•人工智能•

DOI:10.12454/j.jsuese.202201398

本刊网刊

基于改进K-means的局部离群点检测方法

周　玉[1], 夏　浩[1], 岳学震[1], 王培崇[2]

(1.华北水利水电大学 电气工程学院, 河南 郑州 450045; 2.河北地质大学 信息工程学院, 河北 石家庄 050031)

摘　要: 离群点检测任务是指检测与正常数据在特征属性上存在显著差异的异常数据。大多数基于聚类的离群点检测方法主要从全局角度对数据集中的离群点进行检测, 而对局部离群点的检测性能较弱。基于此, 本文通过引入快速搜索和发现密度峰值方法改进K-means聚类算法, 提出了一种名为KLOD(local outlier detection based on improved K-means and least-squares methods)的局部离群点检测方法, 以实现对局部离群点的精确检测。首先, 利用快速搜索和发现密度峰值方法计算数据点的局部密度和相对距离, 并将二者相乘得到γ值。其次, 将γ值降序排序, 利用肘部法则选择γ值最大的k个数据点作为K-means聚类算法的初始聚类中心。然后, 通过K-means聚类算法将数据集聚类成k个簇, 计算数据点在每个维度上的目标函数值并进行升序排列。接着, 确定数据点的每个维度的离散程度并选择适当的拟合函数和拟合点, 通过最小二乘法对升序排列的每个簇的每1维目标函数值进行函数拟合并求导, 以获取变化率。最后, 结合信息熵, 将每个数据点的每个维度目标函数值乘以相应的变化率进行加权, 得到最终的异常得分, 并将异常值得分较高的top-n个数据点视为离群点。通过人工数据集和UCI数据集, 对KLOD、LOF和KNN方法在准确度上进行仿真实验对比。结果表明KLOD方法相较于KNN和LOF方法具有更高的准确度。本文提出的KLOD方法能够有效改善K-means聚类算法的聚类效果, 并且在局部离群点检测方面具有较好的精度和性能。

关键词: 离群点检测; K均值聚类; 最小二乘法; 密度峰值; 目标函数值

中图分类号: TP301.6　　**文献标志码:** A　　**文章编号:** 2096-3246(2024)04-0066-12

离群点被定义为由各种特殊原因产生的数据点[1]。因不同于正常数据点而常被视作噪声点, 也被认为是具有研究价值的点, 且在数据集中占有少量的比重。离群点检测任务是通过分析数据的属性特征后找出这些点, 并分析其潜在的异常信息。离群点检测广泛应用于欺诈检测[2]、网络入侵[3]、环境卫生[4]、图像处理[5]、轨迹异常检测[6]等领域。离群点检测方式有四大类, 分别是基于统计、距离、密度和聚类[7−8]。通常在很难获得数据标签的情况下, 大多数的离群点检测都在无监督的环境下进行。

离群点检测最早出现在统计领域, 将低概率区域内的数据点识别为离群点。例如, 在对高斯分布$N(\mu,\sigma^2)$进行均值−方差检验时, 位于3个或3个以上标准差[9]处的点被认为是离群点。Goldstein等[10]提出了一种基于直方图的离群点检测方法(histogram-based outlier detection method, HBOS), 该方法简单易懂, 易于解释。HBOS的检测方法为数据点的每个维度构建一个直方图, 然后计算它们的密度并将其相加, 得出整体的异常得分。Li等[11]通过Copula函数估计每个给定数据点的尾部概率来检测异常值。虽然基于统计的离群点检测方法在易于理解和实现方面具有优势, 且不需要较多的背景知识或复杂的算法; 然而, 该方法本身依赖于一些假设, 例如, 数据点分布服从正态分布, 维度独立, 这使得该方法难以适应复杂数据集。

为了更加有效地检测离群点, Ramaswamy等[12]基于数据点与其邻域之间的距离提出了k近邻(k-nearest neighbor, KNN)离群点检测方法, 该方法使用

收稿日期:2022 – 12 – 30　　**修回日期:**2023 – 12 – 21　　**网络出版日期:**2024 – 07 – 04

基金项目:国家自然科学基金项目(U1504622; 31671580); 河南省高等学校青年骨干教师培养计划项目(2018GGJS079); 河北省高等学校科学技术研究项目(ZD2020344)

作者简介:周　玉(1979—), 男, 副教授, 博士. 研究方向: 机器学习; 智能计算; 智能控制与决策. E-mail: zhouyu_beijing@126.com

数据点与第k近邻居的距离作为异常值的度量，异常值越大表示数据点的离群程度越高；虽然KNN方法能够有效地检测全局离群点，但由于没有分析数据点与k近邻居间的关系，导致难以适应复杂的数据类型。针对此问题，Zhang等[13]利用k近邻方法，计算数据点的k近邻距离与k近邻数据点间的两两距离的比值，根据该比值的大小来判断数据点的离群程度；Yang等[14]提出均值偏移的离群点检测方法（mean-shift outlier detector, MOD），通过数据点与k邻域的平均值对数据集进行均值偏移处理，通过计算数据点3次均值偏移的总偏移距离得到离群得分；虽然文献[13–14]对全局离群点有较好的检测性能，但由于没有考虑局部特点，从而局部离群点的检测性能较弱。

为进一步对局部离群点进行有效检测，Breunig等[15]提出基于密度的局部异常值的离群点检测方法（local outlier factor, LOF），该方法通过局部可达距离得到数据点的局部密度，并计算每个数据点与其邻居的密度比，以获得异常值分数，数据点的异常值分数与离群程度呈正相关。局部离群点检测方法能够检测不同簇密度中的局部离群点，但LOF算法本身难以适应复杂的数据集。为了进一步提高LOF算法对复杂数据集的处理能力，Tang等[16]提出了基于连通性的离群因子方法（connectivity-based outlier factor, COF），将LOF的邻域计算方法改为增量计算，利用链距离提高了异常值检测性能。Latecki等[17]在LOF的基础上提出了基于局部密度因子的离群点检测方法（local density factor, LDF），LDF方法通过核密度估计来估算样本与k近邻域的局部密度，并通过计算样本的局部密度与其邻域密度的比值来确定样本的离群值。尽管文献[16–17]的方法能够有效地检测复杂数据集中的局部离群点，但这些方法的离群点检测精度受到超参数k的显著影响，且当局部离群点的数量增多时，其检测性能下降。

基于聚类的离群点检测方法，一方面，可以解决超参数对离群点检测精度的影响；例如，相比于其他聚类算法，K-means聚类算法[18]可以直接通过样本与聚类中心之间的距离大小来判断其离群点程度。另一方面，通过与传统离群点检测方法相结合的方式，进一步提高离群点检测精度。He等[19]考虑到数据点形成多个聚类的情况，先利用K均值聚类算法（K-means）进行聚类，并使用倍数β来确定聚类的规模；再将小规模的聚类被视为离群类，并计算数据点的局部离群因子。Al–Zoubi等[20]利用模糊C均值聚类方法（FCM）对离群点进行检测，通过剔除数据点后目标函数值变化量来判断数据点的离群程度。Barai等[21]使用K-means聚类算法对数据集进行聚类，根据数据点间的成对距离大小提出阈值T，并将数据点与聚类中心间的成对距离大于T的数据点视作离群点。Ahmed等[22]提出一种新的基于聚类的离群点检测方法，先根据样本与最近聚类中心间的距离计算其误差平方和SSE、总平方和SST，以对K-means算法进行改进；接着，通过样本与所有聚类中心间的距离来衡量样本的离群程度。上述基于聚类的离群点检测方法对全局离群点有较好的检测性能，但当数据集中存在不同密度的聚类时，对局部离群点的检测性能较弱。为了提高局部离群点的检测性能，Zhou等[23]先运用FCM聚类算法计算数据集的目标函数值；再根据数据点移除对目标函数值的影响，剪枝那些导致目标函数值变化较大的数据点，并将其加入离群候选集；随后，采用LOF方法计算离群候选集中数据点的局部离群因子，并最终选择异常值分数最高的n个数据点作为离群点；虽然该方法能够有效检测局部离群点，但在不同簇之间的数据点分布不均匀的情况下，其剪枝效果较差且离群点检测精度受近邻参数k的影响较大。

