文章编号:1007-5321(2017)04-0054-06 **DOI**:10.13190/j.jbupt.2017.04.009

基于 *k*-近邻域中心偏移的鲁棒性异常检测算法

赵建龙[1], 曲 桦[1,2], 赵季红[2,3]

(1. 西安交通大学 软件学院, 西安 710049; 2. 西安交通大学 电子与信息工程学院, 西安 710049;
3. 西安邮电大学 通信与信息工程学院, 西安 710061)

摘要: 针对大多数基于距离和密度的异常检测算法敏感于近邻参数 k 的问题,提出了一种鲁棒性异常检测标准——k-近邻域中心偏移异常因子(COOF). 数据结点的 k-近邻域中心位置会随着近邻参数 k 的变化而发生迁移,鉴于异常结点要比正常结点对 k-近邻域中心位置偏移量的影响更大,通过累加因递增 k 而产生的偏移量来表征数据结点的异常程度,并在 COOF 基础上实现了鲁棒性的异常检测算法. 通过综合数据和真实数据的实验仿真可知,COOF 不仅对近邻参数 k 具有鲁棒性,而且相比基于距离的 k 最近邻算法、基于局部距离的异常因子和基于密度的局部异常因子具有更稳定且更准确的异常检测性能.

关 键 词: 异常检测; k 最近邻; 局部异常因子; 中心偏移异常因子

中图分类号: TP393 **文献标志码**: A

Robust Outlier Detection Algorithm Based on *k*-Nearest Neighbor Region Center Migration

ZHAO Jian-long[1], QU Hua[1,2], ZHAO Ji-hong[2,3]

(1. School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China;
2. School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China;
3. School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710061, China)

Abstract: Considering the distance- and density-based outlier detection algorithms are often sensitive to a nearest neighbor parameter k, termed k-center offset outlier factor (COOF), a robust outlier detection criterion for the characterization of abnormal degree of each data object was proposed. Each data object is included in a region within its k nearest neighbors, and the center of region will migrate with the change of nearest neighbor parameter k. In general, the variation of center offset of k nearest neighbor region is greater for an outlier than a normal object. According to this observation, for each data object, COOF is defined as the accumulation of this kind of offset when increasing the nearest neighbor parameter from one to k. Finally, the outlier detection algorithm based on COOF was also presented. Through artificial data and real data experimental simulations show that COOF is insensitive to parameter k, and has more stable and accurate outlier detection performance compared to k nearest neighbor, local distance-based outlier factor and local outlier factor, which are the distance-based method and density-based method respectively.

Key words: outlier detection; k nearest neighbor; local outlier factor; center offset outlier factor

收稿日期: 2016-09-12
基金项目: 国家自然科学基金项目(61371087 和 61531013)
作者简介: 赵建龙(1992—), 男, 博士生, E-mail: z.jl199235@stu.xjtu.edu.cn; 曲 桦(1961—), 男, 教授, 博士生导师.

异常检测是数据挖掘领域中的一个重要分支，旨在从数据集中挖掘出那些不符合正常数据特征的少数结点. 通过挖掘少数但关键的隐藏特征信息，异常检测已被多种社会应用所采纳，包括信用卡欺诈检测、电子商务犯罪活动发现、图像处理、视频监控、天气预报、医药研究[1]等，以及基于软件定义网络的流量异常检测相关研究[2].

目前异常挖掘算法[3]主要包括 2 类：监督性和非监督性异常挖掘. 监督性异常挖掘算法需要大量样本进行模型测试，但实际应用中无法事先获取大量训练样本. 对于非监督性异常挖掘算法，主要划分为 2 类：基于距离和基于密度的异常检测算法. k 最近邻(KNN, k nearest neighbor)[4]是一种典型的基于距离的异常检测算法，利用数据结点的第 k-近邻距离作为异常程度高低的判别因子，然而这类方法往往受限于局部密度问题[5]. 基于密度的异常检测算法通过比较数据结点和其近邻结点的局部密度来评价其异常程度，代表算法是局部异常因子(LOF, local outlier factor)，虽然此类算法解决了局部密度问题，但是无法有效处理低密度模式问题[6]. 总结来说，这 2 类检测算法无法同时考虑数据结点的全局和局部特征，近邻参数 k 过小无法准确识别全局异常点；相反，k 过大无法准确识别局部异常点.

