

DOI: 10.7652/xjtuxb201708006

采用多类代价指数损失函数的 代价敏感 AdaBoost 算法

翟夕阳, 王晓丹, 李睿, 贾琪
(空军工程大学防空反导学院, 710051, 西安)

摘要: 为解决由多个二类代价敏感算法扩展而成的多类算法存在时间复杂度高和不能区分错分代价的问题,提出一种采用多类代价指数损失函数的多类代价敏感 AdaBoost 算法(MCCSADA)。为保证算法的代价敏感特性,首先设计一种满足代价敏感损失函数设计准则的多类代价敏感指数损失函数;然后将此损失函数作为评价分类器性能的标准,以最小化损失函数为目的使用逐步叠加模型推导算法的最优基分类器加权系数;最后使用多类代价损失函数和最优基分类器加权系数求解公式替换多类 AdaBoost 算法的损失函数和加权系数求解公式,得到代价敏感的 MCCSADA 算法。使用 UCI 数据集对算法进行验证,实验结果表明:算法的稳定性得到了提升,退化现象被减弱;相比于由二类代价敏感算法通过一对一方法扩展而来的多类代价敏感算法,MCCSADA 算法在大多数情况下能够取得更低的代价,而且具有较低的时间复杂度,在 3 类数据集上的时间复杂度降低约 40%,并且随着类别数的增多效率提升更加明显。

关键词: 代价敏感;AdaBoost 算法;多分类;贝叶斯决策;损失函数

中图分类号: TP391 **文献标志码:** A **文章编号:** 0253-987X(2017)08-0033-07

A Multi-Class Cost Sensitivity AdaBoost Algorithm Using Multi-Class Cost Exponential Loss Function

ZHAI Xiyang, WANG Xiaodan, LI Rui, JIA Qi

(Institute of Air Defense and Anti-Missile, Air Force Engineering University, Xi'an 710051, China)

Abstract: A multi-class cost sensitivity AdaBoost algorithm is proposed to solve the problems of high time complexity and indistinguishable cost among different classes in using the existing multi-class algorithm, which is an extension of some binary cost sensitive AdaBoost algorithms. The new algorithm uses a multi-class exponential loss function, and is named as MCCSADA. A cost sensitive multi-class exponential loss function is designed to satisfy design guidelines of cost-sensitive loss function and to ensure the cost-sensitive characteristic. Then, the loss function is used as a criterion for the evaluation of basis classifiers and the optimal weighted coefficients of base classifiers are obtained by using the forward stack model to minimize the cost loss function. Subsequently, MCCSADA is obtained by using the new loss function and weighted coefficients to replace the original loss function and coefficients in AdaBoost algorithm. MCCSADA is verified by using UCI dataset, and the results and a comparison with the CSOVO expanding from binary algorithm show that MCCSADA has lower cost and lower time complexity in most cases. The time complexity reduces by about 40% when the dataset contains three classes and reduces more

收稿日期: 2017-01-19。 作者简介: 翟夕阳(1994—),男,硕士生;王晓丹(通信作者),女,教授,博士生导师。 基金项目: 国家自然科学基金资助项目(61273275,61503407)。

网络出版时间: 2017-05-15

网络出版地址: <http://www.cnki.net/kcms/detail/61.1069.T.20170515.1824.002.html>

with increasing number of data categories. Moreover, the stability of the algorithm is promoted, and the degeneration is weakened.

Keywords: cost sensitive; AdaBoost; multi-class; Bayes decision; loss function

模式识别中传统的分类学习算法以最小化错误率为目标,把不同类别同等看待,类别间的错分代价视为相等,但是在很多实际问题中,不同类别之间的错分代价是不同的,例如医疗诊断、金融预测和恶意程序检测等,这就是代价敏感问题^[1]。代价敏感问题的解决需要以最小化风险为目标,代价敏感学习算法能更好地解决此类问题。AdaBoost 算法^[2]是一种常用的集成算法,是 Boosting 算法的一种改进,能够很好地提升弱分类算法的分类性能。为了能够解决代价敏感问题,文献[3]提出了一种代价敏感 Boosting 算法,但是效果并不好。之后有学者指出 AdaBoost 算法本身具有代价敏感特性^[4],只需要在初始化样本权时对不同的样本根据重要程度赋予不同的权值即可。为了得到效果稳定的代价敏感 AdaBoost 算法,研究者们针对代价敏感 AdaBoost 算法展开了深入的研究,提出了把代价直接加入 AdaBoost 算法的 AdaC 系列算法^[5-6]和改变权值分配策略的 CSB 系列算法^[7],但是这两类算法缺乏理论支撑,破坏了算法的 Boosting 特性。为解决此问题,文献[8]提出了使用梯度下降法寻找最优参数的 Asymmetric Boosting 算法和以代价敏感 logit 变换为基础的 Asymmetric LogitBoost 算法,但这两种算法的时间复杂度较高。

