(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 105740313 B
(45)授权公告日 2021.03.12

(21)申请号 201510994228.1
(22)申请日 2015.12.25
(65)同一申请的已公布的文献号
申请公布号 CN 105740313 A
(43)申请公布日 2016.07.06
(30)优先权数据
14307193.4 2014.12.27 EP
(73)专利权人 达索系统公司
地址 法国韦利济－维拉库布莱
(72)发明人 I·贝勒吉提
(74)专利代理机构 永新专利商标代理有限公司 72002
代理人 刘瑜 王英
(51)Int.Cl.
G06F 16/29(2019.01)
G06F 16/904(2019.01)
(56)对比文件
US 2009/0077221 A1,2009.03.19
CN 101859324 A,2010.10.13
CN 101976348 A,2011.02.16
Mark Polczynski etc..“Using the k-Means Clustering Algorithm to Classify Features for Choropleth Map”.《Cartographica The International Journal for Geographic Information and Geovisualization》.2014,第49卷(第1期),
XIAOLIN WU etc..“Optimal Quantization by Matrix Search”.《JOURNAL OF ALGORITHMS》.1991,
审查员 李玥

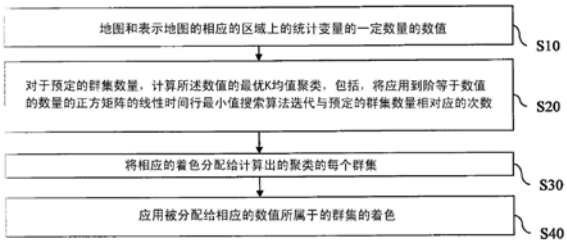
权利要求书2页 说明书11页 附图2页

(54)发明名称

设计等值域图

(57)摘要

本发明特别涉及设计等值域图的计算机实现的方法,其中,方法包括以下步骤:提供(S10)地图和表示地图的相应区域上的统计变量的一定数量(n)的数值(x_1,…,x_n);针对预定的群集数量计算(S20)数值的最优K均值聚类,其中,计算步骤(S20)包括将应用到阶等于数值的数量的正方矩阵的线性时间行最小值搜索算法迭代与预定的群集数量相对应的次数;将相应的着色分配(S30)给计算出的聚类的每个群集;以及在相应的数值被提供的地图的所有区域上,应用(S40)被分配给相应的数值所属于的群集的着色。这样的方法改进了等值域图的设计。



CN 105740313 B

1.一种设计等值域图的计算机实现的方法,其中,所述方法包括以下步骤:

• 提供(S10)地图和表示所述地图的相应区域上的统计变量的一定数量n的数值x_1,…,x_n;

• 针对预定的群集数量K来计算(S20)所述数值的最优K均值聚类,其中,计算步骤(S20)包括将应用到阶n等于所述数值的数量的正方矩阵H的线性时间行最小值搜索算法迭代与所述预定的群集数量相对应的次数;

• 将相应的着色分配(S30)给计算出的聚类的每个群集;以及

• 在相应的数值被提供的所述地图的所有区域上,应用(S40)被分配给所述相应的数值所属于的所述群集的着色,

其中,所述数值x_1,…,x_n被相应地分类和索引,并且所述计算步骤(S20)内的迭代包括,在每个相应的迭代等级k上,并且对于次于所述数值的数量n的每个相应的索引j,根据被应用到所述正方矩阵H的线性时间行最小值搜索算法,计算针对索引小于所述相应的索引,i<=j,的数值x_i的子集能够获取的最小总失真$TD_{min}(j,k)$,群集的数量与所述相应的迭代等级k相对应。

2.根据权利要求1所述的方法,其中,所述方法还包括提供预定的颜色,并且分配给相应的群集的着色是所述预定颜色的强度,所述强度取决于所述相应的群集的数值。

3.根据权利要求2所述的方法,其中,分配给相应的群集的所述着色是预定颜色的强度,所述强度取决于所述相应的群集的中心的值而递增。

4.根据权利要求1所述的方法,其中,在每个相应的迭代等级k,并且对于次于所述数值的数量n的每个相应索引j,对于每个行索引i和每个列索引j,矩阵项H(i,j)与以下的和相对应:

• 在针对所述行索引之前的索引i-1的之前的迭代中计算的最小总失真$TD_{min}(i-1,k-1)$以及

• 所述行索引和所述列索引之间的数值的连续的子集$(x_i,\cdots,x_j)$的失真disto(i,j)。

5.根据权利要求4所述的方法,其中,所述方法还包括,在每个相应的迭代等级k处,存储由所述行最小值搜索算法返回的索引$Cut_{min}(j,k)$。

6.根据权利要求5所述的方法,其中,所述方法还包括在所述计算步骤(S20)处,根据存储的索引来确定最优聚类。

7.根据权利要求6所述的方法,其中,根据所述存储的索引来确定所述最优聚类包括迭代地对所述数值进行划分,从所述存储的索引Cut_{min}中的最后被索引的数值$Cut_{min}(n,K)$开始,其中,在每个相应的迭代等级q处,当前形成的群集的起始数值的索引等于在所述计算步骤(S20)内的迭代期间,在等级K-q的迭代中存储的索引,所述迭代等级K-q等于所述预定的群集数量减去所述相应的迭代等级q,所述相应的迭代等级针对等于所述当前形成的群集的最后被所索引的数值的索引的行索引。

8.一种存储计算机程序的计算机存储介质,所述计算机程序包括用于执行权利要求1-7中的任何一项所述的方法的指令。

9.一种数据存储介质,其具有记录在其上的权利要求8所述的计算机程序。

10.一种设计等值域图的系统,所述系统包括耦合至存储器的处理器,所述存储器具有记录在其上的权利要求8所述的计算机程序。

11.一种能够通过权利要求1-7的任何一项所述的方法获得的等值域图。

12.一种数据存储介质,其具有记录在其上的权利要求11所述的等值域图。

设计等值域图

技术领域

[0001] 本发明特别涉及地图设计领域，并且特别涉及用于设计等值域图的计算机实现的方法、计算机程序、以及系统。

背景技术

[0002] 在地理信息系统(GIS)中经常使用等值域图。其涉及在地图上表示统计变量的问题，地图的每个区域上的统计变量的值由颜色强度表示。正在被显示的统计变量可以具有不同的特性(人口密度、居民汽车数量、降雨量、疾病比例)。

