

基于相似度均值的分类数据层次聚类分析算法

褚轲欣, 荀亚玲

(太原科技大学 计算机科学与技术学院, 山西 太原 030024)

摘要:层次聚类分析在数据挖掘与机器学习等领域是一种广泛使用的无监督学习技术,但是,由于层次聚类分析算法主要是依赖于人为设定的相似度阈值来实现聚类簇的合并或分裂,因此在没有任何先验知识时,难以设定相似度阈值。采用相似度均值以及边界数据对象分配策略,提出了一种基于相似度均值的分类数据层次聚类分析算法。该算法利用相似度均值刻画数据集中数据对象分布的集中趋势以及平稳相似性度量,作为层次聚类簇合并或分裂的重要依据,给出了一种相似度均值的计算公式,从而可以自动确定相似度阈值,解决了层次聚类分析中相似度阈值参数的人为设定问题;利用相似度均值,给出了一种边界数据对象的分配策略,有效提高了边界数据对象分配的准确性及聚类质量。在UCI与人工合成数据集上的实验验证了该算法具有良好的聚类性能和抗噪性,以及相似度均值的稳定性和有效性。

关键词:层次聚类;分类数据;相似度均值;平稳相似性度量;分配策略

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2022)11-0154-10

doi:10.3969/j.issn.1673-629X.2022.11.023

A Hierarchical Clustering Analysis Algorithm of Categorical Data Based on Mean of Similarity

CHU Ke-xin, XUN Ya-ling

(School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, China)

Abstract: Hierarchical clustering analysis is a widely used unsupervised learning technology in the fields of data mining and machine learning. However, it is difficult to set the similarity threshold without any prior knowledge, since the hierarchical clustering analysis algorithm mainly relies on the similarity thresholds by artificial setting to realize the merging or splitting of clusters. Based on the mean of similarity and boundary data object allocation strategy, a hierarchical clustering analysis algorithm of categorical data using the mean of similarity is proposed. As an important basis for the merging or splitting of clusters in hierarchical clustering, the algorithm uses the steady similarity measure and the mean of similarity can capture the central tendency of the distribution of data objects in the data sets. A calculation formula of the mean of similarity is given, which can automatically determine the similarity threshold and solve the artificial setting of the similarity threshold parameters in the hierarchical clustering analysis. A boundary data object allocation strategy is presented by using the mean of similarity, which can effectively improve the accuracy of boundary data objects allocation and clustering quality. Experimental results validate the excellent clustering performance and anti-noise, as well as the stability and effectiveness of the algorithm's mean of similarity on UCI and artificial data sets.

Key words: hierarchical clustering; categorical data; mean of similarity; steady similarity measure; allocation strategy

0 引言

作为一种无监督学习方法,聚类分析是数据挖掘、机器学习等领域的主要研究内容之一。聚类分析根据某种相似性度量将数据对象划分为多个类簇,并使得在同一个类簇中的不同数据对象之间相似度较大,不同类簇中的数据对象之间相似度较小^[1]。在商务智能^[2]、图像处理^[3]、市场分析^[4]、模式识别^[5]、基因研

究^[6]等领域应用广泛。但随着数据类型日益复杂化和多样化,人为设置参数,聚类效果对参数敏感且参数不易确定,成为当前聚类分析面临的主要挑战之一。

层次聚类分析是一类典型的聚类分析方法,通过某种相似性度量确定数据对象之间的相似性,并对数据集中的数据对象不断地合并或分裂,可识别任意形状的簇。但由于层次聚类的合并或分裂依赖于事先人

收稿日期:2021-12-01

修回日期:2022-04-05

基金项目:国家自然科学基金项目(61602335);山西省自然科学基金(201901D211302)

作者简介:褚轲欣(1995-),女,硕士生,研究方向为数据挖掘与并行计算;通讯作者:荀亚玲(1980-),女,博士,研究生导师,副教授,研究方向为数据挖掘、并行计算。

为设定的相似度阈值,相似度阈值的取值直接影响最终的聚类簇个数与聚类质量^[7];在没有先验知识的情况下,相似度阈值参数难以确定;分类数据作为一类重要数据类型,对其层次聚类分析研究较少。根据自适应阈值法的思想^[8-11],该文提出了一种基于相似度均值的分类数据层次聚类分析算法。利用相似度均值,作为层次聚类簇合并或分裂的重要依据,解决了层次聚类分析中的参数人为设定问题。主要贡献如下:

- 提出了一种基于相似度均值的分类数据相似度阈值自动选取方法;
- 给出了一种边界数据对象分配策略;
- 提出了一种基于相似度均值的分类数据层次聚类分析算法。

1 相关工作

聚类分析的目的是使得划分后的数据点簇内彼此相似,簇间彼此相异。目前,聚类算法主要包括密度聚类算法^[12]、模型聚类算法^[13]、网格聚类算法^[14]、划分聚类算法^[15]以及层次聚类算法^[16-18]等。

层次聚类是一种基于原型的聚类方法,试图在不同层次对数据集进行划分,从而形成树形的聚类结构。通过绘制树状图,以可视化的方式来解释聚类结果,可解释性强^[19-20],可以对任意形状的簇进行聚类^[7]。层次聚类分析的典型成果主要包括:C-Ward^[17]算法在Ward算法^[21]的基础上,在层次聚类过程中依据聚类的中间结果动态更新必连和不连约束,以保证最终的聚类结果同时满足必连和不连约束。该算法保证了数据样本点获得更为合理的聚合顺序,从而得到更为准确的聚类结果。ROCK算法^[7]是一种用于分类数据的层次聚类分析算法。该算法采用随机抽样技术与链接的相似性度量计算两个数据对象相似度,考虑了周围数据对象的影响,但要求用户事先选定聚类数,且相似度函数只考虑数据对象之间是否相似而未考虑相似程度,所以对阈值较为敏感。Similarity算法^[22]提出了一种新的局部相似性度量,仅使用星型邻域子图,通过网络节点相似性度量的减函数来定义节点间的广义距离。相对全局的相似性度量,克服了传统局部相似性度量在某些情形下对节点相似性的低估倾向,但矩阵的存储形式难以适用于大型网络。DHCC算法^[23]基于多重对应分析(MCA)初始化,提出了初始化和细化聚类分割的有效步骤,能够无缝发现嵌入在子空间中的集群。该算法不采用全局细化,对初始误差传播问题不敏感,但会受异常对象的影响。MGR算法^[24]是一种基于信息论提出的算法,该算法利用信息论中的平均增益比(MGR)和簇熵概念确定聚类属性,并在属性定义的分区中选择等价类作为一个聚类簇,循环迭

