

# 数据仓库与数据挖掘

Data warehouse and data mining

丁钰

yuding@live.com

南京农业大学人工智能学院

第三章：数据预处理

# 第3章：数据预处理

---

- 数据预处理：概述
- 数据清理
- 数据集成
- 数据归约和变换
- 数据降维
- 小结



# 数据质量：为什么要对数据预处理？

---

- 数据质量的评价：多维角度
  - 准确性:正确或错误，准确或不准确
  - 完整性:未记录，不可用， ...
  - 一致性:有的修改过，有的没有，悬而未决, ...
  - 时效性: 及时更新的？
  - 可信性: 反映有多少数据是用户信赖的？
  - 可解释性: 反映数据是否容易理解？

# 数据预处理的主要任务

---

- 数据清理
  - 填充缺失值，识别/去除离群点，光滑噪音，并纠正数据不一致
- 数据集成
  - 多个数据库，数据立方体，或文件的集成
- 数据归约
  - 得到数据集的简化，它小得多，但能够产生同样的分析结果
- 数据变换
  - 规范化
  - 数据离散化和概念分层产生

# 第3章：数据预处理

---

- 数据预处理：概述
- 数据清理
- 数据集成
- 数据归约和变换
- 数据降维
- 小结



# 数据清理

- 现实世界的的数据是脏的：很多潜在的不正确的数据，比如，仪器故障，人为或计算机错误，许多传输错误
  - 数据缺失:缺少属性值, 缺少某些有趣的属性, 或仅包含聚集数据
    - e.g., 职业=“ ” (missing data)
  - 噪声:包含错误或孤立点
    - e.g., *Salary*=“-10” (an error)
  - 不一致:编码或名字存在差异, e.g.,
    - *Age*=“42”, *Birthday*=“03/07/2010”
    - 以前的等级 “1, 2, 3”, 现在等级 “A, B, C”
    - 重复记录间的差异
  - 人为有意的(e.g.,默认值)
    - Jan. 1 as everyone's birthday?

# 不完整（缺失）的数据

- 数据并不总是可用的
  - 例如，许多元组没有几个属性的记录值，如销售数据中的客户收入
- 数据缺失的原因可能是
  - 设备故障
  - 与其他记录的数据不一致，因此被删除
  - 由于误解而没有输入数据
  - 某些数据在输入时可能不被认为是重要的
  - 没有登记数据的历史或变化
- 缺失的数据可能需要被推断出来

## 如何处理缺失数据？

---

- 忽略元组:通常在类标签缺失时进行（当进行分类时）--当每个属性的缺失值百分比变化很大时，并不有效。
- 手工填写缺失数据:繁琐+不可行？
- 自动填充（采用一些规则）
  - 一个全局常量：e.g., “unknown”, a new class?!
  - 使用属性的中心度量（如均值或中位数）
  - 属于同一类别的所有样本的属性平均值: 更巧妙
  - 最可能的值: 基于推理的方法，如回归、贝叶斯公式或决策树



# 噪声数据

---

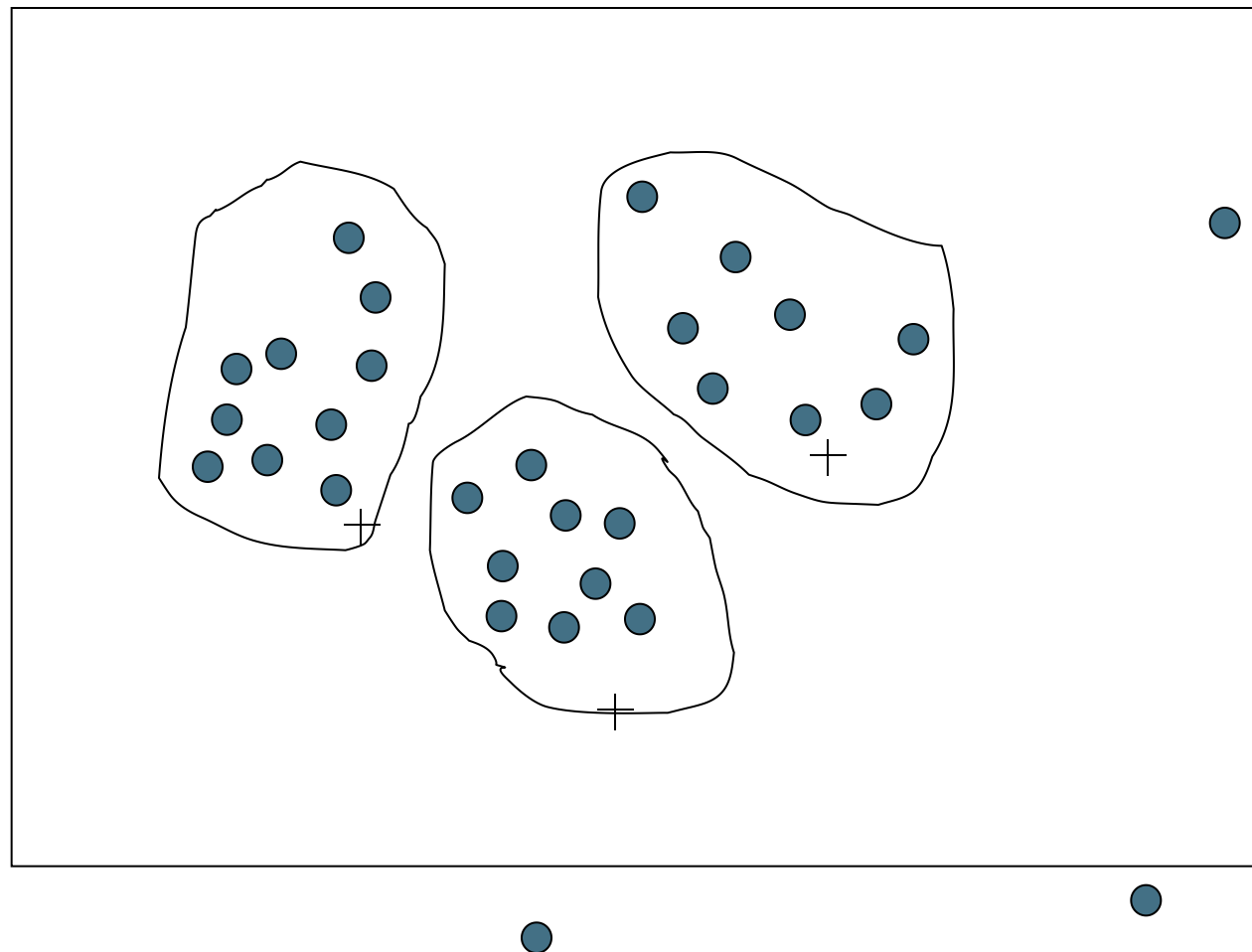
- 噪声：被测量的变量的随机误差
- 不正确的属性值可能由于
  - 错误的数据收集工具
  - 数据录入问题
  - 数据传输问题
  - 技术限制
  - 命名规则不一致
- 其他需要数据清理的问题
  - 重复记录
  - 数据不完整
  - 不一致的数据

