# 数据预处理
# Data Preprocessing

# 目　录

CONTENTS

## 为什么需要预处理?

- **目标：提升数据质量**
- **数据质量的含义**
- 准确性（Accuracy）
  - 设备故障、输入错误、程序bug、格式不一致等导致
- 完整性（Completeness）
- 一致性（Consistency）
- 时效性（Timeliness）
- 可信性（Believability）
- 可解释性（Interpretability）

## 现实世界的数据

- 不完整的
  - 缺少属性值或某些感兴趣的属性，或仅包含聚集数据。
- 含噪声的
  - 包含错误或存在偏离期望的离群值。
- 不一致的
  - 采用的编码或表示不同，如属性名称不同
- 冗余的
  - 如属性之间可以相互导出

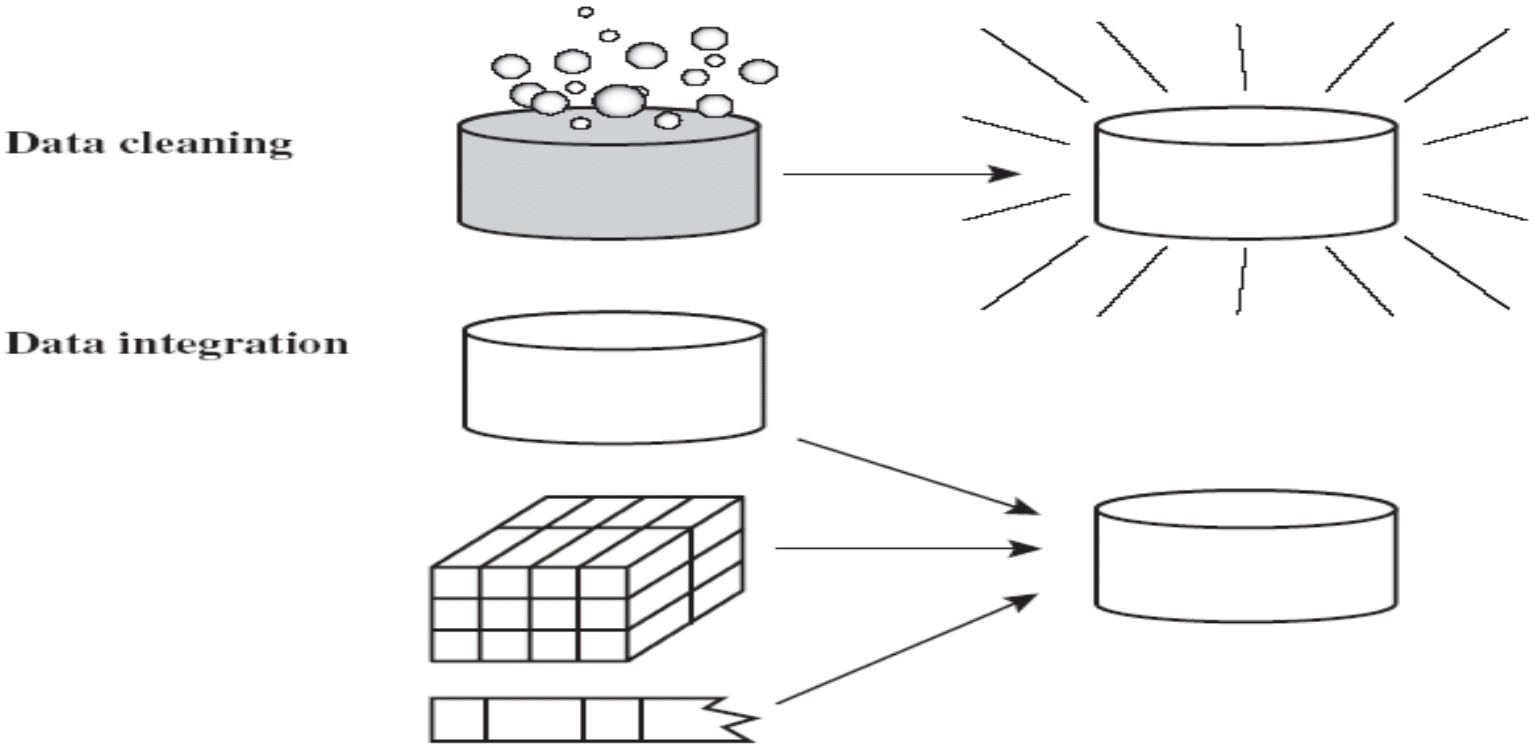
## 数据错误的危害性

- 高昂的操作费用
  - 在进行数据分析和挖掘时，更难得到预期的结果，增加了操作代价
- 糟糕的决策制定
  - 根据错误数据得出的结论可能也是错误的
- 组织的不信任
  - 数据错误导致决策错误，从而失去组织信任
- 分散管理的注意力
  - 在管理决策时，需要耗费更多精力去应对数据错误

## 数据预处理的主要任务

- 数据清洗（Data Cleaning）
  - 填充缺失值、平滑噪声数据、识别或删除离群点、解决数据的不一致性
- 数据集成（Data Integration）
  - 集成多个数据源、数据库或文件的数据
- 数据归约（Data Reduction）
  - 简化数据，同时保留尽可能多的信息，以产生相同或相似的结果
- 数据变换与离散化（Data Transformation and Discretization）
  - 通过归一化、离散化等处理，便于模型进行分析

## 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
attributes
A1 A2 A3 ... A126
transactions
T1
T2
T3
T4
...
T2000
attributes
A1 A3 ... A115
transactions
T1
T4
...
T1456

## 总结：为什么需要数据预处理?

- 现实世界的数据一般是脏的、不完整的和不一致的。
- 数据预处理技术可以改进数据的质量，从而有助于提高后续挖掘过程的精度和性能。
- 高质量的决策必然依赖于高质量的数据，因此数据预处理是知识发现过程的重要步骤。
- 检测异常数据、尽早地调整数据并归约待分析的数据，将在决策过程中得到高回报。

# 目　录

CONTENTS

## 现实生活中的“脏”数据

- 不完整数据：缺少关键属性、属性值缺失，缺少详细信息
  - 如：*Occupation*=“ ” (属性值缺失）
- 噪声：包含噪声数据、异常值、错误值、离群点
  - 如：Salary=“0.01”（错误值）
- 不一致数据，例如：
  - Age=“42”，Birthday=“2010-03-07”
  - 评分“1，2，3”和“A，B，C”混用
  - 重复数据记录之间的不一致
- 人为导致的错误
  - 不恰当的默认值，如：Birthday=“Jan. 1”，Gender=“Male”

## 缺失值

- 现实数据往往包含大量缺失值
  - 如：销售数据中往往没有用户的收入信息
- 产生缺失值的原因
  - 设备采集错误或漏采样
  - 与其他记录产生冲突从而被丢弃
  - 用户没有输入（如，没有理解字段含义）
  - 某些数据在用户录入时不重要
  - 没有对数据的变更进行跟踪和记录

- 处理办法：通常采用某种方法对缺失值进行估计和推断

## 如何处理缺失值?

- 忽略元组，即将该整条记录丢弃
  - 一般在缺少类别标签（分类任务）时这样做
  - 缺点：不能使用元组的剩余属性值
- 人工填充缺失值：耗时耗力☹
- 自动填充缺失值，如：
  - 使用全局常量填充，如：unknown
  - 使用属性的中心度量（如均值或中位数）填充
  - 使用与给定元组属于同一类的所有元组的属性均值或中位数填充
  - 使用最可能的值填充（如采用回归方法推断缺失值）
- 在特定场合，缺失值并不意味着数据错误
  - 如，申请信用卡时，有的用户可能本来就没有驾照信息。

