计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.201000097

MLCPM-UC:一种基于模式实例分布均匀系数的多级 co-location 模式挖掘算法

刘新斌　王丽珍　周丽华
云南大学信息学院　昆明 650500
(xbliu126@163.com)

摘　要　空间 co-location(并置)模式是一组空间特征的子集,其实例在空间中频繁地邻近出现。由于空间数据同时存在关联性和异质性,co-location 模式实例的分布或在整个研究区域中全局出现(全局 co-location 模式),或在研究区域的局部区域出现(区域 co-location 模式),从而提出了多级 co-location 模式挖掘。当前的多级 co-location 模式挖掘方法存在两个问题:1)已有的多级 co-location 模式挖掘方法忽略了模式在空间中的分布特性,未能准确区分全局和区域 co-location 模式;2)已有的多级模式挖掘方法将全局非频繁 co-location 模式作为候选区域 co-location 模式,导致候选区域 co-location 模式数量过多。针对以上问题,首先,定义了模式的实例分布均匀系数,在考虑模式频繁性的同时考虑了模式在空间中的分布情况,从而正确、高效地识别出全局和区域 co-location 模式。其次,基于模式的实例分布均匀系数,设计了一个有效的多级 co-location 模式挖掘算法,提出了有效的剪枝策略以提高算法效率。最后,在真实和合成数据集上进行了广泛的实验,验证了所提方法的正确性和高效性。

关键词:空间数据挖掘;多级 co-location 模式;空间异质性;均匀系数
中图法分类号　TP311

MLCPM-UC:A Multi-level Co-location Pattern Mining Algorithm Based on Uniform Coefficient of Pattern Instance Distribution

LIU Xin-bin, WANG Li-zhen and ZHOU Li-hua
School of Information Science and Engineering, Yunnan University, Kunming 650500, China

Abstract　The spatial co-location pattern is a set of spatial features, and the instances frequently appear together in the spatial region. Due to the correlation and heterogeneity of spatial data, the distribution of co-location instances may appear globally in the whole research area (global co-location pattern), or appear in a local area of the research area (regional co-location pattern), Thus the multi-level co-location pattern mining is proposed. There are two problems with current multi-level co-location pattern mining methods: 1) the existing multi-level co-location pattern mining methods ignore the spatial distribution characteristics of patterns and fail to accurately distinguish global and regional co-location patterns; 2) the existing multi-level pattern mining method uses global non-prevalent co-location patterns as candidate regional co-location patterns, and the number of candidate patterns is too large. In response to the above problems, firstly, we define the uniform coefficient of the instance distribution of the co-location pattern and consider the pattern distribution in space while considering the pattern prevalence, so as to correctly and efficiently identify the global and regional co-location patterns. Secondly, a novel multi-level co-location pattern mining algorithm is designed based on the uniformity coefficient of the instance distribution of the pattern. In this algorithm, an effective pruning strategy is proposed to improve the efficiency of the algorithm. Finally, extensive experiments are carried out on real and synthetic data sets, which verify the correctness and efficiency of the proposed method.

Keywords　Spatial data mining, Multi-level co-location pattern, Spatial heterogeneity, Uniform coefficient

到稿日期:2020-10-15　返修日期:2021-01-25　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家自然科学基金项目(61966036,61662086,61762090);云南省创新团队基金项目(2018HC019);云南大学研究生科研创新基金项目(2020315)
This work was supported by the National Natural Science Foundation of China(61966036,61662086,61762090), Project of Innovative Research Team of Yunnan Province of China(2018HC019) and Yunnan University Graduate Research and Innovation Fund Project(2020315).
通信作者:王丽珍(lzhwang@ynu.edu.cn)

1 引言

近年来，遥感技术、全球定位系统和地理信息系统等空间信息技术得到了飞速发展，并被广泛应用于人们的日常生产生活中，导致每天都在产生大量的空间数据，这些数据均包含位置信息。空间 co-location 模式是一组空间特征的子集，其实例频繁出现在彼此的邻域中，互为邻居。空间关联规则挖掘是空间数据挖掘的主要任务之一，而本文研究的空间 co-location 模式挖掘是挖掘空间关联规则的一个特例，其研究在环境保护[1]、公共安全[2]、城市规划[3]、交通运输[4]和基于位置的服务[5]等领域得到了广泛应用。

由于空间数据的异质性，空间 co-location 模式被划分为在整个研究区域全局出现的全局 co-location 模式和在局部区域出现的区域 co-location 模式[6]。挖掘此类模式的方法被称为多级 co-location 模式挖掘，这种多级 co-location 模式挖掘将为不同空间现象之间的相互作用提供新的见解，并且具有广泛的适用性。例如，在生态学中，发现不同区域、不同物种、不同年龄和不同大小的物种之间的共生关系对于植物多样性保护和理解生态系统的动态特性至关重要。在犯罪学中，由犯罪事件和社会经济因素形成的多级 co-location 模式对于制定有效的警务策略以降低犯罪发生率至关重要。在城市规划中，从城市设施数据（如设施兴趣点）中发现的多级 co-location 模式对于决策者分析不同城市设施之间的联系以优化城市规划具有积极意义。

已有的空间 co-location 模式挖掘仅仅考虑了模式的实例是否在空间中频繁地邻近出现，未考虑模式的实例在空间中的分布情况，也就是仅依据参与度度量模式的频繁性。这带来了一个问题：模式参与度相同但模式实例分布特性不同的模式不能得到很好的区分。另一方面，虽然一些模式的全局参与度不满足参与度阈值，但是其在某个局部子区域的参与度满足阈值。然而，这些子区域是先验未知的，并且这些 co-location 模式的实例通常在研究区域中分布不均，因此区域 co-location 模式很难被完整地发现。

在现有研究中，识别多级 co-location 模式时遵循以下两个步骤。首先，确定全局 co-location 模式，将全局非频繁 co-location 模式识别为候选区域 co-location 模式。其次，根据每个候选区域 co-location 模式的实例，采用分区或者聚类的方法检测区域 co-location 模式实例存在的子区域。这种多级 co-location 模式挖掘方法存在两个问题：1）没有对全局非频繁 co-location 模式是否为潜在区域 co-location 模式进行判断，候选区域 co-location 模式中存在大量不属于区域 co-location 模式的候选模式，增加了区域 co-location 模式挖掘的负担；2）在确定全局频繁 co-location 模式时，存在把区域 co-location 模式错误地识别为全局 co-location 模式的情况。如图 1 所示，空间数据集中共有 3 个特征，特征 A 用圆表示，特征 B 用三角形表示，特征 C 用正方形表示，$A.1$ 代表特征 A 的第 1 个实例，以此类推。特征 A 有 8 个实例，特征 B 和特征 C 各有 4 个实例，用实线将存在空间邻近关系的两个实例连接起来，图 1 中的空间被划分成 4 个区域。依据参与度的定义可计算出空间 co-location 模式$\{A,B\}$和$\{A,C\}$的参与度均为 0.5。但是，从图 1 可以直观地看出，模式$\{A,B\}$的行实例在区域 1、区域 2 和区域 3 中都有分布，但模式$\{A,C\}$的行实例仅在区域 3 中有分布。如果参与度阈值 $\theta \geqslant 0.5$，已有的多级 co-location 模式挖掘方法将$\{A,B\}$和$\{A,C\}$识别为全局频繁 co-location 模式。从图 1 中可以看出，频繁模式$\{A,C\}$的实例仅出现在区域 3 中，应该属于区域 co-location 模式，而已有的多级 co-location 模式挖掘方法忽略了模式空间分布的特性，从而不能做出正确的判断。如果参与度阈值 $\theta \geqslant 0.6$，已有的多级 co-location 模式挖掘方法会将不满足参与度阈值的模式$\{A,B\}$和$\{A,C\}$作为候选区域 co-location 模式，而从图 1 中可以看出，模式$\{A,B\}$在研究区域中的分布相对均匀，不符合区域 co-location 模式在空间中的分布特性，不应将其视为候选区域 co-location 模式。

