[J]. IEEE Trans on Neural Networks,2007, 18(3):935-937.

[73] HAO P Y, CHIANG J H. Fuzy regression analysis by support vector learning approach[J]. IEEE Trans on Fuzzy Systems,2008,16(2):428-441.

[74] CHUANG C C,SU SF,JENGJT,et al. Robust support ector regression networks for function approximation with outliers[J]. IEEE Trans on Neural Networks,2002,13(6): 1322-1330

[75] TRAFALIS T B,GILBERT R C. Robust classification and regression using support vector machines [J]. European Journal of Operational Research,2006,173(3): 893-909.

[76] MITRA P,MURTHY C A,PAL S K. A probabilistic active support vector learning algorithm[J] . IEEE Trans on Pattern Analysis and Machine Intelligence,2004,26(3): 413-418.

[77] TSANG I W,KWOKJT,ZURADA J M. Generalized core vector machines[J]. IEEE Trans on Neural Networks,2006,17(5):1126-1140.

[78] ZANNI L, SERAFINI T,ZANGHIRATI G. Parallel software for training large scale support vector machines on multiprocessor systems[J] . Journal of Machine Learning Research,2006,7(7): 1467-1492.

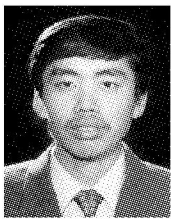
[79] 赵春晖,陈万海,郭春燕.多类支持向量机方法的研究现状与分析[J].智能系统学报,2007,2(4):11-17.
ZHAO Chunhui,CHEN Wanhai,GUO Chunyan. Research and analysis of methods for multiclass support vector machines[J] . CAAI Transactions on Intelligent Systems, 2007,2(4):11-17.

[80] THEODORIDIS S,MAVROFORAKIS M. Reduced convex hulls: a geometric approach to support vector machines [J]. IEEE Signal Processing Magazine,2007,24(3): 119-122.

[81] BRUZZONE L,CHI M,MARCONCINI M. A novel transductive SVMs for semi-supervised classification of remote-ensing images[J]. IEE Trans on Geoscience and Remote Sensing,2006,44(11):3363-3373.

[82] ASTORINO A,FUDULI A. Nonsmooth optimization techniques for semisupervised classification[J]. IEEE Trans on Pattern Analysis and Machine Intelligence,2007,29(12): 2135-2142.

作者简介:



王书舟,男,1972年生,博士研究生,主要研究方向为支持向量机建模、直升机控制与仿真.发表学术论文多篇,6 篇被EI检索.

伞　冶,男,1951年生,教授,博士生导师.中国系统仿真学会理事.主要研究方向为复杂大系统的系统控制与仿真.获国家科技进步二等奖2项,三等奖1项,省部级科技进步奖多项.发表学术论文多篇,40余篇被EI收录.