第55卷第2期 南京航空航天大学学报 Vol. 55 No. 2
2023年4月 Journal of Nanjing University of Aeronautics & Astronautics Apr. 2023

DOI:10.16356/j.1005-2615.2023.02.020

代价敏感惩罚AdaBoost算法的非平衡数据分类

鲁淑霞，张振莲，翟俊海

（河北大学数学与信息科学学院，河北省机器学习与计算智能重点实验室，保定 071002）

摘要：针对非平衡数据分类问题，提出了一种基于代价敏感的惩罚AdaBoost算法。在惩罚Adaboost算法中，引入一种新的自适应代价敏感函数，赋予少数类样本及分错的少数类样本更高的代价值，并通过引入惩罚机制增大了样本的平均间隔。选择加权支持向量机（Support vector machine，SVM）优化模型作为基分类器，采用带有方差减小的随机梯度下降方法（Stochastic variance reduced gradient，SVRG）对优化模型进行求解。对比实验表明，本文提出的算法不但在几何均值（G-mean）和ROC曲线下的面积（Area under ROC curve，AUC）上明显优于其他算法，而且获得了较大的平均间隔，显示了本文算法在处理非平衡数据分类问题上的有效性。

关键词：非平衡数据；惩罚AdaBoost；自适应代价敏感函数；平均间隔；随机梯度下降

中图分类号：TP391 **文献标志码：**A **文章编号：**1005-2615(2023)02-0339-08

Imbalanced Data Classification Based on Cost Sensitivity Penalized AdaBoost Algorithm

LU Shuxia, *ZHANG Zhenlian*, *ZHAI Junhai*

(College of Mathematics and Information Science, Hebei Province Key Laboratory of Machine Learning and Computational Intelligence, Hebei University, Baoding 071002, China)

Abstract: How to improve the classification accuracy of minority instances is one of the hot topics in machine learning research. In order to solve the problem of imbalanced data classification, a penalized AdaBoost algorithm based on cost sensitivity is proposed. In the penalized Adaboost algorithm, a new adaptive cost sensitive function is introduced, which gives higher cost value to the minority instances and the misclassified minority instances. It can obtain a larger average margin by introducing penalty mechanism. The weighted support vector machine (SVM) optimization model is used as the base classifier. The stochastic variance reduced gradient (SVRG) with variance reduction method is used to solve the optimization model. The comparative experiments show that the proposed algorithm is not only superior to other algorithms in terms of geometric-mean (G-mean) and area under ROC curve (AUC), but also can obtain a larger average margin by introducing penalty mechanism, which fully demonstrates the effectiveness of the proposed algorithm in handling imbalanced data classification problems.

Key words: imbalanced data; penalized AdaBoost; adaptive cost sensitive function; average margin; stochastic variance reduced gradient (SVRG)

非平衡数据分类问题一直是机器学习研究的热点之一。处理非平衡数据分类问题大致可分为两类：数据级和算法级。数据级方面常采用的方法有欠采样[1]和过采样[2]。欠采样方法的主要思想是不断缩小多数类的规模，直到与少数类的规模相等，但是这样做往往会丢失重要的信息使得分类效

基金项目：河北省科技计划重点研发项目（19210310D）；河北省自然科学基金（F2021201020）。

收稿日期：2021-07-26；**修订日期：**2022-07-29

通信作者：鲁淑霞，女，教授，E-mail：cmclusx@126.com。

引用格式：鲁淑霞，张振莲，翟俊海. 代价敏感惩罚AdaBoost算法的非平衡数据分类[J]. 南京航空航天大学学报，2023，55(2)：339-346. LU Shuxia, ZHANG Zhenlian, ZHAI Junhai. Imbalanced data classification based on cost sensitivity penalized AdaBoost algorithm[J]. Journal of Nanjing University of Aeronautics & Astronautics, 2023, 55(2): 339-346.

果不理想。过采样技术即不断增加少数类的样本，直到样本数与多数类的样本数相等，但由于添加的数据点与现有数据可能会重复，因此会造成过拟合现象。算法级方法尝试应用或改进各种现有的传统学习算法，Zong等[3]针对类不平衡分类问题提出了一种加权的极限学习机。Tao等[4]提出一种集成算法，该算法针对非平衡数据分类问题在样本权重公式中引入自适应代价敏感函数，使分类器更加关注少数类样本。Wang等[5]提出了一种利用新的加权投票参数对AdaBoost算法进行改进的方法。

