第十章 非监督学习与聚类

苏智勇
可视计算研究组
南京理工大学
suzhiyong@njust.edu.cn
https://zhiyongsu.github.io

主要内容

10.1 引言

10.2 基于模型的聚类方法

10.3 混合模型的估计

10.4 动态聚类算法

10.5 模糊聚类方法

10.6 分级聚类方法

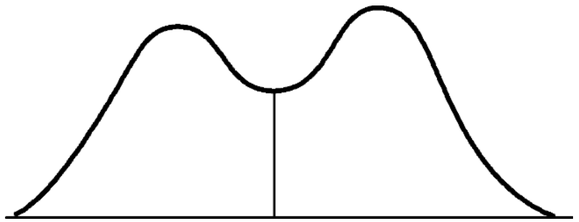
10.7 一致聚类方法

10.1 引言

- 监督模式识别
  - （已知）样本集 → 训练（学习，分类器设计）→ 识别（分类）
- 非监督模式识别
  - （未知）样本集 → 非监督学习（聚类分析） → 后处理
  - 分类
    - ✓基于模型的方法
    - ✓基于相似性度量的方法

10.2 基于模型的聚类方法：单峰子集分离法

- **基本假设**
  - 每个聚类的样本分布是单峰的，根据总体分布中的单峰来划分子集



- **投影方法**
  - **基本思路**

    把样本按照某种准则投影到某个一维坐标上，在这一维度上估计样本的概率密度，在其中寻找单峰并进行聚类划分(如果这一维上只有一个峰，则寻找下一个投影方向)

  - **投影方向**

    使**方差**最大的方向，即协方差矩阵本征值最大的本征向量方向

10.2 基于模型的聚类方法：单峰子集分离法

- **算法步骤**

(1)**主成分分析**：计算所有样本$\{\boldsymbol{x}\}$的协方差矩阵的最大本征值对应的本征向量$\boldsymbol{u}_j$，把样本投影到$\boldsymbol{u}_j$上$v_j = \boldsymbol{u}_j^T \boldsymbol{x}$

(2)用非参数法估计投影后样本$v_j = \boldsymbol{u}_j^T \boldsymbol{x}$的**概率密度函数**$P(v_j)$（用直方图方法或其它方法）

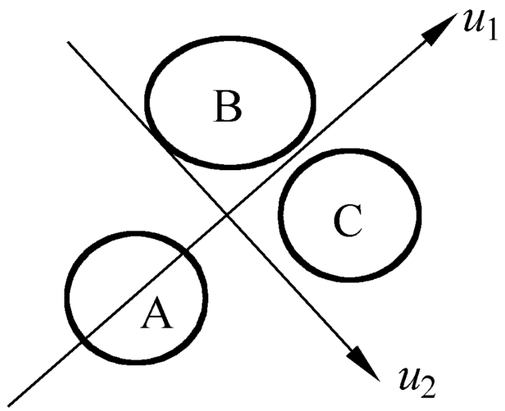
(3)求 $P(v_j)$ 中的极小点（波谷），在这些极小点上作垂直于$\boldsymbol{u}_j$的超平面作为**分类超平面**，得到子集划分

(4)如果 $P(v_j)$ 上没有这种极小点，则用下一个本征值对应的本征向量作为投影方向，重复（2）～（3）

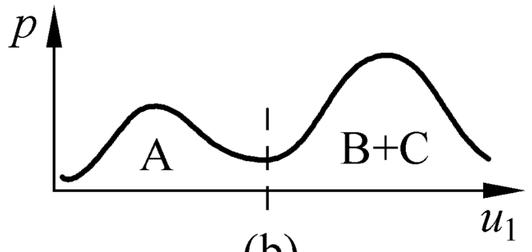
(5)对划分出的每一个子集重复上述过程，直到不能再分（所有方向上都是单峰）

10.2 基于模型的聚类方法：单峰子集分离法

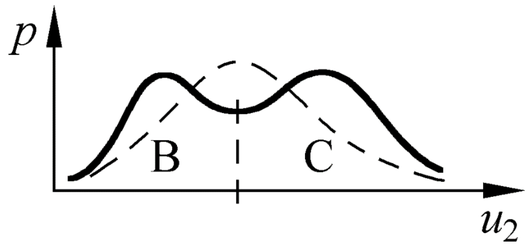
- **问题**
  - 如何选择投影方向?
    - 方差最大的准则（经验准则）有时并不一定最有利于聚类
- **有效的聚类**



(a)



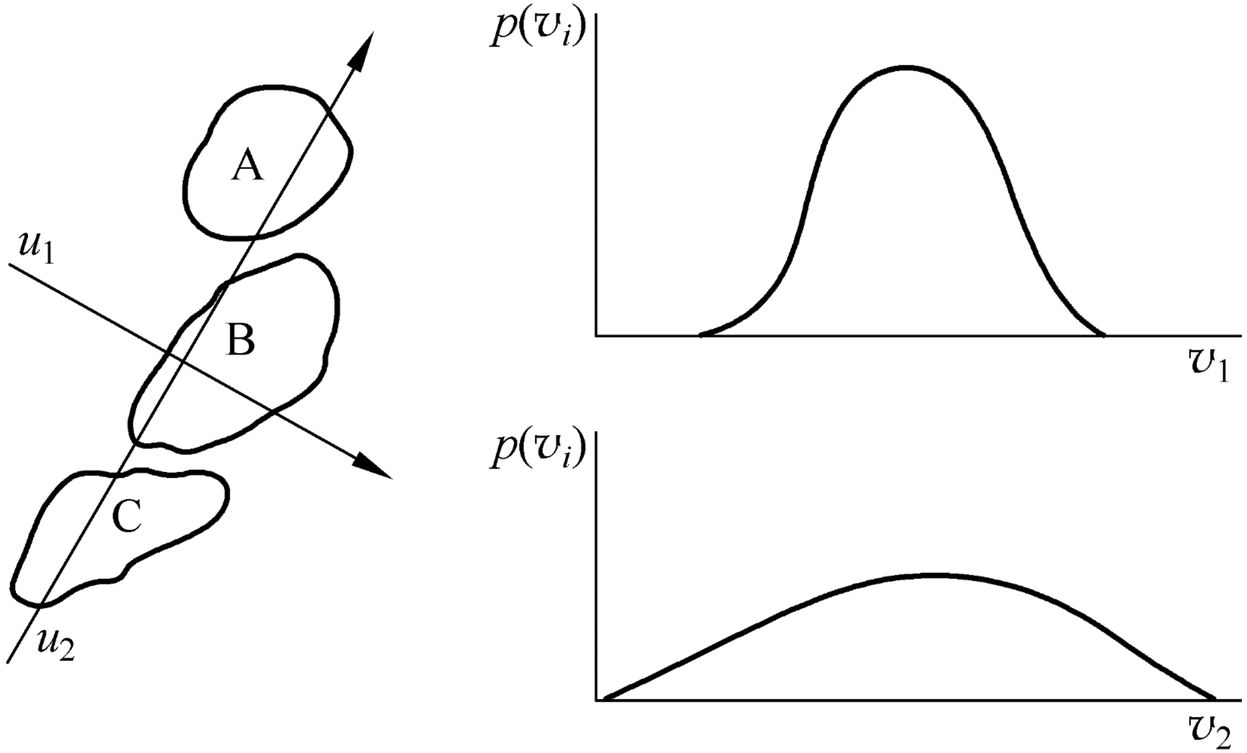
(b)



(c)

10.2 基于模型的聚类方法：单峰子集分离法

- **失败的聚类**
  - 例：在两个主成分方向上都得不到单峰子集



10.3 混合模型的估计：非监督参数估计

- **定义**
  - 非监督参数估计指样本**类别未知**，但各**类条件概率密度函数的形式已知**，根据所有样本估计各类密度函数中的**参数**。
- 本节只介绍非监督最大似然估计的思路