针对上述问题，本文基于改进K-means聚类算法、最小二乘法和信息熵的方法，提出一种改进K-means的局部离群点检测方法（KLOD）。首先，使用快速搜索和发现密度峰值方法、肘部法则确定K-means聚类算法的初始聚类中心。接着，通过K-means聚类算法计算每个簇的数据点的每1维目标函数值，并从小到大排序，使用最小二乘法的函数拟合，并求导获得变化率；最后，结合信息熵将数据点的每1维目标函数值乘以对应的变化率进行加权，获得最终的异常得分。选取异常得分高的top-n个数据点作为离群点。一方面，KLOD方法能够确保初始聚类中心选取的稳定性且不受异常值的影响；另一方面，通过对每个簇单独分析，可以解决不同簇之间的数据点分布密度不均匀的问题。并且，最小二乘法能够突出数据点在各个维度上的异常信息，从而提高局部离群点检测的准确性。

1　预备知识

1.1　K-means聚类算法

K-means聚类算法[18]为传统的聚类方法，用于将数据集中的样本分成不同的组别，使得组内的样本具有高相似性，而组间相似度较低。其对凸形簇的聚类效果好，收敛速度更快。假定数据集$X=\{x_1,x_2,\cdots,x_n\}$，聚类个数为K，初始聚类中心为$C=\{c_1,c_2,\cdots,c_k\}$。以下是K-means聚类算法的具体步骤。

Step1：计算每个数据点x_i（$i=1,2,\cdots,n$）与初始

聚类中心c_l($l=1,2,\cdots,k$)间的距离$d(x_i,c_l)$，如下：

$$d(x_i,c_l)=\sqrt{\sum_{j=1}^{m}(x_{ij}-c_{lj})^2} \tag{1}$$

式中，x_{ij}为第i个数据点的第j维度，c_{lj}为第l个聚类中心的第j维度，m为数据点的维度个数。

Step2：根据数据点与聚类中心之间的距离，找出最小的距离$d_{\min}$并把该数据点归于相对应的簇，$d_{\min}$计算式如下：

$$d_{\min}=\min\{d(x_i,c_1),d(x_i,c_2),\cdots,d(x_i,c_l)\} \tag{2}$$

Step3：在完成数据集的聚类后，根据式(3)对聚类中心进行更新，并利用式(4)计算整个数据集的目标函数值E。

$$c_l'=\frac{\sum_{x_i\in c_l}x_i}{|c_l|} \tag{3}$$

$$E=\sum_{i=1}^{n}\sum_{x_i\in c_l}d^2(x_i,c_l) \tag{4}$$

式(3)中，$|c_l|$为第l簇的数据点个数。

Step4：重复Step1～3，直到聚类中心和目标函数值没有明显变化，达到收敛后结束算法。

K-means聚类算法本身有着硬聚类的特点，即它在对数据集聚类后，每个数据点间存在着非此即彼的关系，又因受离群点的影响较大，故可用于对离群点的检测。其初始聚类中心的选择及聚类个数的确定至关重要，直接影响聚类效果。

1.2 快速搜索和发现密度峰值方法

快速搜索和发现密度峰值聚类算法(clustering by fast search and find of density peaks，DPC)是Rodriguez与Laio在2014年提出的一种聚类算法[24]。关于其聚类中心的解释是：聚类中心密度最大且被密度小于它的数据点所包围，不同聚类中心间的距离较远。

定义1(DPC算法的局部密度)[24]

$$\rho_i=\sum_i\chi(d_{ij}-d_c) \tag{5}$$

式中：令$x=d_{ij}-d_c$，当$x<0$时$\chi(x)=1$，否则$\chi(x)=0$；d_{ij}为数据点i与j之间的欧式距离；d_c为截断距离，当d_c为超参数时，其通常被定义为$d_c=p\%$，表示所有数据点的平均邻居数为数据总量的$p\%$。避免数据点局部密度出现相同的情况，采用式(6)的高斯核函数来代替式(5)。

$$\rho_i=\sum_{i\neq j}\mathrm{e}^{-\left(\frac{d_{ij}}{d_c}\right)^2} \tag{6}$$

式中，ρ_i为第i个数据点的局部密度。

定义2(DPC算法的相对距离)[24]

$$\delta_i=\min_{j:\rho_j>\rho_i}d_{ij} \tag{7}$$

式中，ρ_j为第j个数据点的局部密度，$j\neq i$。通过式(7)计算数据点与密度大于它的最近数据点之间的距离。当数据点x_i的密度为数据集中最大时，式(7)不成立，此时，其相对距离为距离x_i最远点的距离，即$\delta_i=\max\limits_j d_{ij}$。通过构建局部密度−相对距离形成决策图，用于选取初始聚类中心。如图1(a)所示，数据集包含两个簇及对应的聚类中心。利用DPC方法构建局部密度和相对距离决策图，如图1(b)所示。由图1(b)可知，两个聚类中心位置均在右上方，因此可选择决策图右上方的数据点作为K-means聚类算法的初始聚类中心。

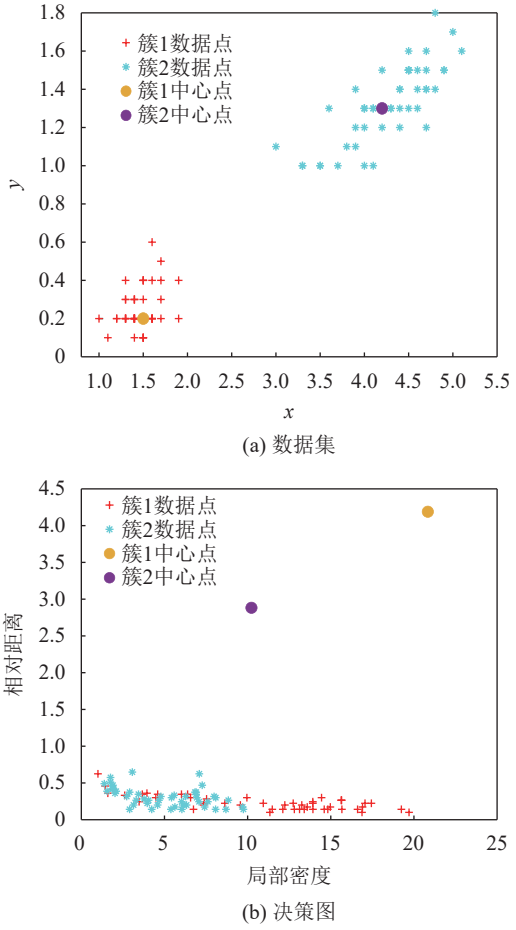


图 1 初始聚类中心形成过程举例

Fig. 1 Example of initial cluster center formation process

然而，快速搜索和发现密度峰值聚类算法存在一定的主观性。当人为设定的参数不准确或者数据集聚类个数较多且较为复杂时，通过决策图难以快速准确地选取初始聚类中心。因此，将决策图中数据点的局部密度和相对距离的值相乘，得到γ：

$$\gamma=\rho\times\delta \tag{8}$$

将γ值从大到小进行排序, 选取γ值最大的k个数据点作为K-means聚类算法的初始聚类中心, 而k值可以通过肘部法则进行准确选取。

1.3 肘部法则

在使用K-means聚类算法时, 需通过肘部法则[23,25]确定未知数据集的最佳聚类个数, 以达到最佳聚类效果。其原理是: 使用聚类算法聚类成不同数目的簇时, 代价函数值会有相应的变化。以聚类个数为自变量, 代价函数值为因变量, 记录代价函数值随聚类个数的变化并绘制出折线图, 当代价函数值随着聚类中心个数的增加无明显减小时, 观察其“肘部”位置(折线图的拐点), 确定最佳聚类个数。代价函数值计算方法如下:

$$P_{\text{cost}}(k)=\sum_{l=1}^{k}\sqrt{\sum_{i=1}^{N(i)}d^2(x_i^l-C_l)} \tag{9}$$