此外,还有一些研究者提出了其他的代价敏感 AdaBoost 算法^[9-12],这些算法虽然在实验数据集上实现了代价敏感分类,但仍存在缺乏理论支撑、算法的稳定性和通用性较差的问题。为了解决这些问题,文献[13]提出了一种具有 Fisher 一致性的代价敏感 Boosting 算法,从理论和实验两方面证明了这种算法的可行性。

对于多类问题,最常用的解决方法是将基本的二分类器扩展为多分类器。通常有两种实现方法:一种是直接扩展二分类算法为多分类算法;另一种是使用多个二分类算法构造多分类器,如一对一方法、一对多方法等。在扩展二分类代价敏感 AdaBoost 算法时,使用一对一的方法进行扩展可以充分考虑数据错分到不同类别的代价,但是算法的时间复杂度较高,而且随着类别数的增加呈指数增长;使用一对多和二叉树形式扩展时,算法的时间复杂度较低,但是不能区分样本的错分代价。为了

使多类代价敏感 AdaBoost 算法具有较高的效率同时又能够区分错分代价,本文在使用多类代价敏感指数损失函数的基础上利用逐步叠加模型^[14]直接把二类的代价敏感 AdaBoost 算法扩展为多类算法,从而提出一种采用多类代价敏感指数损失函数的多类代价敏感 AdaBoost 算法(Multi-class cost sensitive AdaBoost, MCCSADA)。MCCSADA 直接扩展二类代价敏感算法到多类,降低了现有多类算法的时间复杂度,而且与由一对一方法扩展而来的多类代价敏感算法相比取得了更低的代价。

1 多类代价敏感损失函数

损失函数是一种衡量错分代价的函数,有的损失函数对类别不敏感,有的损失函数对类别敏感,被称为代价敏感损失函数。算法是否使用代价敏感损失函数决定了该算法是否具有代价敏感性质^[15]。代价敏感损失函数指的是加入代价矩阵后的损失函数,代价敏感损失函数的决策是否与代价敏感贝叶斯决策相一致是代价敏感算法实现的基础。为了验证代价敏感损失函数是否有效,文献[8,16]中提出了代价敏感损失函数的设计准则,代价敏感损失函数需要满足贝叶斯一致性并且在决策函数边界处取得最值。本文把多类指数损失函数设计为多类代价敏感指数损失函数,并依据代价敏感损失函数设计准则对设计的函数进行了理论证明。

多类代价敏感指数损失函数如下

$$L(\bar{\mathbf{y}}, f(\bar{\mathbf{x}})) = \exp\left(-\frac{1}{K} \mathbf{C}_{c(f(\bar{\mathbf{x}})), c(\bar{\mathbf{x}})} (\bar{\mathbf{y}}_1 f_1(\bar{\mathbf{x}}) + \cdots + \bar{\mathbf{y}}_K f_K(\bar{\mathbf{x}}))\right) \quad (1)$$

式中: K 为总类数; $\bar{\mathbf{x}}$ 为任意样本; $\bar{\mathbf{y}} = [\bar{y}_1, \cdots, \bar{y}_K]^T$ 为 $\bar{\mathbf{x}}$ 的多维类别, $\bar{\mathbf{y}}$ 中元素取值为

$$\bar{y}_k = \begin{cases} 1, & c(\bar{\mathbf{x}}) = k \\ -\frac{1}{K-1}, & c(\bar{\mathbf{x}}) \neq k \end{cases} \quad (2)$$

$f(\bar{\mathbf{x}})$ 为预测的多维输出,与 $\bar{\mathbf{y}}$ 定义相同; $\mathbf{C}$ 为代价矩阵; $\mathbf{C}_{c(f(\bar{\mathbf{x}})), c(\bar{\mathbf{x}})}$ 为分类代价; $c(f(\bar{\mathbf{x}}))$ 为预测类别; $c(\bar{\mathbf{x}})$ 为真实类别。

下面对式(1)是否满足代价敏感损失函数设计准则进行证明。

对于代价敏感 Bayes 算法,假设数据集有 K 个类别,代价矩阵为 $\mathbf{C}$,样本为 $\tilde{\mathbf{x}}$,对应的类别为 $c(\tilde{\mathbf{x}})$ 。把 $\tilde{\mathbf{x}}$ 分为类别 k 的代价可以表示为 $\mathbf{C}_{k,c(\tilde{\mathbf{x}})} P(c=k|\tilde{\mathbf{x}})$,其中 P 表示概率,所对应的决策函数可以表示为 $\arg \min_{T(\tilde{\mathbf{x}})} \mathbf{C}_{T(\tilde{\mathbf{x}}),c(\tilde{\mathbf{x}})} P(c=T(\tilde{\mathbf{x}})|\tilde{\mathbf{x}})$,其中 $T(\tilde{\mathbf{x}})$ 为决策类别。