## 噪声数据

- 噪声：被测量的变量的随机误差或方差
- 不正确的噪声数据的来源
  - 数据采集设备的错误
  - 数据输入错误
  - 数据传输错误
  - 技术缺陷
  - 命名规范的不一致性
- 其他数据问题
  - 重复的记录
  - 不完整数据
  - 不一致数据

## 如何处理噪声数据

- 分箱（Binning）：根据数据的“近邻”（即周围的值）来光滑有序数据值
  - 首先对数据进行排序，并划分到等频的箱子中
  - 然后使用每个箱子的均值、中位数或边界替换箱中的每一个值
- 回归
  - 使用回归函数拟合数据来光滑数据
- 聚类
  - 对数据进行聚类，然后删除离群点
- 将计算机与人工检查相结合
  - 先使用计算机探测可能是噪声的数据，再交由人类专家进行筛查。

## 分箱法光滑数据示例

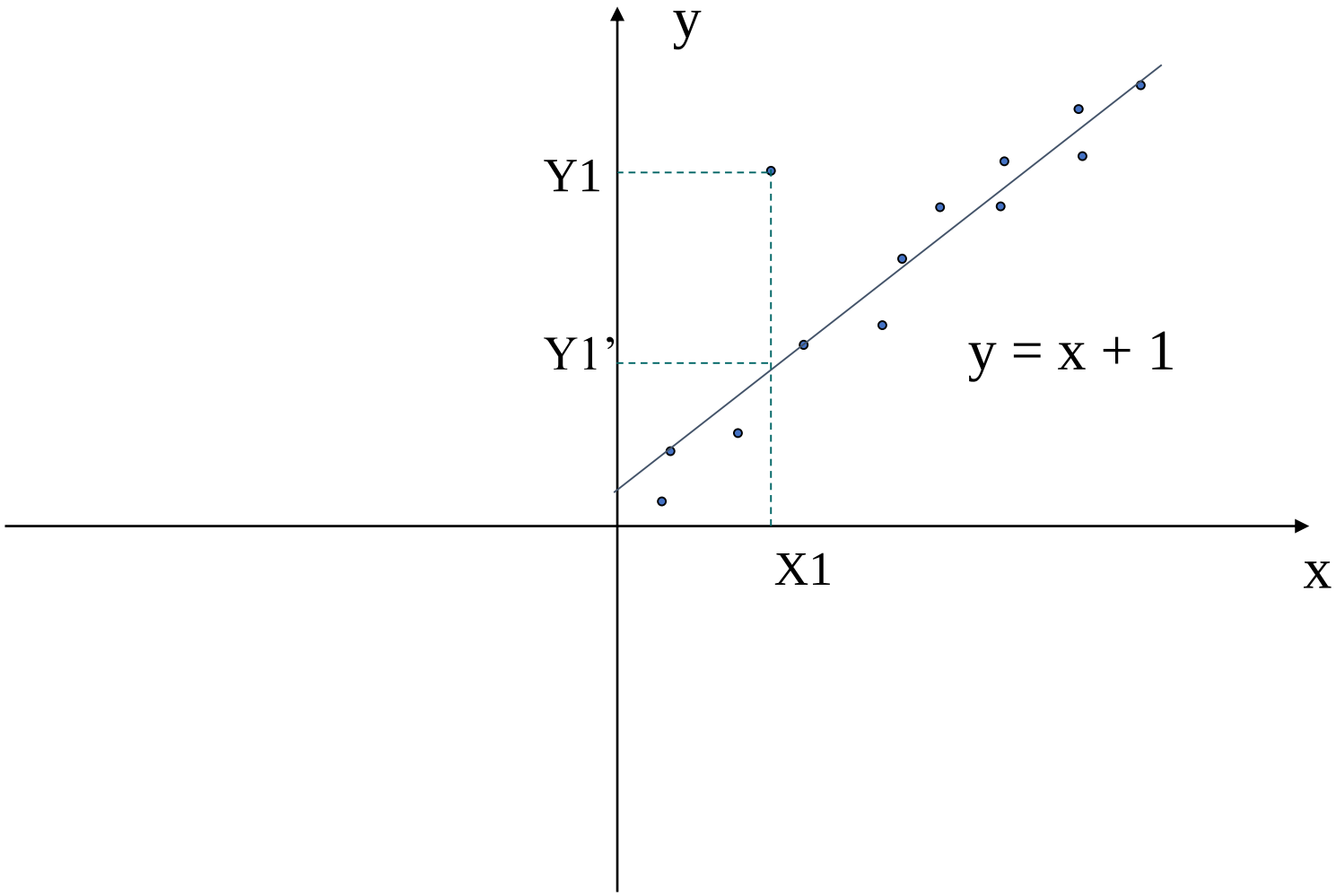
- 排序后的数据: 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* 划分为等频的箱:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34

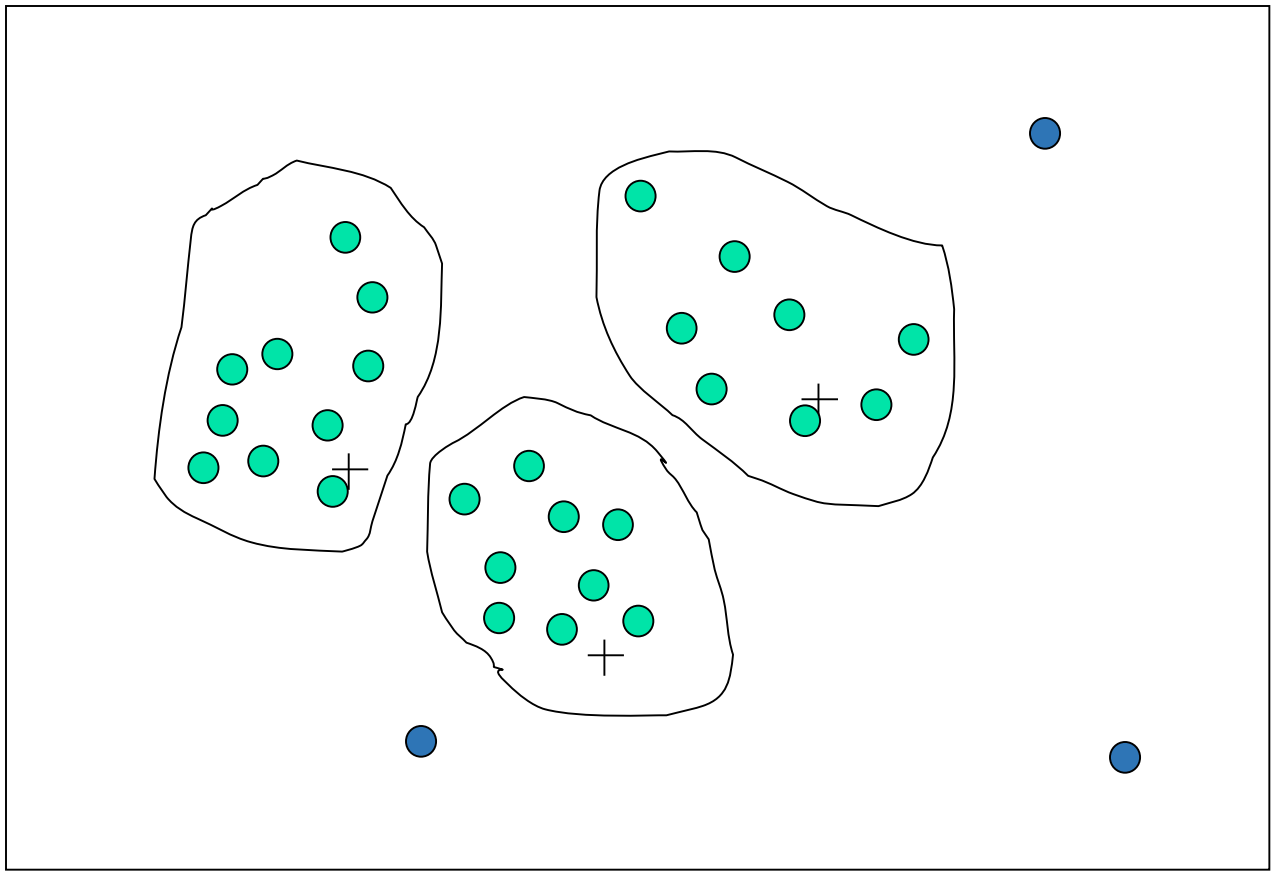
* 用箱均值光滑:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29