代直到输出所有的数据对象,其算法时间复杂度较低。MTMDP算法^[25]采用概率粗糙集理论的分区属性选择方法TMDP与粒度概念相结合,根据等价关系,将数据对象划分为一组子集(粒子)。MTMDP算法的操作数是粒子而不是单独的数据对象,是一种鲁棒的聚类算法,用于处理分类数据聚类过程中的不确定性。MNIG算法^[26]对现有的聚类方法进行了系统的分析,总结了各自的优缺点,建立了一个统一的分层聚类框架并对该框架的每一步进行了改进,得到了性能更好的MNIG算法。

综上所述,随着对分类数据层次聚类分析的逐渐深入,针对分类数据的层次聚类算法获得了较好的聚类效果。但其聚类分析算法大都需要人为设定终止条件与相似度阈值等参数,控制聚类簇的合并或分裂,从而导致聚类效果受人为设定参数的影响较大。

2 相似性度量与聚类分析

聚类分析是指将物理或抽象对象的集合分组为由相似的对象组成的多个类簇的分析过程,因此如何描述数据对象间的相似性是聚类分析的重要问题。相似性度量是度量数据对象之间相似性的重要方法,参照文献^[27],相关概念定义如下:

设分类数据集 D 含有 N 个属性,其中属性 A_i 的取值集合记作: $\text{Dom}(A_i)$, $\text{Dom}(A_i) = \{a_{i,1}, a_{i,2}, \dots, a_{i,f}\}$, 表示属性 A_i 具有 f 个不同的取值。属性值 $a_{i,j} (a_{i,j} \in \text{Dom}(A_i))$ 关于属性 A_i 的支持度是数据集 D 中属性 A_i 取值等于 $a_{i,j}$ 的数据对象的个数,记作:

$$\text{Sup}(A_i | a_{i,j}) = |\{x | x \in D, x_i = a_{i,j}\}|$$

$$1 \leq j \leq f, 1 \leq i \leq N \quad (1)$$

其中, x_i 表示数据对象 x 在属性 A_i 上的取值。

相似度反映了数据对象之间的相似程度,取值区间为 $[0, 1]$ 。对于任意的 $x, y \in D$, x 与 y 的相似度 $S_c(x, y)$ 、相似性度量 $\theta(x_i, y_i)$ 可定义为:

$$S_c(x, y) = \sum_{i=1}^N W_i \theta(x_i, y_i) \quad (2)$$

$$\theta(x_i, y_i) = \begin{cases} 1, & x_i = y_i \\ 0, & x_i \neq y_i \end{cases} \quad (3)$$

其中, W_i 表示属性 A_i 所占的权重,即:

$$W_i = H_c(A_i) / \sum_{i=1}^N H_c(A_i), H_c(A_i) \text{ 表示属性 } A_i \text{ 的}$$

熵,即: $H_c(A_i) = - \sum_{i=1, a_{i,j} \in \text{Dom}(A_i)}^f p(a_{i,j}) \log_2(p(a_{i,j}))$, $p(a_{i,j})$ 表示属性 A_i 中的属性值 $a_{i,j}$ 的概率,即: $p(a_{i,j}) = \text{Sup}(A_i | a_{i,j}) / |D|$ 。

在公式(3)中,分类数据的相似度计算引入相似性度量 $\theta(x_i, y_i)$, 当 $x_i \neq y_i$ 时, $\theta(x_i, y_i) = 0$ 。代入公

式(2)计算相似度 $S_c(x, y)$, 若 $\theta(x_i, y_i) = 0$, 无论该属性维计算得到的权重值 W_i 多大, 那么该属性维的相似程度 $W_i \theta(x_i, y_i)$ 都为 0, 相当于该属性维在计算相似度 $S_c(x, y)$ 的过程中没有发挥作用。因此, 最终计算得到的相似度并未准确代表各数据对象之间真实的相似程度。

3 分类数据的层次聚类分析

3.1 相似度均值与层次聚类分析

层次聚类分析作为一种典型的聚类分析方法, 通过对数据集中的数据对象不断地合并或分裂, 可较容易地发现类的层次关系, 可聚类任意形状的簇^[7]。目前, 层次聚类分析算法主要是根据人为设定的相似度阈值参数, 实现聚类簇的合并或分裂。因此, 相似度阈值作为层次聚类分析中的唯一参数, 受人为因素的影响, 并影响着最终聚类簇个数与聚类质量^[28-29]。

由公式(2)中属性 A_i 的权重 W_i 计算公式可知, 将属性 A_i 的熵 $H_c(A_i)$ 的累加和 Soe 重新定义如下:

$$\text{Soe} = \sum_{i=1}^N H_c(A_i) \quad (4)$$

其中, $H_c(A_i)$ 表示属性 A_i 的熵。

在公式(4)中, 由相似度的计算公式(2)可知, 相似度 $S_c(x, y)$ 是所有属性维的权重与相似性度量乘积的累加和。将 W_i 计算公式代入公式(2)的 $S_c(x, y)$ 中, 显然, $\sum_{i=1}^N H_c(A_i)$ 的数量级与 $S_c(x, y)$ 的数量级基本一致。总之, 由分类属性 A_i 的熵 $H_c(A_i)$ 的累加和 Soe 以预估相似度的大小, 为自动确定相似度阈值做准备。

依据文献[30]中两次归一化过程, 将较大数量级的值归一化到距离目的数量级相差不大的范围内的过程记为第一次归一化过程。根据 Soe 表示的相似度的数量级别, 首先将 Soe 按缩放比例缩放到 $[1, 10]$ 之间。由于余数的取值会影响归一化值的准确性, 对余数做四舍五入处理, 即: 余数小于 5 时, 归一化缩放比例值不变; 余数大于等于 5 时, 归一化缩放比例值加一。因此, 归一化缩放比例值 scale_n , 第一次归一化值 r_{nor} 定义如下:

$$\text{scale}_n = \begin{cases} \lfloor \text{Soe}/10 \rfloor, & \text{Soe} \% 10 < 5 \\ \lceil \text{Soe}/10 \rceil, & \text{Soe} \% 10 \geq 5 \end{cases} \quad (5)$$

$$r_{\text{nor}} = \begin{cases} \text{Soe}, & \text{Soe} < 5 \\ \text{Soe}/\text{scale}_n, & \text{Soe} \geq 5 \end{cases} \quad (6)$$

其中: $\lfloor \cdot \rfloor$ 、 $\lceil \cdot \rceil$ 表示向下、向上取整, Soe 表示分类属性 A_i 的熵 $H_c(A_i)$ 的累加和。

