北京大学信息管理系

《数据挖掘导论》讲义

第四章 聚类分析

北京大学信息管理系

2016 年秋

目录

第四章 聚类分析

假设你是某超市的客户关系主管，现想把所有客户分成 5 个组，分配的原则自然是每个组内的客户尽可能相似、组间客户尽可能相异，从而可以针对每组客户的共同特点，开发相应的促销活动。那么，如何根据现有数据进行自动分组？

与分类不同，每个客户的类别未知，因此我们需要通过对大量客户的描述属性与购买行为进行分析，找出有意义的分组方法。聚类就是这样的一种工具，它把数据对象集划分成多个组或簇，使得簇内的对象具有较高的相似性，并且与其它簇中的对象尽可能不相似。相似性和相异性通过描述对象的属性值来评估，通常涉及距离度量。

本章将从聚类方法的概述出发，学习数据对象与数据类间的相似性度量方法，并具体介绍三种专门的聚类算法。

4.1 概述

聚类是最常用的数据分析技术之一，它已有很长的历史，并且几乎在所有领域中都被用到，例如电子商务、电子政务、信息资源管理、经济学、社会学、医学、心理学、植物学、生物学、考古学等等。近几年由于在线文档和互联网的快速发展，文本文档的聚类也成为一个较为活跃的研究领域。

本节首先对聚类分析做进一步定义，然后概述将数据对象集划分为簇的基本聚类方法，最后介绍聚类分析技术在文本信息处理领域中的应用。

4.1.1 什么是聚类分析

聚类是对数据对象进行划分的一种过程，与分类不同的是，它所划分的类是未知的。在数据分类（data classification）中，我们需要事先给定一个分类体系和训练样本集，如天网的中文网页分类体系，它将所有的网页分为 12 个大类，即任何网页的内容将属于这 12 个大类中某一个（或两个）类别，训练样本集中的每一个网页被人工标号为属于某一类样本。这种利用已标记样本集进行对未知样本进行类别划分的方法称为“有监督的学习”（supervised learning）方法。而聚类是一个“无简单的学习”（unsupervised learning）过程，即聚类算法不需要“教师”的指导，不需要提供训练数据，仅根据该组数据对象的特征而进行的一种自然的划分。

聚类分析（cluster analysis）是一个“无监督的学习” 过程，它将数据对象划分为多个类或簇，使得在同一个簇中个体的具有较高的相似度，而不同簇中的个体差别较大。

中国古语有“物以类聚人以群分”之说，但根据什么来聚类？假设我们把中国的县分类，就有多种方法，可以按照自然条件来分，比如考虑降水、土地、日照、湿度等；也可考虑收入、教育水准、医疗条件、基础设施等社会指标。也就是说，既可以用某一项来聚类，也可以同时考虑多项指标来聚类。

例 4.1 饮料的聚类分析。表 4-1 给出了 16 种饮料的四种变量取值（热量、咖啡因、钠及价格），我们可以根据该数据做哪些聚类分析？

表 4-1 16 种饮料品牌的聚类

饮料编号	热量	咖啡因	钠	价格
1	207.20	3.30	15.50	2.80
2	36.80	5.90	12.90	3.30
3	72.20	7.30	8.20	2.40
4	36.70	.40	10.50	4.00
5	121.70	4.10	9.20	3.50
6	89.10	4.00	10.20	3.30
7	146.70	4.30	9.70	1.80
8	57.60	2.20	13.60	2.10
9	95.90	.00	8.50	1.30
10	199.00	.00	10.60	3.50
11	49.80	8.00	6.30	3.70
12	16.60	4.70	6.30	1.50
13	38.50	3.70	7.70	2.00
14	.00	4.20	13.10	2.20
15	118.80	4.70	7.20	4.10
16	107.00	.00	8.30	4.20

（1）对变量进行聚类（即数据中的列，如热量、咖啡因、钠、价格），对变量的聚类又称为 R 型聚类，在 SPSS Modeler 的 k-均值聚类中需要把数据阵进行转置才可直接分析；

（2）对观测值进行聚类，（即数据中的行，16 种不同品牌的饮料），如仅根据热量这一项对不同饮料聚类，或同时根据四个变量进行聚类等，对观测值的聚类又称为 Q 型聚类。