* 用箱边界光滑:
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

## 回归示例



## 聚类示例



# 目　录

CONTENTS

## 概述

- 数据集成
  - 将多个来源的数据合并，并保持数据的一致性，减少冗余
- 模式（Schema）集成
  - 集成不同数据源的元数据，如: A.cust_id=B.cust_number
- 实体识别问题
  - 从多个数据源识别相同实体，如：Bill Clinton = William Clinton
- 数据值冲突的检测与处理
  - 相同的实体，由于数据源不同导致的属性值不同
  - 可能原因：不同的表示方法，不同的尺度和单位，如：米和英尺

## 数据冗余

- 冗余：某个属性能由另一个或一组属性“导出”，通常发生于集成多个数据源时，例如：
  - 不同数据源都包含某个属性，但属性名称不同
  - 某个属性可由其他属性导出，如月度净收入可由月度收入和支出数据导出
- 如何检测冗余
  - 给定两个标称属性，可使用$\chi^2$（卡方）检验
  - 给定两个数值属性，可使用相关分析（Correlation Analysis）和协方差分析（Covariance Analysis）

- 合理地集成数据，并避免数据的冗余和不一致性，有助于提高后续数据分析与挖掘的速度和质量

## 标称数据的相关检验

- 给定标称属性A和B，假设A有$c$个不同的取值$a_1,a_2,\ldots,a_c$，B有$r$个不同的取值$b_1,b_2,\ldots,b_r$，则$\chi^2$（卡方）检验

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

- 其中，$o_{ij}=count(A=a_i, B=b_j)$，即实际观测值，$e_{ij}$为期望值，计算公式如下：

$$e_{ij}=\frac{count(A=a_i)\times count(B=b_j)}{count(All)}$$

- $\chi^2$值越大，则两个属性相关性越高
- 实际计数值与期望值差异越大，对$\chi^2$值的贡献也越大
- 相关性并不意味着因果性（即A导致B发生）
  - 如：城市的医院数目跟盗窃案数目相关，但这两者并没有因果性，而是共同取决于第三个因素——人口数量

## 卡方检验示例

注：括号中为期望频率$e_{ij}$

| | 喜欢音乐 | 不喜欢音乐 | 合计 |
|---|---|---|---|
| 喜欢小说 | 250(90) | 200(360) | 450 |
| 不喜欢小说 | 50(210) | 1000(840) | 1050 |
| 合计 | 300 | 1200 | 1500 |

- 则$\chi^2$（卡方）检验值：

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

- 对于2x2的表格，卡方检验的自由度(2−1)(2−1)=1，在0.001置信水平下，拒绝假设的值为10.828（通过查表得到）
- 由于$\chi^2$值（507.93）远大于该值，因此可认为属性A和B是强关联的。

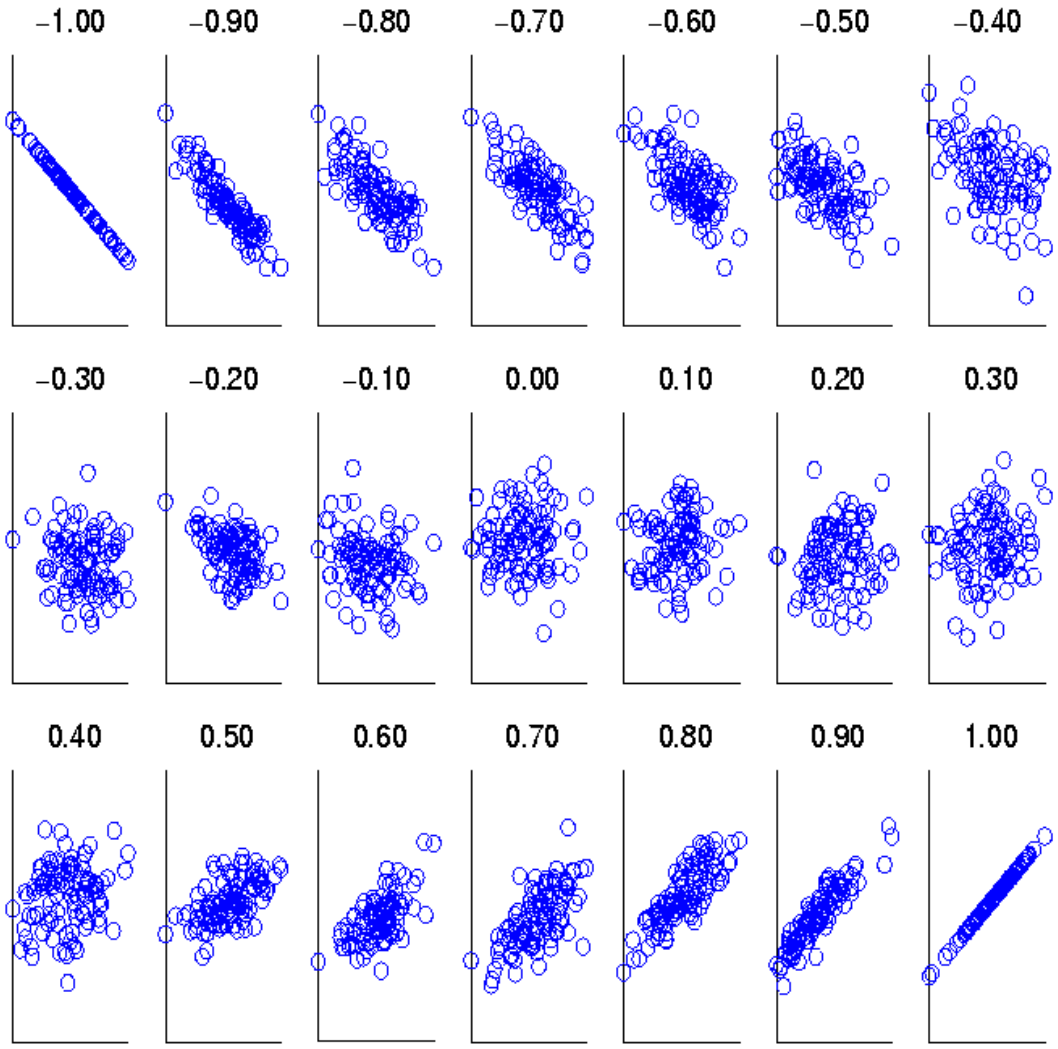
## 数值数据的相关检验

- 通常使用**相关系数**，如Pearson积矩系数（Pearson's product moment coefficient）$r_{A,B}$：

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

- 其中，$n$为元组的数量，$\bar{A}$ 和$\bar{B}$表示A和B的均值，$\sigma_A$和$\sigma_B$表示A和B的标准差。
- 若$r_{A,B} > 0$，表示A和B正相关，值越大，相关性越强
- 若$r_{A,B} = 0$，表示A和B不相关
- 若$r_{A,B} < 0$，表示A和B负相关，值越小，相关性越强

