基于指数损失和 0-1 损失的在线 Boosting 算法

侯 杰 [1] 茅耀斌 [1] 孙金生 [1]

摘 要 推导了使用指数损失函数和 0-1 损失函数的 Boosting 算法的严格在线形式, 证明这两种在线 Boosting 算法最大化样本间隔期望、最小化样本间隔方差. 通过增量估计样本间隔的期望和方差, Boosting 算法可应用于在线学习问题而不损失分类准确性. UCI 数据集上的实验表明, 指数损失在线 Boosting 算法的分类准确性与批量自适应 Boosting (AdaBoost) 算法接近, 远优于传统的在线 Boosting; 0-1 损失在线 Boosting 算法分别最小化正负样本误差, 适用于不平衡数据问题, 并且在噪声数据上分类性能更为稳定.

关键词 AdaBoost, 在线学习, 特征选择, 不平衡数据

引用格式 侯杰, 茅耀斌, 孙金生. 基于指数损失和 0-1 损失的在线 Boosting 算法. 自动化学报, 2014, **40** (4): 635−642

DOI 10.3724/SP.J.1004.2014.00635

Online Boosting Algorithms Based on Exponential and 0-1 Loss

HOU Jie[1] MAO Yao-Bin[1] SUN Jin-Sheng[1]

Abstract In this paper, strict derivation for the online form of Boosting algorithms using exponential loss and 0-1 loss is presented, which proves that the two online Boosting algorithms can maximize the average margin and minimize the margin variance. By estimating the margin mean and variance incrementally, Boosting algorithms can be applied to online learning problems without losing classification accuracy. Experiments on UCI machine learning datasets show that the online Boosting using exponential loss is as accurate as batch AdaBoost, and significantly outperforms the traditional online Boosting, and that the online Boosting using 0-1 loss can minimize classification errors of positive samples and negative samples at the same time, thus applies to imbalance data. Moreover, Boosting using 0-1 loss is more robust on noisy data.

Key words AdaBoost, online learning, feature selection, imbalance data

Citation Hou Jie, Mao Yao-Bin, Sun Jin-Sheng. Online Boosting algorithms based on exponential and 0-1 loss. *Acta Automatica Sinica*, 2014, **40**(4): 635−642

Boosting 是模式识别和机器学习领域近十多年来发展起来的一类集成学习 (Ensemble learning) 算法, 可通过投票的方式将概率近似正确 (Probably approximately correct, PAC) 学习模型中的弱学习算法提升成强学习算法[1]. 自适应 Boosting (Adaptive Boosting, AdaBoost) 算法[2−3] 是第一种实用化的 Boosting 算法, 也是最具代表性的 Boosting 算法之一. AdaBoost 算法对训练样本集进行迭代重采样, 以不断选取最优的特征训练弱分类器. AdaBoost 算法被广泛用于解决模式识别领域的特征选择问题[4−6]. 早期实验研究发现, AdaBoost 具有良好的泛化性能, 训练多轮也不易过拟合. 进一步的理论研究表明, AdaBoost 算法采用函数空间的贪婪式梯度下降方法最小化样本间隔 (Margin) 的指数损失的期望[7]. 凭借坚实的理论基础、良好的泛化性能和特征选择特性, AdaBoost 算法在计算机视觉领域得到了广泛的关注和应用[4, 8−10].

近年来, Boosting 算法越来越多地被用来处理在线学习问题[8, 11−12]. 原始 AdaBoost 算法训练时需要对整个训练集进行重采样, 因此不能直接处理在线学习问题. Oza 等[13] 提出通过模拟 AdaBoost 算法的重采样过程来逼近批量 AdaBoost, 给出了一种在线 Boosting 算法, 并证明给定无穷多训练样本时, 使用贝叶斯弱学习器的在线 Boosting 和批量 AdaBoost 收敛到同一个分类器. 然而实际应用中一般很难采集任意多训练样本. 实验表明[13], 样本数量有限时在线 Boosting 的分类准确率明显劣于批量 AdaBoost. 此外, 由于使用固定的弱分类器, Oza 等的算法缺少特征选择特性[11], 这是在线 Boosting 算法相比批量 AdaBoost 算法的另一个不足. 文献 [8] 中, Grabner 等扩展了在线 Boosting 算法, 通过选择子 (Selector) 进行特征选择, 并成功应用于视觉目标跟踪问题.特征选择子被成功地应用

收稿日期 2013-06-05 录用日期 2013-10-25
Manuscript received June 5, 2013; accepted October 25, 2013
国家自然科学基金 (60974129) 资助
Supported by National Natural Science Foundation of China (60974129)
本文责任编委 刘成林
Recommended by Associate Editor LIU Cheng-Lin
1. 南京理工大学自动化学院 南京 210094
1. Institute of Automation, Nanjing University of Science and Technology, Nanjing 210094

于半监督[14]、多示例[15] 在线特征选择问题. 然而, Grabner 等并未给出算法的严格理论分析和分类性能的实验比较.

本文推导了使用指数损失函数和 0-1 损失函数的 Boosting 算法的严格在线形式, 证明这两种 Boosting 算法等价于两个最大化平均样本间隔、最小化样本间隔方差的学习问题. 通过增量估计样本间隔的期望和方差, Boosting 算法可应用于在线学习问题而不损失分类准确性. UCI 数据集上的实验研究表明本文算法的分类准确性与批量 AdaBoost 一样好, 远优于传统的在线 Boosting 算法[13].

