



南京理工大学
NANJING UNIVERSITY OF SCIENCE & TECHNOLOGY

聚类

王慧慧

huihuiwang@njust.edu.cn

计算机科学与工程学院

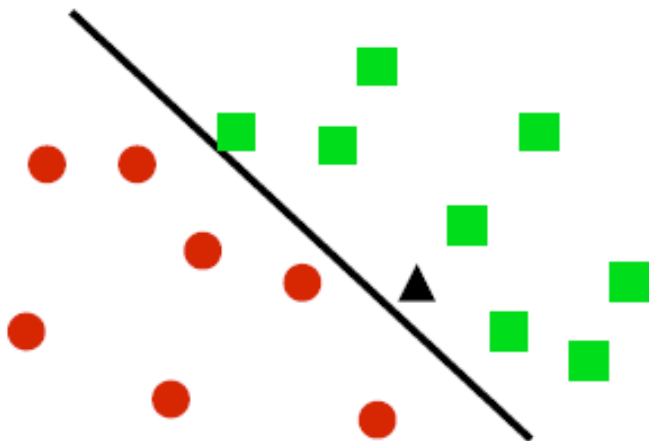


提纲

- 聚类任务描述
- 相似性度量
- 层次聚类
- K-means 算法

分类VS聚类

- 分类 (Supervised)



- 聚类 (Unsupervised)

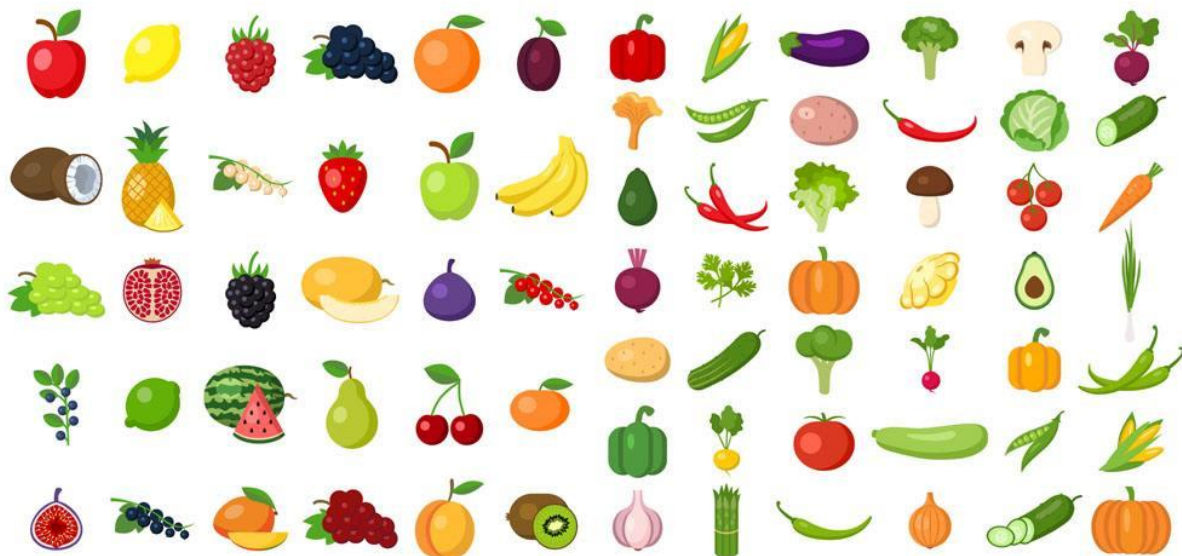


聚类：一种无监督学习（**Unsupervised Learning**）方法，它的目标是将相似的无标签数据点归为一组（簇），使得同一簇内的数据点相似度高，而不同簇之间的数据点相似度低。

例子

- “物以类聚”

对一批没有标出类别的样本集，相似的归为一类，不相似的归为另一类



聚类

- 无监督学习 unsupervised learning
 - 标记未知
 - 揭示数据的内在性质和规律
- 应用最广的无监督学习：聚类

“聚类是把相似的对象通过静态分类的方法分成不同的组别或者更多的子集（subset），这样让在同一个子集中的成员对象都有相似的一些属性。”



“聚类分析指将物理或抽象对象的集合分组成为由类似的对象组成的多个类的分析过程。它是一种重要的人类行为。聚类是将数据分类到不同的类或者簇这样的一个过程，所以同一个簇中的对象有很大的相似性，而不同簇间的对象有很大的相异性。”

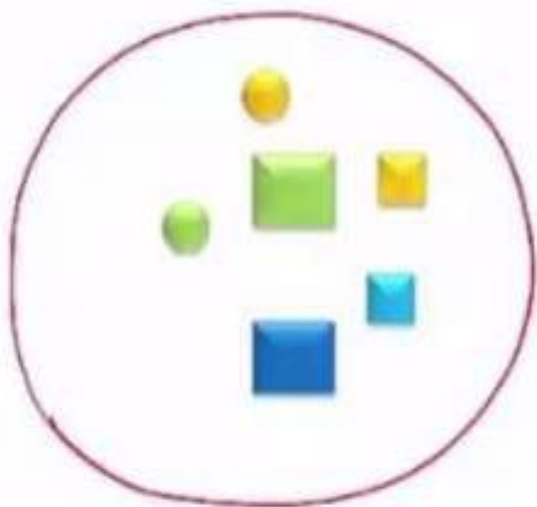
相似性度量

- 在实际应用中，聚类的关键在于寻找一个可以量化任意两个数据点之间相似性的函数。

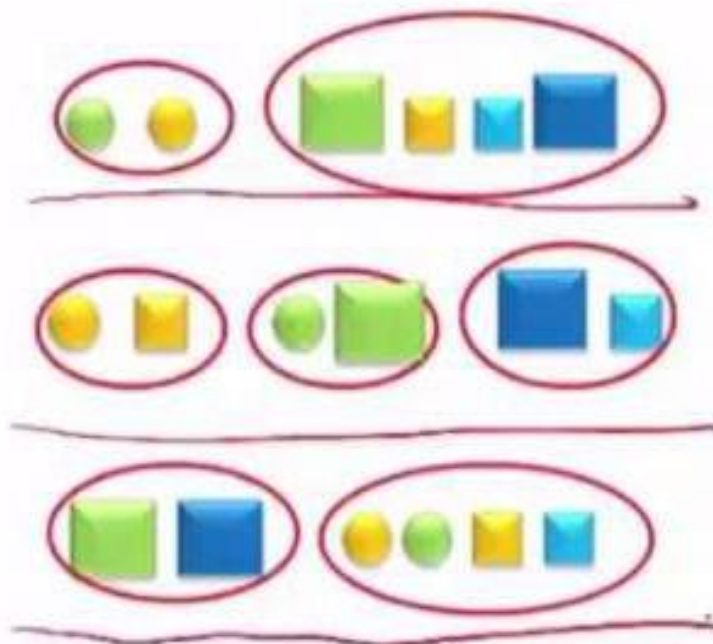
Clustering:

X: (颜色, 形状, 大小)

Data:



For all the data, Y=?



