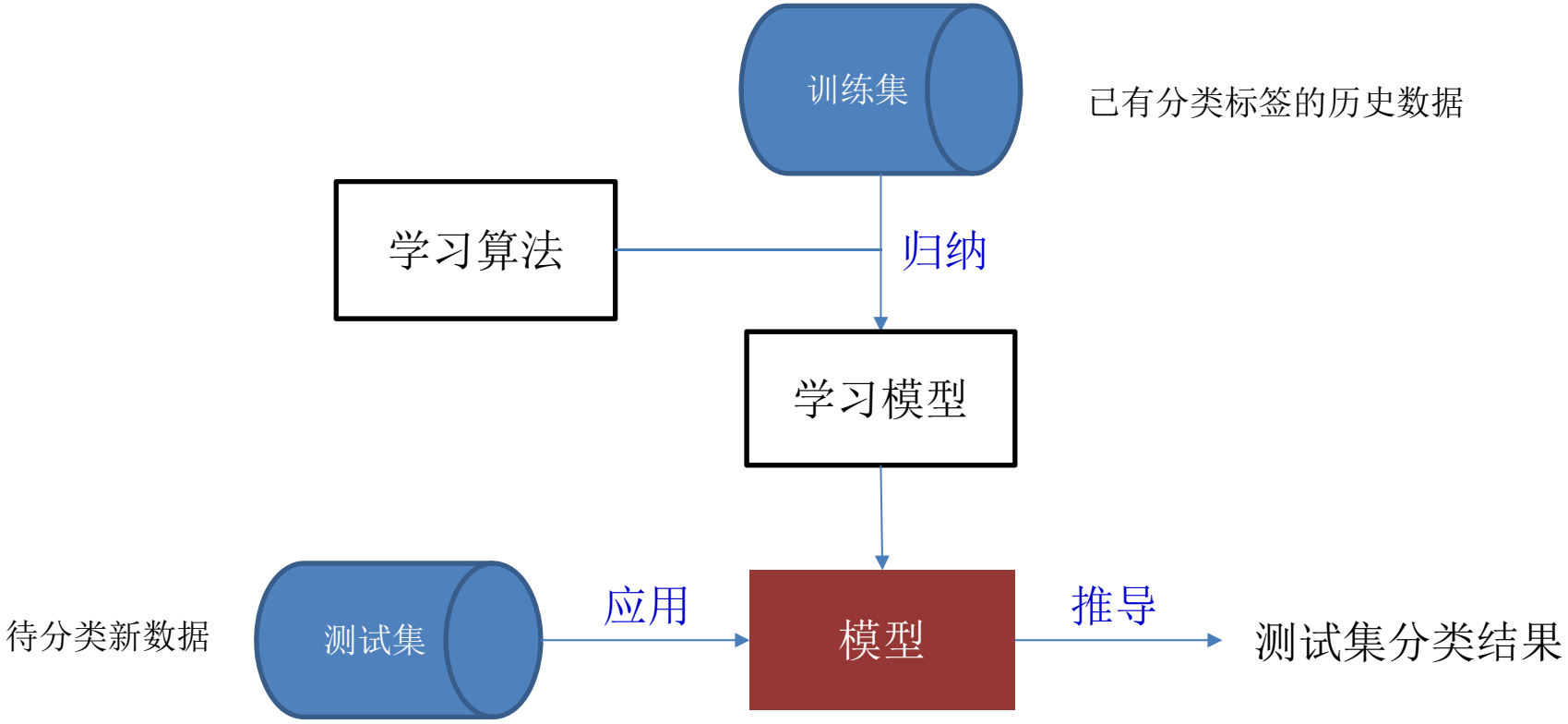
商务智能
Business Intelligence

聚类分析

朱卫平 博士
计算机学院
武汉大学

分类

- 分类是利用一个分类函数（分类模型、分类器），该模型能把数据库中的数据映射到一个给定类别中。



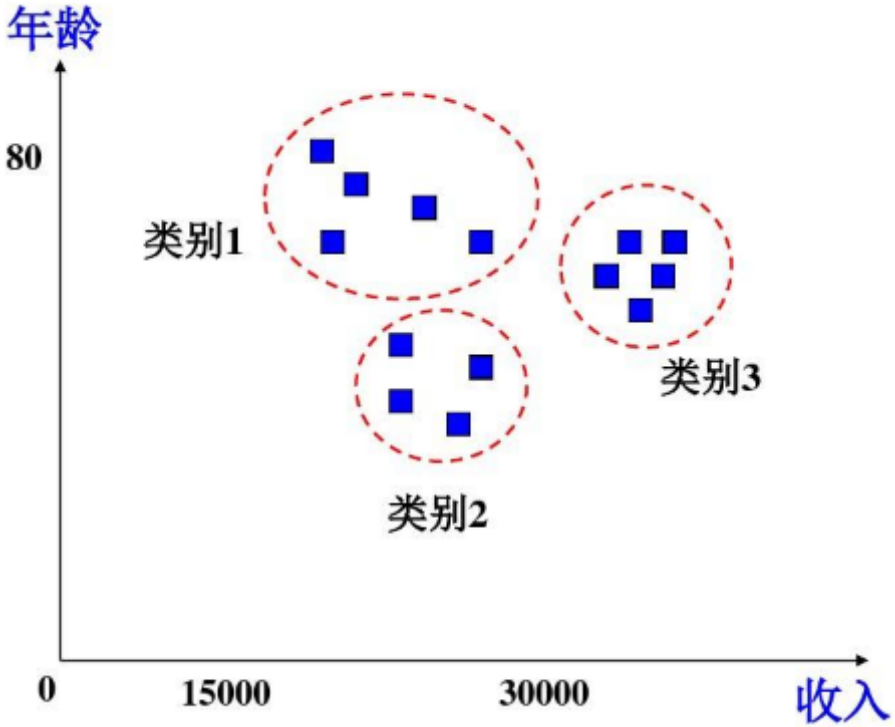
内容提要

1. 什么是聚类分析
2. K-均值和K中心点算法
3. 层次聚类

聚类分析

- 什么是聚类？
  - 聚类分析（cluster analysis）简称聚类（clustering）是把数据划分成子集的过程。
  - 每个子集是一个簇（cluster）。聚类使得簇中的对象彼此相似，但与其他簇中的对象很不相似。

高内聚、低耦合！



聚类分析

- 分类与聚类
  - 分类称作监督学习，事先知道训练样本的标签，通过挖掘将属于不同类别标签的样本分开，可利用得到的分类模型，预测样本属于哪个类别。
  - 聚类是一种无监督的学习，事先不知道样本的类别标签，通过对相关属性的分析，将具有类似属性的样本聚成一类。