# 如何处理噪音数据？

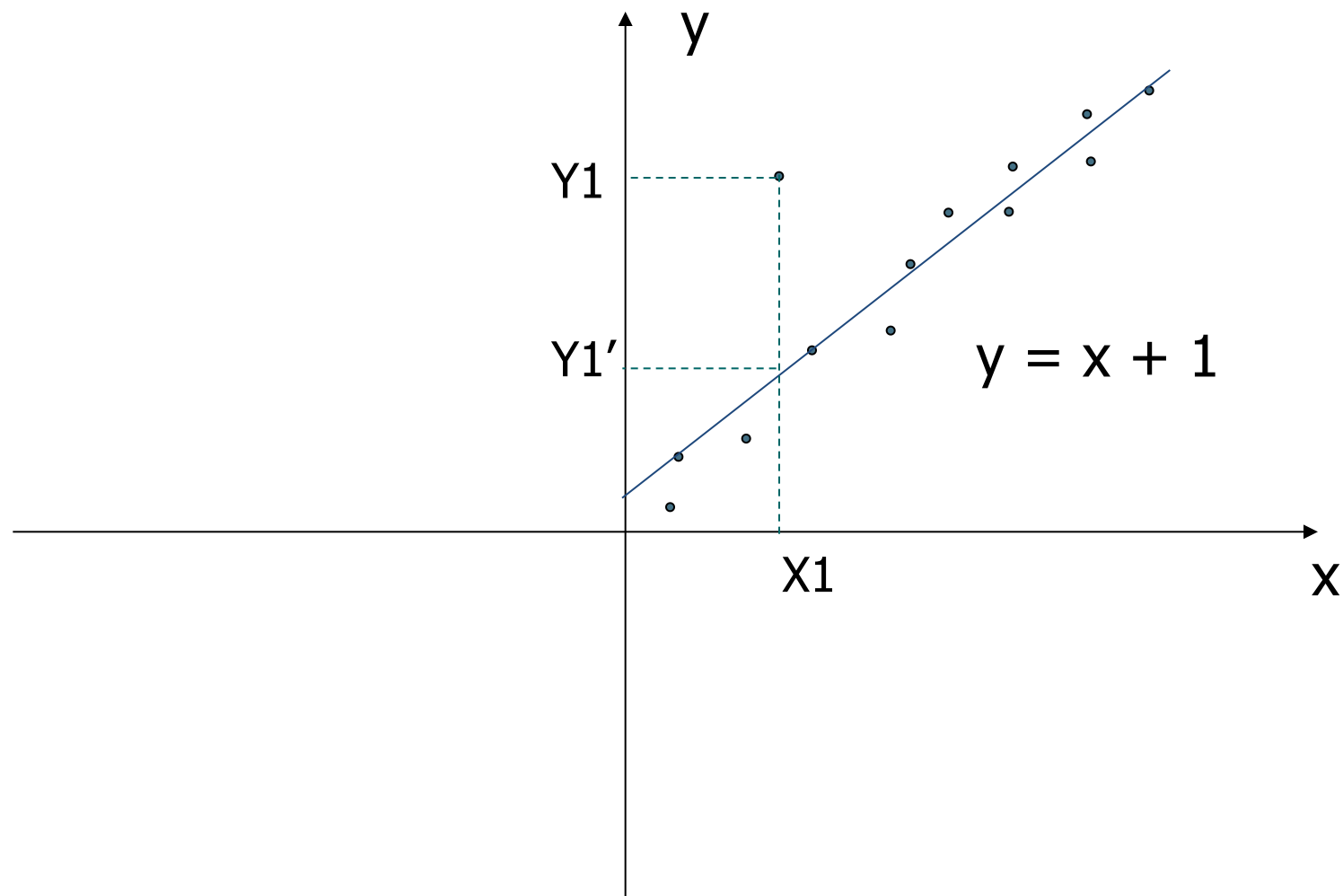
---

- 分箱：
  - 排序数据，分布到等频/等宽的箱/桶中
  - 箱均值光滑、箱中位数光滑、箱边界光滑, etc.
- 聚类
  - 检测和去除 离群点/孤立点
- 计算机和人工检查相结合
  - 人工检查可疑值 (e.g.,处理可能的离群值)
- 回归
  - 回归函数拟合数据

# 聚类分析



# Regression




# 数据清理是一个过程

- **数据偏差检测**
  - 使用元数据(数据性质的知识)(e.g.,定义域, 每个属性可接受值, 统计分布, **IQR**等)
  - 检查字段过载：新属性的定义挤进已经定义的属性的未使用部分
  - 检查唯一性规则, 连续性规则, 空值规则
  - 使用商业工具
    - 数据清洗: 使用简单的领域知识(e.g., 邮编, 拼写检查) 检查纠正错误
    - 数据审计：通过分析数据发现规则和联系发现违规者(孤立点)
- **数据迁移和集成**
  - 数据迁移工具**Data migration tools**:允许指定转换
  - **ETL**（提取/转换/加载）工具: 允许用户通过图形用户界面指定变换
- 整合两个过程
  - 两个过程迭代和交互执行(e.g., **Potter's Wheels**)

# 第3章：数据预处理

---

- 数据预处理：概述
- 数据清理
- 数据集成 
- 数据归约和变换
- 数据降维
- 小结

# 数据集成

- 数据集成：
  - 合并多个数据源中的数据，存在一个一致的数据存储中
  - 涉及3个主要问题：模式集成、冗余数据、冲突数据值
- 模式集成
  - 例如.,  $A.cust-id \equiv B.cust-\#$
  - 实体识别问题：
    - 多个数据源的真实世界的实体的识别, e.g., 南京农业大学= 南农大=NJAU
  - 集成不同来源的元数据
- 冲突数据值的检测 and 解决
  - 对真实世界的实体, 其不同来源的属性值可能不同
  - 原因: 不同的表示, 不同尺度, 公制 vs. 英制

# 数据集成中冗余数据处理

- 冗余数据（集成多个数据库时出现）
  - 对象识别：同一个属性在不同的数据库中有不同的名称
  - 衍生数据：一个属性值可由其他表的属性推导出, e.g., 年收入
- 相关分析/协方差分析
  - 可用于检测冗余数据
- 小心的集成多个来源的数据可以帮助降低和避免结果数据集中的冗余和不一致，提高数据挖掘的速度和质量



# 相关性分析（针对分类数据） 标称属性

- $\chi^2$  (chi-square) 检验。

$$\chi^2 = \sum_i^n \frac{\overset{\text{observed}}{\downarrow} (O_i - E_i)^2}{\underset{\text{expected}}{E_i}}$$

- 无效假设。这两个分布是独立的
  - $\chi^2$ 值越大，变量之间就越有可能发生关系。
- 注：相关关系并不意味着因果关系
  - # 一个城市的医院数量和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

# Chi-Square 卡方值计算: 例子

	下棋	不下棋	Sum (row)
看小说	250(90)	200(360)	450
不看小说	50(210)	1000(840)	1050
Sum(col.)	300	1200	1500

如何推导出90？

$$450/1500 * 300 = 90$$

$$e_{11} = \frac{\text{count(看小说)} * \text{count(下棋)}}{N} = \frac{450 * 300}{1500} = 90$$

- $\chi^2$  (卡方) 计算 (括号中的值为期望计值, 由两个类别的分布数据计算得到)

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

在0.001的置信度下的独立性假设是无效的



- 结果表明like\_fiction 和play\_chess 关联

# 单一变量的方差（数值数据）

- 随机变量 $X$ 的方差提供了一个衡量 $X$ 的值偏离 $X$ 的平均值或预期值的程度。

$$\sigma^2 = \text{var}(X) = E[(X - \mu)^2] = \begin{cases} \sum_x (x - \mu)^2 f(x) & \text{if } X \text{ is discrete} \\ \int_{-\infty}^{\infty} (x - \mu)^2 f(x) dx & \text{if } X \text{ is continuous} \end{cases}$$

- 其中 $\sigma^2$ 是 $X$ 的方差， $\sigma$ 称为标准差  
 $\mu$ 是平均值，而 $\mu = E[X]$ 是 $X$ 的期望值。
- 也就是说，方差是指与平均值的平方偏差的预期值
- 它也可以写成： $\sigma^2 = \text{var}(X) = E[(X - \mu)^2] = E[X^2] - \mu^2 = E[X^2] - [E(x)]^2$

# 两个变量的协方差

- 两个变量 $X_1$ 和 $X_2$ 之间的协方差

$$\sigma_{12} = E[(X_1 - \mu_1)(X_2 - \mu_2)] = E[X_1 X_2] - \mu_1 \mu_2 = E[X_1 X_2] - E[X_1]E[X_2]$$

其中， $\mu_1 = E[X_1]$ 是 $X_1$ 的各自平均值或期望值；类似地  $\mu_2 = E[X_2]$

- $X_1$ 和 $X_2$ 之间的样本协方差。
$$\hat{\sigma}_{12} = \frac{1}{n} \sum_{i=1}^n (x_{i1} - \hat{\mu}_1)(x_{i2} - \hat{\mu}_2)$$

- 样本协方差是样本方差的一般化。
$$\hat{\sigma}_{11} = \frac{1}{n} \sum_{i=1}^n (x_{i1} - \hat{\mu}_1)(x_{i1} - \hat{\mu}_1) = \frac{1}{n} \sum_{i=1}^n (x_{i1} - \hat{\mu}_1)^2 = \hat{\sigma}_1^2$$

- 正的协方差。如果 $\sigma_{12} > 0$
- 负的协方差。如果 $\sigma_{12} < 0$
- 独立的。如果 $X_1$ 和 $X_2$ 是独立的， $\sigma_{12}=0$ ，但反之则不成立。
  - 某些对随机变量可能有一个协方差0，但并不独立
  - 只有在一些额外的假设下（例如，数据遵循多变量正态分布），协方差为0才意味着独立。

# 例子： 协方差的计算

- 假设两只股票 $X_1$ 和 $X_2$ 在一周内有以下价值。
  - (2, 5), (3, 8), (5, 10), (4, 11), (6, 14)
- 问题： 如果这些股票受到相同行业趋势的影响，它们的价格会不会一起上涨或下跌？
- 协方差公式
$$\sigma_{12} = E[(X_1 - \mu_1)(X_2 - \mu_2)] = E[X_1X_2] - \mu_1\mu_2 = E[X_1X_2] - E[X_1]E[X_2]$$
- 其计算可简化为： 。 
$$\sigma_{12} = E[X_1X_2] - E[X_1]E[X_2]$$
  - $e(x_1) = (2 + 3 + 5 + 4 + 6) / 5 = 20/5 = 4$
  - $e(x_2) = (5 + 8 + 10 + 11 + 14) / 5 = 48/5 = 9.6$
  - $\sigma_{12} = (2 \times 5 + 3 \times 8 + 5 \times 10 + 4 \times 11 + 6 \times 14) / 5 - 4 \times 9.6 = 4$
- 因此，由于 $\sigma_{12} > 0$ ， $X_1$ 和 $X_2$ 一起上升。