在公式(6)中, 归一化的目的是使预处理的数据被合理地限定在一定范围内, 提高数据对象相似度的表

现力, 使数据对象的相似度值较平稳分布在 $[1, 10]$ 之间。四舍五入的思想能使被保留部分与实际值差值不超过最后一位的二分之一, 使得误差和最小。因此, 四舍五入的方法在第一次归一化过程中可以减少 Soe 缩放误差, 提高第一次归一化值 r_{nor} 的准确性。

参照文献[30], 将第一次归一化值归一化到目的数量级的过程记为第二次归一化过程。进一步采用归一化与四舍五入相结合的思想, 将 r_{nor} 缩放到 $[0, 1]$ 的数量级, 得到第二次归一化值 f_{nor} 。第二次归一化值 f_{nor} 定义如下:

$$f_{\text{nor}} = r_{\text{nor}}/20 \quad (7)$$

其中, r_{nor} 表示第一次归一化值。

在公式(7)中, 归一化目的是使预处理的数据被限定在一定范围内, 避免由于特征本身表达方式的原因而导致在绝对数值上的小数据被大数据“吃掉”的情况, 以保证每个特征被平等对待并处于同一数量级, 从而使不同维度之间的特征在数值上有可比性。第二次归一化在第一次归一化的基础上, 得到了更准确的归一化值, 并将第二次归一化值 f_{nor} 归一化到目标范围(即: $[0, 1]$)之间。第二次归一化值 f_{nor} 可以较准确地反映相似度总体水平, 缩小相似度极端值的出现频率。

根据上述公式(7)所得 f_{nor} , 为缩小相似度差异, 将公式(3)重新定义为平稳相似性度量, 并描述如下:

$$\theta_s(x_i, y_i) = \begin{cases} 1, & x_i = y_i \\ f_{\text{nor}}, & x_i \neq y_i \end{cases} \quad (8)$$

其中, $\theta_s(x_i, y_i)$ 称之为平稳相似性度量; f_{nor} 表示属性值不相同(即: $x_i \neq y_i$)时的第二次归一化值, 取值范围为 $[0, 1]$ 。

在公式(8)中, $\theta_s(x_i, y_i)$ 保持属性值相同(即: $x_i = y_i$ 时)的取值不变, 等于 1, 属性值不同(即: $x_i \neq y_i$)时的取值等于 f_{nor} , f_{nor} 是一个动态值。对于每个属性维, 不会再出现 $W_i \theta(x_i, y_i) = 0$, 忽略相似度计算过程中丢失重要属性维的情况。 f_{nor} 近似反映了数据集所有数据对象相似度的分布趋势, 使得 $S_c(x, y)$ 变化比较平稳, 可以正确表示数据对象之间的相似度, 为计算相似度阈值做准备。

参照文献[9-11]中的自适应阈值法, 根据数据集的分布特征, 动态地自动确定数据集的相似度阈值, 可有效地降低由单一相似度阈值造成的数据对象的错误分配, 以及人为设定相似度阈值对聚类效果的影响。相对于最大最小值、最大最小值的平均值、高斯卷积、方差等方法, 全局数据对象之间的相似度平均值可以最大程度地表示数据集分布的集中趋势。在公式(8)中, $\theta_s(x_i, y_i)$ 体现了任意两个数据对象的平稳相似性

度量值,利用公式(2)计算得到平稳相似度。所有数据对象之间的相似度均值,可以有效地反映数据集分布的集中趋势,若数据对象之间的相似度大于该均值,表明数据对象之间体现了数据集分布的集中趋势,应出现在同一聚类簇(合并),反之亦然。总之,相似度均值可以取代层次聚类中的相似度阈值,作为聚类簇合并或分裂的重要依据。对于公式(8),相似度均值 \overline{X}_c 可定义如下:

$$\overline{X}_c = \frac{\sum_{|x|=1}^{|D|} \left(\sum_{y=|x|+1}^{|D|} S_c(x,y) / (|D| - |x|) \right)}{|D|} \quad (9)$$

其中, $|x|$ 表示任意数据对象 x 的编码, $|D|$ 表示数据集 D 的数据对象总数。

在公式(9)中,相似度均值 \overline{X}_c 反映了数据集中各数据对象相似度的集中分布,数据对象越远离聚类中心,相似度越小且小于 \overline{X}_c 越容易分裂到不同的聚类簇,反之亦然;所以,阈值作为层次聚类迭代合并的重要依据, \overline{X}_c 可以满足其每次合并或分裂对阈值的要求。相似度阈值作为一种将数据对象划分到不同聚类簇的重要依据,需要体现不同聚类簇之间数据对象的相似度差异。因此,使用相似度均值 \overline{X}_c 来近似确定相似度阈值,可清楚地表示数据集中数据对象的集中分布,作为聚类簇合并或分裂的重要依据,从而解决相似度阈值自动确定的问题。

3.2 边界数据对象分配策略

边界数据对象^[31]是指数据空间中处于高密度区域边沿的一类数据对象,它们的一侧是高密度区域,一侧是相对的低密度区域,其归属并不明确。依据文献[31]中边界数据对象定义,将同时分配到多个簇内,其相似度都大于相似度均值 \overline{X}_c 的数据对象重新定义为边界数据对象;分配到聚类簇内,但与簇外局部区域中少数数据对象的相似度大于相似度均值 \overline{X}_c 的数据对象称为边界数据对象或噪声数据对象。

由于层次聚类方法使得所有的数据对象在全部簇中有且仅出现一次,这样的合并方式使得边界数据对象无法被友好地确定属于哪一个簇的边界点或噪声点。针对以上两种边界数据对象的分布情况,边界数据对象分配策略如下:

(1)对于 $\forall x \in D, A, B$ 分别是任意两个不同的簇,如果 $\text{value}_{x \in A} > \text{value}_{x \in B}$, 则 $x \in A$, 反之 $x \in B$; 如果 $\text{value}_{x \in A} = \text{value}_{x \in B}$, 且 $\sum_{[z]=1, x \in A}^{\text{value}} S_c(x, x^{[z]}) / \text{value} > \sum_{[w]=1, x \in B}^{\text{value}} S_c(x, x^{[w]}) / \text{value}$, 则 $x \in A$, 反之 $x \in B$; 如果

$\text{value}_{x \in A} = \text{value}_{x \in B}$, 且 $\sum_{[z]=1, x \in A}^{\text{value}} S_c(x, x^{[z]}) / \text{value} = \sum_{[w]=1, x \in B}^{\text{value}} S_c(x, x^{[w]}) / \text{value}$, 则 $x \notin A \& x \notin B$, $x \in$ 噪声点。

