# 模式识别

主讲：崔林艳&邹征夏

单位：宇航学院

开课专业：飞行器控制与信息工程

# 第六章 非监督分类器

CONTENTS PAGE

# 6.5 基于密度的聚类

6.5.1 DBSCAN算法

6.5.2 DBSCAN算法相关概念

6.5.3 DBSCAN算法基本实现流程

6.5.4 DBSCAN算法算例分析

6.5.5 DBSCAN算法小结

# 6.5 基于密度的聚类

## 6.5.1 DBSCAN算法

- DBSCAN(**D**ensity-**B**ased **S**patial **C**lustering of **A**pplications with **N**oise)是比较有代表性的基于密度的聚类算法。

- DBSAN与划分（K-均值，ISODATA算法）和层次聚类算法（分级聚类）不同，它将簇（类）定义为密度相连的点的最大集合，能够把具有足够高密度的区域划分为簇，并可在有噪声的空间数据集中发现任意形状的聚类。

- 该算法不需要确定聚类的数量，而是基于数据推测聚类的数目，能够针对任意形状产生聚类。

# 6.5 基于密度的聚类

## 6.5.2 DBSCAN算法相关概念

- Eps邻域

  与点的距离小于等于 Eps 的所有点的集合。

- 密度值

  以每个数据点为圆心，以Eps为半径画个圈（称为邻域eps-neighborhood），然后计算有多少个点在这个圈内，点的数量就是该点密度。

- 核心点（或称为核心对象，core object）：

  如果一个对象在其 Eps 邻域内含有超过 MinPts 数目的点，则该对象为核心点。 MinPts 为定义核心点时的阈值。

# 6.5 基于密度的聚类

## 6.5.2 DBSCAN算法相关概念

- **边界点：**

  如果一个对象在其 Eps 邻域含有点的数量小于 MinPts，但是该对象落在核心点的邻域内，则该对象为边界点。

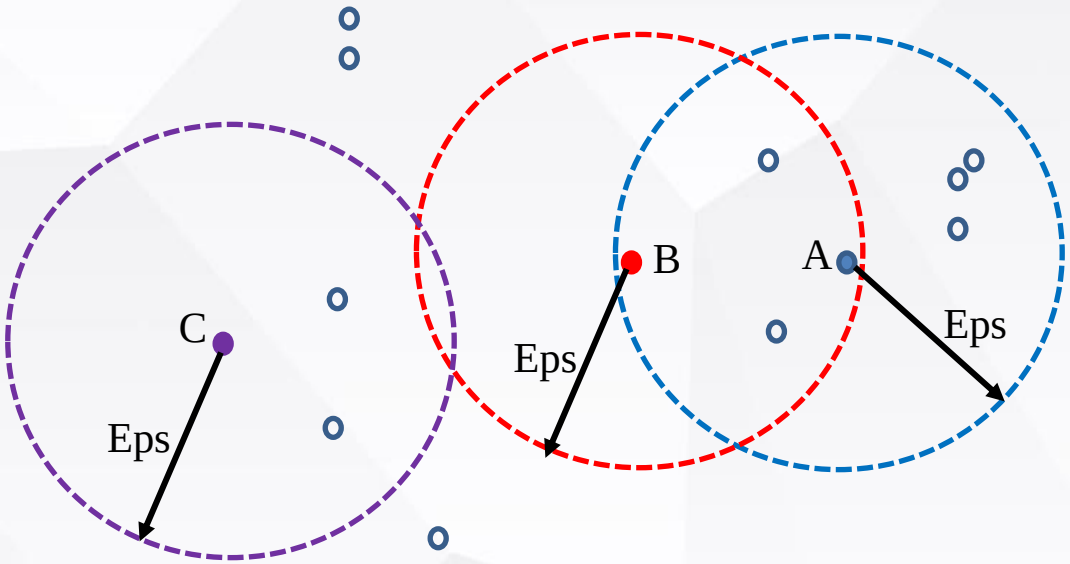
- **噪声点：**

  既不是核心点也不是边界点的点。

