第二章　聚类分析

2.1　聚类分析的概念

2.2　模式相似性测度

2.3　类的定义与类间距离

2.4　准则函数

2.5　聚类的算法

2.5　聚类的算法

2.5.1 聚类的技术方案

聚类分析有很多具体的算法,有的比较简单,有的相对复杂和完善,但归纳起来就是三大类:

1、按最小距离原则简单聚类方法

2、按最小距离原则进行两类合并的方法

3、依据准则函数动态聚类方法

2.5 聚类的算法

(1) 简单聚类方法

针对具体问题确定相似性阈值，将模式到各聚类中心间的距离与阈值比较，当大于阈值时该模式就作为另一类的类心，小于阈值时按最小距离原则将其分划到某一类中。

这类算法运行中模式的类别及类的中心一旦确定将不会改变。

(2) 按最小距离原则进行两类合并的方法

首先视各模式自成一类, 然后将距离最小的两类合并成一类, 不断地重复这个过程，直到成为两类为止。

这类算法运行中，类心不断地修正，但模式类别一旦指定后就不再改变，就是模式一旦划为一类后就不再被分划开，这类算法也称为谱系聚类法。

2.5 聚类的算法

(3) 依据准则函数动态聚类法

设定一些分类的控制参数，定义一个能表征聚类结果优劣的准则函数，聚类过程就是使准则函数取极值的优化过程。

算法运行中，类心不断地修正，各模式的类别的指定也不断地更改。这类方法有—C均值法、ISODATA法等。

2.5 聚类的算法

2.5.2 根据相似性阈值的简单聚类方法

1） 根据相似性阈值和最小距离原则的简单聚类方法

1. 条件及约定

设待分类的模式为 $\{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_N\}$ ，选定类内距离门限 T 。

2. 算法思想

计算模式特征矢量到聚类中心的距离并和门限 T 比较，决定归属该类或作为新的一类中心。这种算法通常选择欧氏距离。

2.5.2 根据相似性阈值的简单聚类方法

1） 根据相似性阈值和最小距离原则的简单聚类方法

3. 算法原理步骤

(1) 取任意的一个模式特征矢量作为第一个聚类中心。例如，令 ω_1 类的中心 $\vec{z}_1 = \vec{x}_1$

(2) 计算下一个模式特征矢量 $\vec{x}_2$ 到 $\vec{z}_1$ 的距离 d_{21}。若 d_{21} d_{21} 则建立新的一类 ω_2，其类心 $\vec{z}_2 = \vec{x}_2$。若 $d_{21} \le T$，则 $\vec{x}_2 \in \omega_1$。

2.5.2 根据相似性阈值的简单聚类方法

1） 根据相似性阈值和最小距离原则的简单聚类方法

3. 算法原理步骤

(3) 假设已有聚类中心 $\vec{z}_1, \vec{z}_2, \cdots, \vec{z}_k$，计算尚未确定类别的模式特征矢量 $\vec{x}_i$ 到各聚类中心 $\vec{z}_j$ $(j = 1,2,\cdots,k)$ 的距离 d_{ij}。如果 $d_{ij} > T$ $(j = 1,2,\cdots,k)$ ，则 $\vec{x}_i$ 作为新的一类 ω_{k+1} 的中心，$\vec{z}_{k+1} = \vec{x}_i$ ；

否则，如果 $d_{il} = \min_j[d_{ij}]$ ，则指判 $\vec{x}_i \in \omega_l$ 。检查是否所有的模式都分划完类别，如果都分划完了则结束；否则返到(3)。

