软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(5):1195−1206 [doi: 10.3724/SP.J.1001.2012.04104]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于加权边界度的稀有类检测算法*

黄 浩, 何钦铭+, 陈 奇, 钱 烽, 何江峰, 马连航

(浙江大学 计算机科学与技术学院,浙江 杭州 310027)

Rare Category Detection Algorithm Based on Weighted Boundary Degree

HUANG Hao, HE Qin-Ming+, CHEN Qi, QIAN Feng, HE Jiang-Feng, MA Lian-Hang

(College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China)

+ Corresponding author: E-mail: hqm@zju.edu.cn, http://www.zju.edu.cn

Huang H, He QM, Chen Q, Qian F, He JF, Ma LH. Rare category detection algorithm based on weighted boundary degree. *Journal of Software*, 2012,23(5):1195−1206. http://www.jos.org.cn/1000-9825/4104.htm

Abstract: This paper proposes an efficient algorithm named CATION (rare category detection algorithm based on weighted boundary degree) for rare category detection. By employing a rare-category criterion known as weighted boundary degree (WBD), this algorithm can make use of reverse *k*-nearest neighbors to help find the boundary points of rare categories and selects the boundary points with maximum WBDs for labeling. Extensive experimental results demonstrate that this algorithm avoids the limitations of existing approaches, has a significantly better efficiency on discovering new categories in data sets, and effectively reduces runtime, compared against the existing approaches.

Key words: rare category detection; boundary point detection; weighted boundary degree; *k*-nearest neighbor; reverse *k*-nearest neighbor

摘 要: 提出了一种快速的稀有类检测算法——CATION(rare category detection algorithm based on weighted boundary degree).通过使用加权边界度(weighted boundary degree,简称 WBD)这一新的稀有类检测标准,该算法可利用反向 *k* 近邻的特性来寻找稀有类的边界点,并选取加权边界度最高的边界点询问其类别标签.实验结果表明,与现有方法相比,该算法避免了现有方法的局限性,大幅度地提高了发现数据集中各个类的效率,并有效地缩短了算法运行所需要的运行时间.

关键词: 稀有类检测;边界点检测;加权边界度;*k* 近邻;反向 *k* 近邻

中图法分类号: TP391 **文献标识码**: A

稀有类检测也称类检测,其目标是发现无类别标签的数据集中存在哪些类,特别是存在哪些稀有类.相对于样本数量占绝对优势的主要类而言,稀有类仅包含少量数据样本.例如,绝大多数的金融交易都是合法的,只有少量交易是利用金融系统的漏洞或采取特定欺诈手段进行的违法操作[1];海量的正常网络访问中包含着少量利用系统漏洞或具有欺诈性质的恶意网络行为[2,3].传统的离群点[4]与普通数据的一般行为或模型不一致,而稀有类的数据样本却往往能够伪装成普通数据[5−7]隐藏于各个主要类中,难以发现.稀有类检测算法面临的主要

* 基金项目: 教育部-英特尔信息技术专项科研基金(MOE-INTEL-11-06)
收稿时间: 2011-05-11; 修改时间: 2011-07-01; 定稿时间: 2011-08-24

挑战就是如何发现这些隐藏的稀有类.为此,这类算法通常在数据集中选取最有可能属于稀有类的数据样本,并向专家询问所选样本的类别.不同于聚类、分类等问题需要识别出每个类的全部数据样本,在稀有类检测中,一旦发现了某类的一个以上的数据样本,即可宣布发现该类存在;当发现数据集中全部类时,即宣告工作结束;专家不必遍历整个数据集,只需告知所选数据样本的类别即可.较好的稀有类检测算法要尽可能地减少发现全部类时所需的询问次数,即使得专家的工作量最小化.理论上,性能最好的稀有类检测算法每次选取的数据样本都应该属于一个新的类.

稀有类检测的结果也有许多用途.由于稀有类检测为每个类都找到了部分数据样本,因此其检测结果可被用于许多半监督的学习方法,如协同训练[8,9]与主动学习[10,11];另外,我们还可以基于检测结果,特别是其中来自稀有类的数据样本,对稀有类进行描绘[12],即估计数据集中还有哪些数据样本属于对应的稀有类.

根据不同的工作原理,现有的稀有类检测算法可以分为基于模型的方法[13]、基于离群程度的方法[14]、基于密度差异的方法[5−7].然而,这些方法在用于真实数据集时都存在着一些局限性.例如,He 等人[5]指出,基于模型的方法往往需要稀有类与主要类在数据分布上线性可分,否则此类方法难以发挥其最佳效果.基于离群程度的方法通常在远离正常数据的簇中选择数据样本进行询问,也更适于稀有类与主要类线性可分的情况.事实上,稀有类往往会隐藏在主要类当中,例如,金融欺诈操作往往会伪装成合法的金融操作,即稀有类数据与主要类数据常重叠在一起,线性不可分.另外,基于密度差异的方法都是在密度变化较为剧烈的区域选择数据样本进行询问.但在真实数据集中,常常存在着一些稀有类,其数据样本与其周边的数据样本的密度差异并不是十分明显,要发现这样的稀有类,基于密度差异的方法所需的询问次数会大幅增加,甚至有时无法发现这样的稀有类.

为了避免现有算法的局限性,本文提出了一种基于加权边界度(weighted boundary degree,简称 WBD)的稀有类检测算法 CATION(rare category detection algorithm based on weighted boundary degree).该算法利用稀有类与主要类的在数据分布上的差异,即稀有类往往在小范围内集中出现、主要类分布范围较大且在小范围内的数据分布局部平滑[5−7],并利用数据点的反向 k 近邻个数随数据分布的不同而变化这一性质[15],通过统计小范围内数据点的反向 k 近邻平均个数来发现稀有类的边界区域;同时,以这些数据点的反向 k 近邻个数的最大差值作为权重,使得我们在大致锁定稀有类边界区域的同时,可以在更加靠近稀有类内部的一端选择数据样本进行询问,从而提高发现稀有类的概率.

本文主要贡献可概括为:(1) 本文充分利用反向 k 近邻的独特性质,提出了一个新的稀有类检测标准,即加权边界度,该标准能够帮助我们避免现有稀有类检测算法的局限性;(2) 基于加权边界度,本文提出了一种快速的稀有类检测算法,即 CATION 算法;(3) 本文给出了丰富实验,证明了在对数据集中各个类的发现效率上,CATION 算法明显优于现有稀有类检测算法,并证明了本文提出的加权边界度比现有边界点检测方法更适用于稀有类检测.

