运筹学与控制论

DOI:10.11721/cqnuj20220101

基于多目标优化方法的一类 k-Means 自适应算法*

陈美杉，夏丹丹，赵克全
(重庆师范大学 数学科学学院，重庆 401331)

摘要:【目的】针对 k-Means 聚类算法及 MinMax k-Means 聚类算法需要人为提前给定聚类数量而导致数据划分准确率偏低以及 MinMax k-Means 算法聚类效果受类簇边缘点影响较大等不足提出解决方案。【方法】将 k-Means 和 MinMax k-Means 算法的目标函数相结合，建立多目标优化模型，提出基于多目标优化方法的 k-Means 算法。分析簇数异常情况下最小中心方差与最大簇内方差之间的关系。【结果】发现当分类簇数大于最优簇数时，最小中心方差小于最大簇内方差，据此提出了基于多目标优化方法的 k-Means 自适应算法。【结论】数值实验表明：提出的自适应算法在人工数据集和 UCI 标准数据集均具有较好的自适应性且聚类效果较优。

关键词:k-Means 聚类算法 MinMax k-Means 聚类算法 多目标优化 k-Means 自适应算法
中图分类号:O221.6　　**文献标志码**:A　　**文章编号**:1672-6693(2022)01-0027-08

聚类是数据分析中的一个基本问题，它出现在模式识别、机器学习、生物信息学和图像处理等各个领域[1-2]。它的目的是根据一定的聚类目标，将一组实例划分为同构的簇，即簇内相似度高，簇间相似度低。然而穷尽地搜索实例到集群的最优分配在计算上是不可行的，因此研究者们通常去寻求集群目标良好的局部最优解。

k-Means 是研究最充分的聚类算法之一[3]，它最早是由 Steinhaus 于 1955 年提出。该算法在无标记样本的条件下将数据对象进行分组，挖掘数据的潜在结构，是数据分析中常用的一种聚类算法。k-Means 聚类算法被提出已经超过 60 年，它具有容易实施、高效的特点及成功的应用案例和后续不断改进的优点是流行至今的主要原因[4-5]。k-Means 聚类算法虽被广泛接受，但它同时也受到了严重的限制。第一个限制是初始中心的选择对聚类结果的好坏有直接影响；第二个限制是需要提前设定好簇数，不能根据数据进行自适应调整分类簇数[6]。MinMax k-Means[7]算法通过最小化最大聚类误差被广泛应用于解决初始化不良的影响。但 MinMax k-Means 算法的聚类精度受类簇边缘点的影响较大，同时未对 k-Means 聚类算法的第二个限制进行优化。

本文在 MinMax k-Means 算法的基础上，建立了多目标优化模型，通过引入多目标优化问题的线性加权标量化方法，提出多目标 k-Means 算法，有效避免了类簇边缘点对聚类效果的影响，且保留了 MinMax k-Means 算法对初始化的优化。通过分析多目标 k-Means 算法在簇数异常的情况下最小中心方差与最大簇内方差之间的关系，发现当分类簇数大于最优簇数时，最小中心方差小于最大簇内方差，据此提出了多目标 k-Means 的自适应算法。数值实验表明：多目标 k-Means 的自适应算法对 k-Means 聚类算法的两个限制均有较好的改进效果。

1 预备知识

1.1 经典 k-Means 聚类算法

为了将数据集 $X=\{x\}_{i=1}^{N}$，$x_i\in\mathbf{R}^d$ 划分为 M 个不相交的簇 $\{C_k\}_{k=1}^{M}$，k-Means 算法的核心思想是最小化簇内方差之和，即：

$$\min \varepsilon_{\text{sum}}=\min\sum_{k=1}^{M}V_k=\min\Big\{\sum_{k=1}^{M}\sum_{i=1}^{N}\delta_{ik}\,\|x_i-m_k\|^2\Big\}。$$

* **收稿日期**:2021-04-14　**修回日期**:2021-05-31　**网络出版时间**:2022-02-09 13:09
资助项目:国家自然科学基金重大项目(No. 11991024)；国家自然科学基金面上项目(No. 11671062；No. 12171063)；重庆市高校创新研究群体项目(No. CXQT20014)；重庆市自然科学基金项目(No. cstc2021jcyj-msxmX0280)；重庆市教委科学技术研究项目(No. KJQN202100521)
第一作者简介:陈美杉，女，研究方向为最优化理论及其应用，E-mail：cms1997@163.com；**通信作者**:赵克全，男，教授，博士生导师，E-mail：kequanz@163.com
网络出版地址:https://kns.cnki.net/kcms/detail/50.1165.N.20220209.1203.008.html