1 AdaBoost 算法

给定训练集 $S=\{(\boldsymbol{x}_i,y)|\boldsymbol{x}_i\in X,y_i=\pm1\}_{i=1}^N$, S 是对 $X\times\pm1$ 上分布 D 的一个随机采样, AdaBoost 算法在弱分类器池 $H=\{h_i(\boldsymbol{x})|h_i:X\mapsto\pm1\}_{i=1}^M$ 上学习如下形式的投票分类器:

$$F(\boldsymbol{x})=\sum_{t=1}^{T}\alpha_t f_t(\boldsymbol{x}),\ \ f_t\in H\ 且\ \sum_{t=1}^{T}\alpha_t=1 \quad (1)$$

其中, $\sum_{t=1}^{T}\alpha_t=1$ 保证了 $F(\boldsymbol{x})\in[-1,+1]$, 实际应用中并非必须. 式 (1) 也可写成如下向量形式[16]:

$$F(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{x}),\ \ \|\boldsymbol{w}\|_1=1 \quad (2)$$

其中, $\boldsymbol{H}(\boldsymbol{x})=[h_1(\boldsymbol{x}),h_2(\boldsymbol{x}),\cdots,h_M(\boldsymbol{x})]^{\mathrm{T}}$ 是样本 $\boldsymbol{x}$ 在弱分类器池 H 上的向量表示. AdaBoost 算法通过最小化样本间隔 $yF(\boldsymbol{x})$ 的指数损失来寻找最优分类器 $F^*(\boldsymbol{x})$[7]:

$$F^*=\arg\min_F \mathrm{E}_S[\mathrm{e}^{-yF(\boldsymbol{x})}] \quad (3)$$

机器学习问题可看成一个函数估计问题 [17], 其目标为最小化分布为 D 的总体上的风险泛函 $R(F)=\mathrm{E}_D[L(y,F(\boldsymbol{x}))]$[18]:

$$F^*=\arg\min_F \mathrm{E}_D[L(y,F(\boldsymbol{x}))] \quad (4)$$

其中, $\mathrm{E}_D[\cdot]$ 表示在真实分布 D 上的期望, $L(y,F(\boldsymbol{x}))$ 为刻画分类代价的损失函数, 分类问题常采用 0-1 损失. Friedman 等[17] 证明, AdaBoost 算法使用指数损失函数 $L(y,F(\boldsymbol{x}))=\mathrm{e}^{-yF(\boldsymbol{x})}$, 通过函数空间上的贪婪式梯度下降方法拟合一个加性模型. 由于真实分布 D 未知, 式 (3) 采用经验风险泛函 $R_{\mathrm{emp}}(F)=\mathrm{E}[\mathrm{e}^{-yF(\boldsymbol{x})}],(\boldsymbol{x},y)\in S$ 代替式 (4) 中 $R(F)$.

2 在线 Boosting 算法

本节分别推导指数损失 Boosting 算法 (即 AdaBoost 算法) 和 0-1 损失 Boosting 算法的严格在线形式.

2.1 指数损失在线 Boosting 算法

2.1.1 AdaBoost 算法的等价形式

实验研究[16, 19] 发现, AdaBoost 算法中样本间隔 $yF(\boldsymbol{x})$ 近似服从高斯分布. 最近, Shen 等[16] 从理论角度对此进行了解释.

引理 1. AdaBoost 分类器的样本间隔近似服从高斯分布 (见文献 [16] 中引理 3.1).

Shen 等[16] 证明, AdaBoost 算法近似地选取相互独立的特征. 根据中心极限定理, 这些特征的加权和服从高斯分布, 并由此证明样本间隔也近似服从高斯分布. 在引理 1 的基础上, Shen 等[16] 通过蒙特卡洛积分证明: 若分类器的样本间隔服从高斯分布 $\mathrm{N}(\mu,\sigma^2)$, 则 AdaBoost 算法近似于一个最大化 $\mu-\frac{1}{2}\sigma^2$ 的学习问题. 进而可知 AdaBoost 算法最大化样本间隔的期望, 最小化样本间隔的方差 (见文献 [16] 中定理 3.3). 本文给出该结论的更严格形式, 推导 AdaBoost 算法的一个等价形式.

定理 1. 若样本间隔服从高斯分布 $yF(\boldsymbol{x})\sim\mathrm{N}(\mu,\sigma^2)$, 则 AdaBoost 算法 (4) 等价于:

$$F^*=\arg\min_F \mathrm{e}^{-\mu+\frac{1}{2}\sigma^2} \quad (5)$$

证明. 由 $yF(\boldsymbol{x})\sim\mathrm{N}(\mu,\sigma^2)$ 可知, $p(yF(\boldsymbol{x}))$ 的矩量母函数为 $M(t)=\mathrm{e}^{\mu t+\frac{1}{2}\sigma^2t^2}$. 根据矩量母函数定义, 有 $M(t)=\mathrm{E}_D[\mathrm{e}^{tyF(\boldsymbol{x})}]$, 因此:

$$M(t)=\mathrm{E}_D[\mathrm{e}^{tyF(\boldsymbol{x})}]=\mathrm{e}^{\mu t+\frac{1}{2}\sigma^2t^2} \quad (6)$$

根据式 (6), AdaBoost 算法 (4) 可重写为

$$\begin{aligned}F^*=\ &\arg\min_F \mathrm{E}_D[\mathrm{e}^{-yF(\boldsymbol{x})}]=\\ &\arg\min_F M(-1)=\\ &\arg\min_F \mathrm{e}^{-\mu+\frac{1}{2}\sigma^2}\end{aligned} \quad (7)$$

即 AdaBoost 算法 (4) 严格等价于一个最小化 $\mathrm{e}^{-\mu+\frac{1}{2}\sigma^2}$ 的学习问题. □

由于真实分布 D 未知, 实际应用中通常使用的 AdaBoost 算法形式为式 (3). 使用经验矩量母函数 $\hat{M}(t)=\mathrm{E}[\mathrm{e}^{tyF(\boldsymbol{x})}],(\boldsymbol{x},y)\in S$ 代替式 (6) 中 $M(t)$ 可得:

$$\begin{aligned}&M(t)=\mathrm{e}^{\mu t+\frac{1}{2}\sigma^2t^2}\approx\hat{M}(t)=\mathrm{e}^{\hat{\mu}t+\frac{1}{2}\hat{\sigma}^2t^2}\\ &M(-1)=\mathrm{e}^{-\mu+\frac{1}{2}\sigma^2}\approx\hat{M}(-1)=\mathrm{e}^{-\hat{\mu}+\frac{1}{2}\hat{\sigma}^2}\end{aligned} \quad (8)$$

