ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 38,No. 3,May 2023,pp. 643－651
DOI:10. 16337/j. 1004-9037. 2023. 03. 013

http://sjcj. nuaa. edu. cn
E-mail:sjcj@nuaa. edu. cn
Tel/Fax:＋86-025-84892742

基于Tukey规则与初始中心点优化的K-means聚类改进算法

柳　菁[1]，邱紫滢[1]，郭茂祖[2]，余冬华[1]

(1. 绍兴文理学院计算机科学与工程系，绍兴 312000;2. 北京建筑大学电子信息工程学院，北京 100044)

摘　要：针对K-means聚类算法存在的初始中心点选择及异常点、离群点极易影响聚类结果等待改进问题，提出了一个基于Tukey规则与优化初始中心点选择的K-means改进算法。该算法利用Tukey规则构造核心与非核心子集，将聚类过程划分成2个阶段。同时，在核心子集上执行中心点逐个递增优化选择策略，选出初始中心点。在来自UCI的20个数据集上聚类结果表明，本文提出的算法优于K-means＋＋聚类算法，有效地提升了聚类性能。

关键词：数据挖掘;K-means聚类算法;Tukey规则;中心点优化

中图分类号：TP391　　**文献标志码：**A

Improved K-means Clustering Algorithm Based on Tukey Rule and Initial Center Point Optimization

LIU Jing[1], QIU Ziying[1], GAO Maozu[2], YU Donghua[1]

(1. Department of Computer Science and Engineering, Shaoxing University, Shaoxing 312000, China;2. School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China)

Abstract: Aiming at shortcomings of the K-means algorithm to be improved, such as selection of initial center points and the problems that abnormal points and outliers can easily affect the clustering results, this paper proposes an improved K-means algorithm based on Tukey rules and optimizing initial center points selection. The proposed algorithm uses Tukey rules to construct core and non-core subsets, and divides the clustering process into two stages. At the same time, the strategy of increasing the center points one by one is implemented on the core subset to optimize the initial center points. The clustering results on 20 real-world datasets from UCI show that the proposed algorithm is better than the most popular K-means＋＋ clustering algorithm and effectively improves the clustering performance.

Key words: data mining; K-means clustering algorithm; Tukey rule; center point optimization

引　　言

K-means是由20世纪60年代提出的一种基于划分的聚类算法。尽管后来的研究者提出其他性能优异的聚类算法，如AP[1]、DPC[2]等，但K-means因其简单、高效且易于理解而成为最受欢迎的聚类分

基金项目：国家自然科学基金(62002227);绍兴文理学院校级科研项目(2021LG004)。
收稿日期：2022-03-24;**修订日期：**2022-06-23