10.3.1 混合密度的最大似然估计

- **假设条件**
  - 样本集 $X = \{x_1, ..., x_N\}$ 中的样本属于c 个类别，但不知各样本属哪类。
  - 类先验概率 $P(\omega_i), i = 1,..., c$ 已知
  - 类条件概率密度形式已知 $P(x \mid \omega_i, \boldsymbol{\theta}_i), i = 1,...,c$
  - 未知的仅是 c 个参数向量 $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_c$ 的值，所有未知参数组成的向量记为 $\boldsymbol{\theta} = \left[\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_c\right]^T$

10.3.1 混合密度的最大似然估计

- **似然函数**
  - 混合密度：$p(\boldsymbol{x}\,|\,\boldsymbol{\theta}) = \sum_{i=1}^{c} p\left(\boldsymbol{x}\,|\,\omega_i, \boldsymbol{\theta}_i\right) P(\omega_i)$
  - 分量密度：类条件密度$p\left(x\,\middle|\,\omega_i, \boldsymbol{\theta}_i\right)$
  - 混合参数：先验概率 $P(\omega_i)$（有时也可未知，一起参与估计）
  - 设样本集X中的样本是从混合密度为$p(x\,|\,\boldsymbol{\theta})$的总体中独立抽取的，即满足独立同分布条件，$\boldsymbol{\theta}$确定但未知，则
    - ✓似然函数： $l(\boldsymbol{\theta}) = p(X\,|\,\boldsymbol{\theta}) = \prod_{i=1}^{N} p(x_i\,|\,\boldsymbol{\theta})$
    - ✓对数似然函数： $H(\boldsymbol{\theta}) = ln\left[l(\boldsymbol{\theta})\right] = \sum_{i=1}^{N} \ln p\left(\omega_i\,\middle|\,\boldsymbol{\theta}_i\right)$
    - ✓最大似然估计$\hat{\boldsymbol{\theta}}$就是使 $l(\boldsymbol{\theta})$ 或 $H(\boldsymbol{\theta})$ 取最大的 $\boldsymbol{\theta}$ 值

10.3.1 混合密度的最大似然估计

- **可识别性问题**
  - 求出$\hat{\boldsymbol{\theta}}$，就得到了$\hat{\boldsymbol{\theta}}_1, ..., \hat{\boldsymbol{\theta}}_c$，即从混合密度函数中恢复出了分量密度函数。可能吗？什么条件下可能?
  - **可识别性定义**：若对$\boldsymbol{\theta} \neq \boldsymbol{\theta}'$，混合分布中总存在$\boldsymbol{x}$使$p(\boldsymbol{x}\,|\,\boldsymbol{\theta}) \neq p(\boldsymbol{x}\,|\,\boldsymbol{\theta}')$，则密度$p(\boldsymbol{x}\,|\,\boldsymbol{\theta})$是可识别的
    - 大部分常见连续随机变量的分布密度函数都是可识别的
    - 离散随机变量的混合概率函数则往往是不可识别的

10.3.1 混合密度的最大似然估计

- **计算问题**
  - 对于可识别的似然函数，如何求最大似然估计?
  - 思路同监督情况，即如果$p(x|\boldsymbol{\theta})$对$\boldsymbol{\theta}$可微，则令$\nabla_{\boldsymbol{\theta}} H(\boldsymbol{\theta}) = 0$
  - 得一系列方程组，它们是最大似然估计的必要条件，若存在唯一极值则就是解

$$\nabla_{\theta_i} H(\theta) = \sum_{k=1}^{N} \frac{1}{p\left(x_k \middle| \theta\right)} \nabla_{\theta_i} \left[\sum_{j=1}^{c} p\left(x_k \middle| \omega_j, \theta_j\right) P(\omega_j) \right]$$

$$= \sum_{k=1}^{N} \frac{1}{p\left(x_k \middle| \theta\right)} \nabla_{\theta_i} \left[p\left(x_k \middle| \omega_i, \theta_i\right) P(\omega_i) \right] = \sum_{k=1}^{N} \frac{1}{p(x_k \middle| \theta)} \nabla_{\theta_i} \ln p(x_k \middle| \omega_i, \theta_i) \text{(设}\theta_i, \theta_j\text{独立)}$$

10.3.1 混合密度的最大似然估计

- **计算问题**
  - 其中后验概率 $P\left(\omega_i \middle| x_k, \theta_i\right) = \dfrac{p\left(x_k \middle| \omega_i, \theta_i\right) P(\omega_i)}{p\left(x_k \middle| \theta\right)}$
  - 有微分方程组： $\nabla_{\theta_i} H\left(\hat{\theta}\right) = 0, i = 1,2,\ldots,c$
  - 另，若$p(\omega_i)$也未知，则可引入限制条件: $P(\omega_i) > 0, i = 1,2,\ldots,c, \quad \sum_{i=1}^{c} P(\omega_i) = 1$
  - 可用拉格朗日法求条件极值问题，定义拉格朗日函数： $H' = H + \lambda \left[\sum_{i=1}^{c} P(\omega_i) - 1 \right]$

10.3.1 混合密度的最大似然估计

- **计算问题**
  - 可得

$$\hat{P}(\omega_i) = \frac{1}{N}\sum_{k=1}^{N}\hat{P}\left(\omega_i \mid x_k, \hat{\theta}_i\right), i = 1,2,\ldots,\mathrm{c}$$

$$\sum_{k=1}^{N} P\left(\omega_i \mid x_k, \omega_i\right)\nabla_{\theta_i}\ln p\left(x_k \mid \omega_i, \theta_i\right) = 0, i = 1,2,\ldots,c$$

其中，$\hat{P}\left(\omega_i \mid x_k, \hat{\theta}_i\right) = \dfrac{p\left(x_k \mid \omega_i, \hat{\theta}_i\right)\hat{P}(\omega_i)}{\sum_{j=1}^{c} p\left(x_k \mid \omega_i, \hat{\theta}_i\right)\hat{P}(\omega_i)}, i = 1,2,\ldots,c$

原则上可以从上述微分方程组中求解出最大似然估计$\hat{\theta}$和$\hat{P}\left(\omega_i\right)$，但实际上多数问题中只能采用某种迭代方法求解

10.3.2 混合正态分布的参数估计

- **混合高斯模型（mixture of Gaussian models）**
  - 混合模型中的各个分布都是多维正态分布，即

$$p\left(x\middle|\omega_i,\theta_i\right)\sim N\left(\mu_i,\Sigma_i\right)$$

  - 主要有三种情况：（“？”表示未知，“✔”表示已知）

情况	μ_i	Σ_i	$P(\omega_i)$	c
1	?	✓	✓	✓
2	?	?	?	✓
3	?	?	?	?

10.3.2 混合正态分布的参数估计

- **情况1：均值向量μ_i未知，其他参数已知**
  - 由上节知最大似然估计满足方程组

$$\sum_{k=1}^{N} \hat{\mathbf{P}}(\omega_i \mid x_k, \hat{\theta}_i) \nabla_{\theta_i} \ln p(x_k \mid \omega_i, \hat{\theta}_i) = 0, i = 1,...,c$$

  - 代入正态分布公式，可得

$$\sum_{k=1}^{N} P(\omega_i \mid x_k, \hat{\mu}_i) \Sigma_i^{-1} (x_k - \hat{\mu}_i) = 0$$

10.3.2 混合正态分布的参数估计

- **情况1：均值向量μ_i未知，其他参数已知**
  - 即

$$\hat{\mu}_i = \frac{\sum_{k=1}^{N} P(\omega_i \mid x_k, \hat{\mu}_i) x_k}{\sum_{k=1}^{N} P(\omega_i \mid x_k, \hat{\mu}_i)}$$

  - 样本的加权平均，物理意义明确；但是权值中包含未知参数，其中

$$\hat{P}(\omega_i \mid x_k, \hat{\mu}_i) = \frac{p(x_k \mid \omega_i, \hat{\mu}_i) P(\omega_i)}{\sum_{j=1}^{c} p(x_k \mid \omega_j, \hat{\mu}_j) P(\omega_j)}$$