4.1.2 基本聚类方法概述

文献中有大量的聚类算法，很难对其提出一个简洁的分类，因为这些类别可能交叉重叠，从而使得一种方法具有几种类别的特征。但是一般而言，主要的基本聚类方法可以分成如下几类：

（1）划分聚类（partitional clustering）：给定一个 n 个对象的集合，划分聚类构建数据的 k 个子集，并且 k≤n。也就是说它把数据划分为 k 个组，使得每个组至少包含一个对象。一般来说，划分聚类是将数据对象集划分成不重叠的子集（即互斥子集），但在模糊划分技术中，可以适当放宽。

使用划分聚类的前提是用户指定簇的个数，但有时用户并不能确定到底需要划分成多少个簇才能达到最好效果，这时可用层次聚类法确定大概的簇数目。

（2）层次聚类（hierarchical clustering）：根据层次分解的形成步骤，可分为自底向上方法和自顶向下方法。自底向上方法将每个对象作为单独的一个组，然后逐次合并相近的对象或组，直到所有的组合并成一个组（即层次的最顶层），或者满足某个终止条件；自顶向下方法将所有的对象置于一个组中，在每次相继迭代中，一个簇被划分成更小的簇，直到最终每个对象在单独的一个簇中，或满足某终止条件。

层次聚类的缺陷在于一旦一个步骤完成，就不能被撤销，也就是说某个步骤进行了错误的划分后，无法更正该错误。

（3）基于密度聚类（density-based clustering）：基于对象之间的距离进行聚类时，往往只能发现球状簇，而当现实中的簇为任意形状时，或有噪声和孤立点时，常常使用基于密度的方法进行聚类。该方法的主要思想是，只要邻域中的密度（数据点的数目）超过某个阈值，才能继续进行划分，也就是说，对给定簇中的每个数据点，在给定半径的邻域中必须至少包含最少数目的点。如图 4-1 所示，两个圆形簇并没有合并，因为它们之间的桥消失在噪声中。

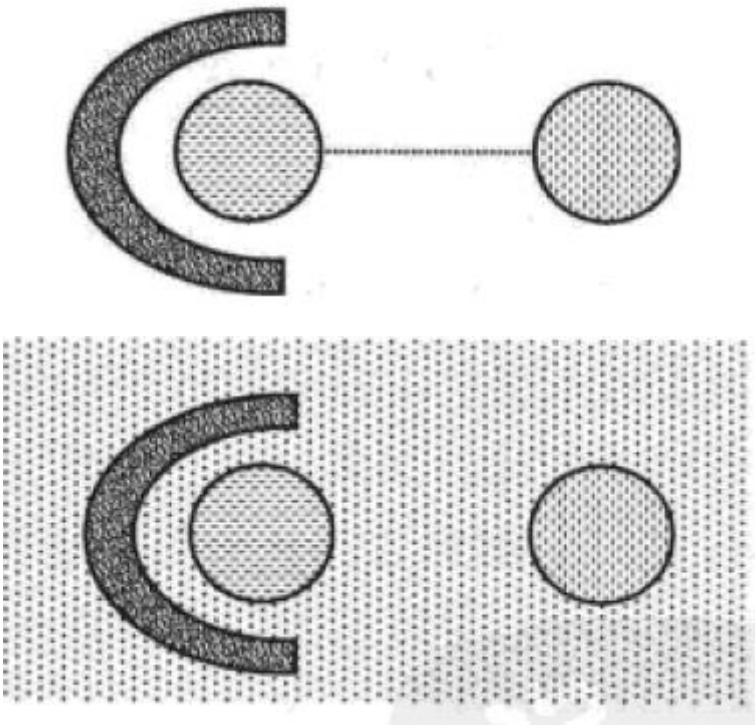


图 4-1 基于密度的簇

4.1.3 文本聚类

在信息检索中，文本聚类的早期应用主要是为了提高系统的查准率与查全率，并被用于寻找给定文本的相近文本，目前主要用于浏览文本、显示文本集合、组织搜索引擎的返回结果，如搜索引擎 Vivisimo 将查询结果进行聚类，这有利于用户快速定位自己所需要的信息。