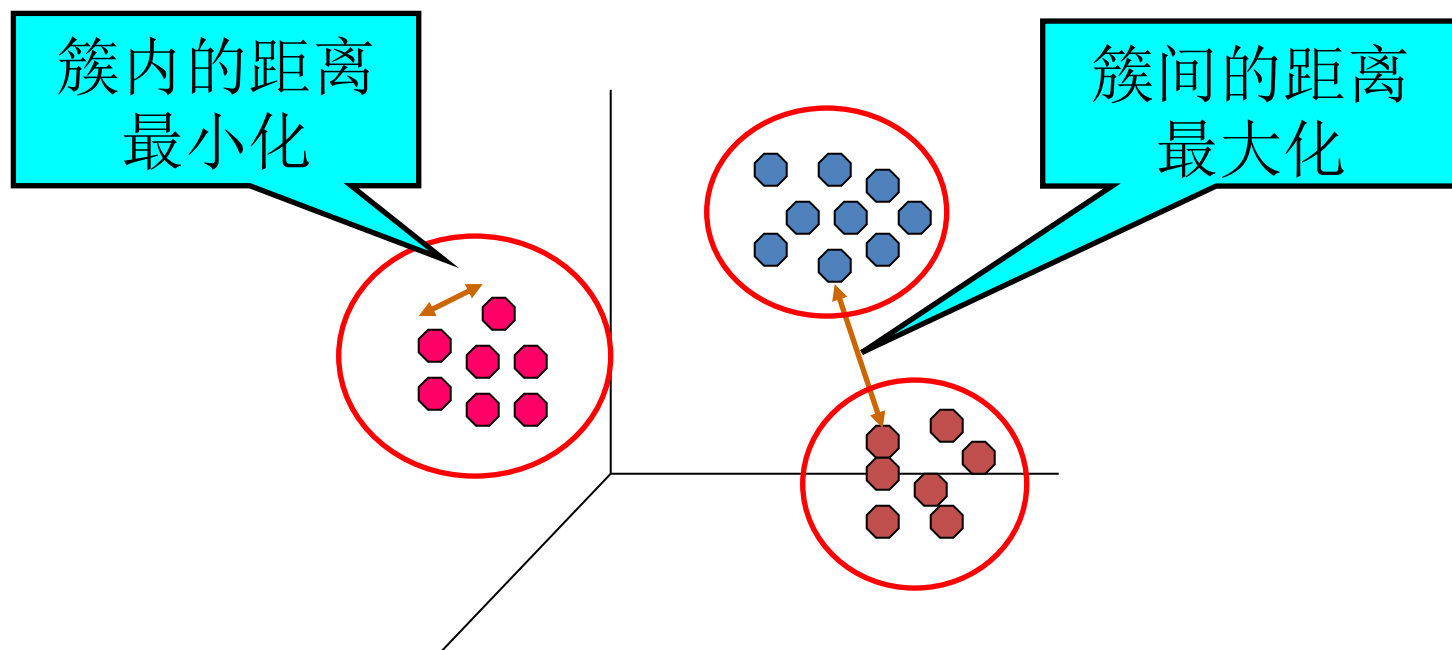
形状

颜色

大小

相似性度量

- 把整个模式样本集的特征向量看成是分布在特征空间中的一些点，**点与点之间的距离**即可作为模式相似性的测量依据。
 - 一个样本由 d 个特征组成，则其特征向量可表示为 $\mathbf{x}^T = [x_1, x_2, \dots, x_d]$ 。



相似度或距离

- 聚类的形式化描述:
- 样本集: $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$
- 每个样本: $\mathbf{x}_i = (x_{i1}; x_{i2}; \dots; x_{id})$
- 划分为k个不相交的簇: $\{C_l \mid l = 1, 2, \dots, k\}$
- 簇标记: $\lambda_j \in \{1, 2, \dots, k\}$
- 聚类的结果可用包含m个元素的簇标记向量 $\boldsymbol{\lambda} = (\lambda_1; \lambda_2; \dots; \lambda_m)$ 表示
- 聚类的重要性:
 - 其它学习任务的前驱过程;
- 可以用距离或者相似性去衡量数据点之间的相似程度

距离度量：距离越小，数据点越相似

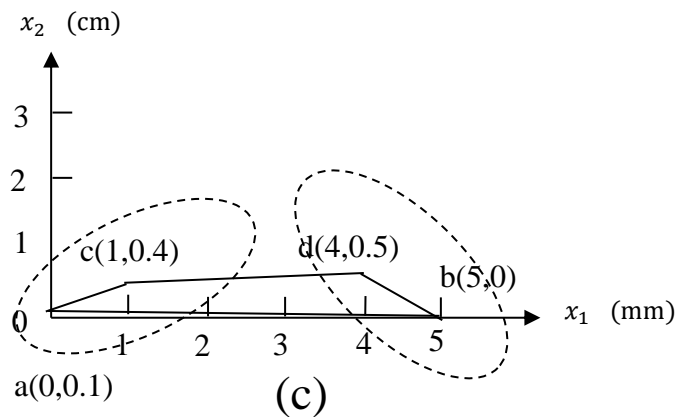
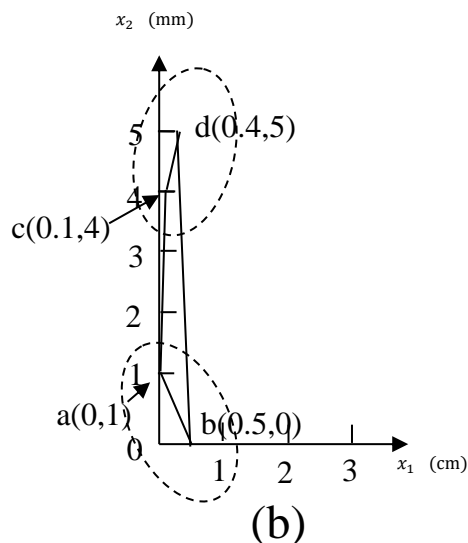
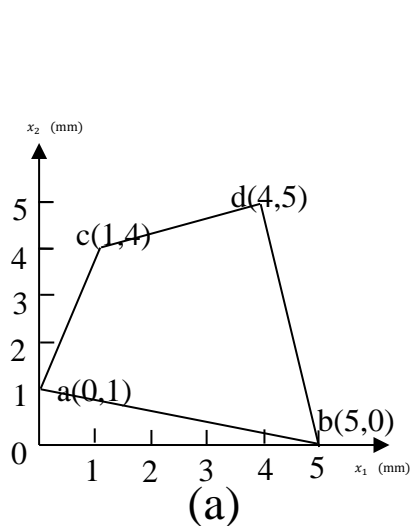
- 闵可夫斯基距离

$$d(x_i, x_j) = (\sum_{u=1}^d |x_{iu} - x_{ju}|^p)^{\frac{1}{p}}$$

- P=2时，欧氏距离

$$d(x_i, x_j) = \sqrt{\sum_{u=1}^d (x_{iu} - x_{ju})^2}$$

- 欧氏距离:适用于低维数据



解决方法：使特征数据标准化，使其与变量的单位无关

距离度量：距离越小，数据点越相似

- $p=1$ 时，曼哈顿距离
$$d(x_i, x_j) = \sum_{u=1}^d |x_{iu} - x_{ju}|$$

- 曼哈顿距离:适用于低维数据更适用于高维空间，因为其计算的是各维度的独立差异，对高维数据更稳定，受“维度灾难”影响较小

- $p \rightarrow \infty$ 时，即为切比雪夫距离。

$$d(x_i, x_j) = \max_u |x_{iu} - x_{ju}| \quad u \in [1, d]$$

距离度量：距离越小，数据点越相似

- 马氏距离:适用于不同尺度的数据

$$D_M(x_i, x_j) = \sqrt{(x_i - x_j)^T \Sigma^{-1} (x_i - x_j)}$$

- Σ 是协方差矩阵，衡量特征之间的相关性；
- **去除特征尺度的影响**：通过 Σ^{-1} 对每个特征除以其方差，使得每个特征的数值范围都被缩放到相同的尺度，因此不同单位的特征不会影响距离计算。
- **消除特征之间相关性的干扰**： Σ^{-1} 用于考虑数据在各个特征上的差异性，通过 Σ^{-1} 使得特征之间正交。
- 当 $\Sigma = I$ 时，马氏距离为欧氏距离。
- 问题：协方差矩阵在实际应用中难以计算。

皮尔逊相关系数

衡量线性相关性

皮尔逊相关系数 (Pearson Correlation Coefficient, PCC) : 衡量两个变量之间**线性关系强度和方向**的统计指标。它的值介于 -1 和 1 之间。

$$\rho(x, y) = \frac{\text{cov}(x, y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad \bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$$

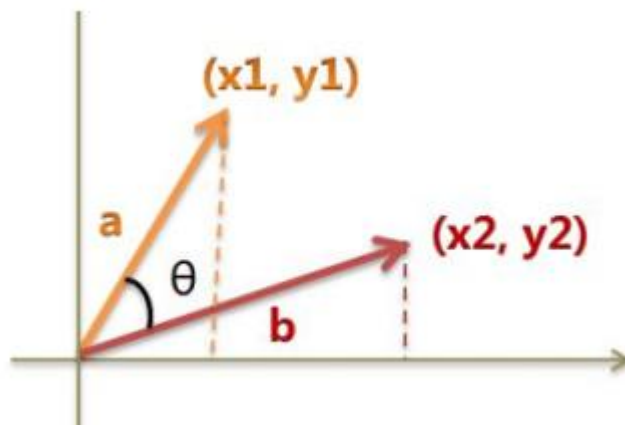
- **分子**: 表示变量 x 和 y 的协方差 (即两者之间的联合变化程度) 。
当两个变量同时变化时, 分子值较大; 当它们变化的方向相反时, 分子为负。
- **分母**: 表示 x 和 y 的标准差的乘积, 衡量了单个变量的波动性。 12

夹角余弦

- 样本之间的相似度也可以用夹角余弦（cosine）来表示。
- 取值介于-1与1之间，夹角余弦越接近于1，表示样本越相似
- 向量A与B之间的夹角余弦相似度定义为

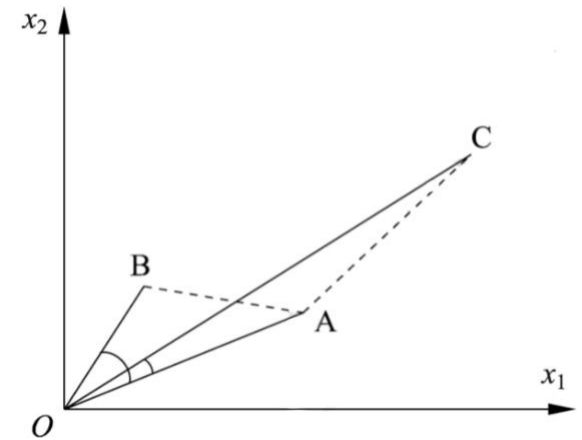
余弦相似度：

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$
$$= \frac{\sum X Y}{\sqrt{\sum_{i=1}^n X_i^2} \sqrt{\sum_{i=1}^n Y_i^2}}$$