2.5 聚类的算法

1） 根据相似性阈值和最小距离原则的简单聚类方法

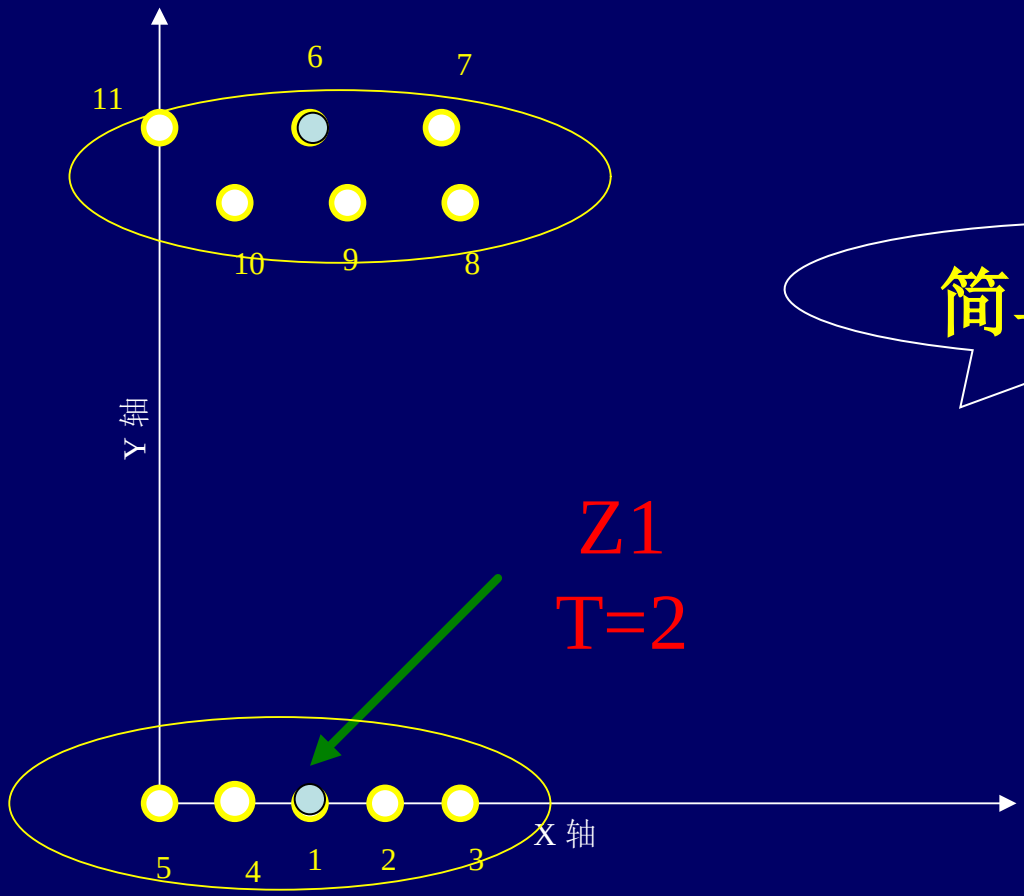
算法特点：

这类算法的突出优点是算法简单。但聚类过程中，类的中心一旦确定将不会改变，模式一旦指定类后也不再改变。

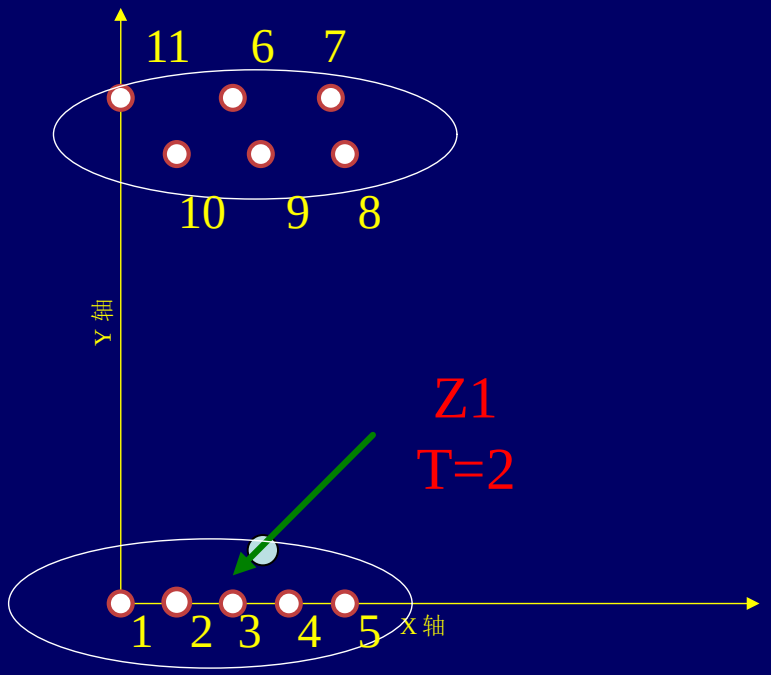
从算法的过程可以看出，该算法结果很大程度上依赖于距离门限T的选取及模式参与分类的次序。如果能有先验知识指导门限T的选取，通常可获得较合理的效果。也可考虑设置不同的T和选择不同的次序，最后选择较好的结果进行比较。