10.3.2 混合正态分布的参数估计

- **情况1：均值向量μ_i未知，其他参数已知**
  - 迭代法求解：用某种方法得到一个较好的初值$\hat{\mu}_i(0)$，然后用下式迭代：

$$\hat{\mu}_i(j+1) = \frac{\sum_{k=1}^{N} P(\omega_i \mid x_k, \hat{\mu}_i(j)) x_k}{\sum_{k=1}^{N} P(\omega_i \mid x_k, \hat{\mu}_i(j))}$$

  - 梯度法，可能不是全局最优解，受初值影响大

10.3.2 混合正态分布的参数估计

- **情况2：类别数目c已知，其他参数均未知**
  - 思路与情况 1类似，将有关分布公式代入上小节方程即可，只是公式复杂一些，也可得到物理意义明确的方程式，但一般也只能用迭代法求解
- **情况3：参数均未知**
  - 无法用最大似然法求解

10.4 动态聚类算法

- **基于模型的方法**
  - 估计概率密度函数：困难
  - 寻找密度函数中的单峰：需要较多样本或先验知识，在非监督学习中不易满足
- **基于相似性度量的聚类方法**
  - 考查样本之间的相似性，根据相似性把样本集划分为若干子集，使某种表示聚类质量的准则函数最优
  - 相似性度量：以某种距离定义
  - 直观理解：同一类的样本的特征向量应是相互靠近的（前提：特征选取合理，能反映所感兴趣的关系）

10.4 动态聚类算法

- **动态聚类方法**
  - 多次迭代，逐步调整类别划分，最终使某准则达到最优
  - 三个要点：
    - ✓选某种距离作为样本相似性度量
    - ✓定义某个准则函数，用于评价聚类质量
    - ✓初始分类方法及迭代算法

10.4.1 C均值算法（K均值算法）

- **误差平方和准则**

$$J_e = \sum_{i=1}^{c} \sum_{y \in \Gamma_i} \| y - m_i \|^2 = \sum_{i=1}^{c} J_i$$

其中，Γ_i是第i个聚类，$i = 1,2,\ldots,\mathrm{c}$，其中样本数为$N_i$，$\Gamma_i$中样本均值为$m_i = \frac{1}{N_i} \sum_{y \in \Gamma_i} y$

- J_e反映了用c个聚类中心代表c个样本子集所带来的总的误差平方和
- J_e是样本集Y与类别集Ω的函数
- C均值算法的目标：最小化 ——最小方差划分
- **用c个码本来代表整个样本集**，使这种表示带来的总体误差最小 ——向量量化
- 无法用解析法求解，只能用迭代法，**通过不断调整样本的类别归属来求解**

10.4.1 C均值算法（K均值算法）

- **算法研究**

设已有一个划分方案，考查Γ_k中的样本y，若把y移入Γ_j，此时Γ_k变成$\tilde{\Gamma}_k$，此时Γ_j变成$\tilde{\Gamma}_j$，有

$$\tilde{m}_k = m_k + \frac{1}{N_k - 1}\left[m_k - y\right],\ \tilde{m}_j = m_j + \frac{1}{N_j + 1}\left[y - m_j\right]$$

$$\tilde{J}_k = J_k - \frac{N_k}{N_k - 1} \parallel y - m_k \parallel^2,\ \tilde{J}_j = J_j + \frac{N_j}{N_j + 1} \parallel y - m_j \parallel^2$$

若$\frac{N_j}{N_j + 1} \parallel y - m_j \parallel^2 < \frac{N_k}{N_k - 1} \parallel y - m_k \parallel^2$，则把$y$从$\Gamma_k$移入$\Gamma_j$会使$J_e$减小

10.4.1 C均值算法（K均值算法）

- **C均值算法**
  - （1）初始划分c个聚类Γ_i，计算$\boldsymbol{m}_i$和$\boldsymbol{J}_e$，$i=1,2,\cdots,c$
  - （2）任取一个样本$\boldsymbol{y}$，设$\boldsymbol{y}\in\Gamma_i$
  - （3）若$N_i=1$，则转（2）；否则继续
  - （4）计算$\rho_i=\dfrac{N_i}{N_i-1}\left\|\boldsymbol{y}-\boldsymbol{m}_i\right\|^2$，$\rho_j=\dfrac{N_j}{N_j+1}\left\|\boldsymbol{y}-\boldsymbol{m}_j\right\|^2, i\neq j$
  - （5）选ρ_j中的最小者，即若$\rho_k<\rho_j$，$\forall j$，则把$\boldsymbol{y}$从Γ_i移到Γ_k中：**有利于$\boldsymbol{J}_e$的减少**
  - （6）重新计算$\boldsymbol{m}_i$和$\boldsymbol{J}_e$，$i=1,2,\cdots,c$
  - （7）若连续N次迭代$\boldsymbol{J}_e$不改变，则停止；否则转（2）

10.4.1 C均值算法（K均值算法）

- **初始划分**：一般先选代表点，再进行初始分类
  - **代表点选择方法**
    - ✓经验选择
    - ✓随机分成*c*类，选各类重心作为代表点
    - ✓“密度”法：计算每个样本的一定球形邻域内的样本数作为“密度”，选“密度”最大的样本点作为第一个代表点，在离它一定距离之外选最大“密度”点作为第二个代表点，…，依次类推。
    - ✓用前*c*个样本点作为代表点
    - ✓用*c-1*聚类求*c*个代表点：各类中心外加离它们最远的样本点，从1类开始…

10.4.1 C均值算法（K均值算法）

- **初始分类方法**
  - 最近距离法：离哪个代表点近就归入哪一类
  - 最近距离法归类，但每次都重新计算该类代表点
  - 直接划分初始分类：第一个样本自成一类，第二个样本若离它小于某距离阈值则归入此类，否则建新类，……
  - 将特征归一化，用样本各特征之和作为初始分类依据

$$SUM(i)=\sum_{j=1}^{D} y_{ij}, i=1,...,N. \quad y_i \in R^D$$

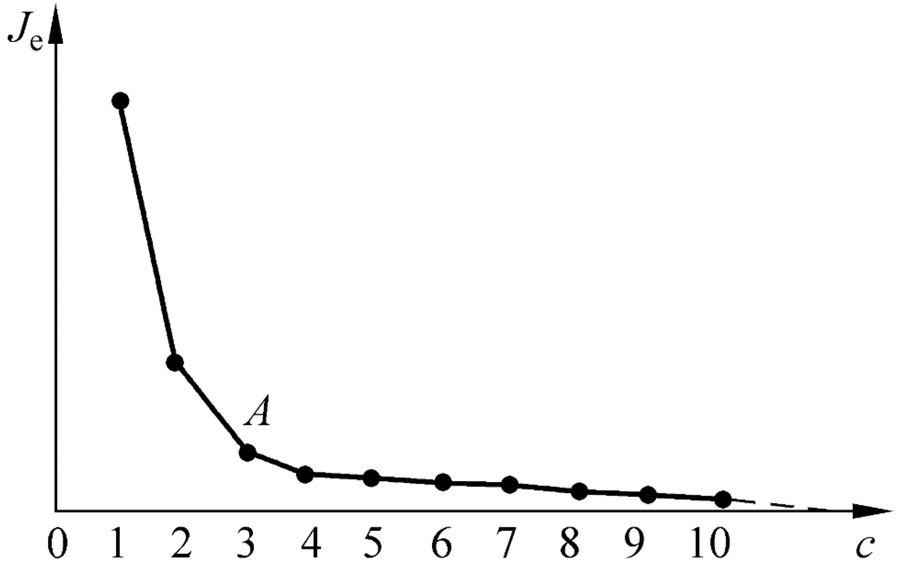
$$MA=\max_i SUM(i), \quad MI=\min_i SUM(i), \quad k \underset{\text{取整}}{=} \frac{(c-1)}{MA-MI}\left[SUM(i)-MI\right]+1$$

10.4.1 C均值算法（K均值算法）

- **说明**
  - 初始划分无一定之规，多为启发式方法
  - C均值方法结果受初值影响，是局部最优解

关于C均值方法的类别数c，多假定已知，由先验知识确定

  - 一种实验确定方法
    - 对$c = 1,2,3,\cdots$聚类，求各自的$J_e(c)$。
    - 找到其中的拐点（图中$\hat{c} = 3$） （注：实际问题中并不一定有明确的拐点）