式中, x_i^l为属于l簇的第i个数据, k为簇的数目, $N(i)$为属于第l簇的数据点个数, C_l为第l簇的聚类中心。

1.4 最小二乘法

最小二乘法[26–27]目的是把一组数据点拟合成相应的曲线。其定义如下:

定义3(最小二乘法)[26–27]　用于拟合数学模型和估计参数, 目标是通过最小化观测值与预测值之间的误差平方和来找到最优参数值, 其中, 误差是观测值与预测值之间的差异。

已知有n个数据点(x_i,y_i), 满足多项式函数$f(x)=a_0+a_1x+a_2x^2+\cdots+a_kx^k$。利用误差平方和的原理得到关于$a_0,a_1,\cdots,a_k$的损失函数$L_{\text{loss}}$, 如下:

$$L_{\text{loss}}=\sum_{i=1}^{n}[y_i-(a_0+a_1x_i+a_2x_i^2+\cdots+a_kx_i^k)]^2 \tag{10}$$

为获得最佳的拟合曲线, 需得到最小的损失函数L_{loss}值, 即对损失函数L_{loss}中的参数求偏导, 令偏导后的多项式值为0。最后关于式(10)中各个变量对应的矩阵或向量$\boldsymbol{X}$、$\boldsymbol{Y}$和$\boldsymbol{A}$的结果如下:

$$\boldsymbol{X}=\begin{bmatrix} n & \sum_{i=1}^{n}x_i & \sum_{i=1}^{n}x_i^2 & \cdots & \sum_{i=1}^{n}x_i^k \\ \sum_{i=1}^{n}x_i & \sum_{i=1}^{n}x_i^2 & \sum_{i=1}^{n}x_i^3 & \cdots & \sum_{i=1}^{n}x_i^{k+1} \\ \sum_{i=1}^{n}x_i^2 & \sum_{i=1}^{n}x_i^3 & \sum_{i=1}^{n}x_i^4 & \cdots & \sum_{i=1}^{n}x_i^{k+2} \\ \vdots & \vdots & \vdots & & \vdots \\ \sum_{i=1}^{n}x_i^k & \sum_{i=1}^{n}x_i^{k+1} & \sum_{i=1}^{n}x_i^{k+2} & \cdots & \sum_{i=1}^{n}x_i^{2k} \end{bmatrix},$$

$$\boldsymbol{Y}=\left(\sum_{i=1}^{n}y_i,\sum_{i=1}^{n}x_iy_i,\cdots,\sum_{i=1}^{n}x_i^ky_i\right),$$

$$\boldsymbol{A}=(a_0,a_1,\cdots,a_k)。$$

本文使用高斯消元法求解参数$a_0,a_1,\cdots,a_k$, 以k=3为例, 求解结果如下:

$$a=\frac{(x_2-x_1)(y_3'-y_1')+(x_3-x_1)(y_1'-y_2')}{(x_2-x_1)(x_3-x_1)(x_3-x_2)},$$

$$b=\frac{(x_3^2-x_1{}^2)(y_2'-y_1')+(x_1^2-x_2^2)(y_3'-y_1')}{(x_2-x_1)(x_3-x_1)(x_3-x_2)},$$

$$c=y_1'-\frac{(x_1x_3^2-x_1^2x_3)(y_2'-y_1')+(x_1^2x_2-x_1x_2^2)(y_3'-y_1')}{(x_2-x_1)(x_3-x_1)(x_3-x_2)}。$$

1.5 信息熵

在数据集中每1维的离散程度不同, 所包含信息也不一样, 那么, 应该对数据集的每个维度赋予不同的权值。信息熵[28]可用来表达数据集的离散程度。

定义4(信息熵)[28]　用于表示一个随机变量的不确定性或信息量的多少。数据维度的离散程度越高, 包含的信息越多, 熵值越大。数据集每个维度的权值的计算步骤如下:

Step1: 对数据集X进行Z-score标准化[29]得到X', 计算数据集X'中第i个数据的第j维属性的比重P_{ij}, 如下:

$$P_{ij}=\frac{x_{ij}'}{\sum_{i=1}^{n}x_{ij}'} \tag{11}$$

式中, x_{ij}'为数据集X'中第i个数据的第j维属性。

Step2: 计算数据集X'中第j维的信息熵H_j, 如下:

$$H_j=-\rho\sum_{i=1}^{n}P_{ij}\ln P_{ij} \tag{12}$$

式中, $\rho=1/\ln n$, n为数据集的数据点个数。

Step3: 计算第j属性的权值, 如下:

$$\omega_j=\frac{H_j}{\sum_{j=1}^{m}H_j} \tag{13}$$

式中, m为数据集X'的维度个数。由此, 可得到权重集合$W=\{\omega_1,\omega_2,\cdots,\omega_m\}$。

2 算　法

本文提出一种基于改进K-means的局部离群点检测算法, 算法步骤如下: 首先, 利用快速搜索和发现密度峰值方法及肘部法则获取K-means聚类算法的初始聚类中心。接着, 在K-means算法对数据集聚

类之后，对目标函数值进行分析，并计算每个簇中数据点在每1维上的目标函数值。最后，先将每个簇中数据点的每1维目标函数值从小到大进行排序，并进行最小二乘法函数拟合及求导以获取斜率，再结合信息熵和斜率对数据点的每1维目标函数值进行加权，从而得到最终的异常得分。

2.1 初始聚类中心的选取

根据第1.1节可知，传统的K-means聚类算法通常采用随机选取的方式确定初始聚类中心，容易出现聚类结果不稳定情况。

为了改进K-means聚类算法的初始聚类中心的选取方式，若仅使用第1.2节的快速搜索和发现密度峰值方法计算数据点的局部密度和相对距离，再通过构建局部密度–相对距离决策图的方式选取聚类中心，有较大的主观性，并且，由于难以确定数据集中截断距离d_c的值，而不同的d_c会导致决策图中的数据集分布不同，从而难以较准确地选取聚类中心。因此，本文改进的K-means聚类算法的最佳初始聚类中心个数方法如下：首先，根据第1.2节的快速搜索和发现密度峰值方法计算数据点的局部密度和相对距离，并将两者相乘得到γ值；接着，将γ值进行降序排列，并使用第1.3节的肘部法则绘制关于聚类个数和代价函数值的折线图；最后，通过折线图中的肘部（曲线的拐点）位置选取准确的聚类中心个数。

2.2 基于K-means的目标函数分析

由于离群点在某些维度上的数值发生了异常，则离群点比正常数据点会更加远离聚类中心，因此本文将这些异常的维度称为离群维度。在剔除数据点后，计算得到相应的目标函数变化量。显然，如果剔除的是离群点，其目标函数变化量比剔除正常点的目标函数变化量会更加明显。基于K-means算法分析数据点对目标函数值影响的过程如下。

步骤1：根据式（4）可知，数据集的目标函数是根据数据点与对应聚类中心之间的距离计算的。当剔除每一个数据点时，整个数据集的目标函数值会减小，该数据点的目标函数值即为目标函数值的变化量ΔE，如下：

$$\Delta E=\sum_{i=1}^{n}\sum_{x_i\in c_l}d^2(x_i,c_l)-\sum_{i=1}^{n-1}\sum_{x_i\in c_l}d^2(x_i,c_l)=\sum_{x_i\in c_l}d^2(x_i,c_l) \tag{14}$$

步骤2：根据式（14）可知，离群点的目标函数值$\Delta E_{\text{outlier}}$、正常数据点的目标函数值$\Delta E_{\text{normal}}$的计算式分别为：

$$\Delta E_{x_{\text{outlier}}}=\sum_{x_{\text{outlier}}\in c_l}d^2(x_{\text{outlier}},c_l) \tag{15}$$

$$\Delta E_{x_{\text{normal}}}=\sum_{x_{\text{normal}}\in c_l}d^2(x_{\text{normal}},c_l) \tag{16}$$