划分方法

■ 思想

- 给定n个数据对象的数据集D，以及要生成的簇数k，划分算法把数据对象组织成k(k≤n)个分区，每个分区代表一个簇。
- 传统的划分方法可能需要穷举所有可能的划分，计算量极大，所以大多应用都采用了启发式方法，如：
- K－均值和 k －中心点算法。

K-均值：一种基于形心的技术

- 基于形心的技术: 使用簇C_i的形心 c_i代表该簇。
- 对象 $p \in C_i$与 c_i之差用dist（p, c_i）度量，簇C_i的质量可以用簇内变差度量，它是C_i中所有对象和型心 c_i之间的误差的平方和

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} (d(p, c_i))^2$$

- 可以穷举所有情况找到最优的划分，但是计算开销非常巨大。所以，经常使用k-均值（K-means）方法进行聚类
- 划分采用互斥划分

K-均值：一种基于形心的技术

- K-均值算法过程：

输入：k：簇的个数

D：包含n个数据的数据集

输出： k个簇的集合方法：

（1）从D中任意选择k个对象作为初始簇中心，计算簇均值；

（2）repeat

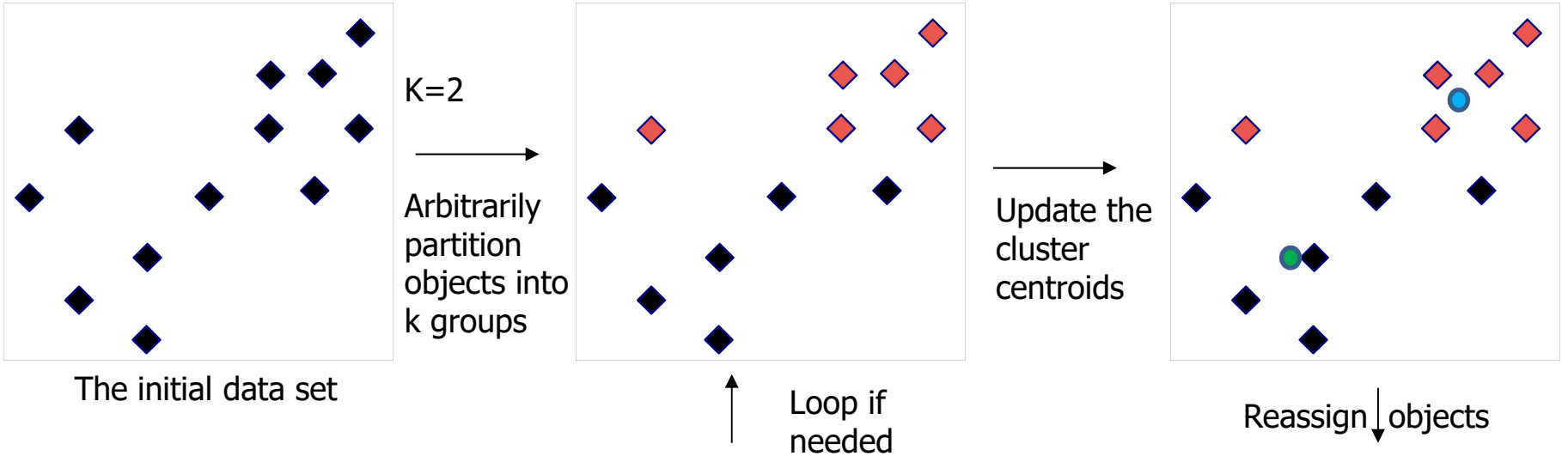
（3）根据每个簇的簇均值，将每个对象分配到最相似的簇；

（4）更新簇均值；

（5）until不再发生变化；

K-均值算法把簇的形心定义为簇均值，即簇内对象的均值

K-均值：一种基于形心的技术



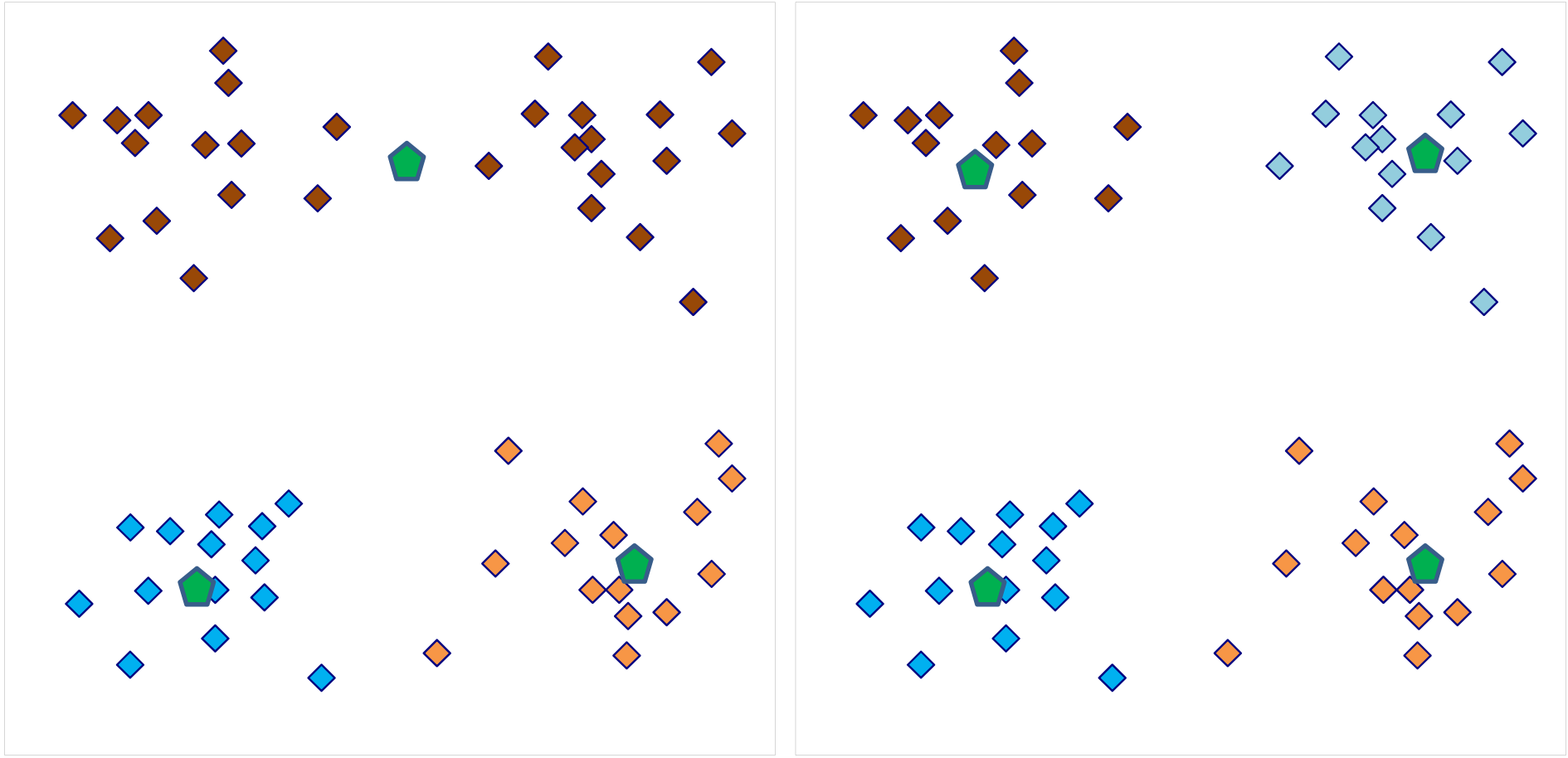
- Partition objects into *k* nonempty subsets
- Repeat
  - Compute centroid (i.e., mean point) for each partition
  - Assign each object to the cluster of its nearest centroid
- Until no change