10.4.1 C均值算法（K均值算法）

- **C均值聚类方法用于非监督模式识别的问题：**
  - 要求类别数已知
  - 是最小方差划分，并不一定能反映内在分布
  - 与初始划分有关，不保证全局最优

10.4.2 ISODATA方法

- **ISODATA**
  - iterative self-organizing data analysis techniques：迭代自组织数据分析技术
  - 一种改进的C均值算法
  - 特点：和C均值算法的区别
    - 成批样本修正，把所有样本调整完后才重新计算均值
    - 引入对类别的评判准则，可进行类别合并与分裂，突破事先给定类别数的限制

10.4.2 ISODATA方法

- **算法步骤**

  设有N个样本组成的样本集$\{\boldsymbol{y}_1, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_N\} \subset R^d$，事先设定如下参数：

  - K：期望得到的聚类数
  - θ_N：一个聚类中的最少样本数
  - θ_s：标准偏差参数
  - θ_c：合并参数
  - L：每次迭代允许合并的最大聚类对数
  - I：允许迭代的次数

10.4.2 ISODATA方法

- **算法步骤**

① 初始化，设初始聚类数c，中心$\boldsymbol{m}_i$，$i = 1,2,\cdots,c$ （不一定等于期望聚类数K）

② **把所有样本分到距离最近的类中**，Γ_i，$i = 1,2,\cdots,c$

③ 若某个类Γ_j中样本数过少（$N_j < \theta_N$），则去掉这一类（根据各样本到其他类中心的距离分别合入其他类），置$c = c - 1$

④ 重新计算均值：$\boldsymbol{m}_j = \dfrac{1}{N_j}\sum_{\boldsymbol{y}\in\Gamma_j}\boldsymbol{y}$，$j = 1,\cdots,c$

⑤ 计算第j类样本与其中心的平均距离和总平均距离

$$\bar{\delta}_j = \frac{1}{N_j}\sum_{\boldsymbol{y}\in T_j}\|\boldsymbol{y}-\boldsymbol{m}_j\|,\ \bar{\delta} = \frac{1}{N}\sum_{j=1}^{c}N_j\bar{\delta}_j,\ j = 1,\cdots,c$$

10.4.2 ISODATA方法

- **算法步骤**

⑥ 若是最后一次迭代（由总迭代次数I确定），则程序停止；

- 若$c \le K/2$，则转⑦（**分裂**）；
- 若$c \ge 2K$，或是偶数次迭代，则转⑧（**合并**）

⑦ 分裂

- 对每个类，求各维标准偏差$\sigma_j = \left[\sigma_{j1}, \sigma_{j2}, \cdots, \sigma_{jd}\right]^T$，

$$\sigma_{ji} = \sqrt{\frac{1}{N_j}\sum_{y_k \in T_j}\left(y_{ki} - m_{ji}\right)^2}, \ j = 1, \cdots, c, \ \ i = 1, \cdots, d$$

10.4.2 ISODATA方法

- **算法步骤**

⑦ 分裂

-对每个类，求出标准偏差最大的分量$\sigma_{j\max}$，　$j = 1,\ldots,c$

-**在$\sigma_{j\max}$对应的特征分量上分裂**：对各类的$\sigma_{j\max}$，存在$\sigma_{jmax} > \theta_s$（标准偏差参数），且$\bar{\delta}_j > \bar{\delta},\ N_j > 2(\theta_N + 1)$,或$c \le k/2$，则$\Gamma_j$分裂为两类，中心分别为$m_j^+$和$m_j^-$，置$c = c + 1$，$m_j^+ = m_j + \gamma_j$，$m_j^- = m_j - \gamma_j$，其中$\gamma_j = k\sigma_{jmax}$，$0 < k \le 1$

10.4.2 ISODATA方法

- **算法步骤**

⑧ 合并

-计算各类中心之间的距离算$\delta_{ij} = \left\| m_i - m_j \right\|, i,j = 1,\cdots,c, i \neq j$

-比较δ_{ij}与θ_c(合并参数)，对小于θ_c者排序$\delta_{i_1 j_1} < \delta_{i_2 j_2} < \ldots < \delta_{i_l j_l}$

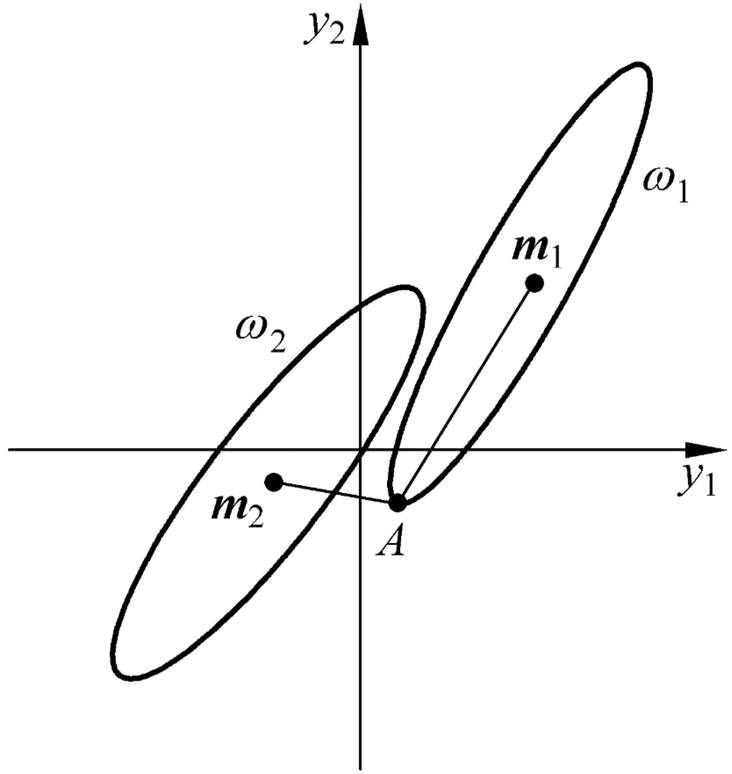
-从最小的$\delta_{i_l j_l}$开始，把每个$\delta_{i_l j_l}$对应的m_{i_l}和m_{j_l}合并：

$m_l = \frac{1}{N_{i_l} + N_{j_l}} \left[N_{i_l} m_{i_l} + N_{j_l} m_{j_l} \right]$， 并置$c = c - 1$。每次迭代中避免同一类被合并两次。

⑨ 若是最后一次迭代，则终止。否则迭代次数加1，转②　（必要时可调整算法参数）

10.4.3 基于核的动态聚类算法

- **C均值方法的缺点**
  - 用均值代表类，只适用于近似球状分布的类
- **改进**
  - 用核$K_j = K(\boldsymbol{y}, V_j)$来代表一个类$\Gamma_i$，$V_j$是参数集
  - 核K 可以是一个函数、一个点集或某种分类模型
  - 定义样本$\boldsymbol{y}$到类Γ_i（核K_j）之间的距离：$\Delta\left(\boldsymbol{y}, K_j\right)$



10.4.3 基于核的动态聚类算法

- **准则函数**

$$J_k = \sum_{i=1}^{c} \sum_{\boldsymbol{y} \in \Gamma_j} \Delta(\boldsymbol{y}, K_j)$$

- **算法流程**

① 选择初始划分，将样本集划分为c类，得到初始核K_j，$j = 1,2,\cdots,c$

② 按以下规则把各样本分类： 若$\Delta\left(\boldsymbol{y}, K_j\right) = \min\limits_{k=1,\cdots,c} \Delta\left(\boldsymbol{y}, K_k\right)$，则$\boldsymbol{y} \in \Gamma_j$