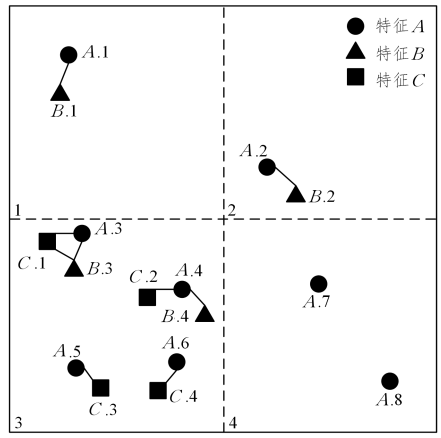

图 1 一个示例空间数据集

Fig. 1 An example of spatial data sets

针对已有的多级 co-location 模式挖掘方法存在的问题，本文定义了模式的实例分布均匀系数，根据模式的实例分布均匀系数提出了全局 co-location 模式，避免了已有的多级 co-location 模式挖掘方法中把区域 co-location 模式错误地识别为全局 co-location 模式的情况。另外，基于模式的实例分布均匀系数，提出了带有剪枝操作的多级 co-location 模式挖掘算法，该算法不仅提高了识别全局和区域 co-location 模式的正确性，而且解决了现有多级 co-location 模式挖掘算法耗时长的问题。

本文的主要贡献包括 3 个方面：

（1）针对已有的多级 co-location 模式挖掘方法忽略了模式实例的空间分布特性，存在误把区域 co-location 模式识别为全局 co-location 模式的问题，本文提出了基于模式实例分布均匀系数的全局 co-location 模式度量新方法。

（2）文献[7-8]中的多级 co-location 模式挖掘方法将全局非频繁 co-location 模式作为候选区域 co-location 模式，未对候选区域 co-location 模式进行有效的剪枝。本文基于模式实例分布均匀系数提前将无意义的候选区域 co-location 模式剪枝，极大地提高了多级 co-location 模式挖掘的效率。

（3）在真实和合成数据集上进行了大量实验，与 Deng 等[7]提出的多级 co-location 模式挖掘算法进行对比，证明了本文算法挖掘全局和区域 co-location 模式的正确性及高效性，解决了现有多级 co-location 模式挖掘算法耗时长和挖掘

结果不准确等问题。

本文第 2 节介绍了相关工作；第 3 节给出了基本概念及问题的形式化；第 4 节详述了多级 co-location 模式挖掘算法；第 5 节给出了实验与结果分析；最后总结全文。

2 相关工作

空间 co-location 模式挖掘首先由 Shekhar 等[9]和 Morimoto 等[10]提出，随后学者们从各个角度提出了各种空间 co-location 模式挖掘的算法[8-24]。

Join-based 是基于先验原理设计的最经典的挖掘算法[10]。之后，为了减少识别候选 co-location 模式实例的计算成本，研究人员开发了各种过滤和修剪方法。如，Yoo 等提出了 partial-join 算法[11]和 join-less 算法[12]，Xiao 等提出了基于密度的挖掘算法[13]，Wang 提出了基于有序团的挖掘算法[14]，Yao 等提出了基于稀疏图和压缩树的算法[15]，Yoo 等还提出了基于 MapReduce 的并行空间 co-location 挖掘算法[16]等。

为了准确客观地评估 co-location 模式的频繁性，研究者考虑了距离衰减效应[17]。Cai[18]提出了基于统计检验方法的挖掘 co-location 模式。2019 年，Zhou 提出了一种基于人类色彩感知的 2 阶 co-location 模式的度量和挖掘算法[19]。

为了提高这些挖掘技术的适应性，一些学者提出了自适应挖掘方法。如，Wan 等[20]提出了一种基于 k 最近特征的 co-location 挖掘算法，其中实例之间的邻近关系取决于最近的对象和 k 的数量。Sundaram 等[21]使用了 Delaunay 三角剖分来查找 co-location 实例，该方法将由三角形边缘连接的节点视为邻居。

为了挖掘区域 co-location 模式，研究人员主要提出了两种方法，一是基于分区的方法，二是基于聚类的方法。现有的基于分区的方法的主要挑战在于如何标识局部区域的空间分区方案。Celik 等[22]提出了基于四叉树的分区方案来划分研究区域。Qian[23]开发了一种基于 k 近邻的划分方法，该方法降低了确定划分方案的主观性。由于基于分区的方法忽略了数据的空间分布，可能破坏潜在的区域 co-location 模式，因此 Cai 等[24]提出了基于聚类的方法，通过聚类对空间对象或 co-location 实例进行分组来识别区域 co-location 模式。基于聚类的方法将区域 co-location 模式的局部性视为由参与该模式的空间特征的集中且并置的实例形成的集群。但是，Li 等[6]发现，基于聚类的方法可能会忽略一些没有对象或 co-location 实例集中的真实区域 co-location 模式，并建议枚举具有不同大小的最小正交边界矩形以识别矩形区域中所有可能的区域 co-location 模式。

2017 年，应用多级 co-location 模式挖掘算法识别区域 co-location 模式的思想被提出。Deng[8]提出了多级方法。首先挖掘全局 co-location 模式，将全局非频繁 co-location 模式识别为候选区域 co-location 模式；其次应用了一种自适应空间聚类方法来检测区域 co-location 模式频繁存在的子区域。2020 年，Deng 等提出了新的多级方法挖掘全局和区域 co-location 模式[7]。但是，该方法将所有全局非频繁 co-location 模式识别为候选区域 co-location 模式，通过候选区域 co-location 模式的实例来确定局部区域的位置，且该算法没有对候选区域 co-location 模式进行剪枝，效率极其低下。另外，该方法没有考虑模式在空间的分布特性，存在将区域 co-location 模式错误地识别为全局 co-location 模式的情况。

本文提出的多级 co-location 模式挖掘算法解决了文献[7]中算法挖掘效率低的问题，同时提高了挖掘全局和区域 co-location 模式的正确性。

3 基本概念及问题的形式化

3.1 基本概念

空间特征代表了空间中不同种类的事物。空间特征集是空间中不同种类的事物的集合，记为 $F=\{f_1, f_2, \cdots, f_n\}$。不同空间位置上分布着不同空间特征所对应的对象，这些对象被称为空间实例。空间实例集是空间实例的集合，记为 $S=S_1 \cup S_2 \cup \cdots \cup S_n$，其中 $S_i(1 \leqslant i \leqslant n)$ 表示空间特征 f_i 的实例集合。为了区别所有的空间实例，给每个空间实例指定一个唯一的编号，于是每个空间实例包含的信息通常有〈实例特征，实例号，空间位置〉。

如图 2 所示，空间数据集中共有 4 个空间特征，即 A，B，C，D，分别表示银行、餐馆、学校和医院。如果定义空间邻近关系为欧几里得距离，设 d 为空间邻近距离阈值，若 $distance(A.1, B.4) \leqslant d$，则称实例 $A.1$ 和 $B.4$ 满足空间邻近关系 R，表示为 $R(A.1, B.4)$，将满足邻近关系 R 的实例用一条实线进行连接。

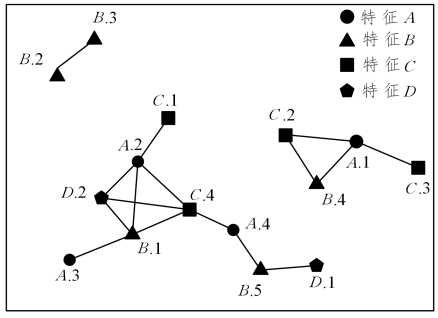

图 2 另一个空间数据集示例

Fig. 2 Another example of spatial data sets

团是一个完全连通子图，如图 2 所示，实例集$\{A.1, B.4, C.2\}$就是一个团。

一个空间 co-location 模式 c 是空间特征集 F 的一个子集，即 $c=\{f_1, f_2, \cdots, f_k\} \subseteq F$。co-location 模式 c 的长度称为此 co-location 模式的阶，记作 $size(c)=|c|$。例如，$\{A, B\}$是一个二阶模式。