通俗地讲，核心点对应稠密区域内部的点，边界点对应稠密区域边缘的点，而噪声点对应稀疏区域中的点。

# 6.5 基于密度的聚类

## 6.5.2 DBSCAN算法相关概念




假设 MinPts=5，Eps 如图中箭头线所示。

点A：在其Eps邻域内含有7个点，核心点；

点B：在其Eps邻域内含有4个点，少于MinPts。落在了点A的Eps邻域内，因此为边界点；

点C：在其Eps邻域内含有3个点，少于MinPts。没有落在任何核心点邻域内，因此为噪声点。

# 6.5 基于密度的聚类

## 6.5.2 DBSCAN算法相关概念

- **密度直达 (directly density-reachable)：**

  假设有样本点$p$，$q$。其中$p$是核心点，$q$在$p$的Eps邻域内，则$p$密度直达$q$。

- **密度可达(density-reachable):**

  对于对象链：$p_1,p_2\ldots\ldots p_n$，且$p_{(i+1)}$ 由$p_i$密度直达，则$p_n$由$p_i$密度可达。

- **密度相连（density-connected）：**

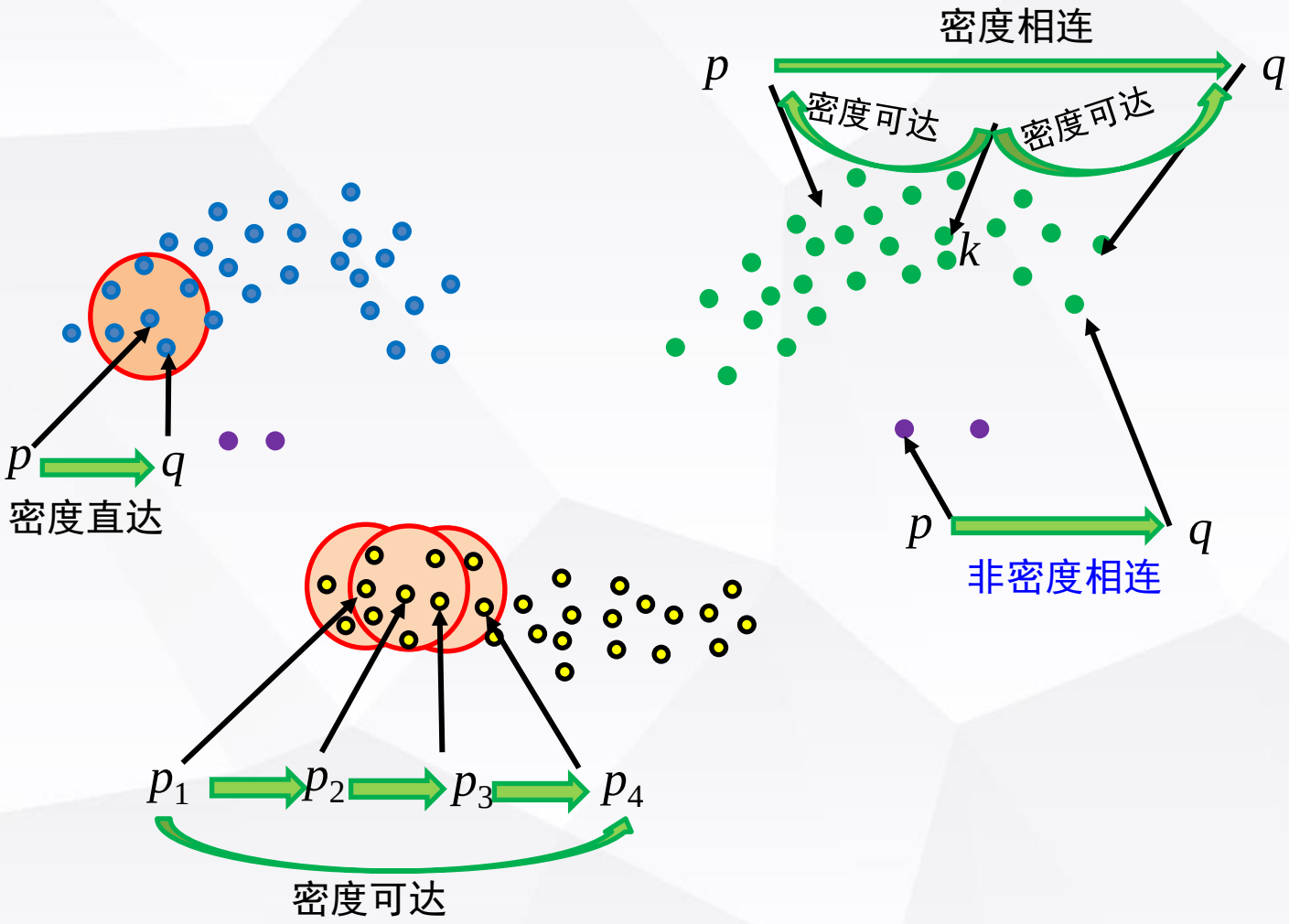
  对于样本点$p$、$q$,若存在样本点$k$使得$p$、$q$均由$k$密度可达，则称$p$，$q$密度相连。

- **密度聚类簇：**

  由核心点和与其密度可达的所有对象构成一个密度聚类簇。

# 6.5 基于密度的聚类

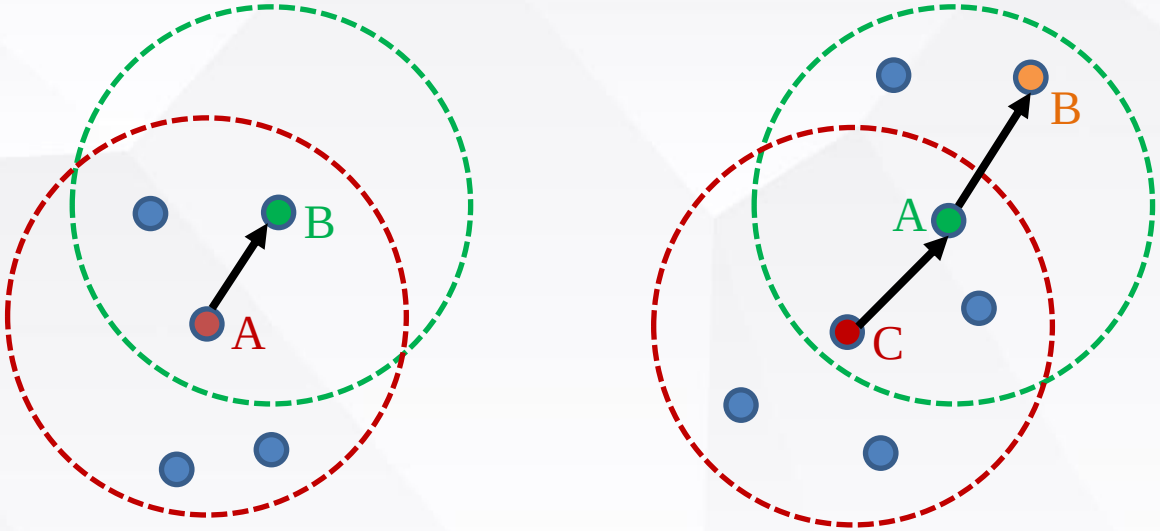
## 6.5.2 DBSCAN算法相关概念



# 6.5 基于密度的聚类

## 6.5.2 DBSCAN算法相关概念

假设 MinPts=5




(a) (b)

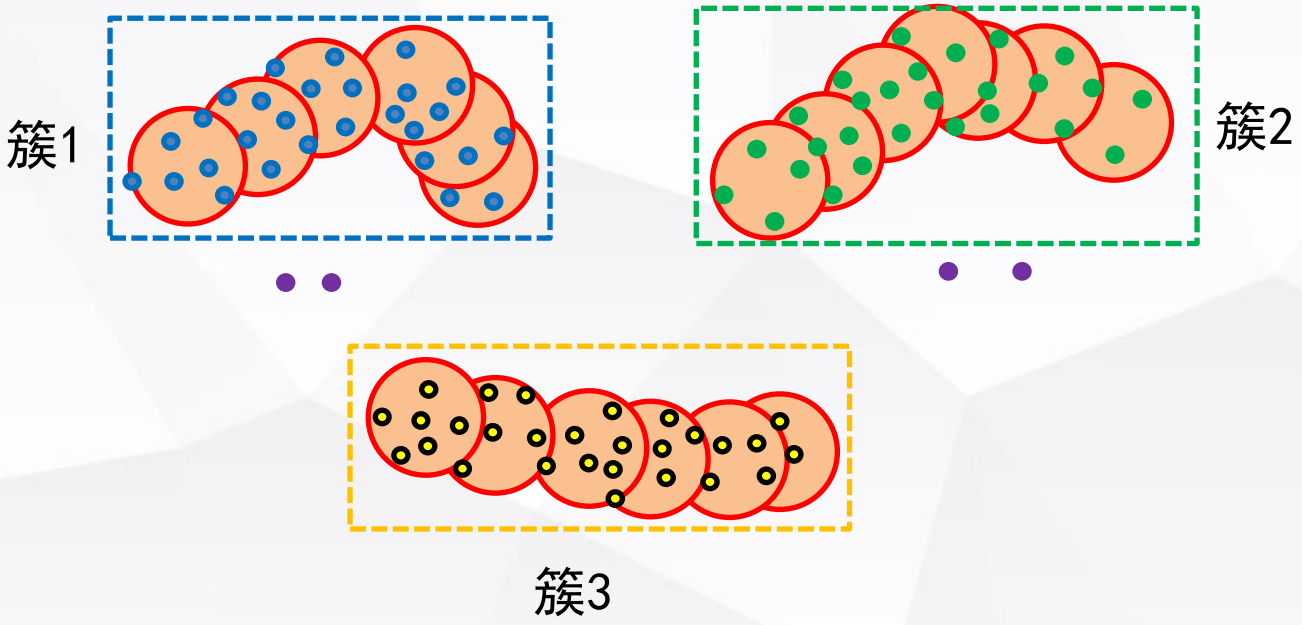
- 如图(a)所示，点A为核心点，点B为边界点，A密度直达B。
- 如图(b)所示，点C和A为核心点。C密度直达A，A密度直达B，所以C密度可达B。