本文第 1 节回顾现有相关工作.第 2 节详细介绍本文所提出的 CATION 算法.第 3 节给出一系列的实验结果与对应的分析结果.第 4 节总结全文,并给出未来工作的方向.

1 相关工作

本文所提出的稀有类检测算法是一种使用边界点检测技术的稀有类检测算法.因此,本节将回顾稀有类检测、边界点检测两个方面的相关工作.

1.1 稀有类检测

现有的稀有类检测算法可大致分为 3 类,即基于模型的方法、基于密度差异的方法和基于离群程度的方法.

1.1.1 基于模型的方法

Pelleg 等人[13]首先提出了稀有类检测这一问题,并且介绍了 Interleave 算法.该算法假设数据是由模型产生的,例如高斯混合模型,并且使用高斯贝叶斯分类器来选出最不满足该模型的数据点.在询问过这些选出的数据点的类别标签后,再使用半监督的训练方式更新该分类器.Interleave 算法不断重复选点、询问与更新分类器这 3 步,直至用户停止这一循环.该算法的时间复杂度为 $O(d\cdot n^2)$,其中,d 为维度,n 为数据集中数据样本的数目.需要

指出的是,该算法往往在稀有类和主要类线性可分的情况下才能有较好的表现[5].

1.1.2 基于密度差异的方法

He 等人提出了 3 种基于密度差值的方法,即 NNDM 算法[5]、GRADE 算法[6]和 SEDER 算法[7].与 Interleave 算法相比,这 3 种方法不要求稀有类与主要类线性可分,而是假设在稀有类出现之处,局部密度会发生剧烈变化;而在局部密度变化较大的区域选择数据样本进行询问,则会有较高几率发现稀有类.为估计出每个数据点的局部密度,NNDM 算法与 GRADE 算法以目标点为球心划超球,并将超球内点的数目作为目标点局部密度.划超球时,NNDM 直接采用欧式距离,时间复杂度为 $O(d\cdot n^{2-1/d}+d\cdot n\cdot \log n)$;而 GRADE 则利用谱分析技术计算两两点对之间的相似度,时间复杂度皆为 $O(d\cdot n^2)$.SEDER 算法使用半参密度估计直接获得每一处的最大密度变化率,时间复杂度为 $O(d^2\cdot n^2)$.需要指出的是,基于密度差异的方法在用于真实数据集时往往会有这样的问题,即当数据集中存在一些“隐藏”很好的稀有类,如与周边数据密度差异不是十分明显的稀有类时,此类方法所需要的询问次数往往会大幅增加,甚至可能无法发现这样的稀有类.

1.1.3 基于离群程度的方法

Vatturi 等人提出了一种基于离群程度的稀有类检测算法,即 HMS 算法[14].该算法首先利用多层次的均值平移获得多层次的聚类结果,继而从所有聚类中选取分布范围小且远离其他数据点的聚类,并询问这些聚类的最靠近聚类中心的数据样本的类别标签.该算法的时间复杂度通常不低于 $O(d\cdot n^2)$.由于 HMS 算法将数据点的离群程度作为稀有类的特征之一,故而该算法与基于模型的方法类似,更适合于处理稀有类远离主要类的情况.

1.2 边界点检测

不同于稀有类检测,边界点检测的目标是提取出聚类的边界,并不关注边界点的确切类别,即其选取边界点实际上既可能是某类数据的靠近其边界的内部数据点,也很有可能是靠近该边界的其他类的数据点.但是,在稀有类边界区域选取数据样本的基本思路却可以用于稀有类检测,本文算法正是基于这一思路提出的.为此,这一节中我们将回顾现有的几种主要的边界点检测算法,包括 BORDER 算法、BRIM 算法和 BAND 算法.

BORDER 算法[15]是基于反向 k 近邻的边界点检测算法.但与本文的加权边界度不同,BORDER 算法仅仅考察每个数据点的反向 k 近邻个数,并将该个数小于某阈值的点判为边界点.薛丽香等人[16]指出,BORDER 算法运行于无噪音的数据集上时,能够有效地提取出聚类的边界;但是当噪音存在时,该算法性能会受到较大影响.

BRIM 算法[17]可以提取任意形状、大小、密度的聚类的边界,同时也对噪音表现出较强的鲁棒性.该算法以每个数据点为圆心、以指定半径划超球,并将超球划分为包含数据点较多的正邻域和包含数据点较少的负邻域,继而利用正负邻域中点数的差值来判定该数据点是否是聚类边界点.

BAND 算法[16]是基于变异系数的边界点检测算法.该算法利用数据点到其 k 近邻平均距离的倒数作为该点的局部密度,并将数据点及其 k 近邻的局部密度的标准差与它们的密度均值的比值作为变异系数.当变异系数大于某阈值时,该点即判为聚类边界点.

2 基于加权边界度的稀有类检测算法

2.1 相关概念

给定无标签数据集 $D=\{x_1,x_2,\ldots,x_n\}$,$x_i\in \boldsymbol{R}^d$,$1\leqslant i\leqslant n$,其中,d 为维度.D 中数据样本共来自于 m 个类,其中,x_i 的类别标签记为 y_i,$y_i\in \mathbf{Z}^+$且 $1\leqslant y_i\leqslant m$.由于 D 为无标签数据集,因此在运行稀有类检测算法之前,用户并不知道每个 y_i 的具体值.因此,对于每一个由算法选出的数据样本,我们都需要向专家询问其确切的类别标签.另外,我们遵循 He 等人[5,6]关于先验知识的假设,即用户对各个类的数据样本数量在整个数据集中所占比例 $p^1,p^2,\ldots,p^m$ 具有先验知识.出于一般性的考虑,我们规定:将主要类的类别标签记为 1,即 $p^1>>p^j(j=2,3,\ldots,m)$.我们的目标是使用尽量少的询问次数为 D 中每个类发现至少 1 个数据样本.下面给出文中将要使用的相关定义.

定义 1(稀有类). 在一个数据集中,稀有类指的是少量数据样本由于在小范围内集中出现而形成的小型簇.