若一个团 I 包含了 co-location 模式 c 中的所有特征，并且 I 的任何一个子集都不能包含 c 中的所有特征，那么 I 被称为 co-location 模式 c 的一个行实例[25]。co-location 模式 c 的所有行实例的集合称为表实例，记为 $T(c)$。图 2 中，空间 co-location 模式 $\{A, B, C\}$ 的表实例为 $\{\{A.1, B.4, C.2\}, \{A.2, B.1, C.4\}\}$。模式$\{A, C\}$是模式$\{A, B, C\}$的子模式，模式$\{A, B, C\}$是模式$\{A, C\}$的超模式。

为了衡量 co-location 模式的频繁性(有趣程度)，在空间数据挖掘中使用参与度[11]来度量模式的频繁性。

给定一个空间 co-location 模式 $c=\{f_1,f_2,\cdots,f_k\}$，空间特征 $f_i(1\leqslant i\leqslant k)$的参与率用 $PR(c,f_i)$表示，它是特征 f_i在空间 co-location 模式 c 中实例不重复出现的个数与总实例个数的比率[26]，即：

$$PR(c,f_i)=\frac{|\pi_{f_i}(T(c))|}{|T\{f_i\}|} \tag{1}$$

其中，π 是关系的投影操作。co-location 模式 c 的参与度是 c 中所有空间特征的 PR 值中的最小值，表示为 $PI(c)$，即：

$$PI(c)=\min_{i=1}^{k}\{PR(c,f_i)\} \tag{2}$$

指定最小参与度阈值为 θ，若 $PI(c)\geqslant\theta$，则称 co-location 模式 c 是频繁的。

特别地，在整个研究区域计算得到的参与度称为全局参与度，表示为 $GPI(c)$。在整个研究区域的局部区域计算得到的参与度称为区域参与度，表示为 $RPI(c)$。类似地，全局参与率可表示为 $GPR(c)$，区域参与率可表示为 $RPR(c)$。

例 1　图 2 中，特征 A 有 4 个实例，特征 B 有 5 个实例，特征 C 有 4 个实例，特征 D 有 2 个实例。模式 $c=\{A,C\}$的表实例为$\{\{A.1,C.2\},\{A.1,C.3\},\{A.2,C.1\},\{A.2,C.4\},\{A.4,C.4\}\}$，则 $GPR(c,A)=3/4$，$GPR(c,C)=4/4$，$GPI(c)=\min\{GPR(c,A),GPR(c,C)\}=0.75$。

引理 1(行实例的反单调性)　一个 co-location 模式的行实例的数目随着模式阶的增大而单调递减。

引理 2(模式的先验原理)　如果一个 co-location 模式是频繁的，那么它的所有子模式也是频繁的。相反，如果一个 co-location 模式是非频繁的，那么它的所有超模式也是非频繁的。

引理 1 和引理 2 的证明见文献[9]。

在考虑模式的空间分布特性时，本文受文献[26]的启发，考虑采用模式的实例分布均匀系数来描述模式在空间中的分布情况。但是，文献[26]采用模式的实例分布均匀系数仅能识别全局 co-location 模式，无法准确识别区域 co-location 模式及其所在区域。不同于文献[26]，本文仅采用模式的实例分布均匀系数来评估模式的空间分布情况，在计算模式的实例分布均匀系数时，本文将模式的每一个行实例抽象为一个“空间点”，这些“空间点”分布在整个研究区域的不同子区域中。下面将给出模式实例分布均匀系数的具体定义，以下定义参考了文献[26]的部分定义。

定义 1(区域行实例数)　给定一个 co-location 模式 c 的所有行实例，将分布于第 i 个空间区域中的行实例个数称为模式 c 的区域行实例数，表示为 $PINum_i(c)$。

图 1 中，模式$\{A,C\}$在区域 3 中的区域行实例数为 $PINum_3(\{A,C\})=4$。

假设将整个研究区域划分为 n 个子区域，可以考虑采用模式 c 的区域行实例数的均值和方差来描述模式的行实例在空间中的分布情况，即：

$$\overline{PINum(c)}=\frac{1}{n}\sum_{i=1}^{n}PINum_i(c)$$

$$s^2_{PINum(c)}=\frac{1}{n-1}\sum_{i=1}^{n}(PINum_i(c)-\overline{PINum(c)})^2$$

根据方差的定义可知，$s^2_{PINum(c)}$反映了模式 c 在各个子区域的区域行实例数与其均值的离散程度，一定程度上反映了模式的行实例在空间的分布特性。但是，其描述模式在空间的分布情况上仍然存在一些问题。例如，设 $n=6$，给定 2 个 co-location 模式 c_1和 c_2，模式 c_1和 c_2的区域行实例数在 6 个区域的分布情况分别为(2,1,1,0,1,1)和(8,4,4,0,4,4)，可以明显看出这两个模式有着相似的空间分布特性，但是通过计算可知 $s^2_{PINum(c1)}\neq s^2_{PINum(c2)}$。因此，本文定义了模式的区域行实例分布离散系数来描述模式的空间分布。

定义 2　将整个研究区域划分为 n 个子区域，一个 co-location 模式 c 的区域行实例分布离散系数定义为模式 c 的区域行实例数的方差与均值平方的比值，即：

$$PRSca(c)=\frac{s^2_{PINum(c)}}{\overline{PINum(c)}^2} \tag{3}$$

通过式(3)可计算得出 $PRSca(c_1)=2/5=PRSca(c_2)$，也就是说模式 c_1和 c_2在空间的分布特性是相似的。因此，模式的区域行实例分布均匀系数能很好地描述模式的行实例在空间的分布特性。$PRSca(c)$值越小，模式 c 的行实例在空间的分布越均匀，表明模式 c 在整个研究区域都频繁地邻近出现。相反，$PRSca(c)$值越大，模式 c 的行实例在空间的分布越集中于一个局部小区域，表明模式 c 在某个局部区域频繁地邻近出现。

虽然模式 c 的区域行实例系数很好地描述了模式的行实例在空间的分布特性，但是该系数的取值并不在闭区间[0,1]中，不利于用户指定阈值去评估模式的空间分布情况。因此，本文定义了模式的实例分布均匀系数，将 $PRSca(c)$的取值限定在[0,1]之间。

引理 3　模式 c 的区域行实例分布离散系数 $PRSca(c)$的最大值为 n。其中，n 为空间区域划分个数。

证明：

$$\begin{aligned}
S^2_{PINum(c)}&=\frac{1}{n-1}\sum_{i=1}^{n}(PINum_i(c)-\overline{PINum(c)})^2\\
&=\frac{1}{n-1}(\sum_{i=1}^{n}(PINum_i(c)^2-2PINum_i(c)\overline{PINum(c)}+\overline{PINum(c)}^2)\\
&=\frac{1}{n-1}(\sum_{i=1}^{n}PINum_i(c)^2-2\overline{PINum(c)}\sum_{i=1}^{n}PINum_i(c)+n\overline{PINum(c)}^2)\\
&=\frac{1}{n-1}(\sum_{i=1}^{n}PINum_i(c)^2-2\overline{PINum(c)}\times n\overline{PINum(c)}+n\overline{PINum(c)}^2)\\
&=\frac{1}{n-1}(\sum_{i=1}^{n}PINum_i(c)^2-n\overline{PINum(c)}^2)\\
PRSca(c)&=\frac{S^2_{PINum(c)}}{\overline{PINum(c)}^2}\\
&=\frac{\frac{1}{n-1}(\sum_{i=1}^{n}PINum_i(c)^2-n\overline{PINum(c)}^2)}{\overline{PINum(c)}^2}\\
&=\frac{1}{n-1}\left(\frac{\sum_{i=1}^{n}PINum_i(c)^2}{\frac{1}{n^2}(\sum_{i=1}^{n}PINum_i(c))^2}-n\right)\\
&\leqslant\frac{1}{n-1}(n^2-n)=n
\end{aligned}$$