# 6.5 基于密度的聚类

## 6.5.2 DBSCAN算法相关概念

- 密度聚类簇：

由密度可达关系导出的最大密度相连的样本集合，即为最终聚类的一个簇。（由核心点和与其密度可达的所有对象构成一个密度聚类簇）



# 6.5 基于密度的聚类

## 6.5.3 DBSCAN算法基本实现流程

输入：数据集Ω，邻域半径Eps，邻域中数据对象数量阈值 MinPts;
输出：密度联通簇。

- 算法基本流程：

① 找寻数据集中的核心点，形成集合 Ω1；

② while Ω1 ！ = null:

a) 从Ω1中任意选取核心点作为种子点，找出由该核心点密度可达的所有样本点，形成一个簇；

b) 将在该簇中的核心点从Ω1中清除；

c) 再从更新后的 Ω1中随机选取一个核心点作为种子点，找出由该核心点密度可达的所有样本点，生成下一个簇；

d) 重复以上步骤，直至Ω1为空。

# 6.5 基于密度的聚类

## 6.5.4 DBSCAN算法算例分析

- DSCAN算法样本数据集

$$\boldsymbol{x}_1=\begin{bmatrix}1\\2\end{bmatrix}\quad \boldsymbol{x}_2=\begin{bmatrix}2\\1\end{bmatrix}\quad \boldsymbol{x}_3=\begin{bmatrix}2\\4\end{bmatrix}\quad \boldsymbol{x}_4=\begin{bmatrix}4\\3\end{bmatrix}$$

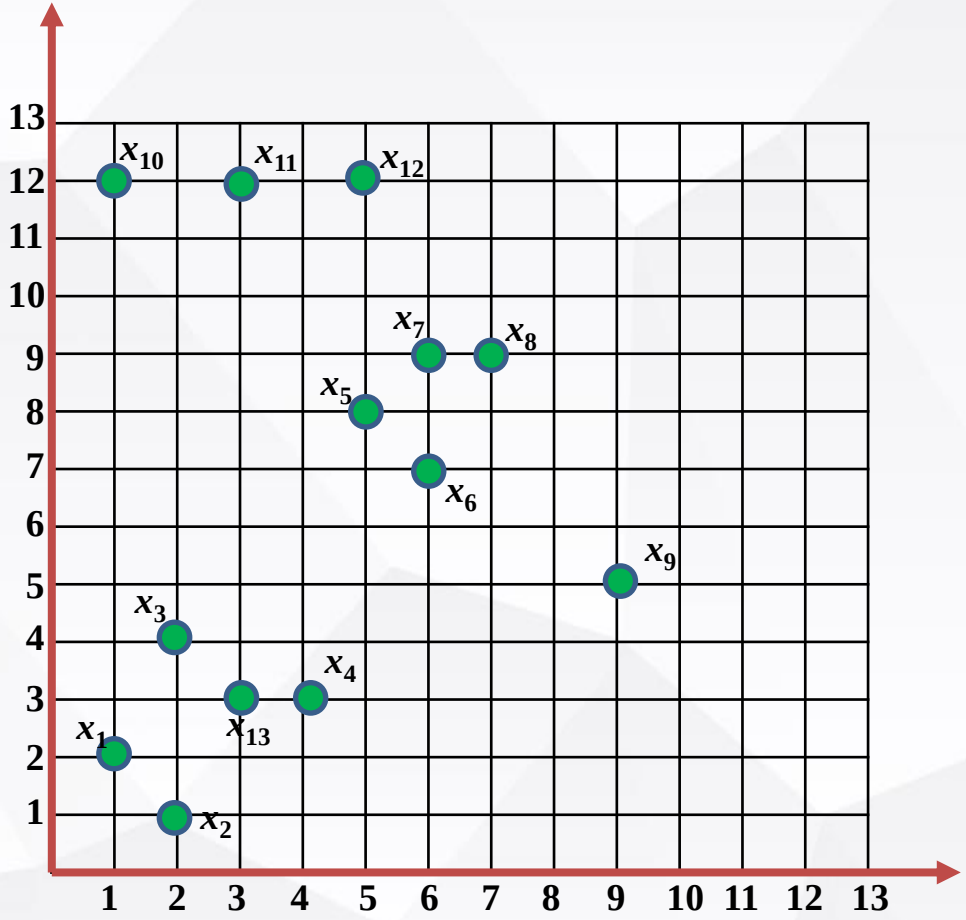
$$\boldsymbol{x}_5=\begin{bmatrix}5\\8\end{bmatrix}\quad \boldsymbol{x}_6=\begin{bmatrix}6\\7\end{bmatrix}\quad \boldsymbol{x}_7=\begin{bmatrix}6\\9\end{bmatrix}\quad \boldsymbol{x}_8=\begin{bmatrix}7\\9\end{bmatrix}$$

$$\boldsymbol{x}_9=\begin{bmatrix}9\\5\end{bmatrix}\quad \boldsymbol{x}_{10}=\begin{bmatrix}1\\12\end{bmatrix}\quad \boldsymbol{x}_{11}=\begin{bmatrix}3\\12\end{bmatrix}\quad \boldsymbol{x}_{12}=\begin{bmatrix}5\\12\end{bmatrix}\quad \boldsymbol{x}_{13}=\begin{bmatrix}3\\3\end{bmatrix}$$