[0003] 在此上下文中，在这种表示中，在被选择的强度集合中进行选择是最初工作。用于这种选择的背景技术的两种主要算法是Jenks/Fisher的自然分解优化(如在Jenks的文章“The Data Model Concept in Statistical Mapping”，1967，以及Fisher的文章“On grouping for maximum homogeneity”，1958中所描述的)和头-尾分解(如在Lin和Yue的文章“A comparison study on natural and head/tail breaks involving digital elevation models”，2013，以及Jiang和Bing的文章“Head/tail breaks for visualizing the fractal or scaling structure of geographic features”，2014或“Ht-index for quantifying the fractal or scaling structure of geographic features”中所描述的)。这些解决方案实际上基于群集分析算法。使用群集分析的等值域图系统的其他示例包括文献US20050278182A1、US8412419B1、以及US8510080B2。

[0004] 等值域图的设计以一种方式涉及量化(即，根据预定距离，由预定的值的集合中的值中最接近的一个值来替换输入值)。实际上，地图的区域被分组到一起，并且针对这样的区域的统计变量的值相应地由该分组的代表值替换。这样的整体框架涉及更一般的群集分析领域。

[0005] 群集分析涉及将一组对象划分成分组(称为群集)的任务，以使得在每个分组中数据是相似的(参考Jain等的文章，“Data Clustering:A Review”)。其作为数据挖掘(参考文章Chen等的文章，“Data mining:an overview from a database perspective”)、机器学习(参考Murphy的书“Machine Learning,A Probabilistic Perspective”)、以及大规模搜索(参考Goodrum的文章“Image Information Retrieval:An Overview of Current Research”)中的核心问题而出现。群集分析是用于量化的重要工具：将中心分配各每个群集，就具有了由将每个点量化成其群集的中心构成的简单量化。

[0006] K均值聚类问题是群集分析的最著名的问题，并且作为用于脉冲编码调制的技术于1957年在贝尔实验室由Stuart Lloyd引入。Lloyd算法将p维点的集合作为输入，并且输出对这些点的划分，该划分的目标是使“总失真”最小化。该算法只是试探性的(其不提供最优聚类)。但实际上我们不能期望精确的算法，这是因为K均值聚类问题在非一维的情况下是NP困难的。当前，Lloyd算法仍然被广泛使用。已经提出了多种变型(参考J.A.Hartigan(1975)，“Clustering algorithms”，John Wiley&Sons,Inc.”)。

[0007] 一维应用特别重要。用于该问题的最著名的算法中的一个实际上就是在1967年中

开发的上文引用的Jenks自然分解优化(参考Jenks的书,“The Data Model Concept in Statistical Mapping”,国际制图年鉴(International Yearbook of Cartography)),并且出于制图的目的而被引入,如上文所述。与Lloyd算法一样,其只是试探性的。在2011年,Wang和Song开发了被称为CK均值的精确算法(参考Wang和Song的文章“Optimal k-means Clustering in One Dimension by Dynamic Programming”)。该算法是文献US1543036A的基础。该算法以时间O(K*n^2)运行,其中,K是所请求的群集的数量,并且n是实数的数量。甚至最近的(2013年),Maarten Hilferink已经开发出更加有效的算法,并且提供了该算法的实现。该实现实际上专用于制图,更准确地说,专用于等值域图,然而,该算法的唯一文档是维基百科页面(Fisher的自然分解分类,在优先权日期在下述URL可访问:http://wiki.objectvision.nl/index.php/Fisher%27s_Natural_Breaks_Classification)。

[0008] 然而,所有这些现有的方法都受到限制,因为它们不产生最优的K均值群集,或者太慢。在此上下文中,仍然存在对用于设计等值域图的改进的解决方案的需求。

发明内容

[0009] 因此提供了一种设计等值域图的计算机实现的方法。该方法包括提供地图和表示地图的相应区域上的统计变量的一定数量的数值的步骤。方法还包括针对预定的群集数量来计算数值的最优K均值聚类的步骤。该计算步骤包括将应用到阶等于数值的数量的正方矩阵的线性时间行最小值搜索算法迭代与预定的群集数量相对应的次数。方法还包括将相应的着色分配给计算出的聚类的每个群集的步骤。方法还包括在提供相应的数值的地图的所有区域处,应用被分配给群集的着色的步骤其中,所述相应的数值属于所述群集。

[0010] 所述方法可以包括以下中的一个或多个:

[0011] -所述方法还包括提供预定的颜色,并且被分配给相应的群集的着色是预定的颜色的强度,该强度取决于相应的群集的数值;

[0012] -被分配给相应的群集的着色是预定颜色的强度,该强度取决于相应的群集的中心的值而递增;

[0013] -数值被相应地分类和索引,并且计算步骤内的迭代包括,在每个相应的迭代等级上,并且对于次于数值的数量的每个相应的索引,根据应用到正方矩阵的线性时间行最小值搜索算法,计算针对索引小于相应的索引的数值的子集能够获取的最小总失真,群集的数量与所述相应的迭代等级相对应;

[0014] -在每个相应的迭代等级上,并且对于次于数值的数量的每个相应索引,对于每个行索引和每个列索引,矩阵项与在针对该行索引之前的索引的先前的迭代中计算的最小总失真,以及行索引和列索引之间的数值的连续的子集的失真之和相对应;

[0015] -该方法还包括在每个相应的迭代等级上,存储由所述行最小值搜索算法返回的索引;

[0016] -该方法还包括在计算步骤中,根据存储的索引来确定最优聚类;和/或

[0017] -根据存储的索引来确定最优聚类包括迭代地对数值进行划分,从存储的索引中的最后被索引的数值开始,其中,在每个相应的迭代等级上,当前形成的群集的起始数值的索引等于在计算步骤内的迭代期间,在等于所述预定的群集数量减去所述相应的迭代等级的等级的迭代处存储的索引,所述相应的迭代等级针对等于所述当前形成的群集的最后被

