



大数据导论

Introduction to Big Data



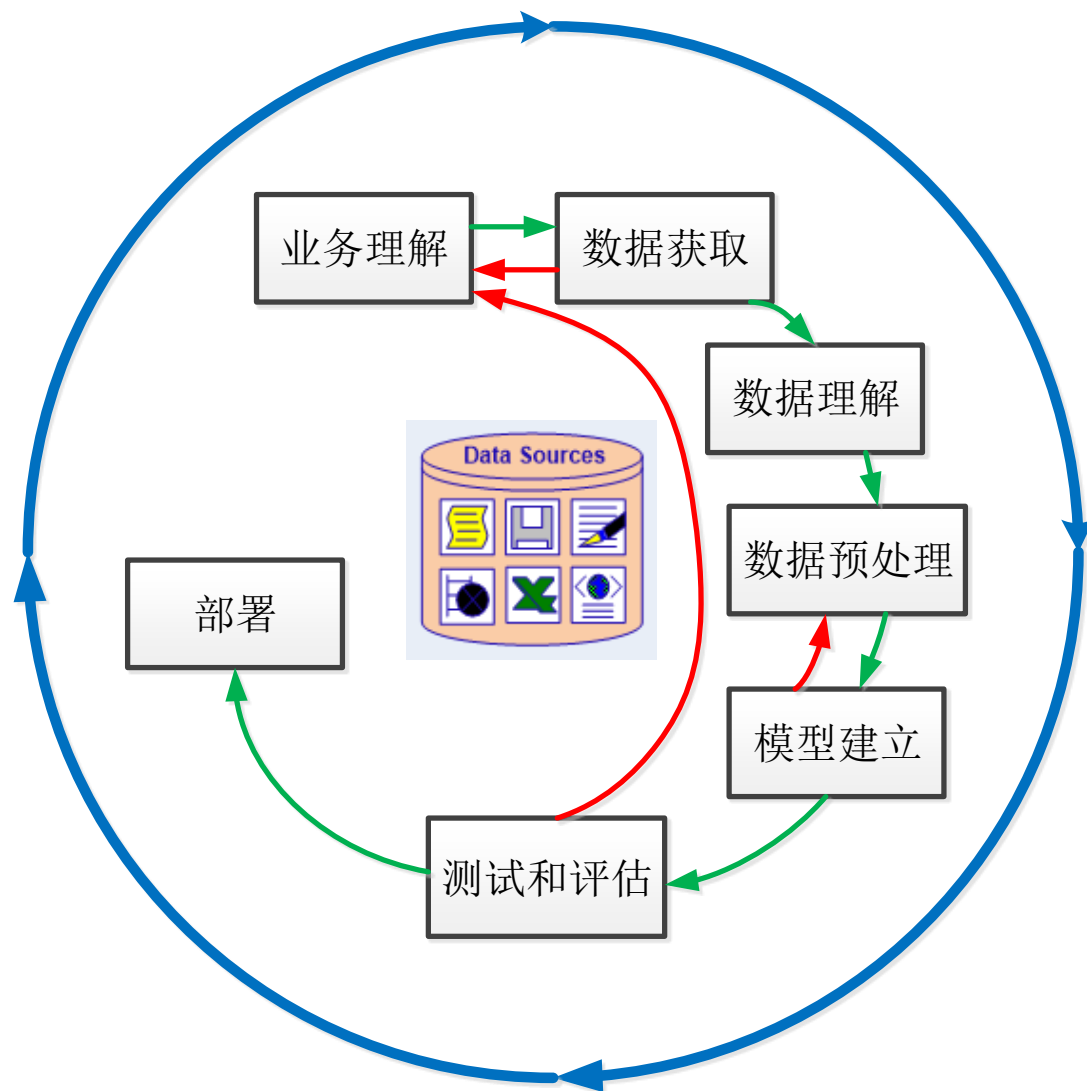
第四章：数据预处理

叶允明

计算机科学与技术学院

哈尔滨工业大学（深圳）

大数据挖掘过程



目录

- 为什么要对数据进行预处理?
- 数据清洗
- 数据集成和数据变换
- 数值属性的离散化
- 数据归约

为什么要对数据进行预处理？

为什么要对数据进行预处理？

- 现实世界中的数据是“脏”的
 - **不完整的**: 缺少属性值, 缺少某些感兴趣的属性或仅包含聚集数据
 - ✓ 例如, 职业=""
 - **含噪声的**: 包含错误或存在偏离期望的值
 - ✓ 例如, 薪水="-1"
 - **不一致的**: 特别是多源数据的集成
 - ✓ 例如, 年龄="20" 生日="03/07/2000"
 - ✓ 例如, 过去评级"1,2,3", 现在评级"A, B, C"
 - ✓ 例如, 重复元组之间的差异

为什么要对数据进行预处理？

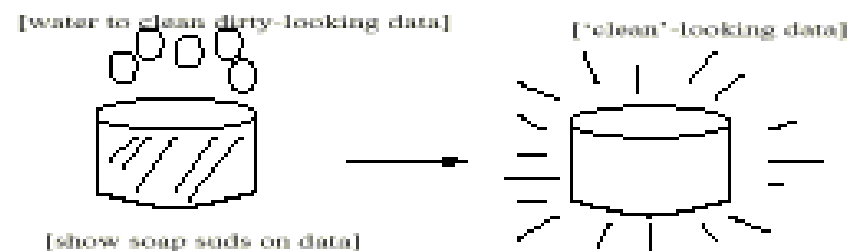
- 没有高质量的数据，就没有高质量的挖掘结果！
 - garbage in, garbage out!
 - 高质量的决策必然依赖于高质量的数据

数据预处理（data preprocessing）的主要任务

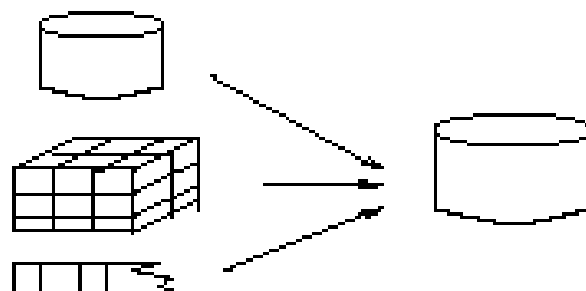
- 数据清洗（data cleaning）
 - 填充缺失值，光滑化噪声数据，识别或删除离群点，并解决不一致问题
- 数据集成（data integration）
 - 集成多个数据源（如多个数据库或文件）
- 数据变换（data transformation）
 - 规范化和聚集
- 数据归约（data reduction）
 - 得到数据集的简化表示，但能够产生同样的（或几乎同样的）分析结果
- 数据离散化（data discretization）
 - 主要针对数值型数据