K-均值：一种基于形心的技术

- K-均值聚类算法的优点
  - 思想简单易行；
  - 时间复杂度接近线性；
  - 对大数据集，是可伸缩和有效的；
- K-均值聚类算法的缺点
  - 要求用户事先给出要生成的簇数k，不适合的k 值可能返回较差的结果
  - 不适合发现非凸形状的簇，或者大小差别很大的簇。
  - 对噪声和离群点敏感。

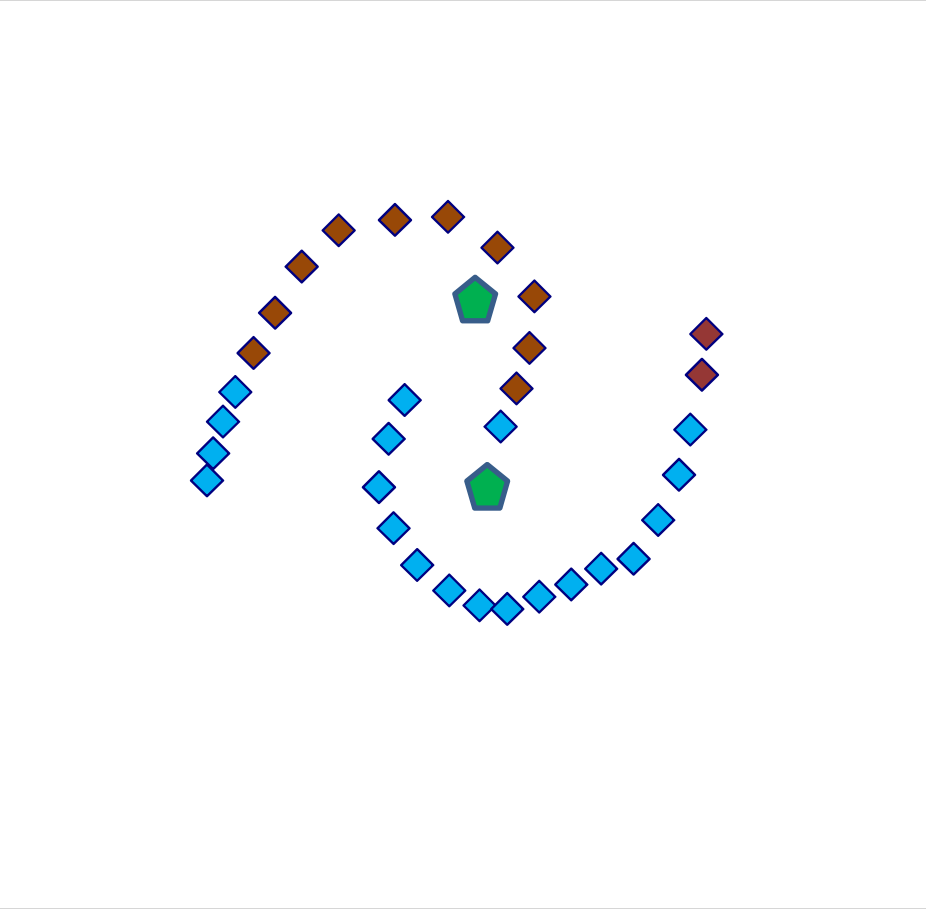
K-均值：一种基于形心的技术

- K-均值聚类缺点1：要求用户事先给出要生成的簇数k，不适合的k 值可能返回较差的结果



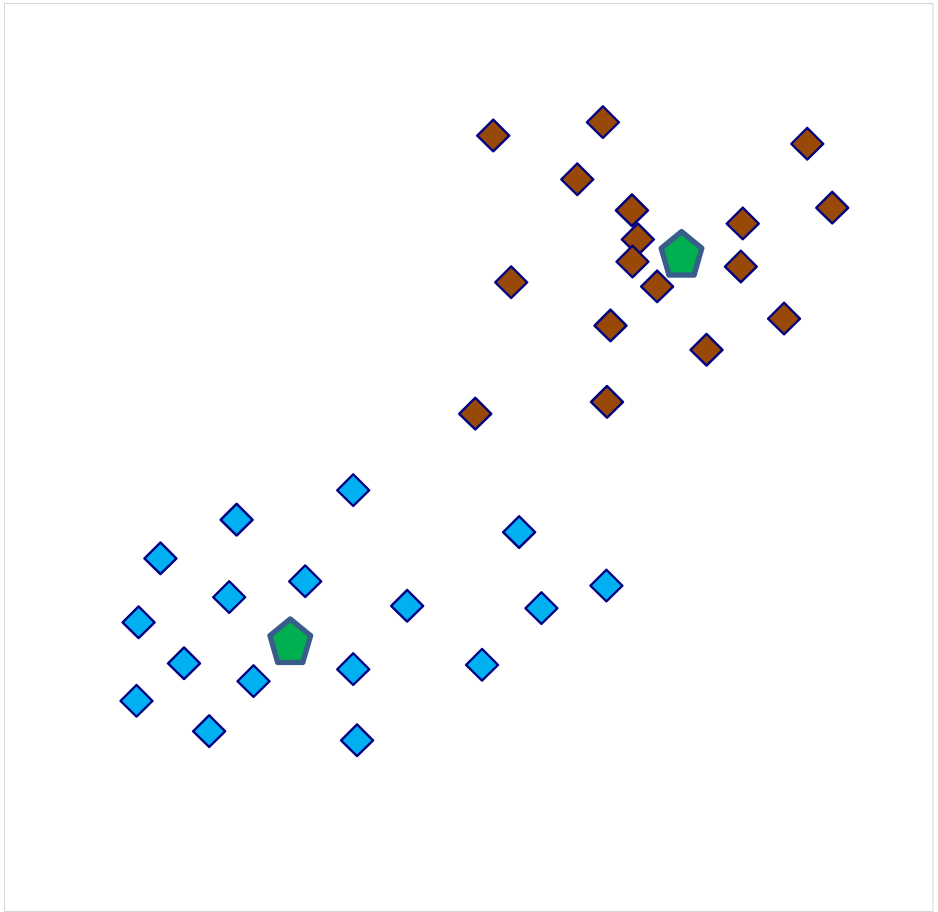
K-均值：一种基于形心的技术