所索引的数值的索引的行索引。

[0018] 还提供了一种计算机程序，包括用于执行该方法的指令。

[0019] 还提供了一种计算机可读存储介质，其具有记录在其上的计算机程序。

[0020] 还提供了一种系统，包括耦合到存储器的处理器，所述存储器具有记录在其上的计算机程序。

[0021] 还提供了一种能够通过该方法获得的等值域图。

[0022] 还提供了一种数据存储介质，具有记录在其上的等值域图。

附图说明

[0023] 现在作为非限制性示例并且参考附图来描述本发明的实施例，在附图中：

[0024] -图1示出了方法的示例的流程图。

[0025] -图2示出了系统的示例；以及

[0026] -图3-4示出了方法。

具体实施方式

[0027] 参照图1的流程图，提出了设计(例如，绘制/产生/定义)等值域图的计算机实现的方法。该方法包括提供S10(例如，空白的和/或地理的)地图(例如，至少包括表示地理区域的线的2D视图)以及多个(例如，任何数量)数值的步骤，所述多个数值表示(例如，通过以下取得的值)在该地图的相应区域上的统计变量(即，任何人口/地理/地质/地缘政治/医疗/商业/营销变量，例如，人口密度、居民汽车数量、降雨量、疾病比例)，(即，数值的每个与表示地图的局部地区/区域的地图位置相关联，由此地图可以被提供有针对线和这样的位置的参考，例如，地图是地理区域的简单绘图并且数值与像素相关联，或者地图被提供作为更加复杂的数据段，其包括描述表示地球或地球的一部分的球体在方形、矩形、圆形或椭圆形平面上的投影的数据，所述数值随后被提供有与投影上的位置相对应的坐标)。所述方法还包括针对预定的群集数量，计算S20数值的最优K均值聚类的步骤。计算步骤S20包括将应用到阶等于数值的数量的正方矩阵的线性时间行最小搜索算法迭代与预定的群集数量相对应的次数。并且方法包括将相应的着色分配(S30)给计算出的聚类的每个群集的步骤，以及在提供了相应数值的地图的所有区域上应用(S40)被分配给相应的数值所属于的群集的着色，由此将视觉细节(即，着色)添加到地图上。这样的方法改进了等值域图的设计。

[0028] 值得注意的是，方法允许对统计数据的值进行聚类S20，以设计等值域图，这也是由现有技术已知的并且被提供的。由于方法通过计算S20最优K均值聚类来执行这样的聚类S30，因此方法设计了相对高质量的等值域图，正如地根据理信息系统的技术所公知的，所述地理信息系统通常目标在于进行K均值聚类以产生等值域图。但最重要的是，方法通过将应用到阶等于数值的数量的正方矩阵的线性时间行最小值搜索算法迭代与预定的群集的数量相对应的次数来执行这样的计算S20。由于由方法实现的该具体算法框架，最优K均值聚类的计算被快速地执行，如下文所详述的。

[0029] 方法是计算机实现的。这表示方法的步骤(或者基本上所有步骤)都是由至少一个计算机或者任何相似的系统来执行的。由此，方法的步骤是由计算机执行的，可能是全自动地(例如，除了提供S10之外的所有步骤)，或者，半自动地。在示例中，可以通过用户-计算机

交互来执行对方法的至少一些步骤的触发(例如,提供S10)。所需的用户-计算机交互的水平可以取决于预见的自动化水平,并且与实现用户的愿望的需求相平衡。在示例中,该水平可以是用户定义的和/或预定义的。

[0030] 方法的计算机实现的典型示例是利用适合于该目的的系统来执行方法。系统可以包括耦合到存储器的处理器,存储器具有记录在其上的包括用于执行方法的指令的计算机程序。存储器还可以存储适于维护由方法处理的数据的数据库。存储器是适于这样的存储的任何硬件,可能包括多个物理不同的部分(例如,一个用于程序,并且可能一个用于数据库)。

[0031] 图2示出了系统的示例,其中,系统是客户端计算机系统,例如,用户的工作站。示例的客户端计算机包括连接到内部通信总线1000的中央处理单元(CPU)1010、同样连接到该总线的随机存取存储器(RAM)1070。客户端计算机进一步被提供有与连接到总线的视频随机存取存储器1100相关联的图形处理单元(GPU)1110。视频RAM 1100在本领域中还被称为帧缓冲器。大容量存储设备控制器1020管理对大容量存储器设备(例如,硬盘驱动器1030)的访问。适合于有形地实现计算机程序指令和数据的大容量存储器设备包括所有形式的非易失性存储器,举例而言,包括诸如EPROM、EEPROM、以及闪速存储器等的半导体存储器设备、诸如内部硬盘和可移动盘等的磁盘;磁光盘;以及CD-ROM盘1040。前述中的任何一项可以由专门设计的ASIC(专用集成电路)来补充或者被集成到专门设计的ASIC当中。网络适配器1050管理对网络1060的访问。客户端计算机还可以包括触觉设备1090,例如,光标控制设备、键盘或其类似物。在客户端计算机中,使用光标控制设备来允许用户选择性地将光标定位在显示器1080上任何期望的位置处。此外,光标控制设备允许用户选择各种命令,以及输入控制信号。光标控制设备包括用于向系统输入控制信号的多个信号生成设备。通常,光标控制设备可以是鼠标,鼠标的按键用于生成信号。可替换地或除此之外,客户端计算机系统可以包括触摸板,和/或触敏屏幕。

[0032] 计算机程序可以包括可由计算机执行的指令,所述指令包括用于使上述系统执行方法的模块。程序在包括系统的存储器的任何数据存储介质上是可记录。程序可以例如在数字电子电路中、或者在计算机硬件、固件、软件或其组合中实现。程序可以被实现为装置,例如,有形地实现在用于由可编程处理器执行的机器可读存储设备中的产品。方法步骤可以由可编程处理器执行,所述可编程处理器通过对输入数据进行操作并且生成输出来执行指令程序,以执行方法的功能。处理器可以由此是可编程的并且被耦合以从数据存储系统、至少一个输入设备、以及至少一个输出设备接收数据和指令,以及将数据和指令发送到所述数据存储系统、至少一个输入设备以及至少一个输出设备。应用程序可以以高级过程或面向对象的编程语言来实现,或者在需要时以汇编或机器语言来实现。在任何情况下,语言都可以是编译的或解释的语言。程序可以是完全安装程序或更新程序。在任何情况下,该程序在系统上的应用产生用于执行方法的指令。

[0033] 方法提出了一种改进的聚类算法,其具体应用到等值域图设计。但是在详述方法的算法解决方案之前,现在详述其背景(即应用)。

[0034] 方法用于设计等值域图。

[0035] 如上文所述以及根据现有技术公知的,这种类型的地图是一种视图(通常为2D,但也可能是3D),其根据地理区域的子区域处的统计变量/测量的值,利用不同的着色(例如,