证毕。

定义 3 将数据空间划分为 n 个区域，一个 co-location 模式 c 的实例分布均匀系数的计算式如下：

$$eve(c)=1-\frac{PRSca(c)}{n}=1-\frac{s^2_{PINum(c)}}{n\times\overline{PINum(c)}^2} \tag{4}$$

由引理 3 可知，模式 c 的实例分布均匀系数的取值在闭区间[0,1]中，与参与度的取值范围一致。因此，该系数便于用户评估一个模式的空间分布特性。$eve(c)$越大，模式 c 的行实例在空间中的分布越均匀，$eve(c)$越小，模式 c 的行实例在空间中的分布越集中。

例 2 图 1 中，由式(2)计算得出 $GPI(\{A,B\})=GPI(\{A,C\})=0.5$，由式(4)计算得出 $eve(\{A,B\})=0.8$，$eve(\{A,C\})=0$。设参与度阈值 $\theta\geqslant 0.5$，模式实例分布均匀系数阈值 $\mu\geqslant 0.65$，容易判定模式$\{A,B\}$是一个全局频繁 co-location 模式，而模式$\{A,C\}$不是，模式$\{A,C\}$满足区域 co-location 模式的特性，需要进一步评估其区域频繁性。

3.2 问题的形式化

根据基本概念中的定义，挖掘多级 co-location 模式遵循以下两个步骤。1)给定参与度阈值和模式实例分布均匀系数阈值，将同时满足这两个阈值要求的 co-location 模式识别为全局频繁 co-location 模式。2)判断全局非频繁 co-location 模式是否满足设定的模式实例分布均匀系数阈值，若不满足，则将其作为候选区域 co-location 模式，并依据该模式的实例确定局部区域的位置，在局部区域评估其区域频繁性。

形式化描述如下。

给定：

(1)空间实例集 $S=S_1\cup S_2\cup\cdots\cup S_n$，其中 $S_i(1\leqslant i\leqslant n)$是对应空间特征 f_i的实例集合，每个对象 $o_j\in S_i$用三元组〈特征 f_i，实例编号 j，位置(x,y)〉表示，其中 $1\leqslant j\leqslant|S_i|$。

(2)参与度阈值 $\theta(0\leqslant\theta\leqslant 1)$。

(3)模式实例分布均匀系数阈值 $\mu(0\leqslant\mu\leqslant 1)$。

约束：

空间区域划分个数 n。

挖掘结果：

满足全局参与度 $GPI(c)\geqslant\theta$ 且 $eve(c)\geqslant\mu$ 的全局 co-location 模式集和满足区域参与度 $RPI(c)\geqslant\theta$ 且 $eve(c)<\mu$ 的区域 co-location 模式集。

目标：

准确识别全局和区域 co-location 模式，同时提高挖掘多级 co-location 模式的效率。

4 多级 co-location 模式挖掘算法

4.1 模式实例分布均匀系数的计算

由定义 3 可知，模式实例分布均匀系数的计算依赖于模式的区域行实例数的计算，而对分布在各个子区域的行实例计数的前提是对空间区域进行划分。根据是否考虑空间数据的分布及数据间的关系，将空间区域划分方法分为两大类，第一类是空间驱动的划分方法，第二类是数据驱动的划分方法。具体划分方法如下。

(1)空间驱动的划分方法

给定一个 $D\times D$ 的空间范围，若要将其划分成 n 个区域，最简单的方法就是将空间范围划分成一个个正方形或者长方形网格，这种区域划分方法被称为网格划分法。

网格划分法不仅操作简单，而且其划分效率高，在特定的应用场景中十分有效。在模式实例分布均匀系数的研究中，网格划分法不仅能快速高效地对空间区域进行划分，而且能将空间区域均匀地划分，能很好地描述模式的行实例在空间的分布情况。因此，本文采用网格划分法对空间区域进行划分是合理的。

(2)数据驱动的划分方法

考虑空间数据在空间的分布及数据的相互关系，也可以采用聚类方法划分空间区域。聚类方法是典型的数据驱动的划分方法。聚类划分方法将类似的空间数据对象划分到同一个类别中，k-Means 聚类算法是最常见且最有效的聚类方法。在计算模式实例分布均匀系数的研究中，引入聚类技术的目的是对空间区域进行划分。因此，聚类技术应满足空间区域划分效率高和划分后的区域不重叠的要求。

根据上述分析，结合引入聚类算法的目的和 k-Means 聚类算法的优缺点，本文采用 k-Means 算法对空间区域进行划分，聚类的个数就相当于需要划分区域的个数。为了便于与网格划分法进行对比，本文选取网格划分法得到的网格中心作为初始聚类中心。

对空间区域进行划分后，可以通过划分结果对模式的行实例计数，进而根据定义 3 计算模式实例分布均匀系数。

如图 3 所示，将空间数据采用网格划分法划分为 9 个正方形区域，模式的行实例对应分布在不同的空间区域中。通过对空间区域的划分，可以轻松计算得出模式的区域行实例数，根据图 3 的网格划分法可得出表 1 所列的行实例计数。

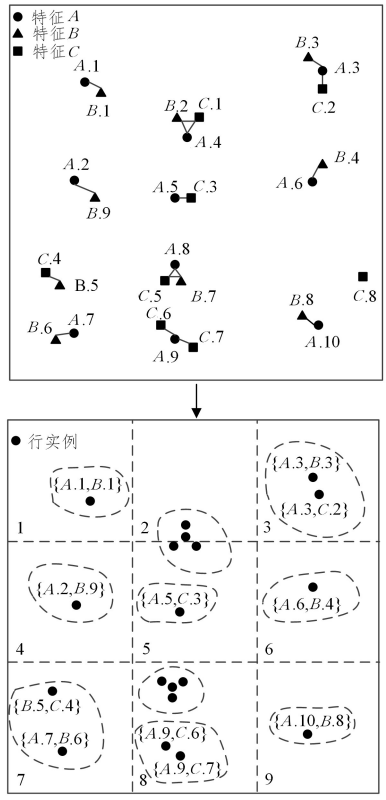

图 3 基于网格和聚类划分方法的行实例空间分布图

Fig. 3 Spatial distribution graph of row instances based on grid and clustering methods

表 1 行实例计数

Table 1 Row instance count

$PINum(c)$	区域号								
	1	2	3	4	5	6	7	8	9
$PINum(\{A,B\})$	1	0	1	1	1	1	1	1	1
$PINum(\{A,C\})$	0	0	1	0	2	0	0	3	0
$PINum(\{B,C\})$	0	1	0	0	0	0	1	1	0
$PINum(\{A,B,C\})$	0	1	0	0	0	0	0	1	0

如图 3 所示，本文将模式的行实例抽象为一个空间点。这样不仅便于确定局部区域的位置，而且避免了如何在模式的行实例跨区域时进行计数的问题。

图 3 中，不规则虚线圈出来的 9 个区域是采用聚类划分法得到的区域划分结果，同样可以通过聚类划分法对行实例进行计数，进而计算模式实例分布均匀系数。

4.2 带剪枝的多级 co-location 模式挖掘算法

本文提出基于模式实例分布均匀系数的多级 co-location 模式挖掘方法，更加准确地对全局和区域 co-location 模式进行了区分，提高了多级模式挖掘的准确性和可靠性。