③ 更新K_j，$j = 1,2,\cdots,c$，若K_j不变，则终止；否则转②

④ C均值可看作基于核的动态聚类算法的特例：K_j为$\boldsymbol{m}_j$， Δ为欧氏距离

10.4.3 基于核的动态聚类算法

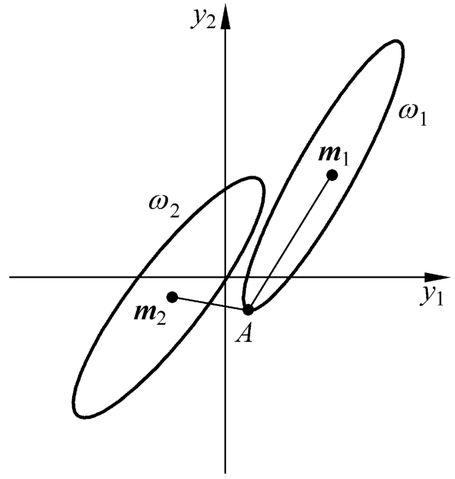
- 两种核函数举例

  -正态核函数

$$K_k(\boldsymbol{y}, V_j) = \frac{1}{(2\pi)^{d/2}\left|\sum_j\right|^{1/2}} exp\left\{-\frac{1}{2}(\boldsymbol{y}-\boldsymbol{m}_j)^T\hat{\sum}_j^{-1}(\boldsymbol{y}-\boldsymbol{m}_j)\right\}$$

$$V_j = \{\boldsymbol{m}_j, \sum_j^{\Lambda}\}$$

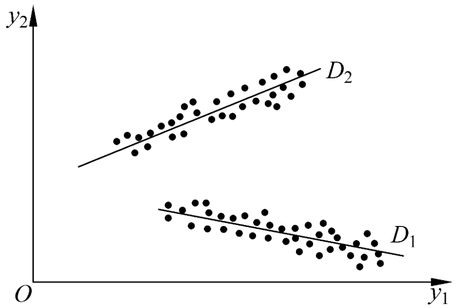
$$\Delta(\boldsymbol{y}, K_j) = \frac{1}{2}(\boldsymbol{y}-\boldsymbol{m}_j)^T\sum_j^{\Lambda}(\boldsymbol{y}-\boldsymbol{m}_j) + \frac{1}{2}log\left|\sum_j^{\Lambda}\right|$$

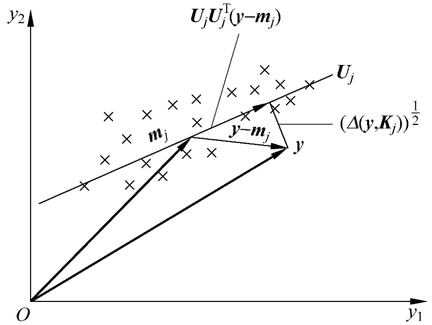


10.4.3 基于核的动态聚类算法

- **两种核函数举例**

  -主轴核函数





  - 用K-L变换得到样本子集的**主轴方向作为核**：$K(y, V_j) = U_j^T y,$

    $U_j^T = \left[u_1, u_2, \cdots, u_{d_j}\right]$是$\hat{\sum}_j$的$d_j$个最大本征值的本征向量系统

  - **计算样本y到Γ_j类主轴之间的欧氏距离**的平方来度量

$$\Delta(y, K_j) = \left[(y - m_j) - U_j U_j^T (y - m_j)\right]^T \left[(y - m_j) - U_j U_j^T (y - m_j)\right]$$

10.5 模糊聚类算法

10.5.1 模糊集的基本知识

- **确定集合**（脆集合）：元素或者属于或者不属于集合
- **模糊集合**：元素以一定的程序属于某集合 ——适于表达自然语言变量和常识性知识
- 几种叫法：模糊集、模糊逻辑、模糊数学、模糊系统、模糊技术、模糊方法
- 与其它技术相结合：模糊神经网络、模糊控制、模糊模式识别

10.5.1 模糊集的基本知识

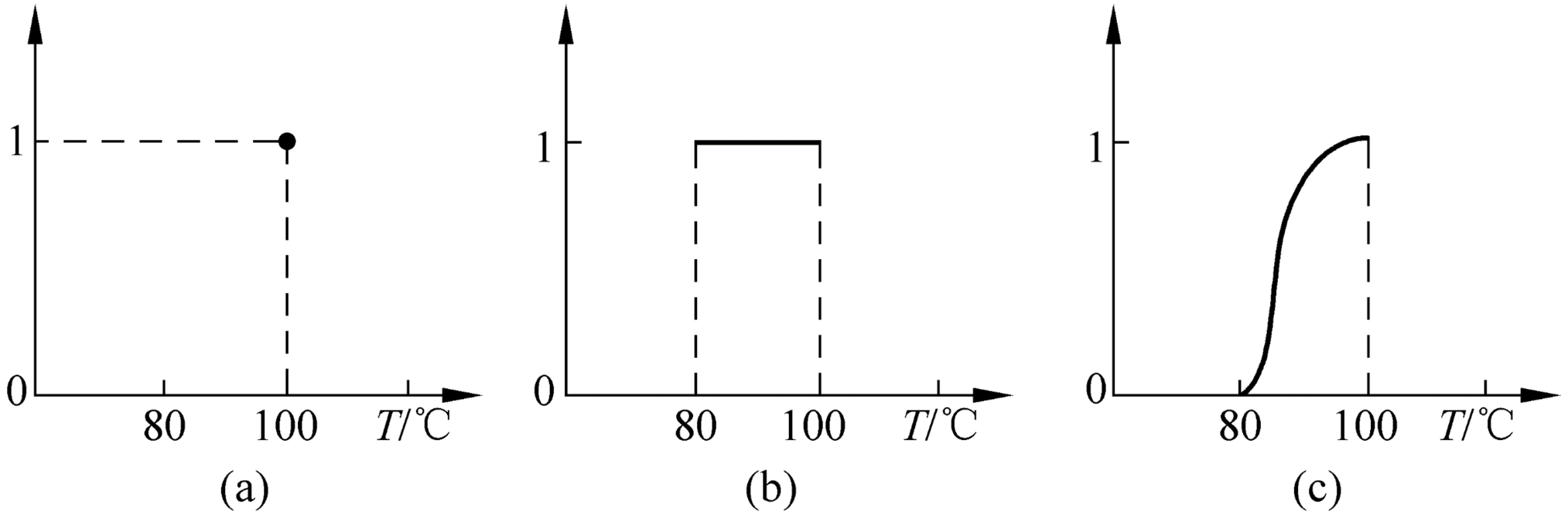
- **隶属度函数**$\mu_A(x)$：x属于集合A的程度
  - 自变量：所有可能属于A的对象，空间$X=\{x\}$
  - 值域：[0, 1]，$0 \le \mu_A(x) \le 1$，$\mu_A(x)=1 \Leftrightarrow x \in A$，$\mu_A(x)=0 \Leftrightarrow x \notin A$
- **模糊集合**：一个定义在$X=\{x\}$上的隶属度函数就定义了一个模糊集合A，或称定义在空间$X=\{x\}$上的模糊子集，表示为：

$$A=\left\{\left(\mu_A(x_i), x_i\right)\right\} \text{ 或 } A=\bigcup_i \mu_i/x_i$$

  - 模糊集A的支持集：$S(A)=\left\{x, x \in X, \mu_A(x)>0\right\}$

10.5.1 模糊集的基本知识

- 例：“开水”概念的表达



10.5.2 模糊C均值算法

- **C均值方法**
  - 把n个样本划分到c个类中，使各样本与其所在类的均值的误差平方和最小

$$J_e = \sum_{i=1}^{c} \sum_{y \in I_i} \| y - m_i \|^2$$

  - 把硬分类变成模糊分类，即得模糊c均值方法

10.5.2 模糊C均值算法

- **模糊C均值方法**
  - **符号约定**

    ✓样本集$\{x_i, i = 1,2,\cdots,n\}$

    ✓聚类中心$m_j,\ j = 1,2,\cdots,c$

    ✓$\mu_j(x_i)$：第i个样本对于第j类的隶属度函数
  - **聚类损失函数**

$$J_f = \sum_{i=1}^{c}\sum_{i=1}^{n}\left[\mu_j(x_i)\right]^b \parallel x_i - m_j \parallel^2$$