AdaBoost算法中的样本间隔[6]指单个样本到最终超平面的距离乘以该样本的真实类别标签。在间隔理论的推动发展下，研究人员致力于设计新的AdaBoost算法来优化间隔分布。Rätsch等[7]提出了AdaBoost$_v$算法，该算法用一种适当的方法在每次迭代中更新训练集中的最小间隔值，使其近似逼近最优间隔，从而提高算法的分类精度。SparsiBoost算法[8]是一种稀疏化算法，该算法减少了假设的数量，同时近似地保留了整个间隔分布，产生更好的测试精度。SMLBoost算法[9]通过模拟类似软间隔的过程来提高噪声数据的敏感性，给间隔区域内的无噪声样本分配更高的权重。LPBoost算法[10]使用线性规划直接最大化最小间隔。Gentle AdaBoost算法[11]具有较好的稳定性，但在训练过程中为小间隔的样本分配的权值越来越大，几个大的权重样本主导了整个数据分布，导致过度拟合。Modest AdaBoost算法[12]加强了那些能正确分类小间隔样本的基分类器，该算法优于Gentle AdaBoost算法，但是算法的性能不太稳定。SoftBoost算法[13]通过优化软间隔来优化算法的分类性能，降低泛化误差。Margin-pruning Boost算法[14]采用一种重置技术，提高了样本的平均间隔，避免了Gentle AdaBoost算法出现的分类器失真现象，但是随着迭代次数的增长，该算法的性能变差。这几种算法的性能虽然优于AdaBoost算法，但是引入了复杂的计算，使得训练时间变长，基于间隔分布的提升算法还需要进一步研究。

为了解决非平衡数据分类问题，本文提出一种基于代价敏感的惩罚AdaBoost算法。结合惩罚AdaBoost算法，引入一种自适应代价敏感函数，该函数使得分类器更加关注少数类样本以及分错的少数类样本，且对噪声样本起到抑制作用，提高了少数类样本的分类精度。同时，通过引入惩罚策略增大样本的平均间隔，进一步提高了算法的分类精度。充分考虑数据分布情况，选择加权支持向量机(Support vector machine，SVM)优化模型作为基分类器，即引入间隔均值项，并根据数据非平衡比对间隔均值项和损失函数项进行加权，采用随机梯度下降(Stochastic variance reduced gradient，SVRG)方法对优化模型进行求解，提高了算法的收敛速度。

1 惩罚AdaBoost算法

在每次迭代中，惩罚AdaBoost算法[15]不仅惩罚间隔小的样本的错误分类，而且抑制间隔小的样本的权值增加，避免了过度拟合，提高整个数据集上的间隔值，具有更低的泛化误差。

在两类分类问题中，训练集$S=\{(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)\}$，$x_i\in X\subseteq R^d$，$y_i\in\{-1,1\}$，负类样本为多数类样本，个数为$n_-$，正类样本为少数类样本，个数为$n_+$，$n=n_++n_-$。

AdaBoost算法中的样本x_i间隔为

$$\mathrm{Margin}_T(x_i)=y_i\sum_{t=1}^{T}\alpha_t h_t(x_i)\Big/\sum_{t=1}^{T}\alpha_t \tag{1}$$

式中：基分类器$h_t(x_i)$：$X\to\{-1,1\}$；系数α_t表示基分类器$h_t(x_i)$在最终分类器中的重要性；T为迭代次数。给定样本的分类间隔定义为正确分类的基分类器的预测置信度与错误分类的基分类器的预测置信度之差。

惩罚AdaBoost算法中的样本x_i间隔为

$$\mathrm{Margin}_T(x_i)=y_i\sum_{t=1}^{T}U_t(x_i)\Big/\sum_{t=1}^{T}\left|U_t(x_i)\right| \tag{2}$$

式中：$U_t(x_i)=\alpha_t h_t(x_i)$，等同于AdaBoost算法中第$t$个基分类器在样本$x_i$处的值与其权重的乘积；$\left|U_t(x_i)\right|=\alpha_t$，等同于AdaBoost算法中基分类器权重。$\left|U_t(x_i)\right|$越大，则$U_t(x_i)$的分类能力就越强。

惩罚AdaBoost算法的样本权重为

$$D_i^{t+1}=\exp\left(-y_i\sum_{q=1}^{t}U_q(x_i)\right)\Big/\sum_{i=1}^{n}\exp\left(-y_i\sum_{q=1}^{t}U_q(x_i)\right) \tag{3}$$

惩罚AdaBoost算法使用上一个循环的间隔分布来限制当前循环中间隔较小的样本的误分类，从而降低误分类小间隔样本的基分类器的预测置信度，使其在后续迭代中更容易被分类正确。

对于样本x_i，如果被分为正类，则$U_t(x_i)=U_t^1(x)$，否则$U_t(x_i)=U_t^2(x)$。基分类器的计算公式为

$$U_t^j(x)=\begin{cases}(W_{t+}^j-W_{t-}^j)(1-M_{t-}^j) & W_{t+}^j>W_{t-}^j\\ (W_{t+}^j-W_{t-}^j)(1-M_{t+}^j) & W_{t+}^j\leqslant W_{t-}^j\end{cases} \tag{4}$$

式中：$W_{t+}^j=\sum_{i:x_i\in S_t^j\wedge y_i=1}D_i^t$，$W_{t-}^j=\sum_{i:x_i\in S_t^j\wedge y_i=-1}D_i^t$；$M_{t+}^j=\sum_{i:x_i\in S_t^j\wedge y_i=1}m_i^t$，$M_{t-}^j=\sum_{i:x_i\in S_t^j\wedge y_i=-1}m_i^t$。$m_i^t$为间隔反馈因子，当$t=1$时，$m_i^t=1/n$；否则，$m_i^t=$