其中:δ_{ik}是群集指示符变量,若 $x_i=C_k$ 则 $\delta_{ik}=1$,否则 $\delta_{ik}=0$;$m_k=\dfrac{\sum_{i=1}^{N}\delta_{ik}x_i}{\sum_{i=1}^{N}\delta_{ik}}$ 表示第 k 个簇中心;V_k 表示第 k 个簇的方差和。集群通过在将实例分配给它们最近的中心和重新计算中心之间交替进行,直到达到局部最小值。

尽管 k-Means 算法简单快速,但它也有一些缺点,最突出的是解对初始中心选择的依赖性和不具有自适应。错误的初始化可能会导致较差的局部最小值,如图 1 所示。因此会执行多次随机重新启动来绕过初始化问题。通常,重启返回的解决方案在所实现的目标值方面有很大的不同,从好的到非常差的,特别是对于具有大的搜索空间(例如许多群集和维度)的问题。因此,需要多次运行该算法来增加定位良好的局部最小值的可能性。

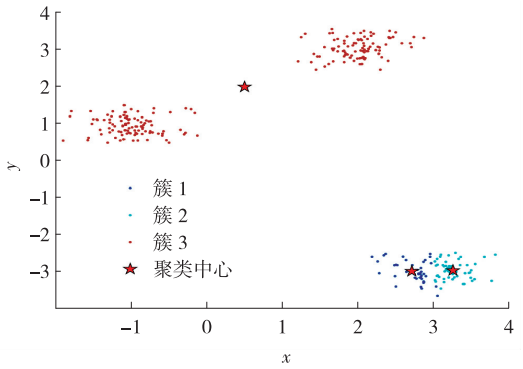

图 1　初始化导致的较差的聚类效果

Fig. 1　Poor clustering effect caused by initialization

1.2 MinMax k-Means 聚类算法

为提高聚类算法的自适应能力和效果,应先解决经典 k-Means 对初始中心的选择的依赖性。本文采用 Tzortzis 在 2014 年提出的 MinMax k-Means 算法[7],该算法通过改变 k-Means 的目标函数来解决 k-Means 初始化问题。MinMax k-Means 算法的核心思想是,将目标函数由 ε_{sum} 替换为最大簇内方差和,然后对目标函数求最小,即:

$$\min \varepsilon_{\max}=\min \max_{1\leqslant k\leqslant M} V_k=\min \max_{1\leqslant k\leqslant M}\left\{\sum_{i=1}^{N}\delta_{ik}\left\|x_i-m_k\right\|^2\right\}。$$

其中:δ_{ik}是群集指示符变量,$m_k=\dfrac{\sum_{i=1}^{N}\delta_{ik}x_i}{\sum_{i=1}^{N}\delta_{ik}}$ 表示第 k 个簇中心,V_k 表示第 k 个簇的方差和。

算法基本原理:在 k-Means 目标函数中,对所有簇的求和允许通过有几个具有较大方差的簇来实现相似的 ε_{sum} 值,这些簇由其他方差较小的簇抵消,或者通过所有簇的中等方差来实现。这意味着没有考虑到集群差异之间的相对差异。但在最小化 $\varepsilon_{\max}$ 时,上述方法并不适用,因为上面的第一种情况会导致更高的目标值。因此,当最小化 $\varepsilon_{\max}$ 时,避免了较大的方差簇,并将解空间限制在具有更多相似方差的簇上。

2 多目标 k-Means 聚类算法

在 MinMax k-Means 中,虽然通过改变目标函数优化了传统 k-Means 对初始中心选择的依赖性,但只考虑最小化 $\varepsilon_{\max}$ 时,簇内的边缘样本可能会对聚类效果产生较差的影响,而最小化 $\varepsilon_{\max}$ 时则能有效避免边缘样本对聚类效果的影响。

为保留传统 k-Means 和 MinMax k-Means 两个算法各自的优点,故建立如下多目标模型:

$$\min\{\varepsilon_{\max},\varepsilon_{\text{sum}}\}。\tag{1}$$

上述多目标模型,既能减少边缘样本对聚类效果的影响,又能解决初始中心的选择对聚类效果的影响。

对多目标模型(1),本文采用线性加权法将多目标模型转化为如下单目标模型:

$$\begin{aligned}&\min \varepsilon_{\text{mul}}\\&\varepsilon_{\text{mul}}=\alpha\varepsilon_{\text{sum}}+(1-\alpha)\varepsilon_{\max}\end{aligned}\tag{2}$$

其中 $\alpha\in[0,1]$。

在 MinMax k-Means 算法的基础上对模型(2)求解,得到多目标 k-Means 算法,具体算法如下。

算法 1　多目标 k-Means

输入　初始化,令 $t=0$,$P_{\text{init}}=0$,初始化权重为:$\boldsymbol{W}_k^{(0)}=\dfrac{1}{M}$,$\forall k=1,\cdots,M$,$P=P_{\text{init}}$,目标权重 ω_1,ω_2。