数据预处理的形式

Data Cleaning



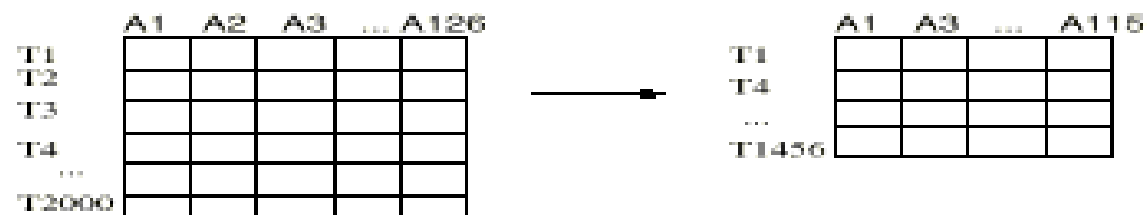
Data Integration



Data Transformation

-2, 32, 100, 59, 48 → -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction



数据清洗

数据清洗

- 重要性

- “数据清洗是数据仓库中三大问题之一”—Ralph Kimball
- “数据清洗是数据仓库中的头号问题”—DCI survey

- 数据清洗任务：

- 缺失值填充
- 识别离群点和光滑化噪声
- 纠正数据中的不一致问题
- 解决由数据集成造成的冗余

缺失值

- 数据并非完整的
 - 例如，许多数据对象的一些属性没有记录值，比如销售数据中的顾客收入
- 缺失值可能主要由于：
 - 未输入、或未记录、未存储的历史数据
 - 设备故障
 - 与其他记录数据不一致，因而被删除
- 可能需要推断缺失的数据

如何处理缺失数据？

- 忽略缺失记录：通常适合于只有少量记录有缺失值的情形
- 人工填写缺失值
- 自动填写
 - 使用一个全局常量：例如，“Unkown”，一个新类？！
 - 使用属性的均值
 - 使用与给定数据对象属同一类的所有对象的属性均值
 - 使用最可能的值：基于推理的，比如贝叶斯公式或决策树

噪声数据

- 噪声：被测量变量的随机误差
- 属性值不正确可能是由于
 - 错误的数据收集工具
 - 数据输入问题
 - 数据传输问题
 - 测量技术的局限性
 - 命名约定不一致

如何处理噪声数据？

- 方法一：删除噪声数据（适合于确定性、显著性噪声）
- 方法二：光滑化（smoothing）
 - 基于分箱的数据光滑化
 - ✓ 首先将数据排序并把有序数据分布到（等频）箱中
 - ✓ 然后通过箱均值光滑化、或通过箱中位数光滑化、或通过箱边界光滑化等。
 - 基于回归的光滑化方法
 - ✓ 用一个函数拟合数据来光滑化数据
- 基于聚类、离群点检测的方法
 - 检测和删除离群点
 - 用簇的均值（或其它集中趋势指标）光滑化数据

简单离散化方法：分箱(Binning)

- 等宽（距离）划分

- 将范围划分为N个相等大小的间隔：均匀网格
- 如果A和B是属性的最低值和最高值，则间隔的宽度将为： $W = (B - A) / N$
- 最直截了当的方式，但容易受到离群点的影响
- 倾斜（左偏、右偏）的数据效果可能不佳

- 等深（频率）划分

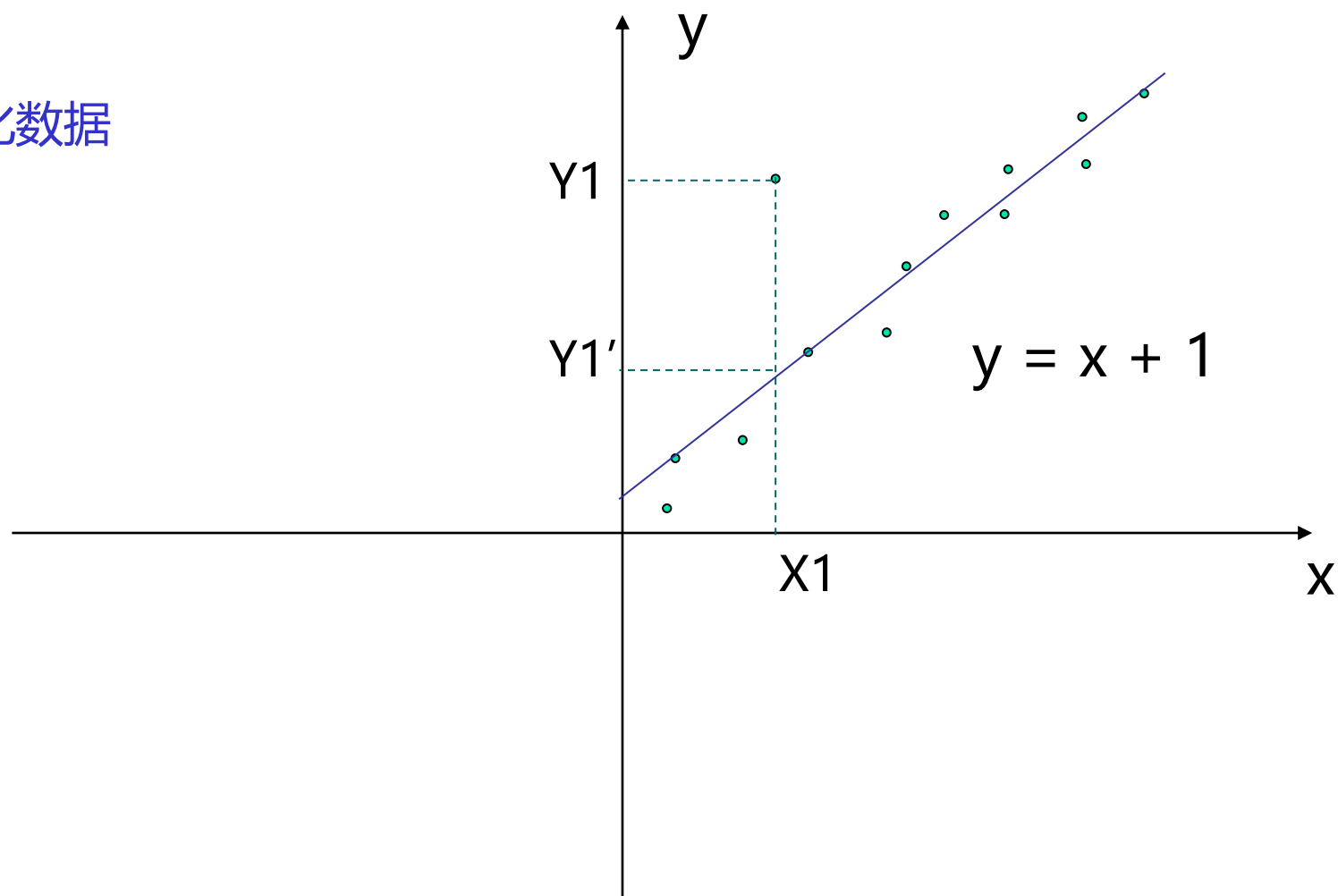
- 将范围分为N个区间，每个区间包含大致相同数量的样本
- 良好的数据扩张
- 管理分类属性可能很棘手