对于代价敏感指数损失函数,假设样本 $\tilde{\mathbf{x}}$ 的类别为 $c(\tilde{\mathbf{x}})$, $f(\tilde{\mathbf{x}})$ 的输出类别用 $c(f(\tilde{\mathbf{x}}))$ 表示,求使分类代价最小的 $f(\tilde{\mathbf{x}})$ 可以表示为

$$\arg \min_{f(\tilde{\mathbf{x}})} E_{y|\tilde{\mathbf{x}}} = \exp\left(-\frac{1}{K} \mathbf{C}_{c(f(\tilde{\mathbf{x}})),c(\tilde{\mathbf{x}})} (\tilde{y}_1 f_1(\tilde{\mathbf{x}}) + \cdots + \tilde{y}_K f_K(\tilde{\mathbf{x}}))\right) \quad (3)$$

式中: $f_1(\tilde{\mathbf{x}}) + \cdots + f_K(\tilde{\mathbf{x}}) = 0$ 。

为了求出 $f(\tilde{\mathbf{x}})$,对式(3)进行拉格朗日变换得到下式

$$\arg \min_{f(\tilde{\mathbf{x}})} \exp\left(-\mathbf{C}_{1,c(\tilde{\mathbf{x}})} \frac{f_1(\tilde{\mathbf{x}})}{K-1}\right) P(c=1|\tilde{\mathbf{x}}) + \cdots + \exp\left(-\mathbf{C}_{K,c(\tilde{\mathbf{x}})} \frac{f_K(\tilde{\mathbf{x}})}{K-1}\right) P(c=K|\tilde{\mathbf{x}}) - \lambda(f_1(\tilde{\mathbf{x}}) + \cdots + f_K(\tilde{\mathbf{x}})) \quad (4)$$

$f_k(\tilde{\mathbf{x}})$ 表示在第 $k(k=1,2,\dots,K)$ 类上的输出。对式(3)中 $f(\tilde{\mathbf{x}})$ 求导,在导数为 0 的情况下可得到使分类代价最小的 $f(\tilde{\mathbf{x}})$ 。对求导得出的方程组进行求解,可以得到 λ 的两种表示形式

$$\lambda = -\frac{1}{K(K-1)} \sum_{k=1}^K \mathbf{C}_{k,c(\tilde{\mathbf{x}})} \exp\left(-\frac{f_k(\tilde{\mathbf{x}})}{K-1}\right) P(c=k|\tilde{\mathbf{x}}) \quad (5)$$

$$\lambda = -\frac{1}{K-1} \mathbf{C}_{k,c(\tilde{\mathbf{x}})} \exp\left(-\frac{f_k^*(\tilde{\mathbf{x}})}{K-1}\right) P(c=k|\tilde{\mathbf{x}}) \quad (6)$$

式中: $f_k^*(\tilde{\mathbf{x}})$ 为在最优决策类别上的输出。

令两个 λ 的表达式相等,对式(6)两边进行消元,然后取对数可得到

$$f_k^*(\tilde{\mathbf{x}}) = (K-1) \mathbf{C}_{1,c(\tilde{\mathbf{x}})} \ln P(c=k|\tilde{\mathbf{x}}) - \frac{1}{K} \sum_{k=1}^K \frac{f_k(\tilde{\mathbf{x}})}{K-1} - \frac{1}{K} \sum_{k=1}^K \mathbf{C}_{k,c(\tilde{\mathbf{x}})} \ln P(c=k|\tilde{\mathbf{x}}) \quad (7)$$

由于 $f_1(\tilde{\mathbf{x}}) + \cdots + f_K(\tilde{\mathbf{x}}) = 0$,所以

$$f_k^*(\tilde{\mathbf{x}}) = (K-1) \mathbf{C}_{k,c(\tilde{\mathbf{x}})} \ln P(c=k|\tilde{\mathbf{x}}) - \frac{1}{K} \sum_{k=1}^K \mathbf{C}_{k,c(\tilde{\mathbf{x}})} \ln P(c=k|\tilde{\mathbf{x}}) \quad (8)$$

因为式(8)中 $\frac{1}{K} \sum_{k=1}^K \mathbf{C}_{k,c(\tilde{\mathbf{x}})} \ln P(c=k|\tilde{\mathbf{x}})$ 为定值,所以代价敏感指数损失函数的决策函数为

$$\arg \min_k f_k^*(\tilde{\mathbf{x}}) = \arg \min_k \mathbf{C}_{k,c(\tilde{\mathbf{x}})} P(c=k|\tilde{\mathbf{x}}) \quad (9)$$

至此,可得出代价敏感指数损失函数满足贝叶斯一致性,并且都是在 $\mathbf{C}_{k,c(\tilde{\mathbf{x}})} P(c=k|\tilde{\mathbf{x}})$ 相等时取得最值,所以本文设计的多类代价敏感指数损失函数满足代价敏感损失函数的设计准则。