其中，$b > 1$可控制聚类结果的模糊程度

10.5.2 模糊C均值算法

- **模糊聚类：** $\min J_f$，对不同的**隶属度定义**，就得到不同的模糊聚类方法
- **模糊C均值:** $\sum_{j=1}^{c} \mu_j(x_i) = 1, \ i = 1, \cdots, n$
  - 令$\partial J_f / \partial m_j = 0$ 和 $\partial J_f / \partial \mu_j(x_i) = 0$，可得
  - $$m_j = \frac{\sum_{i=1}^{n} \left[\mu_j(x_i)\right]^b x_i}{\sum_{i=1}^{n} \left[\mu_j(x_i)\right]^b}, \ j = 1, \cdots, c$$
  - $$\mu_j(x_i) = \frac{\left(1 / \| x_i - m_j \|^2\right)^{1/(b-1)}}{\sum_{k=1}^{c} \left(1 / \| x_i - m_k \|^2\right)^{1/(b-1)}}, \ j = 1, \cdots, c, \ i = 1, \cdots, n$$

10.5.2 模糊C均值算法

- **模糊C均值算法（FCM）：**
  - 设定聚类数目c和常数b
  - 初始化各聚类中心$m_j,\ j=1,\cdots,c$(可参考上两节中的方法)
  - 重复下面的运算，直到各$\mu_j(x_i)$稳定
    - ✓用当前的m_j计算$\mu_j(x_i)$
    - ✓用当前的$\mu_j(x_i)$计算m_j
  - 如需要，可对所有模糊聚类去模糊化

10.5.3 改进的模糊C均值算法（AFC）

- **模糊C均值算法的缺点**
  - 由于 $\sum_{j=1}^{c}\mu_j(x_i)=1$，故对某些野值（本应属于各类程度都很小），隶属度可能较大
- **改进的模糊C均值算法**
  - 放松归一化条件为 $\sum_{j=1}^{c}\sum_{i=1}^{n}\mu_j(x_i)=n$
  - $$m_j=\frac{\sum_{i=1}^{n}\left[\mu_j(x_i)\right]^b x_i}{\sum_{i=1}^{n}\left[\mu_j(x_i)\right]^b},\quad j=1,\cdots,c\ \text{（不变）}$$
  - $$\mu_j(x_i)=\frac{n\left(1/\parallel x_i-m_j\parallel^2\right)^{1/(b-1)}}{\sum_{k=1}^{c}\sum_{l=1}^{n}\left(1/\parallel x_l-m_k\parallel^2\right)^{1/(b-1)}},\ j=1,\cdots,c,\ \ i=1,\cdots,n$$

10.5.3 改进的模糊C均值算法（AFC）

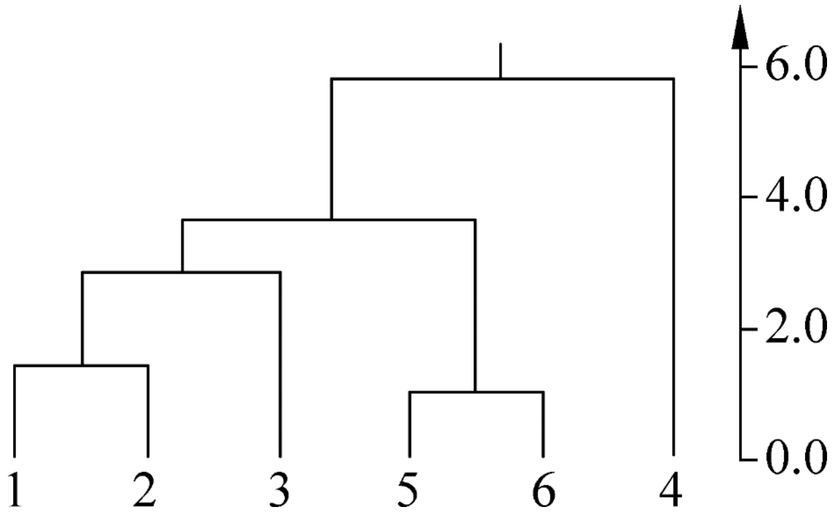
- **改进的模糊C均值算法步骤与模糊C均值相同**
  - 改进的模糊C均值算法所得到的$\mu_j(x_i)$可能大于1，不是严格意义下的隶属度函数。必要时可做归一化
  - 改进的模糊C均值算法有更好的鲁棒，且对给定的聚类数目不十分敏感
  - 但有时可能会出现一个类中只包含一个样本的情况，可通过在距离计算中引入非线性使之不会小于某值来改进
  - 改进的模糊C均值和C均值一样，依赖于初值

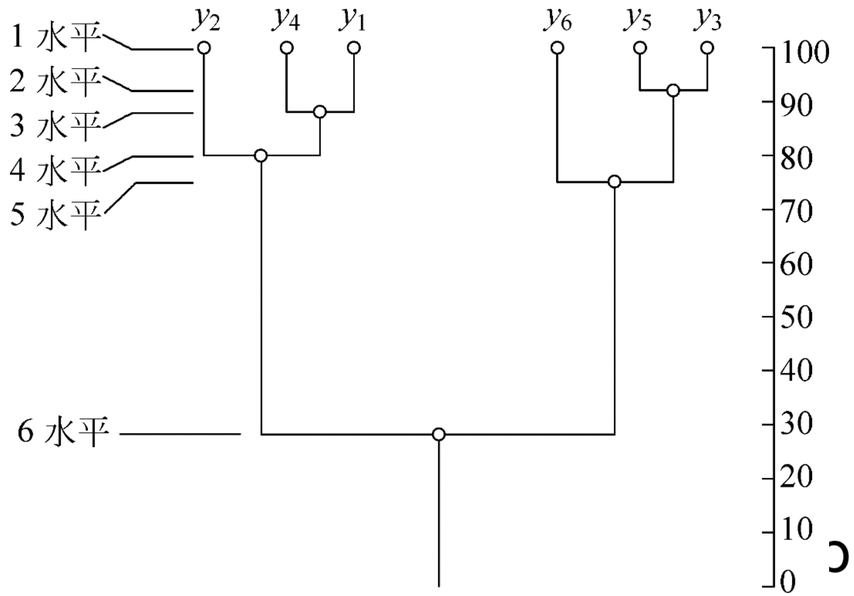
10.6 分级聚类算法

- **思想**
  - 从各类只有一个样本点开始，逐级合并，每级只合并两类，直到最后所有样本都归到一类
  - 聚类过程中逐级考查类间相似度，依此决定类别数

10.6 分级聚类算法

- **思想**
  - 树枝长度：反映结点/树枝之间的相似度或距离
  - 树枝位置：在不改变树的结构的情况下可以任意调整，调整方法需研究
  - 距离/相似性度量：多种选择，如欧式距离、相关、City Block、…





10.6 分级聚类算法

- **距离（相似性度量）**
  - 样本之间的度量
  - 聚类之间的度量
- **两种聚类策略**
  - bottom-up: agglomerative algorithm
  - top-down: divisive algorithm