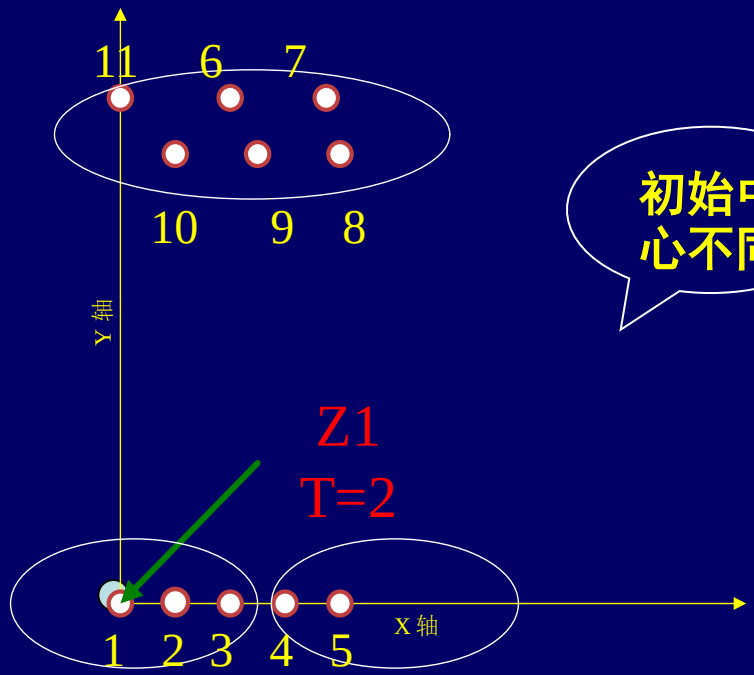
2.5 聚类的算法

1） 根据相似性阈值和最小距离原则的简单聚类方法

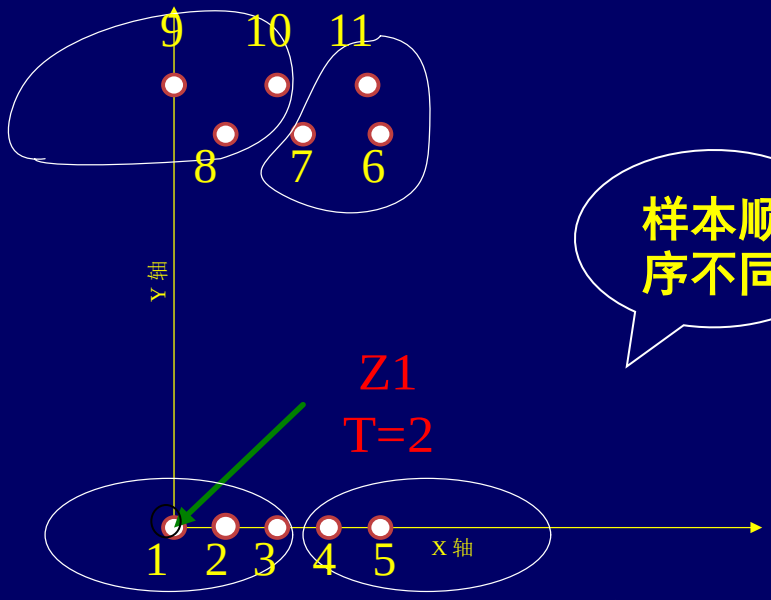


简单聚类图例

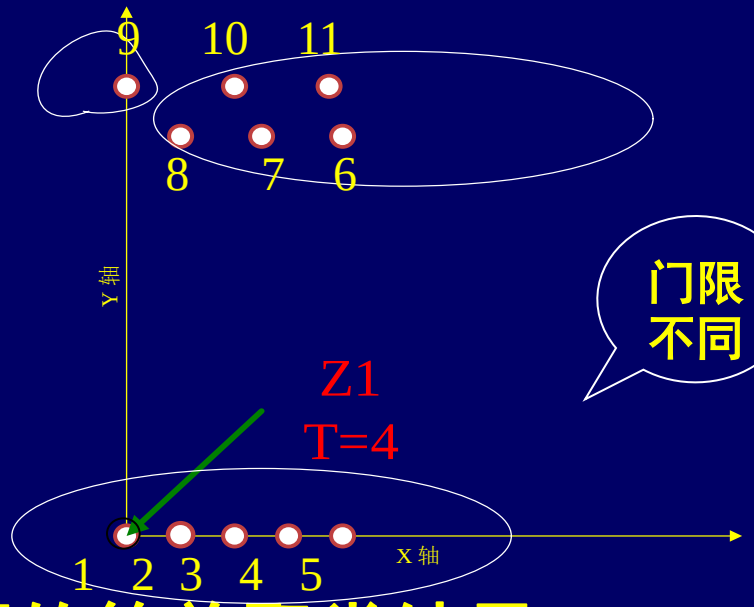




初始中心不同



样本顺序不同



门限不同

初始条件不同的简单聚类结果

2.5.2 根据相似性阈值的简单聚类方法

2） 最大最小距离算法

1. 条件及约定

设待分类的模式为 $\{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_N\}$，选定比例系数 θ 。