颜色/纹理)来表示所述地理区域的子区域,每个着色与统计的相应的值或连续范围相关联。在此上下文中,术语着色(例如,颜色/纹理)一般指定分布在所述子区域上的任何视觉信息。通常,着色是其强度为统计变量的增函数的颜色(例如,RGB或灰度)。但是,在棱柱地图(等值域图的特定情况)的情况中,着色也可以是与统计变量成比例的高度。图3示出了可以通过方法获得的等值域图的示例,不同的灰度强度被分配给地图的不同区域。一般而言,等值域图的设计可以基于预定的有序的并且一维的着色值的集合与统计变量的值之间预定的递增关系(由此与一维有序域相关联(如果不是被包含在其中))。

[0036] 在该方法的情况中,这样的地图的设计遵从现有技术的传统流水线。首先,方法在S10处开始于简单的等值线图(通常为空白的)以及与地图的相应区域相关联的统计变量的值。目标是以某种方式将统计的这种数值表示进行变换,以获得简单并且有意义的视觉化。这就是聚类所介入的地方。实际上,等值域图可以在理论上将不同的着色分配给统计变量的每个值,但是这对计算终端而言将不是繁重的,并且这对于用户而言也不是很有意义(地图将太过“繁杂”)。

[0037] 如已知的,从语义视点,K均值聚类是特别有利的,因为其以有意义的方式来聚集统计变量的值。由此,方法利用K均值聚类来对统计变量的值进行聚类(群集的数量取决于地图的查看者和/或设计者,并且,在该方法的上下文中,群集的数量是预定的),并且随后,与统计变量的值相关联的地图的每个区域(可能是所有区域,取决于在S10中提供的数据)现在与相应的群集(针对该区域的统计变量的值的群集)相关联。现在,由于方法将相应的着色分配S30给计算出的聚类的每个群集,因此方法可以通过S40来产生等值域图。这都是非常传统的做法,并且在上文引用的现有技术中被广泛解释。

[0038] 方法可以实现用于执行着色的任何的传统方式。一种传统的方式是提供预定的颜色,并且随后将该预定颜色的强度分配给相应的群集。例如,如果颜色是以RGB来提供的,则每个相应的强度与应用到RGB权重的因子(例如,在0和1之间)相关联。可以对灰度颜色执行相同的操作。预定颜色的强度可以取决于相应的群集的中心的值而递增。换言之,群集的中心(被分组到该群集中的统计变量的平均值)递增地定义所应用的强度(即,对于可视化区域,强度越大意味着统计变量的值越高)。

[0039] 可以应用很多其他方法,例如,在下面的URL中所描述的那些(在优先权日期是可访问的):

[0040] http://en.wikipedia.org/wiki/Choropleth_map#Color_progression.

[0041] 在示例中,方法可以简单地应用0(黑色)与255(白色)之间的不同灰度等级(如图3上所示)。例如,如果聚类的结果提供了群集中心$c_1,...,c_K$($c_1<c_2<..<c_K$),则方法可以应用将c_1映射到0(白色)并且将c_K映射到255(黑色)的仿射变换:$\varphi: c_i \rightarrow 255 * (c_i - c_1) / (c_K - c_1)$。

[0042] 所有这些都是非常传统的,并且不需要进一步解释。

[0043] 现在,由于方法找到了具有降低的算法复杂度(相对于现有技术的精确解决方案算法,并且甚至相对于合理试探法)的最优K均值聚类,因此其允许使用该最优K均值聚类来设计等值域图。这与现有技术不同,现有技术或者应用不产生最优K均值聚类的试探方法,或者在实践中是不可应用的(因为过高的复杂度)。值得注意的是,方法发现了对n大于1000(例如,10000左右)的值的最优K均值聚类,尤其是对于K高于50,甚至高于200,例如,256(用于灰度级值的8位编码)的值而言。例如,所述方法起作用。

[0044] 方法产生地理统计数据的最优量化，以使得颜色映射层是最优的(相对于作为标准目标函数的“总失真”而言)，比当前的其他最优量化器更快。实际上，该方法的计算时间与最好的试探方法(其产生非最优的量化)相当(并且通常甚至更快)。

[0045] 就此，已经提供了方法的背景以及方法的许多示例，但是尚未提供有关该方法的核心的细节(即，定义后面的设计S30-S40的计算S20步骤)。这在下文中完成，应当注意，下文提供的所有实现示例可以在上文提供的示例应用的任何一个中应用。

[0046] 如上文指示的，方法包括针对群集的预定数量K，计算S20数值的最优K均值聚类的步骤。但这不是暴力地完成的。实际上，计算步骤S20包括将线性时间行最小值搜索算法迭代与K相对应的次数(在下文讨论的示例中为K-1次)。由于其使用的是公知的行最小值搜索算法范畴的任何预定的线性时间算法，因此计算步骤S20具有低的复杂度。由此，方法实现了K均值聚类问题的新的并且具有算法效率的解决方案。

[0047] 在提供有关计算步骤S20的更多细节之前，现在讨论量化。实际上，在示例中，统计数值可以在S30-S40之前以某种方式被“量化”。实际上，每个区域的统计值可以被视为由通过方法与该区域相关联的群集的中心所替换，因为它们的相对应的区域被应用了该群集的着色。

[0048] 如已知的，标量量化是使用有限集$V = \{c_1, \ldots, c_K\} \subset \mathbb{R}$来近似真实值的计算工具，其中，V的元素(称为数字阶梯)是用作近似的值。标量量化器在数学上被定义为映射：

[0049] $$q: \mathbb{R} \longrightarrow V = \{c_1, \ldots, c_K\},$$

[0050] 这使得x和q(x)之间的距离小，该距离是任何预定距离(其中，距离的概念可以取决于上下文)，例如，欧几里得距离。

[0051] 在实现中，总是通过将$\mathbb{R}$划分成区间

[0052] $I_1 = [-\infty, a_1], I_2 = [a_1, a_2], \ldots, I_K = [a_{K-1}, \infty]$ ($a_1 < \cdots < a_{K-1}$)来定义标量量化器，并且随后，针对每一个$x \in \mathbb{R}$，将I_i指示为唯一区间，以使得$x \in I_i$，我们关联了$q(x) = c_i$。实数$a_1, \ldots, a_{K-1}$称为“决策边界”。Gray和Neuhoff的文章“Quantization”提供了对量化的全面调查。