输出　聚类结果 $\{\delta_{ik}^{(t)}\}_{i=1,\cdots,N,k=1,\cdots,M}$,聚类中心 $\{m_k^{(t)}\}_{k=1}^{M}$ 最优簇数及对应的聚类结果。

步骤 1:计算聚类结果:

$$\delta_{ik}^{(t)}=\begin{cases}1,k=\underset{1<k'\leqslant M}{\operatorname{argmin}}(\boldsymbol{W}_{k'}^{(t-1)})^{p}\|X_i-m_{k'}^{(t-1)}\|^2,\\0,\text{否则}。\end{cases}$$

步骤 2：若有某个聚类集合为空集或单点集，则转步骤 2.1，否则转步骤 2.2。

步骤 2.1：令 $P=P-P_{\text{step}}$，若 $P<P_{\text{init}}$，则使用上一次迭代储存的迭代存储的分类结果和权重：

$$\delta_{ik}^{(t)}=[\boldsymbol{\Delta}^{(p)}]_{ik},\forall k=1,\cdots,M,i=1,\cdots,N;\boldsymbol{W}_k^{(t-1)}=[\boldsymbol{W}^{(p)}]_k,\forall k=1,\cdots,M。$$

更新聚类中心 $m_k^{(t)}=\dfrac{\sum_{i=1}^{N}\delta_{ik}^{(t)}X_i}{\sum_{i=1}^{N}\delta_{ik}^{(t)}}$，令 $t=t+1$，转步骤 3。

步骤 2.2：若 $P<P_{\max}$，且所有聚点集合非空，则将目前的取值存储于矩阵 $\boldsymbol{\Delta}^{(P)}=[\delta_{ik}^{(t)}]$，和向量 $\boldsymbol{W}^{(P)}=[\boldsymbol{W}_k^{(t-1)}]$中，更新 P，$P=P+P_{\text{step}}$，更新权重：

$$\boldsymbol{W}_k^{(t)}=\beta\boldsymbol{W}_k^{(t-1)}+(1-\beta)\left(\frac{(v_k^{(t)})^{\frac{1}{1-P}}}{\sum_{k'=1}^{M}(v_{k'}^{(t)})^{\frac{1}{1-P}}}\right),$$

其中：$v_k^{(t)}=\sum_{i=1}^{N}\delta_{ik}^{(t)}\|X_i-m_k^{(t)}\|^2$，令 $t=t+1$，转步骤 3。

步骤 3：计算目标函数：

$$\varepsilon_{\text{mul}}^{(t)}=\alpha\varepsilon_{\max}+(1-\alpha)\varepsilon_{\text{sum}},$$

若 $|\varepsilon_{\text{mul}}^{(t)}-\varepsilon_{\text{mul}}^{(t-1)}|<\varphi$ 或 $t\geqslant t_{\max}$，则终止算法。

上述算法是在 MinMax k-Means 算法的基础上引入了多目标优化模型，并通过线性加权标量化方法将多目标优化模型转化为单目标问题进行求解。上述多目标 k-Means 算法既能减少边缘点对聚类效果的影响，又能降低初始中心的选择对聚类效果的影响。

3 基于的多目标 k-Means 自适应算法

3.1 分析簇数异常情况

假设数据集最优的聚类簇数为$\overline{M}$，中心之间的最小方差为 $D_{\min}=\min_{\alpha\neq\beta}\|m_\alpha-m_\beta\|^2(\alpha,\beta=1,\cdots,M)$，第 k 个簇的最大簇内方差为 $P_k=\max_{i=1,\cdots,N}\delta_{ik}\|x_i-m_k\|^2,(k=1,\cdots,M)$。则数据集 X 的最大簇内方差 $P_{\max}=\max_{i=1,\cdots,M}P_k$。

聚类簇数异常通常分为两种情况，一是聚类簇数大于数据集最优聚类簇数，即 $M>\overline{M}$；二是聚类簇数小于数据集最优的聚类簇数，即 $M<\overline{M}$。

