计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(7):34−42 [doi: 10.15888/j.cnki.csa.006444]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于犹豫模糊语言术语集的正交模糊聚类算法①

王慧冰, 林铭炜, 姚志强

(福建师范大学 数学与信息学院, 福州 350117)
通讯作者: 林铭炜, E-mail: linmwcs@163.com

摘　要: 犹豫模糊语言术语集 (Hesitance Fuzzy Linguistic Term Sets, HFLTSs) 允许决策者们用几个可能的语言术语来评估一个属性. 近来, 采用 HFLTSs 来进行模糊聚类分析的问题越来越受关注. 考虑到目前基于 HFLTSs 的模糊聚类算法还存在计算复杂度高的问题, 提出了一种新的正交模糊聚类算法: 首先计算样本之间的距离测度得到距离测度矩阵, 接着计算其等价矩阵; 然后确定置信水平值, 通过置信水平值对等价矩阵进行切割; 最后根据切割矩阵的列向量之间的正交关系来确定对应样本是否可以放在同一个类别, 以此得到聚类结果. 该算法步骤简单, 计算复杂度低, 并且适合于数据量大的模糊聚类问题. 本文末尾将通过一个实例结合 k-means 聚类算法证明该算法的可行性和高效性.

关键词: 犹豫模糊语言术语集; 距离测量; 犹豫度; 正交模糊聚类算法; k-means 聚类

引用格式: 王慧冰,林铭炜,姚志强.基于犹豫模糊语言术语集的正交模糊聚类算法.计算机系统应用,2018,27(7):34–42. http://www.c-s-a.org.cn/1003-3254/6444.html

Novel Orthogonal Fuzzy Clustering Algorithm Based on Hesitance Fuzzy Linguistic Term Sets

WANG Hui-Bing, LIN Ming-Wei, YAO Zhi-Qiang

(College of Mathematics and Informatics, Fujian Normal University, Fuzhou 350117, China)

Abstract: Hesitance Fuzzy Linguistic Term Sets (HFLTSs) allow decision makers to evaluate a property in several possible linguistic terms. Recently, HFLTSs based fuzzy clustering analysis draws increasing attention. Considering that the current fuzzy clustering algorithm based on HFLTSs still costs large computation, this study proposes a novel orthogonal fuzzy clustering algorithm. Firstly, calculate the distance measures between samples to construct distance measure matrix, and then calculate the matrix's equivalent matrix. Secondly, cut the equivalent matrix according to its confidence level to obtain the corresponding cutting matrix. Finally, obtain the clustering result based on the orthogonal relationship between the column vectors of the cutting matrix. This algorithm has simple steps and low computational complexity. It is also suitable for large-scale fuzzy clustering problems. At last, the feasibility and efficiency of this algorithm are proved by a practical application with k-means clustering algorithm.

Key words: Hesitance Fuzzy Linguistic Term Sets (HFLTSs); distance measure; hesitance; orthogonal fuzzy clustering algorithm; k-means algorithm

聚类算法已经在经济学, 计算机科学, 天文学等各个领域得到广泛应用[1,2]. 传统的聚类算法是根据准确的数值对确定的对象进行划分的, 但是随着社会的进步, 模糊数据、糊模模型成为了一种新的趋势, 这意味着传统的硬划分聚类方法也要逐渐转向软划分聚类方法[3]. 研究模糊聚类的前提是要引入模糊集理论, 因为模糊聚类是基于模糊集进行划分的. Zadeh[4]首先引入模糊语言学理论, 然后将模糊集应用于多标准决策

① 基金项目: 国家自然科学基金 (61502102); 福建省自然科学基金 (2016J05149)
Foundation item: National Natural Science Foundation of China (61502102); Natural Science Foundation of Fujian Province of China (2016J05149)
收稿时间: 2017-11-20; 修改时间: 2017-12-15; 采用时间: 2017-12-20; csa 在线出版时间: 2018-05-24

(MCDM) 问题中, 称之为模糊 MCDM. 之后, Torra[5]提出了犹豫模糊集 (HFSs), 它允许使用多个属于[0, 1]范围的值来评估一个属性, 增强了模糊性. 然而, 在实际问题中, 我们更多的时候得到的数据是定性信息, 不是定量值[6,7]. 例如, 当人们评估汽车的性能时, 他们可能会更偏向于使用“差”, “好”, “非常好”等语言术语来表达他们的评估结果. 因此, Zadeh 提出了采用模糊语言学方法对评估信息进行建模的思想, 最典型的模型有: 二类模糊集合模型[8], 二元语言模型[9]和虚拟语言模型[10]. 这些语言模型的缺陷是: 它们要求一个对象的一个属性只能对应一个语言术语[11]. 基于犹豫模糊集思想和模糊语言学方法, Rodríguez 等人[12]提出了 HFLTSs 的概念, 它允许一个对象的一个属性可以用多个语言术语来描述, 提高了评估属性的灵活性.

目前已经存在许多关于模糊聚类的研究, 比如, 文献[13]和文献[14]提出了基于直觉模糊集 (IFSs) 的聚类方法; 文献[15]提出了犹豫模糊环境下的最小生成树 (MST) 聚类方法; 文献[16]通过计算犹豫模糊集的相关系数得到相关系数矩阵, 然后构造相关系数矩阵的等价矩阵, 最后, 基于 λ 置信值切割矩阵得到聚类结果; 文献[17]提出了一种层次犹豫模糊 k-means 聚类方法, 以层次聚类的结果作为 k-means 的初始聚类中心进行迭代以获得最终聚类结果, 该算法减少了 k-means 的迭代次数, 计算成本和聚类时间; 近期, 文献[18]将文献[16]的方法扩展到犹豫模糊积性集 (HMSs) 上使用, 并取得了一定的成果; 文献[19]则提出了一种基于犹豫模糊环境下的正交聚类算法. 但是目前还没有比较成熟的基于 HFLTSs 的聚类方法, 而 HFLTSs 在实际应用中较 HFSs、IFSs 及 HMSs 的使用更加广泛且灵活性更大, 因此, 本文针对 HFLTSs 提出了一种新的正交模糊聚类算法.

1 理论基础

1.1 犹豫模糊语言术语集

定义 1[12]. 设$\mathrm{S}=\{s_i|i=-\tau,\cdots,-1,0,1,\cdots,\tau\}$是给定的一个语言术语集, 一个 HFLTS, $\mathrm{H_S}$, 指的是S上有限个连续的语言术语的有序子集, 表示为:

$$\mathrm{H_S}=\{s_i,s_{i+1},\cdots,s_j\}$$

其中, $s_k\in\mathrm{S}$, k属于$\{-\tau,\cdots,-1,0,1,\cdots,\tau\}$.

HFLTSs 的上下边界分别是$s_{-\tau}$和s_τ, S满足以下特征:

(1) 如果$\alpha>\beta$, 则$s_\alpha>s_\beta$;

(2) S满足负运算操作: $neg(s_\alpha)=s_{-\alpha}$, 除了$neg(s_0)=s_0$.

备注 1. 在定义 1 中, HFLTSs 是一些离散的数值, 为了避免丢失语言信息, 可以将离散形式扩展为连续形式, 即, $\bar{\mathrm{S}}^*=\{s_\alpha|\alpha\in[-q,q]\}$.