2 多类代价敏感 MCCSADA 算法

本文设计的 MCCSADA 算法希望通过集成基分类器来使分类代价达到最小。权值更新策略和加权系数的确定是 AdaBoost 算法的核心,同时也是构造 MCCSADA 算法所需要解决的主要问题。在构造 SAMME 算法时,文献[16]使用逐步叠加模型来确定算法的基分类器加权系数和权值更新策略,本文同样使用逐步叠加模型来推导 MCCSADA 算法的基分类器加权系数和权值更新策略。

逐步叠加模型是指在已知由前 $t-1$ 次迭代生成的强分类器的条件下求出第 t 次迭代的基分类器的最优加权系数。MCCSADA 算法在迭代时每次迭代生成一个基分类器,所以第 t 次迭代出的基分类器就是第 t 个基分类器(决策函数)。本文构造的是代价敏感的 MCCSADA 算法,所以在求解基分类器最优加权系数时需求得使集成第 t 个基分类器后的集成分类器分类代价最小的加权系数,这保证了集成分类器的代价呈现下降趋势。当然,当分类代价达到极值时集成分类器再添加新的基分类器,代价将会处于波动之中。在求出加权系数之后,通过基分类器的加权系数和训练样本的分类情况指导样本的权值更新。

MCCSADA 算法希望在给定数据集 $D = \{\mathbf{x}_i | i=1,\dots,n\}$ 和代价矩阵 $\mathbf{C}$ 的情况下,找到一个决策函数(或者说一个分类器)来使分类代价达到最小,即寻找 $F(\tilde{\mathbf{x}})$ 使其满足

$$\min_{F(\tilde{\mathbf{x}})} \sum_{i=1}^n L(\mathbf{y}_i, F(\mathbf{x}_i)) \quad (10)$$

式中: L 表示错分损失; n 为数据总数; $\mathbf{x}_i$ 为第 i 个样本; $\mathbf{y}_i$ 为第 i 个样本的多维类别; $F(\mathbf{x}_i)$ 为集成分类器对 $\mathbf{x}_i$ 的输出类别,并且类别输出函数为 $F(\tilde{\mathbf{x}}) = \sum_{t=1}^T \beta^{(t)} f^{(t)}(\tilde{\mathbf{x}})$; T 为总的决策函数个数; $\beta^{(t)}$ 为第 t 次迭代出的决策函数的加权系数; $f^{(t)}(\tilde{\mathbf{x}})$ 为第 t 个决策函数,并且满足 $f_1^{(t)}(\tilde{\mathbf{x}}) + \cdots + f_K^{(t)}(\tilde{\mathbf{x}}) = 0$ 。

使用逐步叠加模型对基分类器最优加权系数 $\beta^{(t)}$ 进行推导

$$(\beta^{(t)}, f^{(t)}(\tilde{\mathbf{x}})) =$$

$$\arg \min_{\beta, f(\mathbf{x})} \sum_{i=1}^n L(Y_i, F^{(t-1)}(\mathbf{x}_i) + \beta f(\mathbf{x}_i)) \quad (11)$$

式中: $F^{(t-1)}(\mathbf{x}_i)$ 为前 $t-1$ 个决策函数的集成。

将式(11)中带有 $F^{(t-1)}(\mathbf{x}_i)$ 的项在第 t 次迭代时设为定值,使用 $w_i^{(t)}$ 进行表示,则式(11)可以表示为

$$\begin{aligned} & (\beta^{(t)}, f^{(t)}(\tilde{\mathbf{x}})) = \\ & \arg \min_{\beta, f(\tilde{\mathbf{x}})} \sum_{i=1}^n w_i^{(t)} \exp \left[-\frac{1}{K} \mathbf{C}_{c(f(\mathbf{x}_i)), c(\mathbf{x}_i)} \mathbf{y}_i^T \beta f(\mathbf{x}_i) \right] \end{aligned} \quad (12)$$

求解使代价最小的加权系数时,可以使用多类分类器 $T(\mathbf{x}_i)$ 来替换 $\mathbf{y}_i^T F(\mathbf{x}_i)$ 对上式定义,假设数据正确分类的代价用 1 表示,其余错分类别代价用单位分类代价的倍数表示,重新定义的求解公式如下

$$\begin{aligned} & (\beta^{(t)}, T^{(t)}(\tilde{\mathbf{x}})) = \\ & \arg \min_{\beta, T(\tilde{\mathbf{x}})} \sum_{c(\mathbf{x}_i) = T(\mathbf{x}_i)} w_i \exp \left(-\frac{\beta}{K-1} \mathbf{C}_{c(\mathbf{x}_i), c(\mathbf{x}_i)} \right) + \\ & \sum_{c(\mathbf{x}_i) \neq T(\mathbf{x}_i)} w_i \exp \left(\frac{\beta}{(K-1)^2} \mathbf{C}_{T(\mathbf{x}_i), c(\mathbf{x}_i)} \right) = \\ & \exp \left(-\frac{\beta}{K-1} \right) \left(\sum_i w_i - \sum_i w_i (c(\mathbf{x}_i) \neq T(\mathbf{x}_i)) \right) + \\ & \sum_{k_1=1}^K \sum_{k_2=1}^K \exp \left(\frac{\beta}{(K-1)^2} \mathbf{C}_{k_2, k_1} \right) \cdot \\ & \sum_i w_i B(c(\mathbf{x}_i), T(\mathbf{x}_i)) B(c(\mathbf{x}_i), k_1) B(T(\mathbf{x}_i), k_2) \end{aligned} \quad (13)$$