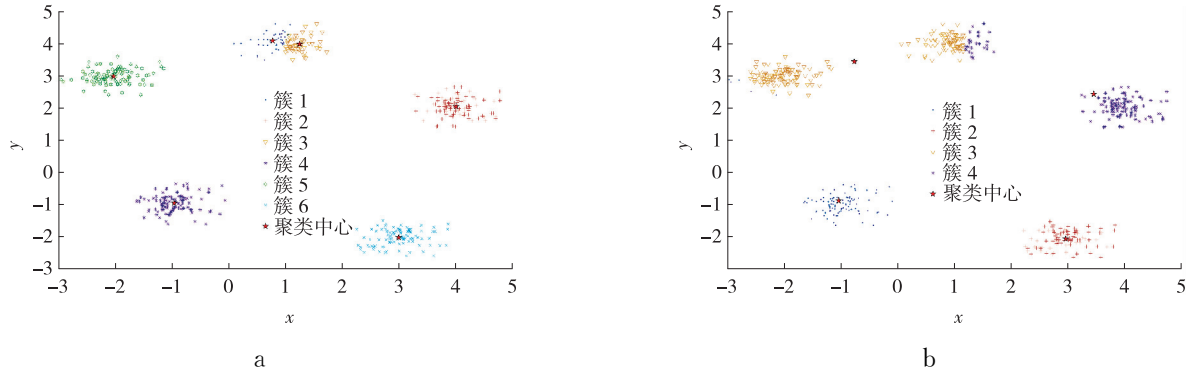

图 2　分类簇数异常时的聚类效果图

Fig. 2　Clustering effect diagram for data set

第一种的簇数异常情况时，以二维的数据集为例，聚类效果如图 2a 所示。可以看出，当 $M>\overline{M}$时，聚类结果容易将应划分一簇的数据划分为一簇或多簇，显然，强行分为一簇或多簇，则有中心之间的最小方差小于数据集的最大簇内方差，即：$D_{\min}\leqslant P_{\max}$。需要注意的是当时，也有 $D_{\min}\leqslant P_{\max}$成立。

第二种的簇数异常情况时,采用与情况一相同的数据集,当$M<\overline{M}$时,聚类效果如图2b所示。可以看出,当$M<\overline{M}$时,聚类结果容易将应新归为一簇的数据分配到已有的一簇或多簇中,此时则有中心之间的最小方差大于数据集的最大簇内方差,即$D_{\min}>P_{\max}$。

上述分析均是以多目标k-Means聚类算法为基础进行,若不能有效解决k-Means聚类算法对初始点的依赖性和边缘点对聚类效果的影响,$M<\overline{M}$时并不能得到$D_{\min}\leqslant P_{\max}$的结果。

3.2 基于多目标k-Means算法改进的自适应算法

根据3.1的分析可知,若以多目标k-Means聚类算法为基础,以初始值为M_0的最少簇数(一般取$M_0=2$),计算中心间的最小方差$D_{\min}$和最大簇内方差$P_{\max}$;若$D_{\min}\leqslant P_{\max}$,则可以试探性的多分一簇;若$D_{\min}>P_{\max}$,此时的簇数刚好比最优簇数大1,即$\overline{M}=M-1$。

具体的多目标自适应k-Means算法(MA k-Means)如下:

算法2 MA k-Means

输入 数据集$X=\{x_i\}_{i=1}^{N}$,初始簇数M_0,算法1相关参数$p_{\max}$,p_{step},β,φ,$t_{\max}$。

输出 最优簇数$\overline{M}$及对应的聚类结果。

步骤0:令$M=M_0$。

步骤1:随机生成初始中心$\{m_k\}_{k=1}^{M}$,运行算法1得到聚类结果。

步骤2:计算方差最小的两个点m',m'',则$D_{\min}=\|m'-m''\|^2$。

步骤3:计算第k个簇的最大簇内方差为$P_k=\max\limits_{i=1,\cdots,N}\delta_{ik}\|x_i-m_k\|^2(k=1,\cdots,M)$,则$P_{\max}=\max\limits_{i=1,\cdots,M}P_k$。

步骤4:若$D_{\min}\leqslant P_{\max}$,则令$M=M+1$,执行步骤1;否则执行步骤5。

步骤5:令$\overline{M}=M-1$,以$\overline{M}$为最优簇数运行算法1得到聚类结果。

上述算法主要是通过对数据集的聚类簇数不断试探,分析簇数不同时聚类效果的差异,即中心之间的最小方差与数据集的最大簇内方差之间的关系,不断向最优的簇数进行逼近,最后达到最优簇数。本算法针对簇间差距明显的数据集有计算时间短、准确率高等优点。

3.3 对簇间差异不明显的改进

为提高算法的准确性,本文针对簇间差异不明显的数据也提出了一种改进的检测机制。该改进机制是判断上述算法的计算结果是否存在将簇间差异不明显的两簇分为一簇的情况。

为方便叙述,本文先引入一个参数,正规化簇间距离[8]$\omega=\dfrac{d_{ij}+d_{ji}}{2\|m_i-m_j\|^2}$,其中$d_{ij}$表示第$i$簇内的样本到第$j$簇中心的最短距离。

显然,$0<\omega\leqslant 1$(当第i,j簇均为单点集时,$\omega=1$);若第i簇与第j簇差异小时,则ω计算公式的分子较小,又由于分母为常量,故ω值较小;若第i簇与第j簇差异大时,则ω计算公式的分子较大,又由于分母为常量,故ω值较大。