稀有类一般具有以下 3 个特征:(1) 每个稀有类所包含的数据样本量非常少;(2) 每个稀有类的数据样本一

般集中的出现在某个局部区域;(3) 稀有类往往被大量的主要类的数据样本所包围,隐藏在主要类内部.我们可以将稀有类与主要类的数据分布情况概化成如图 1 所示的一维数据分布.其中,可将点 $p_8,p_9,p_{10},p_{11},p_{12}$ 构成的聚类视为在小范围内集中出现的、隐藏在主要类数据中的稀有类,可将其余点视为在小范围内数据分布局部平滑的主要类的部分数据点.当然,如图 1 所示的数据分布情况完全可以推广到高维空间中,且稀有类与主要类也可能呈其他分布,如高斯分布等.但是文中为简明起见,仅以一维均匀分布为例.

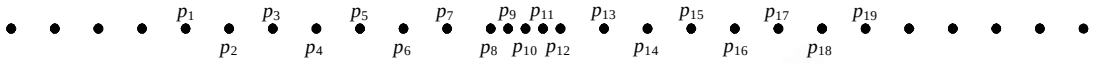


Fig.1 Example of rare category in 1-dimensional data set

图 1 一维数据集中的稀有类实例

定义 2(反向 k 近邻)[15]. 对任意的正整数 k 和数据集 D,点 p 的反向 k 近邻记作 $RkNN(p)$,定义为一组点 $p_i\in D$,其中,每一个 p_i 都满足 $p\in kNN(p_i)$.这里,$kNN(p_i)$代表点 p_i 的 k 近邻.

与 k 近邻相比,数据点的反向 k 近邻的个数不是固定的,会随着数据分布的变化而变化[15].如在图 1 中,当 k=4 时,点 $p_5,p_6,p_7,p_{13},p_{14},p_{15}$ 具有 3 个反向 k 近邻,点 $p_1,p_2,p_3,p_4,p_{16},p_{17},p_{18},p_{19}$ 具有 4 个反向 k 近邻,点 p_8,p_9,p_{11},p_{12} 具有 5 个反向 k 近邻,点 p_{10} 具有 6 个反向 k 近邻.

结合图 1 我们可以发现,反向 k 近邻有如下性质:

性质 1. 当数据分布局部平滑时,数据点一般具有相同数量的反向 k 近邻.

如图 1 中点 p_1,p_2,p_3,p_4 以及点 $p_{16},p_{17},p_{18},p_{19}$.这是因为此时各数据点都会从相似的方向和相对位置上找到 k 近邻,致使各点寻找反向 k 近邻时的方向与相对位置也相似,从而使各点的反向 k 近邻的数量也一致.

性质 2. 在聚类边界附近、远离聚类内部的一端,数据点的反向 k 近邻的个数会有所减少.

如图 1 中点 p_5,p_6,p_7 以及点 p_{13},p_{14},p_{15}.这是因为靠近这些点的聚类内部点拥有足够的邻居,从而没有将这些点纳入其 k 近邻,使得这些点的反向 k 近邻的个数少于处于数据分布局部平滑区域的点.

性质 3. 在聚类边界附近、靠近聚类内部的一端,数据点的反向 k 近邻的个数高于远离聚类内部的一端.

如图 1 中点 p_8,p_9 以及点 p_{11},p_{12}.这是因为与远离聚类内部一端的边界点相比,这些点靠近更多的聚类内部点,即这些点成为其他点的 k 近邻的概率高于远离聚类内部一端的边界点,使得这些点的反向 k 近邻的个数也将高于远离聚类内部一端的边界点.

定义 3(边界度). 对任意的正整数 k 和数据集 D,点 p 的边界度记作 $BD(p,k)$,定义为

$$BD(p,k)=\frac{\max_{q\in D}|RkNN(q)|-\min_{q\in D}|RkNN(q)|}{\frac{1}{k+1}\sum_{q\in p\cup kNN(p)}|RkNN(q)|-\min_{q\in D}|RkNN(q)|} \tag{1}$$

$BD(p,k)$的分子记录的是全局最大反向 k 近邻个数与全局最小反向 k 近邻个数的差值;$BD(p,k)$的分母记录的是点 p 与其 k 近邻共 k+1 个点的平均反向 k 近邻个数与全局最小反向 k 近邻个数的差值.对于数据集 D 来说,$\max_{q\in D}|RkNN(q)|$与 $\min_{q\in D}|RkNN(q)|$是常数.因此,$BD(p,k)$的大小仅与点 p 及其 k 近邻的反向 k 近邻的平均个数成反比.由反向 k 近邻的性质 2 可知,稀有类的出现一般将导致其聚类边界周边小范围内的数据点反向 k 近邻个数的下降.因此,通过寻找较高的 $BD(p,k)$值,我们可以大致地锁定稀有类边界可能出现的区域.

定义 4(权重). 对任意的正整数 k 和数据集 D,点 p 的权重记作 $weight(p,k)$,定义为

$$weight(p,k)=\frac{|RkNN(p)|+1}{\min_{q\in kNN(p)}|RkNN(q)|+1} \tag{2}$$

$weight(p,k)$记录的是点 p 的反向 k 近邻个数加 1 后与其 k 近邻的最小反向 k 近邻个数加 1 后的比值,其中,加 1 是为了防止分母为 0.权重反映的是点 p 与其 k 近邻之间在反向 k 近邻个数上的最大差异,可以帮助我们进一步突出数据分布变化较大的局部区域.另外,在同一个局部区域中,拥有较多的反向 k 近邻的数据点将拥有较高的权重.由反向 k 近邻的性质 3 可知,靠近聚类内部一端的边界点一般要比远离聚类内部一端的边界点具有更多的反向 k 近邻.因此,靠近聚类内部一端的边界点将拥有较高的权重.

定义 5(加权边界度). 对任意的正整数 k 和数据集 D,点 p 的加权边界度记作 $WBD(p,k)$,定义为

$$WBD(p,k)=weight(p,k)\times BD(p,k) \tag{3}$$

$WBD(p,k)$记录的是每个点边界度与权重的乘积.根据边界度和权重的定义,靠近聚类内部一端的边界点会具有较高的加权边界度,而这类点是稀有类的数据样本的概率更大.因此,结合边界度与权重,将使得我们在大致锁定稀有类的边界的同时,能够尽可能地提高发现稀有类的概率.