基于分箱的数据光滑化方法

- price (美元)数据首先排序: 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
 - * 被划分到等频 (等深) 的箱中:
 - ✓ - Bin 1: 4, 8, 9, 15
 - ✓ - Bin 2: 21, 21, 24, 25
 - ✓ - Bin 3: 26, 28, 29, 34
 - * 用箱均值光滑化:
 - ✓ - Bin 1: 9, 9, 9, 9
 - ✓ - Bin 2: 23, 23, 23, 23
 - ✓ - Bin 3: 29, 29, 29, 29
 - * 用箱边界光滑化:
 - ✓ - Bin 1: 4, 4, 4, 15
 - ✓ - Bin 2: 21, 21, 25, 25
 - ✓ - Bin 3: 26, 26, 26, 34

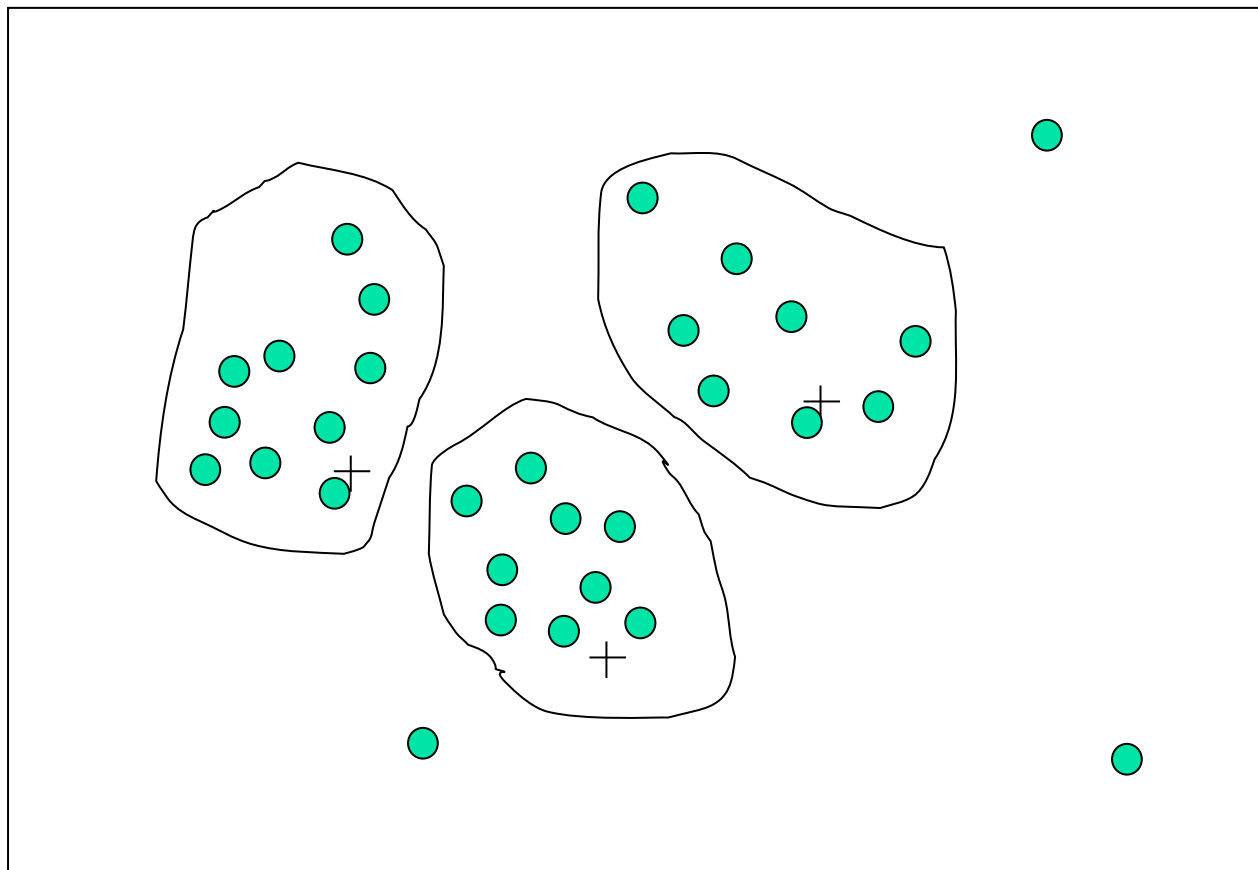
基于回归的光滑化方法

- 用给定数据拟合出函数
- 基于拟合的函数来光滑化数据



聚类分析

- 对给定数据进行聚类分析或离群点检测
- 处理识别出的噪声数据
- 或用集中趋势指标光滑化数据
 - 如均值、中位数、众数



数据集成与数据转换

数据集成及其问题

- 数据集成：
 - 将来自多个数据源的数据集合并到一个统一的数据集（库）中
- 数据库模式（Schema）集成：例如， $A.cust-id \equiv B.cust-\#$
 - 集成多个来源的元数据
- 实体识别问题：
 - 识别来自多个信息源的现实世界的实体
- 数据值冲突问题
 - 对于相同的现实世界实体，来自不同信息源的属性值是不同的
 - 可能的原因：不同的表示，不同的尺度，例如，公制与英制单位

数据集成中的属性冗余问题

- 当多个数据库集成时，通常会出现冗余数据
 - 对象识别：相同的属性或对象在不同的数据库中可能具有不同的名称
 - 可导出数据：一个属性可以是另一个表中的“派生”属性，例如,年收入
- 可以通过相关性分析（correlation analysis）来检测冗余属性
- 去除或减少冗余属性，可提高挖掘速度和质量

相关性分析（类别型属性）

- χ^2 (卡方) 检验

$$\chi^2 = \sum \frac{(\text{Observed} - \text{Expected})^2}{\text{Expected}} = \sum_{i=1}^c \sum_{j=1}^r \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

- χ^2 值越大，变量越可能相关
- 对 χ^2 值贡献最大的单元是那些实际计数与期望计数非常不同的单元
- 相关性并不意味着因果关系
 - 医院数量和城市中的汽车盗窃数量是相关的
 - 两者都与第三个变量有因果关系：人口

卡方计算：示例

	Play chess	Not play chess	Sum (row)
Like science fiction	250 (90)	200 (360)	450
Not like science fiction	50 (210)	1000 (840)	1050
Sum(col.)	300	1200	1500

- χ^2 (卡方) 计算 (括号中的数字是根据两个属性的数据分布计算的期望计数)

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} = 507.93$$

- 它表明, like_science_fiction和play_chess是相关的

➤ 在自由度1、置信水平0.001下, 拒绝假设的值为10.828

相关性分析（数值型属性）

- 相关系数(又称Pearson积矩系数)

$$r_{A,B} = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^n (a_i b_i) - n\bar{A}\bar{B}}{n\sigma_A\sigma_B}$$

- n 是数据对象的个数, \bar{A} 和 \bar{B} 分别是属性 A 和 B 的均值, σ_A 和 σ_B 分别是 A 和 B 的标准差, 而 $\sum(a_i b_i)$ 是 A 、 B 的叉积和。
- 如果 $r_{A,B} > 0$, A 和 B 正相关 (A 的值随 B 的值的增加而增加) 。该值越大, 相关性越强。
- $r_{A,B} = 0$: 独立
- $r_{A,B} < 0$: 负相关