例 4.2 文本聚类。遍布全球的新闻机构每天都会发布不计其数的新闻文章。假设一个网站想要收集这些新闻文章并提供一种整合的新闻服务，那么它必须要根据某种主题层次来组织这些收集到的文章。问题是，这些主题应怎么选择和组织？一种可能的方法是雇用一批专家来做这项工作，但是，通过人力来组织是十分昂贵而且耗时的，这使得这种方法对于新闻和其他具有实时性要求的任务来说并不合适。把未经分类的新闻文章全部扔给用户也显然不是一种可选的方法。尽管分类方法能够把新闻分类到一些事先定义好的类别中去，但是由于分类方法需要训练数据，这使得分类方法在这里并不可行，因为训练数据需要大量的人力标注。更进一步，新闻主题是时刻都在快速变化的，从而训练数据也需要随之发生变化，这使得人力标注变得不可能。显然，聚类成为解决这个问题的一种方案。它根据新闻文章内容的相似度自动地把它们归类。

在文本或网页聚类中，使用最多的聚类方法是：k-均值聚类与层次聚类方法、或这两种基本方法的改进形式。其他方法包括：自组织特征映射、遗传算法等等。

进行文本聚类的大致步骤可概括如下：

对给定的文本集合，（1）首先需要对各个文本进行预处理，包括词法分析，英文的词干提取、中文分词或命名实体识别等；（2）然后进行特征选取、或特征抽取等，构造文本集合的特征向量空间；（3）计算各个文本之间的相似度或距离，即：构造文本间的相异度（或相似度）矩阵；（4）选择某一种聚类算法进行聚类；（5）根据实际应用，进行类别选择与类别命名；（6）将聚类结果以适当的形式（如列表或可视化等形式）输出；（7）对聚类结果进行评估，以检验聚类结果的质量。如结果不好，可以返回到前面的某一步进行重新聚类。图 4-

2 为文本聚类的一般过程。

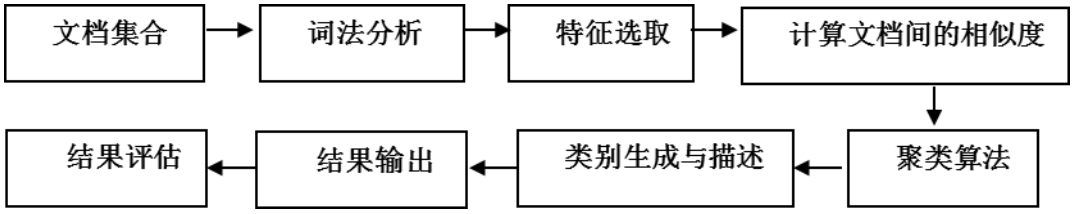


图 4-2 文本聚类的一般过程

4.2 数据间的相似性度量

假设需要对 100 个学生进行分类。当我们仅知道他们的数学成绩时，只好按照数学成绩进行分类，这些成绩在直线上可以形成 100 个点，这样就可以把接近的点放到一类；如果还知道他们的物理成绩，这样数学和物理成绩就形成二维平面上的 100 个点，也可以按照距离远近来分类。三维或者更高维的情况也是类似，只不过三维以上的图形无法直观地画出来而已。在例 4.1 的饮料数据中，每种饮料都有四个变量值，若想测量其距离，就是四维空间点的问题。

为了度量样本间的接近或相似程度，需要定义一些用于划分类别的计量指标。常用的统计指标有距离和相似系数。“距离”属于相异性测度指标，“相似系数”属于相似性测度指标。距离和相似系数成某种反比关系，如 $sim(d_i,d_j)=1/(1+d_{ij})$，其中，$sim(d_i,d_j)$表示数据对象 d_i 与 d_j 间的相似系数，d_{ij} 表示数据对象 d_i 与 d_j 间的距离。

4.2.1 数据对象间的距离

对于有 n 个特征属性的数据对象集合来说，m 个数据对象可以看作 n 维空间中的 m 个点，此时，每一个数据对象被表示为一个 n 维向量。进而我们可以用点间的距离来度量数据对象间的距离。