步骤3：由于离群点比正常点更远离聚类中心，式（17）表示离群点与聚类中心之间的距离大于正常点与聚类中心间的距离。

$$d_{x_{\text{outlier}}}(x_i^l,c_l)>d_{x_{\text{normal}}}(x_i^l,c_l) \tag{17}$$

根据式（15）～（17）可知，式（18）表示离群点的目标函数值大于正常点的目标函数值。

$$\Delta E_{x_{\text{outlier}}}>\Delta E_{x_{\text{normal}}} \tag{18}$$

根据步骤1～3可知：当数据集中每个簇的数据点分布局部密度相差较大时，较大的ΔE值往往会出现在局部密度较小或数据规模较大的簇中，这样会使ΔE的值在整个数据集中进行不同密度簇间的比较显得没有意义。此时，需要对每个簇进行单独分析，这样可以更大程度上避免簇密度不均等的情况。

通过目标函数值的大小可以初步判断数据点是否为离群点。若离群点的离群维度只有1维，或者少数的几维，那么，离群点的目标函数值可能不明显，即式（18）不一定成立。为了更加精确找出离群点，需要对数据点的每1维进行单独分析，计算数据点的每1维的目标函数值，计算式如下：

$$\Delta E_{ij}^l=\sum_{x_i\in c_l}d^2(x_{ij}^l,c_{lj}) \tag{19}$$

式中，ΔE_{ij}^l为属于第l簇的x_i在第j维的目标函数值，x_{ij}^l为第l簇的第i的数据点的第j维，c_{lj}为第l簇的第j维度。通过单独分析数据点的每1维特征，并对离群维度的目标函数值赋予更高的权值来突出离群信息。

2.3 异常得分的计算

2.3.1 每1维目标函数值的最小二乘拟合

首先，计算每个簇的所有数据点在每1维的目标函数值ΔE_{ij}，并将其按照从小到大排序为$\Delta E'_{ij}$。目标函数值小的数据点靠近聚类中心，而目标函数值大的数据点远离聚类中心。其次，将数据点的目标函数值从小到大排序后分为离群维度、非离群维度分析拟合点和拟合函数。在离群维度上，必然会有突增的部分。为了能突出离群点所在的离群维度的信息，需要得到从小到大排序后每1维目标函数值对应的变化率，根据第1.4节的最小二乘法，对按从小到大的顺序排列的每1维目标函数值进行拟合，形成$N(l)$个（$N(l)$为第l簇的数据点个数）以数据点的数量大小为横轴、以对应的目标函数值为纵轴的拟合点。当采用所有拟合点进行拟合时，拟合函数为高阶多项式，且不同簇的不同维度拟合函数各不相同。本文在离群维度上的拟合方式是放大目标函数值从小到大排序

后发生陡增的部分，缩小目标函数值变化不明显的部分。选取最能代表从小到大排序后目标函数值变化的3个拟合点，分别是：距离聚类中心最近的点、目标函数值发生突变前的点，以及目标函数值变化最大的点。由这3个点能确定一条一元二次曲线，为实现上述拟合方式，在离群维度上，采用指数函数作为拟合曲线，最终的拟合曲线函数为$f(x)=\mathrm{e}^{ax^2+bx+c}$。

在非离群维度上，采用一元二次函数$h(x)=ax^2+bx+c$对从小到大排序后目标函数值进行拟合，求导后的函数斜率为一元一次函数$h'(x)=2ax+b$，满足离聚类中心距离越远，斜率越大，则越有可能是离群点的特点。此情况下，拟合曲线的3个点分别为：目标函数值从小到大排序后的第1个点、中间点、最后一个点。离群维度和非离群维度不同拟合函数下，3个点的拟合位置如表1所示。

表 1　3个拟合点的位置

Tab. 1　Location of three fitting points

是否为离群维度	拟合方程	位置		
		第1个拟合点x_1	中间拟合点x_2	最后一个拟合点x_3
是	$f(x)=\mathrm{e}^{ax^2+bx+c}$	$\min(\Delta E'_{ij})$	$\max(\Delta E'_{i+1,j}-\Delta E'_{ij})$	$\max(\Delta E'_{ij})$
否	$h(x)=ax^2+bx+c$	$\min(\Delta E'_{ij})$	$\dfrac{\min(\Delta E'_{ij})+\max(\Delta E'_{ij})}{2}$	$\max(\Delta E'_{ij})$

最后，在确定3个拟合点位置之后，通过定义离散值来确定拟合$\Delta E'_{ij}$所需要的拟合方程。

定义5（离散值）　用$\Delta E'^{l}_{ij}$的后四分之一值之和与前四分之一值之和的比值T^l_j来表示第l簇的第j维离散值，计算式如下：

$$T^l_j=\frac{\sum_{i=3N(l)/4}^{N(l)}\Delta E'^{l}_{ij}}{\sum_{i=1}^{N(l)/4}\Delta E'^{l}_{ij}} \tag{20}$$

当$T^l_j<\sum_{j=1}^{m}T^l_j/2m$时，使用$h(x)=ax^2+bx+c$拟合；反之，使用$f(x)=\mathrm{e}^{ax^2+bx+c}$拟合。将得到的拟合函数$f(x)$或$h(x)$进行求导，获得$x^l_{ij}$所对应的斜率$k^l_{ij}$，分别为：

$$k^l_{ij}=(2ax^l_{ij}+b)\mathrm{e}^{a{x^l_{ij}}^2+bx^l_{ij}+c} \tag{21}$$

$$k^l_{ij}=2ax^l_{ij}+b \tag{22}$$

2.3.2　加权获取最终异常得分

首先，根据式（21）、（22）获取数据点x^l_{ij}的每1维斜率。接着，通过第1.5节的信息熵方法计算数据集的每1维权重。最后，将数据点x^l_{ij}的每1维目标函数值与对应的斜率和权重进行先相乘再相加，得到最终的异常得分：

$$S_{\mathrm{score}^l_i}=\sum_{j=1}^{m}\omega_j k^l_{ij}\Delta E'^{l}_{ij} \tag{23}$$

2.4　具体算法实现

算法1为改进K-means聚类算法，用快速搜索和发现密度峰值方法、肘部法则确定K-means聚类算法的初始聚类中心。算法2为基于改进K-means的局部离群点检测方法（KLOD），其中，离群点检测部分为：先对每个簇的数据点的每1维度目标函数值进行从小到大进行排序，并用最小二乘法进行拟合求导，再结合求导获得的斜率和信息熵计算数据点的异常得分。

算法1　改进K-means聚类算法

输入：数据集X；

输出：初始聚类中心$\{c_1,c_2,\cdots,c_k\}$。

Step1：根据式（6）计算局部密度ρ；

Step2：根据式（7）计算相对距离σ；

Step3：根据式（8）计算γ值，并将γ值进行从大到小排列；

Step4：根据式（9）计算代价函数值；

Step5：绘制关于聚类个数k和对应目标函数值的折线图，确定拐点对应的聚类个数k；

Step6：将γ值最大的k个数据点作为初始聚类中心。

算法2　基于改进K-means的局部离群点检测方法（KLOD）

输入：数据集X、初始聚类中心$\{c_1,c_2,\cdots,c_k\}$；

输出：异常得分$S_{\mathrm{score}^l_i}$。

Step1：使用算法1改进的K-means算法对数据集进行聚类，并获取稳定聚类中心$\{c'_1,c'_2,\cdots,c'_k\}$、对应的簇$Z_1,Z_2,\cdots,Z_k$；

Step2：根据式（19）计算属于第l簇的样本点x_i在第j维的目标函数值；

Step3：将ΔE^l_{ij}从小到大排序得到$\Delta E'^{l}_{ij}$，其中，$i=1,2,\cdots,N(l)$；

Step4：根据式（20）计算每个簇数据点的每1维离散值；

Step5：用拟合函数拟合$\Delta E'^{l}_{ij}$，并根据式（21）、（22）求解斜率k^l_{ij}；

Step6：根据式（11）～（13）计算每1维权重$W=\{\omega_1,\omega_2,\cdots,\omega_m\}$；

Step7：根据式（23）计算数据点的最终异常得分。

算法1对传统K-means聚类算法的初始聚类中心的选取方法进行改进，相比传统K-means算法，本文提出的改进K-means聚类算法能够选取到更加稳定且准确的聚类中心，从而达到更高的聚类精度。算法2是一种全新的离群点检测方法，通过最小二乘法和