针对上述问题，笔者提出一种基于 k-近邻域中心偏移异常因子(COOF, center offset outlier factor)来表征数据结点的异常程度，进而实现鲁棒性的异常检测算法. 随着近邻参数 k 的变化，基于异常结点的 k-近邻域中心偏移程度要远大于正常结点的 k-近邻域中心偏移程度. 根据这一性质，COOF 通过累加因递增 k 而产生的 k-近邻域中心偏移量来表征数据结点的异常程度. 最后通过在综合数据和真实数据上进行实验仿真得出 COOF 不仅能解决近邻参数的敏感性问题并提供稳定的异常检测结果，而且在同时考虑检测准确度和受试者工作特征曲线的面积(AUC, Area Under Curve)[7] 2 种评价标准时，其性能均优于 KNN、基于局部距离的异常因子(LDOF, Local distance-based outlier factor)和 LOF 算法.

1　异常检测问题

对于绝大多数非监督性异常挖掘算法，异常因子的高低直接表达了数据结点异常程度的高低，一般来说异常因子较高的数据结点成为异常结点的可能性更大. 下面分别介绍典型的基于距离的异常检测算法 KNN[4]和 LDOF[8]，以及基于密度的异常检测算法 LOF[5].

定义 1　对象 p 的 k-近邻距离. 对于任意正整数 k，对象 p 的 k-近邻距离表示为 $d_k(p)$，定义为对象 p 和 o 之间的距离 $d(p,o)$，必须同时满足以下 2 个条件：

1)数据集 D 中至少存在 k 个对象 $o' \in D\backslash\{p\}$ 满足 $d(p,o') \leqslant d(p,o)$；

2)数据集 D 中至多存在 $k-1$ 个对象 $o' \in D\backslash\{p\}$ 满足 $d(p,o') < d(p,o)$.

其中 $\{p,o\} \in D$，距离度量方式可以是欧几里得距离、曼哈顿距离或其他任意一种相异性度量方式.

定义 2　$N_k(p)$ 表示对象 p 的 k-近邻域，表示为

$$N_k(p) = \{q \mid d(p,q) \leqslant d_k(p), q \neq p\} \tag{1}$$

其中 $d(p,q)$ 为数据集 D 中对象 p 和 q 间的直达距离.

Ramaswamy 等[4]提出的 KNN 作为基于距离的异常检测算法代表，利用数据结点和其第 k-近邻结点间的距离量化其异常程度. 给定一个数据结点 p 和其 k-近邻距离 $d_k(p)$，那么其异常因子 $f_{\mathrm{KNN}}(p)$ 表示为 p 和其第 k-近邻结点间的距离：

$$f_{\mathrm{KNN}}(p) = d_k(p) \tag{2}$$

Breunig 等[5]提出的 LOF 是一种典型的基于密度的异常检测算法，利用异常结点周围的密度往往低于其近邻的密度这一事实，计算出每个数据结点的局部异常因子，一般 LOF(p)越高表示数据结点 p 成为异常结点的可能性越大. 给定数据结点 p 和正整数 k，异常因子 $f_{\mathrm{LOF}}(p)$ 表示为

$$f_{\mathrm{LOF}}(p) = \frac{1}{|N_k(p)|} \sum_{q \in N_k(p)} \frac{l_k(q)}{l_k(p)} \tag{3}$$

其中：$l_k(p)$ 为数据结点 p 的直接可达密度，通过计算 p 和 $q(q \in N_k(p))$ 之间平均可达距离的倒数得到，$N_k(p)$ 表示结点 p 的 k-近邻域.

Zhang 等[8]提出的 LDOF 是另一种基于距离的局部异常检测因子，LDOF 利用数据结点与其近邻之间的相对距离来度量偏离其散列近邻的程度，偏离程度越大则表明是异常结点的可能性也更大. 给定数据结点 p 和正整数 k，异常因子 $f_{\mathrm{LDOF}}(p)$ 定义为

$$f_{\mathrm{LDOF}}(p) = \bar{d}/\bar{D} = \frac{\frac{1}{k}\sum_{q \in N_k(p)} d(p,q)}{\frac{1}{k(k-1)}\sum_{p,q \in N_k(p), p \neq q} d(p,q)} \tag{4}$$

其中 $d(p,q)$ 为数据结点 p 和 q 之间的距离.