最常用的距离度量方法是欧几里得距离，定义如下：

$$d_{ij}=\sqrt{(w_{i1}-w_{j1})^2+(w_{i2}-w_{j2})^2+\cdots+(w_{in}-w_{jn})^2}$$

其中，第 i 个数据对象与第 j 个数据对象分别用 $(w_{i1},w_{i2},\cdots,w_{in})$ 和 $(w_{j1},w_{j2},\cdots,w_{jn})$ 表示；d_{ij} 表示第 i 个数据对象与第 j 个数据对象间的距离。

另一个度量方法是曼哈坦距离，其定义如下：

$$d_{ij}=|w_{i1}-w_{j1}|+|w_{i2}-w_{j2}|+\cdots+|w_{in}-w_{jn}|$$

上面的两种度量方法都满足对距离函数的如下要求：

$d_{ij}\geq 0$， 即距离是一个非负的数值；

$d_{ii}=0$， 即一个文档与自身的距离为 0；

$d_{ij}=d_{ji}$，即距离函数具有对称性；

$d_{ij}\leq d_{ik}+d_{kj}$，即距离函数满足三角不等式；

明考斯基距离是欧几里得距离和曼哈坦距离的概化，定义如下：

$d_{ij} = (|w_{i1} - w_{j1}|^p + |w_{i2} - w_{j2}|^p + \cdots + |w_{in} - w_{jn}|^p)^{1/p}$，其中 q 是一个正整数， 当 q=1 时，它表示曼哈坦距离；当 q=2 时，它表示欧几里得距离。

如果对每一个变量根据其重要性赋予一个权重，就得到加权的明考斯基距离：

$d_{ij} = (u_1 |w_{i1} - w_{j1}|^p + u_2 |w_{i2} - w_{j2}|^p + \cdots + u_n |w_{in} - w_{jn}|^p)^{1/p}$

4.2.2 数据对象间的相似系数

对于有 n 个特征属性的数据对象集合来说，我们也可以用相似系数来度量它们之间的相近程度，用 sim(i,j)表示第 i 个向量与第 j 个向量之间的相似系数，则我们有：

（1）$\forall i, j \quad |sim(i, j)| \le 1$，即绝对值总不大于 1

（2）$\forall i, j \quad sim(i, j) = sim(j, i)$，即满足对称性

常见的几种相似系数有：

（1）Pearson 相关系数（Pearson Correlation Coefficient）

（2）余弦系数 (Cosine coefficient)

$$sim(i, j) = \frac{\sum_{k=1}^{n} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{n} w_{ik}^2 \sum_{k=1}^{n} w_{jk}^2}}$$

（3）重叠系数 (Overlap coefficient)

$$sim(i, j) = \frac{\sum_{k=1}^{n} w_{ik} w_{jk}}{\min(\sum_{k=1}^{n} w_{ik}, \sum_{k=1}^{n} w_{jk})}$$

（4）雅可比系数 (Jaccard coefficient)

$$sim(i, j) = \frac{\sum_{k=1}^{n} w_{ik} w_{jk}}{\sum_{k=1}^{n} w_{ik} + \sum_{k=1}^{n} w_{jk} - \sum_{k=1}^{n} w_{ik} w_{jk}}$$

对于二值的情况，相似系数的计算还有其它的度量指标。此外，需要指出的是数据对象之间不同度量指标的选择，所得到的聚类结果存在一定的差异，需要结合具体情况来考察哪些度量指标对那些数据更有效。