$\exp\left(-\text{Margin}_{t-1}(x_i)\right)\Big/\sum_i \exp\left(-\text{Margin}_{t-1}(x_i)\right)$。$S_t^j$为被基分类器分类后的样本数据集，$j\in\{1,2\}$，其中：$S_t^1$为分为正类的样本集；$S_t^2$为分为负类的样本集。$W_{t+}^1$为正确分类的正类样本权重之和；$W_{t+}^2$为错误分类的正类样本权重之和；$W_{t-}^1$为错误分类的负类样本权重之和；$W_{t-}^2$为正确分类的负类样本权重之和。$M_{t+}^1$为正确分类的正类样本的间隔反馈因子之和；$M_{t+}^2$为错误分类的正类样本的间隔反馈因子之和；$M_{t-}^1$为错误分类的负类样本的间隔反馈因子之和；$M_{t-}^2$为正确分类的负类样本的间隔反馈因子之和。

最终分类器为

$$f(x)=\sum_{t=1}^{T}U_t(x) \tag{5}$$

惩罚 AdaBoost 算法中只重置间隔为负且权值超过阈值 Q 的样本，降低了过滤样本的重要性，消除了基分类器对过滤样本错误预测的影响。其次，惩罚 AdaBoost 算法引入了惩罚策略，使用上一个循环的间隔分布来限制当前循环中间隔较小的样本的误分类，提高了整个训练集的样本间隔。惩罚 AdaBoost 算法伪代码如下。

算法 1　惩罚 AdaBoost 算法

输入：训练集 S，参数 τ，步长 η，迭代次数 T

输出：$f(x)=\sum_{t=1}^{T}U_t(x)$，$H(x)=\text{sign}(f(x))$

初始化：数据分布 $D_i^1=1/n$，$i=1,\cdots,n$

(1) for $t=1,\cdots,T$

(2) 基于数据分布 D_i^t 训练基分类器，得到正类样本集 S_t^1，负类样本集 S_t^2，由每一个样本集 S_t^j，$j\in\{1,2\}$，计算 W_{t+}^j 和 W_{t-}^j；

(3) 计算间隔反馈因子 m_i^t；

(4) 计算 M_{t+}^j 和 M_{t-}^j；

(5) 计算基分类器 U_t^j；

(6) 对于样本 x_i，如果被分为正类，$U_t(x_i)=U_t^1(x)$，否则 $U_t(x_i)=U_t^2(x)$；

(7) 更新样本权重：$D_i^{t+1}=\exp\left(-y_i\sum_{q=1}^{t}U_q(x_i)\right)$；

(8) 对于样本 x_i，如果 $D_i^{t+1}>Q_{t+1}$ 且 $\text{Margin}_t(x_i)<0$ 则重置样本权重以及前 t 次迭代的基分类器的和：$D_i^{t+1}=1$，$\sum_{q=1}^{t}U_q(x_i)=0$，

$$Q_{t+1}=\max_i\{D_i^{t+1}\}-\frac{\max_i\{D_i^{t+1}\}-\min_i\{D_i^{t+1}\}}{\tau}$$

对样本权重归一化：$D_i^{t+1}=D_i^{t+1}\Big/\sum_i D_i^{t+1}$；

(9) end for

2　代价敏感惩罚 AdaBoost 算法

惩罚 AdaBoost 算法通过引入惩罚机制和重置技术提高了训练样本的间隔分布，从而增强了算法的分类能力。重置技术是指对间隔为负且样本权重超过阈值 Q 的样本，将其第 $t+1$ 次的样本权重重置为 1，前 t 次迭代的基分类器的和重置为 0，这有效抑制间隔小的样本权重增加，避免了过拟合现象，提高了算法的分类性能。针对非平衡数据分类问题，在惩罚 AdaBoost 算法的基础上，提出了一种新的自适应代价敏感函数，该函数可以根据不同样本的类别以及少数类样本分类的错误与否给予不同的代价值，并且也具有一定的抗噪性。该算法采用加权的 SVM 模型[16-17]作为基分类器，使用 SVRG 方法[18]对加权 SVM 模型求解，在提高算法分类精度的同时也加快了算法的收敛速度。

对于多数类样本，权重大小为 $\mu_i=(n_+/n_-)^p$；对于少数类样本，权重大小为 $\mu_i=(n_-/n_+)^p$，$p\in(0,1]$。考虑数据分布情况，这里采用加权的 SVM 优化模型[16]作为基分类器，即引入间隔均值项，并根据数据非平衡比对间隔均值项和损失函数项进行加权，加权线性 SVM 的优化模型为

$$\min_{\boldsymbol{w}}F(\boldsymbol{w})=\frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}-\lambda_1\sum_{i=1}^{n}\mu_i y_i\boldsymbol{w}^{\mathrm{T}}x_i+\lambda_2\sum_{i=1}^{n}\mu_i\max\{0,1-y_i\boldsymbol{w}^{\mathrm{T}}x_i\} \tag{6}$$