析方法之一，并且成功应用于各行各业[3]，如医疗数据[4]、图像聚类与分割[5-6]、电子资源学习分析[7]和网络聚类[8]等。然而，传统K-means聚类算法也存在很多不足之处，归纳起来主要有：(1)初始中心点选择问题；(2)聚类结果敏感问题，也常称为局部最优问题；(3)需要固定簇个数；(4)极易受到异常点、离群点等特殊点影响等等。很多学者针对其中一个或者多个问题进行各种改进研究，提出相应的改进算法。高文欣等[9]提出闪电分叉过程算法优化K-means以改进局部最优；Gan等[10]提出的KMOR算法考虑了异常点、离群点问题，该算法将这些点归属于一个额外的簇，重新定义了聚类目标函数，兼顾了异常点、离群点与中心点的误差平方和，从而实现目标K个簇的聚类。关于优化异常点、离群点的K-means算法，还有Im等[11]的NK-means算法，Olukanmi等[12]的K-means-Sharp算法，以及Shrifan等[13]提出的改进K-means算法。然而，这些算法都强调移除异常点、离群点，然后再进行聚类分析。对于一个待聚类分析的实际数据集来说，很多情形下不一定在意它们是否是异常点，因为异常点本身难以界定，但是一定会希望给它们赋值一个类别(簇)标签。Fränti等[14]研究指出，一个好的中心点初始化技巧或多次重复运行K-means算法可以显著提升K-means聚类性能。荀超等[15]提出混沌理论优化搜索算法以获取更优的精英粒子充当K个中心点。Torrente等[16]提出两阶段层叠式K-means聚类，在第1阶段中运用K-means进行引导复制出初始种子集，在第2阶段中基于种子空间组装聚类。Vassilvitskii等[17]提出了K-means++改进算法，提供了一种称为D^2-sampling的简单却非常有效的中心点选择方式。该算法的中心点选择采用递增方式，并且给每一个潜在中心点赋予不同的选择概率。自从K-means++作为一种默认的K-means聚类算法嵌入到scikit-learn之后，当使用者选择基于划分的聚类算法时，几乎都会选择K-means++聚类算法。Bachem等[18]采用基于MCMC-sampling替换K-means++中的D^2-sampling得到了一种近线性的改进K-means算法，称为K-MC^2，但是其定义了2个依赖于数据的假设量$\alpha(X)$和$\beta(X)$，对结果及计算复杂度都会产生重要影响。他们还在K-means++的D^2-sampling基础上扩展了一个正则项，提出了AFK-MC^{2}算法，克服了K-MC^{2}算法的假设缺陷[19]。Tan等[20]探究了黎曼流形上K-means的初始中心点问题，提出了基于测地线投影的可学习簇个数。以上算法都考虑初始中心点的选择问题，更多关于初始中心点选择优化的方法，可以参考Emre等[21]的总结文献。行艳妮等[22]也总结了在Spark环境下K-means算法初始中心点优化的主要方法及最新进展。本文针对初始中心点选择及敏感性问题，提出了分阶段聚类及初始中心点优化方法，称为K-meansCC(K-means core subset and center points optimization)算法，以进一步提升K-means算法的聚类性能。

1 K-means聚类算法

假设给定数据集$X=\{x_1, x_2, \cdots, x_n\}$，$x_i \in \mathbf{R}^m$，聚类分析就是将其划分成$K$个互不相交的集合(簇)$C=\{C_1, \cdots, C_K\}$，使得$\bigcup_i^K C_i = X$且$C_i \bigcap C_j = \Phi, \forall i, j, i \neq j$。K-means算法最终的数据结果也是$K$个互不相交的集合(簇)$C=\{C_1, \cdots, C_K\}$，其主要思想如下：

(1) 随机初始化K中初始中心点；

(2) 计算每个数据点到K个初始化聚类中心的距离，按照最近原则将该点分配到其中心点所对应的簇，完成第一次聚类划分；

(3) 重新计算每个簇的中心点，更新中心点；

(4) 计算每个数据点到K个新聚类中心的距离，按照最近原则将该点分配到其中心点所对应的簇，重新完成聚类划分；

(5) 如果中心点不再发生变化或者达到规定的最大迭代次数，则输出聚类结果$C=\{C_1, \cdots, C_K\}$，否则，转入步骤(3)。

2　改进的K-means算法

2.1　核心子集

Tukey规则是异常检测中常用技巧[13,23-24],即超过上四分位加上1.5倍四分位距的点,或者下四分位减去1.5倍四分位距的点为异常值。本文将借助Tukey规则构建核心子集。

首先计算数据第j维度上的第一、第三分位数Q_1^j、Q_3^j,即

$$\begin{cases} Q_1^j = x_i^j | i = \text{round}((n+1)\times 0.25) \\ Q_3^j = x_i^j | i = \text{round}((n+1)\times 0.75) \end{cases} \tag{1}$$

式中:round(•)为取整函数;n表示数据个数。

然后计算出上下界B_{lower}和B_{upper},即

$$\begin{cases} B_{\text{lower}}^j = Q_1^j - r\times \text{IQR}^j \\ B_{\text{upper}}^j = Q_3^j + r\times \text{IQR}^j \end{cases} \tag{2}$$