MA k-Means算法已经将数据集大致划分,将划分好的每个簇一一进行检测。检测方式为:先按照需求设置精度要求τ,将被检测簇尝试的划分为两簇,计算正规化簇间距离ω,若$\omega>\tau$时,则判定被检测簇未达到要求的分类精度。故将这一簇再进行细分,即最优簇数加一;若$\omega<\tau$时,则判定被检测簇已达到要求的分类精度,最优簇数不变。具体检测机制算法如下:

算法3 MA-Test

输入 阈值$\tau(\leqslant 1)$,MA k-Means算法运行后的簇,以及分类结果。

输出 检测后的最优簇数及对应的聚类结果。

步骤0:令$k=\overline{M}$,$iter=1$,$np=0$。

步骤1:若$iter\leqslant k$,转步骤2,否则执行步骤5。

步骤2:令$iter=iter+1$,将第$iter$簇划通过MA k-Means划分为i,j两簇,分别计算d_{ij}和d_{ji}。

步骤3:计算i,j两簇的正规化簇间距离:

$$\omega=\frac{d_{ij}+d_{ji}}{2\|m_i-m_j\|^2}。$$

步骤4:若$\omega>\tau$,则$np=np+1$,转步骤5。

步骤 5：$\overline{M}=\overline{M}+np$，以$\overline{M}$为最优簇数运行算法 1 得到聚类结果。

上述算法中只需输入精度要求 τ，若 τ 越小，则要求数据集分得更加精细，数据集被分的簇数更多，簇内最大方差更小；若 τ 越大，则要求数据集分得更加粗糙，数据集被分的簇数更少，簇内最大方差更大。当 $\tau=1$ 时，即对 MA k-Means 算法的结果不做调整。即精度要求 τ 根据分类的需求来确定的，故算法 MA-Test 是用来检测 MA k-Means 输出的结果是否达到要求，若未达到要求则继续调整到满足要求。

检测算法主要是对 MA k-Means 算法分类后的数据进行检测，若 MA k-Means 输出的簇达到精度要求，则输出聚类结果，算法停止；若存在某簇未达到精度要求，则将这一簇继续进行聚类划分，直到达到精度要求为止。

4 数值实验

4.1 多目标 k-Means 聚类结果分析

本文采用 4 个常用于测试聚类算法的 UCI 数据集来测试多目标 k-Means 算法的聚类效果。Iris 数据集也称鸢尾花卉数据集，包含 150 个数据样本，分为 3 类，每个数据包含萼片长度、萼片宽度等 4 个属性；Wine 数据集为意大利某地区的 3 种葡萄酒成分数据，包含 178 个数据样本，分为 3 类，每个数据包含 13 个成分特征；seeds 数据集为小麦种子数据集，包含 210 个数据样本，分为 3 类，每个数据包含周长、压实度、籽粒长度等 7 个属性。

所有被测试的算法都从相同的随机选择的初始中心重新运算 20 次。在整个数值实验过程中，簇的数量被设置为等于每个数据集中的类的数量。为了评估返回解的质量，报告了聚类方差和 ε_{sum} 及最大簇内方差和 $\varepsilon_{\max}$，它们在 20 次运行中的平均值。由于所选数据集都有明确的决策属性，本文选取聚类精度（Cluster accuracy，CA）[9] 作为评估指标，公式为：$x_{\text{CA}}=\sum_{i=1}^{c}\frac{|U_i|}{|U|}$。其中：$|U_i|$ 表示第 i 个类簇中被正确分类的对数目，$|U|$ 表示所有聚类对象的数目。当 x_{CA} 的值越高，说明表示聚类标签和类标签之间的匹配度更好。

将 Iris、Wine、seeds 数据集进行归一化，再通过多目标 k-Means 算法（简记为 Mul）与 MinMax k-Means 算法（简记为 MinMax）和传统 k-Means 进行分类测试，测试结果如下：

表 1　各数据集上的聚类效果对比表

Tab. 1　Comparison table of clustering effect on each dataset

数据集	算法	ε_{sum}	$\varepsilon_{\max}$	x_{CA}/%	运行时间/s
Iris	Mul	11.068 8	4.352 2	88%	0.509 3
	MinMax	14.885 8	3.686 9	86.67%	0.067 7
	k-Means	10.987 8	7.801 8	88%	0.031 6
Wine	Mul	59.380 3	13.936 4	92.70%	0.400 7
	MinMax	67.907 6	18.811 1	91.57%	0.156 5
	k-Means	58.892 6	25.787 4	87.64%	0.140 1
seeds	Mul	23.328 2	6.520 8	90.48%	0.361 8
	MinMax	29.933 4	7.515	88.57%	0.141 5
	k-Means	22.911 4	10.683 9	86.67%	0.037