1.2 上下文无关语法

文献[20]提出了一种上下文无关文法, 我们可以将一些简单而丰富的语言表达通过转换函数[21]转换成 HFLTSs.

定义 2. 假设E_{G_H}表示将语言表达转换成 HFLTSs 的转换函数, G_H表示上下文无关方法, S是语言术语集. 通过G_H将S_{ll}转换成$\mathrm{H_S}$的表达式如下:

$$E_{G_\mathrm{H}}:\mathrm{S}_{ll}\to\mathrm{H_S}$$

具体转换过程如下:

(1) $E_{G_\mathrm{H}}(s_t)=\{s_t|s_t\in\mathrm{S}\}$;

(2) E_{G_H}(至多为s_i)$=\{s_t|s_t\in\mathrm{S},s_t\le s_i\}$;

(3) E_{G_H}(小于s_i)$=\{s_t|s_t\in\mathrm{S},s_t<s_i\}$;

(4) E_{G_H}(至少为s_i)$=\{s_t|s_t\in\mathrm{S},s_t\ge s_i\}$;

(5) E_{G_H}(大于s_i)$=\{s_t|s_t\in\mathrm{S},s_t>s_i\}$;

(6) E_{G_H}(介于s_i和s_j之间)$=\{s_t|s_t\in\mathrm{S},s_i\le s_t\le s_j\}$.

例 1. S={极差, 很差, 差, 一般, 好, 很好, 极好}作为一本书的语言术语集, 假设一位评估者给出的对三本书的三个属性的评估结果如下:

$$\text{评估者}=\begin{pmatrix}\text{一般} & \text{介于好和很好之间} & \text{非常好}\\ \text{顶多为差等级} & \text{在好之上} & \text{好}\\ \text{至少评为好} & \text{介于很坏和坏之间} & \text{坏}\end{pmatrix}$$

通过E_{G_H}转换为 HFLTSs 的形式之后如下:

$$\text{评估者}=\begin{pmatrix}\{\text{一般}\} & \{\text{好, 很好}\} & \{\text{非常好}\}\\ \{\text{极差, 很差, 差}\} & \{\text{很好, 极好}\} & \{\text{好}\}\\ \{\text{好, 很好, 极好}\} & \{\text{很坏, 坏}\} & \{\text{坏}\}\end{pmatrix}$$

2 基于 HFLTSs 的距离测度

距离测度是聚类分析的重要指标之一[22], 本节将介绍基于 HFLTSs 的传统距离测度以及改进之后的距离测度.

2.1 传统距离测度

定义 3[23]. 设$\mathrm{S}=\{s_i|i=-\tau,\cdots,-1,0,1,\cdots,\tau\}$是一个语言术语集, $\mathrm{H_S}^1=\left\{s_{\delta_l^1}|l=1,\cdots,|\mathrm{H_S}^1|\right\}$和$\mathrm{H_S}^2=\{s_{\delta_l^2}|l=1,\cdots,|\mathrm{H_S}^2|\}$是S上的任意两个 HFLTSs, δ_l表示$\mathrm{H_S}$中每一个语言术语的下标, $|\mathrm{H_S}^1|$指的是$\mathrm{H_S}^1$中

的犹豫模糊语言术语元素(HFLTEs)个数, $|H_S^1|=|H_S^2|=L$. 则H_S^1和H_S^2之间的距离测度为:

$$d_{gd}(H_S^1,H_S^2)=\left(\frac{1}{L}\sum_{l=1}^{L}\left(\frac{|\delta_l^1-\delta_l^2|}{2\tau+1}\right)^{\lambda}\right)^{1/\lambda} \tag{1}$$

当$\lambda=1$时, H_S^1和H_S^2的汉明距离如下:

$$d_{hd}(H_S^1,H_S^2)=\frac{1}{L}\sum_{l=1}^{L}\frac{|\delta_l^1-\delta_l^2|}{2\tau+1} \tag{2}$$

当$\lambda=2$时, H_S^1和H_S^2的欧式距离如下:

$$d_{ed}(H_S^1,H_S^2)=\left(\frac{1}{L}\sum_{l=1}^{L}\left(\frac{|\delta_l^1-\delta_l^2|}{2\tau+1}\right)^{2}\right)^{1/2} \tag{3}$$

传统距离测度公式要求两个 HFLTSs 的 HFLTEs 个数一样, 而实际上如例 1 所示, 两个不同的 HFLTSs 的 HFLTEs 个数可能不同. 因此, 传统距离测度采用最大值、最小值或者平均值来补齐 HFLTEs 个数较少的 HFLTSs, 使 HFLTEs 的个数一致[24].

例 2. 设$S=\{s_{-3},s_{-2},s_{-1},s_0,s_1,s_2,s_3\}$是一个语言术语集, $H_S^1=\{s_1,s_2\}$和$H_S^2=\{s_{-1},s_0,s_1\}$是S上的两个 HFLTSs. 为了计算H_S^1和H_S^2之间的距离测度, 需要保证两者的 HFLTEs 个数一样. 本文采用平均值作为补齐元素, 即将$H_S^1=\{s_1,s_2\}$扩展为$H_S^1=\{s_1,s_{1.5},s_2\}$. 计算$H_S^1$和$H_S^2$的欧式距离如下:

$$d_{ed}(H_S^1,H_S^2)=\sqrt{\frac{1}{3}\left(\left(\frac{|1-(-1)|}{7}\right)^2+\left(\frac{|1.5-0|}{7}\right)^2+\left(\frac{|2-1|}{7}\right)^2\right)}=0.222$$

传统距离测度方法, 涉及到有多个 HFLTSs 时, 是对这些 HFLTEs 个数进行两两对比, 得到距离测度.

例 3. 设$S=\{s_{-3},s_{-2},s_{-1},s_0,s_1,s_2,s_3\}$为一个语言术语集,$H_S^1=\{s_1,s_2\}$、$H_S^2=\{s_{-1},s_0,s_1\}$和$H_S^3=\{s_1,s_2,s_3,s_4\}$是S上的三个 HFLTSs.

计算$d(H_S^1,H_S^2)$和$d(H_S^1,H_S^3)$时, 要分别将H_S^1扩展成$H_S^1=\{s_1,s_{1.5},s_2\}$和$H_S^1=\{s_1,s_{1.5},s_{1.5},s_2\}$. 这意味着, $d(H_S^1,H_S^2)$是三维空间下的距离测度, 而$d(H_S^1,H_S^3)$则是在四维空间下计算得到的距离测度, 显然, 将两者进行对比是无意义的.

2.2 新型距离测度

针对上面提到的传统距离测度存在的缺陷, 本文对其做出改进, 重新定义如定义 4.

定义 4. 设$S=\{s_i|i=-\tau,\cdots,-1,0,1,\cdots,\tau\}$是一个语言术语集, $H_S^1=\left\{s_{\delta_l^1}|l=1,\cdots,|H_S^1|\right\}$和$H_S^2=\left\{s_{\delta_l^2}|l=1,\cdots,|H_S^2|\right\}$是S上的任意 HFLTSs, δ_l表示H_S中每一个语言术语的下标, $|H_S^1|$指的是H_S^1中的 HFLTEs 个数, $|H_S^1|=|H_S^2|$. H_S^1和H_S^2的距离测度为:

$$d_{gd}(H_S^1,H_S^2)=\left(\frac{1}{L}\sum_{l=1}^{L}\left(\frac{|\delta_l^1-\delta_l^2|}{2\tau+1}\right)^{\lambda}\right)^{1/\lambda} \tag{4}$$

L表示需要进行对比的所有 HFLTSs 中 HFLTEs 个数最多的 HFLTSs 的长度.