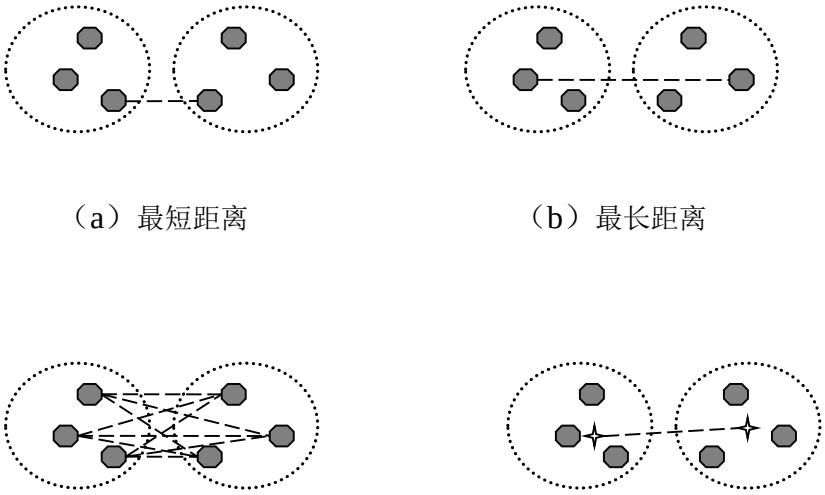
4.2.3 数据类间的距离

按照距离来聚类需要明确两个概念：一个是点和点之间的距离，一个是类和类之间的距离。如何测量 4.2.1 节主要介绍了数据点之间距离的测量方法，本节将对数据类间的距离测量做简单介绍。

由一个点组成的类是最基本的类，如果每一类都由一个点组成，那么点间的距离就是类间距离。但是如果某一类包含不止一个点，那么就要确定类间距离。

设 D_1 和 D_2 是两个数据类，分别包括 m_1 和 m_2 个数据对象，用 $s(D_1,D_2)$表示两类对象之间的距离, $s(d_1,d_2)$表示对象 d_1 和对象 d_2 之间的距离，并假设这两个数据对象集合都具有 n 个特征属性。我们结合图 4-3 来说明如下几种对象类之间距离的常见度量方法，其中图中虚线

围成的两个椭圆形分别表示两类数据对象集 D_1 和 D_2 。



（a）最短距离 （b）最长距离

（c）类平均距离 （d）重心距离

图 4-3 数据类之间的距离度量

（1）最短距离法：定义为两类中最近的两个数据对象间的距离为两类对象间的距离，也称为单链接方法。如图 4-3（a）所示，短虚线所连接的两个对象之间的距离即定义为 $s(D_1,D_2)$；可用下式形式化地表示。

$$s(D_1,D_2)=\min\{s(d_1,d_2)\mid d_1\in D_1,d_2\in D_2\}$$

使用该方法进行聚类时，聚类结果受噪音数据影响较大，可能会形成一些非常奇怪的聚类。该方法的时间复杂度为 $O(n^2)$，其中，n 为数据对象的个数。

（2）最长距离法：定义为两类中最不相近的两个对象间的距离为两类对象间的距离，也称为全链接方法。如图 4-3（b）所示，短虚线所连接的两个对象之间的距离即定义为 $s(D_1,D_2)$；可用下式形式化地表示。

$$s(D_1,D_2)=\max\{s(d_1,d_2)\mid d_1\in D_1,d_2\in D_2\}$$

在最坏情况下，该方法的时间复杂度为 $O(n^2\log n)$，其中，n 为对象的个数。

（3）类平均距离法：定义为两类中任意两个对象间距离的算术平均值，作为两类对象间的距离，也称为平均链接方法。如图 4-3（c）所示，先计算所有短虚线所连接的两个对象之间的距离，然后计算其平均值，该值即定义为 $s(D_1,D_2)$；可用下式形式化地表示。

$$s(D_1,D_2)=\frac{1}{m_1*m_2}\sum_{i=1}^{m_1}\sum_{j=1}^{m_2}s(d_i,d_j),\quad (d_i\in D_1,d_j\in D_2)$$

该方法需要计算两类对象中所有对象之间距离的均值，计算量较大，时间复杂度为 $O(n^2\log n)$，其中，n 为对象的个数。

（4）重心距离法：定义为两类重心之间的距离。其中，类的重心是指类中所有对象向

量在各分量上计算其算术平均值后所得的向量。

设 D_G 为 D 的重心，即 $D_G = \frac{1}{m}\sum_{j=1}^{n} d_j$；则 $s(D_1, D_2) = s(d_{1G}, d_{2G})$

4.2.4 数据标准化

选用的度量单位将直接影响聚类分析的结果，如将高度由“米”改为“厘米”， 重量由“克”改为“千克”，就可能产生非常不同的聚类结果。为了避免对度量单位的依赖，数据应当进行标准化，有多种方法，在此我们介绍三种方法：最小--最大规范化、Z-score 规范化和按小数定标规范化。