信息熵来单独分析数据点的每个维度，旨在提高局部离群点检测的精度。

2.5 复杂度分析

在本文提出的离群点检测方法（KLOD）中，通过改进K-means聚类算法计算数据点的目标函数值的时间复杂度是$O(N^2)$，用最小二乘法拟合目标函数值的时间复杂度是$O(km)$，计算数据点的异常得分的时间复杂度为$O(N)$。其中，k为聚类个数，m为数据维度。因此，局部离群点检测方法（KLOD）的时间复杂度分为$O(N^2)+O(N)+O(km)=O(N^2)$。局部离群点检测方法（KLOD）与传统的离群点检测方法KNN和LOF拥有同样的时间复杂度，但局部离群点检测方法（KLOD）进一步提高了局部离群点检测精度。

3 实验与讨论

本文使用了人工数据集和真实数据集，其中：人工数据集包括2个簇和8个局部离群点，人工数据集维度为2，如图2所示；真实数据集选择了12个UCI数据集，见表2。

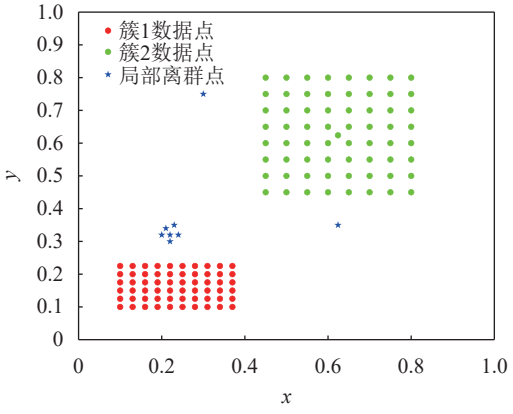


图 2 人工数据集

Fig. 2 Artificial dataset

表 2 12个UCI数据集

Tab. 2 Twelve UCI datasets

数据集	数据量	维度个数	分类个数
Iris	150	4	3
Wine	178	13	3
Seeds	210	7	3
Thyroid	215	5	3
Wbc	378	30	2
Aggregation	788	2	7
Vowel	871	3	6
Jain	373	2	2
Wdbc	569	30	2
Yeast	1844	8	10
Page Blocks	5472	10	5
Pen Digits	10992	16	10

为了评估离群点检测的性能，采用准确度（precision，Pr）和误检率（noise factor，Nf）两个指标[30–31]，分别记为F_{Pr}、F_{Nf}。准确度和误检率的计算式如下：

$$F_{\mathrm{Pr}}=\frac{F_{\mathrm{TP}}}{F_{\mathrm{TP}}+F_{\mathrm{FP}}} \tag{24}$$

$$F_{\mathrm{Nf}}=\frac{F_{\mathrm{FP}}}{F_{\mathrm{TP}}+F_{\mathrm{FP}}} \tag{25}$$

式（24）、（25）中：F_{TP}为真阳性（true positive，TP），表示算法检测到真实的离群点数量；F_{FP}为假阳性（false positive，FP），表示算法把正常数据错分成离群点的数量；F_{Pr}和F_{Nf}的最小取值和最大取值分别为0和1。F_{Pr}值越大，F_{Nf}值越小，说明离群点的检测性能越好。

采用CH指数、Dunn指数、I指数、S指数[32–33]这4个指标来评估本文方法在剔除离群点前后的人工数据集和UCI数据集上的聚类性能。

实验环境为：AMDR7 3.2 GHz CPU、8.00 GB内存，Windows11操作系统，采用MATLABR2020a编写实验中的算法。

3.1 在人工数据集上的结果与讨论

为了直观展示离群点检测方法（KLOD）的全流程，在人工数据集上进行下面的分析。首先，根据式（6）和（7）计算人工数据集中数据点的局部密度和相对距离，并利用式（8）计算数据点的γ值。将γ值从大到小排序，如图3所示。

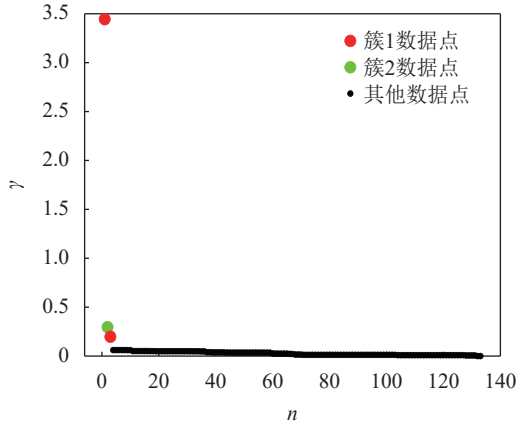


图 3 γ值降序排列

Fig. 3 Descending order of γ values

接着，使用肘部法则绘制关于聚类中心个数与对应代价函数值的折线图，如图4所示。

从图4可以看出，在k=2时，折线出现了明显的拐点，因此改进的K-means聚类算法的初始聚类中心个数为2。

接下来，选取γ值最大的前两个数据点作为改进的K-means聚类算法的初始聚类中心，并对数据集进

行聚类, 聚类结果如图5所示, 由图5可知, 6个分布密集的局部离群点被聚类到簇1, 两个局部离群点被聚类到簇2。

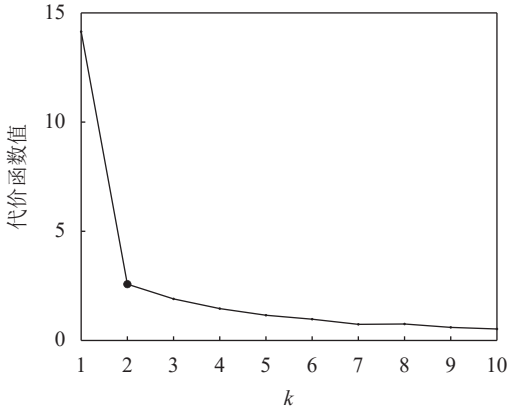


图 4　肘部法则

Fig. 4　Elbow rule

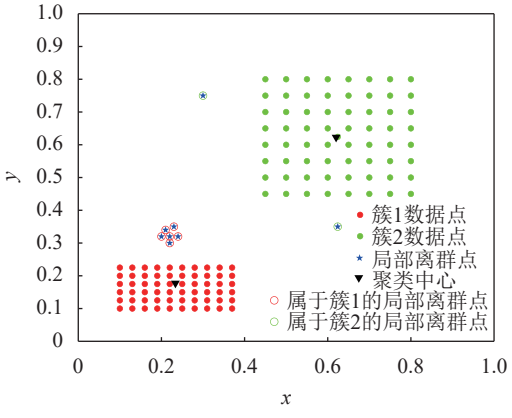


图 5　K-means聚类结果

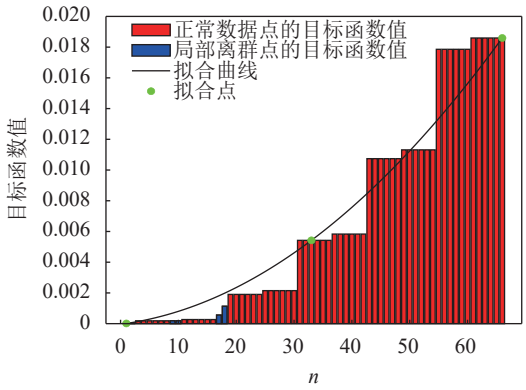
Fig. 5　K-means clustering results

然后, 计算每个簇数据点的每1维目标函数值, 并对其从小到大排序; 根据式(20)计算数据集的每1维的离散值; 根据表1选取的3个拟合点及对应的拟合函数进行拟合, 拟合结果如图6所示。

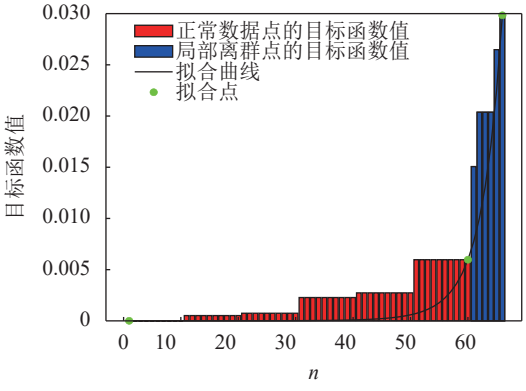
最后, 对拟合函数$f(x)$或$h(x)$进行求导, 获得数据点每1维目标函数值对应的斜率, 乘以该数据点的每1维目标函数值, 并进行加权, 获得最终的异常得分。其中, 由信息熵计算的权重$W=\{0.497\,6, 0.502\,4\}$。图7展示了每个簇数据点的异常得分, 从图7可看出, 局部离群点的异常得分显著高于正常数据点的异常得分。