相似度

- 用距离度量相似度时，距离越小样本越相似
 - 用相关系数时，相关系数越大样本越相似
 - 注意不同相似度度量得到的结果并不一定一致。
-
- 从右图可以看出，如果从距离的角度看，
A和B比A和C更相似
 - 但从相关系数的角度看，
 - A和C比A和B更相似。



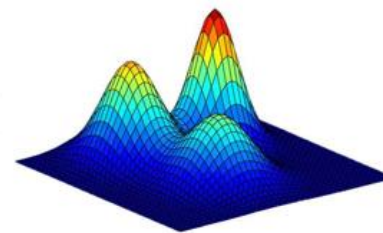
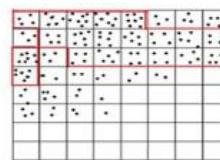
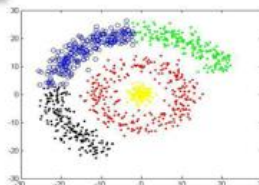
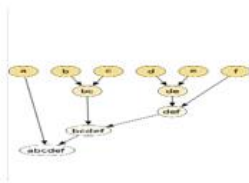
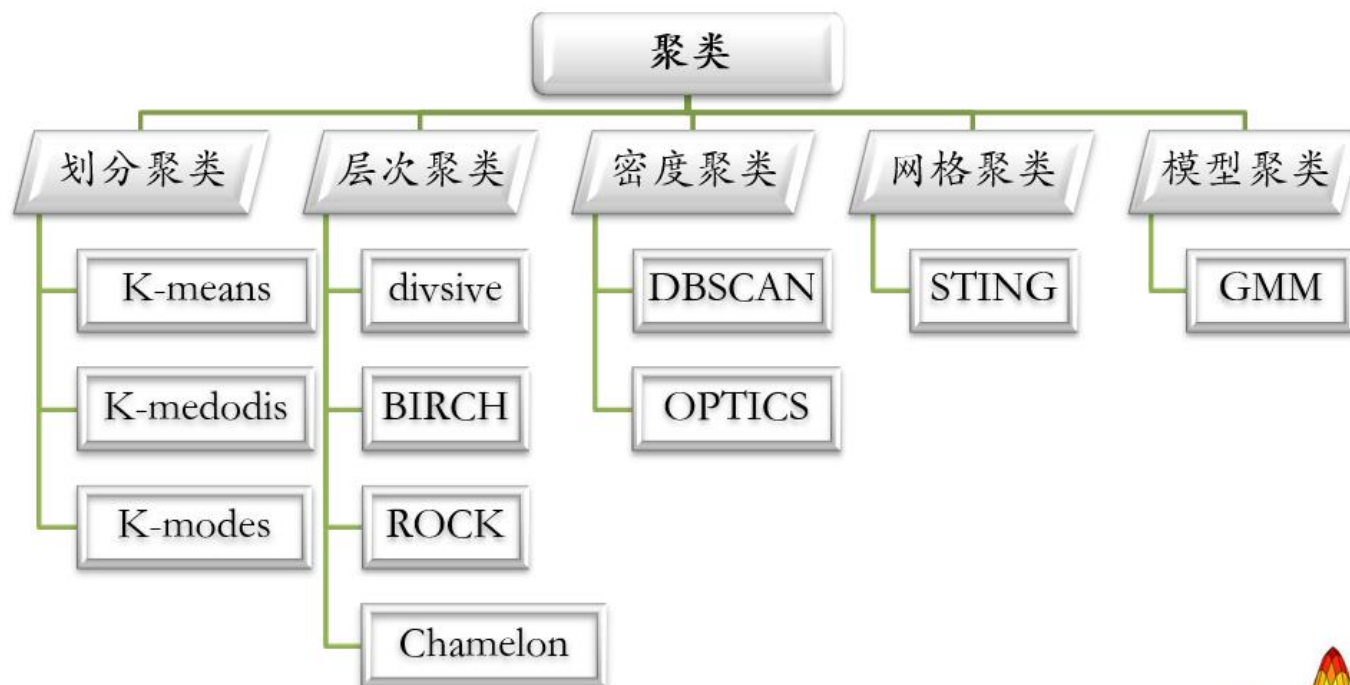
聚类准则

- 聚类准则函数法
 - 一种聚类准则函数 J 的定义

$$J = \sum_{j=1}^c \sum_{x \in S_j} \|x - m_j\|^2$$

其中， c 为聚类类别的数目， S_j 为第 j 个类别的样本集合， $m_j = \frac{1}{N_j} \sum_{x \in S_j} x$ 为属于 S_j 集合的样本均值向量， N_j 为 S_j 中的样本数目。这里，以均值向量 m_j 为 S_j 中样本的代表。

聚类方法



层次聚类

- 层次聚类假设类别之间存在层次结构，将样本聚到层次化的类中。
- 层次聚类又有聚合（agglomerative）或自下而上（bottom-up）聚类、分裂（divisive）或自上而下（top-down）聚类两种方法。
- 因为每个样本只属于一个类，所以层次聚类属于硬聚类

层次聚类

- 聚合聚类开始将每个样本各自分到一个类
 - 之后将相距最近的两类合并，建立一个新的类
 - 重复此操作直到满足停止条件
 - 得到层次化的类别
-
- 分裂聚类开始将所有样本分到一个类
 - 之后将已有类中相距最远的样本分到两个新的类
 - 重复此操作直到满足停止条件
 - 得到层次化的类别

聚合聚类的具体过程

- 对于给定的样本集合，开始将每个样本分到一个类
- 然后按照一定规则，例如类间距离最小，将最满足规则条件的两个类进行合并
- 如此反复进行，每次减少一个类，直到满足停止条件，如所有样本聚为一类。

聚合聚类

- 聚合聚类需要预先确定下面三个要素
 - 距离或相似度
 - 闵可夫斯基距离
 - 相关系数
 - 夹角余弦
 - 合并规则
 - 类间距离最小
 - 类间距离可以是最短距离、最长距离、中心距离、平均距离
 - 停止条件
 - 停止条件可以是类的个数达到闭值（极端情况类的个数是1）
 - 类的直径超过阈值

聚合聚类算法

输入: n 个样本组成的样本集合及样本之间的距离;

输出: 对样本集合的一个层次化聚类。

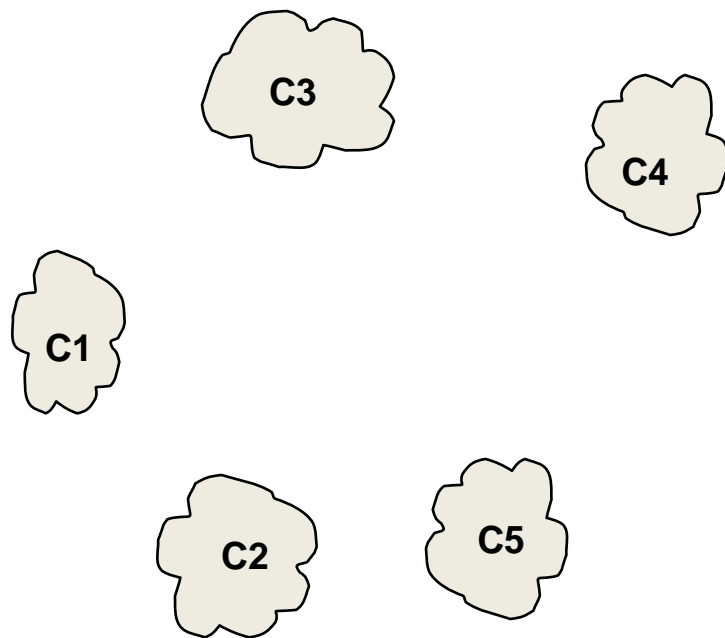
- (1) 计算 n 个样本两两间的欧氏距离 d_{ij} , 记作矩阵 $D = [d_{ij}]_{n \times n}$ 。
- (2) 构造 n 个类, 每个类只包含一个样本。
- (3) 合并类间距离最短的两个类, 其中最短距离为类间距离, 构建一个新类。
- (4) 合并后, 计算新类与当前各类的距离。若满足停止条件, 终止计算, 否则回到步(3)。