式中: K 为类别总数; $T(\mathbf{x}_i)$ 为决策类别; $c(\mathbf{x}_i)$ 为样本类别; w_i 表示样本权重; $B(\cdot)$ 为布尔函数,定义

$$B(a, b) = \begin{cases} 0, & a \neq b \\ 1, & a = b \end{cases}$$

$T(\tilde{\mathbf{x}})$ 与 $f(\tilde{\mathbf{x}})$ 的关系如下

$$f_k(\tilde{\mathbf{x}}) = \begin{cases} 1, & T(\tilde{\mathbf{x}}) = k \\ -\frac{1}{K-1}, & T(\tilde{\mathbf{x}}) \neq k \end{cases} \quad (14)$$

求使式(13)最小的 β , 可以通过对式(13)求导,使导数为 0 得到,求导后的式子如下

$$\begin{aligned} & -\frac{1}{K-1} \exp \left(-\frac{\beta}{K-1} \right) \left(\sum_i w_i - \sum_i w_i (c(\mathbf{x}_i) \neq T(\mathbf{x}_i)) \right) + \\ & \sum_{k_1=1}^K \sum_{k_2=1}^K \frac{\mathbf{C}_{k_2, k_1}}{(K-1)^2} \exp \left(\frac{\beta}{(K-1)^2} \mathbf{C}_{k_2, k_1} \right) \cdot \\ & \sum_i w_i B(c(\mathbf{x}_i), T(\mathbf{x}_i)) B(c(\mathbf{x}_i), k_1) B(T(\mathbf{x}_i), k_2) = 0 \end{aligned} \quad (15)$$

根据上述推导可以得到基于风险最小化的代价

敏感 MCCSADA 算法,算法的具体流程如下。

步骤 1 初始化权值 $w_i^{(1)} = D(i) = 1/m, i=1, 2, \dots, m$ 。

步骤 2 for $t=1, 2, \dots, T$, 执行以下(1)~(5)步:

(1) 依据权值 $w^{(t)}$, 选择训练样本;

(2) 对样本进行分类 $h_t: X \rightarrow Y$, X 为所有样本的集合, Y 为所有类别的集合;

(3) 根据式(14)计算 $\beta^{(t)}$ 的值;

(4) 计算新的权重向量

$$\begin{aligned} w_i^{(t+1)} &= w_i^{(t)} \exp \left[-\frac{1}{K} \beta^{(t)} \mathbf{y}_i^T h_t(\mathbf{x}_i) \right] = \\ & \begin{cases} w_i^{(t)} \exp \left(-\frac{1}{K-1} \beta^{(t)} \right), & \mathbf{y}_i = h_t(\mathbf{x}_i) \\ w_i^{(t)} \exp \left(\frac{1}{(K-1)^2} \beta^{(t)} \right), & \mathbf{y}_i \neq h_t(\mathbf{x}_i) \end{cases} \end{aligned} \quad (16)$$

(5) 归一化 $w^{(t+1)}$ 。

步骤 3 对步骤 2 中迭代得出的基分类器使用最优加权系数进行加权,加权后的分类器即为最终的强分类器。集成的强分类器类别输出如下

$$h_f(\tilde{\mathbf{x}}) = \arg \max_{y \in Y} \sum_{i=1}^T \beta^{(i)} [h_i(\tilde{\mathbf{x}}) = \mathbf{y}] \quad (17)$$

3 仿真结果与分析

实验使用 UCI 数据集和人工数据集对所提出的算法进行验证,通过对比 SAMME 算法和由一对一方法扩展而来的代价敏感 AdaBoost 算法,来说明 MCCSADA 算法的代价敏感性、Boosting 特性和低时间复杂度特性。

3.1 实验数据

实验数据集的信息如表 1 所示,其中前 7 种数据来源于标准的 UCI 数据集, data1 是使用正态分布函数生成的包含 3 类数据的数据集。

表 1 实验数据的详细信息

数据集	训练集数	测试集数	特征数	类别数
sonar	139	69	60	2
wine	119	59	13	3
vehicle	564	282	18	4
page-blocks	728	364	10	5
yeast	990	494	8	10
vowel	660	330	10	11
soybean	204	102	35	18
data1	2 000	1 000	2	3