例 4. 设$S=\{s_{-3},s_{-2},s_{-1},s_0,s_1,s_2,s_3\}$为一个语言术语集,$H_S^1=\{s_1,s_2\}$、$H_S^2=\{s_{-1},s_0,s_1\}$和$H_S^3=\{s_1,s_2,s_3,s_4\}$是S上的三个 HFLTSs.

计算$d(H_S^1,H_S^2)$和$d(H_S^1,H_S^3)$时, H_S^1和H_S^2都要扩展成具有四个元素的 HFLTSs. 即H_S^1扩展成$H_S^1=\{s_1,s_{1.5},s_{1.5},s_2\}$,$H_S^2$扩展成$H_S^2=\{s_{-1},s_0,s_0,s_1\}$.

2.3 考虑犹豫度的新型距离测度

HFLTSs 的传统距离测量只考虑了 HFLTEs 的值的差异, 而不考虑 HFLTEs 的个数差异. 文献[25]在距离测度中考虑到了犹豫度这个影响因素, 提高了计算 HFSs 的距离测度的准确性和可靠性. 文献[26]受此启发, 也在 HFLTSs 的距离测度公式中考虑犹豫度对其的影响, 提出了新的距离测度公式.

定义 5[20]. 设$S=\{s_i|i=-\tau,\cdots,-1,0,1,\cdots,\tau\}$是一个语言术语集, $H_S=\left\{s_{\delta_l}|l=1,\cdots,|H_S|\right\}$是S上任意的一个 HFLTSs, 则$H_S$的犹豫度定义为:

$$\mu(H_S)=\frac{\frac{3}{|H_S|}\sum_{l=1}^{|H_S|}(\delta_l-\bar{\delta})^2}{\tau(\tau+1)} \tag{5}$$

其中, $\bar{\delta}=\frac{1}{|H_S|}\sum_{l=1}^{|H_S|}\delta_l$, δ_l表示H_S中每一个语言术语的下标.

定义 6[25]. 设$S=\{s_i|i=-\tau,\cdots,-1,0,1,\cdots,\tau\}$是一个语言术语集,$H_S^1=\left\{s_{\delta_l}|l=1,\cdots,|H_S^1|\right\}$和$H_S^2=\left\{s_{\delta_l}|l=1,\cdots,|H_S^2|\right\}$是S上任意两个 HFLTSs, 定义$H_S^1$和$H_S^2$的距离测度公式为:

$$d_{dg}(H_S^1,H_S^2)=\left(\alpha|\mu(H_S^1)-\mu(H_S^2)|^{\lambda}+\beta\left(\frac{1}{L}\sum_{l=1}^{L}\left(\frac{|\delta_l^1-\delta_l^2|}{2\tau+1}\right)^{\lambda}\right)\right)^{1/\lambda} \tag{6}$$

其中, α和β分别表示犹豫度和 HFLTEs 个数差异所占权重, $\alpha+\beta=1$. 当$\alpha=0$, 即不考虑犹豫度的影响, 该公式等价于传统距离测度公式; 本文主要考虑两个权重相等的情况, 即$\alpha=\beta=0.5$.

将上述距离测度扩展到多个属性的情况, 则定义为如定义 7 所示形式.

定义 7. 设$S=\{s_i|i=-\tau,\cdots,-1,0,1,\cdots,\tau\}$是一个语言术语集, $X=\{x_1,x_2,\cdots,x_n\}$表示n个属性, 取任意两个HFLTSs, $H_S^1(x_i)=\cup_{s_{\delta_l^1}\in H_S^1}\left\{s_{\delta_l^1}|l=1,\cdots,|H_S^1|\right\}$和$H_S^2(x_i)=\cup_{s_{\delta_l^2}\in H_S^2}\left\{s_{\delta_l^2}|l=1,...,|H_S^2|\right\}$, 其中$L(x_i)=\max\limits_{H_S^l(x_i)\in H_S(x_i)}\left\{|H_S^l(x_i)|\right\}$, 则$H_S^1$和$H_S^2$的标准距离测度公式为:

$$d_{ngd}(H_S^1,H_S^2)=\frac{1}{n}\sum_{i=1}^{n}\left(\alpha|\mu(H_S^1(x_i))-\mu(H_S^2(x_i))|^{\lambda}+\beta\left(\frac{1}{L(x_i)}\sum_{l=1}^{L(x_i)}\left(\frac{|\delta_l^{1i}-\delta_l^{2i}|}{2\tau+1}\right)^{\lambda}\right)\right)^{1/\lambda} \tag{7}$$

其中, $\lambda\geq 1$, $0\leq\alpha,\beta\leq 1$, 且$\alpha+\beta=1$.

3 基于 HFLTSs 的正交模糊聚类算法

3.1 基于犹豫模糊环境的正交聚类算法

近来, 文献[19]提出了基于 HFSs 的正交聚类算法, 简化了聚类过程, 降低了算法复杂度, 提高了算法的效率. 该算法的步骤如下.

算法 1. 基于 HFSs 正交模糊聚类算法.

步骤 1. 设$\{A_1,A_2,\cdots,A_n\}$是对应$X=\{x_1,x_2,\cdots,x_m\}$的一系列 HFSs, $X=\{x_1,x_2,\cdots,x_m\}$指 m 个样本. 计算样本之间的距离得到距离测度矩阵$M=\{d_{ij}\}_{n\times n}$, 其中$d_{ij}=d(A_i,A_j)$.

步骤 2. 选择置信水平λ值, $\lambda\in[0,1]$, 构建距离测度矩阵$M=\{d_{ij}\}_{n\times n}$的对应λ–切割矩阵, M_λ. 具体过程: 从$M=\{d_{ij}\}_{n\times n}$矩阵中按照从大到小的顺序选择λ值, 然后对$M=\{d_{ij}\}_{n\times n}$进行λ–切割, 大于λ值的置为 1, 小于λ值的置为 0, 得到对应的λ–切割矩阵, M_λ.

步骤 3. M_λ的每一列看作一个向量, 表示为$M_\lambda=(\bar{\alpha}_1,\bar{\alpha}_2,\cdots,\bar{\alpha}_n)$, 其中$\bar{\alpha}_j=(\alpha_{1j},\alpha_{2j},\cdots,\alpha_{nj})^{\mathrm{T}}$. 对任意两个列向量进行内积点乘运算$(\bar{\alpha}_i,\bar{\alpha}_j)=\bar{\alpha}_i^{\mathrm{T}}\bar{\alpha}_j$, 如果$(\bar{\alpha}_i,\bar{\alpha}_j)=\bar{\alpha}_i^{\mathrm{T}}\bar{\alpha}_j=0$, 则认为这两个列向量是正交关系.

步骤 4. 根据列向量之间的正交关系对样本进行聚类, 具体原理如下:

如果$(\bar{\alpha}_i,\bar{\alpha}_j)\neq 0$, 则将样本$A_i$和$A_j$归为同一类样本, 称之为直接聚类原理.