为了详细验证本文提出的KLOD方法离群点检测性能, 将其与KNN[14]和LOF[17]方法在人工数据集上进行离群点检测对比实验, 表3为上述3个离群点检测方法的真阳性、假阳性、准确度和误检率。

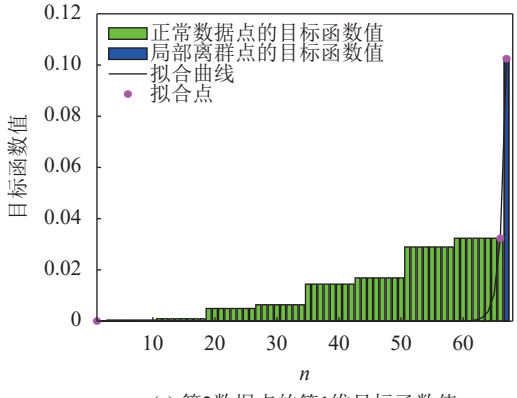
从表3的准确度可知, KLOD、KNN及LOF方法均能检测出簇2中的两个局部离群点。然而, 由图5可知, LOF算法难以检测出簇1中的6个局部离群点, 因为簇1



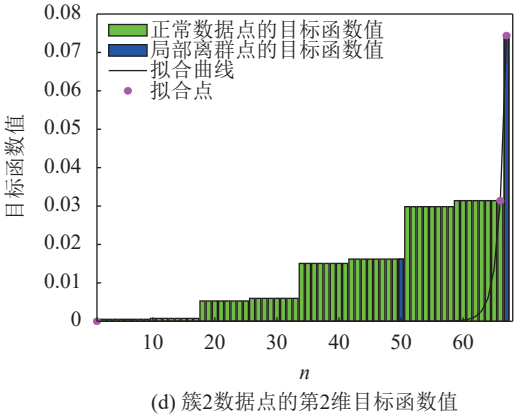
(a) 簇1数据点的第1维目标函数值



(b) 簇1数据点的第2维目标函数值



(c) 簇2数据点的第1维目标函数值

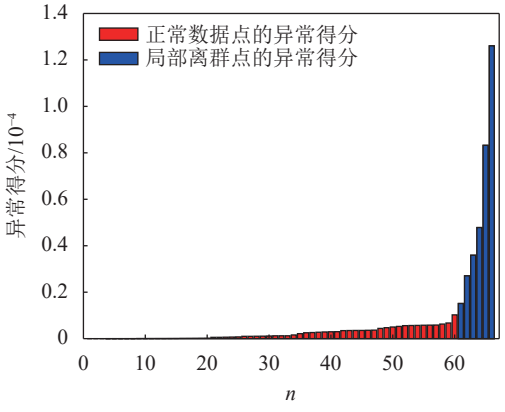


(d) 簇2数据点的第2维目标函数值

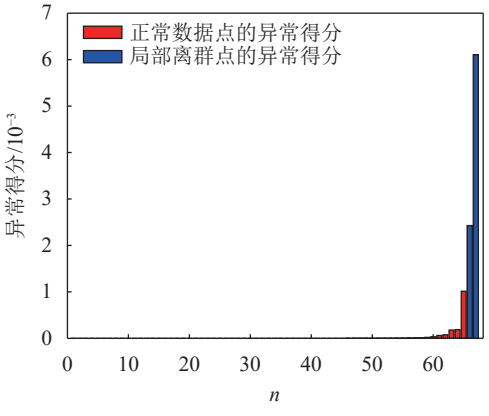
图 6　人工数据集在每个簇每1维的目标函数变化量

Fig. 6　Variation of objective function in each dimension of the artificial dataset in each cluster

中的局部离群点分布密集。由于簇2中的数据点分布稀疏，正常样本间存在很大的距离，这同样导致KNN方法无法检测出数据点分布密集聚类中的局部离群点。相反地，KLOD方法对每个簇进行单独分析，解决了簇间密度不均匀的情况。同时，KLOD方法对每个簇数据点的每1维度进行单独分析，从而突出了簇1中的局部离群点，进而实现了准确的检测。



(a) 簇1数据点的异常得分



(b) 簇2数据点的异常得分

图 7 每个簇数据点的异常得分

Fig. 7 Anomaly scores per cluster data point

表 3 人工数据集上不同方法的离群点检测性能指标

Tab. 3 Performance indicators of outlier detection by different methods in artificial dataset

方法	真阳性	假阳性	准确度	误检率
KNN	0.25	0.75	2	6
LOF	0.25	0.75	2	6
KLOD	1.00	0	8	0

表4展现了本文KLOD方法在人工数据集中剔除离群点前后的聚类性能。从表4可知，人工数据集中，使用本文KLOD方法剔除离群点后的Dunn指数、CH指数、I指数和S指数值均大于剔除离群点前的指。说明了离群点的存在会影响传统K-means算法对人工数据集的聚类效果，也进一步验证了KLOD对离群点检测的有效性。

表 4 人工数据集下本文KLOD方法去除离群点前后聚类指标对比

Tab. 4 Comparison of clustering effects before and after removing outliers by KLOD method in artificial dataset

数据集	Dunn指数	CH指数	I指数	S指数
剔除离群点前	0.4095	587.7159	0.5441	4.0315
剔除离群点后	0.4824	628.4395	0.6355	4.3268

3.2 在UCI数据集上的实验与讨论

验证本文提出的KLOD方法在真实数据集中的离群点检测性能。首先，在12个UCI数据集中构造正常数据点和离群点，如表5所示。在对UCI数据集进行实验时，所涉及的拟合函数均为$f(x)=e^{ax^2+bx+c}$形式。

表 5 构造含有离群点的UCI数据集

Tab. 5 Constructs UCI datasets with outliers

数据集	正常点	离群点
Iris	前两类数据点	第3类的5个数据点
Wine	第1类和第3类数据点	第2类的8个数据点
Seeds	第1类和第3类数据点	第2类的10个数据
Thyroid	第1类和第2类数据点	第3类的5个数据点
Wbc	良性类数据点	恶性类数据点
Aggregation	第2类、第3类、第4类和第6类数据点	第1类、第5类、第7类的各5个数据点
Vowel	最后4类数据点	第2类的5个数据点
Jain	第1类数据点	第2类的10个数据点
Wdbc	第1类数据点	第2类的10个数据点
Yeast	前4类数据点	最后6类的数据点
Page Blocks	第1类和第2类数据点	第3类的10个数据点
Pen Digits	第2类到第10类数据点	第1类的10个数据点

接着，与第3.1节类似，对比计算KLOD、KNN和LOF方法在不同UCI数据集中的真阳性、假阳性、准确率和误检率，如表6所示。

从表6可知，KLOD方法在10个数据集的检测精度达到最优，另外2个数据集Wdbc和Page Blocks的检测精度分别与KNN和LOF的检测精度持平。总体来看，在12种UCI数据集中，相比于KNN和LOF算法，本文提出的KLOD方法同样有较高的离群点检测精度。

由于Wbc、Jain和Wdbc3个数据集在构造离群数据后只剩1个簇，故不再计算这3个数据集的聚类指标，9种UCI数据集中使用本文KLOD方法去除离群点前后的聚类结果具体见表7。从表7看出，在UCI数据集，本文的KLOD方法剔除离群点后的Dunn、CH、I和S指数的数值均比剔除离群点前有所提高，说明剔除离群点后能够改善传统K-means的聚类效果。

表 6　12种UCI数据集上不同方法的检测结果对比

Tab. 6　Comparison of detection results with different methods in twelve UCI datasets