停止条件: 1) 类间最小距离超过阈值; 2) 样本聚成设置的簇数。

聚合聚类算法的复杂度是 $O(n^3m)$, 其中 m 是样本的维数。

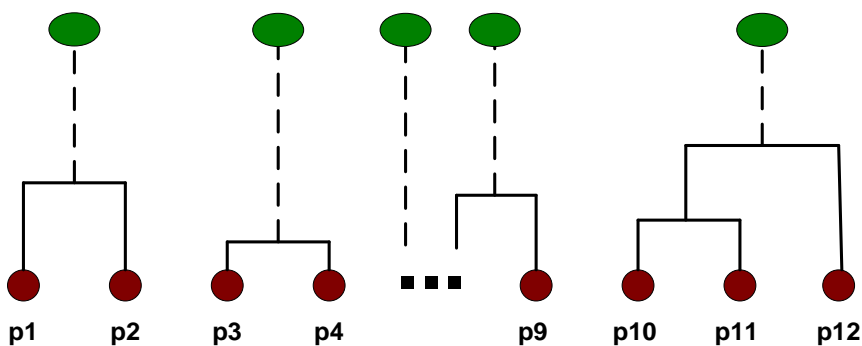
层次聚类算法

— 簇与簇之间的距离

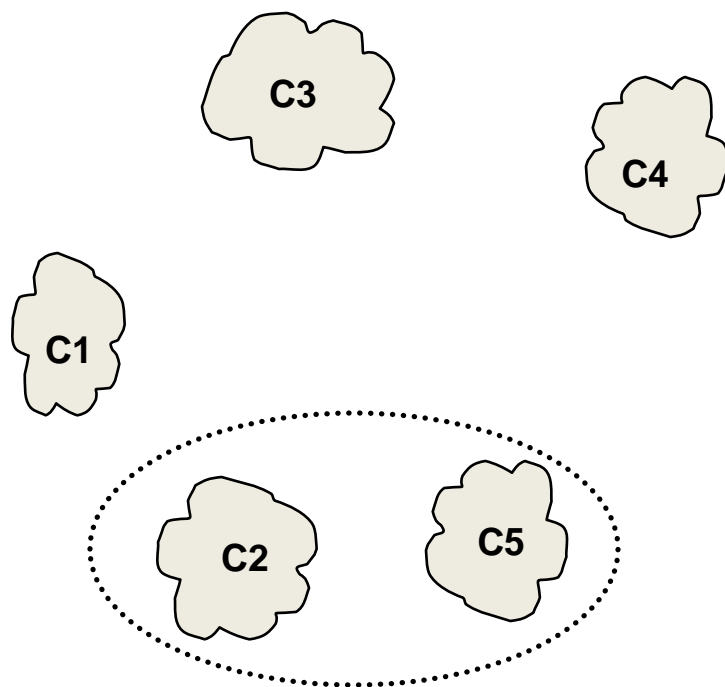


	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

距离矩阵

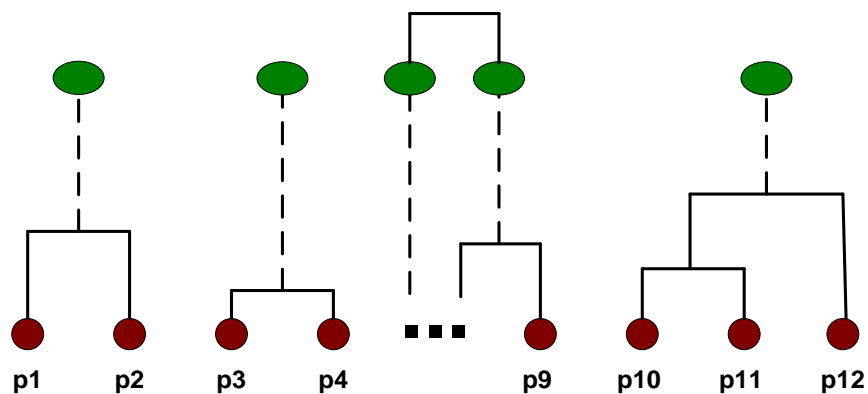


层次聚类算法



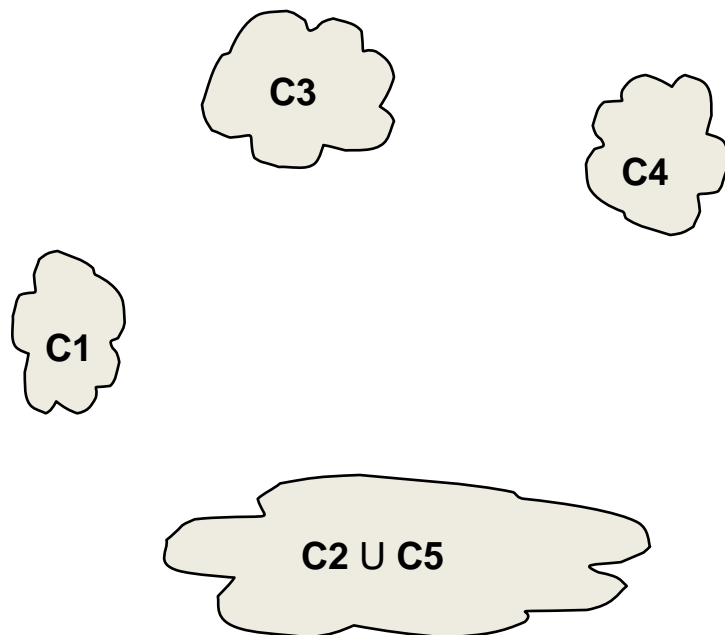
	C1	C2	C3	C4	C5
C1					
C2					
C3					
C4					
C5					

距离矩阵



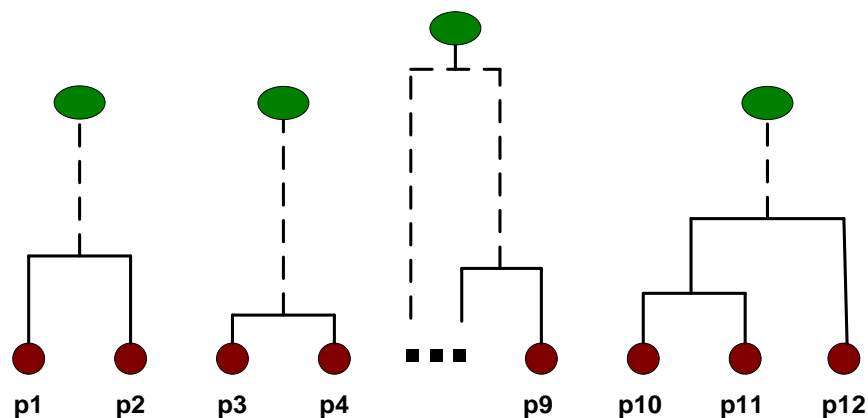
层次聚类算法

合并后如何更新距离矩阵

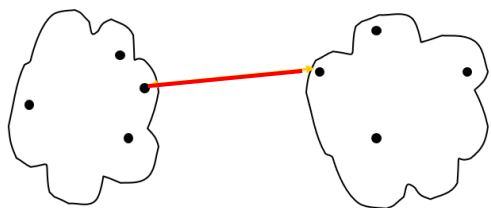


	C1	$C2 \cup C5$	C3	C4
C1		?		
$C2 \cup C5$?	?	?	?
C3		?		
C4		?		

距离矩阵

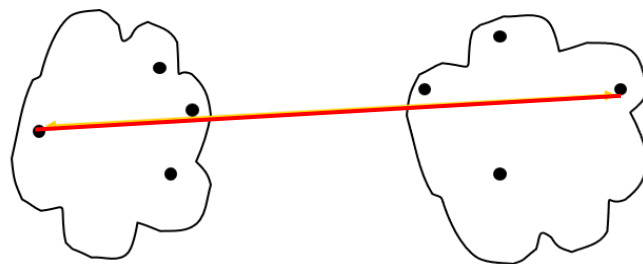


类间距离



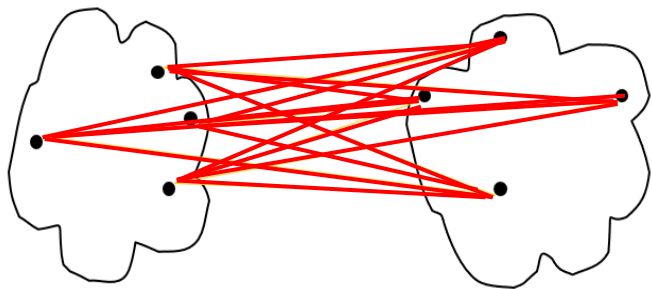
最短距离或单连接

$$D_{pq} = \min \{d_{ij} | x_i \in G_p, x_j \in G_q\}$$



最长距离或完全连接

$$D_{pq} = \max \{d_{ij} | x_i \in G_p, x_j \in G_q\}$$



平均距离

$$D_{pq} = \frac{1}{n_p n_q} \sum_{x_i \in G_p} \sum_{x_j \in G_q} d_{ij}$$

- 中心距离、重心距离等

例子

- 给定6个样本的集合，按类间最短距离进行聚类

$$X_1 = [0, 3, 1, 2, 0]^T \quad X_2 = [1, 3, 0, 1, 0]^T \quad X_3 = [3, 3, 0, 0, 1]^T$$

$$X_4 = [1, 1, 0, 2, 0]^T \quad X_5 = [3, 2, 1, 2, 1]^T \quad X_6 = [4, 1, 1, 1, 0]^T$$