根据式 (8), AdaBoost 算法 (3) 可重写为

$$\begin{aligned} F^* = & \arg\min_F \mathrm{E}[\mathrm{e}^{-yF(\boldsymbol{x})}]|_{(\boldsymbol{x},y)\in S} = \\ & \arg\min_F \hat{M}(-1) = \\ & \arg\min_F \mathrm{e}^{-\hat{\mu}+\frac{1}{2}\hat{\sigma^2}} \end{aligned} \tag{9}$$

根据 e^{-t} 的单调性, 有:

$$\begin{aligned} F^* = & \arg\min_F [\mathrm{e}^{-yF(\boldsymbol{x})}]|_{(\boldsymbol{x},y)\in S} = \\ & \arg\max_F \hat{\mu} - \frac{1}{2}\hat{\sigma}^2 \end{aligned} \tag{10}$$

由式 (10) 易知, AdaBoost 算法最大化样本间隔的期望 $\hat{\mu}$, 最小化样本间隔的方差 $\hat{\sigma}^2$. Shen 等[16] 指出, 该结论形象直观地从理论角度证明了 AdaBoost 算法通过最大化样本间隔的期望、最小化样本间隔的方差来优化样本间隔的分布, 揭示了 AdaBoost 算法为何具有良好的分类性能.

2.1.2 AdaBoost 算法的在线学习

式 (9) 中的分布参数 $\hat{\mu}$ 和 $\hat{\sigma}$ 可通过 $\boldsymbol{m} = \mathrm{E}[y\boldsymbol{H}(\boldsymbol{x})]|_{(\boldsymbol{x},y)\in S}$ 与 $\Sigma = \Sigma[y\boldsymbol{H}(\boldsymbol{x})]|_{(\boldsymbol{x},y)\in S}$ 计算:

$$\begin{aligned} \hat{\mu} = & \mathrm{E}[yF(\boldsymbol{x})] = \\ & \mathrm{E}[y\boldsymbol{w}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{x})] = \\ & \boldsymbol{w}^{\mathrm{T}}\mathrm{E}[y\boldsymbol{H}(\boldsymbol{x})] = \\ & \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m} \\ \hat{\sigma}^2 = & \hat{\sigma}^2[yF(\boldsymbol{w})] = \\ & \hat{\sigma}^2[y\boldsymbol{w}^{\mathrm{T}}\boldsymbol{H}(\boldsymbol{x})] = \\ & \boldsymbol{w}^{\mathrm{T}}\Sigma[y\boldsymbol{H}(\boldsymbol{w})]\boldsymbol{w} = \\ & \boldsymbol{w}^{\mathrm{T}}\Sigma\boldsymbol{w} \end{aligned} \tag{11}$$

对于在线学习问题, $\boldsymbol{m}$ 和 Σ 可增量估计. 令 $\boldsymbol{z}_i = y_i\boldsymbol{H}(\boldsymbol{x}_i)$, 有:

$$\begin{aligned} \boldsymbol{m}_{n+1} = & \mathrm{E}[\boldsymbol{z}_i]_{i=1}^{n+1} = \\ & \gamma\boldsymbol{m}_n + (1-\gamma)\boldsymbol{z}_{n+1} \\ \Sigma_{n+1} = & \mathrm{E}[\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}]_{i=1}^{n+1} - \boldsymbol{m}_{n+1}\boldsymbol{m}_{n+1}^{\mathrm{T}} = \\ & \gamma\mathrm{E}[\boldsymbol{z}_i\boldsymbol{z}_i^{\mathrm{T}}]_{i=1}^{n} + (1-\gamma)\boldsymbol{z}_{n+1}\boldsymbol{z}_{n+1}^{\mathrm{T}} - \\ & \boldsymbol{m}_{n+1}\boldsymbol{m}_{n+1}^{\mathrm{T}} \end{aligned} \tag{12}$$

其中, γ 为学习率. 由上式可知, 式 (9) 用于处理在线学习问题时, 仅需增量估计期望 $\mathrm{E}[\boldsymbol{z}]$ 与 $\mathrm{E}[\boldsymbol{z}\boldsymbol{z}^{\mathrm{T}}]$. 为了和后文提出的 0-1 损失 Boosting 算法相区别, 本文称式 (9) 为 $\mathrm{Boost}_{\exp}$.

当取学习率 $\gamma = \frac{n}{1+n}$ 时, 有:

$$\begin{aligned} \boldsymbol{m}_{n+1} = & \gamma\boldsymbol{m}_n + (1-\gamma)\boldsymbol{z}_{n+1} = \\ & \frac{n}{n+1}\boldsymbol{m}_n + \frac{1}{n+1}\boldsymbol{z}_{n+1} = \\ & \frac{n}{n+1}\left(\frac{n-1}{n}\boldsymbol{m}_{n-1} + \frac{1}{n}\boldsymbol{z}_n\right) + \frac{1}{n+1}\boldsymbol{z}_{n+1} = \\ & \cdots = \\ & \frac{1}{n+1}\boldsymbol{z}_1 + \frac{1}{n+1}\boldsymbol{z}_2 + \cdots + \frac{1}{n+1}\boldsymbol{z}_{n+1} = \\ & \frac{1}{n+1}\sum_{i=1}^{n+1}\boldsymbol{z}_i \end{aligned} \tag{13}$$

此时对 $\boldsymbol{m}$ 和 Σ 的估计结果与批量计算时相同, 式 (12) 可看成是一种无遗忘的增量统计量估计方法.