## 数值数据的相关检验——可视化的方式




相关性散点图

（左上到右下，相关性从–1到1）

## 数值数据的相关检验——协方差

- 协方差（Covariance）的定义

$$Cov(A,B) = E((A-\overline{A})(B-\overline{B})) = \frac{\sum_{i=1}^{n}(a_i-\overline{A})(b_i-\overline{B})}{n}$$

- 协方差与相关系数$r_{A,B}$的关系:

$$r_{A,B} = \frac{Cov(A,B)}{\sigma_A \sigma_B}$$

- 若$Cov(A, B) > 0$，表示A和B倾向于同时大于或小于其均值
- 若$Cov(A, B) < 0$，表示当A大于其均值时，B倾向于小于其均值
- 若A和B独立，则有$Cov(A, B) = 0$。然而，反之不成立。

## 数值数据的相关检验——协方差示例

$$Cov(A,B) = E((A-\overline{A})(B-\overline{B})) = \frac{\sum_{i=1}^{n}(a_i-\overline{A})(b_i-\overline{B})}{n}$$

- 根据协方差的公式，其计算可简化为：

$$Cov(A,B) = E(AB) - E(A)E(B)$$

- 假设两只股票A和B一周的数据如下：
- (2, 5), (3, 8), (5, 10), (4, 11), (6, 14)
- 问题：若两只股票受相同行业趋势影响，它们的价格是否会同时涨跌？
  - E(A)=(2 + 3 + 5 + 4 + 6) / 5=20/5=4
  - E(B)=(5 + 8 + 10 + 11 + 14) / 5 = 48/5 = 9.6
  - Cov(A,B) = (2×5 + 3×8 + 5×10 + 4×11 + 6×14)/5 – 4×9.6 = 4
- 由于Cov(A,B)显著大于0，因此得出结论A和B会同时涨跌。

# 目　录

CONTENTS

## 数据规约的基本概念

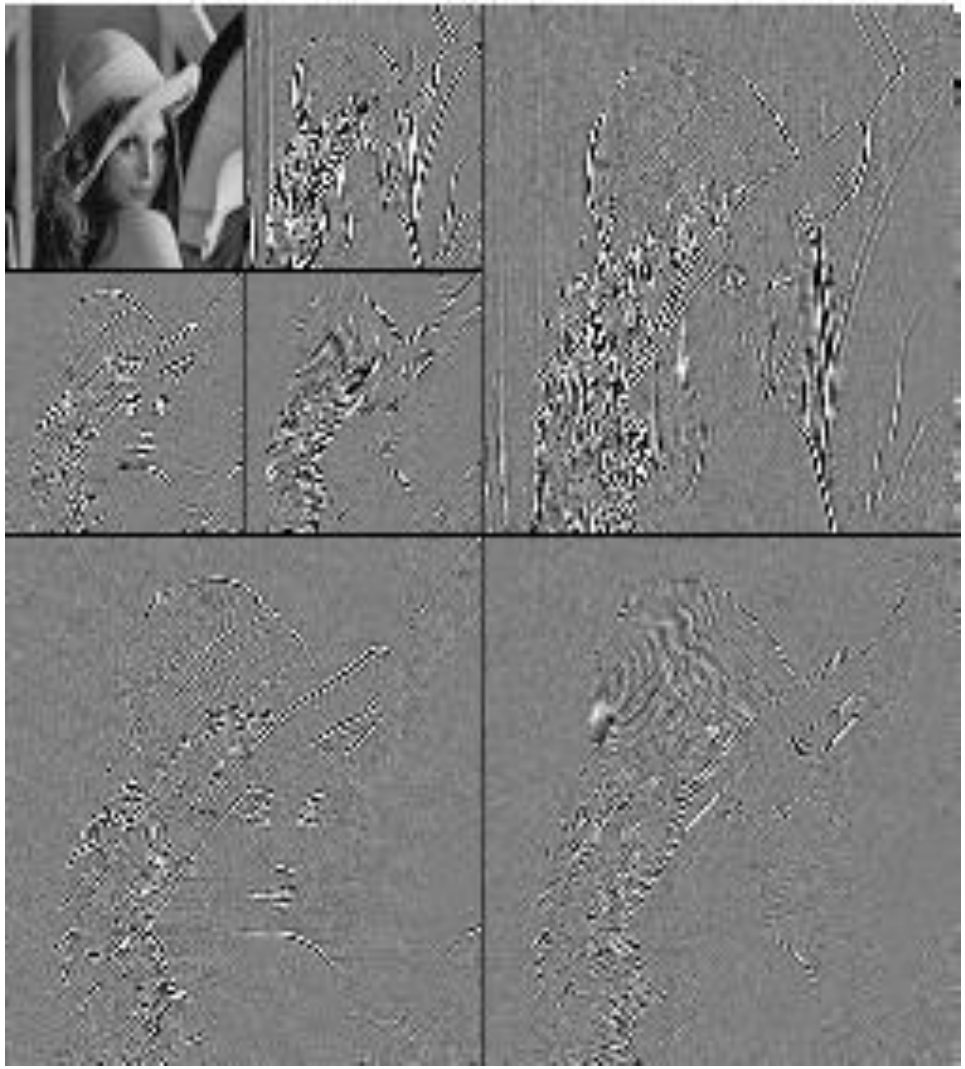
- 什么是数据规约（Data Reduction）？
  - 数据规约是指得到数据集的简化表示，它比原始数据集小很多，但仍接近于保持原始数据的完整性
- 为什么需要数据规约？
  - 原始数据集可能包含海量的数据，直接处理将耗费大量时间
- 数据规约策略
  - 维度规约（Dimensionality Reduction），即减少不重要的属性
    - 小波变换（Wavelet Transforms）
    - 主成分分析（Principal Components Analysis，PCA）
    - 属性子集选择（Feature Subset Selection）
  - 数量规约（Numerosity Reduction）
    - 回归和对数-线性模型
    - 直方图、聚类、采样
  - 数据压缩（Data Compression）

## 4.1 维度规约

- 维度灾难（Curse of Dimensionality)
  - 随着维度的增加，数据变得非常稀疏，给聚类、离群点检测等方法带来困难
  - 子空间的可能组合数随着维度的增加呈指数增长
- 维度规约的目的
  - 避免维度灾难
  - 去除不相关的特征，减少噪声
  - 减少数据挖掘所需的时间和空间
  - 让可视化变得更加容易
- 维度规约的典型技术
  - 小波变换
  - 主成分分析
  - 属性子集选择

## 小波变换（Wavelet Transform）

- 基本原理：将信号分解为不同频率的子频带（subbands）
- 经过变换之后的数据，能够在不同的分辨率下保留样本之间的相对距离
- 在变换之后的数据上进行聚类，效果更加显著
- 常用于图像压缩