# 两个数字变量之间的相关关系

- 两个变量 $X_1$  和 $X_2$ 之间的**相关性**是标准协方差，通过用每个变量的标准差对协方差进行归一化得到的

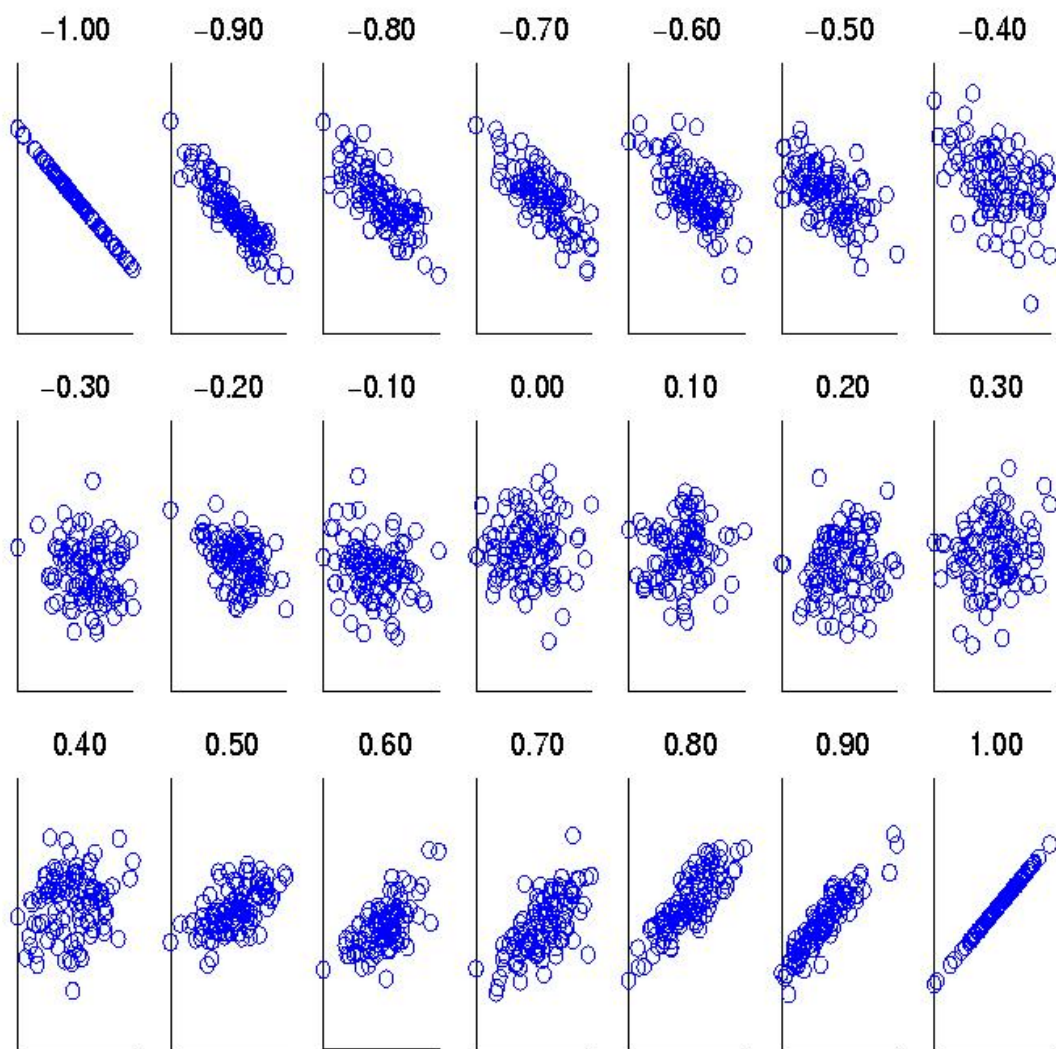
$$\rho_{12} = \frac{\sigma_{12}}{\sigma_1 \sigma_2} = \frac{\sigma_{12}}{\sqrt{\sigma_1^2 \sigma_2^2}}$$

- 两个属性 $X_1$ 和 $X_2$ 的**样本相关性**。
$$\hat{\rho}_{12} = \frac{\hat{\sigma}_{12}}{\hat{\sigma}_1 \hat{\sigma}_2} = \frac{\sum_{i=1}^n (x_{i1} - \hat{\mu}_1)(x_{i2} - \hat{\mu}_2)}{\sqrt{\sum_{i=1}^n (x_{i1} - \hat{\mu}_1)^2 \sum_{i=1}^n (x_{i2} - \hat{\mu}_2)^2}}$$

其中， $n$ 是图元的数量， $\mu_1$ 和 $\mu_2$ 是 $X_1$ 和 $X_2$ 各自的平均值， $\sigma_1$ 和 $\sigma_2$ 是 $X_1$ 和 $X_2$ 各自的标准差。

- 如果 $\rho_{12} > 0$ ：A和B是正相关的（ $X_1$ 的值随着 $X_2$ 的增加而增加）。
  - 越高，关联性越强
- 如果 $\rho_{12} = 0$ ：独立的（在与共变性讨论的相同假设下）。
- 如果 $\rho_{12} < 0$ ：负相关

# 可视化相关系数的变化



- 相关系数值范围。[-1, 1]
- 一组散点图显示了几组点和它们的相关系数从-1到1的变化情况

# 协方差矩阵

- 两个变量 $X_1$ 和 $X_2$ 的方差和协方差信息可以概括为2 X 2 协方差矩阵为

$$\begin{aligned}\Sigma &= E[(\mathbf{X} - \mu)(\mathbf{X} - \mu)^T] = E\left[\begin{pmatrix} X_1 - \mu_1 \\ X_2 - \mu_2 \end{pmatrix} (X_1 - \mu_1 \quad X_2 - \mu_2)\right] \\ &= \begin{pmatrix} E[(X_1 - \mu_1)(X_1 - \mu_1)] & E[(X_1 - \mu_1)(X_2 - \mu_2)] \\ E[(X_2 - \mu_2)(X_1 - \mu_1)] & E[(X_2 - \mu_2)(X_2 - \mu_2)] \end{pmatrix} \\ &= \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{pmatrix}\end{aligned}$$


- 将其推广到  $d$  维，可以得到：

$$D = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & \cdots & x_{dd} \end{pmatrix} \quad \Sigma = E[(\mathbf{X} - \mu)(\mathbf{X} - \mu)^T] = \begin{pmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{pmatrix}$$



# 第3章：数据预处理

---

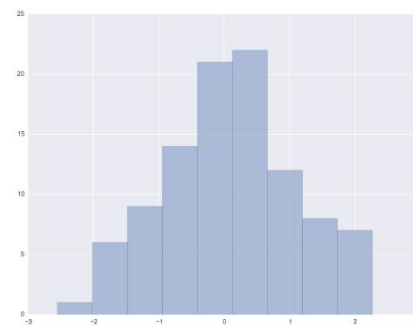
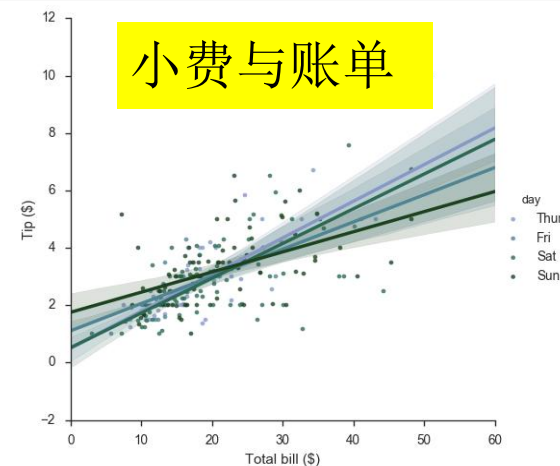
- 数据预处理：概述
- 数据清理
- 数据集成
- 数据归约和变换 
- 数据降维
- 小结

# 数据归约策略

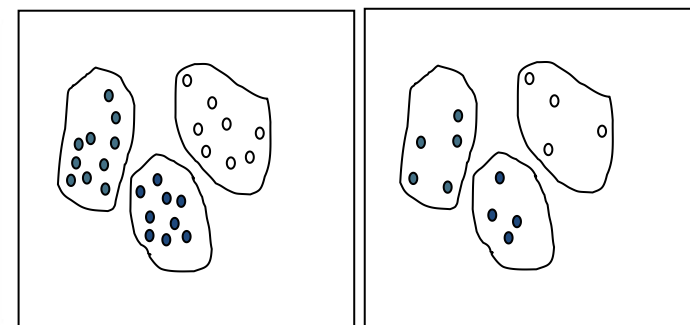
- 数据归约
  - 获得数据集的一个规约表示，小很多，接近保持原数据的完整性，使得可得到相同/几乎相同的分析结果
- 为什么需要数据归约？—数据库和数据仓库可能存储兆兆字节大小数据，在完整的数据库进行复杂数据分析需要花费大量时间.
- 数据归约策略
  - 用较小的数据形式替代原始数据
    - 回归和对数-线性模型
    - 直方图，聚类，抽样
    - 数据立方体的聚合
    - 数据压缩

# 数据归约：参数化与非参数化的方法

- 通过选择替代的、较小的数据表现形式来减少数据量
- 参数方法（例如，回归）。
  - 假设数据适合某个模型，估计模型参数，只存储参数，并丢弃数据（可能的异常值除外）。
  - 例如：对数线性模型--在 $m$ -D空间的某一点上获得的数值为适当的边际子空间上的乘积。
- 非参数方法
  - 不要假设模型
  - 主要系列：直方图、聚类、采样、...



