集成学习:Boosting 算法综述

于 玲 吴铁军

(浙江大学 智能系统与决策研究所 杭州 310027)

摘 要 Boosting 是近年来机器学习领域中一种流行的、用来提高学习精度的算法.本文首先以 AdaBoost 为例对 Boosting 算法进行简单的介绍,并对 Boosting 的各种不同理论分析进行概括,然后介绍了 Boosting 在回归问题中的理论研究,最后对 Boosting 的应用以及未来的研究方向进行了讨论.

关键词 Boosting,机器学习,泛化误差,回归,优化
中图法分类号 TP181

1 引 言

如何根据观测数据学习得到精确估计是机器学习领域中人们非常关注的一个问题,机器学习的一个重要目标就是对新的测试样本尽可能给出最精确的估计.构造一个高精度估计是一件相当困难的事情,然而产生数个只比随机猜测好的粗糙估计却很容易[1].

Boosting 算法的基本思想就是试图通过产生数个简单的、精度比随机猜测略好的粗糙估计(Boosting 算法中称为弱规则 $h_1,\cdots,h_T$),再将这些规则集成构造出一个高精度的估计.这种思想起源于 Valiant 提出的 PAC 学习模型[2].在 PAC 模型中定义了两个概念——强学习和弱学习.

强学习:令 S 为包含 N 个数据点$(x_1,y_1),\cdots,(x_N,y_N)$的样本集,其中 x_n 是按照某种固定但未知的分布 $D(x)$ 随机独立抽取的.$y_n=f(x_n)$,f 属于某个已知的布尔函数集 F.如果对于任意的 D,任意的 $f\in F$,任意的$0\leqslant\varepsilon,\delta\leqslant 1/2$,学习算法生成一个满足 $Pr[h(x)\neq f(x)]\leqslant\varepsilon$ 的估计 h 的概率大于 $1-\delta$,并且学习算法的运行时间与 $1/\varepsilon,1/\delta$ 成多项式关系.则称这种学习算法为强学习算法.

弱学习:其定义与强学习算法类似,只是弱学习算法中只需存在某对 ε,δ 满足上述条件即可.

收稿日期:2003-07-10

Kearns 和 Valiant 提出了弱学习算法与强学习算法间的等价问题[3],即是否能把弱学习算法转化为强学习算法?如果两者等价,那么只要找到一个比随机猜测略好的弱学习算法就可以直接将其提升为强学习算法,而不必直接去找很难获得的强学习算法.在文献[4]中 Kearns 和 Valiant 证明只要有足够的数据,弱学习算法就能通过集成的方式生成任意高精度的估计.

1990 年,Schapire[5]最先构造出一种多项式级的算法,即最初的 Boosting 算法.这种算法可以将弱分类规则转化成强分类规则.一年后,Freund[6]提出了一种效率更高的 Boosting 算法.1993 年,Drucker 和 Schapire[7]第一次以神经网络作为弱学习器,应用 Boosting 算法来解决实际的 ORC 问题.由于早期的 Boosting 算法在解决实际问题时要求事先知道弱学习算法学习正确率的下限,这实际上很难做到.1995 年,Freund 和 Schapire 提出了 Adaboost(Adaptive Boosting)算法[8],这种算法的效率和原来 Boosting 算法的效率一样,但不需要任何关于弱学习器性能的先验知识,因此可以非常容易地应用到实际问题中.

AdaBoost 算法提出后在机器学习领域得到极大的关注,试验结果显示无论是应用于人造数据还是真实数据,AdaBoost 都能显著提高学习精度.但是最

近的研究[9,10]表明AdaBoost算法具有某些缺陷,例如它对噪声非常敏感.研究者分别从不同角度对AdaBoost算法进行理论分析,并针对这些问题提出了不同的改进算法,其中最具代表性的改进算法包括实AdaBoost算法[11],BrownBoost算法[12],LogitBoost算法[13]以及Arcing算法[14].

Boosting算法从提出到理论研究以及众多的试验都是针对于分类问题的,有关Boosting在回归问题上的应用研究和理论研究相对较少.近年来也出现了一些有关Boosting在回归问题中的研究[13,15],并获得了较好的结果.

2 Boosting算法及其理论分析

目前各种不同的Boosting算法有很多,但最具代表性的当属AdaBoost算法.而且各种不同Boosting算法都是在AdaBoost算法的基础上发展起来的,因此下面我们以AdaBoost算法为例对Boosting算法进行简单的介绍.

2.1 AdaBoost算法介绍

AdaBoost算法的主要思想是给定一个训练集$(x_1,y_1),\cdots,(x_m,y_m)$,其中$x_i$属于某个域或者实例空间$X$,$y_i\in\{-1,+1\}$.初始化时AdaBoost指定训练集上的分布为$1/m$,并按照该分布调用弱学习器对训练集进行训练.每次训练后,根据训练结果更新训练集上的分布,并按照新的样本分布进行训练.反复迭代T轮,最终得到一个估计序列$h_1,\cdots,h_T$,每个估计都具有一定的权重,最终的估计H是采用有权重的投票方式获得.AdaBoost算法的伪代码如图1所示,详细的算法说明参见文献[8]和文献[11].