研究分析发现，文献[7]中的多级方法选取全局非频繁模式作为候选区域 co-location 模式，需要计算这些候选模式的实例，再依据候选模式的实例挖掘区域 co-location 模式。这样的方法虽然消除了基于分区和聚类方法挖掘区域 co-location 模式的弊端，但也存在一些问题。该多级方法需要计算所有候选模式的实例，通过候选模式的实例计算其参与度，依据参与度确定该模式是全局频繁 co-location 模式还是作为候选区域 co-location 模式。传统方法（如 join-less 算法）可以通过引理 1 和引理 2 对候选模式进行剪枝，以达到减少识别候选 co-location 模式实例计算成本的目的。但是，该多级方法将所有全局非频繁 co-location 模式作为候选区域 co-location 模式，需要计算这些候选区域模式的实例以确定其所在的局部区域，进而识别该候选模式是否为区域 co-location 模式。因此，该方法需要花费大量时间来识别所有候选区域 co-location 模式是否为区域 co-location 模式。当空间特征数目增多时，候选区域 co-location 模式的数量随之增多，区域 co-location 模式挖掘效率下降，从而导致算法整体效率下降。

针对以上问题，本文提出了基于模式实例分布均匀系数的多级 co-location 模式挖掘算法（见算法 1）。算法 1 利用模式实例分布均匀系数对候选区域 co-location 模式进行剪枝，以达到减少候选区域 co-location 模式数量、提高算法效率的目的。

算法 1 多级 co-location 模式挖掘算法

输入：一个空间实例集 S；参与度阈值 θ；模式实例分布均匀系数阈值 μ

输出：满足 $GPI(c)\geqslant\theta$ 且 $eve(c)\geqslant\mu$ 的全局 co-location 模式集和满足 $RPI(c)\geqslant\theta$ 且 $eve(c)<\mu$ 的区域 co-location 模式集

变量：空间区域划分个数 n；空间特征集 F；区域划分后实例在每个区域的分布 InsDistribution；每个空间实例的邻居集 neighbor；所有空间特征的星型邻居集 SN；模式的阶 k；模式的表实例 TableInstance；模式的全局参与度 GPI；模式的区域参与度 RPI；局部区域范围 region；模式的实例分布均匀系数 eve

```
1. F,InsDistribution=DivideArea(n,S);
2. neighbors=create_neighbors(S);
3. SN=gen_star_neighborhoods(S,neighbors);
4. FOR (k=2; k≤F.length; k++)DO
5.    Candidates_k=combination(F,k);
6.    FOR EACH candidate IN candidates_k DO
7.       GPI,TableInstance=calPatter(candidate,SN);
8.       eve=calEve(InsDistribution,TableInstance); /* 识别全局 co-location 模式 */
9.       IF GPI(c)≥θ AND eve(c)≥μ DO
10.         candidate.level=Global; /* 对候选区域 co-location 模式剪枝 */
11.      ELIF GPI(c)<θ AND eve(c)≥μ DO
12.         CONTINUE; /* 识别区域 co-location 模式 */
13.      ELIF eve(c)<μ DO
14.         RPI,region=RCPMiner(TableInstance);
15.         IF RPI(c)≥θ DO
16.            candidate.level=Regional;
17.         END IF
18.      END IF
19.   END FOR
20. END FOR
```

算法 1 中，第 1 行根据用户指定的空间区域划分个数，选取一种空间区域划分方法对空间区域进行划分。第 2 行，为了便于与文献[7]作对比，依据文献[7]中的邻近关系构造方法创建所有的邻近实例对。第 3 行通过分组邻居实例对生成星型邻居集。对于星型邻居集的生成，文献[11]已做了详细介绍。第 4－5 行，根据特征实例集 F 和模式的阶数 k 生成 k 阶候选 co-location 模式集。第 7 行采用 join-less 算法，根据星型邻居集计算候选 co-location 模式的实例及其全局参与度。第 8 行根据候选 co-location 模式的实例和实例在每个区域的分布情况，计算候选 co-location 模式的实例分布均匀系数 eve。第 9－10 行判断候选 co-location 模式的全局参与度和模式实例分布均匀系数是否都满足相应的阈值，若满足，则将该候选 co-location 模式识别为全局频繁 co-location 模式，若不满足，则将该候选 co-location 模式作为候选区域 co-location 模式。另外，依据引理 1 和引理 2 可知，所有不满足全局参与度阈值的候选 co-location 模式，其超模式也不满足全局参与度阈值，也就是说，其超模式是全局非频繁 co-location 模式。可直接判断这些超模式的区域频繁性，而不考虑其全局频繁性。第 11－16 行考虑候选区域 co-location 模式的实例分布均匀系数，将不满足全局参与度阈值但满足模式实例分布系数阈值的候选区域 co-location 模式剪枝，不满足模式实例分布均匀系数阈值的模式则将其实例传入区域 co-location 模式挖掘算法中，以进一步评估该模式是否为区域 co-location 模式，若候选区域 co-location 模式的区域参与度满足阈值，则将该模式识别为区域 co-location 模式。伪代码中用 *RCPMiner*(*TableInstance*)表示区域 co-location 模式挖掘算法，该算法依据模式的实例构建候选区域 co-location 模式所在的区域，进而计算候选区域 co-location 模式的区域参与度。具体的区域 co-location 模式挖掘算法操作流程已在文献[7]中详细给出。文献[7]表明，在多级 co-location 模式挖掘算法中，相对于全局 co-location 模式挖掘而言，区域 co-location 模式挖掘耗时更长，因为区域 co-location 模式挖掘部分需要检查每个候选区域模式是否为区域 co-location 模式。因此，本文算法对候

选区域 co-location 模式进行有效剪枝能减少区域 co-location 模式挖掘耗时，进而提高多级 co-location 模式挖掘算法的效率。

4.3 复杂度分析

算法首先需要对空间区域进行划分，然后将所有的实例进行区域编号。设空间数据集总共有 N 个实例，空间区域划分个数为 m，则网格划分法的时间复杂度为 $O(m)$，聚类划分法的时间复杂度为 $O(N\times m\times t)$，t 表示迭代次数。区域划分的目的是为了计算模式的实例分布均匀系数，若 2 阶候选模式的平均行实例个数为 $|T_2|$，2 阶候选模式的总个数为 $|C_2|$，则计算 1 个 2 阶模式的实例分布均匀系数的时间复杂度为 $O(|T_2|)$，计算所有 2 阶候选模式的实例分布均匀系数的时间复杂度为 $O(|T_2|\times|C_2|)$。通过实验发现，计算模式实例分布均匀系数的耗时极短，可以忽略其时间和空间复杂度。

本文采用文献[7]中的邻近关系构造方法创建实例之间的邻近关系，其中使用了 KD-tree 搜索 k 最近邻，构造自然邻居的时间和空间复杂度由使用 KD-tree 的 k 最近邻居搜索控制。因此，这部分的时间复杂度约为 $O(N\log N)$，空间复杂度为 $O(N\log N)$，其中 N 是所有空间特征的实例数。当使用 join-less 算法生成候选 co-location 模式实例时，时间复杂度为 $O(N\log N)$，空间复杂度为 $O(\max\{|C_i|\times|I(C_i)|\})$，$|C_i|$ 是 co-location 模式 C_i 的大小，$|I(C_i)|$ 是模式 C_i 的实例数。由文献[7]可知，区域 co-location 模式挖掘部分的时间复杂度为 $O(N\log N+K^2)$，空间复杂度为 $O(N+K\log K)$，K 为区域 co-location 模式的实例数。可以看出，区域 co-location 模式挖掘是该方法中最耗时的部分，因此，对候选区域 co-location 模式剪枝能有效提高算法的挖掘效率。总体而言，本文方法的总时间复杂度约为 $O(N\times m\times t+N\log N+K^2)$，总空间复杂度为 $O(\max\{|C_i|\times|I(C_i)|\})+N+N\log N+K\log K)$。