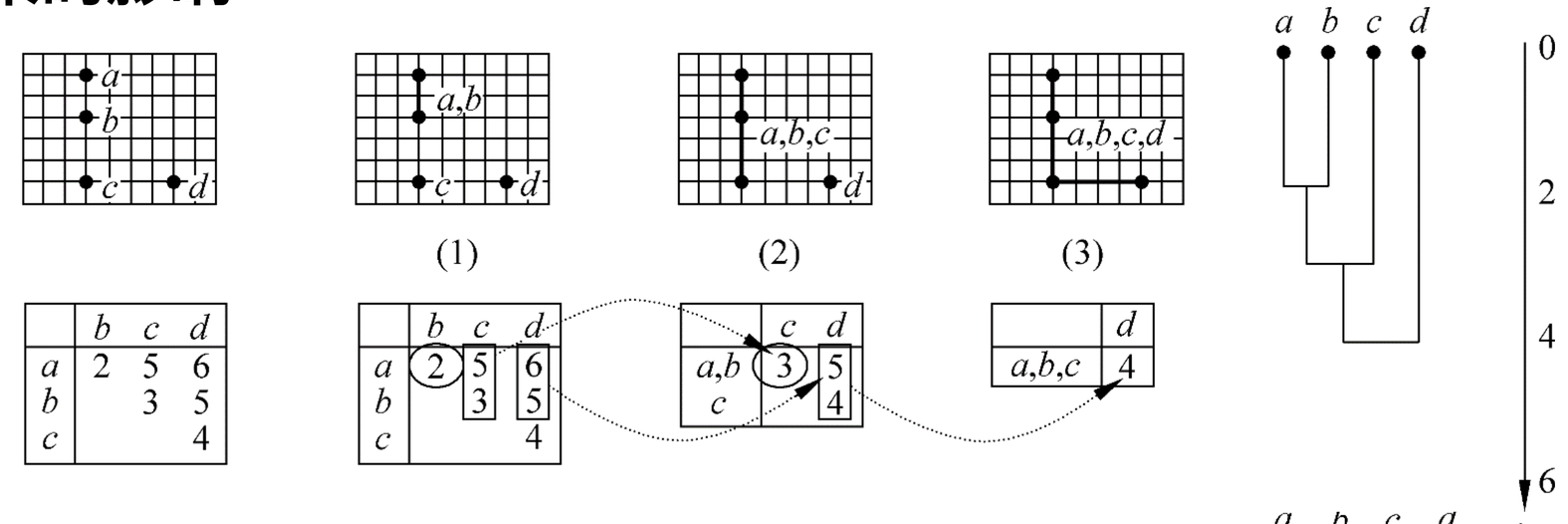
10.6 分级聚类算法

- **算法（从底向上）**
  - （1）初始化，每个样本形成一类
  - （2）把相似性最大（距离最小）的两类合并
  - （3）重复（2），直到所有样本合并为两类
- **常用的几种类间似性度量**
  - 最近距离（single-link），$\Delta\left(\Gamma_i,\Gamma_j\right)=\min_{y\in\Gamma_i,\tilde{y}\in\Gamma_j}\delta(y,\tilde{y})$
  - 最远距离（complete-link），$\Delta\left(\Gamma_i,\Gamma_j\right)=\max_{y\in\Gamma_i,\tilde{y}\in\Gamma_j}\delta(y,\tilde{y})$
  - 均值距离（average-link），$\Delta(\Gamma_i,\Gamma_j)=\delta(m_i,m_j)$

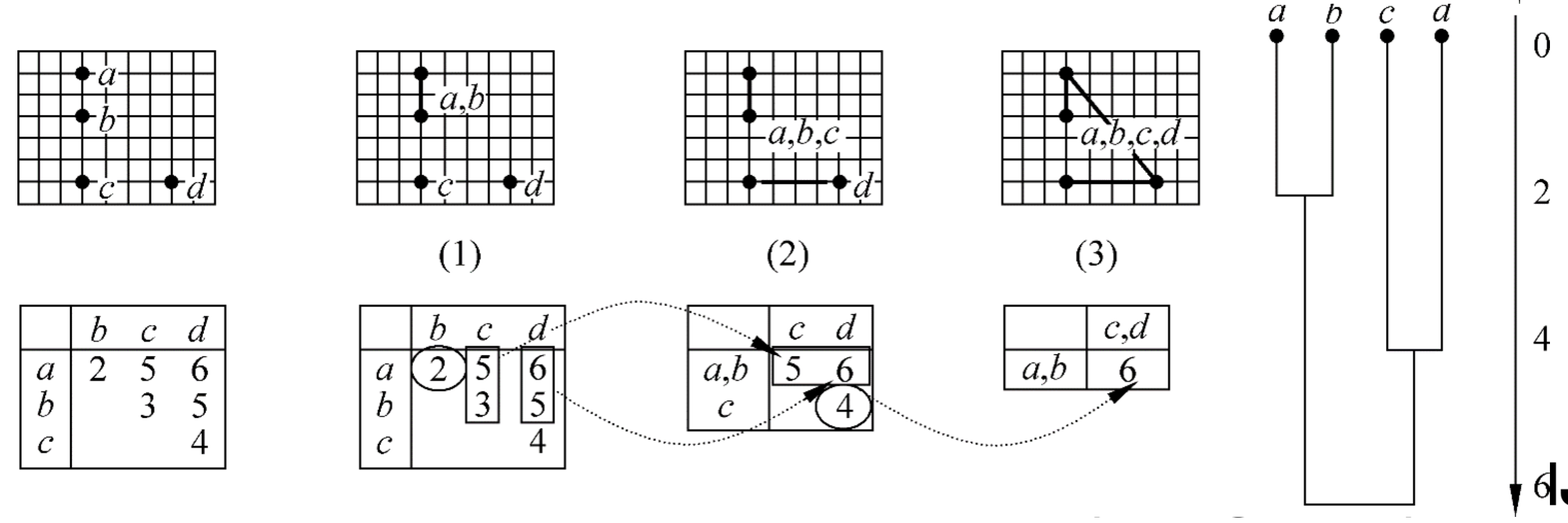
10.6 分级聚类算法

- 不同相似性度量对结果的影响

最近距离连接



最远距离连接



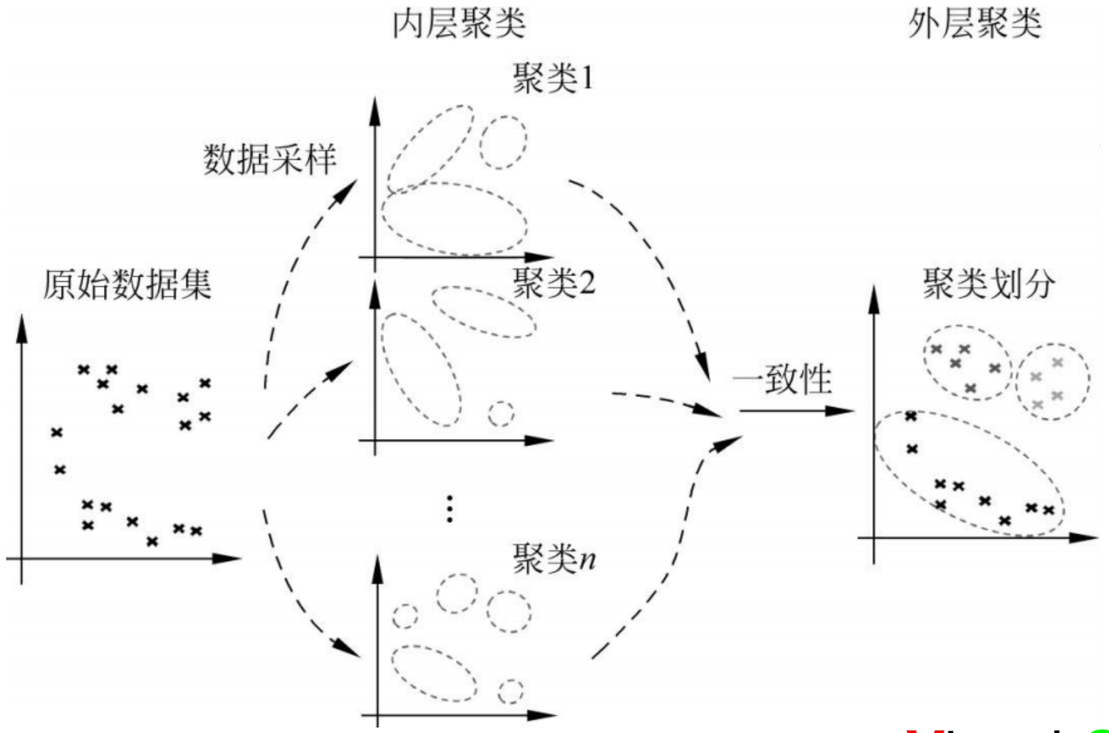
10.6 分级聚类算法

- **几点说明：**
  - 分级聚类是一种局部搜索，对样本中的噪声敏感
  - 聚类树的画法不是唯一的。同一类中的两个分支可以左右互换而不改变聚类结果，但会改变树的外观和分析者的判断