（1）最小-最大规范化：对原始数据进行线性变换。设 $\min_x$ 和 $\max_x$ 分别为变量 x 的最小值与最大值。则最小-最大规范化通过计算

$$w' = \frac{w - \min_x}{\max_x - \min_x}(new\_\max_x - new\_\min_x) + new\_\min_x$$

将 x 的值 w 映射到区间[new_min_x, new_max_x]中的 w’。

（2）Z-score 规范化：变量 x 的值基于 x 的平均值和标准差规范化。x 的值 w 被规范化为 w’，由下式计算：

$$w' = \frac{w - a}{\sigma}$$

其中，a 和 σ 分别为 x 的平均值和标准差。

（3）小数定标规范化：通过移动 x 取值的小数位置进行规范化。小数点的移动位置依赖于 x 取值的最大数的绝对值。x 的值 w 被规范化为 w’，由下式计算：

$$w' = \frac{w}{10^j}$$

其中，j 是使得 Max(|w’|)<1 的最小整数。如：|w|=985，则 j=3，此时 w’=0.985。

4.3 基本聚类方法

4.3.1 k-均值聚类方法

典型的划分方法包括 k-均值方法、k-中心点方法及其它们的变形。本节主要介绍 k-均值方法。

k-均值方法（k-means method）是一个简单高效、应用广泛的聚类方法。前面说过，划分聚类要求先说好要分多少个类。假定分 3 类，那么需要事先确定 3 个点为“聚类种子”，也就是说，把这 3 个点作为三类中每一类的基石；然后根据和这三个点的距离远近，把所有点分成三类，再把这三类的中心（即均值）作为新的基石或种子，再重新按照距离分类；如此迭代下去，直到达到停止迭代的要求（比如各个类最后变化不大，或者迭代次数太多）。显然，前面对聚类种子的选择并不必太认真，因为它们很可能最后被分到同一类中。

图 4-4 较为详细地对这一过程进行了描绘。首先随机选择 k 个对象作为初始值，用以代表一个簇的平均值或中心，其中 k 是用户指定的参数，即所期望的簇的个数。对剩余的数据对象，根据它与各个簇中心的距离将它赋给最近的簇。然后重新计算每个簇的平均值。这个过程不断重复，直至每个数据对象所属的簇不再发生变化。