5 实验与结果

本节通过在真实和合成数据集上的大量实验，验证了所提算法的效果，同时对本文算法与文献[7]中的算法进行了比较。实验中所涉及的算法均用 Python 语言实现。硬件环境为：Intel Core i7 3.70 GHz CPU，16 GB 内存。运行环境为：Microsoft Windows 10，PyCharm2019。

5.1 实验数据集

本文采用了文献[7]中的数据合成方法生成了一部分合成数据集，另外一部分合成数据集采用文献[27]提出的数据合成方法生成。表 2 列出了数据生成所需的参数及其含义。数据的具体生成流程如下：

(1)以距离阈值 d 为网格边长，对 $D\times D$ 的空间范围做网格化处理。

(2)根据用户需要设定生成特征的个数 F，用字母 A，B，C，…表示不同的特征，然后从 F 个特征中随机抽取 S 个特征组成一个主模式。

(3)随机选择一个网格，在该网格中生成 $I\times clumpy$ 个主模式的行实例。其中，主模式的行实例个数服从均值为 I 的泊松分布，主模式的长度服从均值为 S 的泊松分布。$clumpy$ 越大，数据点分布越稠密，团实例越多。最后，将剩余的实例均匀投放在整个空间区域。

表 2 合成数据生成所需参数信息

Table 2 Parameter information for synthetic data generation

参数	意义	默认值
S	主模式平均长度	3
P	主模式个数	10
I	主模式行实例的平均个数	15
$clumpy$	聚集度	1
F	特征个数	10
N	实例总个数	1 万
D	空间范围($D\times D$)	200

本文采用的真实数据集为北京某地的植被分布数据集，该数据集包含 6 种植被的空间分布，共有 18 140 个植株实例，数据分布如图 4 所示。

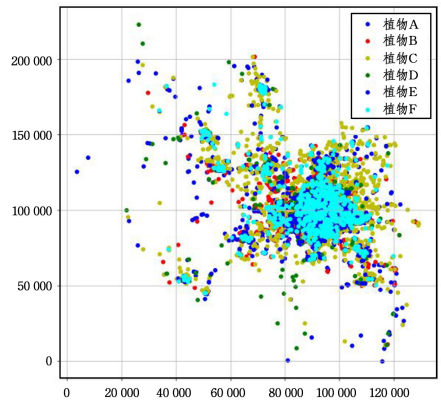

图 4 北京某地的植被数据分布图

Fig. 4 Beijing vegetation data distribution graph

5.2 实验分析

5.2.1 真实数据集上的实验分析

由于邻近关系采用文献[7]中的自然邻居作为邻近关系，考虑参与度阈值和模式实例分布均匀系数阈值的变化，比较多级 co-location 模式挖掘算法挖掘全局和区域 co-location 模式的效果和运行效率。

(1)区域划分方法和个数的影响

本实验中，固定模式的参与度阈值和均匀系数阈值分别为 0.75 和 0.65，设置区域划分个数为 10 到 40 进行实验，通过实验判断区域划分方法和区域个数对所提算法的影响。

随着区域数量的变化，所提算法采用基于网格划分方法和基于聚类划分方法[28]挖掘全局 co-location 模式的数量如图 5 所示。

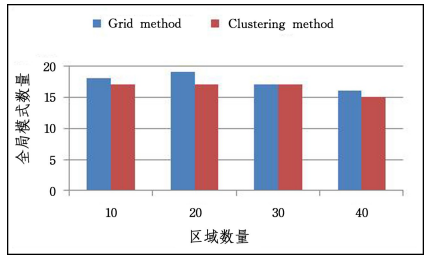

图 5 区域划分方法和区域数量对模式数量的影响

Fig. 5 Impact of regional division methods and quantities on the number of patterns

实验结果表明，两种区域划分方法在不同区域数量下，挖掘得到的全局 co-location 模式数量相近，并且区域划分的耗时远远少于后续步骤的耗时，不影响整个算法的运行时间。后续实验均采用聚类划分法划分区域。

(2)参与度阈值的变化

在本实验中，固定模式的实例分布均匀系数为 0.65，区域划分个数为 20，设置参与度阈值从 0.65 到 0.8 进行实验。

随着参与度阈值的变化，文献[7]中提出的多级 co-location 模式挖掘算法(MLMiner)和本文算法(算法 1)挖掘得到的全局和区域 co-location 模式的数量如图 6 所示。图 6 中，GCPs 表示全局 co-location 模式，RCPs 表示区域 co-location 模式。从图 6 中可以看出，随着参与度阈值的增大，通过 MLMiner 算法得到的 GCPs 减少，通过算法 1 得到的 GCPs 和 RCPs 也同样减少。由于算法 1 引入模式实例分布均匀系数度量，避免了 MLMiner 算法误把 RCPs 识别为 GCPs 的情况，如表 3 中的模式$\{A,C,D\}$，因此，算法 1 挖掘得到的 GCPs 数量略少于 MLMiner 算法。MLMiner 算法选取全局非频繁的模式作为候选区域 co-location 模式，通过文献[7]中的区域挖掘算法得到区域 co-location 模式，随着参与度阈值增大，候选区域 co-location 模式数量增加。由于文献[7]中的算法未考虑模式实例的空间分布特性，并且将模式的实例划分为不同的子区域，在每个子区域中检测该模式是否为区域 co-location 模式，因此，MLMiner 算法检测到的区域 co-location 模式数量远多于算法 1。但是，结合实际考虑，MLMiner 算法挖掘得到的区域 co-location 模式是不准确的，因为这些模式的实例不仅仅存在于某一个区域，也可能在整个研究区域分布得相对均匀。从实验结果可以看出，算法 1 挖掘得到的区域 co-location 模式数量随着参与度阈值的增加而减少，并且数量相对稳定。这是由于算法 1 引入了模式实例分布均匀系数，提高了挖掘区域 co-location 模式的准确性和挖掘效率。

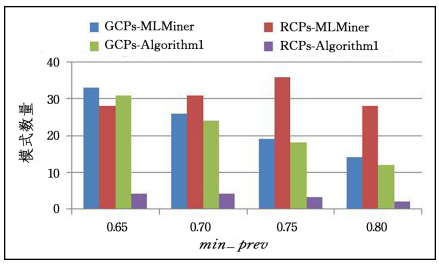

图 6　在不同参与度阈值下挖掘得到的模式数量对比

Fig. 6　Comparison of the number of patterns mined at different participation index thresholds

随着参与度阈值的增大，文献[7]提出的 MLMiner 算法和本文提出的算法 1 的挖掘耗时如图 7 所示。算法 1 的运行时间短于 MLMnier 算法。随着参与度阈值的增大，MLMiner 算法的运行时间大幅度增加。实验研究发现，随着参与度阈值的增大，MLMiner 算法挖掘得到的全局 co-location 模式减少，候选区域 co-location 模式增多，该算法需要花费大量的时间计算所有候选区域 co-location 模式的实例，并构建局部区域以评估候选区域 co-location 模式的区域频繁性。因此，算法的整体运行时间不断增加，大量时间花费在区域 co-location 模式挖掘上；从图 7 可以看出，随着参与度阈值的增大，算法 1 的运行时间上升趋势较缓，当参与度阈值为 0.65 时，算法 1 的效率是 MLMiner 算法的 3 倍，并且随着阈值增大，算法 1 始终保持良好的运行效率。

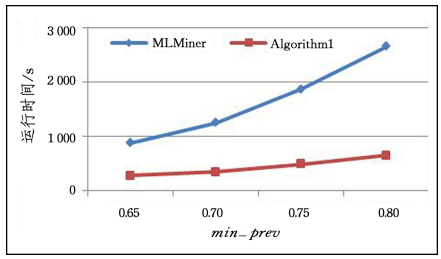

图 7　不同参与度阈值的运行时间比较

Fig. 7　Comparison of running time of different participation index thresholds

算法 1 随着参与度阈值的变化，其剪枝率如图 8 所示，随着参与度阈值的增大，剪枝率变高。由于算法 1 合理地使用了模式实例分布均匀系数对候选区域 co-location 模式剪枝，随着参与度阈值的增大，候选区域 co-location 模式数量增多，剪枝率也不断上升。实验结果表明，本文算法能有效地对候选区域 co-location 模式剪枝，提高算法的效率。