式中：μ_i 为权重；λ_1、λ_2 为折中参数。

对于线性不可分问题，通过非线性映射 $\varphi(x)$ 将原数据映射到高维特征空间，据表示定理获得加权非线性 SVM 优化模型为

$$\min_{\boldsymbol{\alpha}}F(\boldsymbol{\alpha})=\frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{G}\boldsymbol{\alpha}-\lambda_1\sum_{i=1}^{n}\mu_i y_i\boldsymbol{\alpha}^{\mathrm{T}}G_i+\lambda_2\sum_{i=1}^{n}\mu_i\max\{0,1-y_i\boldsymbol{\alpha}^{\mathrm{T}}G_i\} \tag{7}$$

式中：$\boldsymbol{\Phi}=\left[\varphi(x_1),\varphi(x_2),\cdots,\varphi(x_n)\right]$；$\boldsymbol{\alpha}=[\alpha_1,\alpha_2,\cdots,\alpha_n]^{\mathrm{T}}$；$\boldsymbol{G}=\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}$。

在惩罚 AdaBoost 算法的样本权重更新公式中引入自适应代价敏感函数，样本权重更新公式为

$$D_i^{t+1}=C_i^t\exp\left(-y_i\sum_{q=1}^{t}U_q(x_i)\right)\Big/\sum_{i=1}^{n}C_i^t\exp\left(-y_i\sum_{q=1}^{t}U_q(x_i)\right) \tag{8}$$

自适应代价敏感函数为

$$C_i^t=\begin{cases}1 & y_i=-1\\ (n_-/n_+)\left(1+\dfrac{1}{1+\exp\left((1+U_t(x_i))^2\right)}\right) & y_i=1\end{cases} \tag{9}$$

对于负类样本而言，代价值为固定值；而对于正类样本，代价值会根据$U_t(x_i)$的不同而不同，并且大于负类样本的代价值。设定正类样本函数的初始值为1.5，给出对应的正类样本的代价敏感函数图像，如图1所示。图1中横坐标为$U_t(x_i)$，纵坐标C_i^t为第t次迭代的样本x_i的代价值。从图中可以看出，分错的正类样本的代价值大于分对的正类样本的代价值，并且无论是分对的正类样本还是分错的正类样本，其代价值并不会无限增大，这对噪声样本起到了很好的抑制作用。

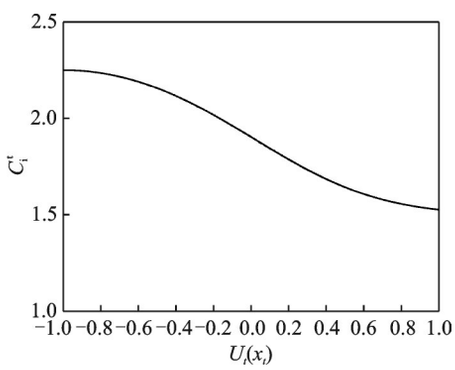


图1 正类样本代价敏感函数图像

Fig.1 Image of cost sensitive function of positive instances

基于代价敏感的惩罚AdaBoost算法(Penalized cost AdaBoost，PCADA)伪代码如下。

算法2 PCADA算法

输入：训练集S，间隔均值项参数λ_1，损失函数项参数λ_2，τ，步长η，迭代次数T

输出：$f(x)=\sum_{t=1}^{T}U_t(x)$，$H(x)=\mathrm{sign}(f(x))$

初始化：数据分布$D_i^1=1/n$，$i=1,\cdots,n$，负类样本代价敏感值为1，正类样本代价敏感值为n_-/n_+；

(1) for $t=1,\cdots,T$

(2) 基于数据分布D_i^t训练加权SVM基分类器，得到正类样本集S_t^1，负类样本集S_t^2，由每一个样本集S_t^j，$j\in\{1,2\}$，计算W_{t+}^j和W_{t-}^j；

(3) 计算间隔反馈因子m_i^t；

(4) 计算M_{t+}^j和M_{t-}^j；

(5) 计算基分类器U_t^j；

(6) 对于样本x_i，如果被分为正类，$U_t(x_i)=U_t^1(x)$；否则$U_t(x_i)=U_t^2(x)$；

(7) 更新样本权重和代价敏感函数：$D_i^{t+1}=C_i^t\exp\left(-y_i\sum_{q=1}^{t}U_q(x_i)\right)$，$C_i^t=\begin{cases}1 & y_i=-1\\ (n_-/n_+)\left(1+\dfrac{1}{1+\exp\left((1+U_t(x_i))^2\right)}\right) & y_i=1\end{cases}$

(8) 对于样本x_i，如果$D_i^{t+1}>Q_{t+1}$且$\mathrm{Margin}_t(x_i)<0$，则重置样本权重以及前t次迭代的基分类器的和：$D_i^{t+1}=1$，$\sum_{q=1}^{t}U_q(x_i)=0$，$Q_{t+1}=\max_i\{D_i^{t+1}\}-\dfrac{\max_i\{D_i^{t+1}\}-\min_i\{D_i^{t+1}\}}{\tau}$，对样本权重归一化：$D_i^{t+1}=D_i^{t+1}\Big/\sum_i D_i^{t+1}$；

(9) end for

3 实验与分析

3.1 实验数据集及评价指标

本节给出本文所提算法和对比算法的实验，实验为五折交叉验证的平均值。编程语言采用Matlab R2017b，除Avila数据集[19]和bank数据集[20]外，其他数据集均来自KEEL[21]，数据集的详细信息如表1所示。