2.2 CATION算法描述

在真实数据集中,稀有类往往隐藏在主要类当中,难以直接发现.但是稀有类通常在小范围内集中出现,而主要类在小范围内的数据分布局部平滑,从而造成在稀有类边界附近存在数据分布上的差异.我们可以利用反向 k 近邻个数的变化来捕获这种数据分布上的差异,发现稀有类的聚类边界,并选取更加靠近聚类内部的边界点,询问其类别.为此,本文提出加权边界度来实现这一目标.

为了减少不必要的询问,在选取数据样本前,我们将离群点(outlier)纳入免责列表,避免它们被选到.这是因为离群点往往孤立地出现于远离正常数据的低密度区域,而我们要发现的稀有类却是常常隐藏主要类中的小而紧实的聚类.另外,在每次询问后,为了防止在同一区域继续重复地选择数据样本,所选样本周边一定范围内的数据点也将被纳入免责列表.

算法 1 列出了 CATION 算法的伪代码.该算法的工作步骤如下:

(1) 当给定无标签数据集 D、稀有类的先验百分比 $p_i(i=2,...,m)$时,首先我们初始化一组空集合,包括用于记录所选数据样本的集合 I、用于记录所选数据样本类别标签的集合 L 以及用于记录哪些数据点被排除在待选范围之外的免责列表 EL.

(2) 然后,在步骤 2 中,我们利用基于密度的局部离群点检测算法 LOF 算法[18]判断哪些点是局部离群点,并将局部离群点加入免责列表.使用LOF算法时需要输入一个正整数参数 mp,该参数规定了用于识别基于密度的簇的最少的点数.不过,LOF 算法的一个优点在于它对 mp 值的选取具有一定的鲁棒性.本文中,我们令 $mp=\min\{n\cdot p^i|2\leqslant i\leqslant m\}$.选取该 mp 值主要是为了在使用 LOF 算法时防止稀有类的数据样本被加入免责列表.因为 D 中各个稀有类聚类所包含的数据样本数大于等于该 mp 值,即以 mp 近邻来判定局部离群点,稀有类聚类中的点被判定为局部离群点风险很小.

(3) 在步骤 3~步骤 7 的循环中,对于每一个稀有类 i,我们首先估算该类的数据样本的数目 k_i,找出每个数据点的 k_i 近邻,记录下数据点至其第 k_i 个近邻的距离的全局最小值,然后统计出每个数据点的反向 k_i 近邻个数.

(4) 在步骤 9~步骤 17 的循环中,我们首先检查还有哪些类未发现数据样本,并选取其中 k_i 值最小者,以 c 记录该类的类别标签,以 k'记录对应的 k_i 值;然后,我们以 k'为参数,计算每一个未被免责的数据点的加权边界度;继而在步骤 12~步骤 16 的循环中不断选取当前加权边界度最高的数据点 x,询问该点类别标签 y,直至为类 c 发现一个数据样本.为避免在同一区域重复选取数据样本,与询问点 x 之间的距离小于 r_y 的数据点将进入免责列表.由于 r_y 是数据点至其第 k_y 个近邻的距离的全局最小值,因此,被免责的数据点的个数小于等于 x 所在类的样本数目 k_y,从而避免过度免责影响到周边的其他类,特别是还未被发现的稀有类.

算法 1. 基于加权边界度的稀有类检测算法(CATION).

输入:无标签数据集 D、稀有类的先验百分比 $p^2,\ldots,p^m$.

输出:所选数据样本集合 I、所选数据样本类别标签集合 L.

1: 令 $I=\varnothing,L=\varnothing$,免责列表 $EL=\varnothing$.

2: $EL=EL\cup\{p|p\in D$,且 p 被 LOF 算法判定为局部离群点$\}$.

3: **for** $i=2:m$

4: 令 $k_i=|D|\cdot p^i$.

5: 对每个 $p\in D$,找到其 $k_iNN(p)$,并记录 p 与其第 k_i 个最近邻之间的距离 $dist_p(k_i)$,令 $r_i=\min_{p\in D}dist_p(k_i)$.

6: 对每个 $q\in D\backslash EL$,统计 q 在所有 $k_iNN(p)$中出现的次数,即得点 q 的反向 k_i 近邻的个数,即$|Rk_iNN(q)|$.

7: **end for**

8: 令 $r_1=\max\{r_i|2<i<m\}$.
9: **while** 已发现的类别数$<m$
10: 令 $k'=\min\{k_i|2<i<m$,且还未发现类 i 的数据样本$\}$,且对应类别标签为 c,即 $k'=k_c$.
11: 对每个 $q\in D\backslash EL$,按照公式(3)计算其加权边界度 $WBD(q,k')$.
12: **for** $t=1:|D|$
13: 询问 $x=\text{argmax}_{q\in D\backslash EL}(WBD(q,k'))$的类别标签 y.
14: $EL=EL\cup\{p|p\in D,\|p-x\|\leqslant r_y\}$.
15: if $y=c$, break.
16: **end for**
17: **end while**

2.3 复杂度分析

在CATION算法的主要步骤中,在判断离群点时,选用薛安荣等人[18]提出的基于多维索引技术的LOF算法,时间复杂度为 $O(d\cdot n\cdot\log n)$,其中,d 为维度,n 为数据集中数据样本的数目;利用 K-d 树[19]确定 k 近邻时,建立 K-d 树的时间复杂度为 $O(d\cdot n\cdot\log n)$,寻找 k 近邻的时间复杂度为 $O(d\cdot n^{2-1/d})$;确定 k 近邻后,用于统计数据点的反向 k 近邻个数的时间为 $O(k\cdot n)$;用于计算加权边界度的时间为 $O(k\cdot n)$.综上,CATION 的时间复杂度为 $O(d\cdot n^{2-1/d}+d\cdot n\cdot\log n)$.此复杂度低于现有的绝大部分稀有类检测算法,与现有算法中时间复杂度最低的 NNDM 算法持平,即理论上可以保证 CATION 算法不会比现有算法的运行速度慢.并且值得注意的是,CATION 算法避免了现有算法的局限性.实验结果表明,CATION 算法在稀有类检测的性能上明显优于现有算法.