1 异常检测模型

2.1 COOF 定义

立体几何中的四脚椎体是一种典型的四面体结构,因其具有重心低和稳定性强的特点作为防波堤结构应用于海岸工程,主要优势在于四脚椎体的重心低,而表面离重心点越近则越不会因外力驱动而发生畸形. 同理,在异常结点挖掘中,如果一个数据结点远离其邻域重心程度越大表明其是异常结点的可能性更大. 由于数据区域不存在质量分布问题,数据区域重心即为数据区域中心,计算见式(5).

假设存在 μ 个离散数据点 $(x_i,y_i)(i=1,2,\cdots,\mu)$,则其覆盖 μ 个数据结点的区域中心 $G(\tilde{x},\tilde{y})$ 表示为

$$\left.\begin{aligned}\tilde{x}&=\frac{\sum_{i=1}^{\mu}x_i}{\mu}\\ \tilde{y}&=\frac{\sum_{i=1}^{\mu}y_i}{\mu}\end{aligned}\right\} \tag{5}$$

这种方法最为简单直观,直接利用离散数据点的横纵坐标即可求得覆盖区域的数据中心. 给定数据结点 p,其 k-近邻区域为 $N_k(p)$,则 p 的 k-近邻区域 $N_k(p)$ 的区域中心表示为

$$c_k(p)=\frac{1}{k}\sum_{q\in N_k(p)}\boldsymbol{x}_q \tag{6}$$

其中 $\boldsymbol{x}_q=(x_{q1},x_{q2},\cdots,x_{qn})\in\mathbb{R}^n$,$n$ 为数据维度.

图1展示了 k-近邻域中心偏移范例,可看出通过递增 k,数据结点的 k-近邻域中心发生了迁移. 对于正常结点 p_2 来说,伴随着 k 的变化,其中心 $c_k(p_2)$ 的变化微小,其中 C_2 表示 p_2 结点在 $k=5$ 时的近邻域. 而对于异常结点 p_1 来说,伴随着 k 的递增,其中心 $c_k(p_1)$ 却发生了明显的迁移变化,图1中 c_1,c_2,c_3,c_4,c_5 标识了 k 从1变化至5时,p_1 结点的 k-近邻域中心位置变化过程,其中 C_1 表示异常点 p_1 在 $k=5$ 时的近邻域. 通过量化数据结点的 k-近邻域中心位置变化造成的距离偏移来表征数据结点的异常程度.

对于数据结点 p,伴随着 k 的自增,其 k-近邻域中心的迁移量表示为

$$\sigma_i(p)=d(c_i(p),c_{i+1}(p)),i=1,2,\cdots,k-1 \tag{7}$$

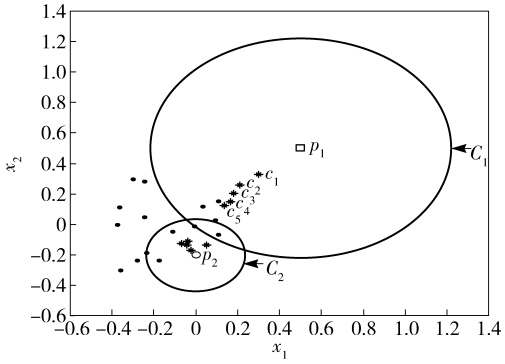


图1 k-近邻域中心偏移范例

为了强调表示 k 对数据结点的 k-近邻域中心位置的影响,采用中心迁移量的绝对误差来定义其 k-近邻中心位置变化的抖动程度. 最后,通过累加数据结点 k-近邻域中心位置迁移的波动程度定义出COOF:

$$f_{\mathrm{COOF}}(p)=\sum_{i=1}^{k-2}|\sigma_i(p)-\sigma_{i+1}(p)| \tag{8}$$

2.2 基于 COOF 的异常检测算法

本节详细介绍基于COOF的异常检测算法流程,如算法1所示. 首先计算每个数据结点的COOF,通过排序得到具有最高异常因子的异常结点. 为了充分考虑所有数据结点的全局和局部特征,按照LOF算法对近邻参数 k 的定义范围[5],COOF选择其上界作为算法输入.