[0053] 方法集中于K均值设置，这是公知的，并且在例如MacQueen的文章“Some Methods for classification and Analysis of Multivariate Observations”中定义，K均值设置是使用最为广泛的。在这样的设置中，方法在S20中，对于按照升序分类的给定元组$(x_1, \ldots, x_n) \in \mathbb{R}^n$并且对于给定的整数K(通常，$K = 2^b$，其中b是可用于对每个$x_i$进行编码的位数)找到实现最小总失真的采用K个数字阶梯的量化器q，总失真被定义为：

[0054] $$TD(q) = \sum_{1 \le i \le n} \left(x_i - q(x_i)\right)^2$$

[0055] 显然，要使该量最小化，方法可以仅处理将每个真实值x映射到其最接近的数字阶梯的量化器。因此，问题正好等价于找到使以下最小化的K个中心$c_1, \ldots, c_K$：

[0056] $$TD(\{c_1, \ldots, c_K\}) = \sum_{1 \le i \le n} \min_{1 \le k \le K} (x_i - c_k)^2$$

[0057] 图4示出了针对十个值和4个数字阶梯的示例。最优量化由集合{c_1,…,c_K}给出，以使得$\forall c'_1,\dots,c'_k, TD(\{c'_1,\dots,c'_k\}) \geq TD(\{c_1,\dots c_K\})$。

[0058] 必须将使总失真最小化理解为在量化步骤(对于给定的K而言)期间丢失尽可能少的信息。注意，每个点x_i被隐含地分配给其最接近的中心，由此，方法可以构建对群集的划分，其中，每个群集与分配给给定中心的点的集合相对应(寻找最优量化器由此成为聚类问题，如Lloyd在“Least square quantization in PCM”中解释的)。对于每个k∈{1,…,K}，让我们通过C_k表示与中心c_k相对应的群集。容易看出，每个中心实际上是其相对应群集的点的均值。此外，注意到，由于假设x_1＜…＜x_n，因此每个群集由连续的点的子集构成。例如，如果具有想要划分到K＝4个群集中的47个实数，则可能的最优聚类可以是：

[0059] $$(\underbrace{[x_1,\dots,x_{17}]}_{C_1},\underbrace{[x_{18},\dots,x_{24}]}_{C_2},\underbrace{[x_{25},\dots,x_{42}]}_{C_3},\underbrace{[x_{43},\dots,x_{47}]}_{C_4})$$

[0060] 对于所有的1≤a≤b≤n，我们引入概念$mean(a,b)=\frac{1}{b-a+1}\sum_{\{a\leq i\leq b\}}x_i$，并且我们还指示disto(a,b)＝$\Sigma_{\{a\leq i\leq b\}}(x_i-\text{mean}(a,b))^2$。之前的示例的相对应的总失真可以被写为：

[0061] TD＝disto(1,17)+disto(18,24)+disto(25,42)+disto(43,47)

[0062] 如前所述，针对该问题的解决方案已经存在，但是它们比该方法慢，并且因为它们当中的大多数是试探性的(即，不产生最优量化)。

[0063] 在方法的示例实现中，数值(x_1,…,x_n)被相应地分类和索引。计算步骤S20内的迭代包括，在每个相应的迭代等级k上，并且对于次于数值的数量n的每个相应的索引j，根据应用到正方矩阵H的线性时间行最小值搜索算法，计算对于索引小于j(由此i≤j)的数值x_i的子集的可获取的最小总失真，记为TD_{min}(j,k)群集数量k对应于相应的迭代等级(由此为k)。

[0064] 在该示例中，在每个相应的迭代等级k上，并且对于次于数值的数量n的每个相应的索引j，对于每个行索引i和每个列索引j，矩阵项H(i,j))可以简单地与下述项的和相对应(例如，等于)：

[0065] •在针对行索引之前的索引(i-1)的前面的迭代中计算的最小总失真(TD_{min}(i-1,k-1))，以及

[0066] •行索引和列索引之间的数值的连续子集(x_i,…,x_j)的失真(disto(i,j))。

[0067] 这样的实现提供了优于现有的聚类方法的系统，这是因为其同样产生了最优K均值聚类，但是运行得更快，特别地其时间复杂度为O(K*n)。注意，对于通常的使用，该示例的方法执行得比“好的”试探方法快十倍以上。

[0068] 现在将讨论示例的聚类算法的更加全面的概述。

[0069] 为了计算最优划分，方法使用动态规划范例(如在Bellman的文章“The theory of dynamic programming”中所描述的)。特别地，示例的方法针对每个j∈{1,…,n}和每个k∈{1,…,K}来计算被定义为最小总失真的值TD_{min}(j,k)，所述最小总失真值是在仅考虑前j个点(x_1,…,x_j)的情况下，利用最多k个群集能够获得的最小总失真值。

[0070] 根据定义，对所有的j∈{1,…,K}具有：TD_{min}(j,1)＝disto(1,j)，因为将一组点划分在一个群集中的唯一方式就是将它们全部纳入。此外，对于所有的k∈{2,…,k}以及对于

所有的j∈{1,…,n},具有以下的公式:

[0071] $$TD_{min}(j,k) = \min_{1\le i\le j}\{TD_{min}(i-1,k-1) + disto(i,j)\}$$

[0072] 该公式表达这样的事实,对于具有最多k个群集的$(x_1,\cdots,x_j)$,能够获得的最小总失真包括,对于一定的i,具有最多k-1个群集的前i-1个点的最优聚类以及作为最后一个群集的$[x_i,\cdots,x_j]$。之前的公式是方法的核心。注意,如果对于给定的k∈{2,…,k},已经针对所有的j计算了值$TD_{min}(j,k-1)$,则可以通过在之前的公式中试验所有可能的i∈{1,…,j}来针对所有的j计算值$TD_{min}(j,k)$。但是这种假设的技术将导致非常慢的算法。

[0073] 为了克服该问题,方法使用用于特定矩阵中的行最小值搜索的特定类别的算法。

[0074] 现在对示例的方法所依赖的行最小值搜索和完全单调性的概念进行讨论。

[0075] 行最小值搜索算法是(如在Bradford和Reinert的文章"Lower Bounds for Row Minima Searching",1996中所详述的)这样的算法,其将函数 $f:[\![1,R]\!]\times[\![1,C]\!]\longrightarrow\mathbb{R}$ 作为输入,使得对于所有的1≤i≤R,1≤j≤C,可以在恒定时间中计算出值f(i,j),并且输出整数向量p=$(p_1,\cdots,p_R)$,使得:

[0076] $$\forall 1\le i\le R, p_i = argmin_{1\le j\le C} f(i,j)$$

[0077] 在下文中,我们通过F来表示矩阵F=$(f(i,j))_{i,j}$。注意,出于完整性的原因,如果矩阵F不具有特殊属性,则可以请求其所有项,以便于计算向量p。然而,在有关F的某些条件下,可以实现极为更快的算法。

[0078] 如果矩阵F满足下述条件,则矩阵F被称为完全单调:对于i,j,k,i<j,如果有F(k,i)<F(k,j),则对于所有的k′≤k,也有F(k′,j)<F(k′,j)。

[0079] 存在用于完全单调矩阵中的行最小值搜索的线性时间算法(如在Alon和Azar的文章"Comparison-Sorting and Selecting in Totally Monotone Matrices"中所解释的)。方法可以在S20中在矩阵H上实现这样的预定算法(即,线性时间行最小值搜索算法)中的任一个。特别地,发明人已经使用在Alon和Azar的文章中介绍的广泛公知的SMAWK算法对方法进行了测试,具有极为快速的收敛(相对于现有技术)。

[0080] 现在将讨论允许方法极为快速地执行的基本特性。在此之前,应当注意,对该特性的认识使得在K均值聚类问题和被提供用于行最小值搜索的广泛公知并且强大的算法之间建立了桥梁,而在有关K均值聚类的研究的漫长历史中还没有认识到这样的桥梁。

[0081] 定理:

[0082] 对于所有的1≤i<j<n,

[0083] 具有:disto(i,j)+disto(i+1,j+1)≤disto(i,j+1)+disto(i+1,j)。

[0084] 证明:

[0085] 首先,注意对于1≤a≤b≤n,disto(a,b)根据定义等于$(x_s,\cdots x_b)$的方差乘以(b-a+1)。

[0086] 因此,根据König-Huygens公式,有:

[0087] $$disto(a,b) = \sum_{a\le i\le b} x_i^2 - \frac{1}{b-a+1}$$

[0088] 让我们考虑i和j,以使得1≤i<j<n。

[0089] 指定p＝(b-a+1)，S＝$\Sigma_{i\leq l\leq j}x_l$，α＝$x_j$+1，和β＝$x_i$，根据先前的恒等式，我们有：

[0090] $$disto(i,j)=\sum_{i\leq l\leq j}x_l^2-\frac{S^2}{p}$$

[0091] $$disto(i,j+1)=\sum_{i\leq l\leq j+1}x_l^2-\frac{(S+\alpha)^2}{p+1}$$

[0092] 并且因此：$$disto(i,j+1)-disto(i,j)=\alpha^2+\frac{S^2}{p}-\frac{(S+\alpha)^2}{p+1}. \quad (1)$$

[0093] 此外，还有：

[0094] $$disto(i+1,j)=\sum_{i+1\leq l\leq j}x_l^2-\frac{(S-\beta)^2}{p-1}$$

[0095] $$disto(i+1,j+1)=\sum_{i+1\leq l\leq j+1}x_l^2-\frac{(S-\beta+\alpha)^2}{p}$$

[0096] 并且因此：$$disto(i+1,j)-disto(i+1,j+1)=-\alpha^2-\frac{(S-\beta)^2}{p-1}+\frac{(S-\beta+\alpha)^2}{p}. \quad (2)$$

[0097] 让我们表示Δ＝disto(i,j+1)-diso(i,j)+disto(i+1,j)-disto(i+1,j+1)

[0098] 因而，我们想要证明的定理简单地等价于Δ≥0。

[0099] 此外，将等式(1)和(2)相加，我们得到：

[0100] $$\Delta=\frac{S^2}{p}-\frac{(S+\alpha)^2}{p+1}-\frac{(S-\beta)^2}{p-1}+\frac{(S-\beta+\alpha)^2}{p}$$

[0101] 现在我们的目标是使用该表达式来说明Δ≥0。

[0102] 不失一般性，我们可以假设S＝0，这是因为问题是变换不变的(其与通过$-\frac{S}{p}$对所有的点进行变换相对应)，使得具有：

[0103] $$\Delta=-\frac{\beta^2}{p-1}-\frac{\alpha^2}{p}+\frac{(\alpha-\beta)^2}{p}$$
$$=\frac{1}{p(p-1)(p+1)}\Delta'$$

[0104] 其中

[0105] Δ′＝-p(p+1)β^2-p(p-1)α^2+(p-1)(p+1)(α-β)^2

[0106] 对各项进行分组，可以写为：

[0107] Δ′＝-(p+1)β^2+(p-1)α^2-2(p+1)(p-1)αβ

[0108] 现在注意S＝x_i+…+x_j＝β+…+α≤β+(p-1)α，因为对于所有的l∈{i+1，…，j}有α≥x_l(记住x_1＜…＜x_n)。由于我们假设S＝0，因此得到：

[0109] (p-1)α≥-β

[0110] 此外，显然具有β≤0，因为该项比作为零的和S小，因此接下来：

[0111] $-(p-1)\alpha\beta \geq \beta^2$

[0112] 将这一不等式再结合到上一个Δ的表达式中，得到：

[0113] $\Delta' \geq -(p+1)\beta^2+(p-1)\alpha^2+2(p+1)\beta^2 \geq (p-1)\alpha^2+(p+1)\beta^2$

[0114] 因此，我们有$\Delta' \geq 0$，因此获得$\Delta \geq 0$，这结束了证明。

[0115] 现在，对于固定的$k \in \{2,\cdots,K\}$，假定方法已经针对所有的j计算了所有$TD_{min}(j,k-1)$。让我们回想，通过以下关系可以针对所有的j取回$(TD_{min}(j,k))_j$：

[0116] $$TD_{min}(j,k) = \min_{1 \leq i \leq j}\{TD_{min}(i-1,k-1) + disto(i,j)\}$$

[0117] 现在我们看到了上文陈述的特性将如何帮助方法以时间复杂度O(n)根据$(TD_{min}(j,k-1))_j$来计算所有的$(TD_{min}(j,k))_j$。

[0118] 首先，让我们表示$H(i,j)=TD_{min}(i-1,k-1)+disto(i,j)$。

[0119] 由于$disto(i,j)+disto(i+1,j+1) \leq disto(i,j+1)+disto(i+1,j)$，因此我们通过在两侧加上$TD_{min}(i-1,k-1)+TD_{min}(i,k-1)$获得：