解：（1）将每一样本看作单独一类，得：

$$G_1(0) = \{X_1\} \quad G_2(0) = \{X_2\} \quad G_3(0) = \{X_3\}$$

$$G_4(0) = \{X_4\} \quad G_5(0) = \{X_5\} \quad G_6(0) = \{X_6\}$$

计算各类间欧氏距离：

$$\begin{aligned} D_{12}(0) &= \|X_1 - X_2\| \\ &= [(x_{11} - x_{21})^2 + (x_{12} - x_{22})^2 + (x_{13} - x_{23})^2 + (x_{14} - x_{24})^2 \\ &\quad + (x_{15} - x_{25})^2]^{1/2} \\ &= [1 + 0 + 1 + 1 + 0]^{1/2} = \sqrt{3} \end{aligned}$$

- 其中 d_{ij} 表示第 i 个样本与第 j 个样本之间的欧氏距离。

得距离矩阵 $D(0)$:

$D(0)$	$G_1(0)$	$G_2(0)$	$G_3(0)$	$G_4(0)$	$G_5(0)$	$G_6(0)$
$G_1(0)$	0					
$G_2(0)$	* $\sqrt{3}$	0				
$G_3(0)$	$\sqrt{15}$	$\sqrt{6}$	0			
$G_4(0)$	$\sqrt{6}$	$\sqrt{5}$	$\sqrt{13}$	0		
$G_5(0)$	$\sqrt{11}$	$\sqrt{8}$	$\sqrt{6}$	$\sqrt{7}$	0	
$G_6(0)$	$\sqrt{21}$	$\sqrt{14}$	$\sqrt{8}$	$\sqrt{11}$	$\sqrt{4}$	0

计算聚类后的距离矩阵 $D(1)$:

由 $D(0)$ 递推出 $D(1)$ 。

(2) 将最小距离 $\sqrt{3}$ 对应的类 $G_1(0)$ 和 $G_2(0)$ 合并为1类，得新的分类。

$$G_{12}(1) = \{G_1(0), G_2(0)\}$$

$$G_3(1) = \{G_3(0)\} \quad G_4(1) = \{G_4(0)\}$$

$$G_5(1) = \{G_5(0)\} \quad G_6(1) = \{G_6(0)\}$$

$D(0)$	$G_1(0)$	$G_2(0)$	$G_3(0)$	$G_4(0)$	$G_5(0)$	$G_6(0)$
$G_1(0)$	0					
$G_2(0)$	$\sqrt{3}$	0				
$G_3(0)$	<u>$\sqrt{15}$</u>	<u>$\sqrt{6}$</u>	0			
$G_4(0)$	<u>$\sqrt{6}$</u>	<u>$\sqrt{5}$</u>	$\sqrt{13}$	0		
$G_5(0)$	<u>$\sqrt{11}$</u>	<u>$\sqrt{8}$</u>	$\sqrt{6}$	$\sqrt{7}$	0	
$G_6(0)$	<u>$\sqrt{21}$</u>	<u>$\sqrt{14}$</u>	$\sqrt{8}$	$\sqrt{11}$	$\sqrt{4}$	0

$D(1)$	$G_{12}(1)$	$G_3(1)$	$G_4(1)$	$G_5(1)$	$G_6(1)$
$G_{12}(1)$	0				
$G_3(1)$	$\sqrt{6}$	0			
$G_4(1)$	$\sqrt{5}$	$\sqrt{13}$	0		
$G_5(1)$	$\sqrt{8}$	$\sqrt{6}$	$\sqrt{7}$	0	
$G_6(1)$	$\sqrt{14}$	$\sqrt{8}$	$\sqrt{11}$	$\sqrt{4}$	0

$D(2)$	$G_{12}(2)$	$G_3(2)$	$G_4(2)$	$G_{56}(2)$
$G_{12}(2)$	0			
$G_3(2)$	$\sqrt{6}$	0		
$G_4(2)$	$\sqrt{5}$	$\sqrt{13}$	0	
$G_{56}(2)$	$\sqrt{8}$	$\sqrt{6}$	$\sqrt{7}$	0

(3) 将 $D(1)$ 中最小值 $\sqrt{4}$
对应的类合为一类,
得 $D(2)$ 。

(4) 将 $D(2)$ 中最小值 $\sqrt{5}$ 对应的类合为一类，得 $D(3)$ 。

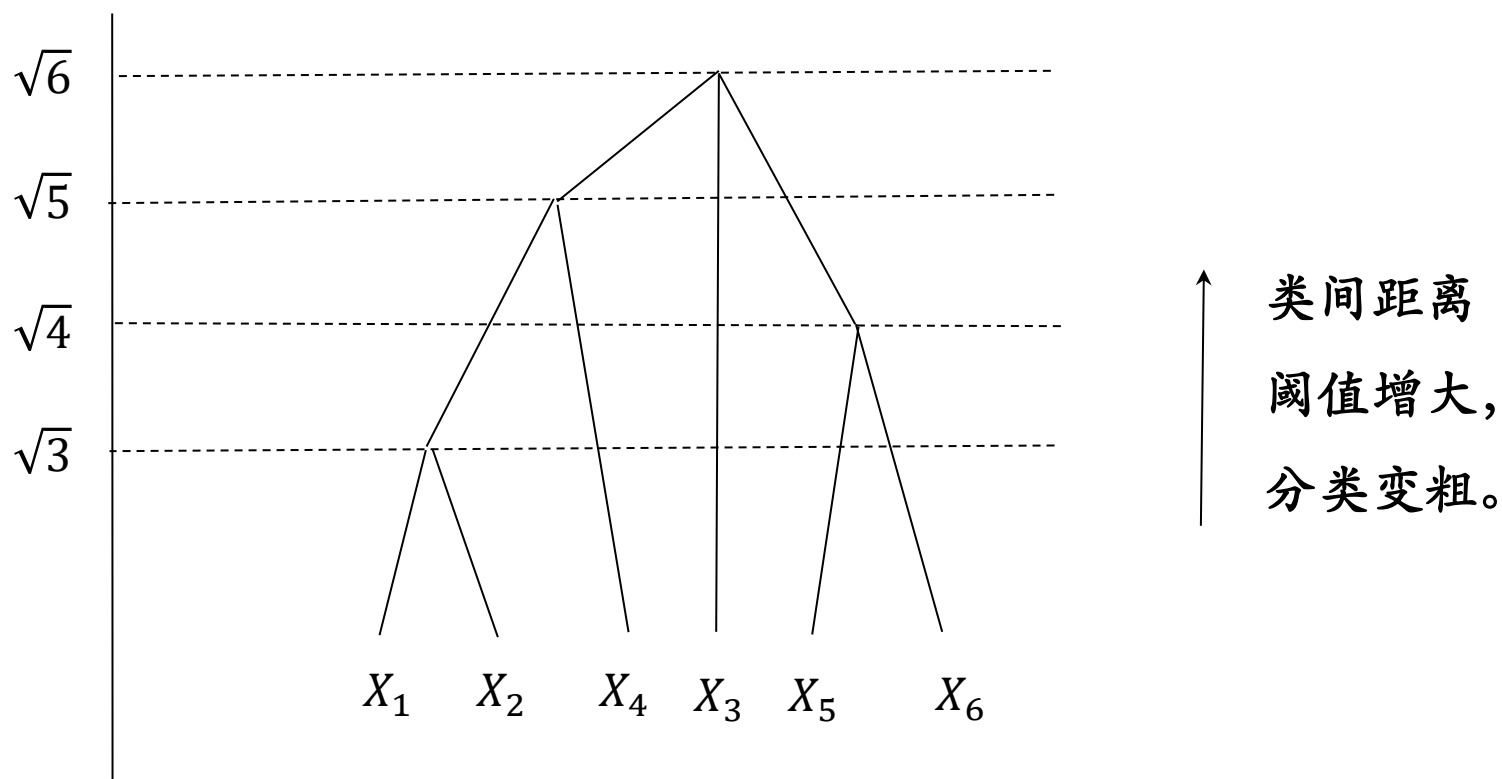
$D(2)$	$G_{12}(2)$	$G_3(2)$	$G_4(2)$	$G_{56}(2)$
$G_{12}(2)$	0			
$G_3(2)$	<u>$\sqrt{6}$</u>	0	<u>$\sqrt{13}$</u>	
$G_4(2)$	* $\sqrt{5}$	$\sqrt{13}$	0	
$G_{56}(2)$	<u>$\sqrt{8}$</u>	$\sqrt{6}$	<u>$\sqrt{7}$</u>	0

$D(3)$	$G_{124}(3)$	$G_3(3)$	$G_{56}(3)$
$G_{124}(3)$	0		
$G_3(3)$	$\sqrt{6}$	0	
$G_{56}(3)$	$\sqrt{7}$	$\sqrt{6}$	0