表1 非平衡数据集

Table 1 Imbalanced datasets

数据集	非平衡比	样本个数	维数
yeast-2vs4	9.08	514	8
vehicle0	3.25	846	18
pima	1.87	768	8
yeast1	2.46	1 484	8
glass-0-1-6vs2	10.29	192	9
segment0	6.02	2 308	19
car-vgood	25.58	1 728	6
kr-vs-k-zero	26.63	2 901	6
abalone9-18	16.4	731	8
vowel0	9.98	988	13
ecoli4	15.8	336	7
page-blocks0	8.79	5 472	10
Avila	8.52	20 867	10
bank-full	7.54	45 211	16

采用几何均值(G-mean)和ROC曲线下面积(Aera under ROC curve，AUC)作为非平衡数据分类问题的评价指标。首先给出混淆矩阵如表2所

表 2　混淆矩阵

Table 2　Confusion matrix

类别	正类预测类标	负类预测类标
正类真实类标	TP	FN
负类真实类标	FP	TN

示。AUC 取值在 0.5～1.0 之间，AUC 越接近于 1.0，模型的性能越好。

根据混淆矩阵，几何均值表示为

$$\text{G-mean}=\sqrt{\frac{\text{TP}}{\text{TP}+\text{FN}}\times\frac{\text{TN}}{\text{TN}+\text{FP}}} \tag{10}$$

3.2　几何均值对比实验

首先给出对比算法的描述。

（1）PADA 算法。该算法为惩罚 AdaBoost 算法[15]，采用标准的 SVM 优化模型，求解方法为 SGD 方法。

（2）GCADA 算法。该算法在经典的 Gentle AdaBoost 算法[11]的基础上，在样本权重策略中引入代价项 C_i^t，如果 $y_i=1$，则 $C_i^t=n_-/n_+$；否则，$C_i^t=1$。模型采用加权 SVM 优化模型，求解方法为 SVRG 方法。

（3）MCADA 算法。该算法在 Margin-pruning Boost 算法[14]的基础上，在其样本权重策略中引入代价项 C_i^t，如果 $y_i=1$，则 $C_i^t=n_-/n_+$；否则，$C_i^t=1$。模型采用加权 SVM 优化模型，求解方法为 SVRG 方法。

（4）SMLBoost 算法[8]。该算法与过采样方法相结合以平衡类分布。模型采用标准的 SVM 优化模型，求解方法为 SVRG 方法。

下面给出本文提出算法与对比算法的几何均值对比实验结果，如表 3 所示。

表 3　本文提出算法与对比算法的几何均值

Table 3　Geometric mean values of the proposed algorithm and the comparison algorithm

数据集	PADA 算法	GCADA 算法	MCADA 算法	SMLBoost 算法	PCADA 算法
yeast2vs4	0.843 4	0.855 1	0.899 6	0.905 4	**0.932 5**
	0.869 1	**0.892 3**	**0.938 5**	**0.926 2**	**0.959 7**
vehicle0	0.717 2	0.709 5	0.736 5	0.687 4	**0.762 1**
	0.895 8	0.902 9	0.902 9	0.857 7	**0.924 0**
pima	0.540 0	0.550 0	0.522 1	0.557 4	**0.571 5**
	0.745 2	0.740 8	0.738 4	0.738 7	**0.745 7**
yeast1	0.608 1	0.707 5	0.699 5	0.665 7	**0.722 4**
	0.726 4	**0.786 2**	**0.741 3**	**0.696 8**	**0.726 1**
glass-0-1-6vs2	0.507 0	0.534 5	0.552 3	0.547 8	**0.560 6**
	0.736 7	0.609 4	0.788 3	0.897 7	**0.925 8**
segment0	0.572 2	0.570 7	0.749 8	0.785 4	**0.882 1**
	0.914 0	0.915 3	0.877 8	0.925 4	**0.942 8**
car-vgood	0.896 0	0.919 4	0.920 2	0.885 7	**0.961 7**
	0.904 7	0.947 5	**0.963 6**	0.896 7	**0.969 4**
kr-vs-k-zero	0.742 0	0.747 7	0.750 1	0.776 5	**0.807 3**
	0.742 8	**0.752 0**	**0.752 5**	**0.785 7**	**0.845 2**
abalone9-18	0.731 1	0.761 3	0.631 9	0.756 5	**0.816 4**
	0.808 0	0.843 7	0.800 8	0.825 7	**0.883 6**
vowel0	0.694 3	0.692 3	0.755 7	0.748 9	**0.794 9**
	0.839 8	0.860 3	0.909 9	0.895 7	**0.911 8**
ecoli4	0.756 5	0.792 1	0.806 5	0.816 7	**0.883 8**
	0.801 4	0.817 4	0.815 6	**0.837 3**	**0.898 5**
page-blocks0	0.746 9	0.800 1	0.793 3	0.756 4	**0.821 9**
	0.769 0	**0.847 5**	**0.814 6**	**0.808 7**	**0.881 0**
Avila	0.541 6	0.602 1	0.613 6	0.639 8	**0.674 9**
	0.690 1	0.745 4	0.710 5	0.758 9	**0.803 9**
bank-full	0.500 1	0.507 4	0.506 4	0.505 4	**0.510 6**
	0.500 1	0.510 2	0.536 9	0.554 1	**0.564 8**