式中:$\text{IQR}^j = Q_3^j - Q_1^j$;$r$为尺度因子,用于控制核心子集大小。

最后计算核心集S_{core}与非核心集S_{noncore},即

$$\begin{cases} S_{\text{core}} = \left\{ x_i \in X | B_{\text{lower}}^j \leqslant x_{ij} \leqslant B_{\text{upper}}^j, \forall j \in \{1,\cdots,m\} \right\} \\ S_{\text{noncore}} = X - S_{\text{core}} \end{cases} \tag{3}$$

式(3)表明:本文将对数据点每一个维度单独考察,然后综合所有m个维度确定其是否属于核心集S_{core},只要存在一个维度不满足约束条件,其将被判断为属于S_{noncore}。式(2)中的尺度因子r是一个可调整的预定义参数,如果对数据集X有充足的先验知识,可以按照经验设置,如果没有,那么推荐设置$r=1.5$。此时,不能将S_{noncore}当作异常点而遗弃掉,尽管在异常检测研究领域,$r=1.5$往往被当作异常值的边界值,仍然需要对其进行类别标签赋值。在本文的15个真实数据集上,每一个样本点都有确切的类别标签,但是几乎所有数据集的S_{noncore}均非空。而S_{core}的构造,往往更有助于获取更优异的初始中心点,不仅如此,在本文中S_{core}有效辅助了初始中心点的选择,还对中心点的更新方面产生了积极方面的影响。无论是S_{core},还是S_{noncore}中的点,在最终的聚类结果中都需要赋予簇标签,这也是聚类分析目标之一。

2.2　初始中心点优化

优化后的初始中心点选择方法采取中心点逐个递增的策略,直至达到所期望的K个中心点。为此,首先需要定义点x与集合S的距离函数,即

$$d(x,S) = \min_{x_j \in S} d(x,x_j) \tag{4}$$

式中$d(x,x_j)$表示两个点x,x_j之间的距离,本文选择欧氏距离。

记$c_i, i=1,\cdots,K$表示簇$C_i, i=1,\cdots,K$的中心点,则第1个中心点c_1的选择方式为

$$c_1 = \frac{1}{|S_{\text{core}}|}\sum_{x_i \in S_{\text{core}}} x_i \tag{5}$$

式中:$|S_{\text{core}}|$表示核心子集S_{core}中元素个数;c_1为核心子集S_{core}的均值点。

记$C^k=\{c_1,\cdots,c_k\}$表示含有k个中心点的集合,则第$k+1$个中心点c_{k+1}的选择方式为

$$c_{k+1} = \arg\max_{x_i \in S_{\text{core}}} d(x_i, C^k) \tag{6}$$

则$C^{k+1} = C^k \bigcup \{c_{k+1}\}$。式(6)表明,$c_{k+1}$是核心子集$S_{\text{core}}$中远离已选中心点中距离最远的那个点。按照式(3,6)的选择方法,很显然,如果不区分S_{core}和S_{noncore},则c_2几乎都会是属于非核心集S_{noncore}中的点,

即$c_2 \in S_{\text{noncore}}$，而$c_i, i>2$也会有很大几率选中非核心集$S_{\text{noncore}}$中的点。

2.3 算法流程

算法1给出了K-meansCC算法的具体描述，其中，步骤(1)对应2.1节，确定S_{core}和S_{noncore}，步骤(2～5)对应2.2节，属于初始中心点优化过程，而步骤(16～19)是对S_{noncore}中点进行簇标签赋值。