直方图

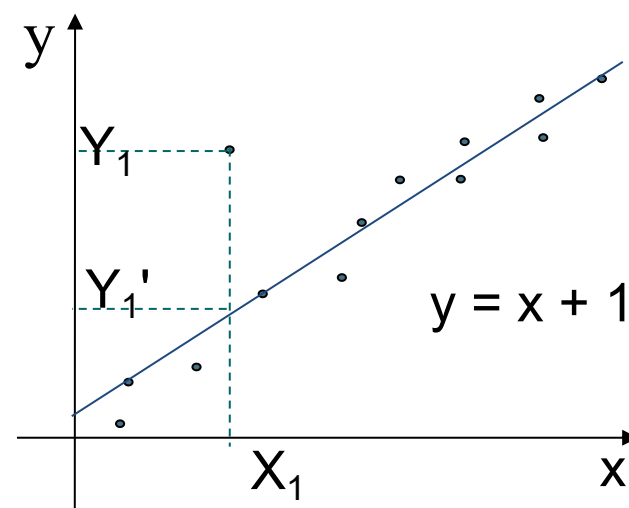


对原始数据进行聚类

分层抽样

# 参数化的数据归约：回归分析

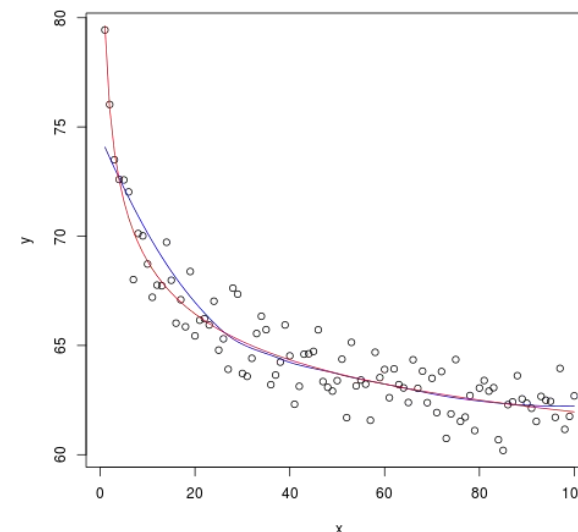
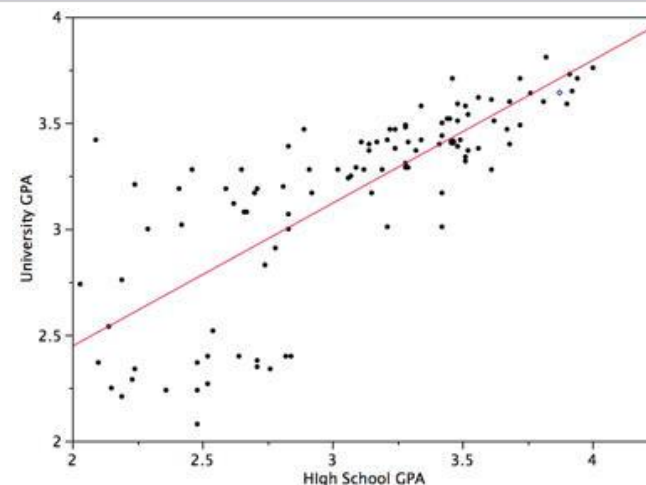
- 回归分析。对由**因变量**（也称为**响应变量**或**测量值**）和一个或多个**自变量**（也称为**解释变量**或**预测因素**）的数值数据进行建模和分析的技术的总称。
- 参数的估计是为了给数据一个“最佳拟合”。
- 最常见的是通过使用**最小二乘法**来评估最佳拟合度，但也有使用其他标准的。



- 用于预测（包括时间序列数据的预测）、推理、假设检验和因果关系的建模

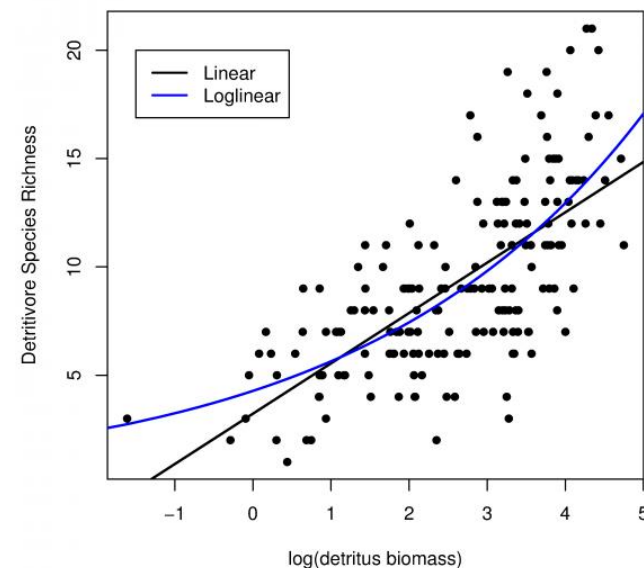
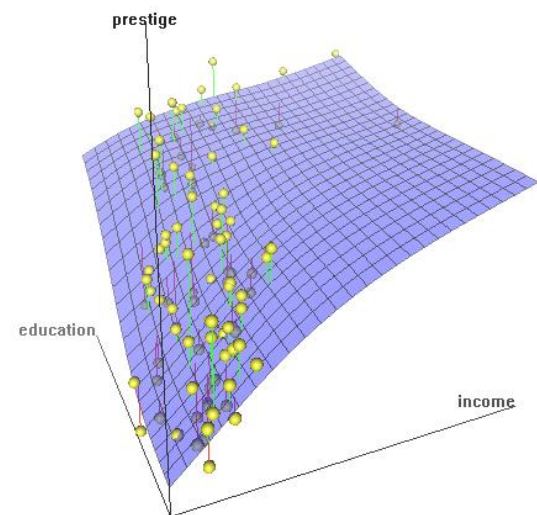
# 线性回归和多元回归

- 线性回归。  $Y = wX + b$ 
  - 拟合直线的数据模型
  - 通常使用最小平方法来拟合直线
  - 两个回归系数， $w$ 和 $b$ ，指定了这条线，并且要用已知的数据来估计。
  - 对  $Y_1, Y_2, \dots, X_1, X_2, \dots$  的已知值使用最小二乘法准则。
- 非线性回归。
  - 数据由一个函数建模，该函数是模型参数的非线性组合，取决于一个或多个自变量。
  - 用连续近似的方法来拟合数据



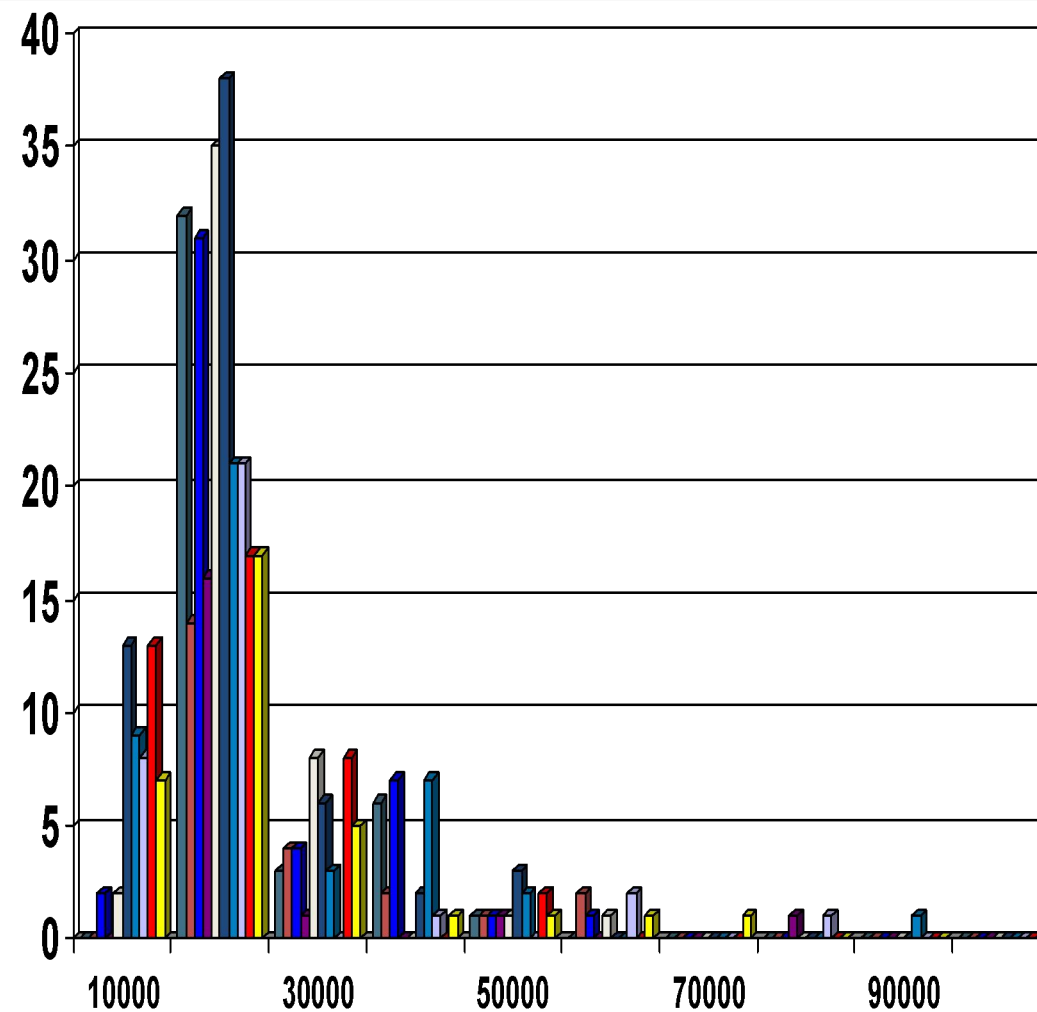
# 多重回归和对数线性模型

- 多重回归。  $Y = b_0 + b_1X_1 + b_2X_2$ 
  - 允许响应变量Y被建模为多维特征向量的线性函数
  - 许多非线性函数可以转化为上述的
- 对数线性模型。
  - 一个数学模型，其形式是一个函数，其对数是模型参数的线性组合，这使得应用（可能是多变量）线性回归成为可能。
  - 根据较小的维度组合，估计一组离散属性的多维空间中每个点（元组）的概率。
  - 有助于降维和数据平滑