[0120] $H(i,j)+H(i+1,j+1) \leq H(i,j+1)+H(i+1,j)$

[0121] 该特性被称为矩阵$H=(H(i,j))_{i,j}$的Monge特性(参考Cechlárová和Szabó的文章“On the Monge property of matrices”)(实际上在$j<i$时所述方法可以丢弃H(i,j)的定义，但是这样的缺失值在实践当中不是问题，并且将不对其进一步讨论)。在一些文献当中，其还被称为Knuth-Yao四边形不等式(例如，参考Bein、Golin、Larmore和Zhang的文章“The Knuth-Yao quadrangle-inequality speedup is a consequence of total-monotonicity”)。

[0122] 根据定理，矩阵H是完全单调的，即：如果对于i，j，k($i<j$)，我们具有$H(k,i)<H(k,j)$，则对于所的有$k' \leq k$，我们也具有$H(k',i)<H(k',j)$。这实际上是Monge矩阵的公知特性并且不需要证明。

[0123] 现在，注意计算$(TD_{min}(j,k))_j$等同于计算矩阵H的每行的最小值。这里，示例的方法调用任何预定的线性时间行最小值搜索算法(例如，SMAWK算法)，该算法碰巧刚好被设计为解决以时间复杂度O(n)解决的该子问题。注意，矩阵H具有大小$n \times n$，但是方法不需要完整地构建该矩阵。方法仅向(例如)SMAWK子例程提供在恒定时间中计算任何H项的方式。

[0124] 因此，在实现当中，方法的算法可以首先计算第一层$(TD_{min}(j,0))_j$，然后其将使用行最小值搜索(RMS)子例程来计算第二层$(TD_{min}(j,1))_j$，并且然后第二次使用RMS算法来计算第三层$(TD_{min}(j,2))_j$等等，直到方法得到所有的$(TD_{min}(j,k))_{j,k}$为止。由于K个层当中的每一个耗费要被计算的时间复杂度O(n)，因此整个算法以时间复杂度O(Kn)运行。

[0125] 就此而言，在示例中，方法还可以包括在每个相应的迭代等级k，将行最小值搜索算法返回的索引存储到(例如)专门的矩阵Cut_{min}当中。该示例的方法还可以包括，在计算步骤S20中根据存储的索引来确定最优聚类。

[0126] 在简单并且直接的实现中，根据存储的索引来确定最优聚类包括在矩阵Cut_{min}内进行工作。具体地，该示例的方法迭代地对数值进行划分，从最后被索引的数值$(Cut_{min}(n,K))$开始。在每个相应的迭代等级q上，当前形成的群集的开始数值的索引等于在等级为K-q的迭代当中存储的索引(在计算步骤S20内的迭代期间)。

[0127] 实际上，如果注意到方法每次计算最小值，则显然，方法还可以得到达到该最小值

的索引。更准确地说，$(TD_{min}(j,k))_{j,k}$的每个值被计算为其索引可以存储在矩阵$(Cut_{min}(j,k))_{j,k}$中的最小值。由此，方法能够仅查看表格Cut_{min}就容易地得到最优划分。

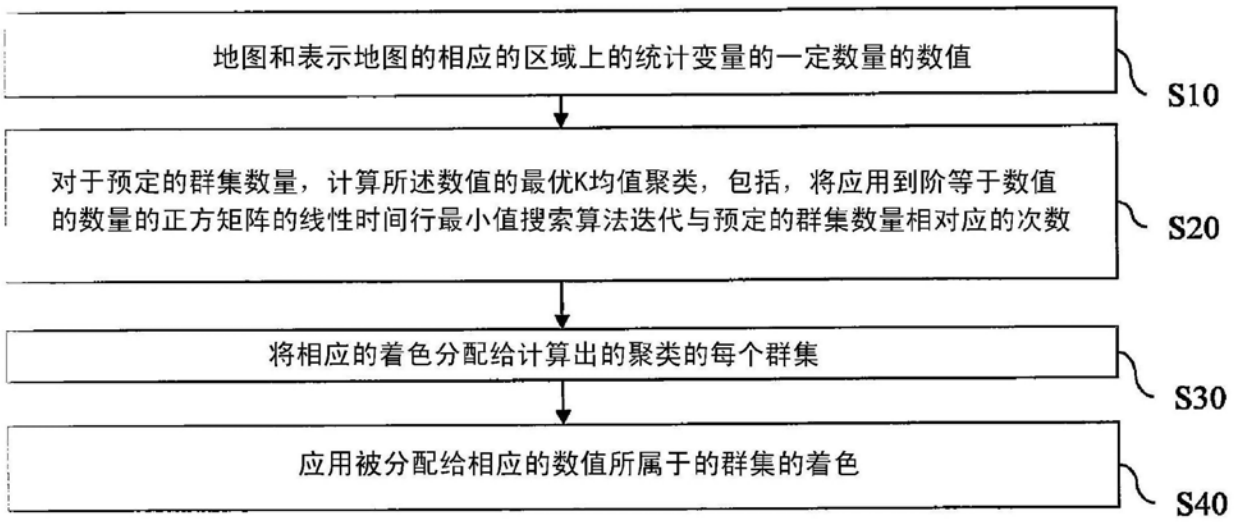

地图和表示地图的相应的区域上的统计变量的一定数量的数值
S10
对于预定的群集数量，计算所述数值的最优K均值聚类，包括，将应用到阶等于数值的数量的正方矩阵的线性时间行最小值搜索算法迭代与预定的群集数量相对应的次数
S20
将相应的着色分配给计算出的聚类的每个群集
S30
应用被分配给相应的数值所属于的群集的着色
S40