算法1 K-meansCC

输入：数据集X，簇个数K，尺度因子r；

输出：聚类结果$C=\{C_1, \cdots, C_K\}$，中心点集$\hat{C}^K$，误差平方和SSE。

//核心子集

(1) 用式(3)确定S_{core}和S_{noncore}；

//初始中心点优化

(2) 用式(5)确定c_1；

(3) for $i=2$ to K do

(4) 用式(6)确定c_i；

(5) end for

//赋值簇标签

(6) for $j=1$ to max_iter do

(7) for $\forall x \in S_{\text{core}}$ do

(8) 按照x与C^K的最近距离原则，将x划归对应簇；

(9) end for

(10) 更新中心点集C^K，并计算SSE；

(11) If SSE不发生变化 then

(12) break；

(13) end if

(14) end for

(15) 计算最优中心点$\hat{C}^K$；

(16) for $\forall x \in S_{\text{noncore}}$ do

(17) 按照x与$\hat{C}^K$的最近距离原则，将x划归对应簇；

(18) end for

(19) 计算误差平方和SSE；

(20) return 聚类结果$C=\{C_1, \cdots, C_K\}$，中心点集$\hat{C}^K$，误差平方和SSE。

3 算法性能分析

本节将分析K-meansCC算法的聚类性能，并与最受欢迎的K-means-Random和K-means++算法[17]进行比较，其中K-means-Random和K-means++算法采用Scikit-learn方式[25]提供，K-meansCCT算法采用Python编写，为了叙述方便，后文用K-means代表K-means-Random算法。

3.1 数据集与评价准则

本文用于性能测试的20个真实数据集均来自UCI(http://archive.ics.uci.edu/ml)，数据集大小n、数据维度m和簇个数K如表1所示。

表1　真实数据集

Table 1　Real-world datasets

数据集	n	m	K	数据集	n	m	K
breast-cancer	569	30	2	banknote	1 372	4	2
bupa	345	6	2	compound	399	2	6
ct	221	36	2	haberman	306	3	2
hayes-roth	132	5	3	iris	150	4	3
libras	360	90	15	newthyroid	215	5	3
parkinsons	195	22	2	pima	768	8	2
seeds	210	7	3	sonar	208	23	2
vowel	990	10	11	waveform21	5 000	21	3
waveform40	5 000	40	3	wdbc	569	30	2
wine	178	13	3	aggregation	788	2	7
abalone	4 168	7	21	HOP_S1	52 482	6	4
sensor	5 456	24	4	R15	600	2	15

本文选择ARI(Adjusted rand index)[26]与NMI(Normalized mutual information)[27]作为聚类性能评价准则,具体见式(7,8),其值越大,表示聚类性能越优异,最大值为1。

$$\mathrm{ARI}=\frac{\mathrm{RI}-E[\mathrm{RI}]}{\max(\mathrm{RI})-E[\mathrm{RI}]} \tag{7}$$

$$\mathrm{NMI}(U,V)=\frac{\mathrm{MI}(U,V)}{\mathrm{mean}(H(U),H(V))} \tag{8}$$

式中:RI为兰德系数;E为期望;MI为互信息;H为熵;U、V分别为聚类簇标签与真实簇标签。

3.2　聚类性能分析

表2,3分别给出了真实数据集中K-means、K-means++和K-meansCC算法的性能度量ARI和NMI值,其中所有数据集均取$r=1.5$用于划分核心集与非核心集。最优聚类性能值加黑表示,并且数值0.000 0表示其实际值<0.000 1,而非0。

从表2中ARI评价值来看,K-means算法性能最差,其次为K-means++算法,K-meansCC算法聚类性能在20个数据集中的15个获取最优评价值,取得聚类性能的显著提升,这也包括数据量超过50 000的HOP_S1数据集。而从表3中NMI评价值看出,尽管有个别数据集,如abalone与HOP_S1,出现性能反转,其K-means++算法的ARI分别为0.049 7和0.021 4,低于K-meansCC算法的0.054 8和0.022 0,但这并不影响K-meansCC性能提升,在最优数据集个数方面,K-meansCC算法仍然以13∶8占据优势。因此,无论是从ARI,还是NMI,均表明K-meansCC算法聚类性能优于K-means与K-means++算法,提升了聚类性能。表2,3还反映出:数据集aggregation在ARI与NMI评价上保持一致,分别以K-means++算法的0.762 4和0.879 2为最优值,即对于该数据集来说,K-means++算法是3种算法中最适宜的。图1中给出了K-means、K-means++与K-meansCC算法的聚类结果散点图,在每一幅图中,相同颜色的点代表同一个簇。从图1(a,b)可以看出,K-means与K-means++算法均将左下方两个点较少的不同簇划归为同一个簇,将下方中间那个点多的簇分裂出3个不同的簇;而K-meansCC算法没有出现簇的消融,正确地分出7个簇,但也错误地划分部分点。因此,K-meansCC算法提供的聚类结果更接近真实类划分。