输入:训练集$S=\{(x_1,y_1),\cdots,(x_m,y_m)\}$,其中$x_i\in X$,$y_i\in\{-1,+1\}$;迭代次数$T$和弱学习器

初始化:权重$D_1(i)=1/m$, $i=1,\cdots,m$

执行:for $t=1,\cdots,T$

1) 对有权重分布的训练集学习,得到一个估计h_t:$x\mapsto\{-1,+1\}$

2) 计算h_t训练偏差$\varepsilon_t=Pr_{i\sim D_t}[h_t(x_i)\neq y_i]$,如果$\varepsilon_t=0$或者$\varepsilon_t\geqslant 1/2$,令$T=t-1$并跳出循环

3) 令:$\alpha_t=0.5\times\ln[(1-\varepsilon_t)/\varepsilon_t]$

4) 更新权重:$D_{t+1}(i)=\{D_t(i)\exp[-\alpha_t y_i h_t(x_i)]/Z_t$,其中$Z_t$是标准化因子

输出:$H(x)=sign\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right)$

图1　AdaBoost算法

上面给出的算法是针对于两分类问题的,关于如何将Boosting算法应用到多分类问题研究者提出了多种不同的方法.最直接的应用于多分类问题的方法就是Freund和Schapire在文献[8]中提出的AdaBoost.M1算法,但是这种方法常常会因为弱学习器不能达到50%的精度而失败.针对这种情况研究者又提出了不同的方法,这些方法通常是将多类问题转化成一个大的两类问题.在文献[11]中所采取的方法AdaBoost.MH是将一个多类问题转化为一系列两分类问题,即对每个样本点x都判断它是否属于某个类别y.在文献[8]中所提到的算法AdaBoost.M2也是基于类似的思想,只是它总是判断样本点x的正确类别是y还是y'.另外还有一些其它的方法,例如借助于偏差校正输出代码方法[16]来识别多分类问题,也能达到与AdaBoost.MH相同的效果[1].

2.2 Boosting算法的理论分析

Schapire、Singer和Freund[8,11]首先从理论上推导出了最终分类规则的训练误差,定义$f(x)=\sum_t\alpha_t h_t(x)$,则有$H(x)=\mathrm{sign}(f(x))$,可以推导出:

$$\frac{1}{m}|\{i:H(x_i)\neq y_i\}|\leqslant\frac{1}{m}\sum_i\exp(-y_i f(x_i))=\prod_i Z_t. \tag{1}$$

从(1)式中看出我们通过选择每一轮的α_t和h_t来最小化Z_t,使训练误差迅速减小.在文献[11]中Schapire和Singer对AdaBoost算法进行修改,将原算法中估计h_t的值域从$\{-1,+1\}$扩展为$[-1,+1]$,即用实数函数替代原来的二值函数,并讨论了在这种情况下如何选择α_t和h_t的问题.对于这种情况,我们把h_t的符号看作是h_t预测的类别,绝对值看作是h_t预测的置信度.这种改进的算法又称作实AdaBoost算法,到目前为止大多数应用都是采用这种算法.

大量的试验[10,14,17,18-20]表明Boosting算法不但可以提高学习精度,而且不易过配.如图2所示为文献[19]中Boosting C4.5对UCI中的样本"letter"进行分类,得到的相对于迭代次数T的误差曲线和类间距分布图.左图中上面一根曲线表示泛化误差,下面一根曲线表示训练误差.

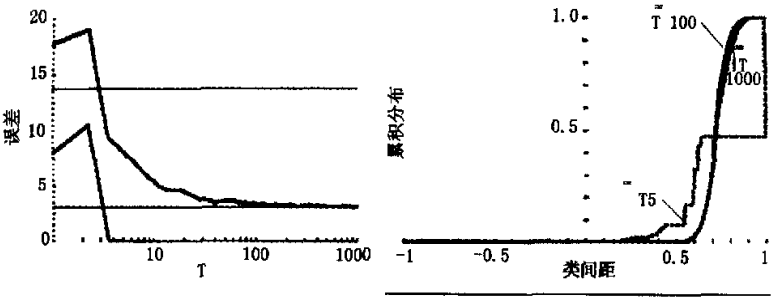

图2 Boosting C4.5对数据库"letter"进行分类的误差曲线和类间距分布图

从左图中可以看到，当训练误差达到零后Boosting仍然会继续降低泛化误差，并没有因为迭代次数的增多出现泛化性能恶化的情景.这个现象很奇怪，似乎与机器学习中通常所遵循的"Occam's Razor"原则[21]相违背.为了解释这一现象，Schapire等[19]从类间距(margin)角度对泛化误差进行了解释.定义样本点(x,y)的类间距：

$$\mathrm{margin}_f(x,y)=\frac{yf(x)}{\sum_t|\alpha_t|}=\frac{y\sum_t\alpha_t h_t(x)}{\sum_t|\alpha_t|},\quad(2)$$