3 实验结果与分析

在第 3.1 节中,我们将在人工数据集上验证 CATION 算法的有效性;在第 3.2 节中,我们首先将 CATION 算法与现有算法在真实数据集上进行比较,以证明 CATION 算法避免了现有算法的局限性,拥有更好的稀有类检测性能,且在时间效率上更具优势;另外,由于 CATION 算法是一种使用边界点检测技术的稀有类检测算法,因此,我们还将本文所提的加权边界度与现有边界点检测方法进行比较,以证明加权边界度更适用于稀有类检测.本文涉及的各种算法的编写与编译是在 MATLAB 7.9 中实现的.实验环境为 Inter Core2 Quo 2.8GHz CPU,2GB 内存,Microsoft Windows XP 专业版.

3.1 人工数据集

图 2 给出了一个 2 维人工数据集,其中,主要类包含 1 000 个数据样本,用浅灰色点表示.另有 4 个稀有类,分布在主要类的内部,分别包含 156,156,71,61 个数据样本,用深灰色点表示.当为每一个类发现至少 1 个数据样本时,CATION 算法共花费 6 次询问,所问各点的位置已用黑色圆点标出.

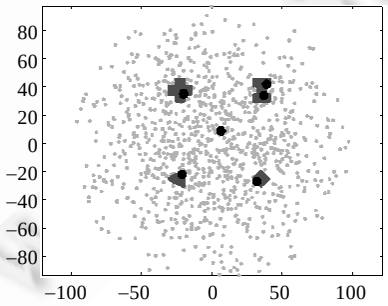
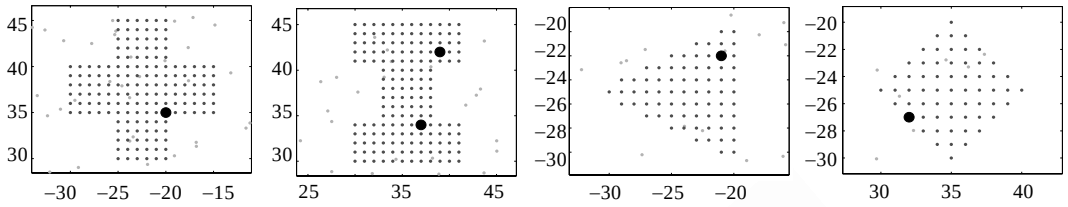


Fig.2 Two-Dimensional synthetic data set
图 2 二维人工数据集

为了更清楚地了解 CATION 算法所选的数据样本的位置,我们在图 3 中给出 4 个稀有类的局部放大图.由其中黑色圆点所在位置可知,CATION 算法所选数据点的确处于稀有类的聚类边界附近、靠近聚类内部的一端.



(a) (b) (c) (d)

Fig.3 Positions of rare-category examples selected by CATION

图 3 CATION 算法所选出的稀有类数据样本的位置

3.2 真实数据集

本文在以下 6 个来自 UCI 数据库[20]的真实数据集上进行实验:Abalone,Ecoli,Pen Digits,Statlog,Wine Quality 及 Yeast 数据集.表 1 列出了这些数据集的相关属性,其中,n 为数据集中数据样本的数目,d 为维度,m 为数据集中类的数目,“Largest Category”和“Smallest Category”两列分别列出数据集中最大类和最小类的数据样本在数据集中所占比例.遵循 HMS 算法[14]的实验设置,Pen Digits 与 Statlog 两个数据集是经过二次抽样的,因为在这两个数据集的原始版本中,每个类数据样本的数目几乎相同.二次抽样是为了获得不平衡的子集,因为不平衡的数据集更符合稀有类检测的应用场景.经过二次抽样,Pen Digits 与 Statlog 数据集内最大类分别包含 512 个与 256 个数据样本,次大类包含数据样本数减半,如此递减,直至两数据集的最小类都只有 8 个数据样本.

Table 1 Properties of the real data sets

表 1 真实数据集的相关属性

ata set	n	d	m	Proportion	
				Largest cat gory (%)	Smallest cat gory (%)
Abalone	4 177	7	20	16.50	0.34
Ecoli	336	7	6	42.56	2.68
Pen digits	1 040	16	10	49.23	0.77
Statlog	512	19	7	50.00	1.56
Wine quality	4 898	11	6	44.88	0.41
Yeast	1 484	8	10	31.20	0.34

3.2.1 与现有稀有类检测算法的比较分析

在本节中,我们将 CATION 算法和现有的基于密度差异的算法 NNDM 与 SEDER、基于离群程度的方法 HMS 以及随机采样方法(random sampling,简称 RS)在真实数据集上进行比较,以证明 CATION 算法避免了现有算法的局限性,能够利用更少的询问次数发现数据集中的所有类,且拥有较高的时间效率.

基于密度差异的算法首先考察数据点与其周边点之间密度的变化程度,然后按照变化程度从大到小的顺序选择数据样本进行询问.因此,当每个类中都存在密度变化程度大的点时,基于密度差异的算法可以很快地发现来自每个类的数据样本.然而在真实数据集中,很难保证每个类都存在密度变化程度大的点,特别是一些在主要类中隐藏得很好的稀有类,其每一个数据样本与周边的密度差异都不会很大.为了证明真实数据集存在这样的稀有类,我们利用 SEDER 算法所提出的 s 值来衡量数据集中每个点与周边点的密度差异,并对每个 s 值赋予一个降序的排名,同时将每个类中数据点所具有的最大 s 值与其对应排名作为该类的 s 值与排名.表 2 列出了每个数据集中 s 值最小的类 $i(1\leqslant i\leqslant m)$的类标签、该类的数据样本在数据集中所占比例、该类的 s 值 $s(i)$以及该 s 值在数据集中的排名,其中,n 为数据集中数据样本的数目.从表 2 中我们可以发现,真实数据集中一般都存在 s 值排名很低的稀有类,即与周边数据点的密度差异不明显的稀有类.显然,要发现这些 s 值排名极低的类的数据样本,按照密度差异从大到小的选择顺序,基于密度差异的算法需要花费较多的询问次数.