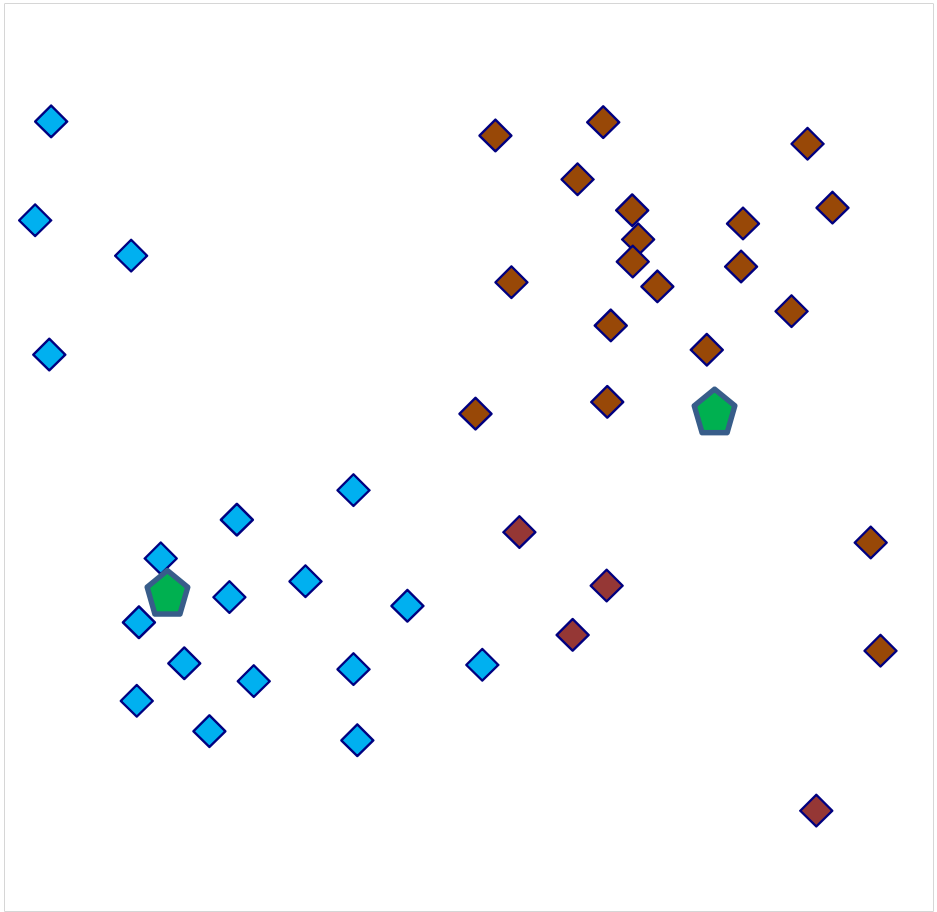
- K-均值聚类缺点2：不适合发现非凸形状的簇，或者大小差别很大的簇。



K-均值：一种基于形心的技术

■ K-均值聚类缺点3：对噪声和离群点敏感。





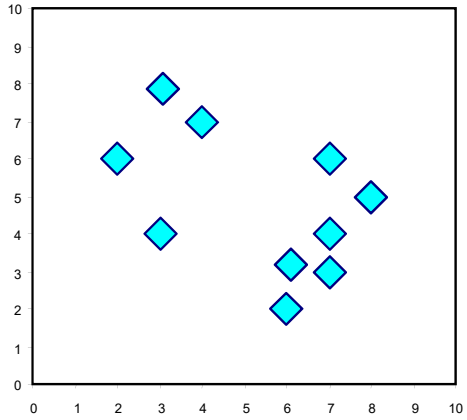
K-中心点：一种基于代表对象的技术

- K均值算法对于离群点是敏感的，因为一个有很大极端值的对象可能显著地扭曲数据的分布。平方差的使用更是严重恶化了这一影响。
- 我们可以在每个簇中选出一个实际的对象来代表该簇。而不是利用均值。其余的每个对象分配到到与其最相似的代表性对象所在的簇中。这样，划分基于绝对误差标准（absolute-error criterion）来进行划分。