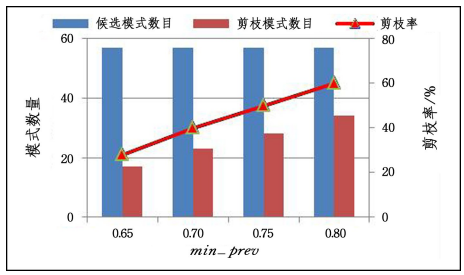

图 8　算法 1 的剪枝率

Fig. 8　Pruning rate of algorithm 1

表 3 列出了两种算法均在参与度阈值为 0.7、模式实例分布均匀系数为 0.5 的条件下挖掘的部分多级 co-location 模式。可以看出，通过算法 1 挖掘到的全局和区域 co-location 模式更具代表性，算法 1 能准确识别全局和区域 co-location 模式，提高挖掘结果的正确性。两种算法均将模式$\{A,D\}$识别为全局 co-location 模式，由于其全局参与度满足阈值，因此不作为候选区域 co-location 模式计算模式的区域参与度。从表 3 中可以看出，本文算法避免了将区域 co-location 模式误识为全局 co-location 模式的情况。例如，模式$\{A,C,D\}$的实例在研究区域的分布相对集中于某个局部区域时，MLMiner 方法仅考虑全局参与度，导致其错误地将该模式识别为全局 co-location 模式。由于 MLMiner 算法没有考虑到模式实例的分布特性，区域 co-location 模式挖掘结果中存在无意义的区域 co-location 模式。例如，模式$\{A,B,E\}$在研究区域中的分布相对均匀，在本文算法中不会被检测为区域 co-location 模式，而 MLMiner 算法将其识别为区域 co-location 模式。

表 3 MLMiner 和所提出的算法 1 挖掘到的多级 co-location 模式

Table 3 Multi-level co-location patterns mined by MLMiner and the proposed algorithm 1

Multil-level co-location patterns	The Proposed Method 1				The MLMiner Method		
	Level	GPI	RPI	eve	Level	GPI	RPI
{A,D}	Global	0.74	—	0.87	Global	0.74	—
{C,D}	Regional	0.49	0.72	0.25	Regional	0.49	0.72
{A,F}	—	0.37	—	0.78	Regional	0.37	0.77
{A,C,D}	Regional	0.76	0.83	0.23	Global	0.76	—
{A,B,E}	—	0.43	—	0.79	Regional	0.43	0.78

随着模式实例分布均匀系数阈值的变化，算法 1 挖掘得到的多级 co-location 模式数量如图 9 所示。随着模式实例分布均匀系数的增大，全局 co-location 模式数量减少，区域 co-location 模式数量增多。由于全局和区域 co-location 模式挖掘不仅依赖参与度阈值，模式实例分布均匀系数也会影响挖掘结果。模式实例分布均匀系数的阈值增大，全局频繁 co-location 模式减少，相应地候选区域 co-location 模式数量会增多，不满足模式实例分布均匀系数阈值的候选区域 co-location 模式也随之增多，导致检测到的区域 co-location 模式增多。

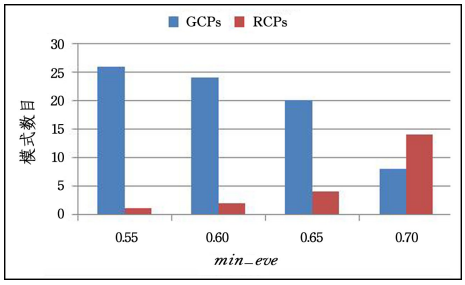

图 9 算法 1 在不同模式实例分布均匀系数阈值下挖掘到的模式数量

Fig. 9 Number of patterns mined under the threshold of uniform coefficient of distribution of different pattern instances by algorithm1

5.2.2 合成数据集上的实验分析

在合成数据集上的实验主要研究本文算法在改变空间特征数量和实例数量时的可扩展性[29]，这部分实验固定参与度阈值为 0.7，模式实例分布均匀系数阈值为 0.65。

(1)空间特征数量的影响

为了测试空间特征数量对多级 co-location 模式挖掘算法运行时间的影响，本文通过实验对比了在不同特征数量下两种算法的运行时间。为了便于与文献[7]中的 MLMiner 算法进行对比，合成数据集采用文献[7]的合成数据生成方法，生成了 6 个空间特征数量不同的数据集。随着特征数量的增多，MLMiner 算法与本文算法的运行时间如图 10 所示。可以看出，随着空间特征数量的增多，MLMiner 算法的运行时间呈现大幅度上升的趋势，而算法 1 的运行时间上升趋势较为缓慢，且算法 1 的运行时间明显短于 MLMiner 算法，并始终保持良好的运行效率。当特征数为 8 时，MLMiner 算法的运行时间远超 2000 s，MLMiner 算法需花费大量时间计算所有候选区域 co-location 模式的实例及构建局部区域所在位置，区域 co-location 模式挖掘的耗时较长。

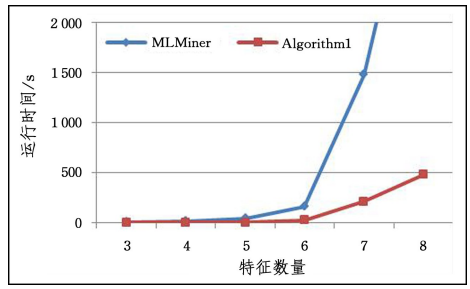

图 10 在不同特征数下的运行时间对比

Fig. 10 Comparison of running time under different feature numbers

(2)实例数的影响

实验模拟数据在 1 000×1 000 的范围内生成，采用文献[27]提出的数据合成方法生成了 5 个实例数不同的合成数据集，固定每个合成数据集的特征数为 3，实例数为 10 000～50 000。从图 11 中可以看出，随着实例数量的增加，MLMiner 算法挖掘多级 co-location 模式的耗时呈现较大幅度的上升趋势，而算法 1 挖掘多级 co-location 模式的耗时的上升趋势较缓，算法 1 的效率明显高于 MLMiner 算法，并始终保持着良好的执行效率。

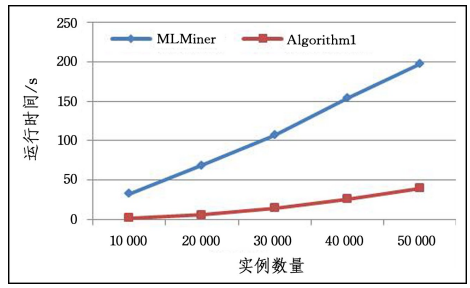

图 11 两种算法在不同实例数下的运行时间

Fig. 11 Running time of the two algorithms under different number of instances

(3)算法内存消耗对比实验

实验数据采用 5 个实例数不同的合成数据集，固定每个合成数据集的特征数为 3，实例数为 10 000～50 000。从图 12 中可以看出，随着实例数的增加，MLMiner 算法与算法 1 挖掘多级 co-location 模式所耗费的内存相近。算法 1 增加模式实例分布均匀系数的计算对内存耗费的影响很小。实例数小于 30 000 时，MLMiner 算法耗费的内存略小于算法 1，这是由于算法 1 在计算模式实例分布均匀系数时需要比 MLMiner 算法多耗费一些内存；实例数大于 30 000 时，MLMiner 算法耗费的内存略大于算法 1 且呈现上升趋势，这是由于算法 1 对候选区域 co-location 模式进行了有效剪枝，减少了区域 co-

location 模式挖掘部分的内存耗费。

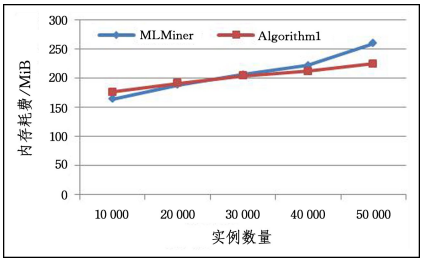

图 12　两种算法在不同实例数下耗费的内存

Fig. 12　Memory consumed by the two algorithms under different number of instances