如果存在$1\leq n_1,n_2,\cdots,n_s\leq n$, 且$(\bar{\alpha}_i,\bar{\alpha}_{n_1})(\bar{\alpha}_{n_1},\bar{\alpha}_{n_2})\cdots(\bar{\alpha}_{n_s},\bar{\alpha}_j)\neq 0$, 则将样本$A_i$和$A_j$归为同一类样本, 称之为间接聚类原理.

如果$(\bar{\alpha}_i,\bar{\alpha}_j)=0$, 并且对任意$1\leq n_1,n_2,\cdots,n_s\leq n$, 满足$(\bar{\alpha}_i,\bar{\alpha}_{n_1})(\bar{\alpha}_{n_1},\bar{\alpha}_{n_2})\cdots(\bar{\alpha}_{n_s},\bar{\alpha}_j)=0$, 则样本$A_i$和$A_j$不能归为同个类别.

为了说明计算的复杂性, 文献[19]随机生成一些 HFSs 用以对比正交模糊聚类算法和模糊网络聚类算法. 表 1 是两种聚类方法得到聚类结果之前的运行时间, 显然, 正交模糊聚类算法消耗更少的时间.

表 1 运行时间对比 (单位: s)

样本个数	6	10	15	20
正交模糊	0.000 164	0.000 523	0.001 056	0.001 619
模糊网络	0.000 205	0.000 852	0.001 431	0.002 057

但是该算法存在一个缺陷, 当样本数量大时, 会得到一个非常高维的距离测度矩阵, 如果矩阵中的所有不相同的值都作为置信水平对距离测度矩阵进行λ–切割, 则需要消耗大量的计算成本, 且其中存在很多重复操作, 因此本文对该算法做出了改进.

3.2 基于 HFLTSs 的正交 k-means 聚类方法

针对算法 1 存在的问题, 如果我们可以解决样本数量大带来的高维矩阵难以计算的问题, 那么就可以进一步降低计算复杂度. 本文采取的解决方法是减少距离测度矩阵内部元素的差异性, 以此缩小置信水平λ的取值空间, 具体原理是采用构造等价矩阵[14](等价矩阵的概念将直接体现在算法步骤中), 替代原始距离测度矩阵, 在等价矩阵的基础上进行正交聚类. 后期, 为了证明该算法的可行性和高效性, 还将通过 k-means 算法对聚类结果进行验证.

基于 HFLTSs 的正交模糊聚类算法过程如算法 2.

算法 2. 基于 HFLTSs 正交模糊聚类算法.

步骤 1. 设$\{A_1,A_2,\cdots,A_n\}$是对应$X=\{x_1,x_2,\cdots,x_m\}$的一系列 HFLTSs, $X=\{x_1,x_2,\cdots,x_m\}$指 m 个样本, 计算样本之间的距离测度, 得到距离测度矩阵$D=(d_{ij})_{n\times n}$, 其中$d_{ij}=d(A_i,A_j)$.

步骤 2. 计算距离测度矩阵$D=(d_{ij})_{n\times n}$的等价矩阵: $D^2=D\circ D=(\bar{d}_{ij})_{n\times n}$, $\bar{d}_{ij}=\min_k\{\max\{d_{ik},d_{kj}\}\}$, $i,j=1,2,\cdots,n$, $D^2\subseteq D$, D^2称为D的关联矩阵, 重复如下操作:

$$D\to D^2\to D^4\to\cdots\to D^{2^k}\to\cdots$$

直到$D^{2^k}=D^{2^{(k+1)}}$, 则$D^{2^{(k+1)}}$是D的等价矩阵, 对$D^{2^{(k+1)}}$进行运算的结果和对矩阵D进行运算的效果一样[14].

步骤 3. 按照从大到小的顺序从$D^{2^{(k+1)}}$中选择置信水平λ值, $\lambda\in[0,1]$, 然后对$D^{2^{(k+1)}}$进行切割得到对应的

λ–切割矩阵.

步骤 4. λ–切割矩阵中的每一列看作一个向量, 表示为 $\mathrm{D}_\lambda=(\bar{\alpha}_1,\bar{\alpha}_2,\cdots,\bar{\alpha}_n)$, 其中 $\bar{\alpha}_j=(\alpha_{1j},\alpha_{2j},\cdots,\alpha_{nj})^{\mathrm{T}}$. 对任意两个列向量进行内积点乘运算 $(\bar{\alpha}_i,\bar{\alpha}_j)=\bar{\alpha}_i^{\mathrm{T}}\bar{\alpha}_j$, 如果 $(\bar{\alpha}_i,\bar{\alpha}_j)=\bar{\alpha}_i^{\mathrm{T}}\bar{\alpha}_j=0$, 则认为这两个列向量是正交关系.

步骤 5. 根据列向量之间的正交关系对样本进行聚类, 得到聚类结果.

K-means 算法是常用的聚类算法, 该算法需要给定 k 值用以指定将目标对象划分成 k 个类别. 算法的第一个步骤是要计算初始数据的质心, 然后计算数据到质心的距离进而得到新的集群质心, 不断迭代这个过程, 直到质心的位置不再变化, 即聚类结束. 该算法精度高, 是最为广泛使用的聚类算法之一, 但是 k-means 的效率高低很大程度上依靠于对 k 值和初始质心的选择, 选择不当往往造成迭代次数多, 计算量大, 消耗时间成本大的问题, 因此本文只借助它的优点来对本文提出的算法结果的准确性进行验证.

本文将算法 2 的聚类结果作为 k-means 的初始数据, 代入到 k-means 算法中, 进行一次迭代运算, 求得迭代之后的聚类结果, 如果该结果与算法 2 的聚类结果一致, 则说明聚类结果准确.

4 实例分析

聚类分析在各行各业的应用十分常见, 对顾客进行细分是最为常见的分析需求, 本文以顾客细分为例, 验证本文提出的正交模糊聚类算法的可行性和高效性.

设某公司要对自己的客户进行划分, 划分客户的主要参考因素为以下 5 个: (1) 消费水平 c_1; (2) 收入水平 c_2; (3) 文化程度 c_3; (4) 上网时间长度 c_4; (5) 外貌长相等级 c_5. 5 个属性分别所占权重为: w=(0.25, 0.2, 0.25, 0.15, 0.15)$^{\mathrm{T}}$, 依据语言评价术语集, S^1={s_{-3}: 非常低, s_{-2}: 很低, s_{-1}: 低, s_0: 一般, s_1: 高, s_2: 很高, s_3: 非常高}, S^2={s_{-3}: 非常短, s_{-2}: 很短, s_{-1}: 短, s_0: 一般, s_1: 长, s_2: 很长, s_3: 非常长}, 给出了 10 位客户 $P=(p_1,p_2,\cdots,p_{10})$ 的评估信息, 如表 2 所示.