2. 算法思想

在模式特征矢量集中以最大距离原则选取新的聚类中心。以最小距离原则进行模式归类。这种方法通常也使用欧氏距离。

2.5.2 根据相似性阈值的简单聚类方法

2） 最大最小距离算法

3. 算法原理步骤

（1） 选任一模式特征矢量作为第一个聚类中心 $\vec{z}_1$

例如，$\vec{z}_1 = \vec{x}_1$。

（2） 从待分类矢量集中选距离 $\vec{z}_1$最远的特征矢量作为第二个聚类中心 $\vec{z}_2$。

2） 最大最小距离算法

(3) 计算未被作为聚类中心的各模式特征矢量 $\{\vec{x}_i\}$ 与 $\vec{z}_1$ 、$\vec{z}_2$ 之间的距离，并求出它们之中的最小值，即

$$d_{ij} = \left\| \vec{x}_i - \vec{z}_j \right\| \qquad (j = 1,2)$$

$$d_i = \min\left[d_{i1}, d_{i2} \right] \quad (i = 1,2,\cdots,N)$$

为表述简洁，虽然某些模式已选做聚类中心，但上面仍将所有模式下角标全部列写出来，因这并不影响算法的正确性。

2.5.2 根据相似性阈值的简单聚类方法

2） 最大最小距离算法

(4) 若

$$d_l = \max_i \left[\min(d_{i1}, d_{i2})\right] > \theta \left\| \vec{z}_1 - \vec{z}_2 \right\|$$

则相应的特征矢量 $\vec{x}_l$ 作为第三个聚类中心，$\vec{z}_3 = \vec{x}_l$

然后转至(5)；否则，转至最后一步(6)。

2.5.2 根据相似性阈值的简单聚类方法

2） 最大最小距离算法

(5) 设存在 k 个聚类中心，计算未被作为聚类中心的各特征矢量到各聚类中心的距离 d_{ij}，并算出

$$d_l = \max_i \left[\min \left[d_{i1}, d_{i2}, \cdots, d_{ik} \right] \right]$$

如果 $d_l > \theta \left\| \vec{z}_1 - \vec{z}_2 \right\|$，则 $\vec{z}_{k+1} = \vec{x}_l$ 并转至(5)；

否则，转至最后一步(6)。

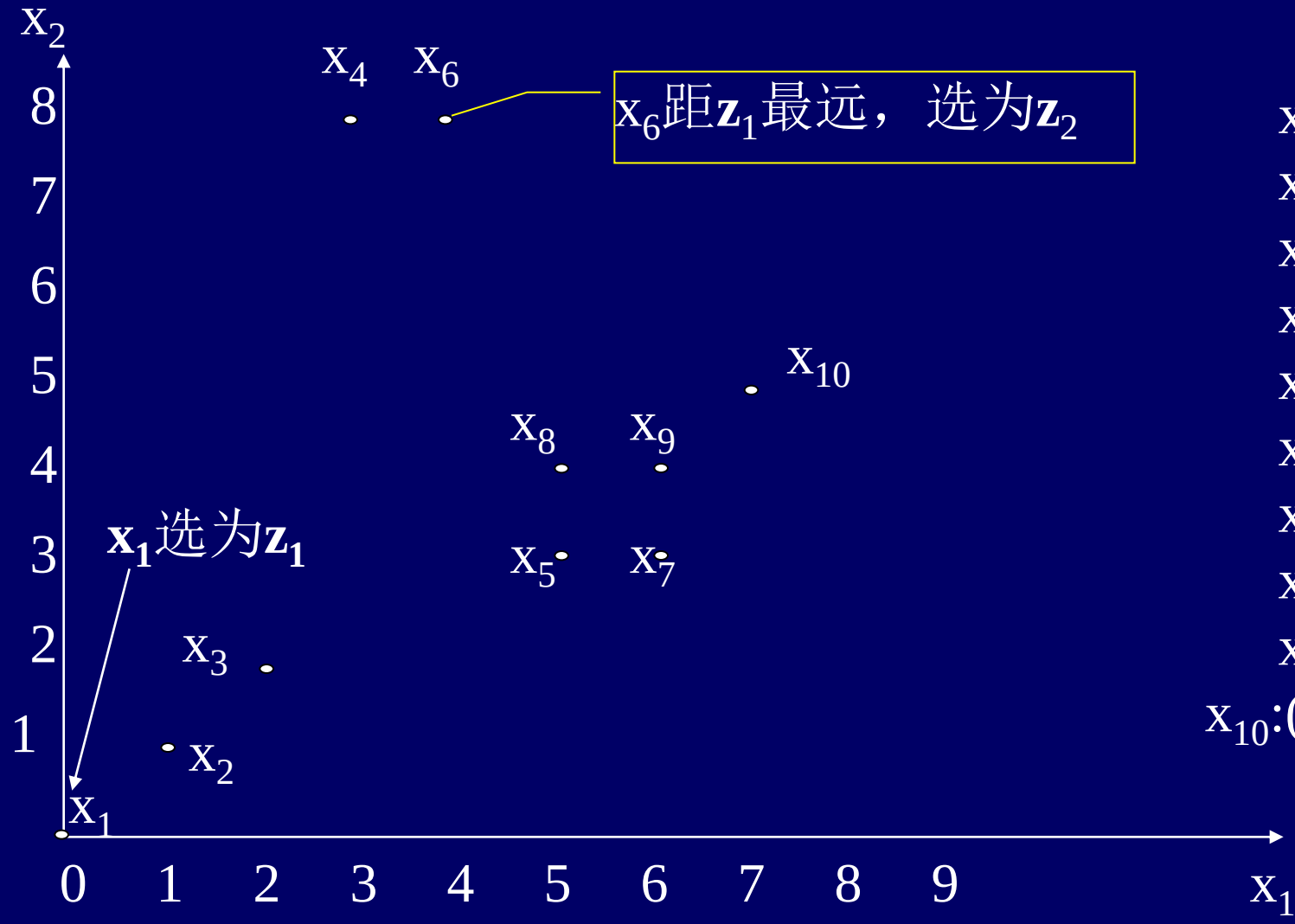
(6) 当判断出不再有新的聚类中心之后，将模式特征矢量 $\{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_N\}$ 按最小距离原则分到各类中去，即计算