2.1.3 算法流程

$\mathrm{Boost}_{\exp}$ 算法学习新样本的过程分为两步:

步骤 1. 根据式 (12) 增量估计 $\boldsymbol{m}$ 和 Σ;

步骤 2. 在 Gradient Boosting 框架下优化式 (9).

Gradient Boosting 框架中, 每轮训练首先选择让准则函数下降最快的分量方向 $\Delta\boldsymbol{w}^*$; 然后选择最优下降步长 α^*; 再将新选择的弱分类器组合到原分类器上, 即: $\boldsymbol{w} = \boldsymbol{w} + \alpha\Delta\boldsymbol{w}^*$. 最优下降方向可通过准则函数 $\Phi(F)$ 的梯度计算:

$$\Delta\boldsymbol{w}^* = \arg\max_{\Delta\boldsymbol{w}} \Delta\boldsymbol{w}^{\mathrm{T}}\left|\frac{\mathrm{d}\Phi}{\mathrm{d}\boldsymbol{w}}\right|, \quad \Delta\boldsymbol{w} \in \{\boldsymbol{e}_i\} \tag{14}$$

其中, $\{\boldsymbol{e}_i\}$ 是 $\mathbf{R}^M$ 上的标准基, $\frac{\mathrm{d}\Phi}{\mathrm{d}\boldsymbol{w}}$ 为准则函数的梯度. $\Delta\boldsymbol{w}^*$ 分量方向上最优下降步长为

$$\alpha^* = \arg\min_{\alpha} \Phi(\boldsymbol{w} + \alpha\Delta\boldsymbol{w}^*) \tag{15}$$

对 $\mathrm{Boost}_{\exp}$ 算法, $\Phi(\boldsymbol{w}) = \mathrm{e}^{-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}+\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{w}}$, 因此准则函数梯度为

$$\frac{\mathrm{d}\Phi}{\mathrm{d}\boldsymbol{w}} = \mathrm{e}^{-\mu+\frac{1}{2}\sigma^2}(-\boldsymbol{m} + \Sigma\boldsymbol{w}) \tag{16}$$

α^* 可通过驻点法求解:

$$\frac{\mathrm{d}\Phi(\boldsymbol{w} + \alpha\Delta\boldsymbol{w}^*)}{\mathrm{d}\alpha} = 0 \tag{17}$$

则 $\alpha^* = \frac{(\Delta\boldsymbol{w}^*)^{\mathrm{T}}\boldsymbol{m} - (\Delta\boldsymbol{w}^*)^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{w}}{(\Delta\boldsymbol{w}^*)^{\mathrm{T}}\boldsymbol{\Sigma}\Delta\boldsymbol{w}^*}$.

接下来给出步骤 2 的具体描述. 算法输入为算法迭代次数 T、均值向量 $\boldsymbol{m}$ 与协方差矩阵 Σ, 算法流程如下:

1) 初始化组合向量 $\boldsymbol{w}$ 为零向量;

2) 计算当前组合向量 $\boldsymbol{w}$ 下, 最佳梯度下降方向 $\Delta\boldsymbol{w}^*$;

3) 计算梯度下降步长 α^*, 并更新组合向量 $\boldsymbol{w} = \boldsymbol{w} + \alpha^*\Delta\boldsymbol{w}^*$;

4) 权向量归一化: $\boldsymbol{w} = \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_1}$;

5) 重复执行步骤 2) ∼ 4) T 轮.

由于 $\boldsymbol{w}$ 和 $\Delta\boldsymbol{w}^*$ 都是稀疏向量 ($M \gg T$), 上述算法在每轮迭代时所需的矩阵乘法运算和二次型求值计算速度都很快. 在完成增量统计量更新后, 算法重新训练一个投票分类器. 采用这种训练方式的主要原因是增量式统计量估计已经起到了增量学习的作用. 并且, 重新训练给出的投票分类器总是与批量训练时相同, 分类性能更可靠.

2.1.4 讨论

Oza 等的在线 Boosting 算法[13] 通过模拟 AdaBoost 的重采样过程来逼近批量 AdaBoost, 并从理论角度给出了其算法收敛到批量 AdaBoost 的条件. Shen 等所证明的近似关系 (文献 [16] 中定理 3.1) 仅能说明式 (7) 是对 AdaBoost 算法的近似, 然而没有给出收敛条件, 难以从理论角度保证其分类性能. 本文定理 1 的主要贡献在于证明了 AdaBoost 算法和式 (7) 的严格等价关系, 从而在理论上保证了本文提出的 $\text{Boost}_{\text{exp}}$ 算法的分类性能与批量 AdaBoost 算法一致.

2.2 0-1 损失在线 Boosting 算法

推论 1. 若分类器的样本间隔服从高斯分布 $yF(\boldsymbol{x}) \sim \text{N}(\mu, \sigma^2)$, 则分类器的分类误差为

$$P\left(\text{sgn}\left(F(\boldsymbol{x})\right) \neq y\right) = P\left(yF(\boldsymbol{x}) < 0\right) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{-\mu}{\sqrt{2\sigma^2}}\right)\right] \tag{18}$$

其中, erf 为高斯误差函数.