相关性分析（数值型属性）

- 协方差:
$$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{n}$$
- 相关系数与协方差的关系:
$$r_{A,B} = \frac{Cov(A, B)}{\sigma_A \sigma_B}$$
- **Positive covariance:** If $Cov_{A,B} > 0$, then A and B both tend to be larger than their expected values.
- **Negative covariance:** If $Cov_{A,B} < 0$, then if A is larger than its expected value, B is likely to be smaller than its expected value.
- **Independence:** $Cov_{A,B} = 0$, but the converse is not true

数据变换

- 光滑化：去掉数据中的噪声
- 规范化：把属性值按比例缩放，使之落入一个特定的小区间
 - 最小-最大值规范化
 - Z-score规范化
 - 小数定标规范化
- 类别型属性：属性值泛化：基于概念层次
- 数值型属性：离散化
- 属性/特征构造
 - 例如，面积=长*宽

数据变换：规范化

- 最小-最大值规范化：映射到 $[new_min_A, new_max_A]$

$$v' = \frac{v - min_A}{max_A - min_A} (new_max_A - new_min_A) + new_min_A$$

- 例，假设收入的最小值与最大值分别为12,000 美元和 98,000 美元，把收入映射到区间 $[0.0, 1.0]$ ，那么73,000 美元将变换为：

$$\frac{73,600 - 12,000}{98,000 - 12,000} (1.0 - 0) + 0 = 0.716$$

- Z-score规范化 (μ : 均值, σ : 标准差) : $v' = \frac{v - \mu_A}{\sigma_A}$

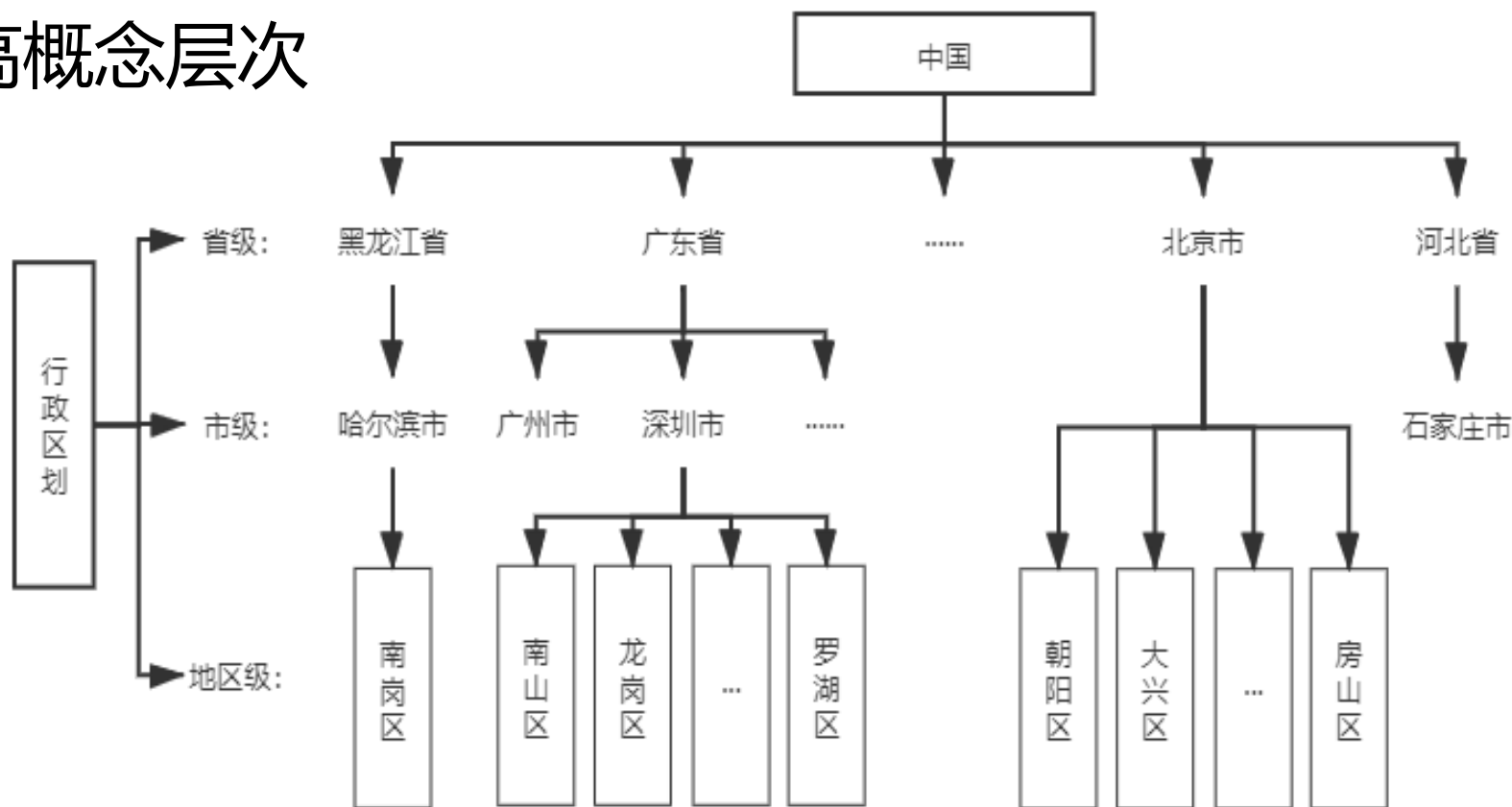
- 例，假设 $\mu = 54,000$, $\sigma = 16,000$ 。那么73,000 美元将变换为：

$$\frac{73,600 - 54,000}{16,000} = 1.225$$

- 小数定标规范化: $v' = \frac{v}{10^j}$, j 是使得 $\text{Max}(|v'|) < 1$ 的最小常数。

类别型属性的属性值泛化

- 类别型属性
- 属性值泛化：替换为高概念层次
- 增强取值的统计意义



数值属性的离散化

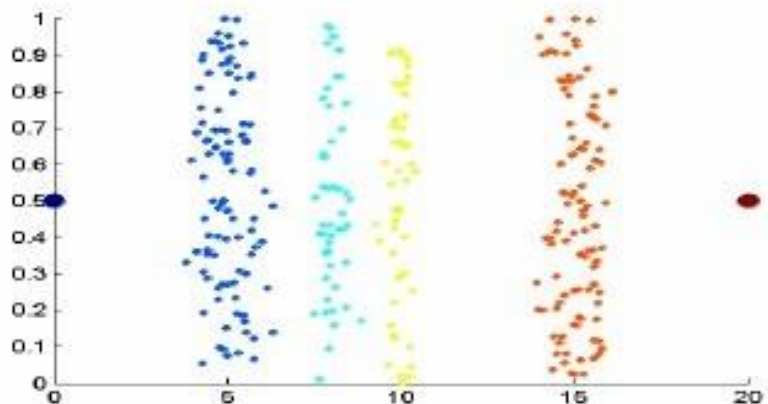
数值属性的离散化问题

- 为什么要对数值属性进行离散化
 - 将连续的数值属性取值范围划分为间隔
 - 一些分类算法仅接受离散型（类别型）属性
 - 通过离散化减少数据大小
- 离散化的相关概念
 - 通过将属性的范围划分为区间来减少给定连续属性取值的数量
 - 然后可以使用区间标签来替换原始的数值型取值
 - 有监督与无监督
 - 拆分（自顶向下）与合并（自底向上）
 - 可以递归地对属性进行离散化