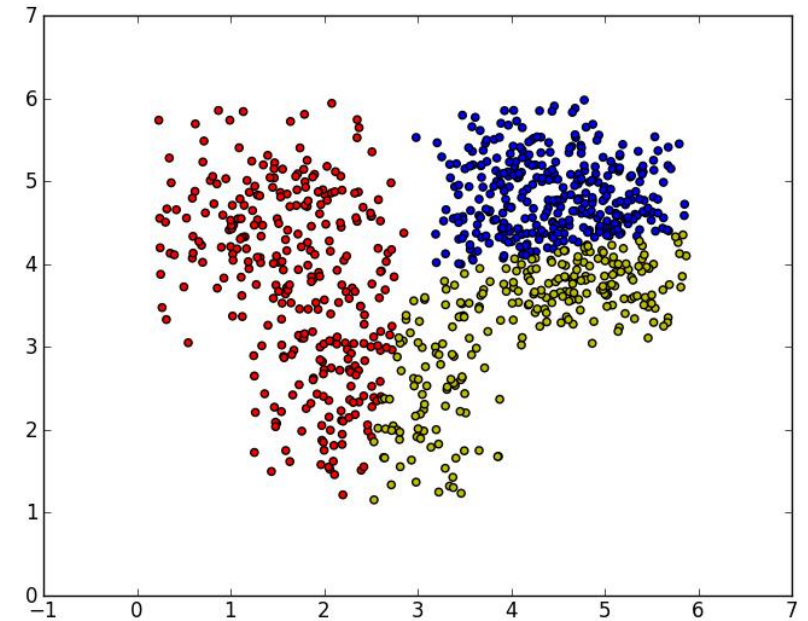
# 直方图分析

- 将数据分成桶，并存储每个桶的平均数（总和）。
- 分区规则。
  - 等宽：等桶的范围
  - 等频（或等深）。



# 聚类

- 根据相似性将数据集划分为聚类，并只存储聚类的表示（如中心点和直径）。
- 如果数据是集中的，可能非常有效，但如果数据是“模糊的”，就不一定了。
- 可以有分层聚类，并存储在多维索引树结构中
- 聚类定义和聚类算法有很多选择
- 聚类分析将在第10章中深入研究。





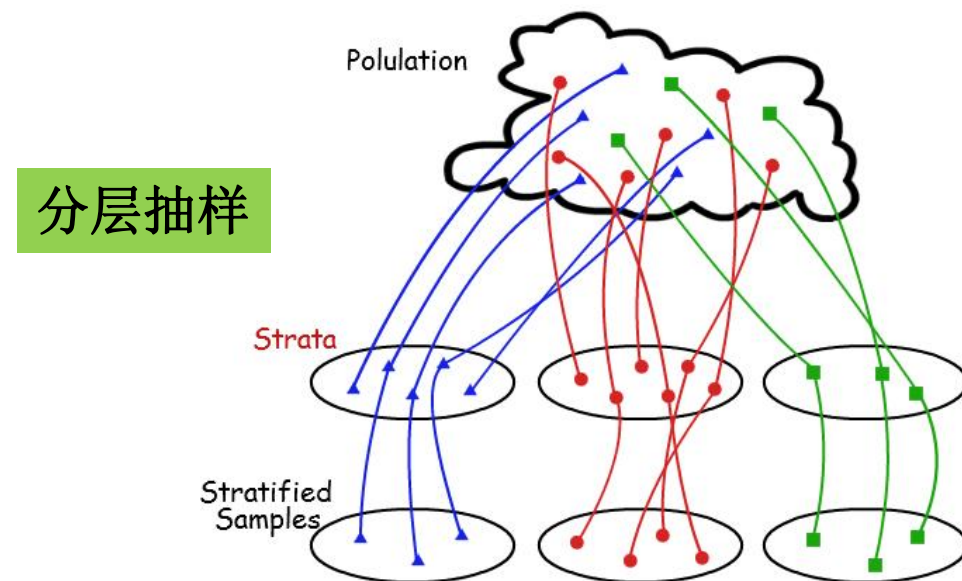
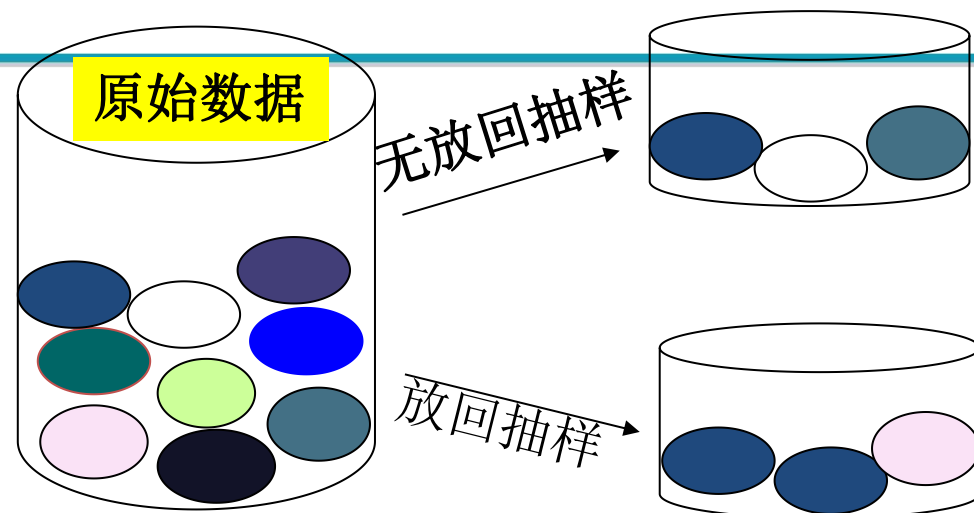
# 抽样调查

---

- 抽样：获得一个小样本 $s$ 来代表整个数据集 $N$
- 允许数据挖掘算法以可能与数据大小呈亚线性的复杂度运行
- 关键原则。选择一个**有代表性的**数据子集
  - 在存在偏斜的情况下，简单的随机抽样可能会有很差的表现
  - 制定适应性的抽样方法，例如分层抽样。
- 注意：取样可能不会减少数据库的I/O（每次都是一页）。

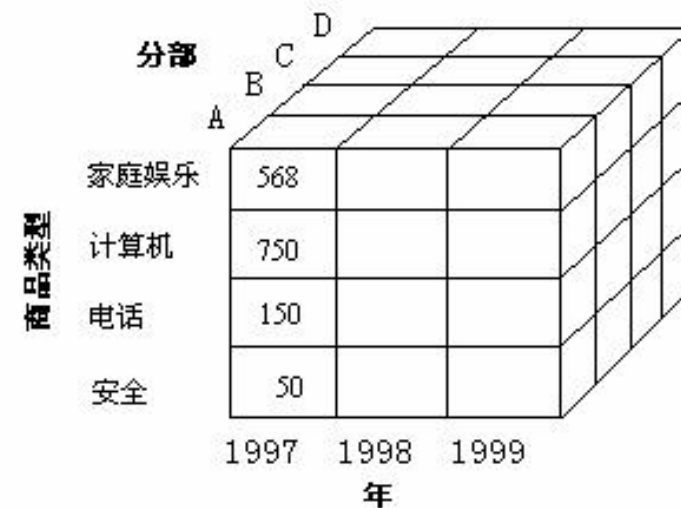
# 抽样的类型

- 简单随机抽样：选择任何特定项目的概率相同
- 无放回抽样
  - 一旦一个对象被选中，它就会从群体中被移除。
- 放回抽样
  - 一个被选中的对象没有从群体中删除
- 分层抽样
  - 对数据集进行分区（或聚类），并从每个分区中抽取样本（按比例，即数据的百分比大致相同）。



# 数据立方体聚合

- 数据立方体的最低层（基础立方体）。
  - 有关单个实体的汇总数据
  - 例如，在一个打电话的数据仓库中的客户
- 数据立方体中的多层次聚合
  - 进一步减少需要处理的数据规模
- 参考适当的水平
  - 使用足以解决任务的最小的表示法
- 在可能的情况下，关于汇总信息的查询应使用数据立方体来回答。