$$d_{ij} = \left\| \vec{x}_i - \vec{z}_j \right\| \quad (j = 1,2,\cdots ;\ i = 1,2,\cdots, N)$$

当 $d_{il} = \min_j \left[d_{ij} \right]$，则判 $\vec{x}_i \in \omega_l$。

这种算法的聚类结果与参数 θ 以及第一个聚类中心的选取有关。如果没有先验知识指导 θ 和 $\vec{z}_1$ 的选取，可适当调整 θ 和 $\vec{z}_1$，比较多次试探分类结果，选取最合理的一种聚类。

2·5 聚类的算法



x1:(0,0)
x2:(1,1)
x3:(2,2)
x4:(3,8)
x5:(5,3)
x6:(4,8)
x7:(6,3)
x8:(5,4)
x9:(6,4)
x10:(7,5)

最大最小距离算法实例

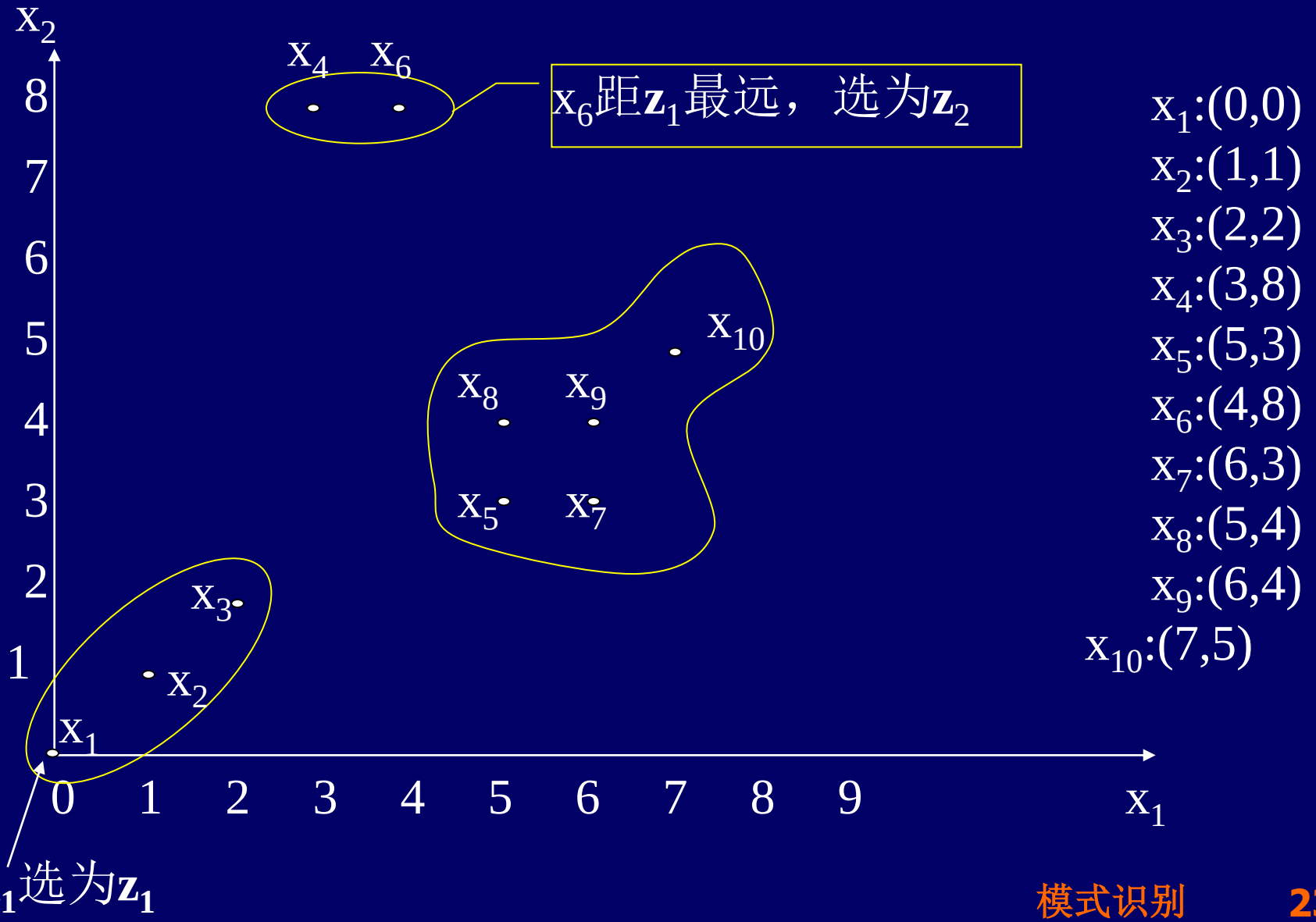
θ =0. 2	z_1	z_2	z_3	z_4
	x_1:(0,0)	x_6:(4,8)		
x_1:(0,0)	0	80		
x_2:(1,1)	1	58		
x_3:(2,2)	8	40		
x_4:(3,8)	73	1		
x_5:(5,3)	34	26		
x_6:(4,8)	80	0		
x_7:(6,3)	45	29		
x_8:(5,4)	41	26		
x_9:(6,4)	52	20		
x_{10}:(7,5)	74	18		