它的值属于$[-1,+1]$之间，只有当H对样本正确分类时类间距的值才会大于0.类间距的值可被看作是对H预测结果的置信度，值越大则预测出来的结果越可信.Schapire等推导出训练集的类间距与泛化误差间的关系，即泛化误差最大为

$$\hat{P}r[margin(x,y)\leqslant\theta]+\tilde{O}\left(\sqrt{\frac{d}{m\theta^2}}\right),\quad(3)$$

其中d表示弱学习算法的VC维[22-24]，为m训练实例个数.注意公式中给出的泛化误差上界与T无关，即表示迭代次数增多不会影响泛化误差，这与左图曲线相吻合.从试验中观测到Boosting对类间距的影响如图2中右图所示，当训练误差降为0后，Boosting仍然继续提高训练样本的类间距，因此使泛化误差能够继续下降.公式(3)给出AdaBoost算法成功的原因是由于对最小类间距实现最大化.

Breiman[9]对最大化最小类间距和泛化误差间的关系进行了试验分析，指出当有噪声存在时两者之间不存在必然的关系.Grove在文献[25]中指出用Schapire提出的类间距理论对Boosting的泛化性能进行解释存在缺陷.为此Grove构造了"LPboosting"算法，这种算法与AdaBoost相比能够找到更大的最小类间距，但试验表明这种算法的泛化性能往往比AdaBoost差.越来越多的文献[9,10]指出当不存在噪声时Boosting算法能取得很好的泛化性能，而当存在较大噪声时，Boosting会产生过配现象[26].

对AdaBoost算法的进一步分析研究[9,27]表明，它实际上是在最小化一个与类间距分布有关的指数函数$\sum_{i=1}^{m}\exp(-margin(x_i,y_i))$，采用的方法是有约束的梯度下降求解.随着迭代次数的增多，算法的重心将转移到难分类的样本点上，即难分类样本点的权值会成指数增长，这也是为什么Boosting算法会有效降低泛化误差的原因.但是如果样本点存在噪声或者样本点错误，相比较而言这些错误的或者存在噪声的样本点更难分类，因此随着迭代的增加它们的权值就会成指数增长，造成Boosting算法的性能下降.这可能就是Boosting对噪声过于敏感的原因[28].由此一些研究者对Boosting算法进行改进[26,29]，用其它形式的函数取代指数函数以减小Boosting算法对噪声的敏感度.

Bätsch在文献[26]和文献[30]中指出AdaBoost算法实际上和SVM算法是一类算法.Boosting算法最终得到的类间距与SVM算法所得到的类间距类似，只是它们对类间距的定义和权向量的优化方法不同，SVM对类间距的定义采用的是2范数形式而Boosting采用的是1范数形式.SVM算法与Boosting一样也对噪声相当敏感，文献[26]中针对SVM算法的这一缺陷提出了改进的方法，即用软类间距代替原来的类间距，结果表明改进后的泛化性能大大提高.Bätsch将这种方法应用到Boosting算法中，采用RBF网络作为弱学习器，对13个人造的和真实的数据集合进行试验，结果显示改进算法的泛化性能明显提高.

在文献[13]中，Friedman等从统计学的角度对Boosting算法进行了研究，指出Boosting算法实际上

是对一种特殊的指数型损失函数进行优化的算法.其他研究者[31]也从不同角度发现Boosting算法与优化算法具有相关性.Friedman还推证出Boosting中所采用的这种指数型损失函数与柏努利似然函数非常相似,并由此提出了一种新的Boosting算法——LogitBoost算法.这种算法是直接对柏努利似然函数进行最大最小优化.这项工作的重要性在于它把计算学习理论和传统的概率统计理论联系起来[32],使得统计理论中的众多模型可应用于计算学习领域中了.

3　Boosting与拟合问题

机器学习主要是应用于两大类问题:分类和回归.一般说来分类是要将样本点映射到一个离散集合,而回归是要将样本点映射到一个连续的域中.因此相比较而言回归比分类更困难.Boosting算法从提出至今主要是应用于分类问题,有关它的理论研究也是针对分类问题开展的.有关Boosting算法如何应用于回归问题的讨论较少,有关应用于回归问题的试验也很少见,更不用说在回归情况下的理论分析了.