Table 2 Properties of the minimum-*s* category *i* in each real data set

表 2 真实数据集中 *s* 值最小的类 *i* 的相关属性

ata set	ategory *i*	Prop rtion (%)	*s*(*i*)	Rankir of *s*(*i*)/*n*
Abalone	2	0.36	2.40E−21	3061/4177
Ecoli	5	5.95	1.43E−17	123/336
Pen digits	6	1.54	1.36E−64	327/1040
Statlog	7	1.56	6.79E−48	116/512
Wine quality	2	0.41	8.45E−41	33/4898
Yeast	10	0.34	4.39E−25	683/1484

与基于密度差异的算法相比,CATION 算法不受密度差异大小的影响.为了证明这一点,图 4 给出 CATION 算法所选出的各个数据样本所对应的 *s* 值.可以发现,它们的 *s* 值并不呈明显的降序,即 CATION 算法不受密度差异大小的影响,可以避免基于密度差异的稀有类检测算法的局限性.

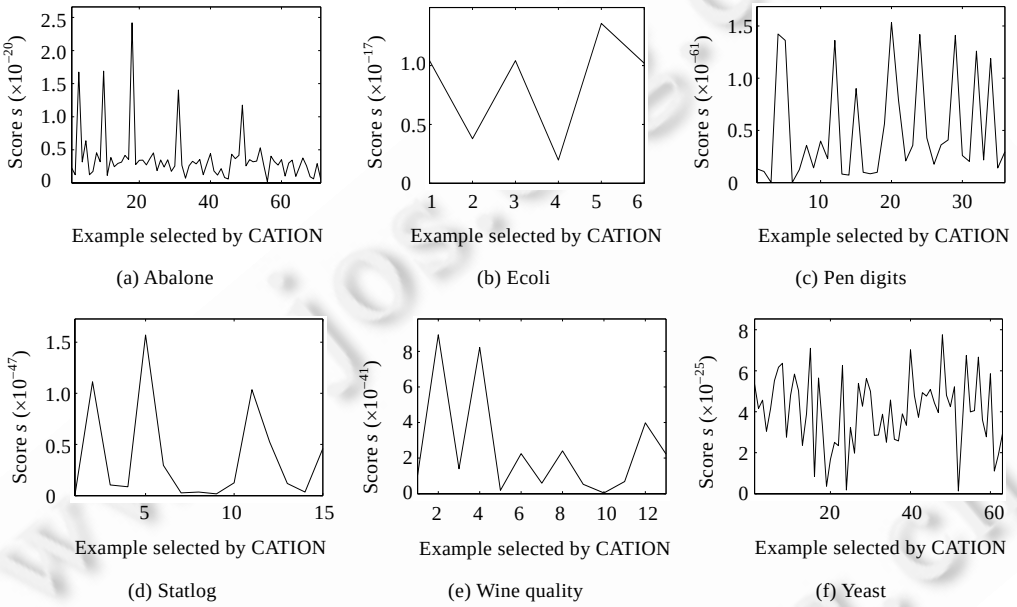


Fig.4 Score *s* of each example selected by CATION

图 4 CATION 算法所选的数据样本的 *s* 值

基于离群程度的 HMS 算法引入 Lueng 等人[21]的 isolation 指标、compactness 指标.这两个指标原本是用在多层次聚类中,用来衡量所形成的聚类是否属于好的、合理的聚类.HMS 算法直接用这两个指标来衡量一个簇可能是稀有类聚类的程度,却没有很好地给出理论上或基于观察的解释与动机,特别是对 isolation 指标的解释.事实上,文献[5−7,22]已经指出,真实数据集中的稀有类往往会隐藏在主要类中,并不会具有太高的离群程度.为了证明这一点,我们按 isolation 指标的定义计算每个类 $i(1\leqslant i\leqslant m)$的离群程度,记为 $iso(i)$.表 3 列出了每个数据集中 iso 值最小的类 *i* 的类标签、该类的数据样本在数据集中所占比例、该类的 iso 值 $iso(i)$与主要类 iso 值 $iso(1)$的对比以及该类的 CI 值 $CI(i)$,其中,CI 值是类 *i* 在 isolation 指标上的得分(即 iso 值)与在 compactness 指标上的得分之和.从表 3 中我们可以看出,在真实数据集中一般都存在 iso 值极小的稀有类,即隐藏在其他类内部的、隐蔽的稀有类.按照 CI 值由高到低的顺序,HMS 算法倾向于在离群程度高的紧实的小簇中选取数据样本.换言之,HMS 算法对发现那些隐蔽得很好的稀有类并没有先天的优势.

Table 3 Properties of the minimum-iso category *i* in each real data set

表 3 真实数据集中 iso 值最小的类 *i* 的相关属性

ata set	ategory *i*	Prop rtion (%)	*iso*(*i*) *so*(1)	*CI*(*i*)
Abalone	19	0.34	0.0076/0.2346	0.097 4
Ecoli	6	2.68	0.0274/0.4340	0.198 5
Pen digits	9	0.77	0.0078/0.5007	0.109 4
Statlog	7	1.56	0.0198/0.5464	0.197 8
Wine quality	2	0.41	7.5931E−23/0.2128	1.2135E−11
Yeast	10	0.34	0.0042/0.3172	0.123 5

与基于离群程度的算法相比,CATION 算法并不关注稀有类是否具有离群的特征,而是关注稀有类出现在主要类内部时造成的局部区域内数据分布的变化.因此,与 HMS 相比,CATION 算法更容易发现数据集中那些隐藏在主要类内部的稀有类.

图 5 给出了 CATION 算法与现有稀有类检测算法的详细的性能对比结果.图 5 中横坐标为询问次数,纵坐标为当前询问次数下已发现的类的数目.