表2 真实数据集上ARI聚类评价结果

Table 2 ARI evaluation of real-world datasets clustering results

数据集	K-means	K-means++	K-meansCC
breast-cancer	0.491 4	0.491 4	0.617 5
compound	0.532 8	0.537 8	0.413 3
ct	0.416 0	0.416 0	0.439 9
hayes-roth	0.016 0	0.020 2	0.022 6
iris	0.730 2	0.730 2	0.730 2
newthyroid	0.546 5	0.579 0	0.097 1
parkinsons	0.085 3	0.000 0	0.062 5
pima	0.074 3	0.074 3	0.074 3
seeds	0.716 6	0.716 6	0.699 8
sonar	0.004 9	0.004 5	0.006 4
vowel	0.218 0	0.202 8	0.220 4
waveform21	0.253 6	0.253 6	0.254 6
waveform40	0.251 6	0.251 6	0.252 5
wdbc	0.491 4	0.491 4	0.617 5
wine	0.371 1	0.371 1	0.371 1
aggregation	0.754 7	0.762 4	0.743 8
abalone	0.047 7	0.049 7	0.054 8
HOP_S1	0.021 4	0.021 4	0.022 0
sensor	0.057 0	0.056 9	0.069 9
R15	0.914 2	0.992 7	0.992 7
最优个数	5	8	15

表3 真实数据集上NMI聚类评价结果

Table 3 NMI evaluation of real-world datasets clustering results

数据集	K-means	K-means++	K-meansCC
breast-cancer	0.464 7	0.464 7	0.537 2
compound	0.722 0	0.719 1	0.624 0
ct	0.329 6	0.329 6	0.348 5
hayes-roth	0.025 0	0.028 7	0.031 7
iris	0.758 1	0.758 1	0.758 1
newthyroid	0.475 7	0.494 5	0.151 2
parkinsons	0.050 5	0.000 0	0.049 3
pima	0.029 5	0.029 5	0.029 5
seeds	0.694 9	0.694 9	0.702 8
sonar	0.016 6	0.016 0	0.019 0
vowel	0.433 2	0.414 1	0.433 7
waveform21	0.362 2	0.362 2	0.364 7
waveform40	0.360 5	0.360 5	0.361 6
wdbc	0.464 7	0.464 7	0.537 2
wine	0.428 7	0.428 7	0.428 7
aggregation	0.869 3	0.879 2	0.837 3
abalone	0.170 7	0.171 9	0.167 7
HOP_S1	0.074 5	0.074 5	0.073 9
sensor	0.112 5	0.112 4	0.091 0
R15	0.964 1	0.994 2	0.994 2
最优个数	7	8	13