表 2 某公司针对 10 位客户的评估信息

	c_1	c_2	c_3	c_4	c_5
p_1	$\{s_0,s_1\}$	$\{s_{-2},s_{-1}\}$	$\{s_0,s_1\}$	$\{s_1,s_2\}$	$\{s_1,s_2,s_3\}$
p_2	$\{s_2\}$	$\{s_{-1},s_0,s_1\}$	$\{s_{-2},s_{-1}\}$	$\{s_1\}$	$\{s_1,s_2\}$
p_3	$\{s_0,s_1\}$	$\{s_0,s_1\}$	$\{s_0,s_1,s_2\}$	$\{s_1,s_2,s_3\}$	$\{s_0,s_1,s_2\}$
p_4	$\{s_{-2},s_{-1}\}$	$\{s_{-2},s_{-1}\}$	$\{s_1,s_2\}$	$\{s_3\}$	$\{s_{-1},s_0,s_1\}$
p_5	$\{s_1,s_2\}$	$\{s_{-2},s_{-1}\}$	$\{s_0,s_1\}$	$\{s_2,s_3\}$	$\{s_0,s_1\}$
p_6	$\{s_{-1},s_0,s_1\}$	$\{s_1\}$	$\{s_1,s_2\}$	$\{s_1,s_2,s_3\}$	$\{s_{-1},s_0,s_1\}$
p_7	$\{s_1,s_2,s_3\}$	$\{s_0,s_1,s_2\}$	$\{s_{-2},s_{-1}\}$	$\{s_0,s_1,s_2\}$	$\{s_1,s_2\}$
p_8	$\{s_0,s_1\}$	$\{s_{-3},s_{-2}\}$	$\{s_0,s_1,s_2\}$	$\{s_1,s_2,s_3\}$	$\{s_1,s_2\}$
p_9	$\{s_1,s_2\}$	$\{s_{-1},s_0,s_1\}$	$\{s_0,s_1\}$	$\{s_2,s_3\}$	$\{s_{-3},s_{-2}\}$
p_{10}	$\{s_0,s_1\}$	$\{s_{-1},s_0,s_1\}$	$\{s_0,s_1,s_2\}$	$\{s_1,s_2\}$	$\{s_0,s_1,s_2\}$

步骤 1. 将得到的评估信息进行规范化, 即为元素较少的 HFLTSs 补齐元素, 使 HFLTEs 个数一致:

$$\{g_{ij}\}_{m\times n}=\begin{pmatrix}
\{s_0,s_{0.5},s_1\} & \{s_{-2},s_{-1.5},s_{-1}\} & \{s_0,s_{0.5},s_1\} & \{s_1,s_{1.5},s_2\} & \{s_1,s_2,s_3\}\\
\{s_2,s_2,s_2\} & \{s_{-1},s_0,s_1\} & \{s_{-2},s_{-1.5},s_{-1}\} & \{s_1,s_1,s_1\} & \{s_1,s_{1.5},s_2\}\\
\{s_0,s_{0.5},s_1\} & \{s_0,s_{0.5},s_1\} & \{s_0,s_1,s_2\} & \{s_1,s_2,s_3\} & \{s_0,s_1,s_2\}\\
\{s_{-2},s_{-1.5},s_{-1}\} & \{s_{-2},s_{-1.5},s_{-1}\} & \{s_1,s_{1.5},s_2\} & \{s_3,s_3,s_3\} & \{s_{-1},s_0,s_1\}\\
\{s_1,s_{1.5},s_2\} & \{s_1,s_{1.5},s_2\} & \{s_0,s_{0.5},s_1\} & \{s_2,s_{2.5},s_3\} & \{s_0,s_{0.5},s_1\}\\
\{s_{-1},s_0,s_1\} & \{s_1,s_1,s_1\} & \{s_1,s_{1.5},s_2\} & \{s_1,s_2,s_3\} & \{s_{-1},s_0,s_1\}\\
\{s_1,s_2,s_3\} & \{s_0,s_1,s_2\} & \{s_{-2},s_{-1.5},s_{-1}\} & \{s_0,s_1,s_2\} & \{s_1,s_{1.5},s_2\}\\
\{s_0,s_{0.5},s_1\} & \{s_{-3},s_{-2.5},s_{-2}\} & \{s_0,s_1,s_2\} & \{s_1,s_2,s_3\} & \{s_1,s_{1.5},s_2\}\\
\{s_1,s_{1.5},s_2\} & \{s_{-1},s_0,s_1\} & \{s_0,s_{0.5},s_1\} & \{s_2,s_{2.5},s_3\} & \{s_{-3},s_{-2.5},s_{-2}\}\\
\{s_0,s_{0.5},s_1\} & \{s_{-1},s_0,s_1\} & \{s_0,s_1,s_2\} & \{s_1,s_{1.5},s_2\} & \{s_0,s_1,s_2\}
\end{pmatrix}$$

步骤 2. 根据距离测量公式 (7) 计算样本之间的距离测度, 其中 $\alpha=\beta=0.5$, $\lambda=2$, 得到距离测量矩阵D:

$$\mathrm{D}=\begin{pmatrix}
0 & 0.2274 & 0.1158 & 0.2222 & 0.2578 & 0.2354 & 0.3091 & 0.0486 & 0.3982 & 0.0791\\
0.2274 & 0 & 0.2707 & 0.6852 & 0.2146 & 0.4478 & 0.0533 & 0.3829 & 0.3892 & 0.2394\\
0.1158 & 0.2707 & 0 & 0.2355 & 0.0736 & 0.0506 & 0.2593 & 0.1878 & 0.2441 & 0.0189\\
0.2222 & 0.6852 & 0.2355 & 0 & 0.4450 & 0.2201 & 0.7787 & 0.1983 & 0.4054 & 0.2163\\
0.2578 & 0.2146 & 0.0736 & 0.4450 & 0 & 0.1132 & 0.1762 & 0.3808 & 0.1854 & 0.1106\\
0.2354 & 0.4478 & 0.0506 & 0.2201 & 0.1132 & 0 & 0.3945 & 0.3134 & 0.2303 & 0.0861\\
0.3091 & 0.0533 & 0.2593 & 0.7788 & 0.1762 & 0.3945 & 0 & 0.4915 & 0.4108 & 0.2610\\
0.0486 & 0.3829 & 0.1878 & 0.1983 & 0.3808 & 0.3134 & 0.4915 & 0 & 0.4163 & 0.1454\\
0.3982 & 0.3892 & 0.2441 & 0.4054 & 0.1854 & 0.2303 & 0.4108 & 0.4163 & 0 & 0.2402\\
0.0791 & 0.2394 & 0.0189 & 0.2163 & 0.1106 & 0.0861 & 0.2610 & 0.1454 & 0.2402 & 0
\end{pmatrix}$$