表 3 中对于不同的数据集每个算法都有两个数值结果，分别为线性算法和非线性算法的数值结果。可以发现：无论线性算法和非线性算法，本文算法的几何均值都优于其他算法，说明本文算法能有效解决非平衡数据分类问题。

基于表 3 中线性算法和非线性算法在 ecoli4 数据集上几何均值的对比结果，本文以几何均值为分类精度，给出在 ecoli4 数据集上各算法随迭代次数变化的泛化误差对比效果图，如图 2 所示。图 2(a)与图 2(b)分别为线性算法和非线性算法在 ecoli4

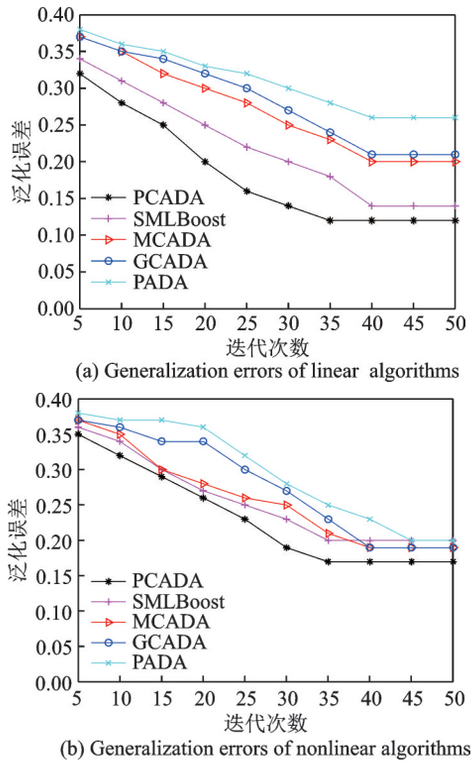


图 2 线性算法和非线性算法的泛化误差

Fig.2 Generalization errors of linear and nonlinear algorithms

数据集上的泛化误差对比效果图，对比两图可以看出：无论线性算法还是非线性算法，随着迭代次数的增加，泛化性能都会逐渐增强并且趋于稳定。PCADA 算法的泛化性能优于其他对比算法，并且在第 35 次迭代时，算法的泛化性能就已经趋于稳定。通过图 2 可以看出，PCADA 算法以较少的迭代次数达到了最优的泛化性能。

3.3 AUC 对比实验

本节给出本文提出算法与对比算法的 AUC 实验结果。从表 4 中可以看出，本文提出算法的 AUC 数值结果相对于对比算法来说是最优的。其中，加了代价项的 GCADA 算法和 MCADA 算法优于不加代价项的 PADA 算法，而加了代价敏感函数的 PCADA 算法优于其他算法，这也充分说明了本文提出算法的优越性。SMLBoost 算法由于采用过采样技术处理非平衡数据问题，因此分类效果优于 PADA 算法。

3.4 参数分析实验

针对本文提出的算法，给出参数 τ 对 ecoli4 数据集几何均值的影响，如图 3 所示。图 3(a)与图 3(b)分别为线性算法和非线性算法中参数 τ 对 eco-

表 4 本文提出算法与对比算法的 AUC

Table 4 AUC of the proposed algorithms and the comparison algorithms

数据集	PADA 算法	GCADA 算法	MCADA 算法	SMLBoost 算法	PCADA 算法
yeast2vs4	0.572 2 0.914 0	0.570 7 0.915 3	0.749 8 0.877 8	0.785 4 0.925 4	**0.882 1** **0.962 8**
vehicle0	0.667 8 0.795 0	0.679 5 0.801 1	0.706 2 0.822 9	0.697 4 0.847 7	**0.792 1** **0.944 0**
pima	0.540 0 0.657 2	0.550 0 0.665 8	0.522 1 0.708 9	0.557 4 0.738 7	**0.671 4** **0.780 0**
yeast1	0.572 2 0.914 0	0.570 7 0.915 3	0.749 8 0.877 8	0.785 4 0.925 4	**0.882 1** **0.942 8**
glass-0-1-6vs2	0.609 8 0.736 7	0.634 5 0.809 9	0.651 1 0.799 3	0.647 8 0.864 5	**0.660 6** **0.925 8**
segment0	**0.842 0** **0.892 8**	0.847 7 **0.898 7**	**0.849 1** **0.902 5**	0.936 5 **0.945 7**	**0.937 3** **0.948 2**
car-vgood	0.796 0 0.762 1	0.879 4 0.847 7	0.821 1 0.832 5	0.895 1 0.901 2	**0.961 7** **0.969 4**
kr-vs-k-zero	**0.842 0** **0.863 8**	**0.847 0** **0.874 3**	**0.850 1** **0.879 5**	**0.836 5** **0.885 7**	**0.867 3** **0.895 2**
abalone9-18	0.654 3 0.708 0	0.647 8 0.754 3	0.708 9 0.794 8	0.776 5 0.825 7	**0.878 6** **0.893 5**
vowel0	0.564 3 0.739 8	0.654 3 0.760 9	0.709 8 0.801 1	0.733 1 0.895 7	**0.794 9** **0.932 1**
ecoli4	0.654 3 0.785 4	0.708 4 0.743 2	0.802 1 **0.813 4**	0.826 4 **0.844 7**	**0.853 1** **0.855 5**
page-blocks0	0.652 1 **0.702 6**	0.700 3 **0.798 7**	0.723 3 **0.814 0**	0.756 4 **0.848 7**	**0.834 7** **0.887 0**
Avila	0.574 5 0.658 8	0.624 5 0.729 8	0.658 8 0.745 6	0.734 6 0.758 9	**0.774 8** **0.823 4**
bank-full	0.506 5 0.509 1	0.527 4 **0.532 4**	0.567 4 **0.570 0**	0.565 4 0.574 1	**0.623 3** **0.678 9**