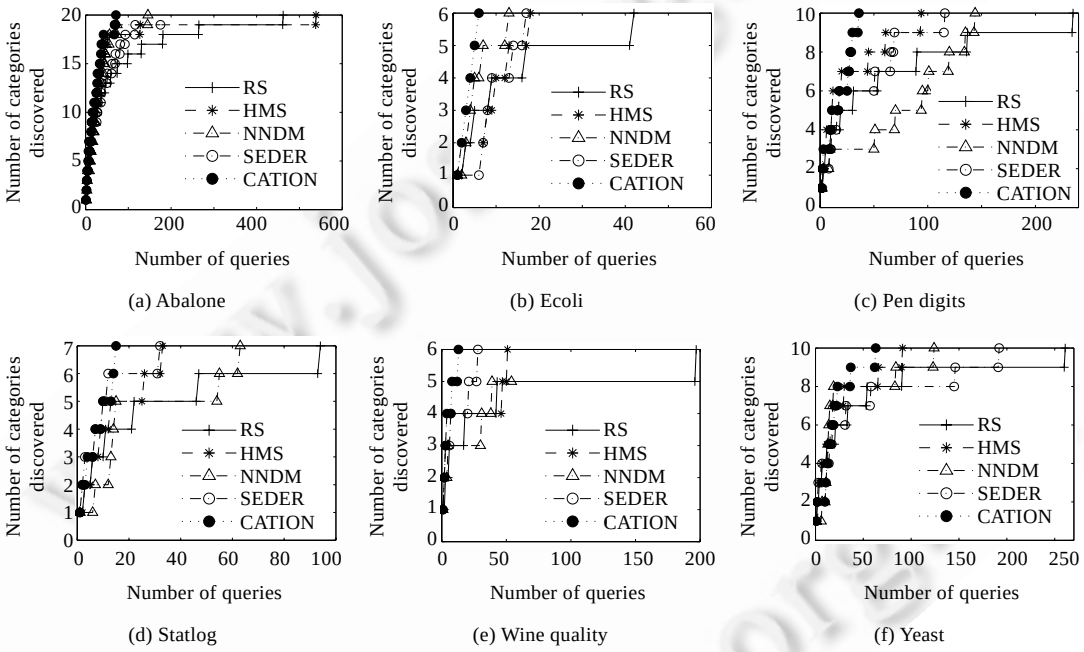


Fig.5 Comparison results on real data sets

图 5 真实数据集上的性能对比结果

由图 5 可知,CATION 算法的性能明显优于现有算法,即与现有算法相比,CATION 算法可以通过最少的询问次数发现各个数据集中的所有类.另外,在基于密度差异的算法中,NNDM 算法在 Wine Quality 数据集中只发现了 5 个类,SEDER 算法在 Abalone 数据集中只发现了 19 个类,都还有一个稀有类未能发现;基于离群程度的 HMS 算法在 Abalone 数据集中表现失色,需要 539 次询问才能发现全部的类,此时,其性能甚至比随机采样(RS)还略逊一筹.可见,CATION 算法除了在性能上优于现有算法以外,在不同数据集中的稳定性也要优于现有算法.

另外,为了验证 CATION 算法的时间效率,表 4 列出了发现全部类时各个算法所需的运行时间,单位为秒.由于随机采样(RS)不需要对数据集做任何分析,故而没有必要报告其运行时间.由表 4 我们可以看出,HMS 算法与 SEDER 算法的时间效率远不及 NNDM 算法与 CATION 算法;而与 NNDM 算法相比,CATION 算法除了 Statlog 数据集上落后 NNDM 算法约 0.02 秒之外,在其他数据集上的时间效率皆明显优于 NNDM 算法.

Table 4 Each algorithm's runtime (s)

表 4 各种算法的运行时间 (秒)

Data set	HMS	SEDER	NNDM	CATION
Abalone	3 789.75	166.98	46.88	29.12
Ecoli	9.98	1.04	0.10	0.06
Pen digits	151.64	44.97	1.82	1.07
Statlog	30.85	13.48	0.24	0.26
Wine quality	8 124.98	515.76	20.01	14.88
Yeast	284.33	25.42	5.26	1.87

3.2.2 与现有边界点检测算法的比较分析

CATION 算法利用加权边界度帮助用户在稀有类的聚类边界附近选取数据样本.为了证明加权边界度比现有的边界点检测算法更适用于稀有类检测,在本节中,我们使用现有边界点检测算法,包括 BORDER[15], BRIM[17]和 BAND[16],作为选择边界点的方法,将使用这些方法的 CATION 算法与使用加权边界度的 CATION 算法做性能比较.表 5 给出了比较结果,其中,"–"表示未能发现全部的类.由表 5 可知,使用 BORDER 与 BAND 时,发现全部类所需要的询问数一般高于使用加权边界度,且在 Wine Quality 数据集中都有无法发现的类;BRIM 方法在 Yeast 数据集中表现优于加权边界度,但是在 Abalone,Pen Digits,Statlog 与 Wine Quality 这 4 个数据集中,BRIM 却无法发现全部的类.因此,与现有的边界点检测算法相比,本文提出的加权边界度有更为良好且稳定的表现,更加适用于稀有类检测.

Table 5 Number of queries needed to discover all the categories using different boundary points detection approaches

表 5 使用不同的边界点检测方法时发现全部类所需的询问数

D a set	ORDER	BRIM	BAND	CATION
Abalone	180	—	92	71
Ecoli	32	25	22	6
Pen digits	49	—	43	36
Statlog	26	—	19	15
Wine quality	—	—	—	13
Yeast	113	52	89	63

4 结束语

本文利用反向 k 近邻的独特性质来发现数据集中稀有类的边界,并通过选取边界区域内靠近稀有类内部的数据样本,询问其类别来发现稀有类的数据样本.实验结果表明,与现有的稀有类检测算法相比,本文提出的基于加权边界度的稀有类检测算法能够避免现有算法的局限性,大幅地提高对数据集中各个类的发现效率,显著地减少发现数据集中全部类所需要的询问次数,并拥有较低的时间复杂度,使得用户可以花费更少的人力成本与时间成本来发现数据集中稀有类.另外,实验结果表明,与现有的边界点检测算法相比,本文提出的加权边界度方法更适用于稀有类检测.

本文提出的 CATION 算法需要用户具有一些先验知识,如类的个数、每个类的数据样本在数据集中所占比例.我们下一步的主要工作是改进 CATION 算法对先验知识的依赖,使其成为无需先验知识的稀有类检测算法.