(2)对于 $\forall x \in D, A \in$ 任意簇, $B \in$ 局部区域, 如果 $\text{value}_{x \in A} > \text{value}_{x \in B}$, 则 $x \in A$, 反之 $x \in B$ (即: x 是噪声点); 如果 $\text{value}_{x \in A} = \text{value}_{x \in B}$, 且 $\sum_{[z]=1, x \in A}^{\text{value}} S_c(x, x^{[z]}) / \text{value} > \sum_{[w]=1, x \in B}^{\text{value}} S_c(x, x^{[w]}) / \text{value}$, 则 $x \in A$, 反之 $x \in B$ ($x \in$ 噪声点); 如果 $\text{value}_{x \in A} = \text{value}_{x \in B}$, 且 $\sum_{[z]=1, x \in A}^{\text{value}} S_c(x, x^{[z]}) / \text{value} = \sum_{[w]=1, x \in B}^{\text{value}} S_c(x, x^{[w]}) / \text{value}$, 则 $x \notin A \& x \notin B, x \in$ 噪声点。

其中: $\text{value}_{x \in A}$ 表示 x 在簇 A 中的 value 值, 同理可得 $\text{value}_{x \in B}$ 。 $x^{[z]}$ 表示簇 A 中第 z 个和 x 相似度大于相似度均值 \overline{X}_c 的数据对象, 同理可得 $x^{[w]}$ 。 $[z]$ 表示簇 A 中相似度值大于相似度均值 \overline{X}_c 的数据对象的个数, 取值为 $[1, \text{value}]$, 同理可得 $[w]$ 。

综合以上两种情况,对于边界数据对象的划分,首先对每一个数据对象添加存储一个 value 值,记录该数据对象到该簇中其他数据对象的相似度大于相似度均值 \overline{X}_c 的数据对象的个数。然后比较该数据对象存在于不同簇内的 value 值, value 值不相同根据 value 值进行划分; value 值相同时,比较该数据对象与不同簇中若干相似度大于相似度均值 \overline{X}_c 的数据对象的相似度的平均值,该平均值不相同根据相似度平均值划分,否则该点属于噪声点。

为了清楚地描述上述边界数据对象分配策略,采用图 1 的实例进行说明。在图 1 中:(1)若边界数据对象 M 与初分配聚类簇 Cluster-B 中 f_1, f_2, f_3 的相似度大于相似度均值 \overline{X}_c , 与初分配聚类簇 Cluster-C 中的 e_1, e_2 的相似度也都大于 \overline{X}_c , M 被同时放入两个初分配聚类簇 Cluster-B、Cluster-C 中。在 Cluster-B 中, M 的 value 值为 3, 在 Cluster-C 中, M 的 value 值为 2。此时,边界数据对象 M 应分配在 Cluster-B 中,删除 Cluster-C 中的 M ;(2)若边界数据对象 X 与 Cluster-A 中 b_1 的相似度大于 \overline{X}_c , 与 Cluster-A 外局部区域 LocalArea-Q 中 a_1, a_2, a_3 的相似度也都大于 \overline{X}_c 。在 Cluster-A 中, X 的 value 值为 1, 在 LocalArea-Q 中, X 的 value 值为 3。边界数据对象 X 应属于噪声数据,删除 Cluster-A 中的 X ;(3)若边界数据对象 Y 与初始聚类簇 Cluster-B 中 d_1, d_2, d_3 的相似度都大于 \overline{X}_c , 与 Cluster-B 外局部区域 LocalArea-P 中 c_1, c_2, c_3 的相

似度也大于 $\overline{X_c}$ 。由于数据对象 Y 的 value 值相等都为 3, 根据数据对象 Y 与若干相似度大于 $\overline{X_c}$ 的数据对象的相似度平均值进行判断, 若边界数据对象 Y 与 Cluster - B 中 d_1, d_2, d_3 的相似度平均值小于 LocalArea - P 中 c_1, c_2, c_3 的相似度平均值 (距离越大, 相似度越小), 边界数据对象 Y 应属于 LocalArea - P, 即噪声数据, 删除 Cluster - B 中的 Y 。

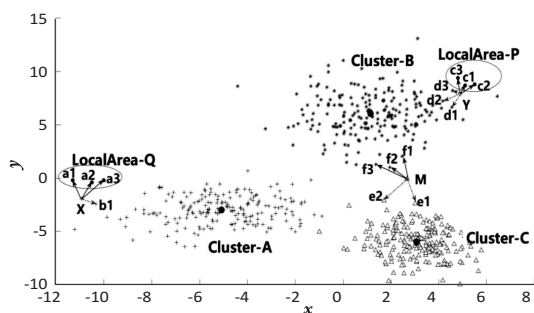


图 1 边界数据对象分配示意图

4 基于相似度均值的分类数据层次聚类分析算法 HCAS

依据第 2 节和第 3 节, 分类数据的层次聚类分析步骤为: (1) 将每一个数据对象视为一个聚类簇; (2) 根据公式 (1) - (2)、(4) - (9), 从第一个数据对象开始, 依次取完后剩余的所有数据对象, 计算两数据对象之间的相似度, 得到相似度矩阵。若相似度大于相似度均值 $\overline{X_c}$, 将两数据对象合并到一起, 反之, 不合并。此时, 每个聚类簇中至多含有两个数据对象。这样的合并方式导致有一个数据对象存在于两个甚至多个簇的问题, 针对此类情况根据边界数据对象分配策略进行调整; (3) 通过两两数据对象之间的相似度, 同步骤 (2) 的遍历方式, 依次将每一个簇与其后的每一个簇进行比较, 如果该聚类簇中有大于等于一个的数据对象与其他聚类簇中的若干数据对象相似度大于 $\overline{X_c}$, 将两个聚类簇合并。当然, 两个聚类簇相似的数据对象越多越好, 说明两个聚类簇越相似。如果一个聚类簇中的任一数据对象都不与其他聚类簇中的任一数据对象相似, 则不合并。其中, 对于一个数据对象存在于多个聚类簇的情况, 根据边界数据对象分配策略进行调整; (4) 重复步骤 (3), 直到所有的簇合并完成。其算法描述如下:

Algorithm: HCAS (Hierarchical clustering analysis for Classified data based on Average Similarity)

Input: 分类数据集 D

Output: Clusters

从数据集 D 提取 N ; // * N 为属性个数

For $i = 1$ to $|D|$ do // * 依据公式 (2), 计算相似度 $S_c(x, y)$, 并插入 List_Sim 中;