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, o_i)$$

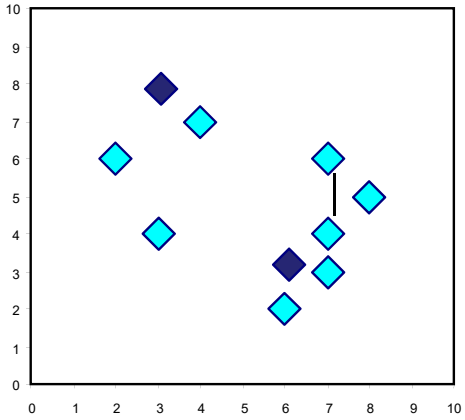
- K-中心点聚类通过最小化该绝对误差，把n个对象划分到k个簇中。

PAM: A Typical K-Medoids Algorithm

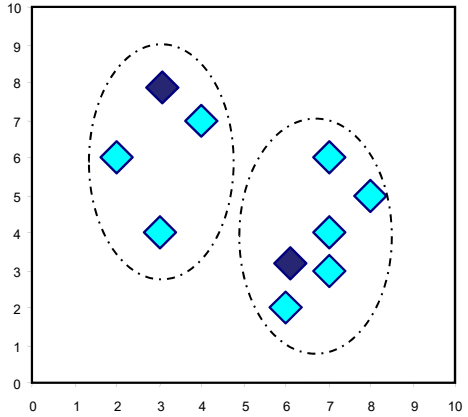


K=2

Arbitrary choose k object as initial medoids

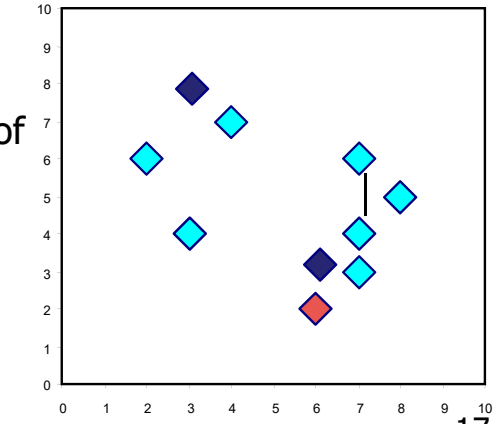


Assign each remaining object to nearest medoids

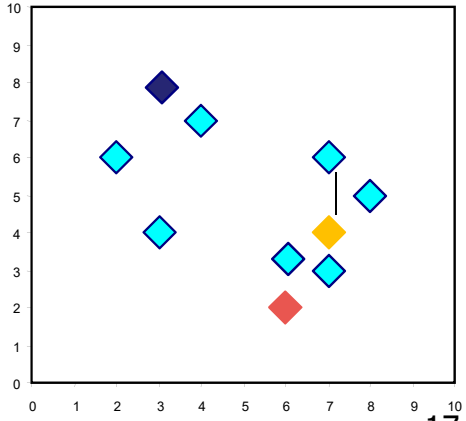


Total Cost = 20

Randomly select a nonmedoid object, O_{ramdom}



Compute total cost of swapping



Total Cost = 26

Swapping O and O_{ramdom}

If quality is improved.

Do loop

Until no change

17

K-中心点：一种基于代表对象的技术

- 每当重新分配发生时，绝对误差E的差对代价函数有影响。因此，如果当前的代表对象被非代表对象所取代，就计算绝对误差值的差。交换的总代价是所有非代表对象所产生的代价之和。

- 如果总代价是负的，实际绝对误差将会减小，o_j可以被o_{random}替代。
- 如果总代价是正的，则当前的代表对象o_j是可接受的，在本次迭代中没有变化发生

K-中心点：一种基于代表对象的技术

输入：

K：结果簇的个数

D：包含n个对象的数据集合。

输出：

k个簇的集合

方法

（1）从D中随机选择k个对象作为初始的代表对象或种子

（2）repeat

（3） 将每一个剩余的对象分配到最近的代表对象所代表的簇

（4） 随机选择一个非代表对象o_{random}

（5） 计算o_{random}代替代表对象o_j的总代价S

（6） if S<0,then o_{random}替换o_j，形成新的k个代表对象的集合

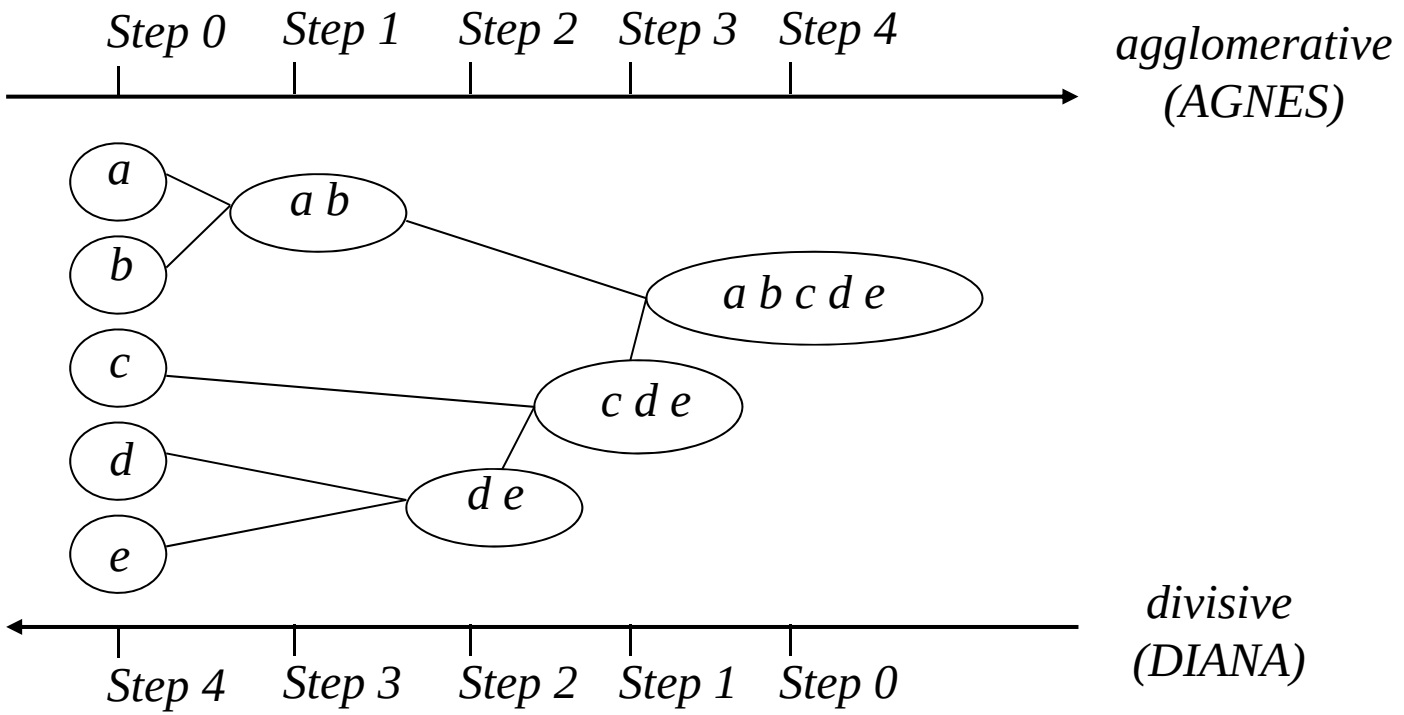
（7）until不再发生变化

层次方法

- 层次聚类方法（hierarchical clustering method）将数据对象组成一棵聚类树。
- 根据层次分解是以自底向上（合并）还是自顶向下（分裂）方式，层次聚类方法可以进一步分为凝聚的和分裂的。
- 一种纯粹的层次聚类方法的质量受限于：一旦合并或分裂执行，就不能修正。也就是说，如果某个合并或分裂决策在后来证明是不好的选择，该方法无法退回并更正。