问题：利用DBSCAN进行样本聚类，取 Eps=3，MinPts=3。

# 6.5 基于密度的聚类

## 6.5.4 DBSCAN算法算例分析



# 6.5 基于密度的聚类

## 6.5.4 DBSCAN算法算例分析

➢ 核心点计算：

$x_1$的Eps邻域为 $\{x_1, x_2, x_3, x_{13}\}$

$x_2$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$

$x_3$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$

$x_4$ 的Eps邻域为 $\{x_3, x_4, x_{13}\}$

$x_{13}$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$

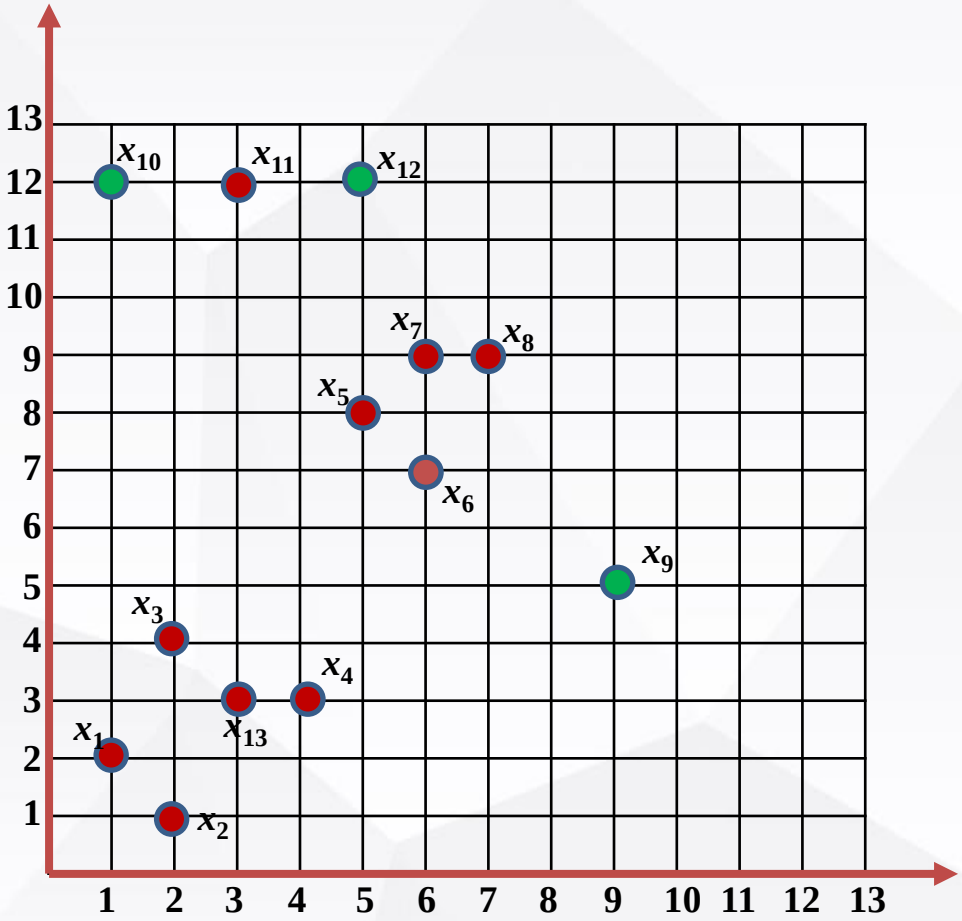
$x_5$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_6$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_7$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_8$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_{11}$的 Eps 邻域为 $\{x_{11}, x_{10}, x_{12}\}$




建立核心点集合 $\Omega 1 = \{x_1, x_2, x_3, x_4, x_{13}, x_5, x_6, x_7, x_8, x_{11}\}$

# 6.5 基于密度的聚类

$\Omega 1=\{x_1, x_2, x_3, x_4, x_{13}, x_5, x_6, x_7, x_8, x_{11}\}$

**第一步，从核心点集合Ω1依顺序选取核心点， 首先取 $x_1(1,2)$。**

以 $x_1$为核心点建立簇 $C_1$，即找出所有从 $x_1$密度可达的点。

① $x_1$ Eps邻域内的点都是 $x_1$密度直达的点，所以$\{x_1, x_2, x_3, x_{13}\}$都属于$C_1$。

② 寻找$x_1$密度可达的点。

a) 核心点$x_2$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$，因为$x_1$密度直达$x_2$，$x_2$密度直达 $x_4$，所以 $x_1$密度可达 $x_4$。因此 $x_4$也属于 $C_1$。

b) 核心点$x_3$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$，Eps邻域的点均已在 $C_1$ 中

c) 核心点$x_{13}$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$，Eps邻域的点均已在 $C_1$ 中。

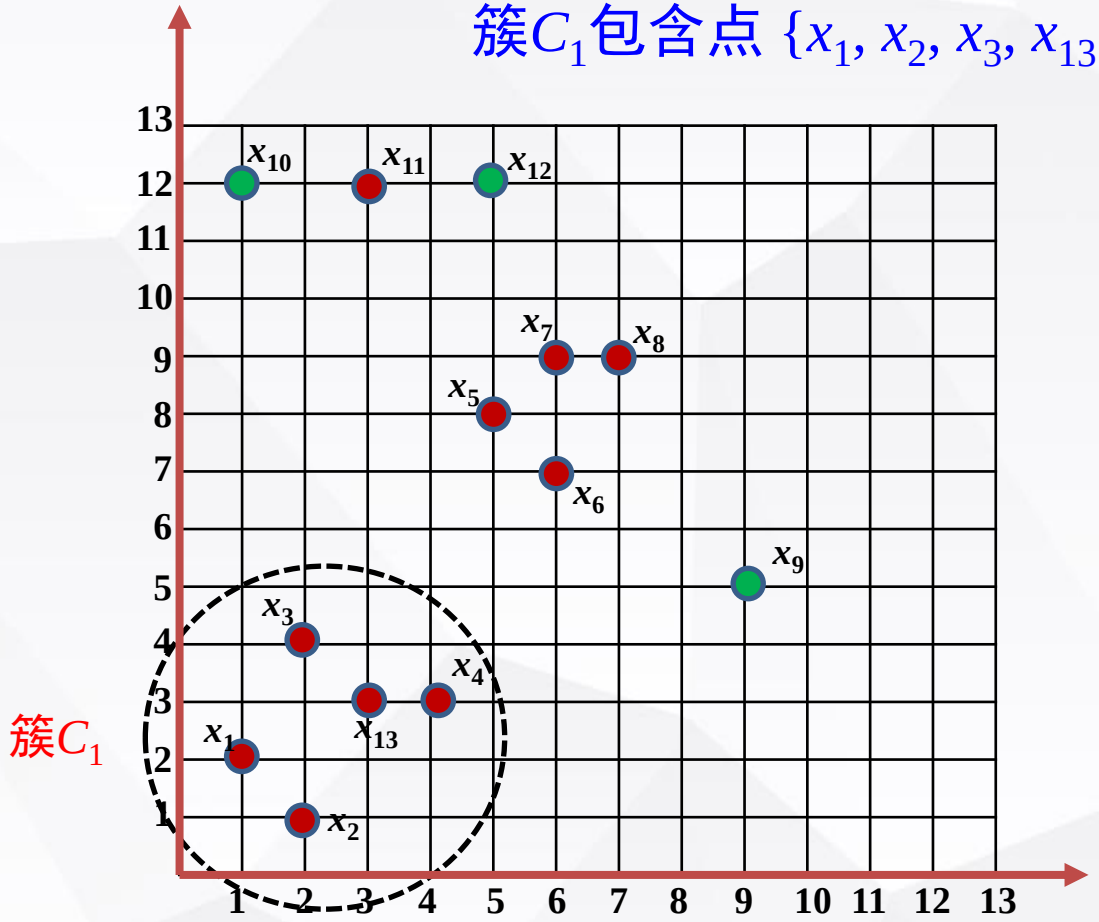
d) 核心点$x_4$ 的Eps邻域为 $\{x_3, x_4, x_{13}\}$，其Eps邻域内所有点均已被处理。