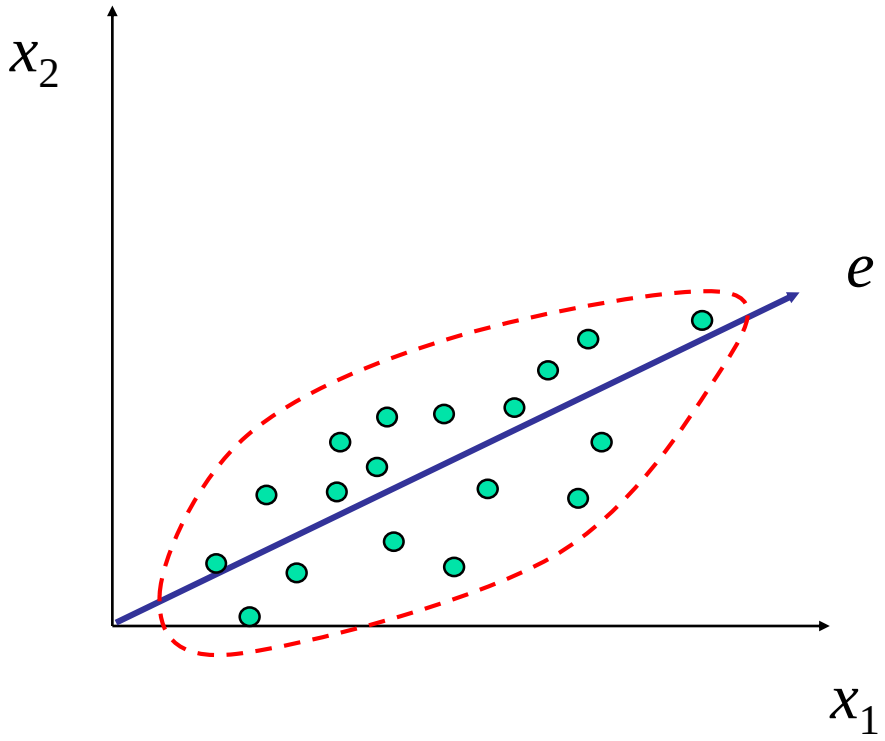


## 离散小波变换（Discrete Wavelet Transform，DWT）

- 什么是DWT？
  - 一种线性信号处理技术，可将$n$维数据向量$\boldsymbol{X}$变换为不同的$n$维数值小波系数向量$\boldsymbol{X}'$。
- DWT如何实现数据压缩？
  - 变换之后的数据可以截短，即仅保留一部分最强的小波系数，就能保留近似的压缩数据。
- DWT的优点
  - 利用数据的稀疏性，在变换之后的小波空间进行计算非常高效
  - 局部性好，有助于保留局部细节
  - 能用于消除噪声，且不会光滑掉数据的主要特征

# 主成分分析（Principal Component Analysis，PCA）

- PCA原理
  - 搜索 $k$ 个最能代表数据的$n$维正交向量，其中 $k \le n$，从而将原始的 $n$ 维数据投影到一个小得多的$k$维空间，并保留尽可能多的方差信息，从而实现维度规约



## 主成分分析（Principal Component Analysis，PCA）

- PCA步骤
  - 归一化输入数据，使得每个属性都落入相同的区间
  - 计算 $k$ 个单位正交向量作为主成分
  - 将原始空间的每个输入数据表示为$k$个主成分的线性组合
  - 对主成分按“重要性”或强度降序排列。每个主成分可看作是新的坐标轴，其重要性对应该坐标轴显示数据的方差
  - 去掉较弱的主成分（即方差较小的部分），只保留最强的主成分，从而实现维度规约

- PCA一般仅用于数值数据

## 属性子集选择（Attribute Subset Selection）

- 原始数据可能包含上百个属性，其中大部分可能与分析任务无关或是冗余的
- 属性子集选择
  - 通过删除不相关或冗余的属性减少数据量
  - 目标：找到最小属性子集，使得数据类的概率分布尽可能地接近使用所有属性得到的原始分布
  - 更少的属性使得模型更易于理解，增加了模型可解释性

- 冗余属性：不同属性的大部分或全部信息重复
  - 例如：商品售价和税额
- 不相关属性：对分析挖掘任务不提供任何有用信息的属性
  - 例如：学生的学号通常对于预测其GPA没有任何帮助

## 属性子集选择方法

- 穷举法：对于$d$维属性，存在$2^d$种属性组合 ☹

## 启发式算法

- 逐步前向选择
  - 从空属性集开始，每次选择一个最好的属性，并加入属性集中
- 逐步后向删除
  - 从整个属性集开始，每次删除一个最差的属性
- 逐步前向选择和逐步后向删除的组合
  - 将上述两种方法结合在一起，每一步选择一个最好的属性，并在剩余属性中删除一个最差的属性
- 决策树归纳
  - 使用决策树算法对属性排序，并选择“最好”的属性

## 属性构造（特征生成）

- 目的
  - 构造新的属性（特征），使其能够更好地捕捉数据集中的重要信息，一般需要结合领域知识。

- **常用方法**
- 属性抽取
  - 依赖于领域知识
- 将数据映射到新的空间
  - 如：傅里叶变换、小波变换
- 属性合并
- 数据离散化

## 4.2 数量规约（Numerosity Reduction）

- 数量规约的目的：用替代的、较小的数据表示形式替换原始数据，以达到减少数据量的目的

- 参数方法（Parametric Methods）
  - 假设数据是通过某种模型生成的，通过估计模型参数，从而只需要存储模型参数，不需要存储实际数据（离群点除外）
  - 如：对数线性模型（Log-Linear Model）

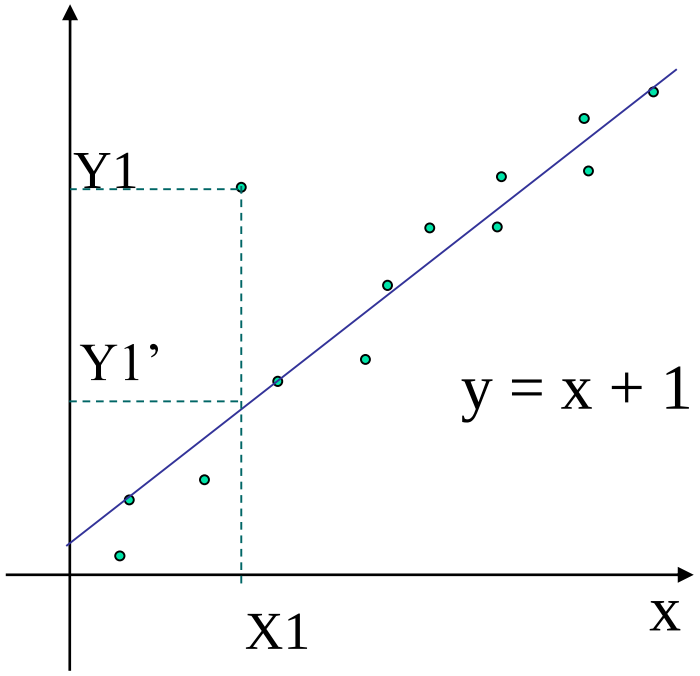
- 非参数方法（Non-Parametric Methods）
  - 不对数据分布做任何模型假设
  - 如：直方图、聚类、抽样、等

## 参数方法——回归与对数线性回归

- 线性回归（Linear Regression）
  - 对数据建模，使其拟合到一条直线
  - 通常使用最小二乘法进行拟合
- 多元线性回归（Multiple Regression）
  - 简单线性回归的扩展，支持两个或多个自变量的线性函数对因变量*y*建模
- 对数线性模型（Log-Linear Model）
  - 用于估计离散多维度概率分布

## 回归分析

- 回归分析是一类方法的总称，其主要用于建模一个因变量（也叫反应变量）与一个或多个自变量（也叫解释变量或预测因子）之间的关系
- 通过对模型参数进行估计，从而“最好”地拟合数据
- 常用最小二乘法进行参数估计
- 常用于预测任务，如时间序列预测、假设检验、统计推断、因果建模等。