凝聚的与分裂的层次聚类

- 下图描述了一种凝聚层次聚类算法AGNES和一种分裂层次聚类算法DIANA对一个包含五个对象的数据集合{a,b,c,d,e}的处理过程。

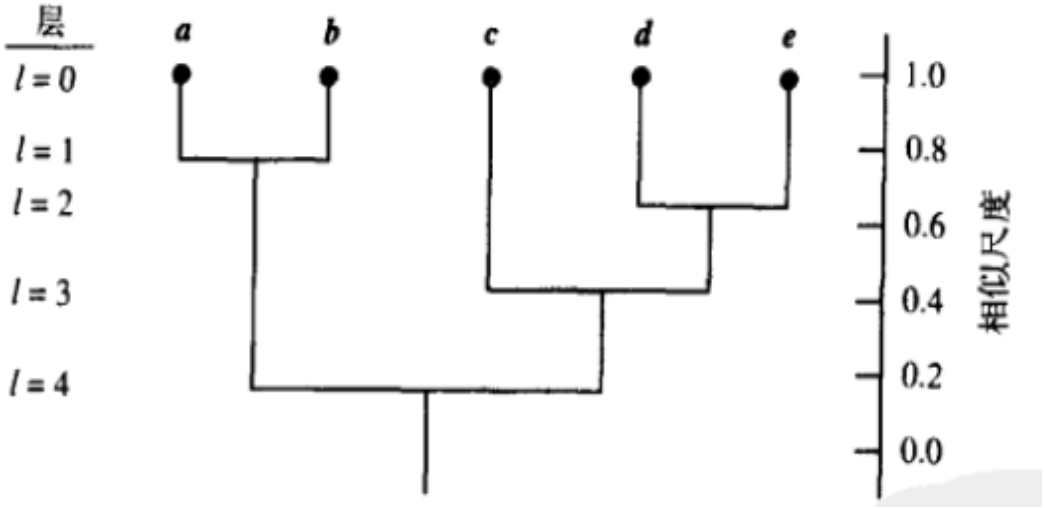


凝聚的与分裂的层次聚类

- 初始，AGNES将每个对象自为一簇，然后这些簇根据某种准则逐步合并，直到所有的对象最终合并形成一个簇。
  - 例如，如果簇C1中的一个对象和簇C2中的一个对象之间的距离是所有属于不同簇的对象间欧氏距离中最小的，则C1和C2合并。
- 在DIANA中，所有的对象用于形成一个初始簇。根据某种原则（如，簇中最近的相邻对象的最大欧氏距离），将该簇分裂。簇的分裂过程反复进行，直到最终每个新簇只包含一个对象。
- 在凝聚或者分裂层次聚类方法中，用户可以定义希望得到的簇数目作为一个终止条件。