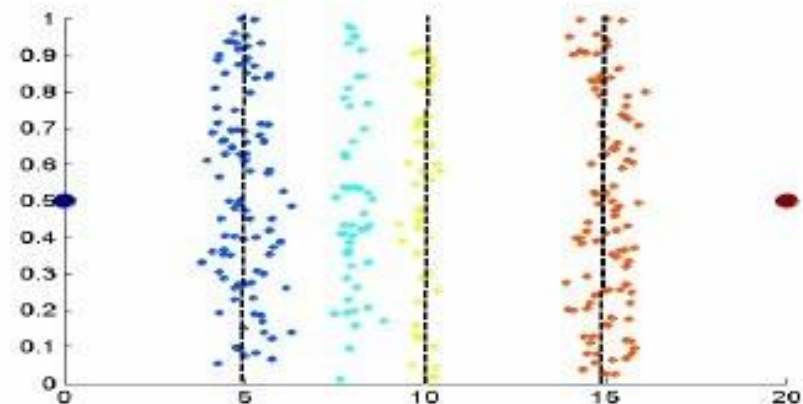
数值数据的离散化方法

- 典型方法：所有方法都可以递归应用
 - 分箱和直方图分析（如上所述）
 - ✓ 自顶向下划分，无监督
 - 聚类分析（如上所述）
 - ✓ 自顶向下划分或自底向上合并，无监督
 - 基于熵的离散化：有监督，自顶向下划分
 - 基于 χ^2 分析的区间合并：有监督，自底向上合并

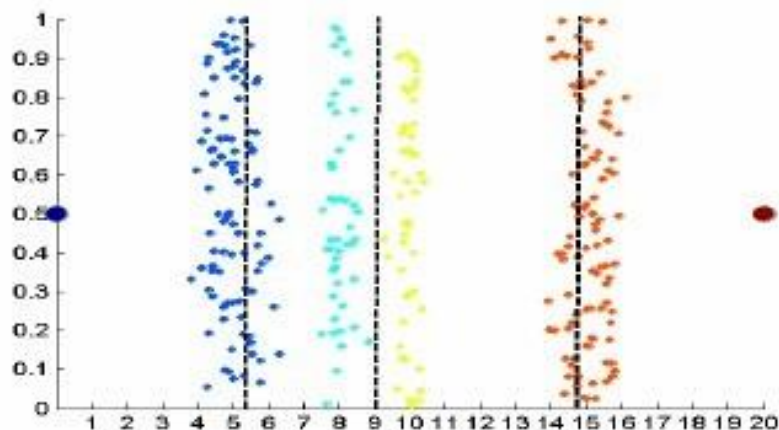
不使用类标签的无监督离散化



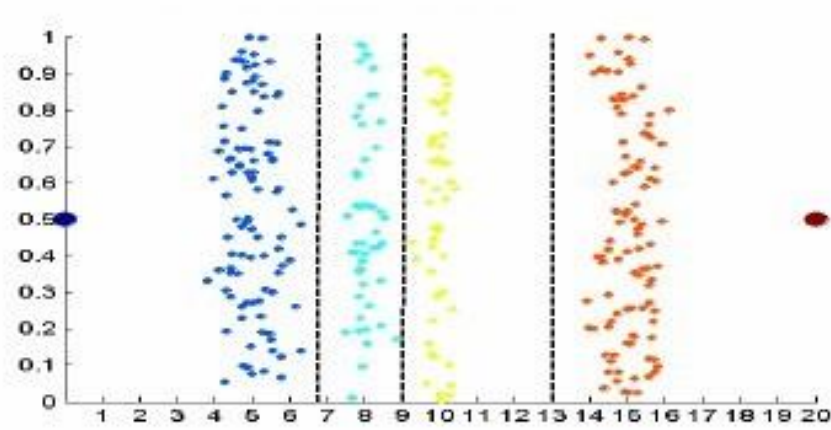
Data



Equal interval width



Equal frequency



K-means

基于熵的离散化

- 给定一组样本 S , 如果使用边界 T 将 S 划分为两个区间 S_1 和 S_2 , 则划分后的信息熵为:

$$I(S, T) = \frac{|S_1|}{|S|} \text{Entropy}(S_1) + \frac{|S_2|}{|S|} \text{Entropy}(S_2)$$

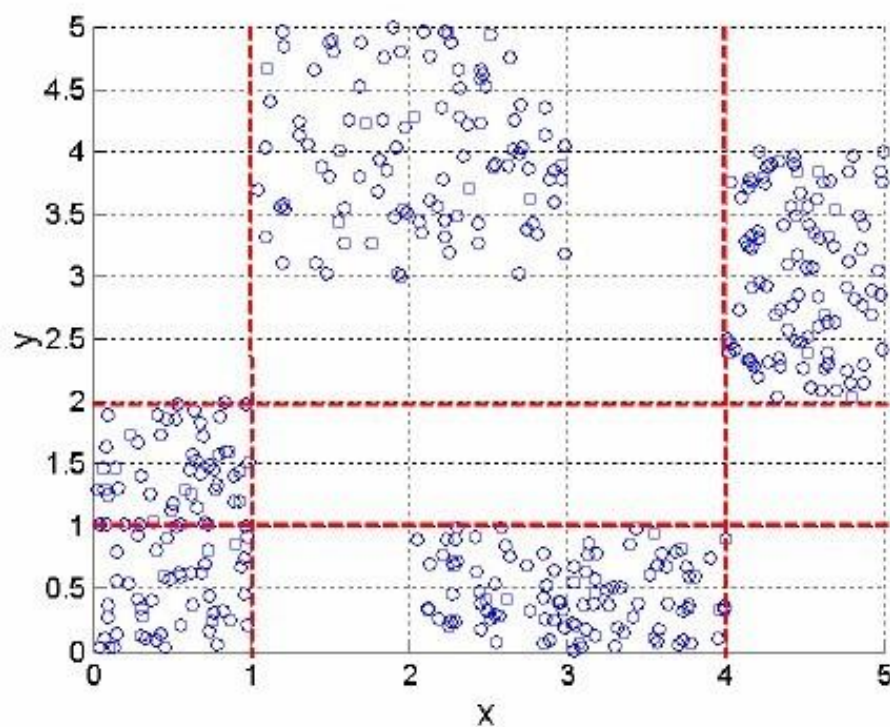
- 熵是根据集合中样本的类别分布计算的。给定 m 个类, p_i 是 S_1 中类 i 的概率