步骤 3. 计算距离测量矩阵D的等价矩阵:

$$D^2=\begin{pmatrix}
0 & 0.2274 & 0.0791 & 0.1983 & 0.1106 & 0.0861 & 0.2274 & 0.0486 & 0.2354 & 0.0791\\
0.2274 & 0 & 0.2146 & 0.2274 & 0.1762 & 0.2146 & 0.0533 & 0.2274 & 0.2146 & 0.2146\\
0.0791 & 0.2146 & 0 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.1158 & 0.1854 & 0.0189\\
0.1983 & 0.2274 & 0.1983 & 0 & 0.2163 & 0.2163 & 0.2593 & 0.1983 & 0.2303 & 0.1983\\
0.1106 & 0.1762 & 0.0736 & 0.2163 & 0 & 0.0736 & 0.1762 & 0.1454 & 0.1854 & 0.0736\\
0.0861 & 0.2146 & 0.0506 & 0.2163 & 0.0736 & 0 & 0.1762 & 0.1454 & 0.1854 & 0.0506\\
0.2274 & 0.0533 & 0.1762 & 0.2593 & 0.1762 & 0.1762 & 0 & 0.2593 & 0.1854 & 0.1762\\
0.0486 & 0.2274 & 0.1158 & 0.1983 & 0.1454 & 0.1454 & 0.2593 & 0 & 0.2402 & 0.0791\\
0.2354 & 0.2146 & 0.1854 & 0.2303 & 0.1854 & 0.1854 & 0.1854 & 0.2402 & 0 & 0.1854\\
0.0791 & 0.2146 & 0.0189 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0
\end{pmatrix}$$

$$D^4=\begin{pmatrix}
0 & 0.1762 & 0.0791 & 0.1983 & 0.0791 & 0.0791 & 0.1762 & 0.0486 & 0.1854 & 0.0791\\
0.1762 & 0 & 0.1762 & 0.2146 & 0.1762 & 0.1762 & 0.0533 & 0.1762 & 0.1854 & 0.1762\\
0.0791 & 0.1762 & 0 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0.0189\\
0.1983 & 0.2146 & 0.1983 & 0 & 0.1983 & 0.1983 & 0.1983 & 0.1983 & 0.1983 & 0.1983\\
0.0791 & 0.1762 & 0.0736 & 0.1983 & 0 & 0.0736 & 0.1762 & 0.0791 & 0.1854 & 0.0736\\
0.0791 & 0.1762 & 0.0506 & 0.1983 & 0.0736 & 0 & 0.1762 & 0.0791 & 0.1854 & 0.0506\\
0.1762 & 0.0533 & 0.1762 & 0.1983 & 0.1762 & 0.1762 & 0 & 0.1762 & 0.1854 & 0.1762\\
0.0486 & 0.1762 & 0.0791 & 0.1983 & 0.0791 & 0.0791 & 0.1762 & 0 & 0.1854 & 0.0791\\
0.1854 & 0.1854 & 0.1854 & 0.1983 & 0.1854 & 0.1854 & 0.1854 & 0.1854 & 0 & 0.1854\\
0.0791 & 0.1762 & 0.0189 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0
\end{pmatrix}$$

$$D^8=\begin{pmatrix}
0 & 0.1762 & 0.0791 & 0.1983 & 0.0791 & 0.0791 & 0.1762 & 0.0486 & 0.1854 & 0.0791\\
0.1762 & 0 & 0.1762 & 0.1983 & 0.1762 & 0.1762 & 0.0533 & 0.1762 & 0.1854 & 0.1762\\
0.0791 & 0.1762 & 0 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0.0189\\
0.1983 & 0.1983 & 0.1983 & 0 & 0.1983 & 0.1983 & 0.1983 & 0.1983 & 0.1983 & 0.1983\\
0.0791 & 0.1762 & 0.0736 & 0.1983 & 0 & 0.0736 & 0.1762 & 0.0791 & 0.1854 & 0.0736\\
0.0791 & 0.1762 & 0.0506 & 0.1983 & 0.0736 & 0 & 0.1762 & 0.0791 & 0.1854 & 0.0506\\
0.1762 & 0.0533 & 0.1762 & 0.1983 & 0.1762 & 0.1762 & 0 & 0.1762 & 0.1854 & 0.1762\\
0.0486 & 0.1762 & 0.0791 & 0.1983 & 0.0791 & 0.0791 & 0.1762 & 0 & 0.1854 & 0.0791\\
0.1854 & 0.1854 & 0.1854 & 0.1983 & 0.1854 & 0.1854 & 0.1854 & 0.1854 & 0 & 0.1854\\
0.0791 & 0.1762 & 0.0189 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0
\end{pmatrix}$$

$$D^{16}=\begin{pmatrix}
0 & 0.1762 & 0.0791 & 0.1983 & 0.0791 & 0.0791 & 0.1762 & 0.0486 & 0.1854 & 0.0791\\
0.1762 & 0 & 0.1762 & 0.1983 & 0.1762 & 0.1762 & 0.0533 & 0.1762 & 0.1854 & 0.1762\\
0.0791 & 0.1762 & 0 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0.0189\\
0.1983 & 0.1983 & 0.1983 & 0 & 0.1983 & 0.1983 & 0.1983 & 0.1983 & 0.1983 & 0.1983\\
0.0791 & 0.1762 & 0.0736 & 0.1983 & 0 & 0.0736 & 0.1762 & 0.0791 & 0.1854 & 0.0736\\
0.0791 & 0.1762 & 0.0506 & 0.1983 & 0.0736 & 0 & 0.1762 & 0.0791 & 0.1854 & 0.0506\\
0.1762 & 0.0533 & 0.1762 & 0.1983 & 0.1762 & 0.1762 & 0 & 0.1762 & 0.1854 & 0.1762\\
0.0486 & 0.1762 & 0.0791 & 0.1983 & 0.0791 & 0.0791 & 0.1762 & 0 & 0.1854 & 0.0791\\
0.1854 & 0.1854 & 0.1854 & 0.1983 & 0.1854 & 0.1854 & 0.1854 & 0.1854 & 0 & 0.1854\\
0.0791 & 0.1762 & 0.0189 & 0.1983 & 0.0736 & 0.0506 & 0.1762 & 0.0791 & 0.1854 & 0
\end{pmatrix}$$

根据上面的计算结果可知$D^{16}=D^8$, 所以D^8是D等价矩阵.

步骤 4. 按照从大到小的顺序从D^8中选择置信水平λ值, 然后对D^8进行切割得到对应的λ–切割矩阵:

例如当$\lambda = 0.1983$时,

$$\mathrm{D}_{\lambda=0.1983} = \begin{pmatrix} 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1\\ 0&0&0&1&0&0&0&0&0&0\\ 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1\\ 1&1&1&0&1&1&1&1&1&1 \end{pmatrix}$$

当$\lambda = 0.0486$时,

$$\mathrm{D}_{\lambda=0.0486} = \begin{pmatrix} 1&0&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&0&1 \end{pmatrix}$$

D^8中有 10 个不同的值, 意味着要进行 10 次切割, 而如果直接对原始距离测度矩阵D进行该项操作, 则需要做 47 次切割, 显然, 采用等价矩阵可以大大降低计算复杂度.

步骤 5. 把λ–切割矩阵中的每一列看作一个向量, 即$\mathrm{D}_\lambda = (\bar{\alpha}_1, \bar{\alpha}_2, \cdots, \bar{\alpha}_{10})$, $\bar{\alpha}_j = (\alpha_{1j}, \alpha_{2j}, \cdots, \alpha_{10j})^{\mathrm{T}}$. 对任意两个列向量进行内积点乘运算$(\bar{\alpha}_i, \bar{\alpha}_j) = \bar{\alpha}_i^T \bar{\alpha}_j$, 如果$(\bar{\alpha}_i, \bar{\alpha}_j) = \bar{\alpha}_i^{\mathrm{T}} \bar{\alpha}_j = 0$, 则认为这两个列向量是正交关系, 对应的两个样本不能归为同一个类别. 比如, 当$\lambda = 0.1983$时, $(\bar{\alpha}_4, \bar{\alpha}_5) = \bar{\alpha}_4^{\mathrm{T}} \bar{\alpha}_5 = 0$, 因此, 样本 4, p_4, 和样本 5, p_5, 不能归为同一类. 据此原理, 得到聚类结果如表 3.