For $j = i + 1$ to $|D|$ do // * x, y 分别为第 i 个, 第 j 个数据对象;

For $z = 1$ to N do

利用公式 (4) - (8), 计算平稳相似性度量 $\theta_s(x_i, y_i)$;

利用公式 (1) - (2) 计算相似度 $S_c(x, y)$ 并插入

List_Sim 中;

End For

End For

End For

依据公式 (9), 利用 $S_c(x, y)$, 计算相似度均值 $\overline{X_c}$;

Clusters = { };

For all $S_c(x, y)$ in List_Sim do // * 依据相似度均值, 获得初始聚类簇, 并标记边界数据对象

If $S_c(x, y) > \overline{X_c}$ then // * 合并 x, y

Switch(Clusters 中的聚类簇 Q)

Case x 属于 Q : $Q = Q + \{y\}$, Break;

Case y 属于 Q : $Q = Q + \{x\}$, Break;

Case x 与 y 都不属于 Q ;

Clusters = Clusters + $\{x, y\}$, Break;

End Switch

End If

将出现在两个以上初始聚类簇中的数据对象 x , 在 List_B 中, 标记 x 为边界数据对象;

End For

For all 边界数据对象 x in List_B do // * 按边界数据对象分配策略分配边界数据对象

依据 3.2 节中边界数据对象分配策略, 将 x 重新分配到 Clusters 中的聚类簇;

End For

return Clusters

时间复杂性分析:

HCAS 算法的时间复杂度是衡量算法效率的重要指标^[31], 在 HCAS 中, 其算法的时间复杂度主要依赖于相似度计算以及层次聚类过程。首先计算相似度, 其时间复杂度是 $O(N \times n)$; 然后利用层次聚类, 自下而上, 遍历一次, 完成数据对象划分, 其时间复杂度为 $O(\frac{n(n-1)}{2})$ 。因此, 根据上述分析, 算法的总时间复杂度为 $O(N \times n + \frac{n(n-1)}{2})$ 。

5 实验结果分析

实验环境: CPU Intel® 酷睿 TMi7-4710MQ, 内存为 12 GB, 操作系统为 Microsoft Windows 10, 采用 java 语言, 在 jdk1.8、jre1.8、eclipse-jee-neon-3 环境下, 实现了 HCAS 算法与对比的分类数据层次聚类算法: ROCK^[7]、MGR^[24]、MTMDP^[25]、MNIG^[26]。

数据集: 人工合成和 UCI^[32] 数据集, 详见表 1。其中: Class 表示聚类个数, Instance 表示数据对象个数,

N 表示属性个数。

评价指标:准确率 Accuracy (ACC)、兰德指数 Rand Index (RI)、Fowlkes and Mallows Index (FMI)、F1 指数^[33-34]四个评价指标,更客观地衡量聚类效果。ACC 表示衡量分类正确的记录个数占总记录个数的比例;RI 表示衡量聚类结果与真实簇标签之间的相似性;FMI 表示两两精度和召回率的几何平均值;F1 表示准确率与召回率二者的调和均值。

实验对比的分类数据层次聚类算法参数设置:将对比算法 ROCK、MGR、MTMDP 与 MNIG 的聚类个数参数设置为正确的聚类个数,然后设置 ROCK 算法参数 θ 的取值区间,运行多次,选择最优的聚类效果作为最终的结果^[27]。实验数据集的预处理方式为:删除数据集中含有缺失属性值的记录;删除数据集中分布在不同类别中的相同数据对象。

表 1 数据集

Datasets	Data Sources	N	Class	Instance
Soybean Small	UCI	35	4	47
Zoo	UCI	17	7	101
Congressional Voting	UCI	16	2	435
Statlog(German Credit)	UCI	20	2	1 000
Online Shoppers Purchasing Intention	UCI	17	2	12 330
Internet Advertisements	UCI	1 558	2	3 279
default of credit card clients	UCI	23	2	30 000
Bank Marketing	UCI	16	2	45 211
Type1	人工合成	27	2	20 000

5.1 相似度均值的稳定性分析

在 HCAS 算法中,相似度均值 $\overline{X_c}$ 作为一种自动确定层次聚类阈值的选取方法,决定着数据的分配过程,

并影响着最终的聚类质量。为了实验验证相似度均值

$\overline{X_c}$ 选取的合理性与有效性,采用表 1 所示的 UCI 数据集,其实验结果如表 2 所示:

表 2 相似度均值 $\overline{X_c}$ 的稳定性

数据集	评价指标		$\overline{X_c}$					
Soybean Small	$\overline{X_c}$	$\overline{X_c} \approx 0.870$	0.860	0.865	0.875	0.880	0.885	0.890
	ACC	0.936 2	0.766 0	0.787 2	0.936 2	0.936 2	0.851 1	0.819 6
	FMI	0.879 7	0.668 1	0.688 1	0.879 7	0.902 8	0.786 5	0.746 0
	RI	0.936 2	0.837 2	0.837 2	0.936 2	0.951 0	0.888 1	0.820 5
	F1	0.936 7	0.769 0	0.769 0	0.936 7	0.935 9	0.848 1	0.788 3
Zoo	$\overline{X_c}$	$\overline{X_c} \approx 0.761$	0.760	0.765	0.770	0.775	0.780	0.785
	ACC	0.905 5	0.885 6	0.952 7	0.905 9	0.838 9	0.544 6	0.445 6
	FMI	0.890 3	0.859 4	0.953 6	0.911 7	0.805 9	0.790 3	0.694 5
	RI	0.898 7	0.882 6	0.909 3	0.915 5	0.842 1	0.870 1	0.793 5
	F1	0.859 8	0.859 1	0.929 6	0.890 8	0.774 6	0.482 6	0.374 6
Congressional Voting	$\overline{X_c}$	$\overline{X_c} \approx 0.809$	0.804	0.806	0.808	0.810	0.812	0.814
	ACC	0.932 8	0.839 2	0.859 7	0.906 4	0.947 4	0.964 9	0.657 9
	FMI	0.891 4	0.789 6	0.807 0	0.856 9	0.912 5	0.937 9	0.740 6
	RI	0.874 2	0.729 3	0.758 0	0.829 9	0.900 0	0.932 1	0.548 5
	F1	0.930 7	0.823 2	0.848 3	0.902 1	0.946 2	0.964 4	0.522 1
Statlog(German Credit)	$\overline{X_c}$	$\overline{X_c} \approx 0.633$	0.624	0.627	0.630	0.636	0.639	0.642
	ACC	0.945 0	0.974 0	1.0	0.976 0	0.910 0	0.872 0	0.838 0
	FMI	0.907 7	0.957 3	1.0	0.959 0	0.852 5	0.796 5	0.750 7
	RI	0.896 0	0.949 3	1.0	0.953 1	0.836 0	0.776 6	0.728 2
	F1	0.946 2	0.973 7	1.0	0.976 3	0.912 8	0.876 8	0.844 6