算法1 基于COOF的异常检测算法

输入:数据集 D,近邻数 k,异常结点数 m

输出:具有最高COOF值的 m 个异常结点

for $p\in D$ **do**
  for $i=1$ **to** k **do**
    根据 p 的 k-近邻距离 $d_k(p)$ 定义 $N_k(p)$;
    计算 p 的 k-近邻域 $N_k(p)$ 的中心 $c_k(p)$:

$$c_k(p)=\frac{1}{k}\sum_{q\in N_k(p)}\boldsymbol{x}_q$$

    计算 $N_k(p)$ 的中心位置绝对偏移量 $\Delta\sigma_i(p)$:

$$\Delta\sigma_i(p)=|d(c_i(p),c_{i+1}(p))-d(c_{i+1}(p),c_{i+2}(p))|$$

  end for
  计算 $f_{\mathrm{COOF}}(p)=\sum_{i=1}^{k-2}\Delta\sigma_i(p)$;
  排序 $f'_{\mathrm{COOF}}\leftarrow\mathrm{Sort}(f_{\mathrm{COOF}},\mathrm{descending})$;
end for

10 **return** 具有最高 COOF 的 m 个异常结点.

3　仿真实验

3.1　评价指标

针对异常检测算法的性能评估,将采用准确度和受试者工作特征曲线(ROC, receiver operating characteristic curve)[7]来综合评估数据集的异常检测结果. 基于异常检测的准确度 E_{Acc} 定义为

$$E_{\mathrm{Acc}}=\frac{n}{m} \tag{9}$$

其中:m 为原始数据中的真实异常结点数, n 为检测到的真实异常结点数.

考虑到样本数据的不均衡分布问题,补充采用 ROC 来评价异常检测结果. 而 AUC 作为 ROC 的定量统计可以直观地表达出分类性能的高低,一般来说 AUC 值越高表示分类器性能越好. AUC[7] 的定义为

$$E_{\mathrm{AUC}}=\left(S_0-\frac{n_0(n_0+1)}{2}\right)\Big/ n_0n_1 \tag{10}$$

其中:n_0 和 n_1 分别为正常样本数和异常样本数; $S_0=\sum r_i, r_i$ 为在按样本所述类别概率进行排序后第 i 个正常样本的排序位置.

3.2　综合数据检测

本节将讨论 COOF 对比 KNN、LOF 和 LDOF 算法在综合数据中的异常检测性能. 图 2 所示的综合数据 D 同时包含了局部密度[5]和低密度模式[6]特征的数据集,一共包含 1 140 个数据结点,划分为 5 个类簇,分别是抛物线状和 4 个高斯分布类簇. 抛物线分布的数据结点数是 400,4 个高斯分布类簇满足自相关矩阵 $\boldsymbol{\Sigma}_t=t\boldsymbol{I}_{2\times2}$,其中 $\boldsymbol{I}_{2\times2}$ 为 2×2 单位矩阵,$t=0.05, 0.03, 0.02, 0.02$ 为数据分布的离散度. 这 4 个高斯类簇所包含的数据结点数分别是 200、