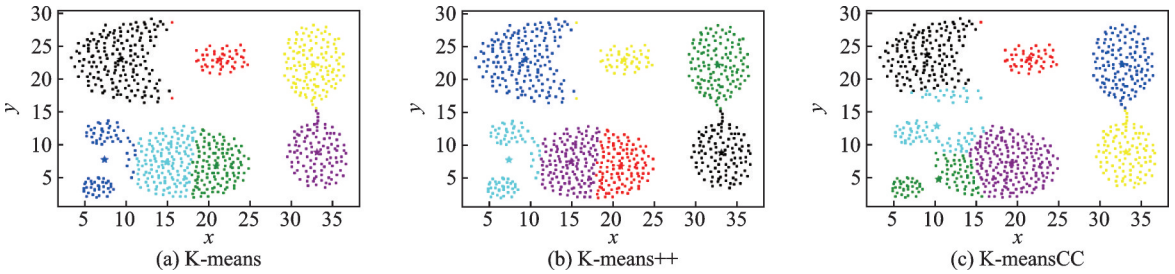


图1 3种算法在数据集aggregation上聚类结果

Fig.1 Clustering results of three algorithms on aggregation dataset

K-meansCC算法借助于Tukey规则实现S_{core}与S_{noncore}的划分，因此需要给定尺度因子r。Tukey规则来自异常检测领域，一般地，将尺度因子设置为$r=1.5$，如果某个点任一维度不满足Tukey规则，将会判定为异常点。在大部分情形下，这些点将被直接丢弃。这个想法被引入到聚类分析中后，一般用于数据预处理阶段，被检测为异常点的点也被丢弃，而不赋值簇标签。这将带来巨大隐患。表4给出了当设置$r=1.5$时K-meansCC算法聚类20个真实数据集的S_{core}与S_{noncore}结果。除了数据集compound、aggregation和R15的S_{noncore}为空集之外，其他剩余的17个数据集的S_{noncore}均非空。然而这些数据集都

表4　K-meansCC聚类中核心与非核心子集点个数

Table 4　Numbers of core and non-core subset elements in K-meansCC clustering

数据集	$\|S_{core}\|$	$\|S_{noncore}\|$	数据集	$\|S_{core}\|$	$\|S_{noncore}\|$
breast-cancer	398	171	compound	399	0
ct	164	57	hayes-roth	102	30
iris	146	4	newthyroid	163	52
parkinsons	148	47	pima	639	129
seeds	205	5	sonar	149	59
vowel	960	30	waveform21	4 740	260
waveform40	4 116	884	wdbc	398	171
wine	161	17	aggregation	788	0
abalone	4 016	152	HOP_S1	48 165	4 317
sensor	1 164	4 292	R15	600	0

是真实数据集,而且所有点都标注了真实标签。因此,就算是考虑异常点的聚类算法,也不应该将这些点直接移除。而本文的K-meansCC算法采用了Tukey规则,但是最终也对$S_{noncore}$中的点赋值了簇标签,最终的聚类性能也获得提升。

表5给出了3种算法聚类结果的误差平方和(Sum of squared error, SSE)。整体上来看,K-meansCC算法的SSE相较于K-means和K-means++算法来说几乎均维持在同一个量级,但是前者稍大。这是

表5　3种算法聚类结果误差平方和

Table 5　Sum of squared errors of clustering results for three algorithms

数据集	K-means	K-means++	K-meansCC
breast-cancer	7.794 3E+07	7.794 3E+07	1.046 0E+08
compound	3.866 1E+03	3.865 9E+03	5.333 2E+03
ct	1.287 0E+02	1.287 0E+02	1.331 9E+02
hayes-roth	2.174 1E+04	2.171 8E+04	2.186 1E+04
iris	7.885 1E+01	7.885 1E+01	7.906 6E+01
newthyroid	2.887 7E+04	2.856 0E+04	4.089 9E+04
parkinsons	1.165 8E+06	1.165 8E+06	1.534 0E+06
pima	5.142 4E+06	5.142 4E+06	5.858 6E+06
seeds	5.873 2E+02	5.873 2E+02	5.881 5E+02
sonar	3.291 6E+01	3.291 6E+01	3.996 4E+01
vowel	1.935 3E+03	1.925 6E+03	1.940 4E+03
waveform21	1.331 2E+05	1.331 2E+05	1.331 5E+05
waveform40	2.275 9E+05	2.275 9E+05	2.276 5E+05
wdbc	7.794 3E+07	7.794 3E+07	1.046 0E+08
wine	2.370 7E+06	2.370 7E+06	2.372 8E+06
aggregation	1.100 0E+04	1.099 7E+04	1.124 3E+04
abalone	3.149 2E+01	3.100 0E+01	4.171 0E+01
HOP_S1	4.235 4E+05	4.235 4E+05	4.316 5E+05
sensor	1.480 9E+05	1.480 9E+05	2.050 0E+05
R15	1.639 7E+02	1.086 2E+02	1.086 2E+02