$$\text{Entropy}(S_1) = - \sum_{i=1}^m p_i \log_2(p_i)$$

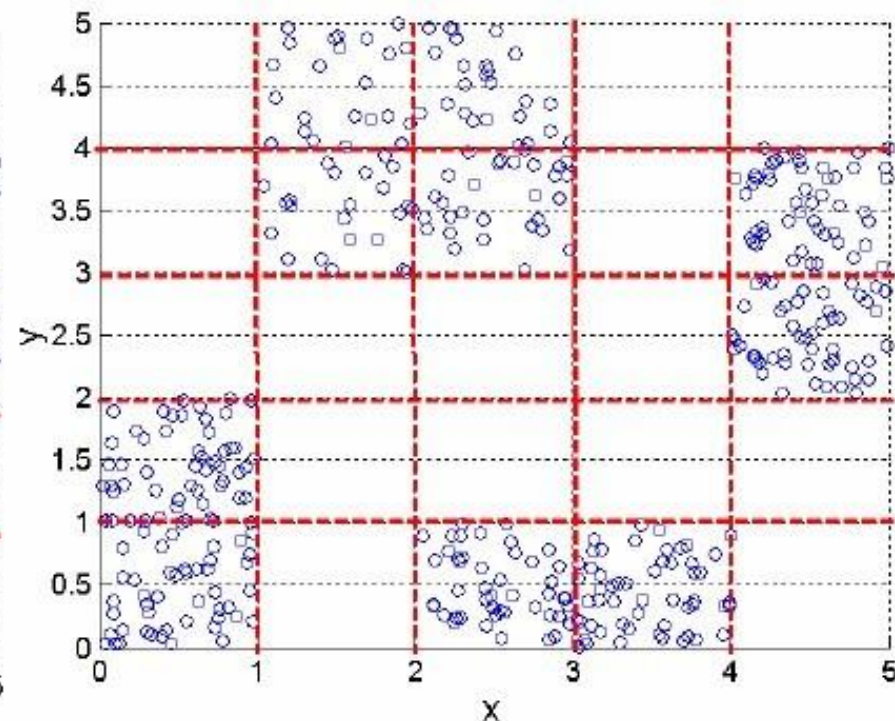
- 选择使所有可能边界上的信息熵最小化的边界作为二元离散化
- 递归地将该过程应用于获得的区间, 直到满足某些停止条件
- 这样的边界可以减小数据大小并提高分类准确性

使用类标签的离散化

基于熵的方法



3 categories for both x and y



5 categories for both x and y

基于 χ^2 分析的离散化

- 基于合并（自底向上）的方法
- 合并：找到最佳的相邻区间并将它们递归的合并以形成更大的间隔
- ChiMerge [Kerber AAAI 1992, See also Liu et al. DMKD 2002]
 - 最初，数值属性 A 的每个不同值被认为是一个区间
 - 对每对相邻的区间进行 χ^2 检验
 - 具有最小 χ^2 值的相邻区间被合并在一起，因为低 χ^2 值表明它们具有相似的类分布
 - 该合并过程递归地进行，直到满足预定义的停止条件（例如显著性水平，最大间隔，最大不一致性等）。

数据归约

数据归约

- 为什么要数据归约？
 - 复杂的数据分析/挖掘在海量的数据集上运行时间很长
- 数据归约的目标：
 - 得到数据集的简化表示，但能够产生同样的（或几乎同样的）分析结果
- 数据归约的主要方法
 - 维归约（dimension reduction）
 - 数据对象（记录）归约（data record reduction）
 - 数值归约（numerosity reduction）

维归约：属性子集选择

- 特征选择（即属性子集选择）：
 - 选择最小属性集，使得数据类的概率分布尽可能的接近使用所有属性得到的原始分布。
 - 基于更少属性挖掘出的“模式”更易于理解。
- 启发式方法（由于指数级的选择数量）：
 - 逐步向前选择
 - 逐步向后删除
 - 逐步向前选择和逐步向后删除的组合
 - 评价属性好坏的启发式指标：
 - ✓ 有监督： χ^2 （数值型用相关系数）、信息熵、互信息等
 - ✓ 无监督：方差、文本数据中的TF、IDF等