数据集	KLOD方法				KNN方法				LOF方法			
	真阳性	假阳性	准确度	误检率	真阳性	假阳性	准确度	误检率	真阳性	假阳性	准确度	误检率
Iris	1.000	0	5	0	0.4000	0.6000	2	3	0.6000	0.4000	3	2
Wine	0.7500	0.2500	6	2	0.3750	0.6250	3	5	0.6250	0.3750	5	3
Seeds	0.7500	0.2500	6	2	0.5000	0.5000	4	4	0.3750	0.6250	3	5
Thyroid	0.8000	0.2000	4	1	0.4000	0.6000	2	3	0.8000	0.2000	4	1
Wbc	0.8000	0.2000	4	1	0.6000	0.4000	3	2	0.6000	0.4000	3	2
Aggregation	1.000	0	15	0	0.8700	0.1300	13	2	0.3300	0.6700	5	10
Vowel	1.000	0	5	0	0	1.0000	0	5	0.4000	0.6000	2	3
Jain	1.000	0	10	0	0.6000	0.4000	6	4	0.7000	0.3000	7	3
Wdbc	0.6000	0.4000	6	4	0.6000	0.4000	6	4	0.3000	0.7000	3	7
Yeast	0.4054	0.5946	75	110	0.2649	0.7351	49	136	0.1514	0.8486	28	157
Page Blocks	0.4000	0.6000	4	6	0.1000	0.9000	1	9	0.4000	0.6000	4	6
Pen Digits	0.5000	0.5000	5	5	0.3000	0.7000	3	7	0.4000	0.6000	4	6

表 7　9种UCI数据集中本文KLOD方法去除离群点前后的聚类效果对比

Tab. 7　Comparison of clustering effect before and after removing outliers by KLOD method in nine UCI datasets

数据集	剔除离群点前				剔除离群点后			
	Dunn指数	CH指数	I指数	S指数	Dunn指数	CH指数	I指数	S指数
Iris	0.3626	469.3304	20.3218	6.4088	0.6041	576.8686	20.9299	10.2840
Wine	0.0394	340.4017	24.1400	10.7546	0.0751	337.4735	23.8106	13.4792
Seeds	0.0593	155.9754	9.0970	5.6559	0.0673	172.9566	7.9647	5.7966
Thyroid	0.0249	232.0610	4.2987	6.1075	0.0527	276.7681	4.3282	8.7195
Aggregation	0.0172	1698.4208	0.6055	2.5794	0.0326	1856.0150	1.3420	11.4857
Vowel	0.0416	1751.6323	0.3674	1.4926	0.0556	1766.9117	0.7030	2.1105
Yeast	0.0260	237.0314	0.0144	1.1709	0.0271	293.5319	0.0335	2.9325
Page Blocks	0.0022	3126.5723	0.1374	1.0691	0.0040	3145.2793	0.1380	1.1471
Pen Digits	0.0317	2383.5902	0.1573	55.0716	0.0329	2447.8901	0.1976	60.9416

4　结　论

本文针对局部离群点难以检测的问题提出一种基于改进K-means聚类的局部离群点检测KLOD方法，通过改进K-means聚类算法以达到更高的聚类精度，并通过最小二乘法和信息熵对数据点的每1维度进行加权，突出局部离群点的异常信息。首先，为了改进K-means聚类算法，采用快速搜索密度峰值方法计算数据点的局部密度和相对距离，并通过这两者计算数据点的γ值。然而，γ值的大小受到截断距离d_c的影响，从而导致难以直观地选择k个初始聚类中心。因此，通过使用肘部法则的方法，选取γ值最大的k个数据点作为K-means聚类算法的初始聚类中心。接着，用最小二乘法对每1维从小到大排序后的目标函数值进行拟合，该方法能够更加突出离群点的离群程度，使最终的异常得分中含有更多的离群信息。最后，在人工数据集和UCI真实数据集上进行仿真实验；结果表明，相比于KNN和LOF方法，KLOD方法能够检测出数据集中不同簇密度间的局部离群点，在离群点检测精度方面有明显提升。

然而，受到K-means算法本身的限制，对于包含形状任意的簇的数据集，本文方法的聚类效果较差，从而会影响到检测性能。因此，接下来的研究工作可以致力于提高本文的离群点检测方法在任意形状的数据集中的检测性能。

参考文献：

[1] Hawkins D M.Identification of outliers[M].London:Chapman and Hall,1980.

[2] Hilal W,Gadsden S A,Yawney J.Financial fraud:A review of anomaly detection techniques and recent advances[J].Expert Systems with Applications,2022,193:116429.

[3] Jin Fusheng,Chen Mengnan,Zhang Weiwei,et al.Intrusion detection on Internet of vehicles via combining log-ratio oversampling,outlier detection and metric learning[J].Information Sciences,2021,579:814–831.

[4] Kang S,Kim K S.Outlier behavior detection for indoor environment based on t-SNE clustering[J].Computers, Materials & Continua,2021,68(3):3725–3736.

[5] Alaverdyan Z,Jung J,Bouet R,et al.Regularized Siamese neural network for unsupervised outlier detection on brain multiparametric magnetic resonance imaging:Application to epilepsy lesion screening[J].Medical Image Analysis,2020,60:101618.

[6] Belhadi A,Djenouri Y,Srivastava G,et al.Deep learning for pedestrian collective behavior analysis in smart cities:A model of group trajectory outlier detection[J].Information Fusion,2021,65:13–20.