θ =0. 2	z_1	z_2	z_3	z_4
	x_1:(0,0)	x_6:(4,8)		
x_1:(0,0)	0	80		
x_2:(1,1)	1	58		
x_3:(2,2)	8	40		
x_4:(3,8)	73	1		
x_5:(5,3)	34	26		
x_6:(4,8)	80	0		
x_7:(6,3)	45	29		
x_8:(5,4)	41	26		
x_9:(6,4)	52	20		
x_{10}:(7,5)	74	18		

$\theta = 0.2$	z_1	z_2	z_3	z_4
	x_1:(0,0)	x_6:(4,8)	x_7:(6,3)	
x_1:(0,0)	0	80	45	
x_2:(1,1)	1	58	29	
x_3:(2,2)	8	40	17	
x_4:(3,8)	73	1	34	
x_5:(5,3)	34	26	1	
x_6:(4,8)	80	0	29	
x_7:(6,3)	45	29	0	
x_8:(5,4)	41	26	2	
x_9:(6,4)	52	20	1	
x_{10}:(7,5)	74	18	5	

θ =0. 2	z_1	z_2	z_3	z_4
	x_1:(0,0)	x_6:(4,8)	x_7:(6,3)	
x_1:(0,0)	0	80	45	
x_2:(1,1)	1	58	29	
x_3:(2,2)	8	40	17	
x_4:(3,8)	73	1	34	
x_5:(5,3)	34	26	1	
x_6:(4,8)	80	0	29	
x_7:(6,3)	45	29	0	
x_8:(5,4)	41	26	2	
x_9:(6,4)	52	20	1	
x_{10}:(7,5)	74	18	5	

2.5 聚类的算法



x_1:(0,0)
x_2:(1,1)
x_3:(2,2)
x_4:(3,8)
x_5:(5,3)
x_6:(4,8)
x_7:(6,3)
x_8:(5,4)
x_9:(6,4)
x_{10}:(7,5)

2.5.3 谱系聚类法

按最小距离原则不断进行两类合并

层次聚类法 (Hierarchical Clustering Method)(系统聚类法、 谱系聚类法）

2.5.3 谱系聚类法

1. 条件及约定

设待分类的模式特征矢量为 $\{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_N\}$，$G_i^{(k)}$ 表示第 k 次合并时的第 i 类。

2. 算法思想

首先将 N 个模式视作各自成为一类，然后计算类与类之间的距离，选择距离最小的一对合并成一个新类，计算在新的类别分划下各类之间的距离，再将距离最近的两类合并，直至所有模式聚成两类为止。

2.5.3 谱系聚类法

3. 算法原理步骤

(1) 初始分类。令 $k=0$ ，每个模式自成一类，即

$$G_i^{(0)}=\{\vec{x}_i\}\ (i=1,2,\cdots,N)$$

(2) 计算各类间的距离 D_{ij} ，由此生成一个对称的距离矩阵 $D^{(k)}=(D_{ij})_{m\times m}$ ，m 为类的个数（初始时 $m=N$）。