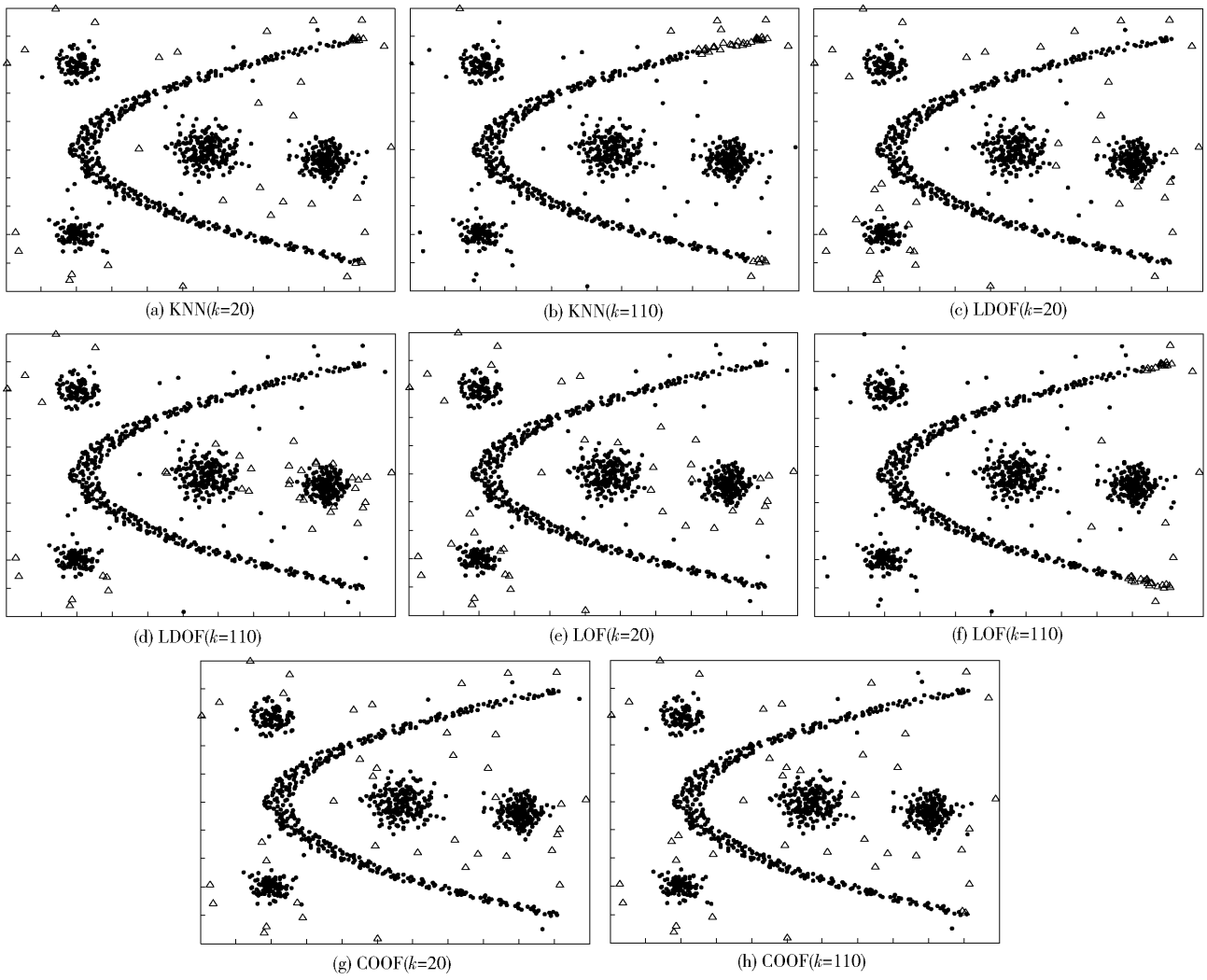


图 2　综合数据集异常检测结果

200、100、100，另外随机产生40个异常数据结点添加到D中. 基于该综合数据集D，通过改变近邻数k来对比KNN、LOF、LDOF和COOF在异常检测中的性能. 对于这4种方法，通过异常检测得到40个具有最高异常因子的异常结点. 从图2中可看出，当$k=20$时，这4种方法均表现出较好的异常检测结果；但当$k=110$时，KNN、LDOF和LOF出现较大的检测误差，而COOF却依然能保持较高精度的异常检测性能.

对于KNN和LDOF这类基于距离的异常检测算法，当k值递增时，倾向于检测“全局异常”结点，直观地解释为“全局异常”结点容易被KNN和LDOF检测到，但局部异常结点却容易被忽视. 而对于LOF算法，考虑图2(c)中左下角类簇中的数据结点，由于离其最近的类簇大致包含100个数据结点，当k增长时，其k-近邻结点逐渐考虑到密度更小的抛物线数据集. 因此，LOF不能准确发现高密度类簇周围的真实异常结点，却反而把低密度类簇中的结点误认为异常结点. 另外，图3和图4分别展示了k值在递增过程中异常检测指标的变化过程. 4种算法在k值较小时均表现出了高效的检测性能. 然而，随着k值的递增，KNN、LDOF和LOF检测性能下降明显，但是所提出的COOF却依然保持高检测性能.