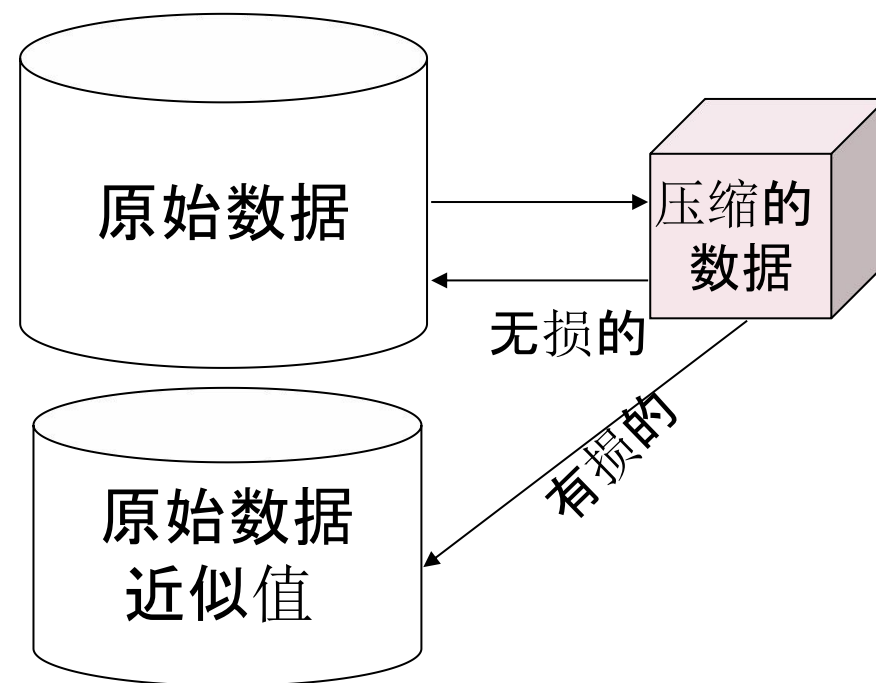


年=1999	
年=1998	
年=1997	
季度	销售额
Q1	\$224,000
Q2	\$408,000
Q3	\$350,000
Q4	\$586,000

年	销售额
1997	\$1,568,000
1998	\$2,356,000
1999	\$3,594,000

# 数据压缩

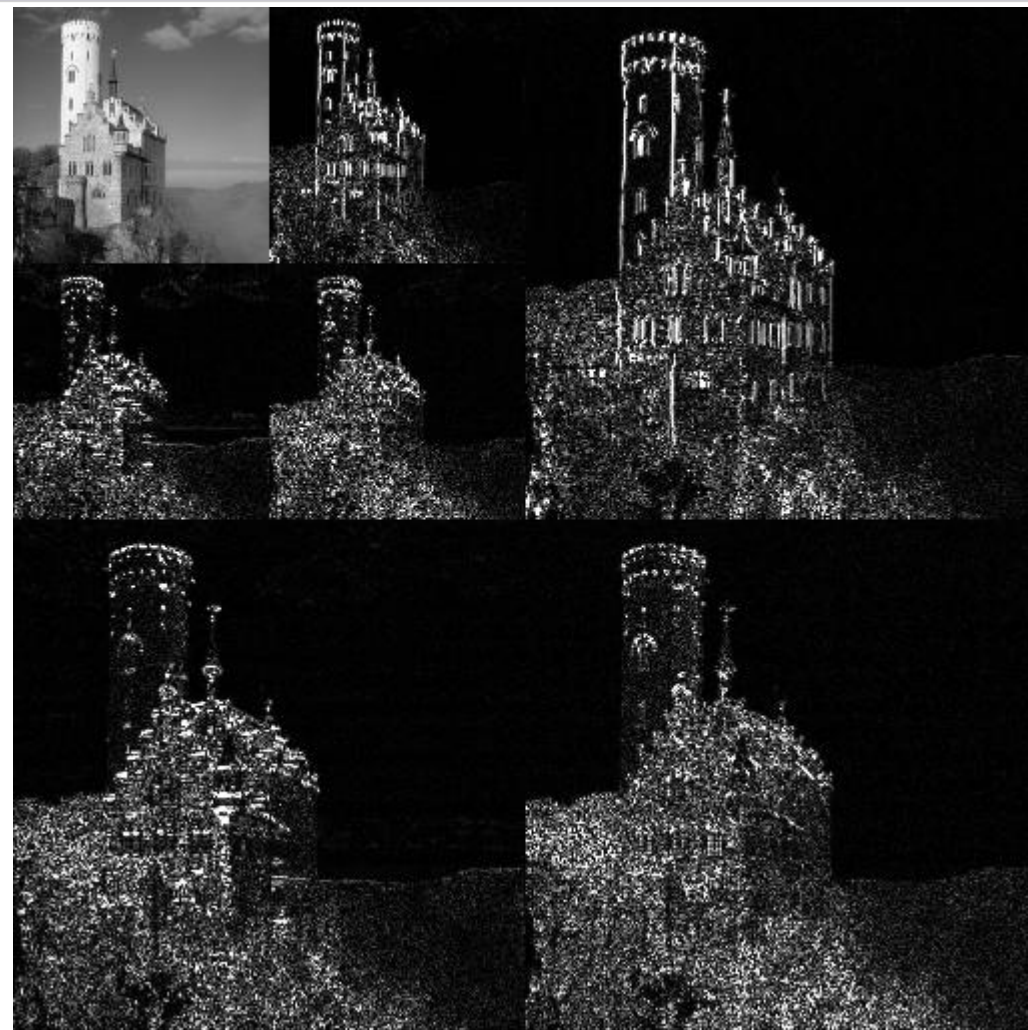
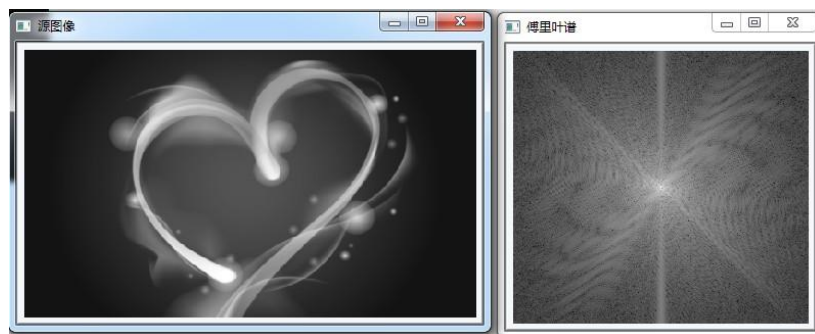
- 字符串压缩
  - 有大量的理论和精心调校的算法
  - 通常是无损的，但在没有扩展的情况下只能进行有限的操作
- 音频/视频压缩
  - 通常是有损压缩，并逐步细化
  - 有时，小的信号片段可以被重建，而不需要重建整个信号。
- 数据归约和降维也可被视为数据压缩的形式



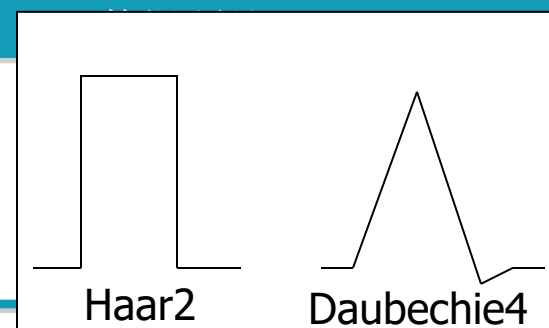
有损压缩与无损压缩

# 小波变换—一种数据压缩技术

- 小波变换
  - 将一个信号分解成不同的频率子带
  - 适用于n维信号
- 对数据进行转换，以保持不同级别分辨率下物体之间的相对距离
- 让自然集群变得更有区别性
- 用于图像压缩



# 小波变换



- 离散小波变换（DWT）用于线性信号处理，多分辨率分析
- 压缩的近似值：只存储小波系数中最强的一小部分
- 类似于离散傅里叶变换（DFT），但有损压缩效果更好，在空间上进行定位
- 方法：
  - 长度， $L$ ，必须是2的整数倍（必要时用0填充）。
  - 每个变换都有2个功能：平滑、差异
  - 适用于成对的数据，产生两组长度为 $L/2$ 的数据
  - 递归地应用两个函数，直到达到所需的长度。



# 小波分解

- 小波。一种用于空间效率高的函数分层分解的数学工具
- $S=[2, 2, 0, 2, 3, 5, 4, 4]$ 可以转化为 $S^{\wedge}=[2^{3/4}, -1^{1/4}, 1/2, 0, 0, -1, -1, 0]$
- 压缩：许多小的细节系数可以用0代替，只保留重要的系数。

Resolution	Averages	Detail Coefficients
8	$[2, 2, 0, 2, 3, 5, 4, 4]$	
4	$[2, 1, 4, 4]$	$[0, -1, -1, 0]$
2	$[1\frac{1}{2}, 4]$	$[\frac{1}{2}, 0]$
1	$[2\frac{3}{4}]$	$[-1\frac{1}{4}]$