为了验证算法的稳定性和通用性,选取类别数不同的数据集。在实验中采用三折交叉验证的方法

来获取训练数据集和测试数据集。实验数据中: wine 数据集的可分性较强,几种类别的数据分布在不同区域;sonar、vehicle、vowel 和 soybean 数据集的可分性略差,类与类之间有部分重叠区域;page-blocks 和 yeast 数据集的可分性较差,类与类之间重叠区域较大;data1 数据集由正态分布函数生成,3 种类别的数据有部分重叠现象。

3.2 实验设计

把本文所提出的算法与传统的 SAMME 算法以及以由文献[13]所提出的由二类代价敏感 AdaBoost算法通过一对一方法扩展而来的多类代价敏感算法(CSOVO)进行对比。3 种算法所使用的弱分类算法均为决策树算法,设置算法的迭代次数为 50 次。由于实验数据的类别总数不同,根据类别数目设计代价矩阵,对每类数据设计 3 个代价矩阵。

由于算法本身的要求,固定对角元素为 1,而对于实际问题分类正确所带来的风险代价为 0,在分类结束后可减去这部分增加的代价。对于矩阵中的其他元素第 1 个矩阵按从左到右从上到下的顺序从 2 开始赋值,第 2 个矩阵逆序从 2 开始赋值,第 3 个矩阵随机生成 1~10 之间的元素,3 个矩阵分别记为 C_1 、 C_2 、 C_3 。对于数据类别总数大于等于 10 的数据集,随机生成 3 个对角元素为 1,其余元素为 1~10 的矩阵。在代价矩阵中的数字表示的是单位代价数,假设一个样本最小错误分类代价为 2 个单位代价,其余类别间的错分代价可以依据单位代价转换为单位代价数。

3.3 实验结果分析

采用 SAMME、CSOVO 和 MCCSADA 这 3 种算法对实验数据进行分类,所得到的分类代价和分类时间如表 2 所示。

表 2 3 种算法的分类代价对比

数据集	代价矩阵	总的单位代价数			分类时间/s		
		SAMME	CSOVO	MCCSADA	SAMME	CSOVO	MCCSADA
sonar	C_1	91.7	93.3	92.7	2.04	3.2	3.1
	C_2	95.0	92.3	94.7	2.01	3.1	3.2
	C_3	118.7	113.3	108.0	2.07	3.1	3.2
wine	C_1	83.3	74.0	72.0	2.1	5.5	3.3
	C_2	96.7	74.0	74.0	2.1	5.4	3.2
	C_3	92.0	68.0	66.3	2.1	5.5	3.3
vehicle	C_1	838.0	827.0	784.7	8.3	25.1	14.4
	C_2	781.7	718.3	664.0	7.5	25.4	14.5
	C_3	572.3	565.3	564.0	8.1	25.4	14.4
page-blocks	C_1	425.7	417.3	413.3	4.5	22.3	7.9
	C_2	711.0	610.7	594.0	4.4	22.4	7.9
	C_3	394.0	389.3	392.7	4.0	21.3	7.8
yeast	C_1	220.3	214.3	190.7	10.7	169.4	36.3
	C_2	334.0	155.0	152.7	13.1	175.3	36.4
	C_3	300.0	234.0	213.3	12.4	171.4	35.5
vowel	C_1	229.0	163.0	147.3	8.8	125.8	22.1
	C_2	243.0	143.3	140.0	8.9	127.4	22.5
	C_3	378.3	199.3	175.7	8.8	125.6	22.3
soybean	C_1	230.7	177.7	151.7	8.8	125.3	22.4
	C_2	256.3	178.7	178.7	8.9	125.8	22.2
	C_3	197.3	179.0	159.3	8.8	125.1	22.1
data1	C_1	1 823.0	1 870.3	1 803.7	23.0	79.5	44.7
	C_2	2 985.0	2 062.3	1 872.7	24.1	79.8	44.6
	C_3	1 727.3	1 725.3	1 715.3	23.1	78.0	44.8

注:表中黑体数据表示在代价矩阵下的最优分类代价。

从表 2 的分类结果可以看出,MCCSADA 算法具有代价敏感特性,其分类代价几乎在所有情况下都小于 SAMME 算法。从表 2 中的代价结果可以看出,在数据分类正确率较高的情况下,3 种算法的分类代价相差不大,这是因为分类错误的的数据较少,并且 3 种算法都在向降低分类代价的趋势迭代。从表 2 中还可以看出:MCCSADA 算法在 wine、vehicle、yeast、vowel、soybean 和 data1 数据集上取得了比 CSOVO 算法更好的代价敏感效果,这更加证明了 MCCSADA 算法代价敏感特性的可信度;MCCSADA 算法在可区分性较差的数据集上表现一般,这说明 CCSADA 算法在面对类与类之间重叠较严重的数据时,其代价敏感效果较差,因此 MCCSADA 算法还需要进一步研究改进。