若给定的阈值为 $T = \sqrt{5}$ ， $D(3)$ 中的最小元素 $\sqrt{6} > T$ ，聚类结束。

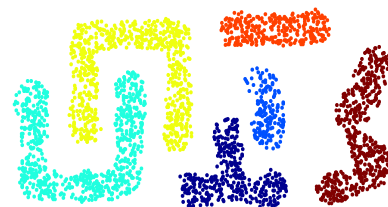
$$G_1 = \{X_1, X_2, X_4\} \quad G_2 = \{X_3\} \quad G_3 = \{X_5, X_6\}$$

若无阈值，继续分下去，最终全部样本归为一类。可给出聚类过程的树状表示图。



层次聚类法的树状表示

优点



优点

- 1) 生成层次化聚类结构，通过聚类树直观展示效果
- 2) 不需要事先指定聚类个数，可选择合适数目
- 3) 可以处理各种类型数据，包括数值型、离散型等。

缺点

- 1) 计算复杂度高，不适合大规模数据。
- 2) 对于“噪声”和孤立点数据敏感，容易受到离群点影响错误分类

K-Means聚类

- K-Means聚类是基于样本集合划分的聚类算法。
- K-Means聚类将样本集合划分为K个子集，构成K个类，将n个样本分到K个类中，每个样本到其所属类的中心的距离最小。
- 每个样本只能属于一个类，所以K-Means聚类是硬聚类。

模型

- K-Means聚类的模型是一个从样本到类的函数。
- 基于聚类准则函数最小化
- 准则函数：簇类中每一样本点到该类中心的距离平方和最小化。
- 对于第 j 个簇，准则函数定义为

$$J_j = \sum_{i=1}^{N_j} \|X_i - Z_j\|^2, \quad X_i \in S_j$$

- S_j 是第 j 个簇，聚类中心为 Z_j ； N_j 是第 j 个簇中所包含的样本个数。

策略

- 对所有 K 个模式类有

$$J = \sum_{j=1}^K \sum_{i=1}^{N_j} \|X_i - Z_j\|^2, \quad X_i \in S_j$$

K-均值算法的聚类准则：聚类中心的选择应使准则函数 J 极小，即使 J_j 的值极小。

- 相似的样本被聚到同类时，准则函数值最小，这个目标函数的最优化能达到聚类的目的。但是，这是一个组合优化问题， n 个样本分到 K 类，所有可能分法的数目是：

$$S(n, k) = \frac{1}{k!} \sum_{l=1}^k (-1)^{k-l} \binom{k}{l} k^l$$

- NP困难问题。现实中采用迭代的方法求解。

策略

- 对所有 K 个模式类有

$$J = \sum_{j=1}^K \sum_{i=1}^{N_j} \|X_i - Z_j\|^2, \quad X_i \in S_j$$

K-均值算法的聚类准则：聚类中心的选择应使准则函数 J 极小，

因
$$\frac{\partial}{\partial Z_j} \sum_{i=1}^{N_j} \|X_i - Z_j\|^2 = \frac{\partial}{\partial Z_j} \sum_{i=1}^{N_j} (X_i - Z_j)^T (X_i - Z_j) = 0$$

可解得
$$Z_j = \frac{1}{N_j} \sum_{i=1}^{N_j} X_i, \quad X_i \in S_j$$

上式表明， S_j 类的聚类中心应选为该类型样本的均值。

算法

• 算法描述

- (1) 任选 K 个初始聚类中心： $Z_1(1), Z_2(1), \dots, Z_K(1)$
- (2) 按最小距离原则将其余样本分配到 K 个聚类中心中的某一个，即：

$$\min\{\|X - Z_i(k)\|, i = 1, 2, \dots, K\} = \|X - Z_j(k)\| = D_j(k), \text{ 则 } X \in S_j(k)$$