维归约：属性子集选择

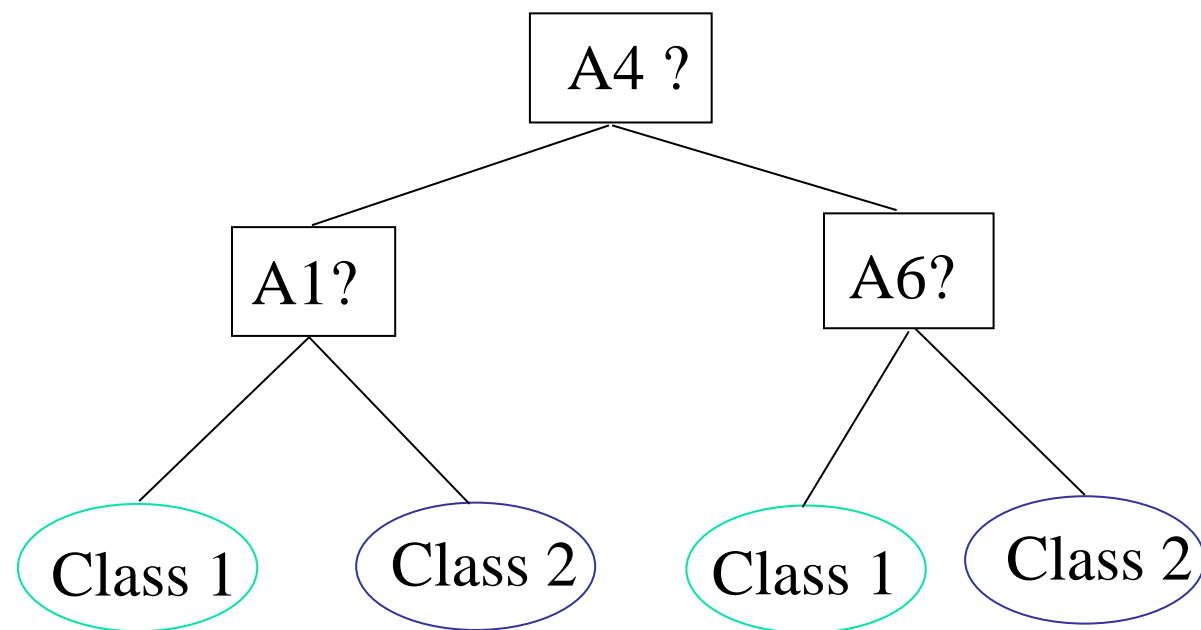
- 基于挖掘结果的方法：

- 例如基于决策树归纳的特征选择
- 基于正则化线性回归的特征选择方法
- 后续章节将讲解

决策树归纳的例子：

初始属性集：

{A1, A2, A3, A4, A5, A6}

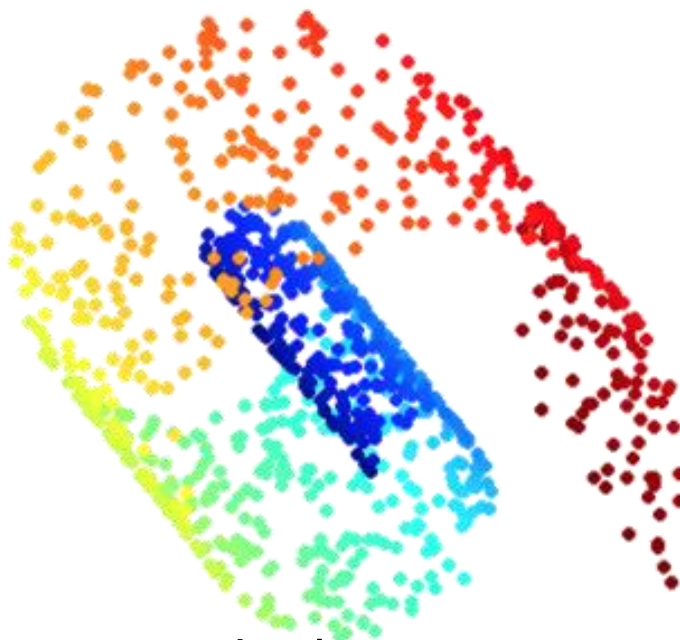


归约后的属性集：{A1, A4, A6}

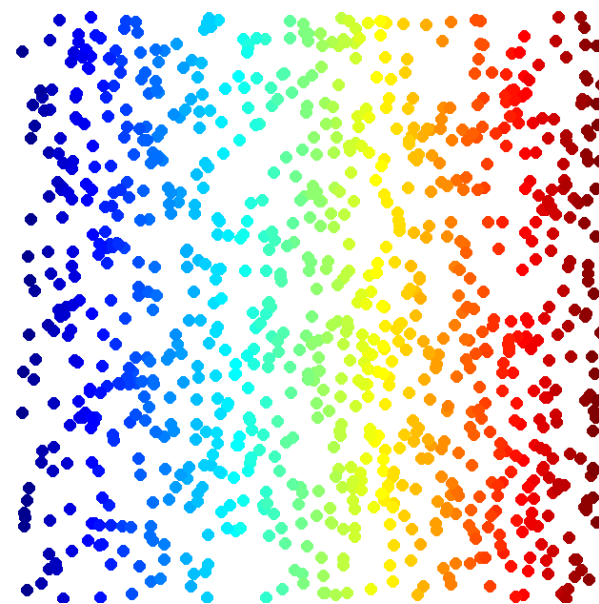
维归约：子空间投影

- 典型方法：

- 思想：embedding
- 主元分析PCA类方法
- Autoencoder
- Word-2-vector
- 其它表示学习方法



看起来, 3-D



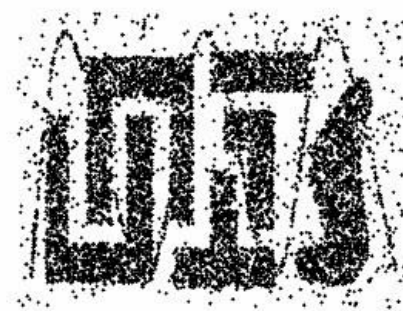
实际, 2-D

数据对象归约方法：抽样

- 抽样：用一个更小的数据对象集合 S 来表示整个数据集 N
- 抽样的复杂度可能亚线性于数据的大小
- 选择数据的代表性子集
 - 当数据的类别分布不平衡时，简单的随机采样可能具有非常差的性能

抽样方法

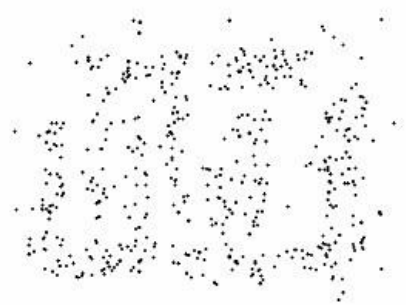
- 无放回简单随机抽样
 - 每次抽取一个样本，不放回
- 有放回简单随机抽样
 - 一个元组被抽取后，又被放回原处
 - 同一个元组可以被再次抽取
- 分层抽样 (stratified sampling)
 - 将数据划分为多个互不相交的分區
 - 然后从每个分區中随机抽取样本