③ 此时，以$x_1$为核心点出发的那些密度可达的对象全部处理完毕，得到簇$C_1$，包含点 $\{x_1, x_2, x_3, x_{13}, x_4\}$。

将在簇$C_1$中的核心点从核心点集合Ω1中清除，得到新的$\Omega 1=\{x_5, x_6, x_7, x_8, x_{11}\}$

# 6.5 基于密度的聚类

簇$C_1$包含点 $\{x_1, x_2, x_3, x_{13}, x_4\}$




将在簇$C_1$中的核心点从核心点集合Ω1中清除，得到新的Ω1= $\{x_5, x_6, x_7, x_8, x_{11}\}$

# 6.5 基于密度的聚类

**第二步，从新核心点集合Ω1= {$x_5$, $x_6$ , $x_7$ , $x_8$, $x_{11}$} 中继续遍历核心点，取到$x_5$(5,8)。**

① 核心点$x_5$的Eps邻域为{$x_5$ , $x_6$ , $x_7$ , $x_8$ }。

② $x_5$ Eps邻域内的点都是 $x_5$密度直达的点，所以{$x_5$, $x_6$, $x_7$, $x_8$}都属于$C_2$。

③ 寻找$x_5$密度可达的点。

a) 核心点 $x_6$的Eps邻域为 {$x_5$ , $x_6$ , $x_7$ , $x_8$ } ， Eps邻域的点均已在 $C_2$ 中。

b) 核心点 $x_7$的Eps邻域为 {$x_5$ , $x_6$ , $x_7$ , $x_8$ } ， Eps邻域的点均已在 $C_2$ 中。

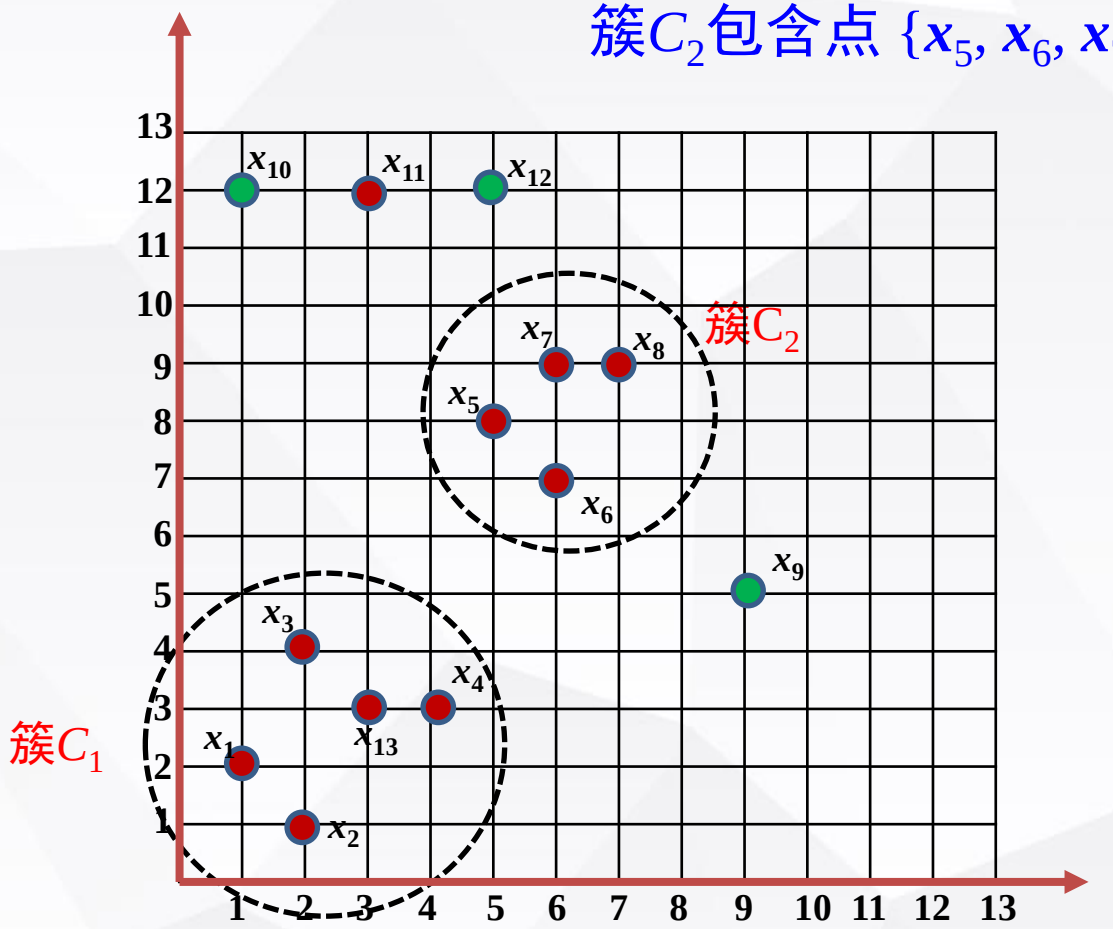
c) 核心点 $x_8$的Eps邻域为 {$x_5$ , $x_6$ , $x_7$ , $x_8$ } ， Eps邻域的点均已在 $C_2$ 中。