Freund和Schapire最先提出可应用于回归问题的AdaBoost.R算法[8],这种算法实际上是将回归问题转化为两分类问题然后调用AdaBoost算法.给定回归用的训练样本$(x_1,y_1),\cdots,(x_m,y_m)$,其中$y_i\in[-1,1]$,AdaBoost.R算法首先将每个样本点$(x_i,y_i)$转化成一个包含无限数据对的集合$((x_i,z),\tilde{y}_i)$,其中$z\in[-1,1]$且$\tilde{y}_i=\mathrm{sign}(y_i-z)$.这样就将一个回归问题转换成为一个两分类问题,然后调用AdaBoost算法.Drucker对AdaBoost.R算法进行了微小的改进并首先应用于实际的回归问题.Ridgeway等[33]也提出将回归问题映射成分类问题,生成分类器后再将分类器重新转换成回归函数的方法.试验显示[33,34]这类方法具有一定的效果,但它仍然存在着缺陷.首先这类方法需要将每个原始样本点转化成多个数据对,使得计算量大大增加.而且这类算法中所采用的损失函数在每轮的迭代中都不同[15].

Duffy和Helmbold在文献[15]对回归和分类进行了细致的比较和研究,从梯度下降的角度出发,提出了三种可以直接应用于回归问题的Boosting型算法:SquareLev.R、SquareLev.C和Explev.这些算法都遵循相同的模式,只是采用迭代梯度下降方法对不同的损失函数进行求解.Duffy和Helmbold还从理论上对这三种算法的性能进行了分析并推导出它们的泛化误差边界,但是遗憾的是没有对这些算法进行仿真研究和试验比较.

在文献[13]中Friedman指出回归实际上是个函数空间的优化问题,对于任意一个函数估计问题,我们都是希望寻找一个使损失函数$\Psi(y,G)$的期望最小的回归函数$\hat{G}(x)$:

$$\hat{G}(x)=\arg\min_{F(x)}E_{y,x}\Psi(y,G(x))$$
$$=\arg\min_{F(x)}E_x[E_{y|x}\Psi(y,G(x))\mid x].$$

与以往的参数回归模型不同,这里是直接对函数进行更新,即通过加入一个新的函数$g(x)$来更新当前的函数估计.令$G_i=G(x_i)$,目标函数变为

$$J(G)=\sum_{i=1}^{m}\Psi(y_i,G(x_i))=\sum_{i=1}^{m}\Psi(y_i,G_i).$$

沿$J(G)$的梯度下降方向对G进行更新,有:$\hat{G}\leftarrow\hat{G}-\rho\nabla J(G)$.但是这样求出的$G$函数的泛化性能很差,为此Friedman进行了修改,提出了一种通用的算法——梯度Boosting算法:

① 计算目标函数的负梯度;

② 根据的协变量信息以及负梯度选择一个回归模型$g(x)$;

③ 计算ρ;

④ 更新$\hat{G}(x)\leftarrow\hat{G}(x)-\rho g(x)$.

Friedman还针对不同的损失函数给出了不同的算法实现的伪代码,并对这些算法进行了一些仿真研究.仿真结果显示这类算法确实能有效提高拟合精度并且取得了较好的泛化性能.

4　Boosting的试验和应用

Boosting自从提出后就得到很广泛的应用.Drucker等[7]甚至在AdaBoost算法提出之前就尝试将Boosting技术与神经网络相结合应用到光学字符识别(OCR)问题.文献[7]中采用多层前向神经网络作为弱学习器,网络的隐层数以及隐层节点数随着数据库的不同而不同,对四种不同的手写体数据库,分别是来自美国邮电部门(USPS)的12 000个邮政编码、来自美国国家标准和技术协会(NIST)的220 000个数字、45 000个大写字母以及45 000个小写字母,经过3次迭代学习后采用投票的方式生成一个集成的分类器.试验结果显示集成的分类器与单个神经网络相比识别效果大大提高,测试误差最小降低了17%,最大降低了43%.