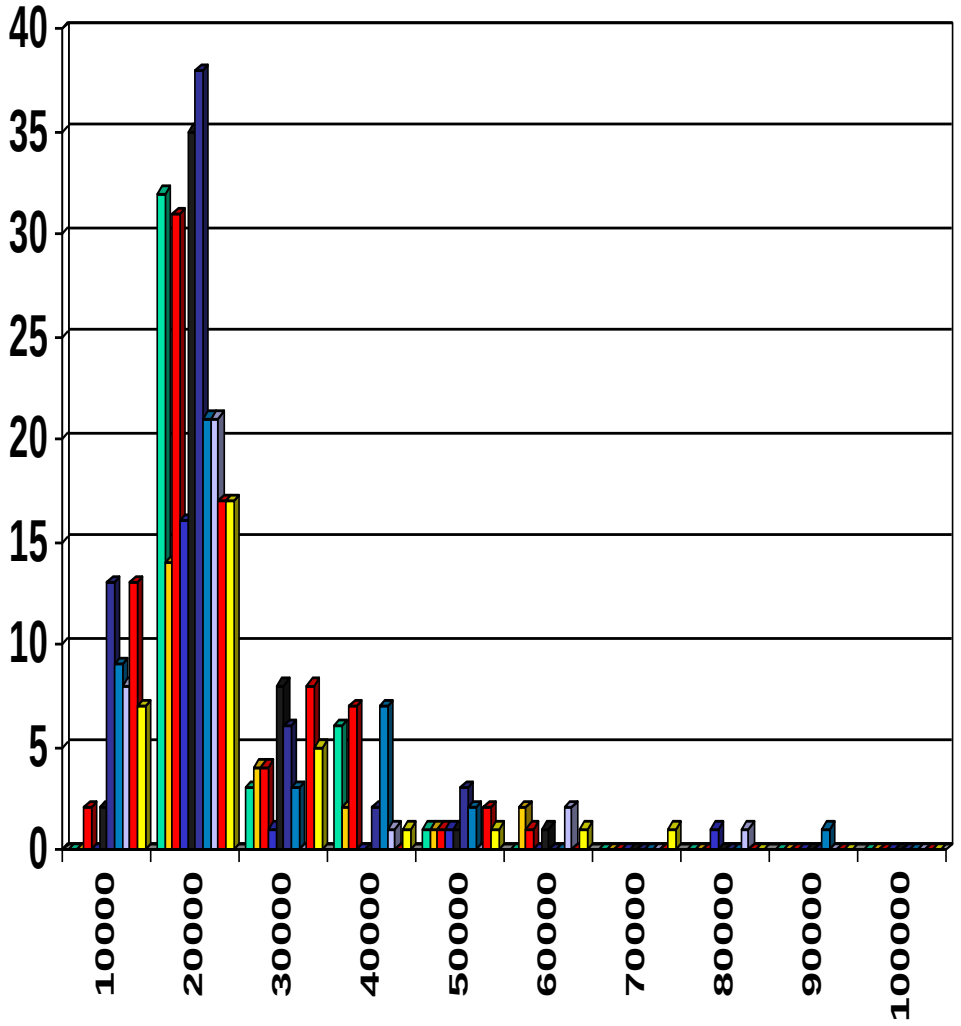


## 回归分析与对数线性模型

- 线性回归：$y=wx+b$
  - 包含2个回归系数：w对应斜率，b对应截距
  - 使用最小二乘法拟合，即：$w^*,b^*=\underset{w,b}{\operatorname{argmin}}\sum_{i=1}^{m}(wx_i+b-y_i)^2$
- 多元线性回归：$y=b_0+b_1x_1+b_2x_2+\ldots+b_nx_n$
  - 包含n个自变量，n+1个回归系数$b_0, b_1, \ldots, b_n$
  - 可用于表示许多非线性函数
- 对数线性回归
  - 用于近似离散的多维概率分布
  - 使用离散属性集合的一个较小子集，估计多维离散空间中每个点的概率
  - 也可以用于维度规约和数据平滑

## 数量规约——直方图分析

- 使用分箱来近似数据分布
- 将数据划分到不同的桶，并存储桶内的平均值
- 划分规则
  - 等宽划分：每个桶的宽度区间相同
  - 等频划分：每个桶的频率大致相等



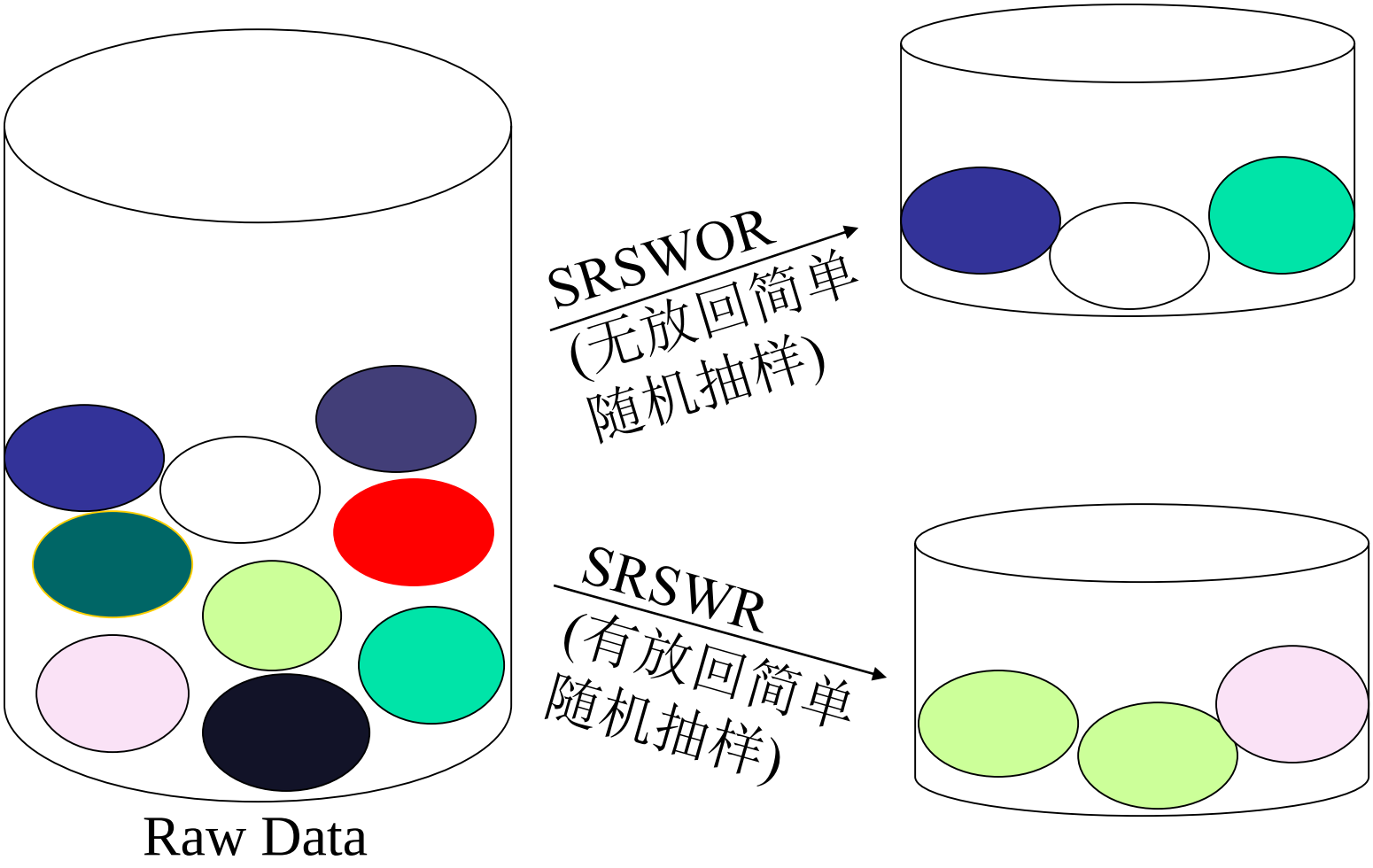
## 数量规约——聚类

- 将数据元组看作对象，将对象划分为群或簇，使得一个簇中的对象相互“相似”，而与其他簇“相异”
- 使用簇中心和直径来表示数据对象，从而达到规约目的
- 可包含层次化聚类结构，此时可用多维索引树结构表示
- 有大量的聚类算法可供选择，如K-means、DBSCAN等