④ 此时，以$x_5$为核心点出发的那些密度可达的对象全部处理完毕，得到簇$C_2$，包含点 {$x_5$, $x_6$, $x_7$, $x_8$}。

将在簇$C_2$中的核心点从核心点集合Ω1中清除，得到新的Ω1= {$x_{11}$}

# 6.5 基于密度的聚类

簇$C_2$包含点 $\{x_5, x_6, x_7, x_8\}$




将在簇$C_2$中的核心点从核心点集合Ω1中清除，得到新的Ω1= $\{x_{11}\}$

# 6.5 基于密度的聚类

**第三步，从新的核心点集合Ω1= $\{x_{11}\}$ 中继续遍历核心点，仅有$x_{11}$(3,12) 。**

① 核心点$x_{11}$的Eps邻域为$\{x_{11}, x_{10}, x_{12}\}$ 。

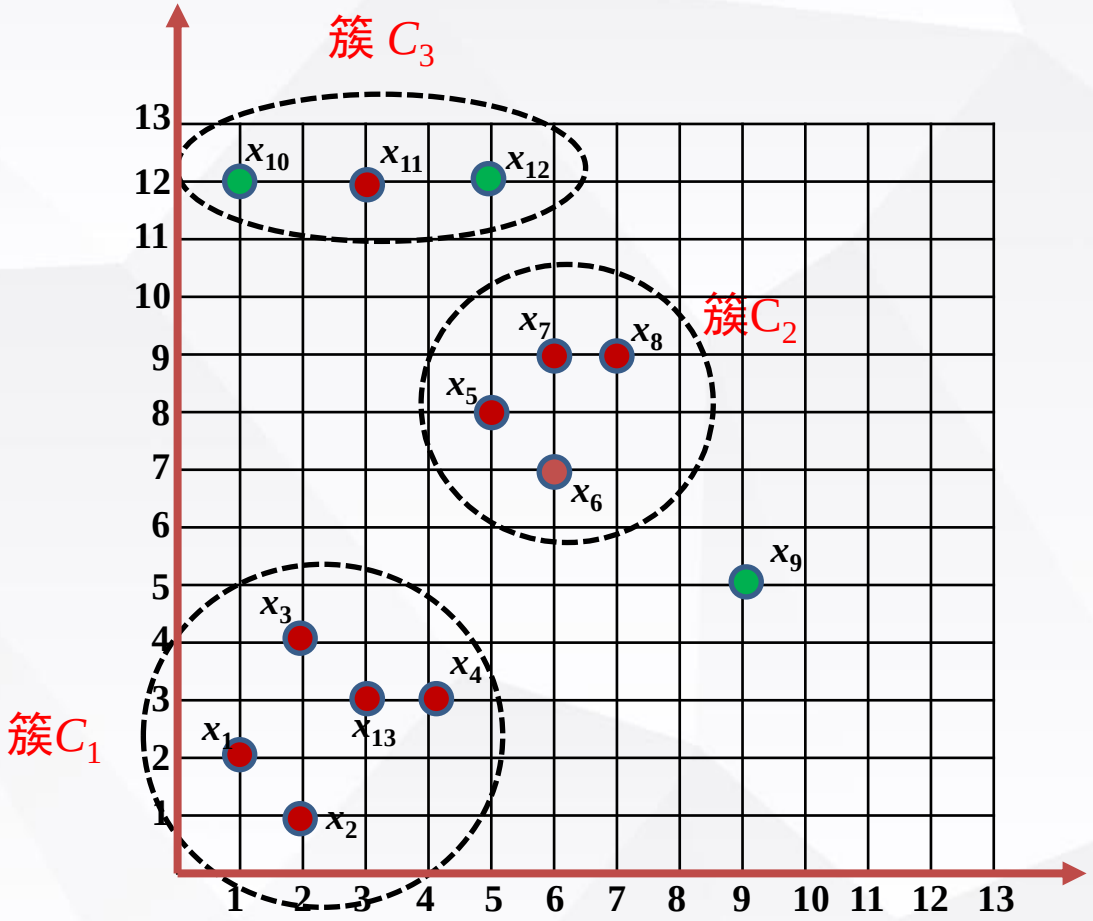
② $x_{11}$ Eps邻域内的点都是 $x_{11}$密度直达的点，所以$\{x_{11}, x_{10}, x_{12}\}$都属于$C_3$。

③ 寻找$x_{11}$密度可达的点。没有更多的核心点，所有无密度可达的点。

④ 此时，以$x_{11}$为核心点出发的那些密度可达的对象全部处理完毕，得到簇$C_3$，包含点 $\{x_{11}, x_{10}, x_{12}\}$ 。

将在簇$C_3$中的核心点从核心点集合Ω1中清除，得到新的Ω1= null，至此遍历结束，得到最终聚类结果。

# 6.5 基于密度的聚类




最终，样本聚为3类。

簇$C_1$包含点 $\{x_1, x_2, x_3, x_{13}, x_4\}$

簇$C_2$包含点 $\{x_5, x_6, x_7, x_8\}$

簇$C_3$包含点 $\{x_{11}, x_{10}, x_{12}\}$

$x_9$为异常点

# 6.5 基于密度的聚类

- 核心点：

$x_1$的Eps邻域为 $\{x_1, x_2, x_3, x_{13}\}$

$x_2$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$

$x_3$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$

$x_4$ 的Eps邻域为 $\{x_3, x_4, x_{13}\}$

$x_{13}$的Eps邻域为 $\{x_1, x_2, x_3, x_4, x_{13}\}$

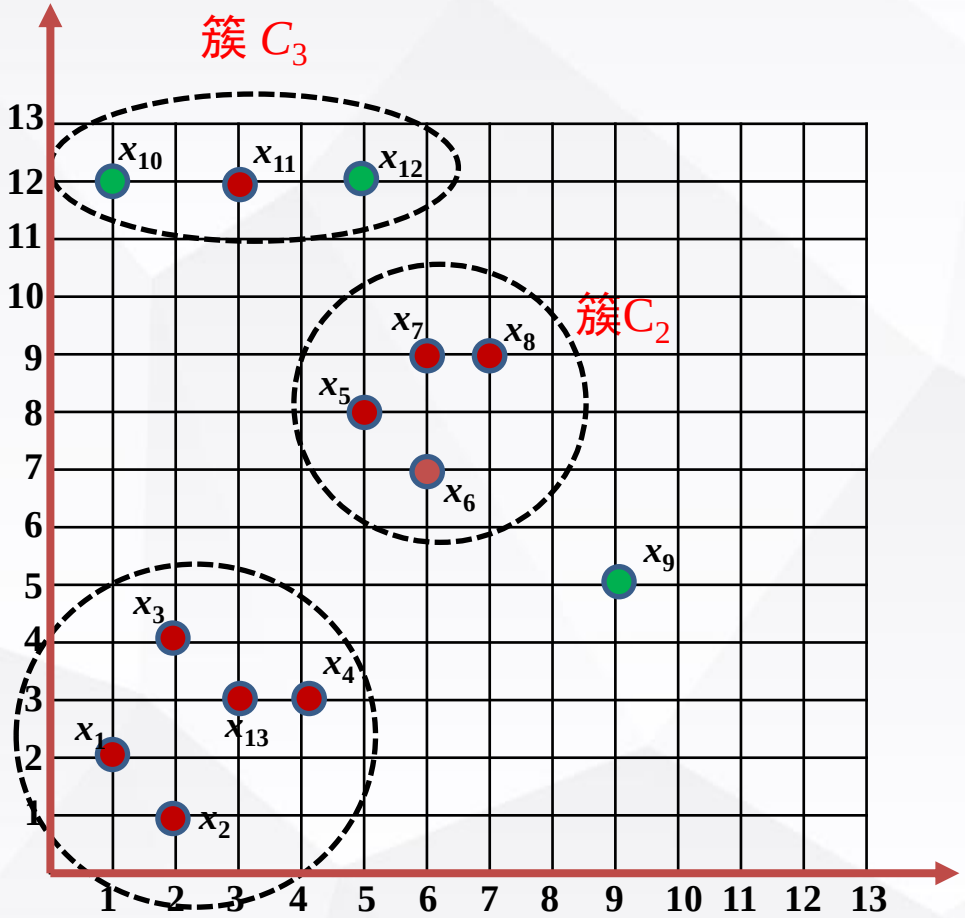
$x_5$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_6$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_7$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_8$的Eps邻域为$\{x_5, x_6, x_7, x_8\}$