证明. $yF(\boldsymbol{x})$ 服从高斯分布, 因此

$$P(yF(\boldsymbol{x}) < \theta) = \int_{-\infty}^{\theta} \frac{1}{\sqrt{2\pi}\sigma} \text{e}^{-\frac{(t-\mu)^2}{\sqrt{2\sigma^2}}} \text{d}t = \frac{1}{2}\left[1 + \text{erf}\left(\frac{\theta - \mu}{\sqrt{2\sigma^2}}\right)\right] \tag{19}$$

代入 $yF(\boldsymbol{x}) < 0$, 可得式 (18). □

定理 2. 若分类器的样本间隔服从高斯分布 $yF(\boldsymbol{x}) \sim \text{N}(\mu, \sigma^2)$, 使用 0-1 损失函数的 Boosting 算法等价于如下学习问题:

$$F^* = \arg\min_F \frac{\mu}{\sigma} = \arg\min_F \text{e}^{-\frac{\mu}{\sigma}} \tag{20}$$

证明. 考虑如下 0-1 损失 Boosting 算法:

$$F^* = \arg\min_F \text{E}[L_{0\text{-}1}(y, F(\boldsymbol{x}))] \tag{21}$$

其中, $L_{0\text{-}1}(y, F(\boldsymbol{x}))$ 定义如下:

$$L_{0\text{-}1}(y, F(\boldsymbol{x})) = \begin{cases} 1, & \text{若 } y \neq \text{sgn}(F(\boldsymbol{x})) \\ 0, & \text{若 } y = \text{sgn}(F(\boldsymbol{x})) \end{cases} \tag{22}$$

代入式 (21) 有:

$$\begin{aligned} F^* = & \arg\min_F \text{E}[L_{0\text{-}1}(y, F(\boldsymbol{x}))] = \\ & \arg\min_F P\left(\text{sgn}(F(\boldsymbol{x}) \neq y)\right) = \\ & \arg\min_F P(yF(\boldsymbol{x}) < 0) = \\ & \arg\min_F \frac{1}{2}\left[1 + \text{erf}\left(\frac{-\mu}{\sqrt{2\sigma^2}}\right)\right] \end{aligned} \tag{23}$$

根据 $\frac{1}{2}\left[1 + \text{erf}(t)\right]$ 的单调性, 上式可整理为

$$\begin{aligned} F^* = & \arg\min_F \frac{-\mu}{\sqrt{2\sigma^2}} = \\ & \arg\max_F \frac{\mu}{\sqrt{2\sigma^2}} = \\ & \arg\max_F \frac{\mu}{\sigma} = \\ & \arg\min_F \text{e}^{-\frac{\mu}{\sigma}} \end{aligned} \tag{24}$$

由式 (23) 与式 (24) 可知, 使用 0-1 损失函数的 Boosting 算法等价于一个最大化 $\frac{\mu}{\sigma}$、最小化 $\text{e}^{-\frac{\mu}{\sigma}}$ 的学习问题. □

在计算机视觉、医疗诊断等领域, 机器学习方法常用于处理稀有事件检测问题[20]. 在这类学习问题中, 正样本数量远远小于负样本数量, 即训练数据是不平衡数据集[21]. 传统的 Boosting 算法训练时最小化训练集上整体误差, 而未考虑正负样本的不平衡性, 往往对正样本的误分率很高[21−23], 在实际应用中容易造成稀有事件的漏检[24]. 针对不平衡数据分类问题, 本文在式 (20) 基础上设计了如下 0-1 损失 Boosting 算法:

$$F^* = \arg\min_F \sum_{c=\pm 1} \text{e}^{-\frac{\hat{\mu}_c}{\hat{\sigma}_c}} \tag{25}$$

其中, $\hat{\mu}_c$ 和 $\hat{\sigma}_c^2$ 是类别为 c 的样本的平均间隔和间隔方差. 根据定理 2, 式 (25) 分别最小化正负样本误差, 从而避免分类器对正样本误差过高. $\hat{\mu}_c$ 和 $\hat{\sigma}_c^2$ 可通过下式计算:

$$\begin{aligned} \hat{\mu}_c = & \text{E}[yF(\boldsymbol{x})]|_{y=c} = \\ & \boldsymbol{w}^{\text{T}}\text{E}[yH(\boldsymbol{x})]|_{y=c} = \\ & \boldsymbol{w}^{\text{T}}\boldsymbol{m}_c \\ \hat{\sigma}_c^2 = & \hat{\sigma}^2[yF(\boldsymbol{w})]_{y=c} = \\ & \boldsymbol{w}^{\text{T}}\Sigma[yH(\boldsymbol{w})]|_{y=c}\boldsymbol{w} = \\ & \boldsymbol{w}^{\text{T}}\Sigma_c\boldsymbol{w} \end{aligned} \tag{26}$$

其中, $\boldsymbol{m}_c$ 和 Σ_c 分别为 c 的样本的均值和协方差矩阵. 式 (25) 分别最小化正负样本的分类误差, 避免了因数据不平衡性造成的正样本分类准确率过低的问题. 式 (25) 中指数函数 e^{-t} 用来保证准则函数的凸性. 根据式 (26), $\boldsymbol{m}_c$ 和 Σ_c 可以通过式 (12) 增量估计, 因此式 (25) 也可作为一种在线 Boosting 算法, 我们称其为 $\mathrm{Boost}_{0\text{-}1}$.

$\mathrm{Boost}_{0\text{-}1}$ 算法 (25) 也可以在 Gradient Boosting 框架下进行优化, 优化过程与 $\mathrm{Boost}_{\exp}$ 算法类似 (见第 2.1.3 节). $\mathrm{Boost}_{0\text{-}1}$ 算法准则函数为 $\Phi(\boldsymbol{w}) = \sum_{c=\pm1} \mathrm{e}^{-\frac{\hat{\mu}_c}{\hat{\sigma}_c}}$, 准则函数梯度为

$$\frac{\mathrm{d}\Phi}{\mathrm{d}\boldsymbol{w}} \sum_{c=\pm1} \mathrm{e}^{-\frac{\hat{\mu}_c}{\hat{\sigma}_c}} \frac{\hat{\sigma}_c \boldsymbol{m}_c - \hat{\mu}_c \Sigma_c \boldsymbol{w}_c}{\hat{\sigma}_c^3} \tag{27}$$

最优下降步长 α^* 可通过线性搜索求取.