已有的多级 co-location 模式挖掘算法仅考虑模式的参与度，未考虑到模式实例在研究区域的空间分布，因此在区分全局和区域 co-location 模式时缺乏准确性。本文基于模式实例分布均匀系数挖掘多级 co-location 模式，由于引入了模式实例分布均匀系数，在考虑参与度的同时考虑了模式在整个研究区域的空间分布。因此，本文提出的算法能更准确地区分全局和区域 co-location 模式，提高整个挖掘算法的准确率。本文还利用模式实例分布均匀系数对候选区域 co-location 模式进行了有效剪枝，提前删除了无意义的候选区域 co-location 模式，提高了区域 co-location 模式挖掘部分的效率。因此，本文算法的效率优于文献[7]中的 MLMiner 算法。

结束语　由于现有的多级 co-location 模式挖掘算法自身的特点，需要计算所有候选区域 co-location 模式的实例来构建局部区域，进而识别模式是否为区域 co-location 模式，未对候选区域 co-location 模式进行有效剪枝，区域 co-location 模式挖掘耗时长，导致整个算法的效率极其低下。另外，算法在区分全局和区域 co-location 模式时未考虑模式实例的空间分布特性，不能准确地识别全局和区域 co-location 模式。本文提出了基于模式实例分布均匀系数有效挖掘多级 co-location 模式的方法，该方法不仅挖掘到了更为合理的全局模式和区域模式，而且提高了算法的挖掘效率。本文在真实和合成数据集上进行了广泛的实验，将其与现有的多级 co-location 模式挖掘算法(MLMiner)在挖掘效果和效率上进行了全面的对比，验证了其正确性和高效性。在未来的工作中，计划考虑全局/区域概念的模糊特性，研究基于模糊度量指标的更有效的全局和区域 co-location 模式挖掘算法。

参 考 文 献

[1] GOREAUD F，PÉLISSIER R. Avoiding misinterpretation of biotic interactions with the intertype K12-function：population independence vs. random labelling hypotheses[J]. Journal of Vegetation Science，2003，14(5)：681-692.

[2] PHILLIPS P，LEE I. Mining co-distribution patterns for large crime datasets[J]. Expert Systems with Applications，2012，39(14)：11556-11563.

[3] YUE H，ZHU X，YE X，et al. The local co-location patterns of crime and land-use features in Wuhan，China[J]. ISPRS International Journal of Geo-Information，2017，6(10)：307.

[4] SAINJU A M，AGHAJARIAN D. Parallel grid-based co-location mining algorithms on GPUs for big spatial event data[J]. IEEE Transactions on Big Data，2020，6：107-118.

[5] YU W H. Spatial co-location pattern mining of facility points-of-interest improved by network neighborhood and distance decay effects[J]. International Journal of Geographical Information Science，2017，31(2)：280-296.

[6] LI Y，SHEKHAR S. Local co-location pattern detection：a summary of results[C]// Proceedings of the 10th International Conference on Geographic Information Science (GIScience 2018). Melbourne，Australia，2018：1-15.

[7] LIU Q，LIU W，DENG M，et al. An adaptive detection of multi-level co-location patterns based on natural neighborhoods[J]. International Journal of Geographical Information Science，2020(5)：1-26.

[8] DENG M. Multi-level method for discovery of regional co-location patterns[J]. International Journal of Geographical Information Science，2017，31(9)：1846-1870.

[9] SHEKHAR S，HUANG Y. Discovering spatial co-location patterns：a summary of results[C]// Advances in Spatial & Temporal Databases，International Symposium，SSTD. Redondo Beach，CA，USA，2001.

[10] MORIMOTO Y. Mining frequent neighboring class sets in spatial databases[C]// Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York：ACM，2001：353-358.

[11] YOO J S，SHEKHAR S. A partial join approach for mining co-location patterns[C]// Proceedings of the 12th ACM International Symposium on Advances in Geographic Information Systems. New York：ACM，2004：241-249.

[12] YOO J S，SHEKHAR S. A joinless approach for mining spatial colocation patterns[J]. IEEE Transactions on Knowledge and Data Engineering，2006，18(10)：1323-1337.

[13] XIAO X Y. Density based co-location pattern discovery[C]// Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM，2008：102-114.

[14] WANG L. An order-clique-based approach for mining maximal co-locations[J]. Information Sciences，2009，179(19)：3370-3382.

[15] YAO X J. A fast space-saving algorithm for maximal co-location pattern mining[J]. Expert Systems with Applications，2016，63：310-323.

[16] YOO J S，BOULWARE D，KIMMEY D. A parallel spatial co-location mining algorithm based on MapReduce[C]// Proceedings of the 3rd IEEE International Congress on Big Data. New York：IEEE，2014：25-31.

[17] YAO X J. A co-location pattern-mining algorithm with a density-weighted distance thresholding consideration[J]. Information Sciences，2017，396：144-161.

[18] CAI J N. Nonparametric significance test for discovery of net-

work-constrained spatial co-location patterns[J]. Geographical Analysis, 2019, 51 (1): 3-22.

[19] ZHOU M. A visualization approach for discovering colocation patterns[J]. International Journal of Geographical Information Science, 2019, 33 (3): 567-592.

[20] WAN Y, ZHOU J. KNFCOM-T: a k-nearest features-based co-location pattern mining algorithm for large spatial data sets by using T-trees[J]. International Journal of Business Intelligence and Data Mining, 2008, 3(4): 375-389.

[21] SUNDARAM V M, THNAGAVELU A, PANEER P. Discovering co-location patterns from spatial domain using a Delaunay approach[J]. Procedia Engineering, 2012, 38: 2832-2845.

[22] CELIK M, KANG J M, SHEKHAR S. Zonal co-location pattern discovery with dynamic parameters[C]// Proceedings of the 7th IEEE International Conference on Data Mining. IEEE, 2007: 28-31.

[23] QIAN F. Mining regional co-location patterns with kNNG[J]. Journal of Intelligent Information Systems, 2014, 42(3): 485-505.

[24] CAI J N. Adaptive detection of statistically significant regional spatial co-location patterns[J]. Computers, Environment and Urban Systems, 2018, 68: 53-63.

[25] FANG Y. Mining high quality spatial co-location patterns[D]. Kunming: Yunnan University, 2018.

[26] ZHAO J S. Research on mining spatial co-location patterns based on region partition[D]. Kunming: Yunnan University, 2018.

[27] ZHAO J, WANG L, YANG P, et al. Mining high utility co-location patterns based on importance of spatial region[C]// Proceedings of the International Conference on Geo-Spatial Knowledge and Intelligence. Singapore: Springer, 2017: 720-731.

[28] LIANG Z L, YUAN C A, QIN X, et al. Hot regzion mining appoach based on improved specral clustering[J]. Journal of Chongqing University of Technology (Natural Science), 2021, 35(1): 129-137.

[29] ZHANG P Z, ZHANG H Y. A review of features and labels dimensionality reduction methods of multi label data[J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2020, 37(5): 23-29.

LIU Xin-bin, born in 1996, postgraduate. His main research interests include spatial data mining and parallel computing.

WANG Li-zhen, born in 1962, Ph. D, professor, Ph. D supervisor, is a senior member of China Computer Federation. Her main research interests include spatial data mining, interactive data mining, big data analytics and their applications, etc.