$x_{11}$的 Eps 邻域为 $\{x_{11}, x_{10}, x_{12}\}$



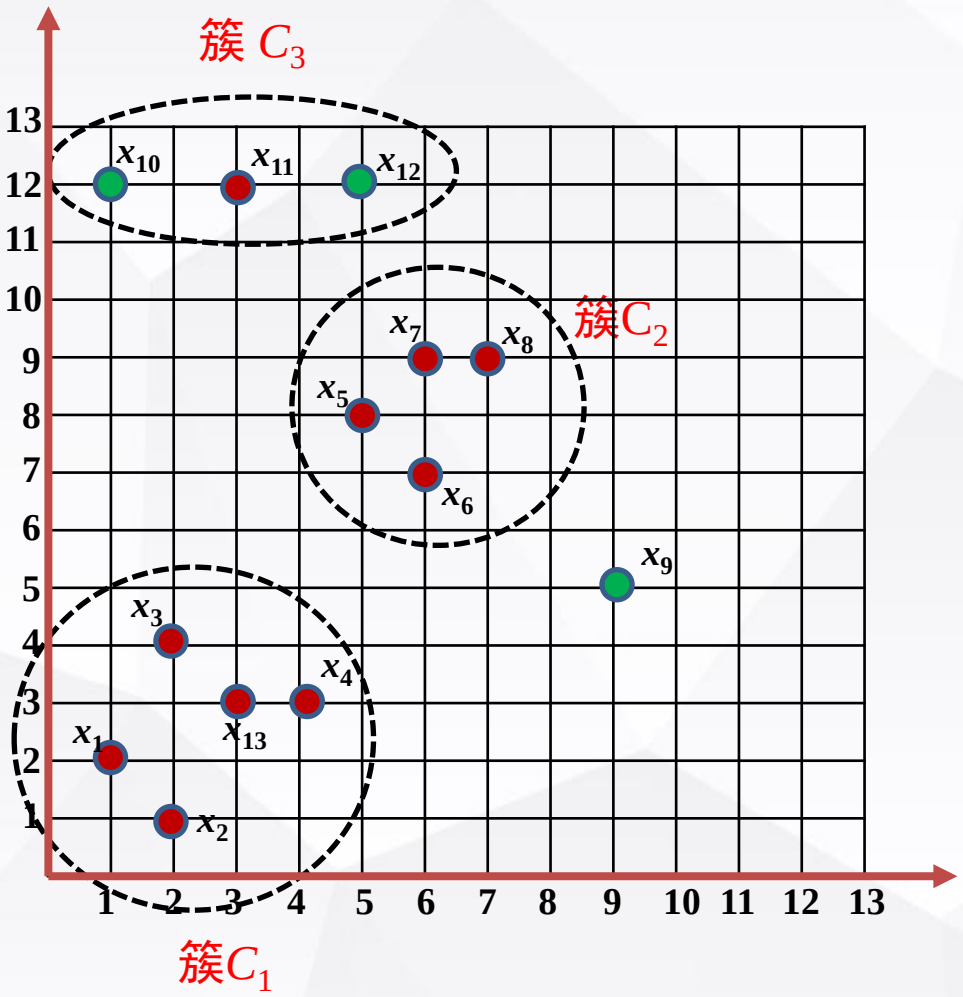
# 6.5 基于密度的聚类

➢ 由聚类结果可以看出：

① 每个簇里面可以有一个或者多个核心点。

② 如果只有一个核心点，则簇里其他的非核心点样本都在这个核心点的Eps邻域里。

③ 如果有多个核心点，则簇里的任意一个核心点的Eps邻域中一定有一个其他的核心点。这些核心点的Eps邻域里所有的样本的集合组成一个 DBSCAN 聚类簇。



# 6.5 基于密度的聚类

## 6.5.5 DBSCAN算法小结

- 优点：

① 不用人为设定簇的个数；

② 可以在聚类的同时发现异常点，对数据集中的异常点不敏感。

③ 能发现任意形状的空间聚类，如能识别“非球形”的簇，而K-均值这种基于距离的聚类方法只能找出球状的簇；

④ 聚类结果几乎不依赖于遍历的顺序，而K-均值聚类算法的初始值对聚类结果有很大影响。

# 第六章 非监督分类器

CONTENTS PAGE

# 6.6 模糊聚类

6.6.1 问题引入

6.6.2 模糊理论

6.6.3 FCM算法

# 6.6 模糊聚类

## 6.6.1 问题引入

- K-均值算法

一种硬聚类算法，算法的依据是类内误差平方和最小化准则，给集合中的每个样本赋予一个明确的类别。但在实际应用中，有时样本在形态和类属方面存在中介性，不存在严格的属性划分。

- 模糊C均值算法（Fuzzy C-Means，FCM）

为了更好的解决这类问题，将模糊理论引入K-均值算法（也称C-均值算法），从硬聚类推广到模糊聚类，即FCM算法。

# 6.6 模糊聚类

## 6.6.2 模糊理论

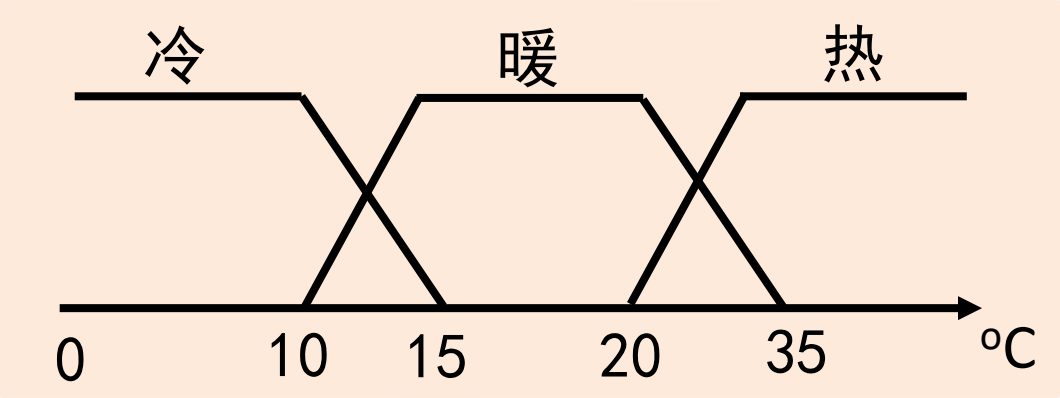
模糊控制是自动化控制领域的一项经典方法。其原理则是模糊数学、模糊逻辑。1965年美国加州大学柏克莱分校的扎德教授发表模糊集合“Fuzzy Sets”的论文，首次引入隶属度函数的概念，打破了经典数学“非0即1”的局限性，用[0,1]之间的实数来描述中间状态。

很多经典的集合（即：论域U内的某个元素是否属于集合A，可以用一个数值来表示。在经典集合中，要么0，要么1）不能描述很多事物的属性，需要用模糊性词语来判断。比如天气冷热程度、人的胖瘦程度等等。模糊数学和模糊逻辑把只取1或0二值（属于/不属于）的普通集合概念推广到0~1区间内的多个取值，即隶属度。用“隶属度”来描述元素和集合之间的关系。