图1

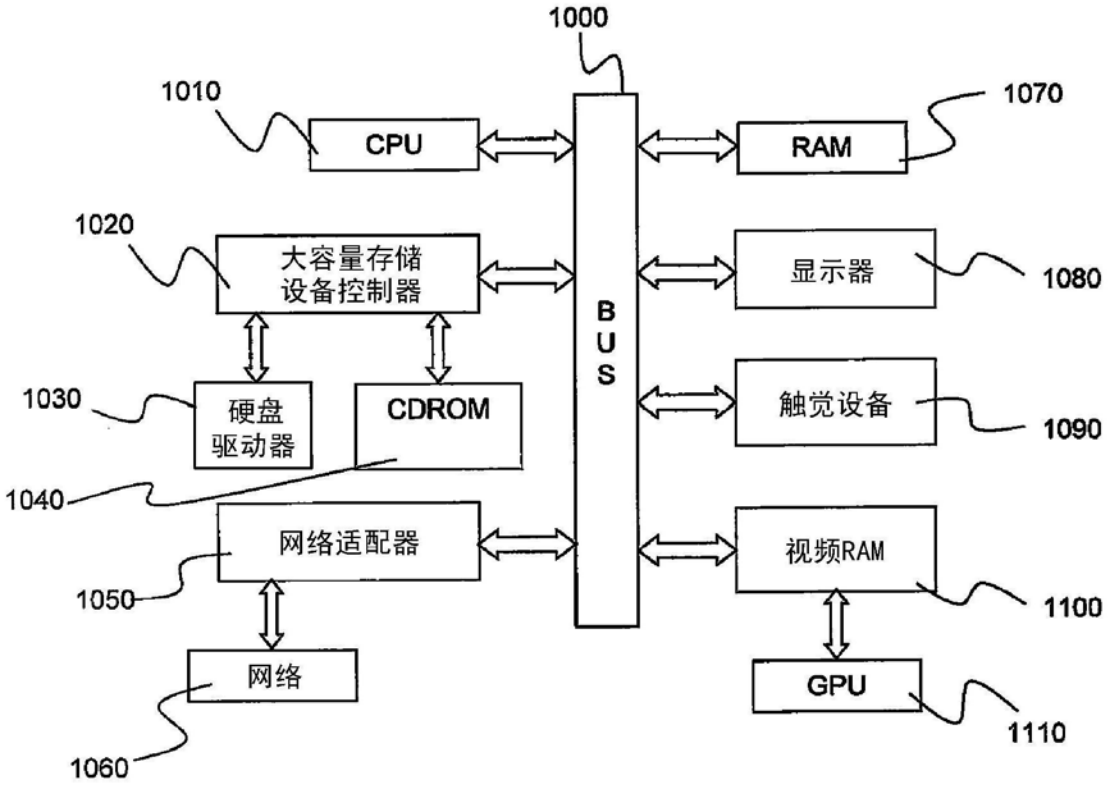

1000
1010
CPU
1070
RAM
1020
大容量存储设备控制器
显示器
1080
BUS
1030
硬盘驱动器
CDROM
触觉设备
1090
1040
网络适配器
视频RAM
1050
1100
网络
GPU
1060
1110

图2

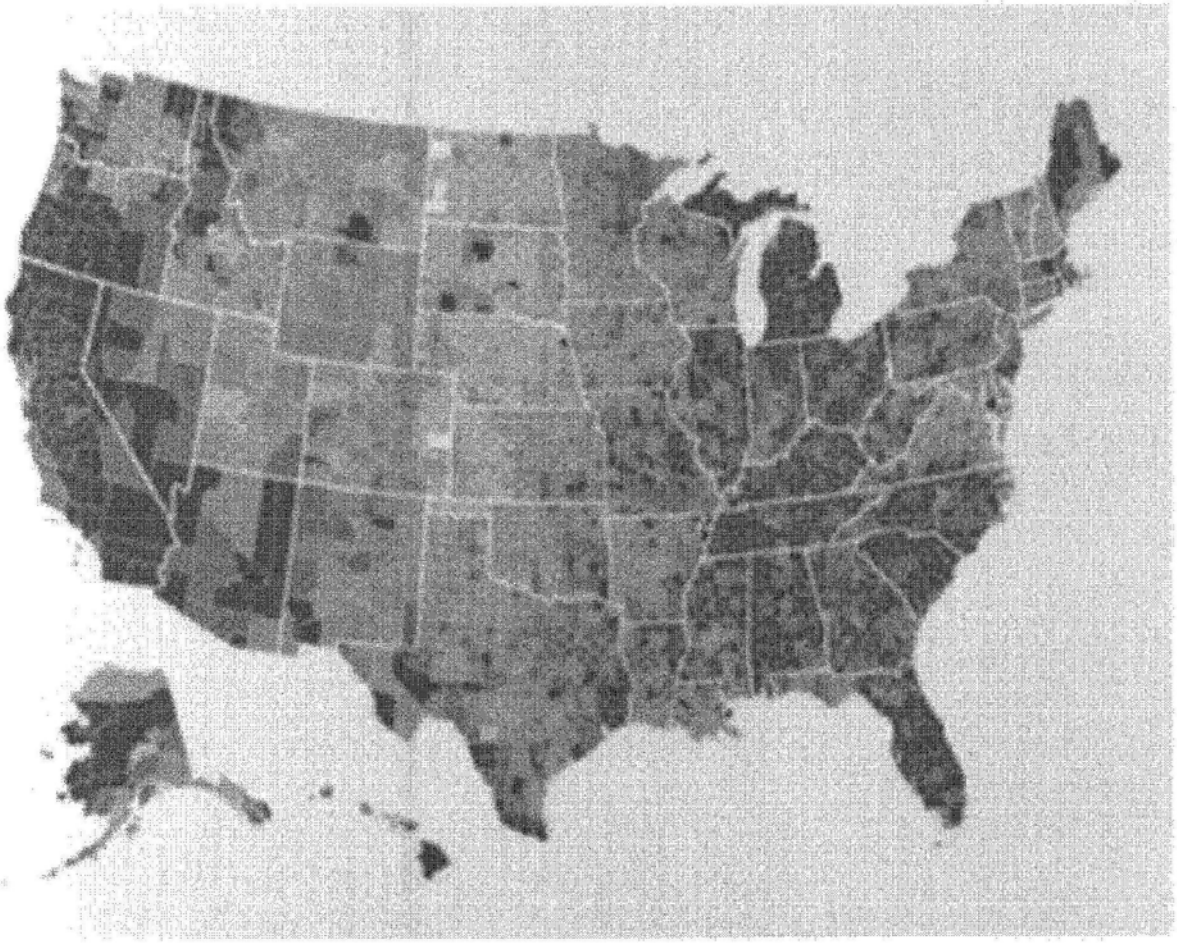


图3

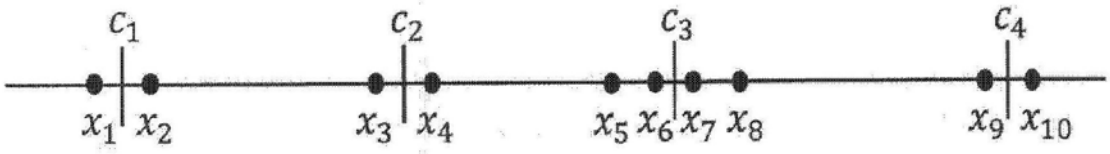


图4