步骤 6. 将上面得到的聚类结果作为 k-means 算法的初始集群, 做进一步聚类分析, 验证本文算法聚类结果的准确性. 因为分为 10 个类和 1 个类的结果都只有一种, 所以下面只对分为 2–9 个类的结果进行验证.

当$k = 9$时, 将样本分为 9 类, 分别为: $C_1 = \{p_1\}$, $C_2 = \{p_2\}$, $C_3 = \{p_3, p_{10}\}$, $C_4 = \{p_4\}$, $C_5 = \{p_5\}$, $C_6 = \{p_6\}$, $C_7 = \{p_7\}$, $C_8 = \{p_8\}$, $C_9 = \{p_9\}$.

$C_3 = \{p_3, p_{10}\}$的质心为p_3和p_{10}的中点:

$$\{\{s_0\}\{s_{-0.5}, s_{0.25}, s_0\}\{s_0\}\{s_0, s_{1.75}, s_{2.5}\}\{s_0\}\}$$

表 3 正交模糊聚类结果

类	置信范围	聚类结果
10	$0 \le \lambda \le 0.0189$	$\{p_1\},\{p_2\},\{p_3\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_8\},\{p_9\},\{p_{10}\}$
9	$0.0189 < \lambda \le 0.0486$	$\{p_1\},\{p_2\},\{p_3,p_{10}\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_8\},\{p_9\}$
8	$0.0486 < \lambda \le 0.0506$	$\{p_1,p_8\},\{p_2\},\{p_3,p_{10}\},\{p_4\},\{p_5\},\{p_6\},\{p_7\},\{p_9\}$
7	$0.0506 < \lambda \le 0.0533$	$\{p_1,p_8\},\{p_2\},\{p_3,p_6,p_{10}\},\{p_4\},\{p_5\},\{p_7\},\{p_9\}$
6	$0.0533 < \lambda \le 0.0736$	$\{p_1,p_8\},\{p_2,p_7\},\{p_3,p_6,p_{10}\},\{p_4\},\{p_5\},\{p_9\}$
5	$0.0736 < \lambda \le 0.0791$	$\{p_1,p_8\},\{p_2,p_7\},\{p_3,p_5,p_6,p_{10}\},\{p_4\},\{p_9\}$
4	$0.0791 < \lambda \le 0.1762$	$\{p_1,p_3,p_5,p_6,p_8,p_{10}\},\{p_2,p_7\},\{p_4\},\{p_9\}$
3	$0.1762 < \lambda \le 0.1854$	$\{p_1,p_2,p_3,p_5,p_6,p_7,p_8,p_{10}\},\{p_4\},\{p_9\}$
2	$0.1854 < \lambda \le 0.1983$	$\{p_1,p_2,p_3,p_5,p_6,p_7,p_8,p_9,p_{10}\},\{p_4\}$
1	$0.1983 \le \lambda \le 1$	$\{p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9,p_{10}\}$

计算每一个样本到类之间的距离测度:

$d(p_1,C_1) = 0$ $d(p_1,C_2) = 0.2274$ $d(p_1,C_3) = 0.0933$

$d(p_1,C_4) = 0.2222$ $d(p_1,C_5) = 0.2578$

$d(p_1,C_6) = 0.2354$ $d(p_1,C_7) = 0.3091$

$d(p_1,C_8) = 0.0486$ $d(p_1,C_9) = 0.3982$

由此可知, p_1与C_1类的距离最小, 因此p_1属于C_1类;

$d(p_2,C_1) = 0.2274$ $d(p_2,C_2) = 0$ $d(p_2,C_3) = 0.2513$

$d(p_2,C_4) = 0.6852$ $d(p_2,C_5) = 0.2146$

$d(p_2,C_6) = 0.4478$ $d(p_2,C_7) = 0.0533$

$d(p_2,C_8) = 0.3829$ $d(p_2,C_9) = 0.3892$

由此可知, p_2属于C_2类;

$d(p_3,C_1) = 0.1158$ $d(p_3,C_2) = 0.2707$

$d(p_3,C_3) = 0.0059$

$d(p_3,C_4) = 0.2355$ $d(p_3,C_5) = 0.0736$

$d(p_3,C_6) = 0.0506$ $d(p_3,C_7) = 0.2593$

$d(p_3,C_8) = 0.1878$ $d(p_3,C_9) = 0.2441$

由此可知, p_3属于C_3类;

$d(p_4,C_1) = 0.2222$ $d(p_4,C_2) = 0.6852$

$d(p_4,C_3)=0.2217$ $d(p_4,C_4)=0$ $d(p_4,C_5)=0.4450$
$d(p_4,C_6)=0.2201$ $d(p_4,C_7)=0.7788$
$d(p_4,C_8)=0.1983$ $d(p_4,C_9)=0.4054$
由此可知, p_4属于C_4类;
$d(p_5,C_1)=0.2578$ $d(p_5,C_2)=0.2146$
$d(p_5,C_3)=0.0886$ $d(p_5,C_4)=0.4450$ $d(p_5,C_5)=0$
$d(p_5,C_6)=0.1132$ $d(p_5,C_7)=0.1762$
$d(p_5,C_8)=0.3808$ $d(p_5,C_9)=0.1854$
由此可知, p_5属于C_5类;
$d(p_6,C_1)=0.2354$ $d(p_6,C_2)=0.4478$
$d(p_6,C_3)=0.0648$ $d(p_6,C_4)=0.2201$
$d(p_6,C_5)=0.1132$ $d(p_6,C_6)=0$ $d(p_6,C_7)=0.3945$
$d(p_6,C_8)=0.3134$ $d(p_6,C_9)=0.2303$
由此可知, p_6属于C_6类;
$d(p_7,C_1)=0.3091$ $d(p_7,C_2)=0.0533$
$d(p_7,C_3)=0.2581$ $d(p_7,C_4)=0.7787$
$d(p_7,C_5)=0.1762$ $d(p_7,C_6)=0.3945$ $d(p_7,C_7)=0$
$d(p_7,C_8)=0.4915$ $d(p_7,C_9)=0.4108$
由此可知, p_7属于C_7类;
$d(p_8,C_1)=0.0486$ $d(p_8,C_2)=0.3829$
$d(p_8,C_3)=0.1637$ $d(p_8,C_4)=0.1983$
$d(p_8,C_5)=0.3808$ $d(p_8,C_6)=0.3134$
$d(p_8,C_7)=0.4915$ $d(p_8,C_8)=0$ $d(p_8,C_9)=0.4163$
由此可知, p_8属于C_8类;
$d(p_{10},C_1)=0.0933$ $d(p_{10},C_2)=0.2513$
$d(p_{10},C_3)=0.0067$
$d(p_{10},C_4)=0.2217$ $d(p_{10},C_5)=0.0886$
$d(p_{10},C_6)=0.0648$ $d(p_{10},C_7)=0.2581$
$d(p_{10},C_8)=0.1637$ $d(p_{10},C_9)=0.2394$
由此可知, p_{10}属于C_3类.