# 6.6 模糊聚类

## 6.6.2 模糊理论

- 举例




如图所示，对于冷热程度，采取三个模糊子集：冷、暖、热。对于某一个温度，可能同时属于两个子集。要进一步具体判断，就需要提供一个描述“程度”的函数，即隶属度。

## 6.6.2 模糊理论

- 传统集合论思维方式

a） 一个元素或者属于某个集合，或者不属于该集合；
b） 二值逻辑

- 模糊集合论思维方式

a） 一个元素以一定的程度属于某个集合，也可以同时以不同的程度属于多个集合；
b） 模糊集合

## 6. 6. 2 模糊理论

### ➢ 隶属度函数

表示一个对象$x$隶属于集合$A$的程度，记作 $\mu_A(x)$，取值范围是[0,1]

a) $\mu_A(x)=1$ ，表示$x$完全属于$A$， 相当于$x\in A$

b) $\mu_A(x)=0$ ，表示$x$完全不属于$A$，相当于$x\notin A$

### ➢ 模糊集

对于有限个对象$\{x_1, x_2, \ldots, x_n\}$，模糊集合$A$为：

$$A=\{(\mu_A(x_i), x_i)\}$$

## 6.6.2 模糊理论

- 常见的模糊集运算：

并：模糊集$A$和$B$的并集$C=A\cup B$的隶属度函数定义为：

$$\mu_C(x)=\max\{\mu_A(x),\mu_B(x)\}$$

交：模糊集$A$和$B$的交集$C=A\cap B$的隶属度函数定义为：

$$\mu_C(x)=\min\{\mu_A(x),\mu_B(x)\}$$

有时也把交集的隶属度函数定义为

$$\mu_C(x)=\mu_A(x)\bullet\mu_B(x)$$

补：模糊集$A$和补集$C=A'$的隶属度函数定义为

$$\mu_C(x)=1-\mu_A(x)$$

# 6.6 模糊聚类

K-均值： $J_e = \sum_{i=1}^{C} \sum_{x \in \omega_i} \| x - m_i \|_2^2$

## 6.6.3 FCM算法

- FCM聚类融合了模糊理论的精髓。相较于K-均值的硬聚类，FCM提供了更加灵活的聚类结果。通过为每个对象和每个簇赋予一个权值，指明对象属于该簇的程度。

- FCM算法原理：

$\{x_i,\ i=1,2,\ldots,N\}$ 是有 $N$ 个样本的集合，$C$ 为预定的类别数目；$m_j(j=1,2,\ldots,C)$ 为每个聚类的中心，$\mu_j(x_i)$ 是第 $i$ 个样本对于第 $j$ 类的隶属度函数。则基于隶属度函数定义的聚类损失函数为：

$$J_f = \sum_{j=1}^{C} \sum_{i=1}^{N} \left[\mu_j(x_i)\right]^b \left\| x_i - m_j \right\|_2^2$$

其中，$b > 1$ 为加权指数，又称为平滑因子，控制模式在模糊类间的分享程度，多数情况取为2。

## 6.6.3 FCM算法

➢ FCM算法原理：

$$J_f = \sum_{j=1}^{C}\sum_{i=1}^{N}\left[\mu_j(\boldsymbol{x}_i)\right]^b \left\|\boldsymbol{x}_i - \boldsymbol{m}_j\right\|_2^2$$

$$\sum_{j=1}^{C}\mu_j(\boldsymbol{x}_i) = 1, \quad \forall i = 1,2,\ldots,N$$

利用拉格朗日乘数法把条件极值转化为无条件极值问题。

a) 引入$N$个拉格朗日因子：

$$\min J'_f = \sum_{j=1}^{C}\sum_{i=1}^{N}\left[\mu_j(\boldsymbol{x}_i)\right]^b \left\|\boldsymbol{x}_i - \boldsymbol{m}_j\right\|_2^2 + \sum_{i=1}^{N}\lambda_i\left(\sum_{j=1}^{C}\left[\mu_j(\boldsymbol{x}_i)\right] - 1\right)$$

b) 对各个变量($\boldsymbol{m}_j$和$\mu_j(\boldsymbol{x}_i)$)求导，得到各个变量的极值点。

## 6.6.3 FCM算法

➢ FCM算法原理：

$$\mu_j(\boldsymbol{x}_i)=\frac{\left(1\Big/\left\|\boldsymbol{x}_i-\boldsymbol{m}_j\right\|_2^2\right)^{1/(b-1)}}{\sum_{k=1}^{C}\left(1\Big/\left\|\boldsymbol{x}_i-\boldsymbol{m}_k\right\|_2^2\right)^{1/(b-1)}},i=1,2,\ldots,N;\ j=1,2,\ldots,C$$

$$\boldsymbol{m}_j=\frac{\sum_{i=1}^{N}\left[\mu_j(\boldsymbol{x}_i)\right]^b\boldsymbol{x}_i}{\sum_{i=1}^{N}\left[\mu_j(\boldsymbol{x}_i)\right]^b},\ j=1,2,\ldots,C$$

采用迭代方法求解如上两公式，直至满足收敛条件，得到最优解。

## 6.6.3 FCM算法

➢ FCM算法原理：

c) 迭代的终止条件:

$$\max_{ij}\left\{\left|\mu_j\left(x_i\right)^{(k+1)}-\mu_j\left(x_i\right)^{(k)}\right|\right\}<\delta$$

其中：$k$为迭代步数，$\delta$ 为误差阈值

继续迭代下去，隶属程度也不会发生较大的变化，达到比较优（局部最优或全局最优）状态。

# 6.6 模糊聚类

## 6.6.3 FCM算法

$$\mu_j(x_i)=\frac{\left(1/\left\|x_i-\mathbf{m}_j\right\|_2^2\right)^{1/(b-1)}}{\sum_{k=1}^{C}\left(1/\left\|x_i-\mathbf{m}_k\right\|_2^2\right)^{1/(b-1)}}$$

$$\mathbf{m}_j=\frac{\sum_{i=1}^{N}\left[\mu_j(x_i)\right]^b x_i}{\sum_{i=1}^{N}\left[\mu_j(x_i)\right]^b}$$

➢ 算法步骤：

**Step1**：设定聚类数目$C$和参数$b$；

**Step2**：初始化各个聚类中心 $\mathbf{m}_i$ ；

**Step3**：重复下面的运算，直到各个样本的隶属度值稳定：

① 用当前的聚类中心计算隶属度函数；

② 用当前的隶属度函数更新各类聚类中心。

当算法收敛时，即可根据各类的聚类中心和各个样本对于各类的隶属度值完成模糊聚类划分。

# 本章小结

6.1 引言

6.2 聚类原理

6.3 分级聚类

6.4 动态聚类

6.5 基于密度的聚类

6.6 模糊聚类