续表 2

数据集	评价指标	$\overline{X_c}$						
Online Shoppers Purchasing Intention Dataset	$\overline{X_c}$	$\overline{X_c} \approx 0.567$	0.564	0.565	0.566	0.568	0.569	0.570
	ACC	0.843 7	0.154 5	0.153 9	0.843 7	0.843 7	0.846 5	0.845 4
	FMI	0.858 0	0.859 0	0.859 3	0.858 0	0.858 0	0.859 4	0.858 9
	RI	0.736 2	0.738 8	0.739 6	0.736 2	0.736 2	0.740 0	0.738 6
	F1	0.772 1	0.041 8	0.041 7	0.772 1	0.772 1	0.778 8	0.776 3
Internet Advertisements	$\overline{X_c}$	$\overline{X_c} \approx 0.895$	0.893	0.894	0.896	0.897	0.898	0.899
	ACC	0.829 0	0.357 0	0.366 8	0.829 0	0.419 2	0.359 1	0.213 0
	FMI	0.846 4	0.675 1	0.668 5	0.846 4	0.638 7	0.673 7	0.800 1
	RI	0.716 4	0.540 7	0.535 2	0.716 4	0.512 8	0.539 5	0.664 6
	F1	0.751 5	0.363 2	0.376 8	0.751 5	0.445 4	0.366 1	0.131 7
default of credit card clients	$\overline{X_c}$	$\overline{X_c} \approx 0.524$	0.400	0.425	0.450	0.475	0.525	0.550
	ACC	0.778 9	0.456 2	0.778 9	0.778 9	0.778 9	0.778 9	0.472 8
	FMI	0.809 6	0.478 1	0.809 6	0.809 6	0.809 6	0.809 6	0.468 7
	RI	0.655 5	0.379 2	0.655 5	0.655 5	0.655 5	0.655 5	0.390 5
	F1	0.682 0	0.402 0	0.682 0	0.682 0	0.682 0	0.682 0	0.429 3
Bank Marketing	$\overline{X_c}$	$\overline{X_c} \approx 0.580$	0.575	0.585	0.590	0.595	0.600	0.650
	ACC	0.883 0	0.117 0	0.883 0	0.883 0	0.883 0	0.883 0	0.117 0
	FMI	0.890 7	0.890 7	0.890 7	0.890 7	0.890 7	0.890 7	0.890 7
	RI	0.793 4	0.793 4	0.793 4	0.793 4	0.793 4	0.793 4	0.793 4
	F1	0.828 2	0.024 5	0.828 2	0.828 2	0.828 2	0.828 2	0.024 5

表 2 的实验结果表明,由公式(9),计算得到相似度均值 $\overline{X_c}$,可以近似作为最优相似度阈值,取得较好的聚类效果,其聚类结果基本接近最优聚类效果,体现了方法的合理性与有效性。其主要原因:相度均值利用平稳相似性度量 $\theta_s(x_i, y_i)$,避免忽略重要属性维,反映了数据集中数据对象的集中分布。数据对象越接近聚类中心,其相似度越大,越容易合并为一个聚类簇。反之,数据对象越远离聚类中心,其相似度越小,

越容易分配到不同的聚类簇,并可能作为边界数据对象实现聚类分配。

5.2 聚类效果分析

为了验证 HCAS 算法的聚类性能,在 UCI 数据集上,给出了 ROCK、MGR、MTMDP、MNIG 和 HCAS 算法性能比较,其实验结果详见表 3,其中 k 表示聚类簇个数。

表 3 UCI 数据集上的聚类算法性能比较

数据集	算法	参数	ACC	FMI	RI	F1
Soybean Small	ROCK	$\theta = 0.58, k = 4$	0.803 1	0.786 2	0.792 2	0.811 6
	MTMDP	$k = 4$	0.723 4	0.700 7	0.501 4	0.792 2
	MGR	$k = 4$	0.361 7	0.500 7	0.250 7	0.192 2
	MNIG	$k = 4$	0.752 9	0.748 1	0.631 9	0.832 6
	HCAS	$\overline{X_M} = 0.880$	0.936 2	0.870 7	0.936 2	0.936 7
Zoo	ROCK	$\theta = 0.65, k = 7$	0.827 9	0.806 4	0.811 7	0.833 1
	MTMDP	$k = 7$	0.851 4	0.831 0	0.822 0	0.811 2
	MGR	$k = 7$	0.930 7	0.945 9	0.885 7	0.919 7
	MNIG	$k = 7$	0.939 4	0.945 9	0.892 3	0.920 5
	HCAS	$\overline{X_M} = 0.765$	0.952 7	0.953 6	0.909 3	0.929 6