由于K-meansCC算法基于Tukey规则将聚类分为2个阶段进行，而S_{noncore}中的点不参与中心点更新，致使K-meansCC算法最后的中心点相较于K-means和K-means＋＋算法会更偏离分布中心，因此导致最终的SSE会增大。

4 结束语

本文提出了一种改进的K-means聚类算法K-meansCC，主要是将Tukey规则逐维度应用于数据集，以获得S_{core}和S_{noncore}，进而将聚类过程划分成2个阶段，然后在S_{core}上执行中心点逐个递增优化选择策略，优化初始中心点。K-meansCC优化了初始中心点选择，同时降低了聚类结果对中心点的敏感性，在20个真实数据集实验结果表明，其优于K-means和K-means＋＋算法，提升了聚类性能。同时也表明，如果在聚类分析中借鉴了异常检测相关想法，也需要对被检测出的异常点进行簇标签赋值，而不应该直接遗弃。Tukey规则的引入促使K-meansCC算法性能的提升，但是如何更合适的预定义Tukey规则中参数需要进一步的研究。同时，本文Tukey规则是逐个维度作用于数据，对于高纬度数据可能会过渡标记异常点，如何更加合理地应用Tukey规则也值得更多探究。

参考文献：

[1] BRENDAN J F, DELBERT D. Clustering by passing messages between data points[J]. Science, 2007, 315(5814): 972-976.

[2] RODRIGUEZ A, LAIO A. Clustering by fast search and find of density peaks[J]. Science, 2014, 344(6191): 1492-1496.

[3] XIE Hailun, LI Zhang, CHEE P L, et al. Improving K-means clustering with enhanced firefly algorithms[J]. Applied Soft Computing, 2019, 84: 105763.

[4] XU Z, SHEN D, NIE T, et al. A cluster-based oversampling algorithm combining SMOTE and K-means for imbalanced medical data[J]. Information Sciences, 2021, 572: 574-589

[5] CHEN W, HE C, JI C, et al. An improved K-means algorithm for underwater image background segmentation[J]. Multimedia Tools and Applications, 2021, 80(14): 21059-21083.

[6] MEDEGHRI H, SABEUR S A. Anatomic compartments extraction from diffusion medical images using factorial analysis and K-means clustering methods: A combined analysis tool[J]. Multimedia Tools and Applications, 2021, 80(16): 23949-23962.

[7] MOUBAYED A, INJADAT M, SHAMI A, et al. Student engagement level in an e-learning environment: Clustering using K-means[J]. American Journal of Distance Education, 2020, 34(2): 137-156.

[8] SIERANOJA S, FRÄNTI P. Adapting K-means for graph clustering[J]. Knowledge and Information Systems, 2022, 64(1): 115-142.

[9] 高文欣，刘升，肖子雅.闪电分叉过程算法优化的K-means聚类[J].运筹与管理，2021，30(12): 35-41.
GAO Wenxin, LIU Sheng, XIAO Ziya. K-means clustering optimized by lightning attachment procedure optimization[J]. Operations Research and Management Science, 2021, 30(12): 35-41.

[10] GAN Guojun, MICHAEL K N. K-means clustering with outlier removal[J]. Pattern Recognition Letters, 2017, 90:8-14.

[11] IM S, QAEM M M, MOSELEY B, et al. Fast noise removal for K-means clustering[C]//Proceedings of International Conference on Artificial Intelligence and Statistics.[S.l.]: PMLR, 2020: 456-466.

[12] OLUKANMI P O, TWALA B. K-means-sharp: Modified centroid update for outlier-robust K-means clustering[C]// Proceedings of 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech).[S.l.]: IEEE, 2017: 14-19.