$\mathrm{Boost}_{\exp}$ 算法与 $\mathrm{Boost}_{0\text{-}1}$ 算法可看成一类在线 Boosting 算法: 1) 两者都最大化样本间隔的期望, 最小化样本间隔的方差; 2) 两者都通过增量估计样本间隔的方差处理在线学习问题.

3　实验及分析

为了检验本文提出的两种在线 Boosting 算法的有效性, 本文从 UCI 机器学习数据库[25] 中选取 7 个数据集对指数损失在线 Boosting 算法 ($\mathrm{Boost}_{\exp}$) 与 0-1 损失在线 Boosting 算法 ($\mathrm{Boost}_{0\text{-}1}$) 的分类性能进行实验分析, 然后讨论两者在不平衡数据和噪声数据上分类性能的差异.

本文选择以下 7 个 UCI 数据集进行实验: Promoters, Balance-scale, Breast-cancer, German, Car, Chess 和 Mushroom. 数据集描述见表 1.

表 1　数据集

Table 1　Datasets description

名称	属性个数	正/负样本数	正负样本比值
Promoters	57	53/53	1.0
Balance-scale	4	288/288	1.0
Breast-cancer	10	241/458	0.53
German	24	300/700	0.43
Car	6	384/1 210	0.31
Chess	6	1 527/1 669	0.91
Mushroom	22	3 916/4 208	0.93

本文在 Matlab 环境下实现并比较以下五种 Boosting 算法: 批量 AdaBoost 算法 (ADA)、指数损失在线 Boosting 算法 ($\mathrm{OB}_{\exp}$)、0-1 损失在线 Boosting 算法 ($\mathrm{OB}_{0\text{-}1}$)、Oza 等的在线 Boosting 算法 (OB) 以及文献 [16] 中通过列生成 (Column Generation) 最大化 $\hat{\mu} - \frac{1}{2}\hat{\sigma}^2$ 的算法 (CG). 其中对 $\mathrm{OB}_{\exp}$ 和 $\mathrm{OB}_{0\text{-}1}$ 两种在线 Boosting 算法, 实验采用学习率 $\gamma = \frac{n}{n+1}$ 进行无遗忘的增量统计量估计.

3.1　分类准确性

首先比较几种算法在 UCI 数据集上的分类准确率. 实验使用 stump 弱分类器, 采用 5 折交叉验证, 所有算法迭代 100 轮, 分类器在测试数据上的准确率见表 2 (其中粗体表示比 ADA 算法分类准确率高的实验结果). 比较 ADA、CG 与 $\mathrm{OB}_{\exp}$ 三种算法分类准确率发现, 三者在所有 7 个数据集上分类准确率差异不大. 根据定理 1, 三者实际上优化相同的准则函数, 因此上述结果不难解释. Shen 等的进一步实验研究发现, 由于 CG 算法采用了列生成方法, 训练误差收敛速度较传统的 ADA 算法更快, 然而其测试误差收敛速度并未快于 ADA (参见文献 [16] 中图 5). $\mathrm{OB}_{0\text{-}1}$ 算法的分类准确率在 5 个数据集上优于 ADA 算法, 在另两个数据集上与 ADA 接近. 由表 1 可知, 这两个数据集中负样本数量多于正样本, 是不平衡数据集. 本文将在进一步实验中分析 $\mathrm{OB}_{0\text{-}1}$ 算法在这两个不平衡数据集上的性能.

表 2　几种 Boosting 算法在 UCI 数据集上的实验结果

Table 2　Experiment results on UCI datasets

数据集	ADA	CG	$\mathrm{OB}_{\exp}$	$\mathrm{OB}_{0\text{-}1}$	OB
Promoters	0.8500	0.8500	0.8500	**0.8857**	0.6571
Balance-scale	0.9259	0.9125	**0.9845**	**0.9810**	0.9000
Breast-cancer	0.9231	**0.9408**	**0.9385**	**0.9612**	0.8396
German	0.7550	0.7526	0.7450	0.7250	0.7120
Car	0.8621	0.8518	0.8458	0.8389	0.7705
Chess	0.8990	**0.9042**	**0.9216**	**0.9416**	0.7969
Mushroom	0.9828	0.9743	0.9540	**0.9893**	0.9025

比较 $\mathrm{OB}_{\exp}$、$\mathrm{OB}_{0\text{-}1}$ 和 OB 三种在线 Boosting 算法的分类准确性可以得到如下结论: 1) OB 算法的分类性能明显劣于批量 AdaBoost 算法; 2) $\mathrm{OB}_{\exp}$ 和 $\mathrm{OB}_{0\text{-}1}$ 算法分类性能与 ADA 算法接近, 远优于 OB 算法; 3) 采用无遗忘的增量统计量估计时, $\mathrm{OB}_{\exp}$ 和 $\mathrm{OB}_{0\text{-}1}$ 算法给出分类器分类性能与 ADA 算法接近, 可直接代替批量 AdaBoost 算法.

分类准确性与批量 AdaBoost 算法接近, 这是 $\mathrm{OB}_{\exp}$ 和 $\mathrm{OB}_{0\text{-}1}$ 作为在线 Boosting 算法的主要优势. 此外, Oza 等在其论文中指出在线 Boosting 算法的分类性能受样本到达顺序的影响. 采用无遗忘的增量统计量估计时, 本文提出的 $\mathrm{OB}_{\exp}$ 和 $\mathrm{OB}_{0\text{-}1}$

算法显然不受样本到来顺序影响, 更为稳定可靠.

本文进一步研究了样本间隔的期望、方差、分类器理论误差 (式 (18)) 与实际误差之间的关系. 实验选取样本数量较多的数据集 Chess 与 Mushroom, 采用 5 折交叉验证, 所有算法迭代 20 轮, 结果见表 3. 从实验结果可以看到, 三种 Boosting 算法分类误差的理论估计值与实际误差非常接近, 偏差小于 3%, 说明了定理 1 中高斯分布假设的合理性, 并验证了推论 2 的正确性.