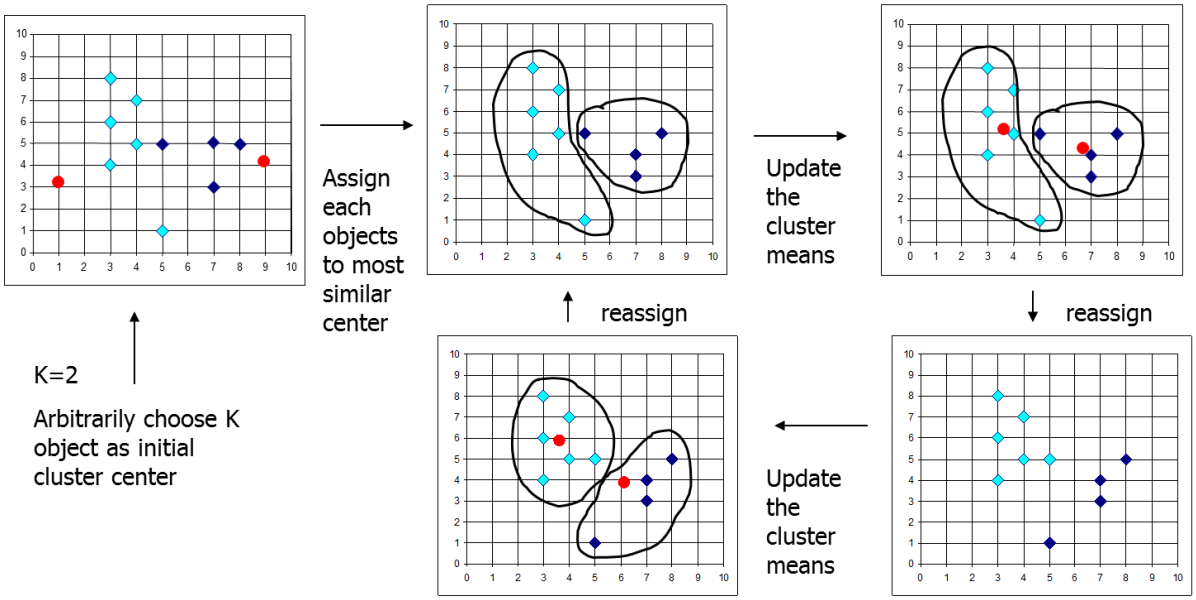


图 4-4 k-均值算法的过程

step1：任意选择 k 个点作为初始的类的中心（即图中的红色点）

step2：根据类中数据对象据初始类中心的距离，将每个数据对象赋给最相近的类

step3：更新类的平均值

step4：重复以上过程

step5：直到不再发生变化，即没有对象进行被重新分配时过程结束。

该算法试图找出使平方误差值最小的 k 个划分，所以当结果簇是密集的，并且簇与簇之间区分明显时，它的效果较好。但是由于该算法在每次迭代时，都要计算每个小类的均值作为该类的代表元，因此该算法对异常数据十分敏感，因为少量的该类数据对平均值会产生较大影响。

从算法复杂度来说，该算法易于实现且效率较高，时间复杂度为 $O(tkn)$。其中，n 为数据点的个数，k 是聚类的个数，t 是循环迭代的次数，由于 k 和 t 通常都远远小于 n，因此该算法相对于数据点数目而言常常是线性时间内可以完成的。

目前，k-均值方法有多种变形形式，不同改进在于初始 k 个数据对象的选择策略；相异度的计算方法；如何计算类的平均值等。产生较好聚类结果的一个有趣策略是，首先用层次聚类方法决定结果簇的个数，并找到初始的聚类，然后用迭代重定位来改进聚类结果。

4.3.2 层次聚类方法

层次聚类方法能对数据对象进行层次化的组织，即一个大的主题可以包含若干个小的主题，形成一个目录层次结构，尽管该目录结构可能并不十分精确。

层次聚类方法通常用树图来表示其聚类过程。

例 4.2 文档层次聚类。图 4-5 给出了一个文档聚类的例子，在树的最底层包含 5 个文档（d1，d2，d3，d4，d5）。在上一层中，聚类点 6 包含两个文档 d1 和 d2，聚类点 7 包含另两个文档 d4 和 d5。随着我们自下而上遍历该树，聚类的数目越来越少，用户可以选择查看在树的任意层次上的聚类。

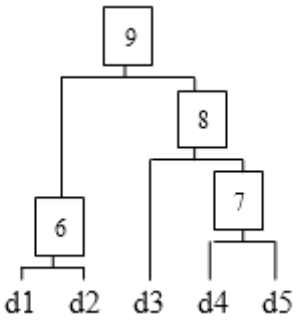


图 4-5 文档层次聚类

前面提过，层次聚类方法主要包括自底向上（合并）的层次聚类方法与自顶向下（分裂）的层次聚类方法。目前合并的层次聚类方法比分裂的层次聚类方法的应用更为广泛，例 4.2 就是一个自底向上策略实施的例子。我们可将其步骤简单表示为：