AdaBoost算法提出之后,研究者采用不同的弱

学习器对 Boosting 的性能进行测试,并与其它集成方法进行比较. Drucker[17]选择 Quinlan 的 C4.5 算法作为弱学习器,对 NIST 数据库中的样本进行分类.文中没有将数据库中的所有样本点都拿来学习和测试,而是先对样本进行过滤,选出较难的进行学习和测试.随着难度的提高,需要集成的决策树就越多,与单个决策树相比误差降低得就越多.而且还发现随着集成的决策树增多,测试误差渐近下降,从来没有出现过增大的情况.这也是研究者首次发现 Boosting 算法不易过配的特性. Quinlan[10]以 C4.5 作为弱学习器进行 10 次迭代,以 UCI 提供的 27 个数据库为对象,对 Boosting 和 Bagging[35]进行了比较.结果显示单个决策树的平均误差为 15.66%, Bagging 的平均误差为 14.11%, Boosting 的平均误差为 13.36%.由此看出 Boosting 和 Bagging 都能够明显减小泛化误差,并且 Boosting 的平均效果比 Bagging 的好,但是对于某些数据库 Bagging 的效果更好.随后 Freund 和 Schapire[18]也选择 C4.5、FindAttrTest 和 FindDecRule 等三种不同的算法作为弱学习器,以 UCI 的 27 个数据库为对象,其中部分数据库与文献[10]中所采取的数据库相同,对 AdaBoost 和 Bagging 经过 100 次迭代后的效果进行比较.试验结果显示当采用 C4.5 作为弱学习器时, Boosting 平均减小误差 24.8% 而 Bagging 平均减小误差 20%, Boosting 略好于 Bagging;当采用更简单的算法 FindAttrTest 和 FindDecRule 时 Boosting 的效果更显著,平均减小误差分别为 55.2% 和 53.0%,而 Bagging 分别减少 11.0% 和 18.8%.在文献[18]中 Freund 和 Schapire 还以近邻分类器作为弱学习器,对 USPS 的 12 000 个手写体数字进行分类,其中 9 709 个数字用于训练,2 007 个数字用于测试.迭代 30 次后的测试误差为 6.4%,单纯采用近邻分类器的测试误差达到 8.6%,误差降低 26%.在这次试验中还观测到经过 13 次迭代后训练误差就已经为 0 了,然后随着迭代次数从 13 次增加到 30 次,测试误差继续从 8.1% 下降到 6.4%.

Schwenk 和 Bengio[20]采用两层前向神经网络为弱学习器,以一个含有 300 个样本点的手写体数字集为样本,对 Boosting 与神经网络结合的性能进行了测试.对于隐层节点数为 10、30、50 等三种不同的网络结构,分别迭代 100 次,误差分别从 8.8%、3.3%、2.8% 降至 2.9%、1.6%、1.5%.同样文中也对 Boosting 和 Bagging 进行了比较,结果显示两种性能相差不大,作者认为是数据量不够造成的.最后文中还以 UCI 的 Letter 数据库和 satellite 数据库为对象进行试验,误差分别为 1.4%、1.5% 和 8.1%,比采用决策树作为弱学习器要好. Drucker[36]也采用多层前向神经网络为弱学习器,对 USPS 的 12 000 个手写体数字进行分类试验,其中 9 709 个数字用于训练,2 007 个数字用于测试,并比较了 Boosting 和 Bagging 的性能.类似的试验还很多,文献[37]中以决策树和贝叶斯网络为弱学习器,以 UCI 的 14 个数据库为 14 个样本,每个数据库至少包含 1 000 个样本点,对 Boosting、Bagging 以及 Arc-x4 在迭代 25 次的情况下的性能进行了比较.文献[38]中对 AdaBoost、Logiboost 和 Brownboost 在有噪声情况下的性能进行了比较.

Drucker[34]以回归树作为弱学习器,第一次对 Boosting 和 Bagging 在回归问题上的性能进行了比较.文中选用四个回归对象,分别是 Friedman[39]提出的包含 10 个自变量的非线性预测问题 Friedman #1、包含 4 个自变量的 Friedman #2、#3 以及来自 UCI 的包含 506 个样本点的 Boston 数据集,其中中间两个对象的信噪比为 3:1.对于前三个问题,选择 200 个训练样本点、40 个修剪样本点以及 5 000 个测试样本点,结果显示在大多数情况下 Boosting 都优于 Bagging.对于最后一个问题,选择 401 个训练样本点、80 个剪枝样本点和 25 个测试样本点,重复进行 100 次试验中有 72 次试验 Boosting 的测试误差比 Bagging 小,并且求出 Boosting 的平均误差为 10.7%,而 Bagging 的平均误差为 12.4%.由这四个试验可以看出就回归问题来说, Boosting 能够更有效地降低误差.在文献[36]中 Drucker 还以神经网络作为弱学习器,对这四个回归问题进行了试验并得出了相同的结论.

Boosting 算法对预测精度的显著改善一直深深吸引着广大研究者,自从它在 ORC 问题上成功应用之后,越来越多的研究者将它应用于其它领域,包括图像识别和检索[40,41]、文本识别[42-47]、自然语言对话处理[48-51]、医疗诊断[52-55]等过程.近年来将 Boosting 算法应用得最多的当属文本过滤和自然语言处理了,AT&T 实验室开发了一套自动语音客户服务系统[44-46],这个系统首先必须正确识别出语音并转换成相应的文本,然后根据文本得出语义,最后根据语义进行相关应答.由于计算机不懂得语义和句法,这个过程的难点在于将语音正确转换成文本以及根据文本得出正确的意思. Schapire 和 Singer 开发了一套基于 Boosting 的系统 BoosTexter[45],利用该系统来理解文本的意思并获得了成功.文献[48]~文献[51]中也分别将 Boosting 与自回归神经