观察表 2 中 3 种算法的分类时间,在 2 类数据集上 CSOVO 和 MCCSADA 算法分类时间大概相同,比 SAMME 算法稍长;在 多类数据集上,MCCSADA 算法的分类时间比 CSOVO 算法短,并且随着类别数目的增多,CSOVO 算法的分类时间快速增长,而 MCCSADA 算法分类时间增长较慢。MCCSADA 算法在一定程度上解决了多类代价敏感算法时间复杂度高的问题,但是相比 SAMME 算法,其时间复杂度依然较高,还需要进一步改进。

为了清楚地展现出 MCCSADA 算法的分类代价随迭代次数增加的变化情况以及同其他 2 种算法分类效果的对比情况,给出了几种算法在实验数据上分类代价随迭代次数变化的对比图,如图 1~3 所示。

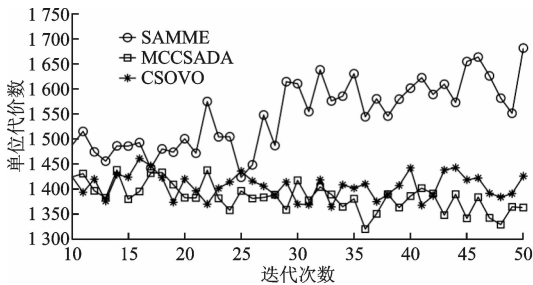


图 1 3 种算法在 yeast 数据集上的分类代价对比

从图 1 可以看出: SAMME 算法的分类代价随着迭代次数增加逐渐上升,这可能是由于 SAMME 算法陷入退化;MCCSADA 和 CSOVO 两种算法的分类代价逐渐下降并保持稳定状态,反映了 MCCSADA 算法的稳定性。从图 1~3 可以看出,MCCSADA 算法具有代价敏感特性和 Boosting 特性,算法的分类代价小于 SAMME 算法,并且达到了不弱于

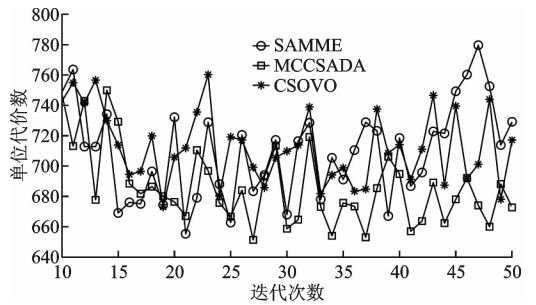


图 2 3 种算法在 vehicle 数据集上的分类代价对比

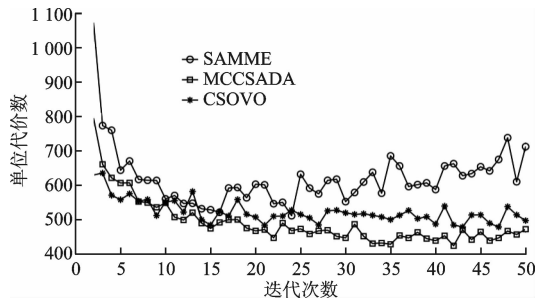


图 3 3 种算法在 vowel 数据集上的分类代价对比

CSOVO 算法的效果,甚至超越了 CSOVO 算法,算法的分类代价随迭代次数增加逐渐减小并趋于稳定。3 种算法的分类代价随迭代次数变化的对比结果证明了 MCCSADA 算法的有效性。

4 结 论

为解决现有多类代价敏感 AdaBoost 算法时间复杂度高和不能区分错分代价的问题,本文提出了一种采用多类代价指数损失函数的多类代价敏感 AdaBoost 算法,算法利用逐步叠加模型直接扩展二类代价敏感 AdaBoost 算法到多类。多类代价指数损失函数和逐步叠加模型保证了算法的代价敏感特性和 Boosting 特性,从理论上保证了算法的有效性。从实验结果可以看出,相比于由一对一方法扩展而来的代价敏感 AdaBoost 算法,本文所提 MCCSADA 算法具有更低的分类代价和时间复杂度。但是,MCCSADA 算法在数据分布复杂的情况下效果一般,而且相比于 SAMME 算法时间复杂度还具有一定的提升空间,以上两个问题还需要进一步研究。

参考文献:

[1] LOPEZ V, FERNANDEZ A, MORENO-TORRES J G, et al. Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification [J]. Expert Systems with Applications, 2012, 39(7): 6585-6608.