凝聚的与分裂的层次聚类

- 通常，使用一种称作树状图的树形结构表示层次聚类的过程。它展示出对象是如何一步步分组的。图2显示图1的五个对象的树状图

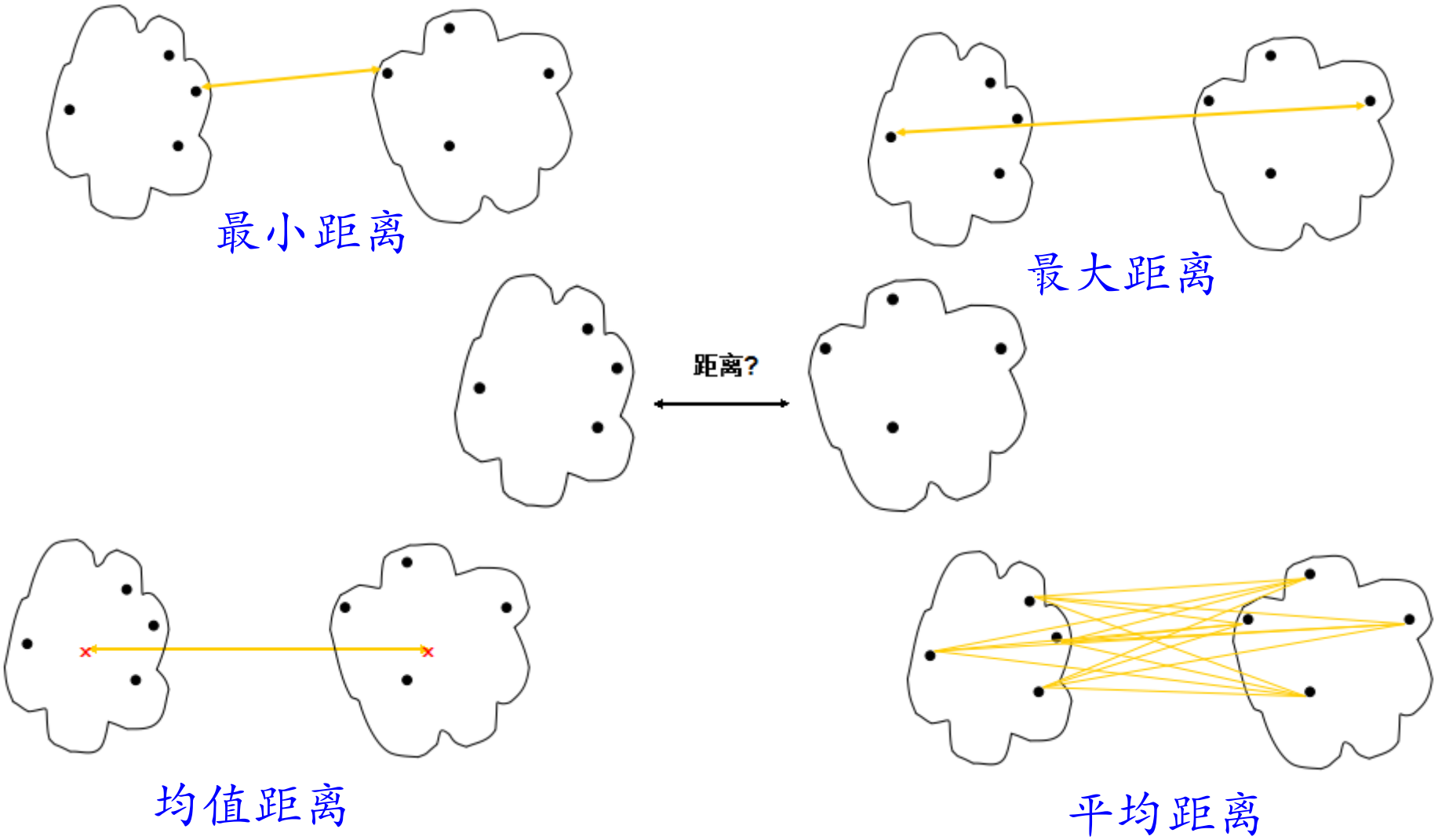


描述簇之间的相似程度。例如，{a, b}和{c, d, e}的相似度大约为0.16。

凝聚的与分裂的层次聚类

- 算法方法的距离度量：
- 四个广泛采用的簇间距离度量方法如下，其中|p-p'|是两个对象或点p和p'之间的距离，m_i是簇C_i的均值，而n_i是簇C_i中对象的数目。
  - 最小距离：$dist_{min}(C_i, C_j) = min_{p \in c_i, p' \in c_j}\{|p - p'|\}$
  - 最大距离: $dist_{max}(C_i, C_j) = max_{p \in c_i, p' \in c_j}\{|p - p'|\}$
  - 均值距离：$dist_{mean}(C_i, C_j) = |m_i - m_j|$
  - 平均距离：$dist_{avg}(C_i, C_j) = \frac{1}{n_i n_j}\sum_{p \in c_i, p' \in c_j}|p - p'|$

凝聚的与分裂的层次聚类



凝聚的与分裂的层次聚类

- 当算法使用最小距离$dist_{min}(C_i, C_j)$ 衡量簇间距离时，有时称它为最近邻聚类算法(nearest-neighbor clustering algorithm)。此外，如果当最近的簇之间的距离超过某个任意的阈值时聚类过程就会终止，则称其为单连接算法(single-linkage algorithm)。

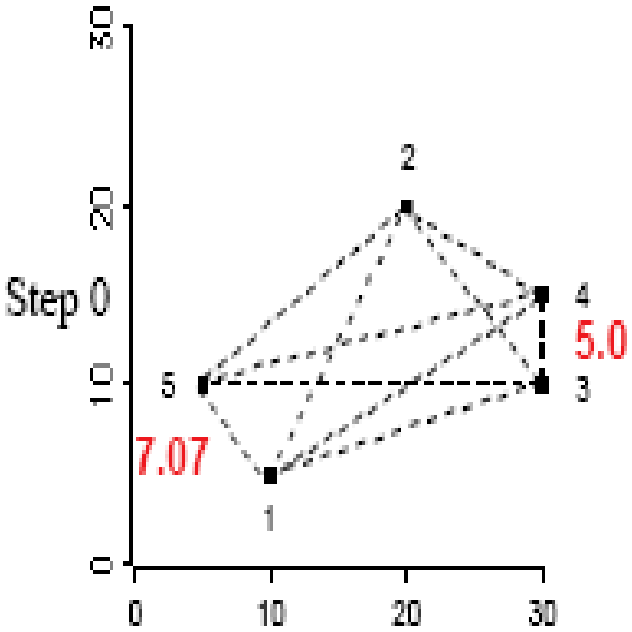
- 当一个算法使用最大距离$dist_{max}(C_i, C_j)$ 度量簇间距离时，有时称为最远邻聚类算法(farthest-neighbor clustering algorithm) 。如果当最近簇之间的最大距离超过某个任意阈值时聚类过程便终止，则称其为全连接算法(complete-linkage algorithm)。

单连接算法例子

- 先将五个样本都分别看成是一个簇，最靠近的两个簇是3和4，因为他们具有最小的簇间距离D（3，4）=5.0。
- 第一步：合并簇3和4，得到新簇集合1，2，(34)，5

	x1	x2
1	10	5
2	20	20
3	30	10
4	30	15
5	5	10

	1	2	3	4	5
1	0.00				
2	18.0	0.00			
3	20.6	14.1	0.00		
4	22.4	11.2	5.00	0.00	
5	7.07	18.0	25.0	25.5	0.00



单连接算法例子

- 更新距离矩阵：

D(1,(34))=min(D(1,3)，D(1,4))=min(20.6，22.4)=20.6

D(2,(34))=min(D(2,3)，D(2,4))=min(14.1，11.2)=11.2

D(5,(34))=min(D(3,5)，D(4,5))=min(25.0，25.5)=25.0

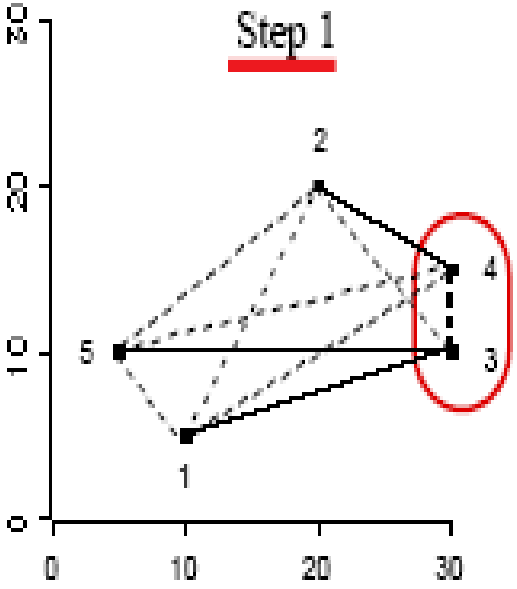
原有簇1，2，5间的距离不变，修改后的距离矩阵如图所示，在四个簇1，2，(34)，5中，最靠近的两个簇是1和5，它们具有最小簇间距离D（1,5）＝7.07。