网络、动态贝叶斯网络等结合起来应用于语音识别系统中的语音识别部分. Boosting也陆续应用于医疗诊断中,文献[54]中将Boosting与模糊分类器结合起来应用于肺癌的诊断,文献[55]中将Boosting与决策树一起应用于皮肤癌的诊断,在文献[52]和文献[53]中自主开发一套LCDS系统,该系统将Boosting与神经网络结合起来应用于肺癌细胞识别,并已投入实际应用. Boosting在其它方面的应用也很多,例如文献[40]中将Boosting和自己设计的查询学习器相结合应用于图片检索过程,文献[41]中将近邻分类器与Boosting相结合应用于快速人脸识别,文献[56]中与简单的阈值函数结合应用于网络上的声音文件检索,文献[57]中与决策树结合用于地理信息系统(GIS)中的风险建模,文献[58]中Boosting与前向神经网络一起作为电子鼻的数据分析部分,文献[59]中用Boosting和RBF网络一起对家用电器的运行情况进行检测,以帮助电力公司对电力的波峰和波谷进行调整.

5 结束语

目前Boosting算法是机器学习中比较热的一个研究方向[60],获得了越来越广泛的关注.本文中对Boosting算法的理论以及应用做了简要的介绍. Boosting算法实际上属于计算学习理论的一部分,它使得人们只要找到一个精度略好于随机猜测的弱学习算法就能大幅提高预测精度,从而促进了机器学习成果的广泛应用.

Boosting算法具有很多优点:首先它简单易用,除了迭代次数T以外不需要调节任何参数;其次它不需要先验知识;最后它还具有理论支持,只要有足够多的数据以及弱学习器就能达到任意预测精度.当然Boosting算法也有一些缺点:它过于依赖数据和弱学习器,对数据噪声很敏感,如果弱学习器过弱也不能达到任意高的精度.

目前Boosting还有很多方面值得研究.例如在存在噪声的情况下Boosting的泛化性能会降低,而实际应用中或多或少都会存在噪声,如何能够降低Boosting对噪声的敏感性是一项关键技术;目前有关Boosting的研究和应用越来越多,但是在回归中的理论研究和应用非常少,实际应用还没有发现.而回归是机器学习领域中很重要的一块,有着非常重大的理论价值和实际意义,亟待进一步的研究;在实际应用过程中必然会存在许多先验知识,如果能将先验知识与Boosting算法结合起来一定会取得更好的效果,如何将先验知识和Boosting相结合也是一个值得研究的问题;目前对Boosting的泛化性能的理论解释并不尽人意,有些试验显示类间距与泛化误差之间的对应关系并不明确,因此如何找到一种更合理的解释就显得非常重要了;一些理论研究表明Boosting和SVM优化理论之间存在着一定的关联,如何将它们进一步统一起来也有待更深入的研究和改进.

参考文献

[1] Schapire R E. The Boosting Approach to Machine Learning: An Overview. In: Proc of the Mathematical Sciences Research Institute (MSRI) Workshop on Nonlinear Estimation and Classification. Berkeley, California, 2001, 149-172

[2] Valiant L G. A Theory of the Learnable. Communications of the ACM, 1984, 27(11): 1134-1142

[3] Kearns M. The Computational Complexity of Machine Learning. Cambridge: MIT Press, 1990

[4] Kearns M, Valiant L G. Cryptographic Limitations on Learning Boolean Formulae and Finite Automata. Journal of the ACM, 1994, 41(1): 67-95

[5] Schapire R E. The Strength of Weak Learnability. Machine Learning, 1990, 5(2): 197-227

[6] Freund Y. Boosting a Weak Learning Algorithm by Majority. Information and Computation, 1995, 121(2): 256-285

[7] Drucker H, Schapire R E, Simard P. Boosting Performance in Neural Networks. International Journal of Pattern Recognition and Artificial Intelligence, 1993, 7(4): 705-719

[8] Freund Y, Schapire R E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. Journal of Computer and System Sciences, 1997, 55(1): 119-139

[9] Breiman L. Arcing the Edge. Technical Reprot, No. 486, Statistics Department, University of California, Berkeley, 1997

[10] Quinlan J R. Bagging, Boosting, and C4.5. In: Proc of 14th National Conference on Artificial Intelligence. Portland, Oregon, 1996, 725-730

[11] Schapire R E, Singer Y. Improved Boosting Algorithms Using Confidence-Rated Predictions. Machine Learning, 1999, 37(3): 297-336

[12] Freund Y. An Adaptive Version of the Boost by Majority Algorithm. Machine Learning, 2001, 43(3): 293-318

[13] Friedman J H, Hastie T, Tibshirani R. Additive Logistic Regression: A Statistical View of Boosting. Annals of Statistics, 2000, 28(2): 337-374

[14] Breiman L. Arcing Classifiers. Annals of Statistics, 1998, 26(3): 801-849

[15] Duffy N, Helmbold D. Boosting Methods for Regression. Machine Learning, 2002, 47(2-3): 153-200