由表 1 不难看出，多目标 k-Means 算法的 x_{CA} 明显优于其他 3 种算法；在运算时间上看，k-Means 算法由于过程简单，所需时间最短，但所有算法运行时间均不超过 1 秒，对实际需求影响不大；在 ε_{sum} 和 $\varepsilon_{\max}$ 这两个指标上，多目标 k-Means 算法介于 MinMax k-Means 算法和传统 k-Means 之间。

4.2 最佳类簇数目选取测试分析

首先对簇间差异大的数据进行实验，人工生成包含 500 个对象的正态分布数据集 1，参数设置如下：

$$\alpha_1=\alpha_2=\alpha_3=\frac{1}{3},\boldsymbol{\mu}_1=(0\quad 0)^{\mathrm{T}},\boldsymbol{\mu}_2=(7\quad 0)^{\mathrm{T}},\boldsymbol{\mu}_3=(14\quad 0)^{\mathrm{T}},\boldsymbol{\Sigma}_1=\boldsymbol{\Sigma}_2=\boldsymbol{\Sigma}_3=\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}。$$

图 3a 为原始数据，可见生成的 500 个数据对象形成边界分明的 3 个类簇，且簇间差异明显。

运行 MA k-Means 算法后结果如图 3b 所示。

当 $M=3$ 时，簇内最大方差为 $P_{\min}=3.154\ 8$，中心之间的最小方差为 $D_{\min}=7.035\ 8$，此时有 $P_{\min}<D_{\min}$ 成

立;当 $M=4$ 时,簇内最大方差为 $P_{\min}=2.538\ 1$,中心之间的最小方差为 $D_{\min}=1.823\ 4$,此时有 $P_{\min}>D_{\min}$ 成立;即 $M=4$ 时,此时已经将适合分到一簇的数据集分为了两类,故最优分类簇数为 $\overline{M}=3$。由于数据集 1 的簇间差异较大,并不需运行检测算法 MA-test。

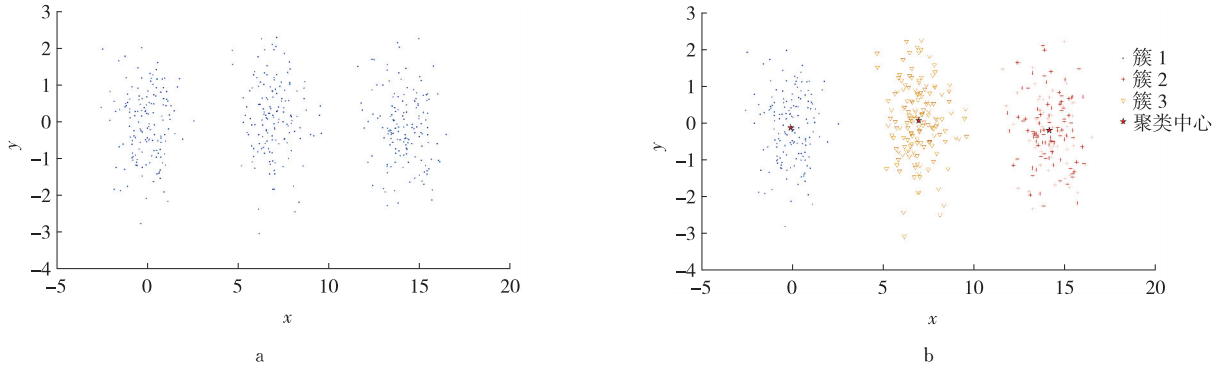

图 3　MA k-Means 算法在人工数集 1 上的聚类效果图

Fig. 3　Clustering effect Diagram of MA k-Means algorithm on artificial data set 1

下面对簇间差异大的数据进行实验,人工生成包含 360 个对象的正态分布数据集 2,由 6 个类簇组成,且类簇之间有交叉区域. 参数设置如下所示:

$$\alpha_k=\frac{1}{6},\boldsymbol{\mu}_1=(1\ 3)^{\mathrm{T}},\boldsymbol{\mu}_2=(3\ 2)^{\mathrm{T}},\boldsymbol{\mu}_3=(3\ 7)^{\mathrm{T}},\boldsymbol{\mu}_4=(5\ 5)^{\mathrm{T}},\boldsymbol{\mu}_5=(7\ 8)^{\mathrm{T}},\boldsymbol{\mu}_6=(9\ 6)^{\mathrm{T}},$$

$$\boldsymbol{T}_k=\begin{pmatrix}0.4 & 0\\ 0 & 0.4\end{pmatrix},k=1,2,\cdots,6。$$

图 4a 为原始数据,生成 600 个数据对象形成边界模糊的 6 个类簇. 且以 $\boldsymbol{\mu}_1$ 与 $\boldsymbol{\mu}_2$ 为中心的两个正态分布的交叉面积较大;以 $\boldsymbol{\mu}_3$ 与 $\boldsymbol{\mu}_4$ 为中心的两个正态分布与以 $\boldsymbol{\mu}_5$ 与 $\boldsymbol{\mu}_6$ 为中心的两个正态分布情况相同,存在交叉面积,但交叉面积较小。