step1：将数据对象集 D 中的每一个数据对象看做一个类。

step2：计算 D 中每两个对象之间的距离

step3：将距离最小（或相似度最大）的两个对象类 D_i, D_j 合并为新的一类 C。其中 $D_i, D_j \subset D$

step4：计算 C 与其它各类之间的距离，并重复以上过程。

Step5：直到所有对象合并为一类或达到某个终止条件。

对于例 4.1 中提到的饮料聚类，可以通过层次聚类的方法，用 SPSS 软件进行相关分析，如图 4-6。

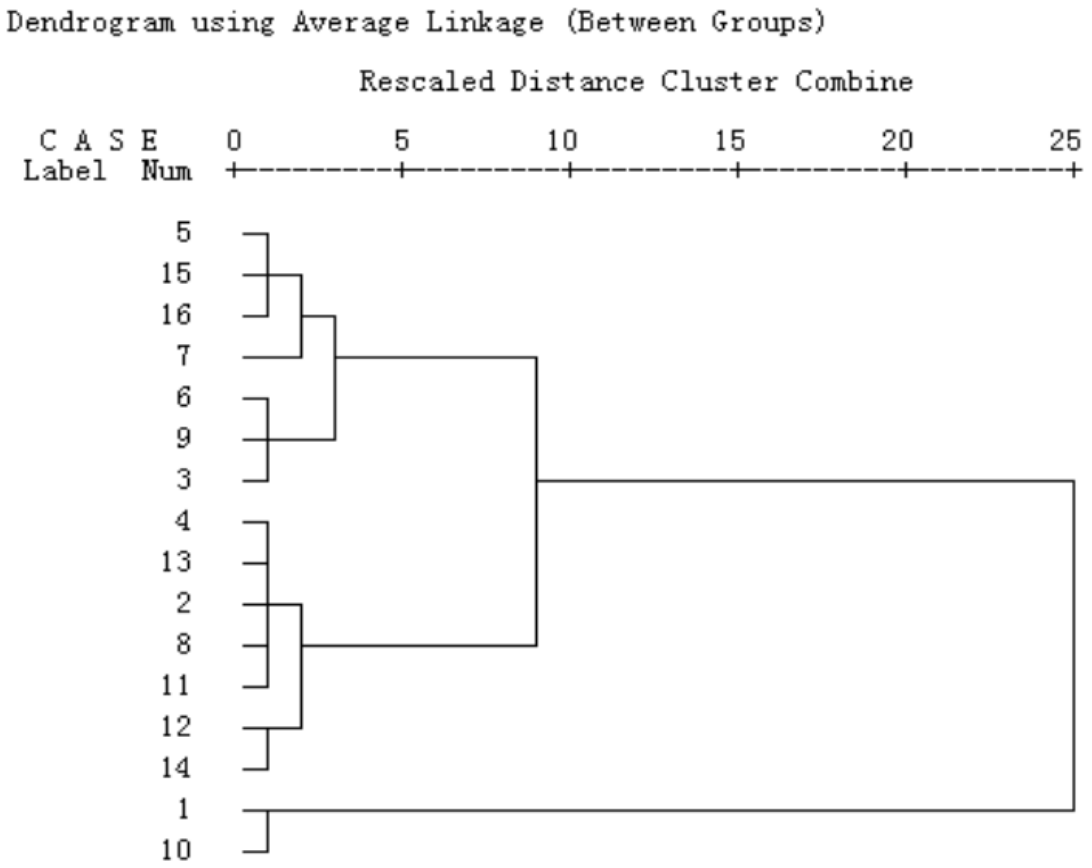


图 4-6 对 16 种不同品牌的饮料进行层次聚类分析

层次聚类算法实质上是一种贪心算法，该方法的时间复杂度 $O(n^2)$，n 为对象数。尽管层次聚类法比较简单实用，但存在一些问题。在聚类过程中，一旦一个步骤（合并或分裂）完成就不能进行修正，如果存在错误，此错误聚类将一直保持到聚类结束。目前已有多种改进方法，如凝聚层次的聚类与迭代重定位的集成等。

4.3.3 聚类要注意的问题

聚类结果主要受所选择的变量影响。如果去掉一些变量，或者增加一些变量，结果会很不同。相比之下，聚类方法的选择则不那么重要了。因此，聚类之前一定要目标明确。

另外就分成多少类来说，也要有道理。只要你高兴，从分层聚类的计算机结果可以得到任何可能数量的类。但是，聚类的目的是要使各类之间的距离尽可能地远，而类中点的距离尽可能的近，并且分类结果还要有令人信服的解释。这一点并不是数学就可以解决的。

4.4 基于密度的聚类（待更新）

4.5 聚类结果的评估

在使用某种聚类方法对数据对象进行聚类后，如何对聚类结果进行评估是一件比较困难的事情，这与分类有着截然的不同，因为在聚类中任何人都不知道某一数据集的正确聚类是什么样的。对聚类结果进行评估的常见方法包括：

用户验证，即邀请一个专家组来对聚类结果进行评估。

真实数据，即使用真实的分类数据集对聚类算法进行评估。

间接评估，即聚类可能是其他任务的一个步骤，如：数据预处理、孤立点检测等。