## 数量规约——抽样

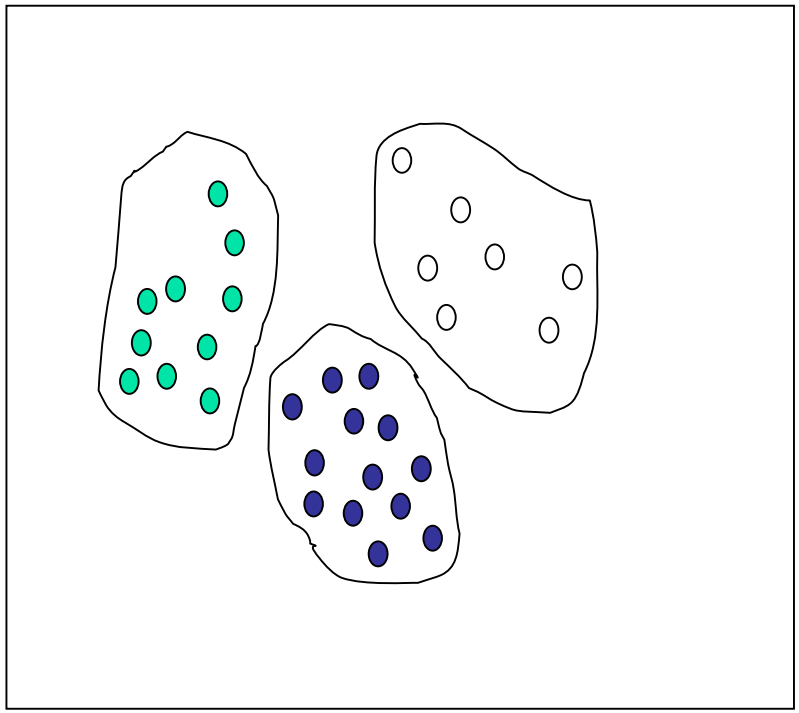
- 使用数据中小得多的随机样本（子集）表示大型数据集
- 假设原始数据集$D$中包含$N$个样本，常用抽样方法如下：
- 无放回简单随机抽样（SRSWOR）
  - 从 $D$ 的 $N$ 个样本中抽取 $s$ 个样本($s < N$)，任意样本被抽取的概率均为$1/N$
- 有放回简单随机抽样（SRSWR）
  - 类似于SRSWOR，不同之处在于一个样本被抽取后，又被放回$D$，一遍它可以被再次抽取
- 簇抽样（Cluster Sampling）
  - 将$D$中的样本分为$M$个不相交的簇，对每个簇进行简单随机抽样
- 分层抽样（Stratified Sampling）
  - 将 $D$ 划分为不相交的“层”，然后对每一层进行简单随机抽样，例如：根据顾客的年龄段分层进行抽样