# 为什么是小波变换？

---

- 使用帽子形状的过滤器
  - 强调点聚集的区域
  - 抑制其边界内较弱的信息
- 有效去除异常值
  - 对噪声不敏感，对输入顺序不敏感
- 多分辨率
  - 在不同尺度上检测任意形状的群组
- 高效
  - 复杂度 $O(N)$
- 只适用于低维数据



# 数据变换

- 数据被变换或统一成适合于挖掘的形式
- 一个函数，它将一个给定属性的整个值集映射到一个新的替代值集，并且每个旧值可以与一个新值相识别。
- 方法
  - 平滑化：去除数据中的噪音
  - 属性/特征的构建
    - 从给定的属性中构建新的属性
  - 聚合。归纳，数据立方体的构建
  - 规范化。按比例缩小到一个较小的、指定的范围内
    - 最小-最大规范化
    - z-score规范
    - 小数定标规范化
  - 离散化。概念层次的攀升

# 规范化Normalization

- 最小-最大规范化：到 $[\text{new\_min}_A, \text{new\_max}_A]$

$$v' = \frac{v - \min_A}{\max_A - \min_A} (\text{new\_max}_A - \text{new\_min}_A) + \text{new\_min}_A$$

- 例子：让收入范围从12,000美元到98,000美元归一化为 $[0.0, 1.0]$ 。

- 然后73,000美元被映射到  $\frac{73,600 - 12,000}{98,000 - 12,000} (1.0 - 0) + 0 = 0.716$

- Z-score规范化（ $\mu$ ：平均值， $\sigma$ ：标准差）。

$$v' = \frac{v - \mu_A}{\sigma_A}$$

Z-score原始分数与总体平均值之间的距离，以标准差为单位。

- 例子： $\mu=54,000$ ， $\sigma=16,000$ 。那么  $\frac{73,600 - 54,000}{16,000} = 1.225$

- 小数定标规范化

$$v' = \frac{v}{10^j} \quad \text{其中 } j \text{ 是最小的整数, 使 } \text{Max}(|v'|) < 1$$

- 假设取值由-986到917，最大绝对值为986，因此 $j=3$ （1000）。

# 离散化

- 三种类型的属性
  - 标称-来自无序集的值，如颜色、职业等。
  - 有序值-来自一个有序集合的数值，如军衔或学历
  - 数值-实数，如整数或实数
- 离散化。将一个连续属性的范围划分为若干区间
  - 然后可以用区间标签来代替实际的数据值
  - 通过离散化减少数据大小
  - 有监督的与无监督的
  - 分割（自上而下）与合并（自下而上）的关系
  - 可以在一个属性上递归地进行离散化处理
  - 为进一步的分析做准备，例如，分类

# 数据离散化方法

---

- 分箱
  - 自上而下的分割，无监督的
- 直方图分析
  - 自上而下的分割，无监督的
- 聚类分析
  - 无监督的、自上而下的分割或自下而上的合并
- 决策树分析
  - 受监督的、自上而下的分割
- 相关性（如  $\chi^2$ ）分析
  - 无监督的、自下而上的合并
- 注意：所有的方法都可以递归应用

# 分箱：简单的离散化方法

- **等宽度**剖分:

- 分成大小相等的 $n$ 个区间: 均匀网格
- 若 $A$ 和 $B$ 是属性的最低和最高取值, 区间宽度为:  $W = (B - A)/N$ .
- 孤立点可能占据重要影响
- 倾斜的数据处理不好

- **等频剖分/等深**:

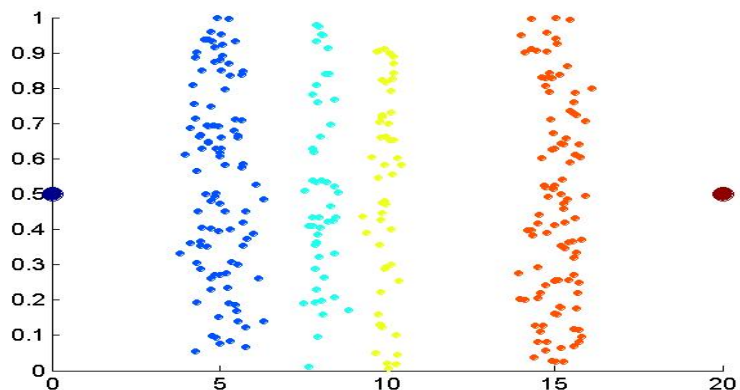
- 分成 $n$ 个区间, 每一个含近似相同数目的样本
- **Good data scaling**
- 类别属性可能会非常棘手.

# Binning Methods for Data Smoothing

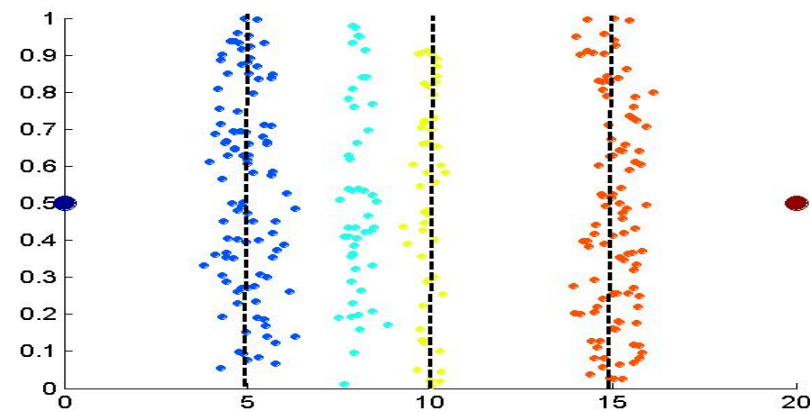
---

- \* Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
- \* Partition into (equi-depth) bins:
  - **Bin 1:** 4, 8, 9, 15
  - **Bin 2:** 21, 21, 24, 25
  - **Bin 3:** 26, 28, 29, 34
- \* Smoothing by bin means:
  - **Bin 1:** 9, 9, 9, 9
  - **Bin 2:** 23, 23, 23, 23
  - **Bin 3:** 29, 29, 29, 29
- \* Smoothing by bin boundaries:
  - **Bin 1:** 4, 4, 4, 15
  - **Bin 2:** 21, 21, 25, 25
  - **Bin 3:** 26, 26, 26, 34

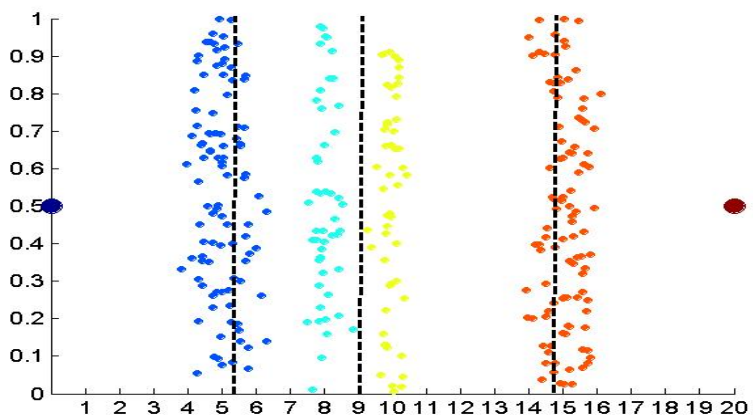
# 无监督的离散化-分箱与聚类



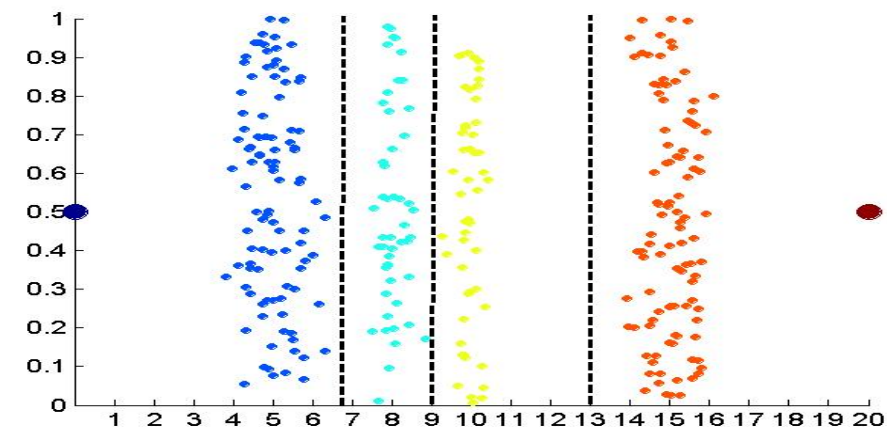
数据



等宽(距离)分选



等深度(频率)(分选)



K-means聚类导致更好的结果

# 通过分类和相关分析进行离散化

- 分类（如决策树分析）。
  - 有监督的。给出类别标签，例如，癌性与良性。
  - 使用熵值来确定分割点（离散点）。
  - 自上而下，递归分割
  - 细节将在“分类”一章中介绍。
- 相关分析（如Chi-merge：基于 $\chi^2$ 的离散化）。
  - 监督：使用类信息
  - 自下而上的合并。找到最佳的相邻区间（具有相似的类别分布，如低  $\chi^2$  值）进行合并。
  - 递归地进行合并，直到预定的停止条件。