采用 k-means 进行聚类的结果为: $C_1=\{p_1\}$, $C_2=\{p_2\}$, $C_3=\{p_3,p_{10}\}$, $C_4=\{p_4\}$, $C_5=\{p_5\}$, $C_6=\{p_6\}$, $C_7=\{p_7\}$, $C_8=\{p_8\}$, $C_9=\{p_9\}$, 集群的质心没有发生改变, 聚类结果与迭代之前一致, 迭代结束.

同理可以得到当k分别取 2-8 时的 k-means 聚类结果, 最后发现, 聚类的结果和正交聚类的结果一致, 证明了本文提出的算法 2 的准确性. 与算法 1 相比, 算法 2 基于等价矩阵的基础上进行正交运算的, 降低了计算复杂度, 优化了算法性能, 更加适应于样本数据量大的情况. 综上, 本文提出的基于 HFLTSs 的正交模糊聚类算法算法复杂度相对较低, 准确性高, 解决了传统模糊聚类算法存在的缺陷.

5 总结

模糊聚类逐渐成为新的研究热点, 许多模糊聚类算法已经被提出, 但是基于 HFLTSs 的模糊聚类算法尚未成熟, 存在计算复杂度高的缺陷, 而 HFLTSs 是比较流行而且灵活度很高的语言术语, 因此本文提出了计算复杂度相对较低的基于 HFLTSs 的正交模糊聚类算法. 该算法基于 HFLTSs 的距离测量矩阵采用正交思想, 确定无法划分为同个类别的样本, 得到聚类结果. 为了验证算法的准确性和高效性, 本文还通过一个实例结合 k-means 算法对本文算法进行了验证. 未来 , 我们将继续研究将该算法扩展延伸至可以应用于更多类型的语言术语, 例如概率语言术语集 (PLTSs), 以及为了使该算法可以更好地应用于大数据做进一步的研究和努力.

参考文献

1 Wong CC, Lai HR. A grey-based clustering algorithm and its application on fuzzy system design. International Journal of Systems Science, 2003, 34(4): 269–281. [doi: 10.1080/00207720310001585 19]

2 Alzate C, Suykens JAK. Hierarchical kernel spectral clustering. Neural Networks, 2012, 35: 21–30. [doi: 10.1016/j.neunet.2012.06.007]

3 赵纯, 高俊波. 基于区间直觉模糊的情感分类模型. 计算机系统应用, 2014, 23(10): 207–211. [doi: 10.3969/j.issn.1003-3254.2014.10.037]

4 Zadeh LA. The concept of a linguistic variable and its application to approximate reasoning-II. Information Sciences, 1975, 8(4): 301–357. [doi: 10.1016/0020-0255(75)90046-8]

5 Torra V. Hesitant fuzzy sets. International Journal of Intelligent Systems, 2010, 25(6): 529–539.

6 Rodríguez RM, MartíNez L, Herrera F. Hesitant fuzzy linguistic term sets for decision making. IEEE Transactions on Fuzzy Systems, 2012, 20(1): 109–119. [doi: 10.1109/TFUZZ.2011.2170076]

7 卢志刚, 陈行娟. 基于信息熵的模糊多属性决策供应商选择方法. 计算机系统应用, 2012, 21(8): 170–173, 232.

8 Degani R, Bortolan G. The problem of linguistic approximation in clinical decision making. International Journal of Approximate Reasoning, 1988, 2(2): 143–162. [doi: 10.1016/0888-613X(88)90105-3]

9 Herrera F, Martinez L. A 2-tuple fuzzy linguistic representation model for computing with words. IEEE Transactions on Fuzzy Systems, 2000, 8(6): 746–752. [doi: 10.1109/91.890332]

10 Xu ZS, Wang H. On the syntax and semantics of virtual linguistic terms for information fusion in decision making. Information Fusion, 2017, (34): 43–48. [doi: 10.1016/j.inffus.2016.06.002]

11 廖虎昌, 缑迅杰, 徐泽水. 基于犹豫模糊语言集的决策理论与方法综述. 系统工程理论与实践, 2017, 37(1): 35–48. [doi: 10.12011/1000-6788(2017)01-0035-14]

12 Beg I, Rashid T. TOPSIS for hesitant fuzzy linguistic term sets. International Journal of Intelligent Systems, 2013, 28(12): 1162–1171.

13 张洪美, 徐泽水, 陈琦. 直觉模糊集的聚类方法研究. 控制与决策, 2007, 22(8): 882–888.

14 Xu ZS, Chen J, Wu JJ. Clustering algorithm for intuitionistic fuzzy sets. Information Sciences, 2008, 178(19): 3775–3790. [doi: 10.1016/j.ins.2008.06.008]

15 Zhang XL, Xu ZS. An MST cluster analysis method under hesitant fuzzy environment. Control and Cybernetics, 2012, 41(3): 645–666.

16 Chen N, Xu ZS, Xia MM. Correlation coefficients of hesitant fuzzy sets and their applications to clustering analysis. Applied Mathematical Modelling, 2013, 37(4): 2197–2211. [doi: 10.1016/j.apm.2012.04.031]

17 Chen N, Xu ZS, Xia MM. Hierarchical hesitant fuzzy K-means clustering algorithm. Applied Mathematics-A Journal of Chinese Universities, 2014, 29(1): 1–17. [doi: 10.1007/s11766-014-3091-8]

18 Yang X, Xu ZS, Liao HC. Correlation coefficients of hesitant multiplicative sets and their applications in decision making and clustering analysis. Applied Soft Computing, 2017, (61): 935–946. [doi: 10.1016/j.asoc.2017.08.011]

19 Liu YM, Zhao H, Xu ZS. An orthogonal clustering method under hesitant fuzzy environment. International Journal of Computational Intelligence Systems, 2017, 10(1): 663–676. [doi: 10.2991/ijcis.2017.10.1.44]

20 Wei C, Zhao N, Tang X. Operators and comparisons of hesitant fuzzy linguistic term sets. IEEE Transactions on Fuzzy Systems, 2014, 22(3): 575–585.

21 Xu ZS, Wang H. On the syntax and semantics of virtual linguistic terms for information fusion in decision making. Information Fusion, 2017, (34): 43–48. [doi: 10.1016/j.inffus.2016.06.002]

22 蒋圆, 徐泽水, 舒轶昊. 直觉积性模糊集的距离测度及其在卫星地球站选址问题中应用. 系统工程理论与实践, 2016, 36(12): 3210–3219. [doi: 10.12011/1000-6788(2016)12-3210-10]

23 Liao HC, Xu ZS. Approaches to manage hesitant fuzzy linguistic information based on the cosine distance and similarity measures for HFLTSs and their application in qualitative decision making. Expert Systems with Applications, 2015, 42(12): 5328–5336. [doi: 10.1016/j.eswa.2015.02.017]

24 胡辉, 徐泽水. 基于 TOPSIS 的区间直觉模糊多属性决策法. 模糊系统与数学, 2007, 21(5): 108–112.

25 Li DQ, Zeng WY, Li JH. New distance and similarity measures on hesitant fuzzy sets and their applications in multiple criteria decision making. Engineering Applications of Artificial Intelligence, 2015, (40): 11–16. [doi: 10.1016/j.engappai.2014.12.012]

26 Tan QY, Wei CP, Liu Q, *et al*. The hesitant fuzzy linguistic TOPSIS method based on novel information measures. Asia-Pacific Journal of Operational Research, 2016, 33(5): 1650035. [doi: 10.1142/S0217595916500354]