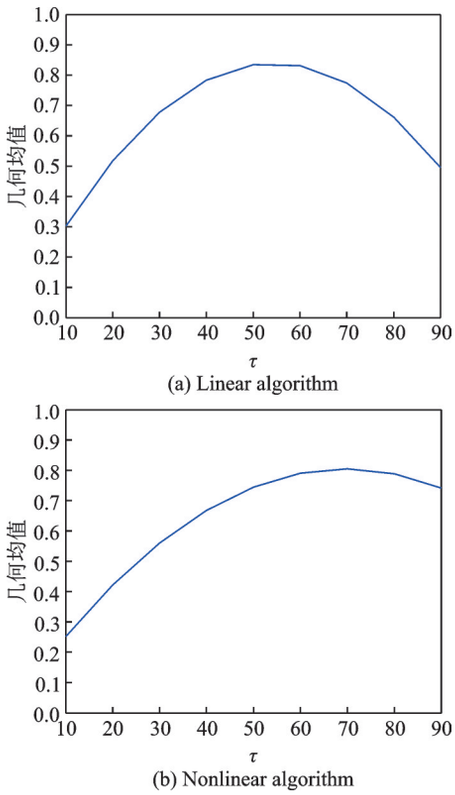
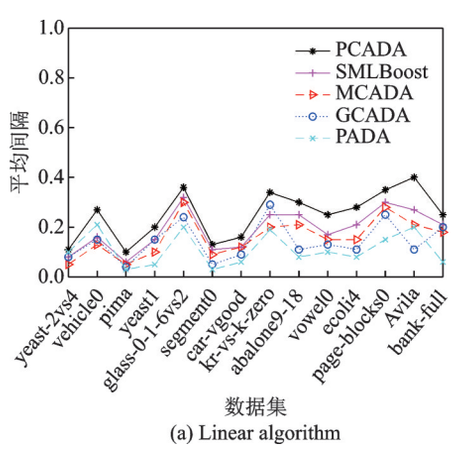


图 3　参数 τ 对 ecoli4 数据集几何均值的影响结果

Fig.3　Influence of the parameter τ on the geometric mean of ecoli4 dataset

li4 数据集上几何均值的影响趋势图。从图 3(a)中可以发现，一开始随着参数 τ 的不断增大，算法的几何均值走势呈现上升趋势，而后呈现下降趋势。当 τ=50 时，线性算法的几何均值最优。而对于图 3(b)，随着参数 τ 的不断增大，算法的几何均值走势亦呈现上升趋势，而后呈现稍微下降的趋势。当 τ=70 时，非线性算法的几何均值最优。

下面给出本文算法和对比算法在不同数据集上的平均间隔折线图，如图 4 所示。图 4(a)和(b)分别为线性算法与非线性算法在不同数据集上的平均间隔对比。从图 4 可以看出，无论是线性算法还是非线性算法，提出的 PCADA 算法的平均间隔最大，SMLBoost 算法的平均间隔略低于 PCADA



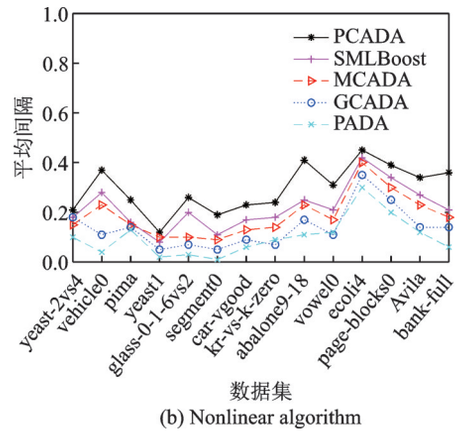


图 4　各种算法在不同数据集上的平均间隔

Fig.4　Average interval of various algorithms on different datasets

算法，其次是 MCADA 算法和 GCADA 算法，平均间隔最低的是 PADA。可见，PCADA 算法的分类性能最优。

4　结　　论

为了解决非平衡数据分类问题，提出一种基于代价敏感的惩罚 AdaBoost 算法，在惩罚 AdaBoost 算法样本权重中引入自适应代价敏感函数，使分类器在分类时更加关注少数类样本和分错的样本，且对噪声样本起到抑制作用，提高了少数类样本的分类性能。通过引入惩罚策略增大了样本的平均间隔，进一步提高了分类精度。充分考虑数据分布情况，选择加权 SVM 优化模型作为基分类器，即引入间隔均值项，并根据数据非平衡比对间隔均值项和损失函数项进行加权，采用 SVRG 方法对优化模型进行求解，提高了算法的收敛速度。对比实验表明，本文提出的算法在几何均值和 AUC 上都优于其他算法，并且可产生更大的平均间隔。本文采用加权的 SVM 优化模型，该模型引入了间隔均值项，在此基础上引入间隔方差项，并将该模型应用在惩罚 AdaBoost 算法上，分析算法对分类精度的影响是进一步要做的工作。