续表 3

数据集	算法	参数	ACC	FMI	RI	F1
Congressional Voting	ROCK	$\theta = 0.61, k = 2$	0.779 1	0.762 6	0.757 3	0.794 8
	MTMDP	$k = 2$	0.848 7	0.803 3	0.798 1	0.829 0
	MGR	$k = 2$	0.827 5	0.780 8	0.713 7	0.808 4
	MNIG	$k = 2$	0.878 7	0.849 2	0.812 3	0.865 4
	HCAS	$\bar{X}_M = 0.812$	0.964 9	0.939 7	0.932 1	0.964 4
Statlog(German Credit)	ROCK	$\theta = 0.53, k = 2$	0.552 4	0.519 2	0.547 3	0.596 7
	MTMDP	$k = 2$	0.700 0	0.761 3	0.579 6	0.576 5
	MGR	$k = 2$	0.972 0	0.952 4	0.945 5	0.972 3
	MNIG	$k = 2$	0.987 6	0.967 9	0.952 3	0.982 3
	HCAS	$\bar{X}_C = 0.627$	1.0	1.0	1.0	1.0
Online Shoppers Purchasing Intention	ROCK	$\theta = 0.57, k = 2$	0.636 8	0.615 6	0.527 6	0.531 0
	MTMDP	$k = 2$	0.656 0	0.646 6	0.548 4	0.702 2
	MGR	$k = 2$	0.714 3	0.722 6	0.591 4	0.714 3
	MNIG	$k = 2$	0.730 0	0.728 3	0.605 4	0.735 3
	HCAS	$\bar{X}_C = 0.569$	0.846 4	0.859 4	0.740 0	0.778 8
Internet Advertisements	ROCK	$\theta = 0.77, k = 2$	0.458 0	0.441 1	0.432 6	0.469 4
	MTMDP	$k = 2$	0.672 1	0.642 7	0.555 6	0.711 6
	MGR	$k = 2$	0.827 9	0.844 2	0.712 6	0.749 9
	MNIG	$k = 2$	0.827 9	0.844 2	0.712 6	0.749 9
	HCAS	$\bar{X}_C = 0.895$	0.829 0	0.846 4	0.716 4	0.751 5
default of credit card clients	ROCK	$\theta = 0.38, k = 2$	0.722 0	0.717 4	0.621 0	0.646 2
	MTMDP	$k = 2$	0.770 2	0.803 6	0.645 8	0.670 2
	MGR	$k = 2$	0.764 4	0.748 0	0.631 3	0.663 7
	MNIG	$k = 2$	0.767 3	0.772 5	0.638 6	0.667 0
	HCAS	$\bar{X}_C = 0.524$	0.778 9	0.809 6	0.655 5	0.682 0
Bank Marketing	ROCK	$\theta = 0.49, k = 2$	0.624 9	0.601 1	0.622 5	0.637 2
	MTMDP	$k = 2$	0.617 9	0.651 0	0.527 5	0.683 2
	MGR	$k = 2$	0.783 9	0.790 2	0.661 0	0.775 2
	MNIG	$k = 2$	0.802 4	0.815 6	0.701 2	0.786 9
	HCAS	$\bar{X}_M = 0.580$	0.883 0	0.890 7	0.793 4	0.828 2

由表 3 可知,HCAS 算法在四种评价指标上的聚类性能都好于 ROCK、MGR、MTMDP、MNIG 算法。其主要原因是:在 HCAS 算法中,采用了平稳相似度度量,避免忽略重要属性维,使数据对象之间的相似度计算更加准确;HCAS 由于采用了边界数据对象分配策略,使得模糊边界数据对象被最大概率分配到正确的聚类簇中,提高了层次聚类分析的聚类质量;ROCK 算法对于相似度较小但属于一个聚类簇的边界数据对象不友好,从而导致边界数据对象分配不准确,聚类效果较差。MTMDP 算法由于提高了每次分裂叶子节点选

取的准确度,所以聚类效果较稳定。MGR 算法用度量分区之间的相似性,来构造与每个属性关联的分区尽可能接近的分区,所以聚类效果较好。MNIG 算法综合现有的聚类方法建立统一的分层聚类框架,并对框架的每一步进行改进,从 MNIR 选择的属性生成的分区中选择最好的分区,所以其聚类效果优于 MGR 算法。

5.3 抗噪性

为了验证 HCAS 算法的抗噪性,采用表 1 所示 Type1 数据集,分别添加 5%、10%、15% 的噪声数据,

比较 HCAS 算法的聚类性能指标,其实验结果如图 2 所示。

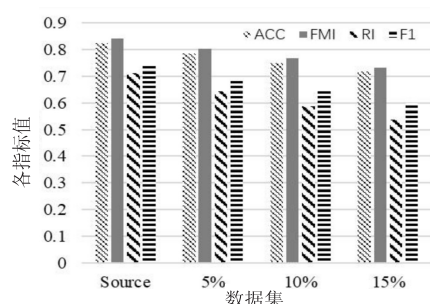


图 2 性能指标 (Type1 数据集)

由图 2 可知,随着数据集中噪声数据比例的增大,ACC、FMI、RI 和 F1 都呈现降低趋势。其原因是:随着噪声数据的增多,噪声数据干扰了相似度计算的准确性,使 HCAS 算法错将一部分噪声数据对象当作边界数据对象添加到聚类簇中,造成了评价指标的下降。随着在数据集中每增加 5% 的噪声数据,评价指标降低约 0.03 到 0.05 之间,噪声数据对聚类结果影响不大,HCAS 算法在有噪声的数据集中依然具有较高的聚类质量,说明 HCAS 算法具有良好的抗噪性。

6 结束语

针对层次聚类相似度阈值需要人为设定的问题,定义了一种相似度均值计算方法,并作为层次聚类簇合并或分裂的重要依据,有效解决了相似度阈值参数人为设定问题;采用相似度均值,给出了一种边界数据对象分配策略,并提出了一种基于相似度均值的分类数据层次聚类分析算法。该算法充分利用数据对象在数据集中的分布特点,自动确定相似度阈值,降低了人为因素的干扰,提高了聚类质量。下一步工作是针对混合属性数据层次聚类分析,研究相似度阈值的自动设定。

参考文献:

- [1] SLIMEN Y B, ALLIO S, JACQUES J. Model-based co-clustering for functional data [J]. *Neurocomputing*, 2018, 291: 97–108.
- [2] ZHOU Q, XIA B, XUE W, et al. An advanced inventory data mining system for business intelligence [C]//2017 IEEE third international conference on big data computing service and applications (BigDataService). Redwood City: IEEE, 2017: 210–217.
- [3] 冯浩哲, 张鹏, 徐欣楠, 等. 面向 3D CT 影像处理的无监督推荐标注算法 [J]. *计算机辅助设计与图形学学报*, 2019, 31(2): 183–189.
- [4] NANDA S R, MAHANTY B, TIWARI M K. Clustering Indian stock market data for portfolio management [J]. *Expert Systems with Applications*, 2010, 37(12): 8793–8798.
- [5] SINGH S, GANIE A H. Applications of picture fuzzy similarity measures in pattern recognition, clustering, and MADM [J]. *Expert Systems with Applications*, 2021, 168: 114264.
- [6] GUPTA A, WANG H, GANAPATHIRAJU M. Learning structure in gene expression data using deep architectures, with an application to gene clustering [C]//2015 IEEE international conference on bioinformatics and biomedicine (BIBM). Washington D. C.: IEEE, 2015: 1328–1335.
- [7] GUHA S, RASTOGI R, SHIM K. ROCK: a robust clustering algorithm for categorical attributes [J]. *Information Systems*, 2000, 25(5): 345–366.
- [8] ZOU Y, FANG L, DONG F, et al. Median-based thresholding, minimum error thresholding, and their relationships with histogram-based image similarity [C]//Sixth international conference on digital image processing (ICDIP 2014). Athens: International Society for Optics and Photonics, 2014: 915915.
- [9] YAN F, ZHANG H, KUBE C R. A multistage adaptive thresholding method [J]. *Pattern Recognition Letters*, 2005, 26(8): 1183–1191.
- [10] HU T, WU W, LIU L. Combination of hard and soft classification method based on adaptive threshold [C]//2014 IEEE geoscience and remote sensing symposium. Quebec City: IEEE, 2014: 4180–4183.
- [11] SUN H, CHEN S P, XU L P. Research on cloud computing modeling based on fusion difference method and self-adaptive threshold segmentation [J]. *International Journal of Pattern Recognition and Artificial Intelligence*, 2018, 32(6): 1859010.
- [12] HU L, LIU H, ZHANG J, et al. KR-DBSCAN: a density-based clustering algorithm based on reverse nearest neighbor and influence space [J]. *Expert Systems with Applications*, 2021, 186: 115763.
- [13] 张建朋, 裴雨龙, 刘聪, 等. 基于因子图模型的动态图半监督聚类算法 [J]. *自动化学报*, 2020, 46(4): 670–680.
- [14] 聂瑶瑶, 胡立华, 张继福, 等. 基于网格多密度的古建筑图像特征匹配方法 [J]. *计算机辅助设计与图形学学报*, 2020, 32(3): 437–444.
- [15] LONG X, WU S, CUI B, et al. Analysis of satellite observation task clustering based on the improved clique partition algorithm [C]//2019 IEEE congress on evolutionary computation (CEC). Wellington: IEEE, 2019: 1314–1321.
- [16] JAFARZADEGAN M, SAFI-ESFAHANI F, BEHESHTI Z. Combining hierarchical clustering approaches using the PCA method [J]. *Expert Systems with Applications*, 2019, 137: 1–10.
- [17] 周晨曦, 梁循, 齐金山. 基于约束动态更新的半监督层次聚类算法 [J]. *自动化学报*, 2015, 41(7): 1253–1263.
- [18] PANG Ning, ZHANG Jifu, ZHANG Chaowei, et al. Parallel hierarchical subspace clustering of categorical data [J]. *IEEE Transactions on Computers*, 2019, 68(4): 542–555.
- [19] ZHONG C, MIAO D, FRÄNTI P. Minimum spanning tree

- based split-and-merge;a hierarchical clustering method[J]. Information Sciences,2011,181(16):3397-3410.
- [20] BOUGUETTAYA A, YU Q, LIU X, et al. Efficient agglomerative hierarchical clustering[J]. Expert Systems with Applications,2015,42(5):2785-2797.
- [21] MURTAGH F, LEGENDRE P. Ward's hierarchical agglomerative clustering method; which algorithms implement ward's criterion? [J]. Journal of Classification, 2014, 31(3):274-295.
- [22] 刘旭, 易东云. 基于局部相似性的复杂网络社区发现方法[J]. 自动化学报, 2011, 37(12):1520-1529.
- [23] XIONG T, WANG S, MAYERS A, et al. DHCC: divisive hierarchical clustering of categorical data[J]. Data Mining and Knowledge Discovery, 2012, 24(1):103-135.
- [24] QIN H, MA X, HERAWAN T, et al. MGR: an information theory based hierarchical divisive clustering algorithm for categorical data[J]. Knowledge-Based Systems, 2014, 67:401-411.
- [25] LI M, DENG S, WANG L, et al. Hierarchical clustering algorithm for categorical data using a probabilistic rough set model[J]. Knowledge-Based Systems, 2014, 65:60-71.
- [26] WEI W, LIANG J, GUO X, et al. Hierarchical division clustering framework for categorical data[J]. Neurocomputing, 2019, 341:118-134.
- [27] 邱保志, 张瑞霖, 李向丽. 基于残差分析的混合属性数据聚类算法[J]. 自动化学报, 2020, 46(7):1420-1432.
- [28] BALCAN M F, LIANG Y, GUPTA P. Robust hierarchical clustering[J]. The Journal of Machine Learning Research, 2014, 15(1):3831-3871.
- [29] BAJCSY P, AHUJA N. Location-and density-based hierarchical clustering using similarity analysis[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(9):1011-1015.
- [30] 李隼韬, 钱志余. 一种两次归一化数据匹配方法: 中国, 200910183814.2[P]. 2009-12-30.
- [31] XIA C, HSU W, LEE M L, et al. Border: efficient computation of boundary points[J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(3):289-303.
- [32] UCI machine learning repository[EB/OL]. 2018-04-21. <http://archive.ics.uci.edu/ml/datasets.html>.
- [33] PANG Ning, ZHANG Jifu, ZHANG Chaowei, et al. PUMA: parallel subspace clustering of categorical data using multi-attribute weights[J]. Expert Systems with Applications, 2019, 126:233-245.
- [34] 庞宁, 张继福, 秦啸. 一种基于多属性权重的分类数据子空间聚类算法[J]. 自动化学报, 2018, 44(3):517-532.
- +++++
- (上接第 153 页)
- [28] PAN J H, GAO J, ZHENG W S. Action assessment by joint relation graphs[C]//Proceedings of the IEEE/CVF international conference on computer vision. Seoul: IEEE, 2019:6331-6340.
- [29] PARISI G I, MAGG S, WERMTER S. Human motion assessment in real time using recurrent self-organization[C]//2016 25th IEEE international symposium on robot and human interactive communication (RO-MAN). [s. l.]: IEEE, 2016:71-76.
- [30] NEKOUİ M, CRUZ F O T, CHENG L. EAGLE-Eye: extreme-pose action grader using detail bird's-eye view[C]//Proceedings of the IEEE/CVF winter conference on applications of computer vision. Waikoloa: IEEE, 2021:394-402.
- [31] PATRONA F, CHATZITOFIS A, ZARPALAS D, et al. Motion analysis: action detection, recognition and evaluation based on motion capture data[J]. Pattern Recognition, 2018, 76:612-622.
- [32] TANG Y, NI Z, ZHOU J, et al. Uncertainty-aware score distribution learning for action quality assessment[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Seattle: IEEE, 2020:9839-9848.
- [33] ZHANG B, CHEN J, XU Y, et al. Auto-encoding score distribution regression for action quality assessment[J]. arXiv: 2111.11029, 2021.
- [34] YU X, RAO Y, ZHAO W, et al. Group-aware contrastive regression for action quality assessment[C]//Proceedings of the IEEE/CVF international conference on computer vision. Waikoloa: IEEE, 2021:7919-7928.
- [35] PARMAR P, MORRIS B. Action quality assessment across multiple actions[C]//2019 IEEE winter conference on applications of computer vision (WACV). Waikoloa: IEEE, 2019:1468-1476.