References:

[1] Bay S, Kumaraswamy K, Anderle MG, Kumar R, Steier DM. Large scale detection of irregularities in accounting data. In: Proc. of the ICDM 2006. Washington: IEEE Computer Society, 2006. 75–86. [doi: 10.1109/ICDM.2006.93]

[2] Wu JJ, Xiong H, Wu P, Chen J. Local decomposition for rare class analysis. In: Proc. of the KDD 2007. New York: ACM Press, 2007. 814–823. [doi: 10.1145/1281192.1281279]

[3] Stokes JW, Platt JC, Kravis J, Shilman M. ALADIN: Active learning of anomalies to detect intrusions. Technical Report, MSR-TR-2008-24, Microsoft Research, 2008. http://research.microsoft.com/en-us/um/people/jstokes/aladintechreport.pdf

[4] Breunig MM, Kriegel HP, Ng RT, Sander J. LOF: Identifying density-based local outliers. In: Proc. of the SIGMOD 2000. New York: ACM Press, 2000. 93−104. [doi: 10.1145/335191.335388]

[5] He JR, Carbonell J. Nearest-Neighbor-Based active learning for rare category detection. In: Platt JC, Koller D, Singer Y, Roweis S, eds. Advances in Neural Information Processing Systems 20. Cambridge: MIT Press, 2008. 633−640. http://books.nips.cc/papers/files/nips20/NIPS2007_0051.pdf

[6] He JR, Liu Y, Lawrence R. Graph-Based rare category detection. In: Proc. of the ICDM 2008. Washington: IEEE Computer Society, 2008. 833−838. [doi: 10.1109/ICDM.2008.122]

[7] He JR, Carbonell J. Prior-Free rare category detection. In: Proc. of the SDM 2009. Sparks, 2009. 155−163. http://www.siam.org/proceedings/datamining/2009/dm09_015_hej.pdf

[8] Wang W, Zhou ZH. A new analysis of co-training. In: Proc. of the ICML 2010. Haifa, 2010. 1135−1142. http://www.icml2010.org/papers/275.pdf

[9] Deng C, Guo, MZ. Tri-Training and data editing based semi-supervised clustering algorithm. Journal of Software, 2008,19(3): 663−673 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/19/663.htm [doi: 10.3724/SP.J.1001.2008.00663]

[10] Hospedales TM, Gong SG, Xiang T. Finding rare classes: Adapting generative and discriminative models in active learning. In: Huang JZ, Cao L, Srivastava J, eds. Advances in Knowledge Discovery and Data Mining (PAKDD 2011). LNAI 6635, Heidelberg: Springer-Verlag, 2011. 296−308. [doi: 10.1007/978-3-642-20847-8_25]

[11] Jian P, Kapoor A. Active learning for large multi-class problems. In: Proc. of the CVPR 2009. Washington: IEEE Computer Society, 2009. 762−769. [doi: 10.1109/CVPR.2009.5206651]

[12] He JR, Tong HH, Carbonell J. Rare category characterization. In: Proc. of the ICDM 2010. Washington: IEEE Computer Society, 2010. 226−235. [doi: 10.1109/ICDM.2010.154]

[13] Pelleg D, Moore A. Active learning for anomaly and rare-category detection. In: Saul LK, Weiss Y, Bottou L, eds. Advance in Neural Information Processing Systems 17. Cambridge: MIT Press, 2005. 1073−1080. http://books.nips.cc/papers/files/nips17/NIPS2004_0438.pdf

[14] Vatturi P, Wong WK. Category detection using hierarchical mean shift. In: Proc. of the KDD 2009. New York: ACM Press, 2009. 847−856. [doi: 10.1145/1557019.1557112]

[15] Xia CY, Hsu W, Lee ML, Ooi BC. BORDER: Efficient computation of boundary points. IEEE Trans. on Knowledge and Data Engineering, 2006,18(3):289−303. [doi: 10.1109/TKDE.2006.38]

[16] Xue LX, Qiu BZ. Boundary points detection algorithm based on coefficient of variation. Pattern Recognition and Artificial Intelligence, 2009,22(5):799−802 (in Chinese with English abstract).

[17] Qiu BZ, Yue F, Shen JY. BRIM: An efficient boundary points detecting algorithm. In: Zhou ZH, Li H, Yang Q, eds. Advances in Knowledge Discovery and Data Mining (PAKDD 2007). LNAI 4426, Heidelberg: Springer-Verlag, 2007. 761−768. [doi: 10.1007/978-3-540-71701-0_83]

[18] Xue AR, Ju SG, He WH, Chen WH. Study on algorithms for local outlier detection. Chinese Journal of Computers, 2007,30(8): 1455−1463 (in Chinese with English abstract).

[19] Moor A. A tutorial on *kd*-trees. Technical Report, University of Cambridge Computer Laboratory, 1991. http://www.autonlab.org/autonweb/documents/papers/moore-tutorial.pdf

[20] Asuncion A, Newman D. UCI machine learning repository. Irvine: University of California, 2007. http://archive.ics.uci.edu/ml/datasets.html

[21] Leung Y, Zhang JS, Xu ZB. Clustering by scale-space filtering. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000, 22(12):1396−1410. [doi: 10.1109/34.895974]

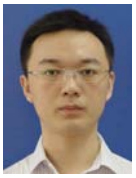
[22] Huang H, He QM, He JF, Ma LH. RADAR: Rare category detection via computation of boundary degree. In: Huang JZ, Cao L, Srivastava J, eds. Advances in Knowledge Discovery and Data Mining (PAKDD 2011). LNAI 6635, Heidelberg: Springer-Verlag, 2011. 258−269. [doi: 10.1007/978-3-642-20847-8_22]

附中文参考文献:

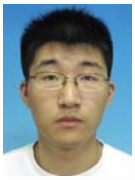
[9] 邓超,郭茂祖.基于 Tri-training 和数据剪辑的半监督聚类算法.软件学报,2008,19(3):663−673. http://www.jos.org.cn/1000-9825/19/663.htm [doi: 10.3724/SP.J.1001.2008.00663]

[16] 薛丽香,邱保志.基于变异系数的边界点检测算法.模式识别与人工智能,2009,22(5):799−802.

[18] 薛安荣,鞠时光,何伟华,陈伟鹤.局部离群点挖掘算法研究.计算机学报,2007,30(8):1455−1463.



黄浩(1986－),男,湖北潜江人,博士生,主要研究领域为机器学习,数据挖掘.



钱烽(1983－),男,博士生,主要研究领域为数据挖掘.

何钦铭(1965－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为机器学习,数据挖掘,虚拟化技术.

何江峰(1983－),男,硕士,主要研究领域为数据挖掘.

陈奇(1963－),男,副教授,主要研究领域为数据挖掘,GIS.

马连航(1983－),男,博士生,主要研究领域为数据挖掘.