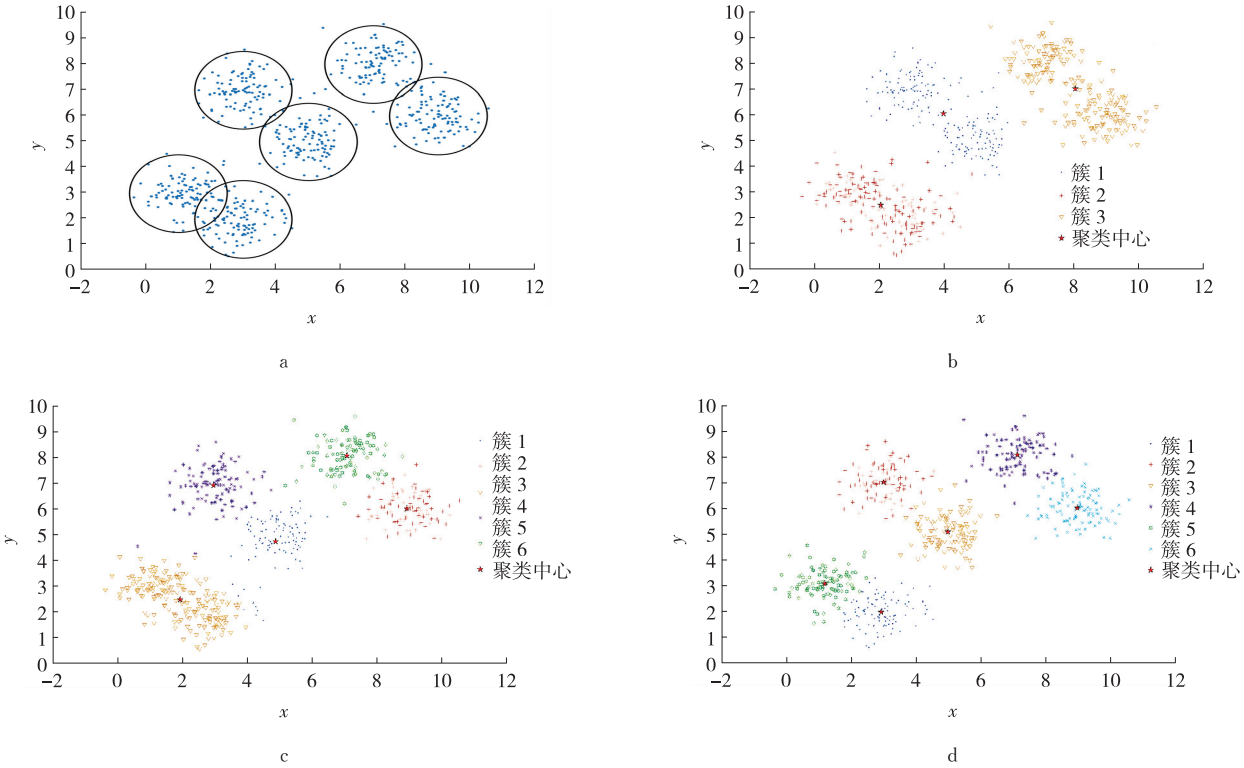

图 4　MA k-Means 算法在人工数集 2 上的聚类效果图

Fig. 4　Clustering effect diagram of MA k-Means algorithm on artificial data set 2

直接运用 MA k-Means 算法进行对人工数据集 2 进行分类，输出的最优分类簇数为 3，分类结果如图 4b 所示；若对人工数据集 2 先运用 MA k-Means 算法再用算法 3 测试结果，当 $0.07 \leqslant \tau \leqslant 0.23$ 时，算法输出的最优分类簇数为 5，分类结果如图 4c 所示；当 $0 < \tau \leqslant 0.07$ 时，算法输出的最优分类簇数为 6，分类结果如图 4d 所示。从图 4 可以看出，对于人工数据集 2，虽然不同取值的 τ 对应的输出结果不同，但 3 种结果均具有实际意义。

将 MA k-Means 和检测算法结合与常见的几种自适应算法 RCA、OB、CSDP[10-12] 在人工数据集和 UCI 的标准数据集上的计算结果，如下所示：

表 2　各个算法的计算结果对照表

Tab. 2　Comparison table of calculation results of each algorithm

数据集	真实聚类数目	MA k-Means	RCA	OB	CSDP
人工数据集 1	3	3	3,4	3,4	3
人工数据集 2	6	3,5,6	4,5,6,7,8,9,10	4,6	2,3,5,6
Iris	3	3	2,3	2,3	2
Wine	3	2	2,3,4	2,3	2,3,4
seeds	3	3	2,3,4	2	2,3,4