2. 5. 3 谱系聚类法

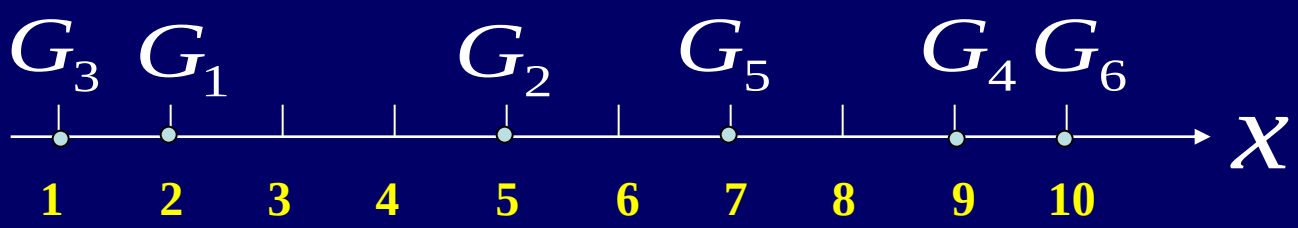
3. 算法原理步骤

(3) 找出前一步求得的矩阵 $D^{(k)}$ 中的最小元素，设它是 $G_i^{(k)}$和 $G_j^{(k)}$间的距离，将 $G_i^{(k)}$ 和 $G_j^{(k)}$ 两类合并成一类，于是产生新的聚类 $G_1^{(k+1)}, G_2^{(k+1)}, \cdots$

令 $k = k+1, m = m-1$

(4) 检查类的个数。如果类数 m 大于2，转至(2)；否则，停止。

2.5 聚类的算法



D(0)

	ω_1	ω_2	ω_3	ω_4	ω_5
ω_2	3				
ω_3	1	4			
ω_4	7	4	8		
ω_5	5	2	6	2	
ω_6	8	5	9	1	3

1、设全部样本分为6类，

2、作距离矩阵D(0)

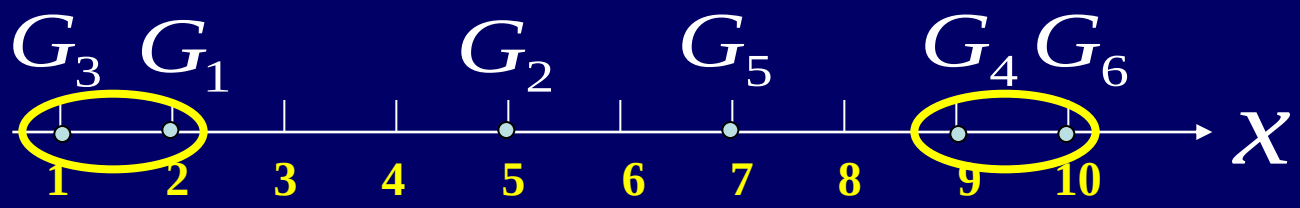
3、求最小元素：

4、把ω1，ω3合并ω7=(1，3)

ω4，ω6合并ω8=(4，6)

5、作距离矩阵D(1)

2.5 聚类的算法



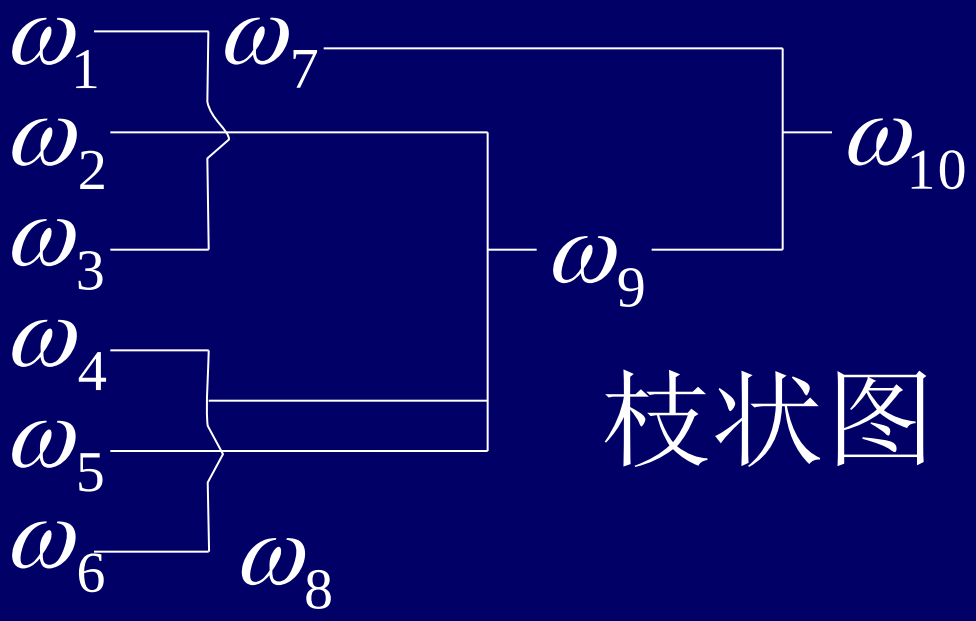
D(1)