## 抽样示例——有放回抽样 vs 无放回抽样

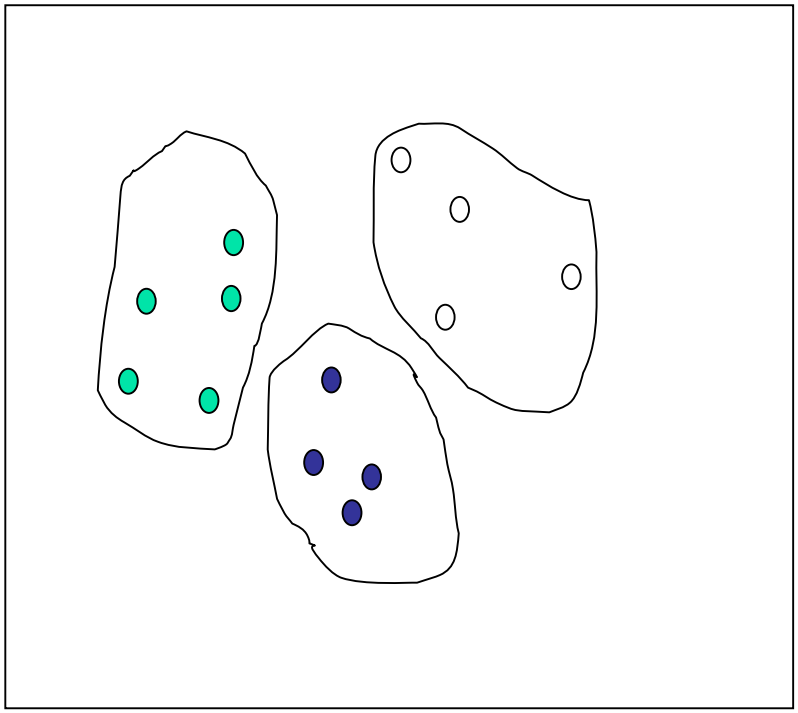


## 抽样示例——簇抽样和分层抽样

原始数据



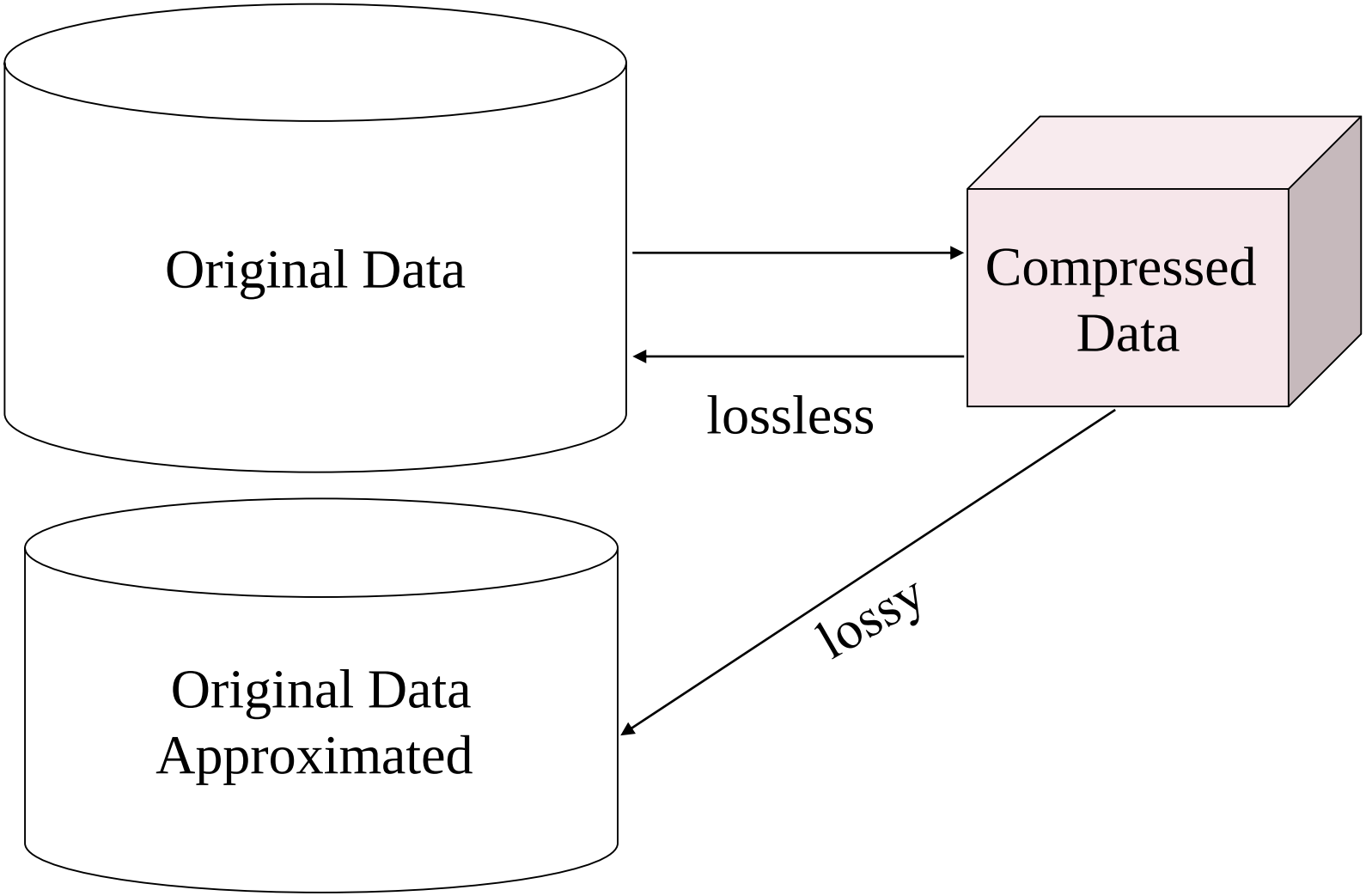
簇抽样/分层抽样



## 4.3 数据压缩（Data Compression）

- 无损压缩：能够从压缩后的数据重构原始数据，而不损失任何信息
- 无损压缩：只能通过压缩后的数据近似重构原始数据
- 字符串压缩
  - 有成熟的理论和大量的优秀算法
  - 通常能提供“无损”压缩，但压缩数据上只能做有限的操作
- 音频/视频压缩
  - 通常是有损的
  - 一些算法可以重构原始数据的部分片段

## 4.3 数据压缩（Data Compression）



# 目　录

CONTENTS

## 5.1 数据变换

- 数据变换的定义
  - 通过某种函数映射，将整个数据集的某个属性变换为新的值
- 目的：使得数据分析和挖掘过程更加有效
- 数据变换方法
  - 平滑（Smoothing）：去掉数据中的噪声
  - 属性构造（Attribute/feature construction）
    - 由给定属性构造新的属性
  - 聚集（Aggregation）：对数据进行汇总或聚集
  - 归一化：把属性数据按比例缩放，使其落入指定小区间
    - 最小-最大（min-max）归一化
    - 零均值（z-score）归一化
    - 小数定标（decimal scaling）规一化
  - 离散化（Discretization）：概念分层，如日期分为年、月、日

## 归一化（Normalization）

- 最小-最大（min-max）归一化
  - 假设需要将属性A归一化到[new_min$_A$, new_max$_A$]区间，则：

$$v' = \frac{v - min_A}{max_A}(new\_max_A - new\_min_A) + new\_min_A$$

  - 例如：将原始收入数据从[$12,000, $98,000]归一化到[0.0, 1.0]，则$73,000经过归一化之后为0.716
- 零均值（z-score）归一化：归一化之后服从$N(0,1)$正太分布
  - $v' = \frac{v - \mu_A}{\sigma_A}$　　$\mu_A$：属性A的均值
  - $\sigma_A$：属性A的方差
  - 例：假设$\mu$ = 54,000, $\sigma$=16,000，则$73,600归一化之后为1.225
- 小数定标（decimal scaling）规一化

$$v' = \frac{v}{10^j}$$　其中，$j$ 是使得max(|v'|) < 1成立的最小整数

## 离散化（Discretization）

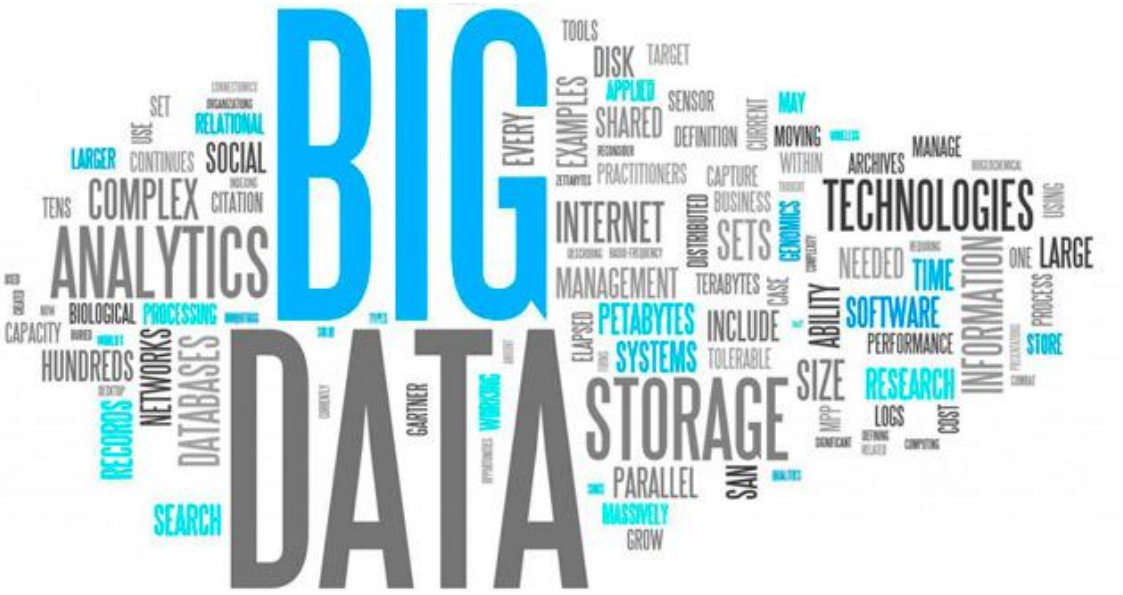
- 三种类型的属性
  - 标称属性——取值来自于无序集合，如：颜色、专业、风格等
  - 序数属性——取值来自有序集合，如：成绩等级、职称等
  - 数值属性——取实数值，如：整数或小数

- 离散化：将连续属性值划分到离散的区间
  - 区间标签可用于替代真实属性值
  - 减少数据规模
  - 无监督方法 vs 有监督方法
  - 基于分裂（自顶向下） vs 基于合并（自底向上）
  - 可以在同一个属性上递归执行离散化操作
  - 离散化之后的数据可用于后续分析，如分类问题

## 离散化——常用方法

- 分箱法
  - 采用自顶向下分裂策略，无监督方法
  - 如：等宽分箱、等频分箱
- 直方图分析
  - 与分箱法类似
- 聚类分析
  - 将属性划分为簇，用簇中心代替
  - 无监督，自顶向下或自底向上
- 决策树分析
  - 有监督，自顶向下
- 相关分析（例：基于$\chi^2$检验的方法）
  - 无监督，自底向上，递归合并邻近区间

# 谢　谢！


BIG
DATA
ANALYTICS
COMPLEX
SOCIAL
INTERNET
STORAGE
TECHNOLOGIES
INFORMATION
SEARCH
DATABASES
NETWORKS
RECORDS
HUNDREDS
CAPACITY
BIOLOGICAL
PROCESSING
MANAGEMENT
PETABYTES
SYSTEMS
INCLUDE
SOFTWARE
RESEARCH
SIZE
PARALLEL
SHARED
EXAMPLES
EVERY
TOOLS
DISK
TARGET
SENSOR
DEFINITION
MOVING
WITHIN
ARCHIVES
MANAGE
CAPTURE
BUSINESS
SETS
NEEDED
TIME
LARGE
ONE
PROCESS
STORE
PERFORMANCE
ABILITY
LOGS
COST
SAN
MASSIVELY
GROW
TERABYTES
TOLERABLE
ELAPSED
GARTNER
WORKING
RELATIONAL
CONTINUES
CITATION
TENS
LARGER
SET
USE
APPLIED
PRACTITIONERS
DISTRIBUTED
GENOMICS
MAY
CURRENT
USING