由表 2 不难看出，多目标自适应 k-Means 算法（MA k-Means）计算效果表现最佳。虽然 MA k-Means 计算出数据集 Wine 的最优簇数与真实聚类数目不相同，可能是由于数据集 Wine 的维数较高，故 MA k-Means 在高维数据中的自适应效果还有待提高。经过数据测试表明，MA k-Means 对低维数据和簇间差异明显的数据集有很好的计算结果。

5 结论

在对 k-Means 聚类算法及相关衍生算法的研究中，类簇数目的确定和初始类簇中心的选取直接影响聚类精度和算法的收敛速度，本文将传统 k-Means 和 MinMax k-Means 算法的目标函数相结合建立多目标 k-Means 算法，并在此基础上研究最大簇内方差间、中心间的最小方差与最优簇数间的关系，提出了多目标 k-Means 的类簇自适应算法。数值实验表明，在低维数据集中，该算法有效提高聚类精度且有较好得自适应效果。如何提高该算法在高维数据中的自适应效果将是下一步拟开展的研究工作。

参考文献：

[1] XU R，D C，W LI. Survey of clustering algorithms[J]. IEEE Transactions on Neural Networks，2005，16(3)：645-678.

[2] FILIPPONE M，CAMASTRA F，MASULLI F，et al. A survey of kernel and spectral methods for clustering[J]. Pattern Recognition，2008，41(1)：176-190.

[3] LLOYD S P. Least square quantization in PCM[J]. IEEE Transactions on Information Theory，1982，28(2)：129-136.

[4] SCHOLKOPF B，SMOLA A，MULLER K. Nonlinear component analysis as a kernel eigenvalue problem[J]. Neural computation，1998，10(5)：1299-1319.

[5] PENA J M，LOZANO J A，LARRANAGA P. An empirical comparison of four initialization methods for the k-Means algorithm [J]. Pattern Recognition Letters，1999，20(10)：1027-1040.

[6] CELEBI M E，KINGRAVI H A，VELA P A. A comparative study of efficient initialization methods for the k-means clustering algorithm[J]. Expert Systems with Applications：An International Journal，2013，40(1)：200-210.

[7] TZORTZIS G，LIKAS A. The MinMax k-Means clustering algorithm[J]. Pattern Recognition，2014，47(7)：2505-2516.

[8] GHOSH A，MISHRA N S，GHOSH S. Fuzzy clustering algorithms for unsupervised change detection in remote sensing images [J]. Information Sciences，2011，181(4)：699-715.

[9] FAHAD A，ALSHATRI N，TARI Z，et al. A Survey of clustering algorithms for big data：taxonomy and empirical analysis[J]. IEEE Transactions on Emerging Topics in Computing，2014，2(3)：267-279.

[10] FRIGUI H，KRISHNAPURAM R. A robust competitive clustering algorithm with applications in computer vision[J]. Pattern Analysis & Machine Intelligence IEEE Transactions on，1999，21(5)：450-465.

[11] FAZENDEIRO P，DE OLIVEIRA J V. Observer-biased fuzzy clustering[J]. IEEE Transactions on Fuzzy Systems，2015，23

(1):85-97.

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

Operations Research and Cybernetics

A Class of *k*-Means Adaptive Algorithms Based on Multi-Objective Optimization Method

CHEN Meishan, XIA Dandan, ZHAO Kequan

(School of Mathematical Science, Chongqing Normal University, Chongqing 401331, China)

Abstract: [**Purposes**]In view of the shortcomings of *k*-Means clustering algorithm and MinMax *k*-Means clustering algorithm, which need to be artificially given the number of clusters in advance, the accuracy of data division is low, and the clustering effect of MinMax *k*-Means algorithm is greatly affected by the edge points of clusters. [**Methods**]The multi-objective optimization model is established by combining the objective function of *k*-Means and MinMax *k*-Means algorithm, and the *k*-Means algorithm based on multi-objective optimization method is proposed. Analyze the relationship between the minimum central variance and the maximum intra-cluster variance in the case of abnormal cluster number. [**Findings**]It is found that when the number of classified clusters is greater than the optimal number of clusters, the minimum central variance is less than the maximum intra-cluster variance. Based on this, a *k*-Means adaptive algorithm based on multi-objective optimization method is proposed. [**Conclusions**]Numerical experiments show that the adaptive algorithm proposed here has good adaptability and clustering effect in both artificial dataset and UCI standard dataset.

Keywords: *k*-Means clustering algorithm; MinMax *k*-Means clustering algorithm; multi-objective optimization; *k*-Means adaptive algorithm

(责任编辑 陈 乔)