[7] Zhou Yu,Zhu Wenhao,Fang Qian,et al.Survey of outlier de-

tection methods based on clustering[J].Computer Engineering and Applications,2021,57(12):37–45.[周玉,朱文豪,房倩,等.基于聚类的离群点检测方法研究综述[J].计算机工程与应用,2021,57(12):37–45.]
[8] Chandola V,Banerjee A,Kumar V.Anomaly detection[J].ACM Computing Surveys,2009,41(3):1–58.
[9] Mutalib S S S A,Satari S Z,Wan Yusoff W N S.A review on outliers-detection methods for multivariate data[J].Journal of Statistical Modelling and Analytics,2021,3(1):1–15.
[10] Goldstein M,Dengel A.Histogram-based outlier score (HBOS):A fast unsupervised anomaly detection algorithm [EB/OL].[2022–12–01].https://api.semanticscholar.org/CorpusID:3590788.
[11] Li Zheng,Zhao Yue,Botta N,et al.COPOD:Copula-based outlier detection[C]//Proceedings of the 2020 IEEE International Conference on Data Mining(ICDM).Sorrento:IEEE,2020:1118–1123.
[12] Ramaswamy S,Rastogi R,Shim K.Efficient algorithms for mining outliers from large data sets[J].ACM SIGMOD Record,2000,29(2):427–438.
[13] Zhang Ke,Hutter M,Jin Huidong.A new local distance-based outlier detection approach for scattered real-world data [M]//Advances in Knowledge Discovery and Data Mining.Berlin:Springer,2009:813–822.
[14] Yang Jiawei,Rahardja S,Fränti P.Mean-shift outlier detection and filtering[J].Pattern Recognition,2021,115:107874.
[15] Breunig M M,Kriegel H P,Ng R T,et al.LOF:Identifying density-based local outliers[C]//Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data.Dallas Texas:ACM,2000:93–104.
[16] Tang Jian,Chen Zhixiang,Fu A W C,et al.Enhancing effectiveness of outlier detections for low density patterns[M]//Advances in Knowledge Discovery and Data Mining.Berlin:Springer,2002:535–548.
[17] Latecki L J,Lazarevic A,Pokrajac D.Outlier detection with kernel density functions[C]//Proceedings of the 5th international conference on Machine Learning and Data Mining in Pattern Recognition.New York:ACM,2007:61–75.
[18] MacQueen J.Some methods for classification and analysis of multivariate observations[C]//Proceedings of the fifth Berkeley symposium on mathematical statistics and probability.Berkeley:University of California Press,1967:281–297.
[19] He Zengyou,Xu Xiaofei,Deng Shengchun.Discovering cluster-based local outliers[J].Pattern Recognition Letters,2003,24(9/10):1641–1650.
[20] Al–Zoubi M B,Al–Dahoud A,Yahya A A.New outlier detection method based on fuzzy clustering[J].WSEAS Transactions on Information Science and Applications,2010,7(5):681–690.
[21] Barai A,Dey L.Outlier detection and removal algorithm in K-means and hierarchical clustering[J].World Journal of Computer Application and Technology,2017,5(2):24–29.
[22] Ahmed M,Mahmood A N.A novel approach for outlier detection and clustering improvement[C]//Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications(ICIEA).Melbourne:IEEE,2013:577–582.
[23] Zhou Yu,Zhu Wenhao,Sun Hongyu.A local outlier detection method based on objective function[J].Journal of Northeastern University(Natural Science),2022,43(10):1405–1412.[周玉,朱文豪,孙红玉.一种基于目标函数的局部离群点检测方法[J].东北大学学报(自然科学版),2022,43(10):1405–1412.]
[24] Rodriguez A,Laio A.Clustering by fast search and find of density peaks[J].Science,2014,344(6191):1492–1496.
[25] Liu Fan,Deng Yong.Determine the number of unknown targets in open world based on elbow method[J].IEEE Transactions on Fuzzy Systems,2021,29(5):986–995.
[26] Lenth R V.Least-squares means:The *R* package lsmeans[J].Journal of Statistical Software,2016,69(1):1–33.
[27] Mo Xiaoqin.Linear and nonlinear fitting based on least squares method[J].Wireless Internet Technology,2019,16(4):128–129.[莫小琴.基于最小二乘法的线性与非线性拟合[J].无线互联科技,2019,16(4):128–129.]
[28] Yu Qingying,Luo Yonglong,Chen Chuanming,et al.Neighborhood relevant outlier detection approach based on information entropy[J].Intelligent Data Analysis,2016,20(6):1247–1265.
[29] Al Shalabi L,Shaaban Z,Kasasbeh B.Data mining:A preprocessing engine[J].Journal of Computer Science,2006,2(9):735–739.
[30] Zhang Zhongping,Qiu Jingyang,Liu Cong,et al.Outlier detection based on cluster outlier factor and mutual density[J].Computer Integrated Manufacturing Systems,2019,25(9):2314–2323.[张忠平,邱敬仰,刘丛,等.基于聚类离群因子和相互密度的离群点检测算法[J].计算机集成制造系统,2019,25(9):2314–2323.]
[31] Feng Guilan,Li Zhengnan,Zhou Wengang,et al.Entropy-based outlier detection using spark[J].Cluster Computing,2020,23(2):409–419.
[32] Liu Yanchi,Gao Xuedong,Guo Hongwei,et al.Ensembling clustering validation indices[J].Computer Engineering and Applications,2011,47(19):15–17.[刘燕驰,高学东,国宏伟,等.聚类有效性的组合评价方法[J].计算机工程与应用,2011,47(19):15–17.]
[33] Gan Guojun,Ng M K P.K-means clustering with outlier removal[J].Pattern Recognition Letters,2017,90:8–14.

Local Outlier Detection Method Based on Improved K-means

ZHOU Yu[1], *XIA Hao*[1], *YUE Xuezhen*[1], *WANG Peichong*[2]

(1.School of Electrical Eng., North China Univ. of Water Resources and Electric Power, Zhengzhou 450045, China;
2.School of Info. Eng., Hebei Univ. of Geosciences, Shijiazhuang 050031, China)

Abstract:

Objective Outliers are defined as data points generated for various special reasons. They are often regarded as noise points due to their deviation from normal data points and are considered points of research value, occupying a small proportion of the dataset. The task of outlier detection in-

volves identifying these points and analyzing their potential abnormal information through the analysis of data attribute features. This process aims to uncover unusual patterns or behaviors within the dataset that can provide insights into unique phenomena or anomalies. Most clustering-based outlier detection methods primarily detect outliers in the dataset from a global perspective, with weaker performance in detecting local outliers. Hence, an improved K-means clustering algorithm is proposed by introducing fast search and discovering density peak methods. A local outlier detection method, named KLOD (local outlier detection based on improved K-means and least squares methods), is developed to achieve precise detection of local outliers.

Methods The K-means clustering algorithm is characterized by hard clustering, meaning that after clustering the dataset, each data point has a clear association with one cluster or another. This property makes it suitable for outlier detection, as outliers significantly affect the clustering process. However, selecting initial cluster centers and determining the number of clusters is crucial as they directly impact the clustering effectiveness. To select the accurate cluster center, clustering by fast search and finding density peaks is utilized to compute the local density and relative distance of data points, constructing a decision graph based on these metrics. The challenge lies in accurately determining the cutoff distance d_c, making it difficult to precisely identify the number of cluster centers from the decision graph obtained using a single d_c value. The elbow method is employed to determine the optimal number of clusters for an unknown dataset for the best clustering effectiveness to address the challenge of determining the number of clusters. When clustering data into different numbers of clusters, the cost function value changes accordingly. The number of clusters is depicted on the x-axis, and the cost function value is on the y-axis. The changes in the cost function value with the number of clusters are recorded and plotted as a line graph. When there is no significant decrease in the cost function value with an increase in the number of cluster centers, the position of the "elbow" is observed to determine the optimal number of clusters. After determining the initial cluster centers and the number of clusters k, the dataset is clustered using the K-means clustering algorithm to obtain k clusters and their corresponding cluster centers. The objective function value for each data point in each dimension within each cluster is then computed. Then, the objective function values for each dimension of the data points in each cluster are sorted in ascending order. The objective function values, sorted in ascending order, are fitted using the least squares approach to obtain a curve. The derivative of this fitted curve is then calculated to obtain the slope, providing insight into the rate of change of the objective function values within each cluster. Each dimension's degree of dispersion and information content can vary in the dataset, so different weights are assigned to each dimension. Information entropy is employed to measure the dataset's degree of dispersion, and higher weightage is given to dimensions with higher outlier degrees to represent their impact on the overall dataset. By incorporating information entropy, each dimension's objective function value for each data point is weighted by the corresponding change rate. This process results in the final anomaly score, and the top-n data points with high anomaly scores are considered outliers.

Results and Discussions The experimental results indicated that in the artificial dataset, KLOD, KNN, and LOF all detect sparse local outliers effectively. However, the LOF algorithm struggles to detect outliers within outlier clusters. Additionally, the KNN method cannot detect local outliers within densely distributed clusters when there is a considerable distance between normal data points. In contrast, the KLOD method analyzes each cluster individually, addressing the issue of uneven cluster densities. The KLOD method analyzes each dimension of the data points within each cluster separately, achieving accurate detection. In the UCI dataset, the KLOD method achieves optimal detection accuracy in 10 datasets, with detection accuracy on par with KNN and LOF in 2 datasets. Compared to the KNN and LOF algorithms, KLOD also demonstrates high accuracy in outlier detection. The fast search density peak method is applied to calculate the local density and relative distance of data points, and the γ value of each data point is determined based on these two metrics to improve the K-means clustering algorithm. However, the size of γ is influenced by the cutoff distance d_c, making it difficult to intuitively choose k initial cluster centers. Hence, the elbow method selects the k data points with the largest γ values as initial cluster centers for the K-means clustering algorithm. Least squares fitting is employed to fit the objective function values for each dimension sorted in ascending order. This method highlights the degree of outlierness of outliers, incorporating more outlier information into the final anomaly score.

Conclusions Experimental results on artificial and UCI real datasets demonstrated that the KLOD method can detect local outliers with moderate outliers. Compared to the KNN and LOF methods, it significantly improves detection accuracy. However, due to limitations of the K-means algorithm itself, its clustering performance is poor for datasets containing arbitrarily shaped clusters, affecting detection performance. Therefore, future studies can focus on enhancing the performance of outlier detection methods on datasets with arbitrary cluster shapes.

Key words: outlier detection; K-means; least squares method; peak density; objective function

（编辑　赵　婧）

引用格式: Zhou Yu,Xia Hao,Yue Xuezhen,et al.Local outlier detection method based on improved K-means[J].Advanced Engineering Sciences,2024,56(4):66–77.[周玉,夏浩,岳学震,等.基于改进K-means的局部离群点检测方法[J].工程科学与技术,2024,56(4):66–77.]