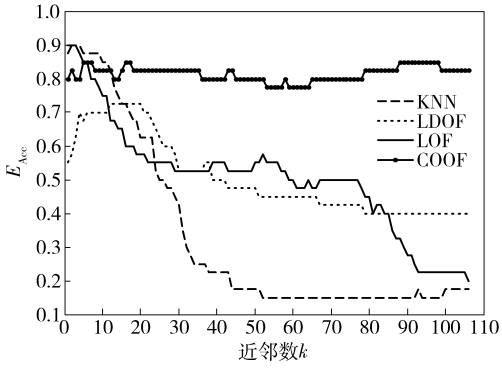


图3 准确度对比结果

3.3 真实数据检测

笔者同时将COOF应用到2种真实网络数据集Iris和Wine中，均来源于UCI机器学习库[9]. Iris数据集一共包含150个实例，每个实例包括4维属性，所有实例划分为3类，类标签分别是setosa、versicolour和virginica. Wine数据集一共包含178个实例，每个实例包括13维属性，所有实例也划分为3类(Class 1、Class 2和Class 3). 针对这2种数据集，实验中分别选择其中任意2个类簇的数据作为正类数据点，从剩下的一类中任意选择5个数据点作为异常结点.

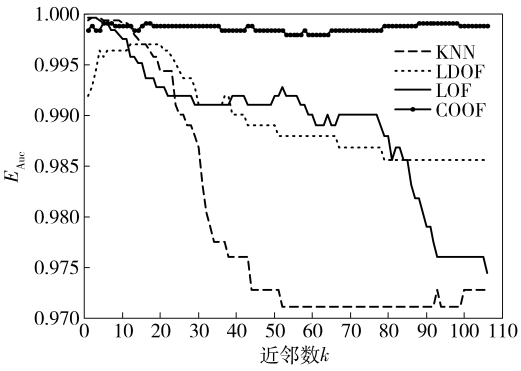
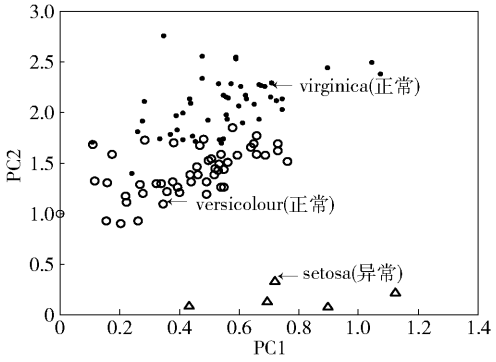
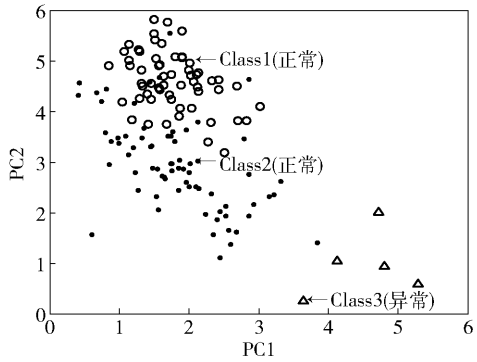


图4 AUC对比结果

由于这2种数据集均是高维数据集，为便于在二维坐标系中可视化展示异常检测结果，因此采用非负矩阵分解[10]算法对数据进行降维处理，从而提取能够反映原始数据主特征的二维属性，然后根据这二维属性构建二维可视化坐标系. 图5展示了经降维处理过的数据集应用COOF进行异常检测的结果，横纵坐标分别表示2个主成分(PC1和PC2). 其中上三角代表检测到的异常结点，而空心圆和实心点代表正常结点，可看出COOF同样可以很准确



(a) Iris数据集异常检测结果



(b) Wine数据集异常检测结果

图5 基于COOF的真实数据集异常检测结果

地检测出数据集中的真实异常结点.