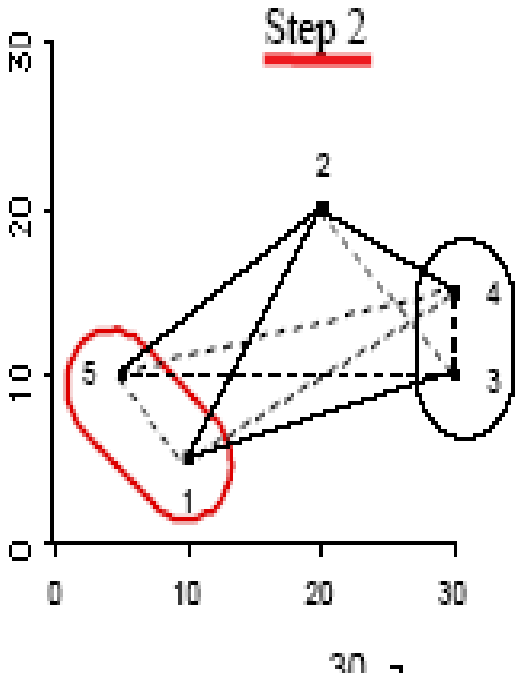
单连接算法例子

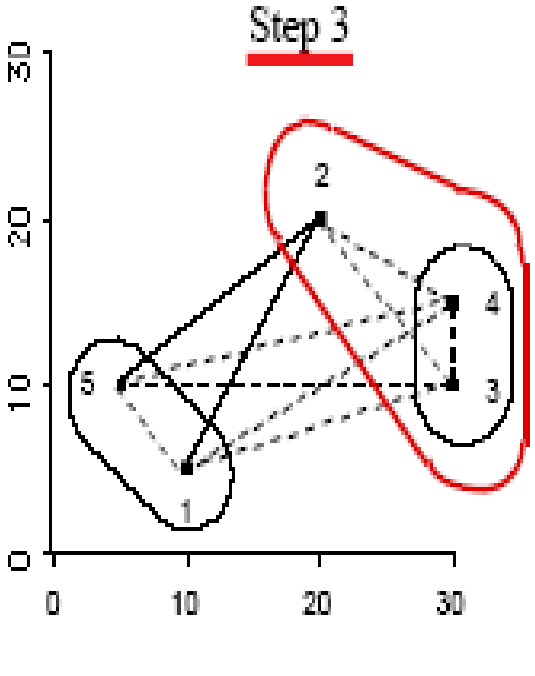
	1	2	5	(34)
1	0.00			
2	18.0	0.00		
5	17.0	18.0	0.00	
(34)	20.6	11.2	25.0	0.00

	2	(34)	(15)
2	0.00		
(34)	11.2	0.00	
(15)	18.0	20.6	0.00

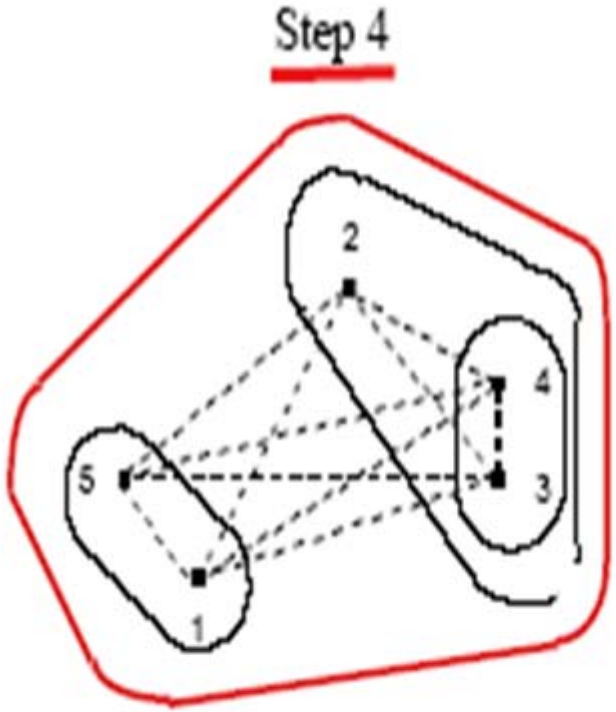
	(15)	(234)
(15)	0.00	
(234)	18.0	0.00

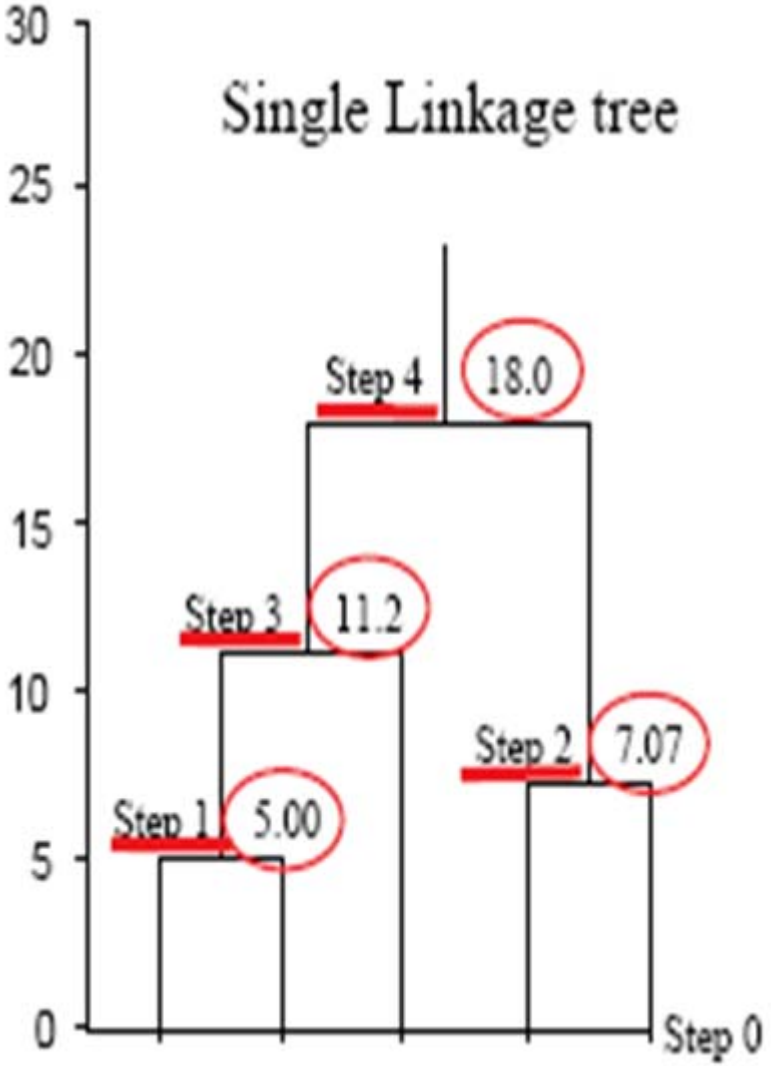






单连接算法例子


Step 4
2
4
5
3
1


Single Linkage tree
30
25
20
15
10
5
0
Step 4
18.0
Step 3
11.2
Step 2
7.07
Step 1
5.00
Step 0

凝聚的与分裂的层次聚类

- 最小和最大度量代表了簇间距离度量的两个极端。它们趋向对离群点或噪声数据过分敏感。
- 使用均值距离和平均距离是对最小和最大距离之间的一种折中方法，而且可以克服离群点敏感性问题。
- 层次聚类方法的困难之处:

(1)层次聚类方法尽管简单，但经常会遇到合并或分裂点选择的困难。因为一旦一组对象合并或者分裂，下一步的处理将对新生成的簇进行。

(2)不具有很好的可伸缩性，因为合并或分裂的决定需要检查和估算大量的对象或簇。

Thank You!

Q&A