[16] Dietterich T G, Bakiri G. Solving Multiclass Learning Problems via Error-Correcting Output Codes. Journal of Artificial Intelligence

Research, 1995, 2: 263-286

[17] Drucker H, Cortes C. Boosting Decision Trees. In: Touretzky D S, Mozer M C, Hasselmo M E, Cortes C. eds. Advances in Neural Information Processing. Cambridge: MIT Press, 1996, Ⅷ: 479-485

[18] Freund Y, Schapire R E. Experiments with a New Boosting Algorithm. In: Proc of the 13th International Conference on Machine Learning. San Francisco, CA, 1996, 148-156

[19] Schapire R E, Freund Y, Bartlett P, Lee W S. Boosting the Margin: A New Explanation for the Effectiveness of Voting Methods. *The Annals of Statistics*, 1998, 26(5): 1651-1686

[20] Schwenk H, Bengio Y. Boosting Neural Netwroks. Neural Computation, 2000, 12(8): 1869-1887

[21] Angluin D. Computational Learning Theory: Survey and Selected Bibliography. In: Proc of the 24th Annual ACM Symposium on Theory of Computing. Vancouver, 1992, 351-369

[22] Vapnik V N. Statistical Learning Theory. New York: John Wiley & Sons, Inc, 1998

[23] Vapnik V N. An Overview of Statistical Learning Theory. IEEE Trans on Neural Networks, 1999, 10(5): 988-999

[24] Vapnik V N,著;张学工,译.统计学习理论的本质.北京:清华大学出版社,2000

[25] Grove A J, Schuurmans D. Boosting in the Limit: Maximizing the Margin of Learned Ensembles. In: Proc of the 15th National Conference on Artificial Intelligence (AAAI-98). Madison, WI, 1998, 692-699

[26] Rätsch G, Onoda T, Müller K R. Soft Margins for AdaBoost. *Machine Learning*, 2001, 42(3): 287-320

[27] Friedman J H. Greedy Function Approximation: A Gradient Boosting Machine. Annals of Statistics, 2001, 29(5): 1189-1232

[28] Duffy N, Helmbold D. Potential Boosters? In: Advances in Neural Information Processing. USA: MIT Press, 2000, Ⅻ: 258-264

[29] Duffy N, Helmbold D. A Geometric Approach to Leveraging Weak Learning. In: Proc of the 4th European Conference on Computational Learning Theory. Nordkirchen, Germany, 1999, 18-33

[30] Rätsch G, Schölkopf B, Mika S, Müller K R. SVM and Boosting: One Class. Technical Report, No. 119, GMD FIRST, Berlin, 2000

[31] Breiman L. Prediction Games and Arcing Algorithms. Neural Computation, 1999, 11(7): 1493-1517

[32] Ridgeway G. The State of Boosting. Computing Science and Statistics, 1999, 31: 172-181

[33] Ridgeway G, Madigan D, Richardson T. Boosting Methodology for Regression Problems. In: Proc of the 7th International Workshop on Artificial Intelligence and Statistics. Fort Lauderdate, Florida, 1999, 152-161

[34] Drucker H. Improving Regressors Using Boosting Techniques. In: Fisher D H, ed. Proc of the 14th International Conference on Machine Learning. USA: Morgan Kaufmann, 1997, 107-115

[35] Breiman L. Bagging Predictors. Technical Repoort, No. 421, Statistics Department, University of California, Berkeley, 1994

[36] Drucker H. Boosting Using Neural Nets. In: Sharkey A J C, ed. Combining Artificial Neural Nets: Ensemble and Modular Learning. London: Springer, 1999, 51-77

[37] Bauer E, Kohavi R. An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting and Variants. Machine Learning, 1999, 36(1-2): 105-139

[38] McDonald R A, Hand D J, Eckley A. An Empirical Comparison of Three Boosting Algorithms on Real Data Sets with Artificial Class Noise. In: Proc of Multiple Classifier Systems Workshop 2003. Guilford, UK: Springer-Verlag, 2003, 35-44

[39] *Friedman J H. Multivariate Adaptive Regression Splines.* Annals of Statistics, 1991, 19(1): 1-82

[40] Tieu K, Viola P. Boosting Image Retrieval. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). Hilton Head Island, South Carolina, 2000, Ⅰ: 228-235

[41] Guo G D, Zhang H J. Boosting for Fast Face Recognition. In: Proc of 2nd International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems. Vancouver, Canada, 2001, 96-100

[42] Abney S, Schapire R E, Singer Y. Boosting Applied to Tagging and PP Attachment. In: Proc of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora. New Brunswick, NJ, 1999, 38-45

[43] Rochery M, Schapire R E, Rahim M, Gupta N. BoosTexter for Text Categorization in Spoken Language Dialogue. In: Automatic Speech Recognition and Understanding Workshop. Madonna di Campiglio Trento, Italy, 2001. Available at http://www.cs.princeton.edu/~schapire/publist.html