注意： k 是迭代运算次序号； K 是聚类中心的个数。

- (3) 计算各个聚类中心的新向量值： $Z_j(k+1) \quad j = 1, 2, \dots, K$

$$Z_j(k+1) = \frac{1}{N_j} \sum_{X \in S_j(k)} X \quad j = 1, 2, \dots, K$$

N_j ：第 j 类的样本数。

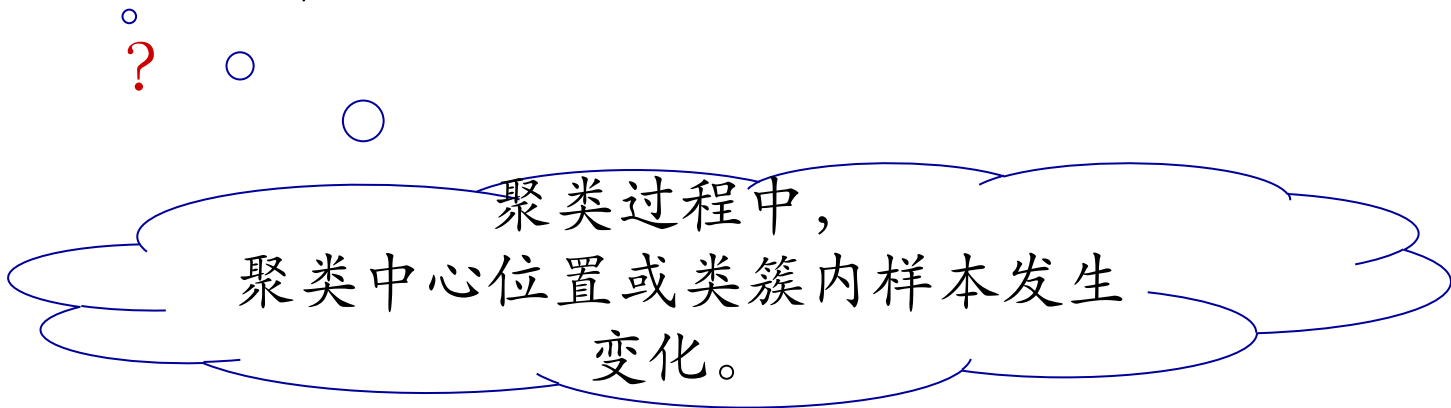
算法

(4) 如果 $Z_j(k+1) \neq Z_j(k)$ $j = 1, 2, \dots, K$, 则回到 (2) , 将样本逐个重新分类, 重复迭代计算。

如果 $Z_j(k+1) = Z_j(k)$ $j = 1, 2, \dots, K$, 算法收敛, 计算完毕。

“动态”聚类法

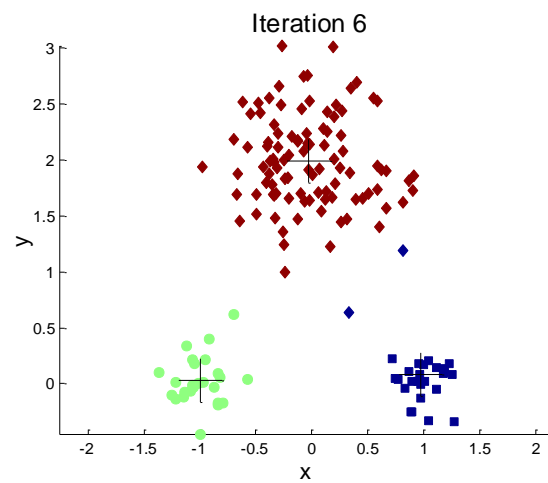
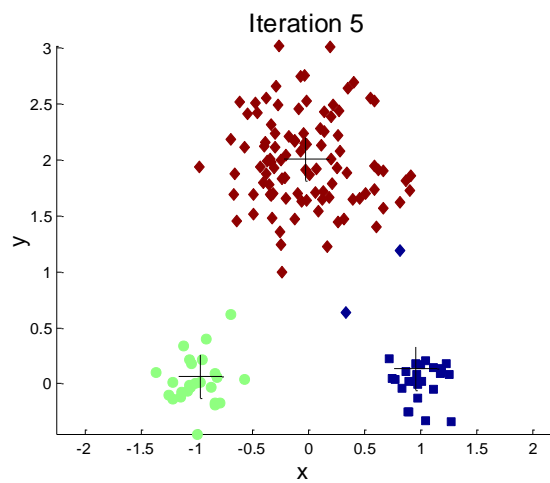
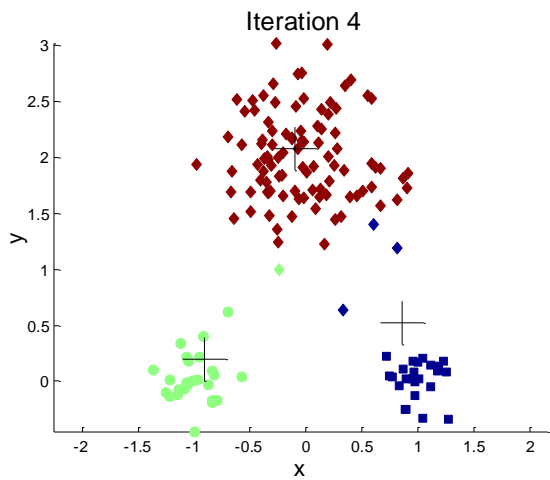
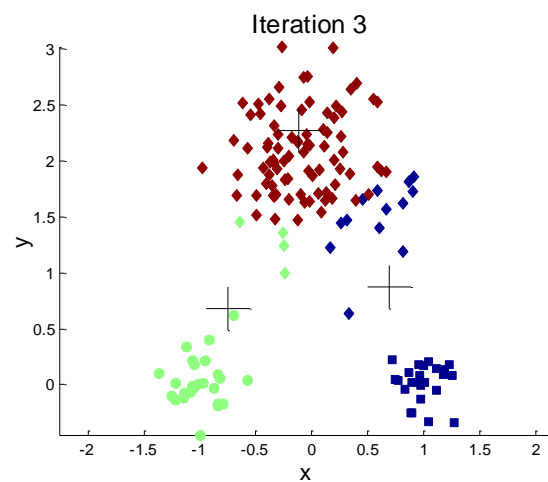
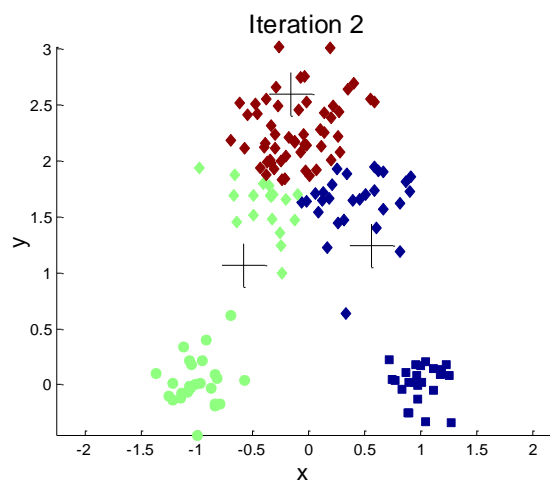
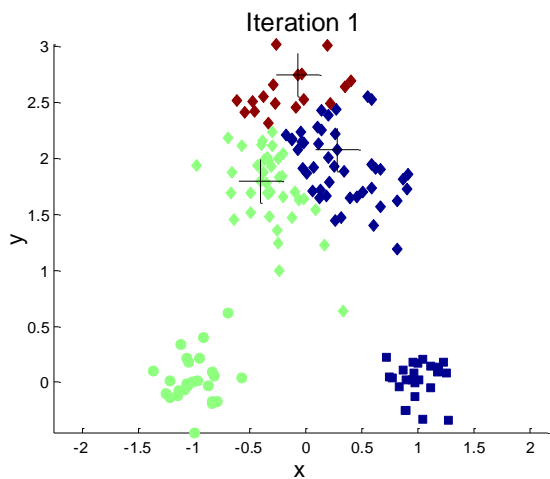
?



聚类过程中,
聚类中心位置或类簇内样本发生
变化。

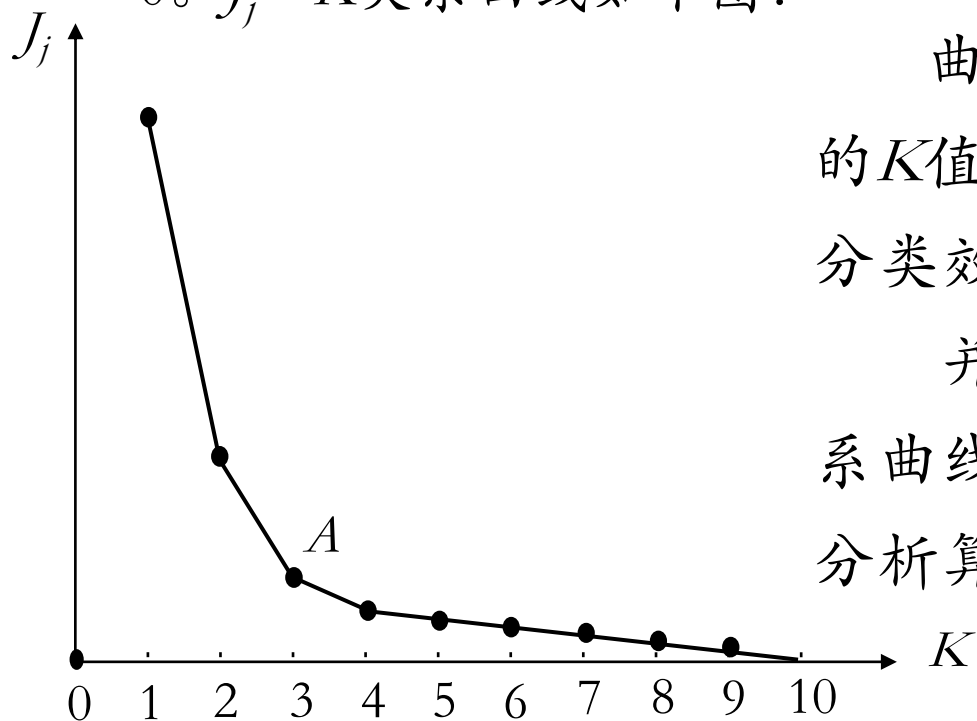
一般停止条件: 各类样本的均值都不再发生变化

例子



K-Means

- 上述K-均值算法，其类型数目假定已知为 K 个。当 K 未知时，可以令 K 逐渐增加，此时 J_j 会单调减少。最初减小速度快，但当 K 增加到一定数值时，减小速度会减慢，直到 $K = \text{总样本数 } N$ 时， $J_j = 0$ 。 $J_j - K$ 关系曲线如下图：



曲线的拐点 A 对应着接近最优的 K 值（ J 值减小量、计算量以及分类效果的**权衡**）。

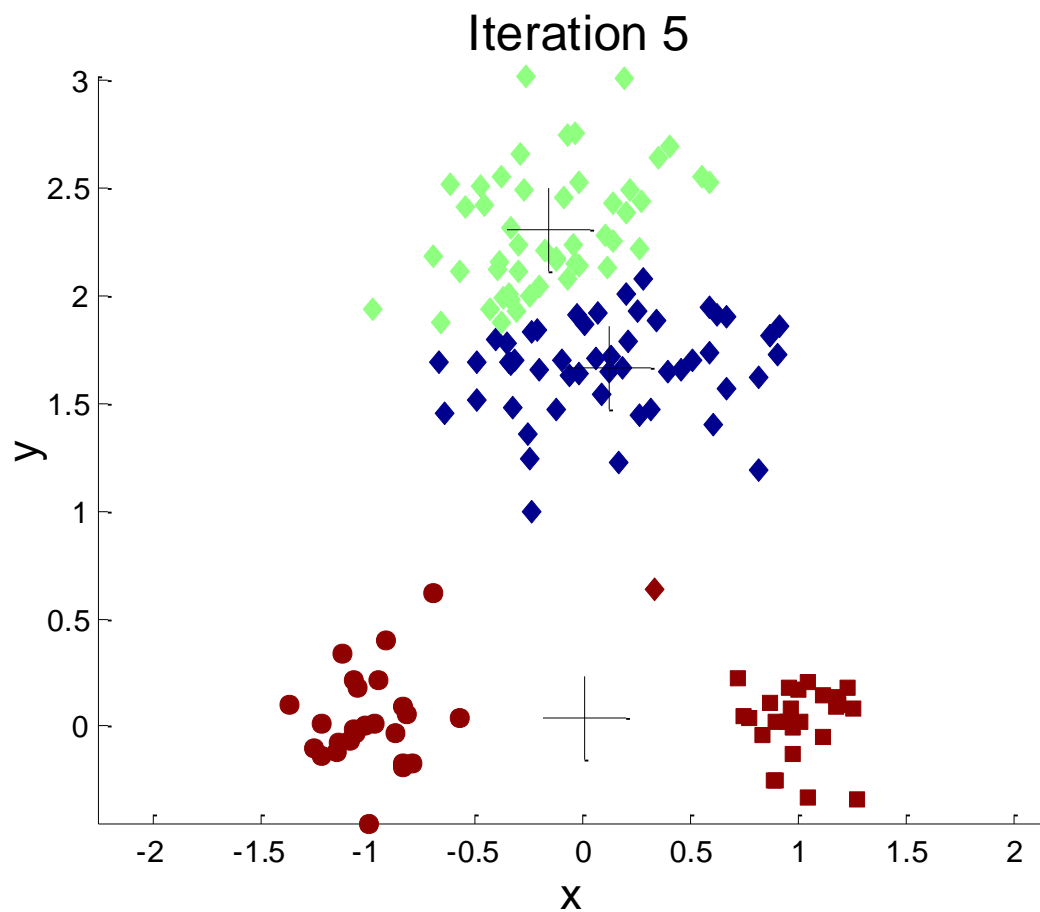
并非所有的情况都容易找到关系曲线的拐点。迭代自组织的数据分析算法可以确定模式类的个数 K 。