参考文献：

[1] HAN Xu, CUI Runbang, LAN Yanfei, et al. A Gaussian mixture model based combined resampling algorithm for classification of imbalanced credit data sets[J]. International Journal of Machine Learning and Cybernetics, 2019, 10(1): 3687-3699.

[2] SHAHEE S A, ANANTHAKUMAR U. An adaptive oversampling technique for imbalanced datasets [J]. Computer and Information Engineering, 2018, 12: 1-16.

[3] ZONG Weiwei, HUANG Guangbin, CHEN Yiqiang.

Weighted extreme learning machine for imbalance learning[J]. Neurocomputing, 2013, 101: 229-242.

[4] TAO Xinmin, LI Qing, GUO Wenjie, et al. Self-adaptive cost weights-based support vector machine cost-sensitive ensemble for imbalanced data classification[J]. Information Sciences, 2019, 52(4): 132-140.

[5] WANG Wenyang, SUN Dongchu. The improved AdaBoost algorithms for imbalanced data classification [J]. Information Sciences, 2021, 563: 358-374.

[6] SCHAPIRE R E, FREUND Y, BARTLETT P, et al. Boosting the margin: A new explanation for the effectiveness of voting methods[J]. The Annals of Statistics, 1998, 26(5):1651-1686.

[7] RÄTSCH G, WARMUTH M K. Efficient margin maximizing with boosting [J]. Journal of Machine Learning Research, 2005, 6(11): 2131-2152.

[8] ALLAN G, LARSEN K G, MATHIASEN A. Optimal minimal margin maximization with boosting[EB/OL]. (2019-01-30) [2019-02-20]. https://doi.org/10.48550/arXiv.1901.10789.

[9] CHEN Zhi, DUAN Jiang, YANG Cheng, et al. SML-Boost-adopting a soft-margin like strategy in boosting[J]. Knowledge-Based Systems, 2020, 195(31):105705.

[10] DEMIRIZ A, BENNETT K P, SHAWE-TAYLOR J. Linear programming boosting via column generation [J]. Machine Learning, 2002, 46(1/2/3):225-254.

[11] FRIEDMAN J, HASTIE T, TIBSHIRANI R. Additive logistic regression: A statistical view of boosting [J]. Annals of Statistics, 2000, 28(2):337-374.

[12] VEZHNEVETS A, VEZHNEVETS V. Modest Adaboost-teaching Adaboost to generalize better [J]. Graphicon, 2005, 12(4): 987-997.

[13] WARMUTH M K, GLOCER K A, RTSCH G. Boosting algorithms for maximizing the soft margin [C]// Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems. Vancouver, British Columbia, Canada: DBLP, 2007: 3-6.

[14] WU Shuqiong, NAGAHASHI H. A new method for solving overfitting problem of gentle AdaBoost[C]// Proceedings of International Conference on Graphic & Image Processing. Washington State, USA: International Society for Optics and Photonics, 2013:123-128.

[15] WU Shuqiong, NAGAHASHI H. Penalized AdaBoost: Improving the generalization error of gentle AdaBoost through a margin distribution [J]. IEICE Transactions on Information & Systems, 2015(11): 1906-1915.

[16] CHENG Fanyong, ZHANG Jing, WEN Cuihong, et al. Large cost-sensitive margin distribution machine for imbalanced data classification [J]. Neurocomputing, 2016, 24(8): 45-57.

[17] WANG Jun, ZHOU Zhihua. Margin distribution analysis[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 99: 1-13.

[18] RIE J, ZHANG Tong. Accelerating stochastic gradient descent using predictive variance reduction [C]// Proceedings of Advanced in Neural Information Systems. New York, USA: ACM, 2013: 315-323.

[19] DE STEFANO C, MANIACI M, FONTANELLA F, et al. Reliable writer identification in medieval manuscripts through page layout features: The "Avila" Bible case[J]. Engineering Applications of Artificial Intelligence: The International Journal of Intelligent Real-Time Automation, 2018, 72: 99-110.

[20] MORO S, CORTEZ P, LAUREANO R. Using data mining for bank direct marketing: An application of the CRISP-DM Methodology [C]// Proceedings of European Simulation & Modelling Conference. Athens, Greece: IJMO, 2011: 201-205.

[21] KEEL. A software tool to assess evolutionary algorithms for data mining problems [EB/OL]. (2005-11-05) [2019-05-30]. http: //www.keel.es/.

(编辑:刘彦东)