表 3 样本间隔期望、方差和分类误差分析

Table 3 Average margin, margin variance, and classification error analysis

		Chess	Mushroom
ADA	样本间隔期望	0.1240	0.1343
	样本间隔方差	0.0285	0.0488
	理论误差	0.2302	0.3077
	实际误差	0.2305	0.2732
OB_{exp}	样本间隔期望	0.1363	0.3624
	样本间隔方差	0.0420	0.1354
	理论误差	0.2530	0.1616
	实际误差	0.2675	0.1498
$OB_{0\text{-}1}$	样本间隔期望	0.2296	0.3156
	样本间隔方差	0.0468	0.0247
	理论误差	0.1446	0.0224
	实际误差	0.1180	0.0483

3.2 不平衡分类问题

表 4 研究了使用不同准则函数的 Boosting 算法对正负样本的分类准确性, 并用粗体标出了正负类别分类准确率差异较大的实验结果. 对指数损失 Boosting, 实验以 ADA 与 OB_{exp} 为代表. 实验所采用的 7 个数据集中有三个不平衡数据集 (即两类数据集中, 一类样本远多于另一类样本): Breast cancer、German 和 Car. 表 4 中实验结果显示, 在数据集 German 和 Car 上 ADA 与 OB_{exp} 算法对负样本的分类准确率很高, 而对正样本的分类准确率非常低, 将大量正样本误分为负样本. 这主要因为 ADA 与 OB_{exp} 算法最小化训练集整体的分类误差. 当负样本远多于正样本时, 负样本是影响分类误差的主要因素, 对正样本误差较高的分类器仍能在整个训练集上表现出很好地分类性能. 因此直接最小化训练集上的分类误差不能很好地处理不平衡数据集. $OB_{0\text{-}1}$ 算法由于分别最小化正负样本的分类误差, 不受样本集不平衡性的影响, 在正样本上也有很好的分类准确率.

3.3 噪声数据上的稳定性分析

AdaBoost 算法的另一个主要问题是对数据噪声敏感. 由于采用了指数损失函数, 其进行重采样时会提高噪声和野点样本的权重, 从而影响分类器稳定性, 造成分类准确性下降. 而 0-1 损失函数能够有效避免这种情况. 本文在 7 个 UCI 数据集上人为加入噪声, 从正负训练样本中各随机选取 5 个样本改变其类别标注, 然后比较 ADA、CG、OB_{exp} 和 $OB_{0\text{-}1}$ 算法四种算法在噪声训练数据上的性能. 实验采用 5 折交叉验证, 所有算法迭代 100 轮, 结果见表 5. 噪声对 4 种算法都有不同程度的影响, 其中使用指数损失函数的 ADA 与其两个变种 (CG 和 OB_{exp}) 所受影响较大, 分类准确性下降显著. 而 $OB_{0\text{-}1}$ 算法所受影响较小, 在多个数据集上分类性能下降小于 5%. 清晰起见, 表 5 以粗体标出了分类性能下降小于 5% 的算法 (以表 2 为参考).

表 4 正负样本准确性 (正样本准确性/负样本准确性)

Table 4 Accuracy on positive samples/negative samples

数据集	ADA		OB_{exp}		$OB_{0\text{-}1}$	
	正样本	负样本	正样本	负样本	正样本	负样本
Promoters	0.8400	0.8600	0.8400	0.8600	0.9273	0.8400
Balance-scale	0.8793	0.9724	0.9862	0.9828	0.9793	0.9828
*Breast cancer	0.8813	0.9648	0.9495	0.9275	0.9582	0.9560
*German	**0.4700**	**0.8771**	**0.3800**	**0.9014**	0.7600	0.7100
*Car	**0.5169**	**0.9719**	**0.3714**	**0.9967**	1.0000	0.7959
Chess	0.9010	0.8970	0.9377	0.9056	0.9534	0.9298
Mushroom	0.9748	0.9907	0.9195	0.9886	0.9793	0.9993

* 号的数据集为非平衡数据集, 粗体字实验结果代表分类器对正负样本分类准确率不平衡.

表 5 噪声训练数据上分类准确率

Table 5 Classification accuracy on noisy training data

数据集	ADA	CG	OB_{exp}	$OB_{0\text{-}1}$
Promoters	0.5429	0.5714	0.5238	0.7429
Balance-scale	0.8345	0.8345	0.7362	**0.9483**
Breast cancer	0.7108	0.7338	0.8288	**0.9571**
German	0.6270	0.6590	**0.7060**	**0.7370**
Car	0.7379	0.7379	0.7498	**0.8364**
Chess	0.7587	0.7562	0.7446	**0.8923**
Mushroom	0.6881	0.6871	0.7471	**0.9514**

4 结论

本文推导了指数损失 Boosting 算法的严格在线形式. 新算法通过增量估计样本间隔的均值和方差来学习新样本, 分类性能与批量 AdaBoost 算法接近, 远优于传统的在线 Boosting 算法. 研究了样本间隔分布与分类器分类误差之间的关系, 设计了分别最小化正负样本误差的 0-1 损失在线 Boosting 算法, 避免了传统 AdaBoost 算法在不平衡数据上正样本分类误差过高的问题. 进一步研究了噪声数据上的分类性能, 发现使用指数损失函数的 Boosting 算法受噪声影响较大, 而 0-1 损失 Boosting 算法在噪声数据上更为稳定. 本文给出的两种在线 Boosting 算法的分类性能与批量 AdaBoost 一样好, 能够代替批量 AdaBoost 算法, 用于处理海量数据的学习问题. 下一步将研究本文算法在目标检测问题中的应用, 设计能够在线学习的目标检测器.