[44] Rochery M, Schapire R, Rahim M, Gupta N, Riccardi G, Bangalore S, Alshawi H, Douglas S. Combining Prior Knowledge and Boosting for Call Classification in Spoken Language Dialogue. In: Proc of International Conference on Accoustics, Speech and Signal. Orlando, Florida. 2002. Available at http://www.cs/princeton.edu/~schapire/whatsnew.html

[45] Schapire R E, Singer Y. BoosTexter: A Boosting-Based System for Text Categorization. Machine Learning, 2000, 39(2-3): 135-168

[46] Schapire R E, Rochery M, Rahim M, Gupta N. Incorporating Prior Knowledge into Boosting. In: Proc of the 19th International Conference on Machine Learning. Sydney, 2002, 538-545

[47] Schwenk H, Bengio Y. AdaBoosting Neural Networks: Application to On-Line Character Recognition. In: Proc of the International Conference on Artificial Neural Networks (ICANN'97). Lausanne, Switzerland: Springer-Verlag, 1997, 967-972

[48] Schwenk H. Using Boosting to Improve a Hybrid HMM/Neural Network Speech Recognizer. In: Proc of the IEEE International Conference on Acoustics, Speech, and Signal (ICASSP 99). Phoenix, Arizona, 1999, Ⅱ:1009-1012

[49] Myers K, Kearns M, Singh S, Walker M A. A Boosting Approach to Topic Spotting on Subdialogues. In: Proc of the 17th International Conference on Machine Learning. San Francisco: Morgan Kaufmann, 2000, 655-662

[50] Choudhury T, Rehg J M, Pavlović V, Pentland A. Boosting and Structure Learning in Dynamic Bayesian Networks for Audio-Visual Speaker Detection. In: Proc of the International Conference on Pattern Recognition. Quebec City, Canada, 2002, Ⅲ: 789－794

[51] Cook G, Robinson T. Boosting the Performance of Connectionist Large Vocabulary Speech Recognition. In: Proc of International Conference on Spoken Language Processing. Philadelphia, 1996. Available at http://svr－www.eng.cam.ac.uk/gdc/papers/icslp96－preprint.ps.Z

[52] 姜　远,周志华,谢　琪,陈兆乾.神经网络集成在肺癌细胞识别中的应用.南京大学学报(自然科学), 2001, 37(5): 529－533

[53] Zhou Z H, Jiang Y, Yang Y B, Chen S F. Lung Cancer Cell Identification Based on Artificial Neural Network Ensembles. Artificial Intelligence in Medicine, 2002, 24(1): 25－36

[54] Hoffmann F. Boosting a Genetic Fuzzy Classifier. In: Proc of the Joint 9th International Fuzzy Systems Association World Congress and 20th International Conference of North American Fuzzy Information Processing Society. Vancouver, Canada, 2001, 1564－1569

[55] Merler S, Furlanello C, Larcher B, Sboner A. Tuning Cost-Sensitive Boosting and Its Application to Melanoma Diagnosis. In: Kittler J, Roli F, eds. Multiple Classifier Systems, 2nd International Workshop. MCS Cambridge: Springer-Verlag, 2001, 32－42

[56] Moreno P, Logan B, Raj B. A Boosting Approach for Confidence Scoring. In: Proc of 7th European Conference on Speech Communication and Technology (Eurospeech). Aalborg, 2001, 2109－2112

[57] Furlanello C, Merler S. Boosting of Tree-Based Classifiers for Predictive Risk Modeling in GIS. In: Kittler J, Roli F, eds. Multiple Classifier Systems, Lecture Notes in Computer Science 1857. New York: Springer-Verlag, 2000, 220－229

[58] Masulli F, Pardo M, Sberveglieri G, Valentini G. Boosting and Classification of Electronic Nose Data. In: Proc of 3rd International Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science. New York: Springer-Verlag, 2002, 262－271

[59] Onoda T, Rätsch G, Müller K R. A Non-Intrusive Monitoring System for Household Electric Appliances with Inverters. In: Proc of International Computer Science Conventions Symposium on Neural Computation (NC). Berlin, 2000. Available at http://mlg.anu.edu.au/~raetsch/Publications.html

[60] Dietterich T G. Machine Learning Research: Four Current Directions. AI Magazine, 1999, 18(4): 97－136

ASSEMBLE LEARNING: A SURVEY OF BOOSTING ALGORITHMS

Yu Ling, Wu Tiejun

(*Institute of Intelligent System and Decision Making, Zhejiang University, Hangzhou* 310027)

ABSTRACT

Boosting is a general method for improving the accuracy of any given learning algorithm. This paper primarily introduces the AdaBoost algorithm and explains the different underlying theory of boosting, then describes some theoretical analyses in regression. In the end some recent applications and future research issues are present.

Key Words　Boosting, Machine Learning, Generalization Error, Regression, Optimization