- [2] FREUND Y, SCHAPIRE R E. A decision theoretic generalization of on-line learning and an application to boosting [M]// Lecture Notes in Computer Science; Vol 904. Berlin, Germany: Springer Verlag, 1995: 23-37.
- [3] TING K M, ZHENG Z. Boosting trees for cost-sensitive classification [M]// Lecture Notes in Computer Science; Vol 1398. Berlin, Germany: Springer Verlag, 1999: 190-195.
- [4] FAN W, STOLFO S J, ZHANG J, et al. AdaCost: misclassification cost-sensitive boosting [C]//Proceedings of the 16th International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999: 97-105.
- [5] SUN Y, WONG A, WANG Y. Parameter inference of cost-sensitive boosting algorithms [C]//Proceedings of the 4th International Conference on Machine Learning and Data Mining in Pattern Recognition. Berlin, Germany: Springer-Verlag, 2005: 21-30.
- [6] SUN Y, KAMEL M S, WONG A, et al. Cost-sensitive boosting for classification of imbalanced data [J]. Pattern Recognition, 2007, 40(12): 3358-3378.
- [7] MASNADI-SHIRAZI H, VASCONCELOS N. Asymmetric boosting [C]//Proceedings of the 24th International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007: 609-619.
- [8] MASNADI-SHIRAZI H, VASCONCELOS N. Cost-sensitive boosting [J]. Pattern Analysis and Machine Intelligence, 2010, 33(2): 294-309.
- [9] 付忠良. 多标签代价敏感分类集成学习算法 [J]. 自动化学报, 2014, 40(6): 1075-1080.
FU Zhongliang. Cost-sensitive ensemble learning algorithm for multi-label classification problems [J]. Acta Automatica Sinica, 2014, 40(6): 1075-1080.
- [10] 付忠良. 多分类代价敏感 AdaBoost 算法 [J]. 自动化学报, 2011, 37(8): 973-983.
FU Zhongliang. Cost-sensitive AdaBoost algorithm for multi-class classification problems [J]. Acta Automatica Sinica, 2011, 37(8): 973-983.
- [11] PAN Shirui, WU Jia, ZHU Xingquan. CogBoost: boosting for fast cost-sensitive graph classification [J]. Knowledge and Data Engineering, 2015, 27(11): 2933-2946.
- [12] YUAN Xiaohui, MOHAMED A. A multi-class boosting method for learning from imbalanced data [J]. Rough Sets and Intelligent Systems, 2015, 4(1): 13-28.
- [13] 曹莹, 苗启广, 刘家辰, 等. 具有 Fisher 一致性的代价敏感 Boosting 算法 [J]. 软件学报, 2013, 24(11): 2584-2596.
CAO Ying, MIAO Qiguang, LIU Jiacheng, et al. Fisher consistent cost sensitive boosting algorithm [J]. Journal of Software, 2013, 24(11): 2584-2596.
- [14] ZHU J, ZOU H, ROSSET S, et al. Multi-class AdaBoost [J]. Statistics and Its Interface, 2009, 2(3): 349-360.
- [15] LU Hengyang, CHEN Feiyu, XU Ming, et al. Never ignore the significance of different anomalies: a cost-sensitive algorithm based on loss function for anomaly detection [C]//Proceedings of the 27th International Conference on Tools with Artificial Intelligence. Piscataway, NJ, USA: IEEE, 2015: 1099-1105.
- [16] 李秋洁, 赵亚琴, 顾洲. 代价敏感学习中的损失函数设计 [J]. 控制理论与应用, 2015, 32(5): 689-694.
LI Qiujie, ZHAO Yaqin, GU Zhou. Design of loss function for cost-sensitive learning [J]. Control Theory and Applications, 2015, 32(5): 689-694.

[本刊相关文献链接]

- 付学中, 方宗德, 关亚彬, 等. 采用 NSGA-II 算法的面齿轮副小轮拓扑修形多目标优化. 2017, 51(7): 98-104. [doi: 10.7652/xjtuxb201707015]
- 李莉, 冯林, 吴俊, 等. 一种三结构描述子的图像检索方法. 2017, 51(6): 86-91. [doi: 10.7652/xjtuxb201706014]
- 杨扬, Chun-Ta LU, 王菲菲, 等. 众筹项目的社交网络影响力预测与分析. 2017, 51(04): 91-96. [doi: 10.7652/xjtuxb201704014]
- 张国和, 黄凯, 张斌, 等. 最大稳定极值区域与笔画宽度变换的自然场景文本提取方法. 2017, 51(01): 135-140. [doi: 10.7652/xjtuxb201701021]
- 龚春园, 梁晋, 温广瑞, 等. 投影仪全参数平面线性估计的高精度标定方法. 2016, 50(11): 36-42. [doi: 10.7652/xjtuxb201611006]
- 高智勇, 董荣光, 高建民, 等. 采用聚类特征的基本概率分配生成方法及应用. 2016, 50(10): 8-14. [doi: 10.7652/xjtuxb201610002]
- 刘星, 梅雪松, 陶涛, 等. 一种零耦合滚齿全误差模型及其预测方法. 2016, 50(10): 42-48. [doi: 10.7652/xjtuxb201610007]
- 杨源, 库涛, 查宇飞, 等. 快速多特征金字塔的尺度目标跟踪方法. 2016, 50(10): 49-56. [doi: 10.7652/xjtuxb201610008]
- 张胜杰, 查宇飞, 李运强, 等. 一种利用局部结构信息的加权哈希图图像检索算法. 2016, 50(10): 78-85. [doi: 10.7652/xjtuxb201610012]

(编辑 刘杨)