[13] SHRIFAN N H, AKBAR M F, MAT ISA N A. An adaptive outlier removal aided K-means clustering algorithm[J]. Journal of King Saud University-Computer and Information Sciences, 2022, 34(8): 6365-6376.

[14] FRÄNTI P, SIERANOJA S. How much can K-means be improved by using better initialization and repeats?[J]. Pattern Recognition, 2019, 93: 95-112.

[15] 荀超，陈伯建，吴翔宇，等.基于改进K-means算法的电力短期负荷预测方法研究[J].电力科学与技术学报，2022(1): 90-95.
XUN Chao, CHEN Bojian, WU Xiangyu, et al. Research on short-term power load forecasting method based on improved K-

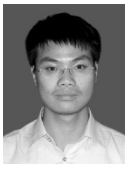
means algorithm[J]. Journal of Electric Power Science and Technology,2022(1): 90-95.
[16] TORRENTE A, ROMO J. Initializing K-means clustering by bootstrap and data depth[J]. Journal of Classification, 2021, 38(2): 232-256.
[17] VASSILVITSKII S, ARTHUR D. K-means++: The advantages of careful seeding[C]//Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms. [S.l.]: ACM, 2006: 1027-1035.
[18] BACHEM O, LUCIC M, HAMED HASSANI S, et al. Approximate K-means++ in sublinear time[C]//Proceedings of Thirtieth AAAI Conference on Artificial Intelligence. [S.l.]: AAAI, 2016: 1459-1467.
[19] BACHEM O, LUCIC M, HASSANI H, et al. Fast and provably good seedings for K-means[J]. Advances in Neural Information Processing Systems, 2016, 29: 55-63.
[20] TAN C, ZHAO H, DING H. Statistical initialization of intrinsic K-means clustering on homogeneous manifolds[J]. Applied Intelligence, 2022, 53: 1-20.
[21] EMRE C M, KINGRAVI H A, VELA P A. A comparative study of efficient initialization methods for the K-means clustering algorithm[J]. Expert Systems with Applications, 2013, 40(1): 200-210.
[22] 行艳妮,钱育蓉,南方哲,等.Spark环境下K-means初始中心点优化研究综述[J].计算机应用研究,2020,37(3): 641-647.
XING Yanni, QIAN Yurong, NAN Fangzhe, et al. Survey of optimization on K-means algorithm in Spark[J]. Application Research of Computers ,2020,37(3): 641-647.
[23] HUYGHUES-BEAUFOND N, TINDEMANS S, FALUGI P, et al. Robust and automatic data cleansing method for short-term load forecasting of distribution feeders[J]. Applied Energy, 2020, 261: 114405.
[24] SEO S. A review and comparison of methods for detecting outliers in univariate data sets[D]. Pittsburgh: University of Pittsburgh, 2006.
[25] PEDREGOSA F, VAROQUAUX G, GRAMFORT A, et al. Scikit-learn: Machine learning in Python[J]. Journal of Machine Learning Research, 2011,12: 2825-2830.
[26] STEINLEY D. Properties of the hubert-arable adjusted rand index[J]. Psychological Methods, 2004,9(3): 386-396.
[27] XUAN VINH N, EPPS J, BAILEY J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance[J]. The Journal of Machine Learning Research, 2010,11: 2837-2854.

作者简介:

柳菁(1979-),女,实验师,研究方向:计算机图文识别与处理、视频监控、无线传感控制与调度、信息化处理系统的研究与应用,E-mail:maillj@usx.edu.cn。

邱紫滢(2000-),女,本科,研究方向:网络工程、数据挖掘,E-mail:1124633167@qq.com。

郭茂祖(1966-),男,博士,教授,博士生导师,研究方向:机器学习、智慧城市、生物信息,E-mail: guomaozu@bucea.edu.cn。



余冬华(1988-),**通信作者**,男,博士,讲师,研究方向:机器学习、数据挖掘、生物信息,E-mail:donghuayu163@163.com。

(编辑:刘彦东)