References

1 Freund Y, Schapire R E, Abe N. A short introduction to Boosting. *Journal-Japanese Society for Artificial Intelligence*, 1999, **14**(5): 771−780

2 Freund Y, Schapire R E. A desicion-theoretic generalization of on-line learning and an application to Boosting. *Journal of Computer and System Sciences*, 1997, **55**(1): 119−139

3 Cao Ying, Miao Qi-Guang, Liu Jia-Chen, Gao Lin. Advance and prospects of Adaboost algorithm. *Acta Automatica Sinica*, 2013, **39**(6): 745−758
(曹莹, 苗启广, 刘家辰, 高琳. Adaboost 算法研究进展与展望. 自动化学报, 2013, **39**(6): 745−758)

4 Viola P, Jones M J. Robust real-time face detection. *International Journal of Computer Vision*, 2004, **57**(2): 137−154

5 Zhang C, Zhang Z Y. A Survey of Recent Advances in Face Detection, Technical Report MSR-TR-2010-66, Microsoft Research, Redmond, WA, 2010

6 Wu J X, Rehg J M, Mullin M D. Learning a rare event detection cascade by direct feature selection. [Online], available: http: //papers.nips.cc/paper/2353-learning-a-rare-event-detection-cascade-by-direct-feature-selection.pdf, October 25, 2012

7 Bartlett P, Freund Y, Lee W S, Schapire R E. Boosting the margin: a new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 1998, **26**(5): 1651−1686

8 Grabner H, Grabner M, Bischof H. Real-time tracking via on-line Boosting. In: Proceedings of the 2006 British Machine Vision Conference. Edinburgh, British, 2006, **1**: 47−56

9 Kuo C H, Nevatia R. How does person identity recognition help multi-person tracking? In: Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Providence, RI: IEEE, 2011. 1217−1224

10 Yang B, Nevatia R. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Providence, RI: IEEE, 2012. 1918−1925

11 Grabner H, Bischof H. On-line Boosting and vision. In: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. New York, NY: IEEE, 2006, **1**: 260−267

12 Liu X M, Yu T. Gradient feature selection for online Boosting. In: Proceedings of the 11th IEEE International Conference on Computer Vision (ICCV). Rio de Janeiro: IEEE, 2007. 1−8

13 Oza N C. Online Ensemble Learning [Ph. D. dissertation], The University of California, Berkeley, 2001

14 Grabner H, Leistner C, Bischof H. Semi-supervised on-line Boosting for robust tracking. *Computer Vision-ECCV 2008*. Berlin Heidelberg: Springer, 2008: 234−247

15 Babenko B, Yang M H, Belongie S. Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, **33**(8): 1619−1632

16 Shen C H, Li H X. On the dual formulation of Boosting algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010, **32**(12): 2216−2231

17 Friedman J H. Greedy function approximation: a gradient Boosting machine. *Annals of Statistics*, 2001, **29**(5): 1189−1232

18 Vapnik V. *The Nature of Statistical Learning Theory*. New York: Springer, 2000

19 Wu J, Brubaker S C, Mullin M D, Rehg J M. Fast asymmetric learning for cascade face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008, **30**(3): 369−382

20 Ge Jun-Feng, Luo Yu-Pin. A comprehensive study for asymmetric Adaboost and its application in object detection. *Acta Automatica Sinica*, 2009, **35**(11): 1403−1409
(葛俊锋, 罗予频. 非对称 AdaBoost 算法及其在目标检测中的应用. 自动化学报, 2009, **35**(11): 1403−1409)

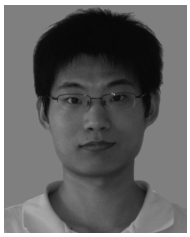
21 He H B, Garcia E A. Learning from imbalanced data. *Transactions on Knowledge and Data Engineering*, 2009, **21**(9): 1263−1284

22 Li Qiu-Jie, Mao Yao-Bin, Wang Zhi-Quan. Research on Boosting-based imbalanced data classification. *Computer Science*, 2011, **38**(12): 224−228
(李秋洁, 茅耀斌, 王执铨. 基于 Boosting 的不平衡数据分类算法研究. 计算机科学, 2011, **38**(12): 224−228)

23 Li Qiu-Jie, Mao Yao-Bin. AUC optimization Boosting based on data rebalance. *Acta Automatica Sinica*, 2013, **39**(9): 1467−1475
(李秋洁, 茅耀斌. 基于数据重平衡的 AUC 优化 Boosting 算法. 自动化学报, 2013, **39**(9): 1467−1475)

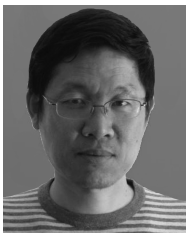
24 Sun Y M, Kamel M S, Wong A K C, Wang Y. Cost-sensitive Boosting for classification of imbalanced data. *Pattern Recognition*, 2007, **40**(12): 3358−3378

25 Frank A, Asuncion A. UC Irvine Machine Learning Repository [Online], available: http://archive.ics.uci.edu/ml, January 18, 2013



侯 杰 南京理工大学自动化学院博士研究生. 主要研究方向为视觉目标检测与跟踪, 模式识别与机器学习.
E-mail: reiase@gmail.com
(**HOU Jie** Ph. D. candidate at the Institute of Automation, Nanjing University of Science and Technology. His research interest covers visual object detection and tracking, pattern recognition, and machine learning.)

茅耀斌 南京理工大学自动化学院副教授. 主要研究方向为图像处理, 计算机视觉和模式识别. 本文通信作者.
E-mail: maoyaobin@163.com
(**MAO Yao-Bin** Associate professor at the School of Automation, Nanjing University of Science and Technology. His research interest covers image processing, computer vision, and pattern recognition. Corresponding author of this paper.)



孙金生 南京理工大学自动化学院教授. 主要研究方向为网络拥塞控制, 质量控制. E-mail: jssun67@163.com
(**SUN Jin-Sheng** Professor at the School of Automation, Nanjing University of Science and Technology. His research interest covers network congestion control and quality control.)