8000 points

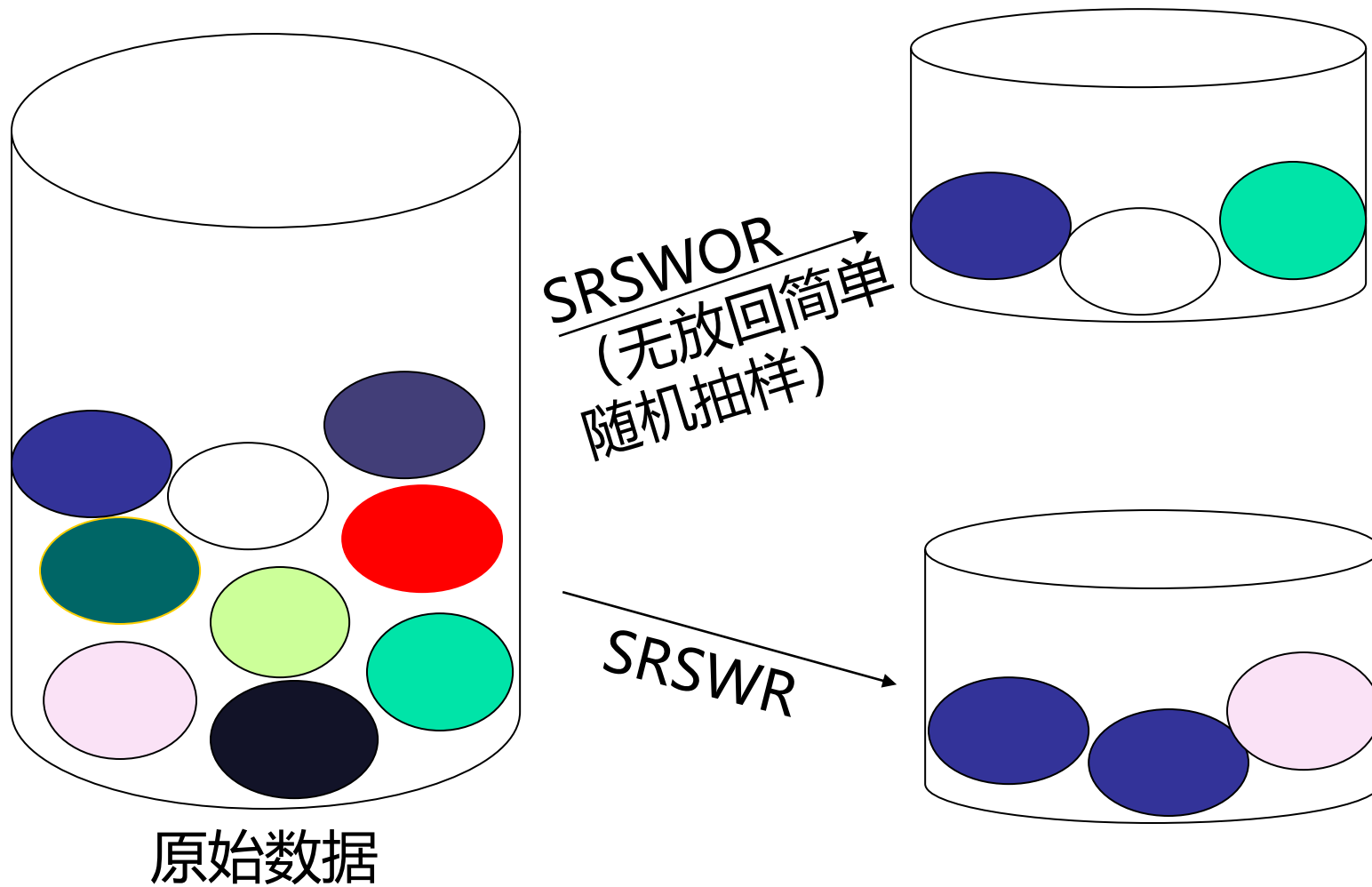


2000 Points



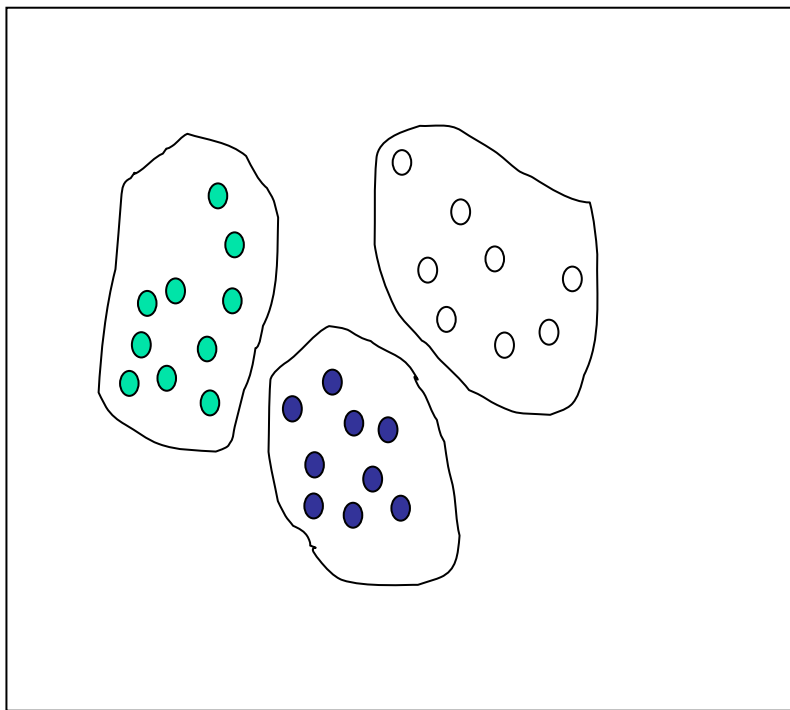
500 Points

抽样：有放回和无放回

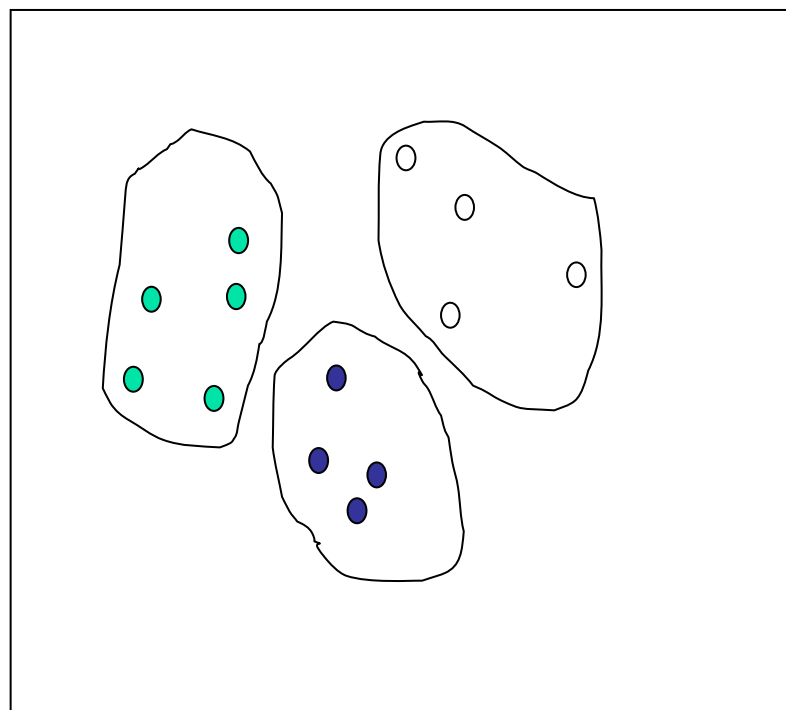


抽样：簇抽样或分层抽样

原始数据



簇抽样/分层抽样

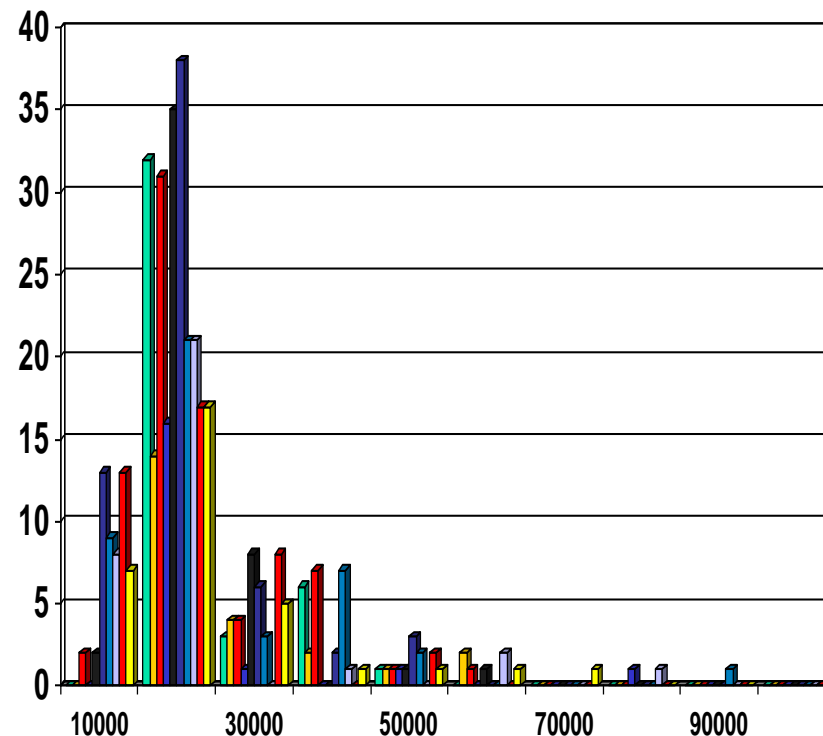


数值归约

- 用较小的取值集合（定义域）替换原数据
- 非参数方法
 - 不假设模型
 - 主要方法：直方图，聚类，抽样
- 参数方法：
 - 使用模型拟合数据：只需要存放模型参数，而不是实际数据
 - 例如：基于线性模型的数值规约

数值归约方法：直方图

- 将数据划分为不相交的桶
- 划分规则：
 - 等宽：每个桶宽度区间一致
 - 等频：每个桶频率一致（等深）
 - V-optimal：具有最小直方图方差（每个桶代表的原始值的加权和）
 - MaxDiff：设置每对之间的桶边界对，具有 $\beta-1$ 最大差异



致谢

- 部分图表、文字参考教材、互联网等资料，仅供公益性的学习参考，在此表示感谢！如有版权要求请联系：yym@hit.edu.cn，谢谢！