10.7 一致聚类方法

- **基本思路**

通过不同的数据抽样和不同方法进行多次聚类，再对结果进行合并，将在大多数结果中一致的结果作为最终的聚类划分依据。

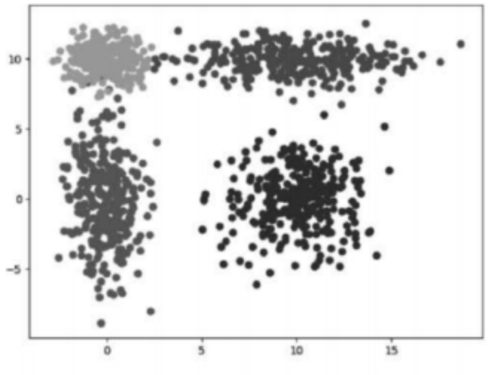


10.7 一致聚类方法

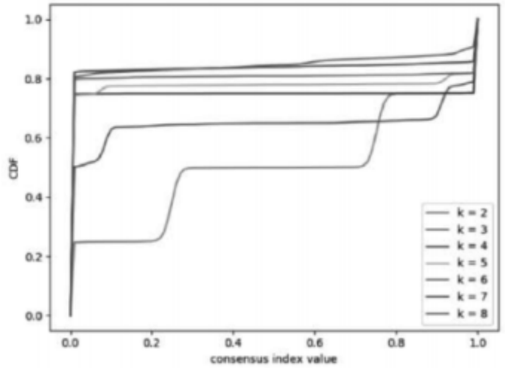
- **原始数据集**：$D = \{x_1, x_2, \ldots, x_N\}$
- **数据子集**：$D^{(1)}, \ldots, D^{(S)}$ (S次采样，无放回重采样)
- **内层聚类分析**：用不同的K对每一个数据子集进行聚类
  - **连接矩阵**$M_{(s)}^{(K)}$：表示在$D^{(S)}$上将数据聚类为K类时，样本i和样本j处于同一个类中，则$M_{(s)}^{(K)}(i,j) = 1$；否则$M_{(s)}^{(K)}(i,j) = 0$
  - 聚类数为K时的**整体一致性矩阵**$M^{(K)}$：$M^{(K)}(i,j) = \dfrac{\sum_s M_{(s)}^{(K)}(i,j)}{\sum_s I_s(i,j)}$，其中$I_{(s)}$为**指示矩阵**，但样本$i$和$j$都出现在数据集$D^{(S)}$中时，$I_s(i,j) = 1$

10.7 一致聚类方法

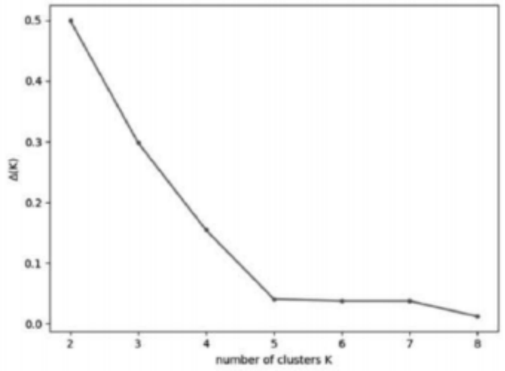
- 一致性矩阵$M^{(K)}$是对称矩阵，元素取值在[0,1]之间；可以可视化观察聚类结果的稳定性



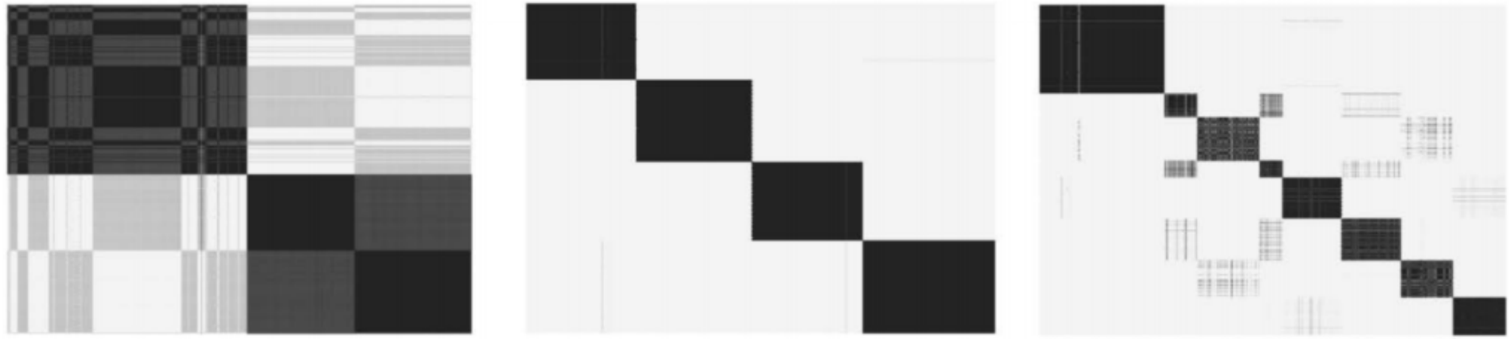
(a) 仿真数据
k=2



(b) CDF曲线
k=4



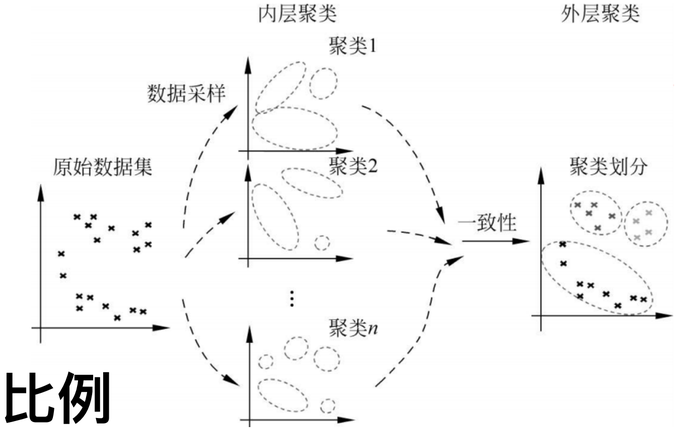
(c) $\Delta(K)$随K的变化
k=8



(d) $M^{(K)}$矩阵的可视化分析

10.7 一致聚类方法

- **外层聚类分析**
  - **距离度量矩阵**：$Dist^{(K)} = \left(1 - M^{(K)}\right)$，在此距离矩阵进行聚类（如层次聚类），将在大多数聚类结果中一致的结果最为最终的聚类划分依据，得到最终的聚类结果
- **类别数K**
  - CDF函数：$\mathrm{CDF}^{(K)}(t) = \dfrac{\sum_{i<j} I\left\{M^{(K)} \le t\right\}}{N(N-1)/2}$，$t \in [0,1]$
    - **表示一致性矩阵M中取值小于阈值t的样本对占总样本对数量的比例**
    - 若一致性高，则CDF的取值在t靠近*0*和*1*附近变化较大，中间平稳。
    - 若一致性差，则CDF的取值随着t增大缓慢上升。



10.7 一致聚类方法

- **类别数K**
  - 比较CDF函数曲线的线下面积AUC来确定聚类数K

$$A(K) = \sum_{i=2}^{N(N-1)/2} \left[x_i - x_{i-1}\right] CDF^{(K)}(x_i)$$

  - 其中$\left\{ x_1, x_2, \ldots, x_{\frac{1}{2}N(N-1)} \right\}$是对$M^{(K)}$元素取值从小到大的排序
  - 定义衡量$A(K)$变化的指标：$\Delta(K) = \begin{cases} A(K), K = 2 \\ \dfrac{A(K+1) - \hat{A}^{(K)}}{A^{(K)}}, K > 2 \end{cases}$，其中$\hat{A}^{(K)} = \max_{k \in \{2,\ldots,K\}} A(K)$，可以取$\Delta(K)$趋近稳定前一时刻的$K$