1、求最小元素：

2、把 ω_2, ω_5, ω_8 合并

	ω_7	ω_2	ω_8
ω_2	3		
ω_8	7	4	
ω_5	5	2	2

2. 5. 3 谱系聚类法

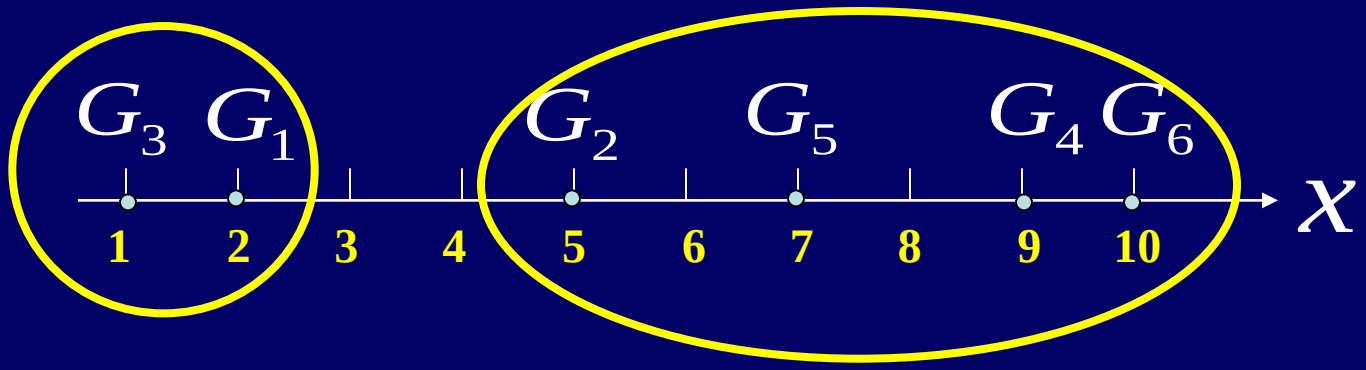


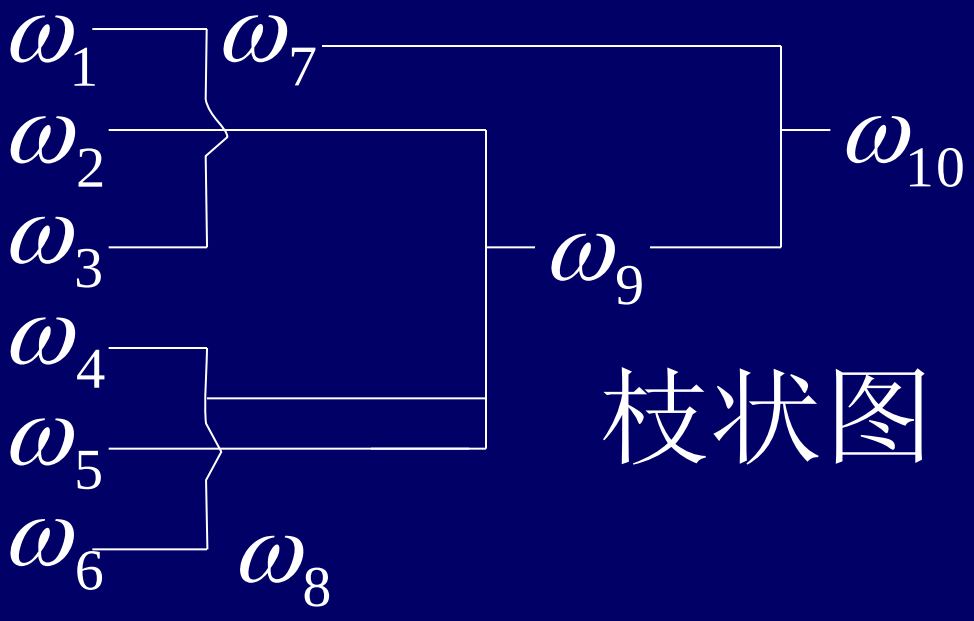
枝状图

D(1)

	ω_7	ω_2	ω_8
ω_2	3		
ω_8	7	4	
ω_5	5	2	2

2·5 聚类的算法


G_3 G_1 G_2 G_5 G_4 G_6
1 2 3 4 5 6 7 8 9 10
x

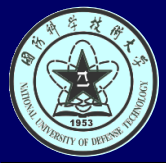

ω_1
ω_2
ω_3
ω_4
ω_5
ω_6
ω_7
ω_8
ω_9
ω_{10}

枝状图

作业

1、有样本集　{(0,0),(0,1),(4,4),(4,5),(5,4),(5,5),(1,0)}，试用普系聚类算法对其分类。

2、P64:　2.11



谢 谢！