K-Means

- K-均值算法的结果受如下选择的影响：
 - 所选聚类的数目
 - 聚类中心的初始分布
 - 模式样本的几何性质
- 在实际应用中，需要试探不同的K值和选择不同的聚类中心的起始值。
- 如果模式样本可以形成若干个相距较远的孤立的区域分布，一般都能得到较好的收敛效果。
- K-均值算法比较适合于分类数目已知的情况。

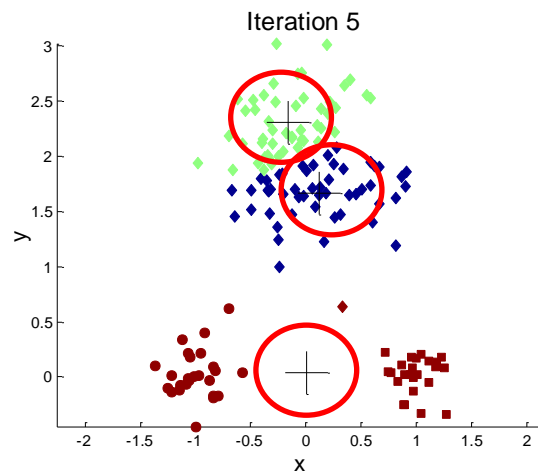
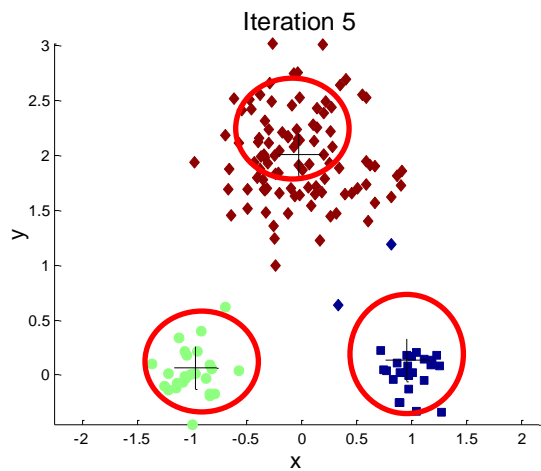
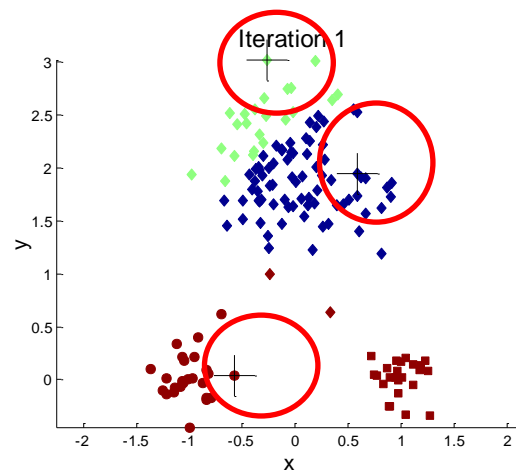
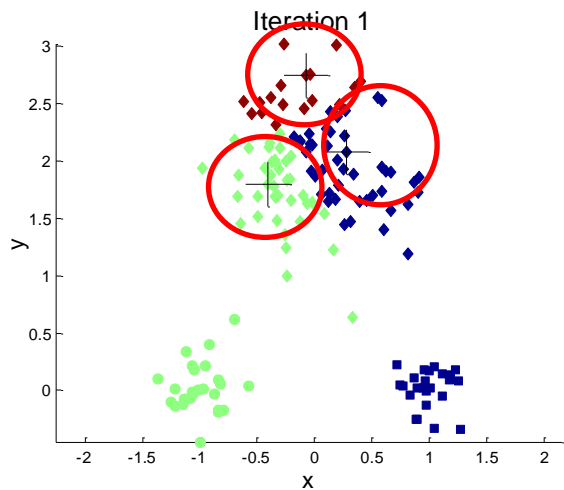
例子

- 不同的初始中心

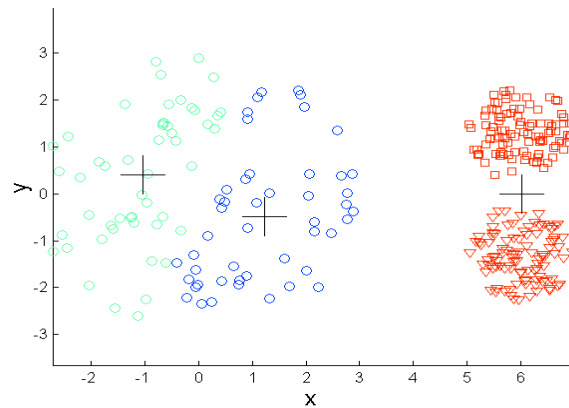
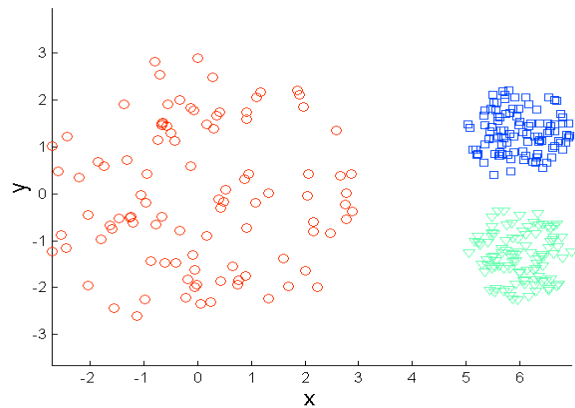


例子

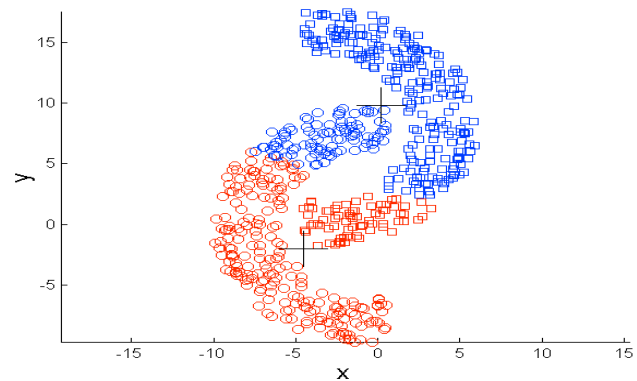
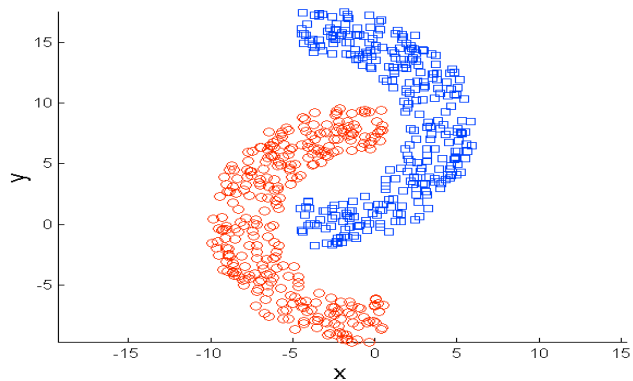
- 不同的初始中心



例子

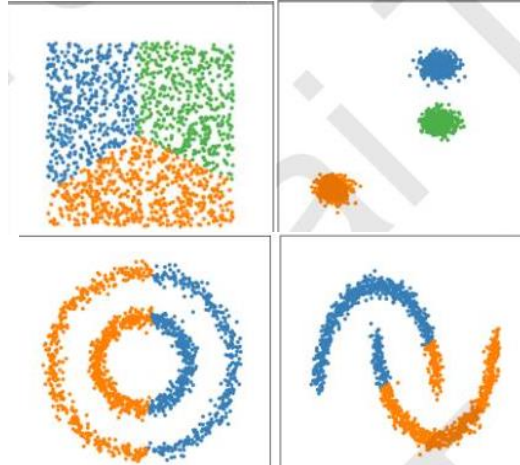


变化的密度



非球形

K-Means



优点

- 1) 算法简单、快速。
- 2) 能处理大数据集，因为它的复杂度大约是 $O(nKT)$ 。
- 3) 当簇是密集的、球状或团状的，而簇与簇之间区别明显时，它的聚类效果较好。

缺点

- 1) 要求用户必须事先给出要生成的簇的数目 k 。
- 2) 对初值敏感，对于不同的初始值，可能会导致不同的聚类结果。
- 3) 不适合于发现非凸面形状的簇，或者大小差别很大的簇。
- 4) 对于“噪声”和孤立点数据敏感

K-Means算法总结

- 总体特点
 - 基于划分的聚类方法
 - 类别数K事先指定，在实际应用中最优的K值未知，尝试用不同的K值聚类，检验得到聚类结果的质量，推测最优的K值。
 - 以样本和其所属类的中心之间的距离的总和为最优化的目标函数。
 - 算法是迭代算法，不能保证得到全局最优。
 - 得到的类别是平坦的、非层次化的。
 - 一般地，类别数变小时，平均直径会增加，类别数变大超过某个值以后，平均直径会不变，而这个值正是最优的k值。实验时，可以采用二分查找，快速找到最优的k值。

Thank You!