在利用 KNN、LDOF、LOF 和 COOF 4 种算法进行真实数据集的异常检测时,首先计算每个结点的异常因子,然后将具有最高异常因子的 5 个数据结点标识为异常结点,具体检测结果如表 1 所示. 从表 1 中总结得出 LDOF 和 LOF 两种算法的检测性能随着 k 值变化发生了较大改变,而 COOF 算法却基本保持稳定且具有较高的准确度和 AUC. 由于 Wine 和 Iris 这 2 种数据集的数据分布特征较为简单,不存在密度差异较大的类簇,因此 KNN 算法也表现出了较高的检测性能. 但如图 2 所示在具有密度差异较大的数据集中,KNN 检测性能下降明显,而 COOF 依然表现出高检测性能.

表 1　检测性能

数据集	k	KNN		LDOF		LOF		COOF	
		Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
Wine	5	0.80	0.99	0	0.96	0.20	0.97	0.80	0.99
	10	1.00	1.00	0.20	0.97	0.80	0.99	0.80	0.99
	20	1.00	1.00	0.80	0.99	0.80	0.99	0.80	0.99
Iris	5	1.00	1.00	0	0	0.00	0.00	0.80	0.79
	10	1.00	1.00	0	0	0.80	0.79	1.00	1.00
	20	1.00	1.00	0.40	0.39	1.00	1.00	1.00	1.00

4　结束语

笔者提出了一种基于 k-近邻域“中心偏移的鲁棒性异常检测算法”,利用数据结点的 k-近邻区域中心位置偏移量来定义一种新的表征结点异常程度的异常因子,而具有较高 COOF 的结点代表异常结点的可能性更大. 基于 COOF 的异常检测算法不敏感于近邻参数 k 的选择,能够同时考虑数据结点的全局和局部特征. 通过综合数据和真实数据的实验仿真得出 COOF 不仅能实现模型参数 k 的鲁棒性选择,而且相比于 KNN、LDOF 和 LOF 检测算法具有更稳定且更准确的异常检测性能.

参考文献:

[1] Ha J, Seok S, Lee J S. A precise ranking method for outlier detection [J]. Information Sciences, 2015, 324: 88-107.

[2] 左青云, 陈鸣, 王秀磊, 等. 一种基于 SDN 的在线流量异常检测方法[J]. 西安电子科技大学学报, 2015, 42(1): 155-160.
Zuo Qingyun, Chen ming, Wang Xiulei, et al. Online traffic anomaly detection method for SDN [J]. Journal of Xidian University, 2015, 42(1): 155-160.

[3] 刘敬, 谷利泽, 钮心忻, 等. 基于单分类支持向量机和主动学习的网络异常检测研究[J]. 通信学报, 2015, 36(11): 136-146.
Liu Jing, Gu Lize, Niu Xinxin, et al. Research on network anomaly detection based on one-class SVM and active learning [J]. Journal on Communications, 2015, 36(11): 136-146.

[4] Ramaswamy S, Rastogi R, Shim K. Efficient algorithms for mining outliers from large data sets [J]. ACM SIGMOD Record, 2000, 29(2): 427-438.

[5] Breunig M M, Kriegel H P, Ng R T, et al. LOF: identifying density-based local outliers[J]. ACM SIGMOD record, 2000, 29(2): 93-104.

[6] Tang Jian, Chen Zhixiang, AW Fu, et al. Enhancing effectiveness of outlier detections for low density patterns [C]// Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining. Berlin: Springer-Verlag, 2002: 535-548.

[7] Huang Jin, Ling Charles X. Using AUC and accuracy in evaluating learning algorithms [J]. IEEE Transactions on Knowledge & Data Engineering, 2005, 17(3): 299-310.

[8] Zhang Ke, Hutter Marcus, Jin Huidong. A new local distance-based outlier detection approach for scattered real-world data[C]// Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining. Berlin: Springer-Verlag, 2009: 813-822.

[9] UCI. UCI Repository of Machine Learning Databases [EB/OL]. Irvine, CA: University of California(2007) [2007]. http://www.ics.uci.edu/~mlearn/MLRepository.html.

[10] Lee D D, Seung H S. Learning the parts of objects by non-negative matrix factorization [J]. Nature, 1999, 401(6755): 788.