# 概念分层

---

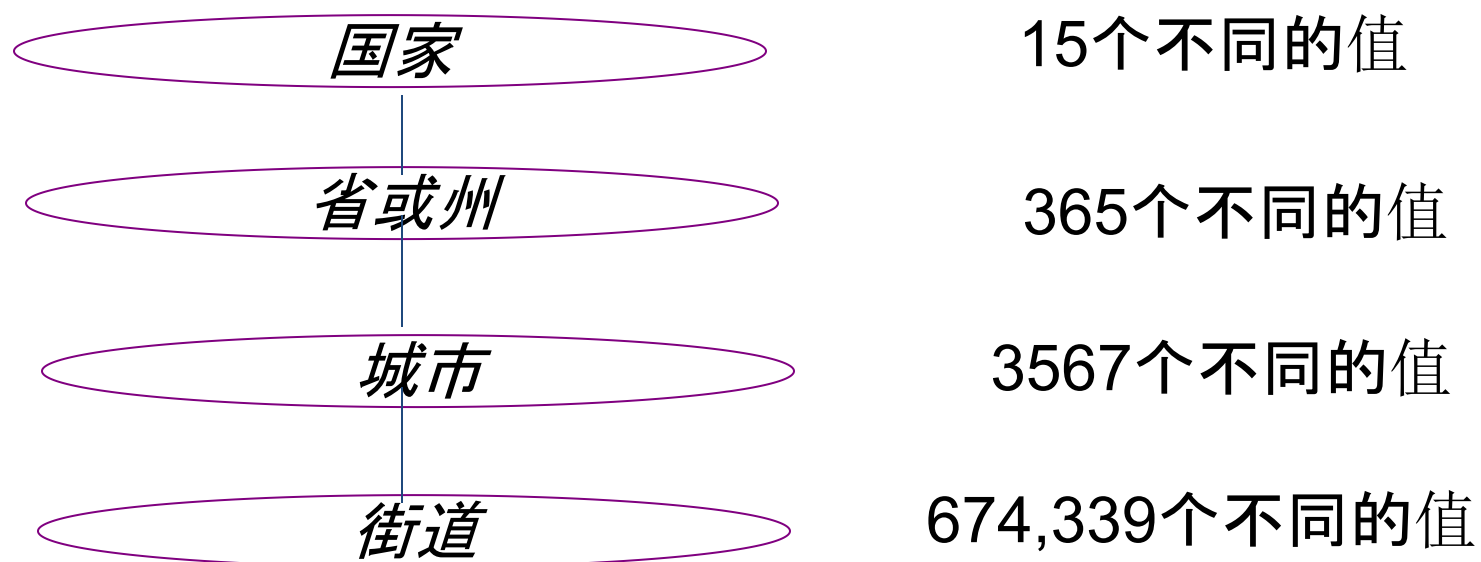
- **概念分层**将概念（即属性值）分层次地组织起来，通常与数据仓库中的每个维度相关。
- 概念分层便于在数据仓库中进行钻取和滚动，以查看多个粒度的数据
- 概念层次的形成。通过收集和用更高层次的概念（如 *青年*、*成人*或*老年*）取代低层次的概念（如 *年龄的数值*）来反复减少数据。
- 概念层次可以由领域专家和/或数据仓库设计师明确指定
- 对于数字和标称数据都可以自动形成概念层次-对于数字数据，使用离散化方法

# 标称数据的概念层次生成

- 由用户或专家在模式层面明确指定属性的部分/总排序
  - *街道* < *城市* < *省* < *国家*
- 通过明确的数据分组为一组数值指定层次结构
  - {南京、苏州、徐州} < 江苏省
- 只指定部分属性集
  - 例如，只有 *街道* < *城市*，没有其他。
- 通过分析不同值的数量，自动生成层次结构（或属性级别）。
  - 例如，对于一组属性。{*街道*、*城市*、*省*、*国家*}。

# 自动生成概念层次结构

- 一些层次结构可以根据对数据集中每个属性的不同数值的分析而自动生成。
  - 具有最多不同值的属性被放在层次结构的最低层
  - 例外情况，如：工作日、月份、季度、年份



# 第3章：数据预处理

---

- 数据预处理：概述
- 数据清理
- 数据集成
- 数据归约和变换
- 数据降维
- 小结



# 降维

---

## ■ 维度灾难

- 当维度增加时，数据变得越来越稀疏
- 对聚类、离群点分析至关重要的密度和点间距离，变得不那么有意义了
- 子空间的可能组合将呈指数级增长

## ■ 降低维度

- 通过获得一组主变量，减少所考虑的随机变量的数量

## ■ 降维的优势

- 避免维度灾难
- 帮助消除不相关的特征，减少噪音
- 减少数据挖掘中所需要的时间和空间
- 允许更容易的视觉化

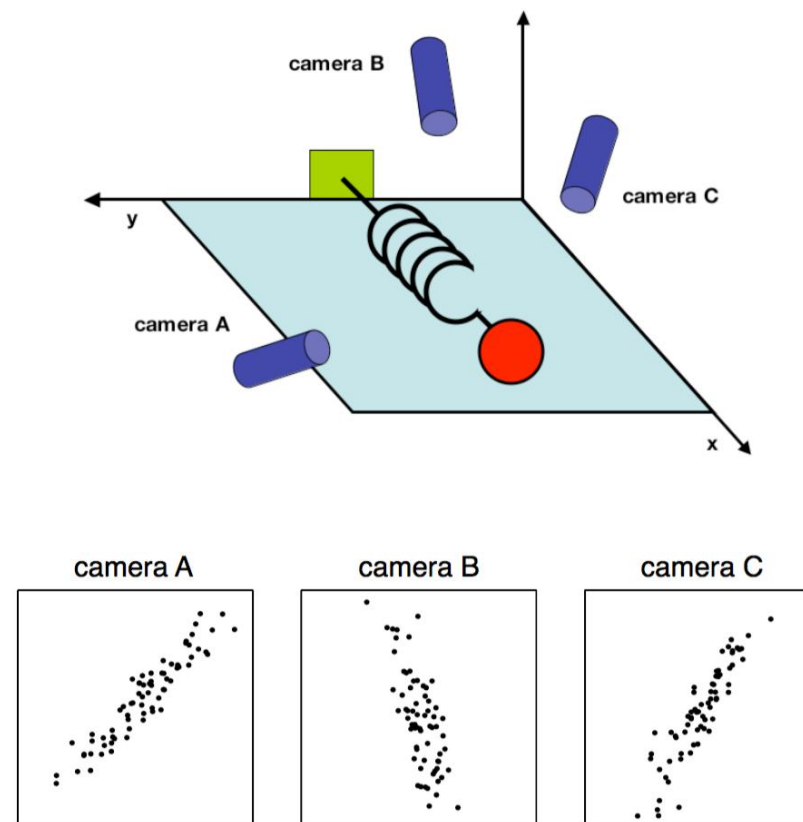
# 降维技术

---

- 降低维度的方法
  - **特征选择**。找到原始变量（或特征、属性）的一个子集
  - **特征提取**。将高维空间的数据转化为较少维度的空间
- 一些典型的维度方法
  - 主成分分析
  - 监督和非线性技术
    - 特征子集选择
    - 特征创建

# 主成分分析（PCA）

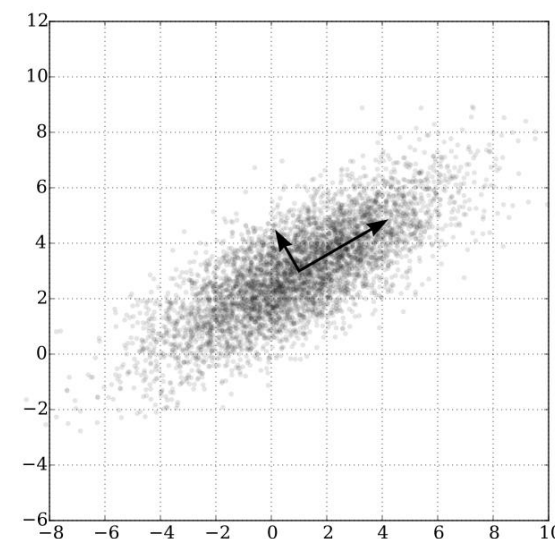
- PCA：一种统计程序，使用正交变换，将一组可能相关的变量观测值转换成一组线性不相关的变量值，**称为主成分**。
- 原始数据被投射到一个更小的空间，从而实现降维。
- 方法：找到协方差矩阵的特征向量，这些特征向量定义新的空间



球在一条直线上行驶。来自三台摄像机的数据包含很多冗余内容

# 主成分分析（方法）

- 给出 $n$ 维的 $N$ 个数据向量，找出 $k \leq n$ 的正交向量（主成分），用来表示数据。
  - 对输入数据进行标准化。每个属性都落在相同的范围内
  - 计算 $k$ 个正交（单位）向量，即主成分
  - 每个输入数据（向量）是 $k$ 个主成分向量的线性组合
  - 主成分按“重要性”或强度递减的顺序进行排序
  - 由于成分是分量的，通过消除弱成分，即那些低方差的成分，可以减少数据的大小（即使用最强的主成分，以重建一个良好的原始数据近似）。
- 只对数值数据起作用

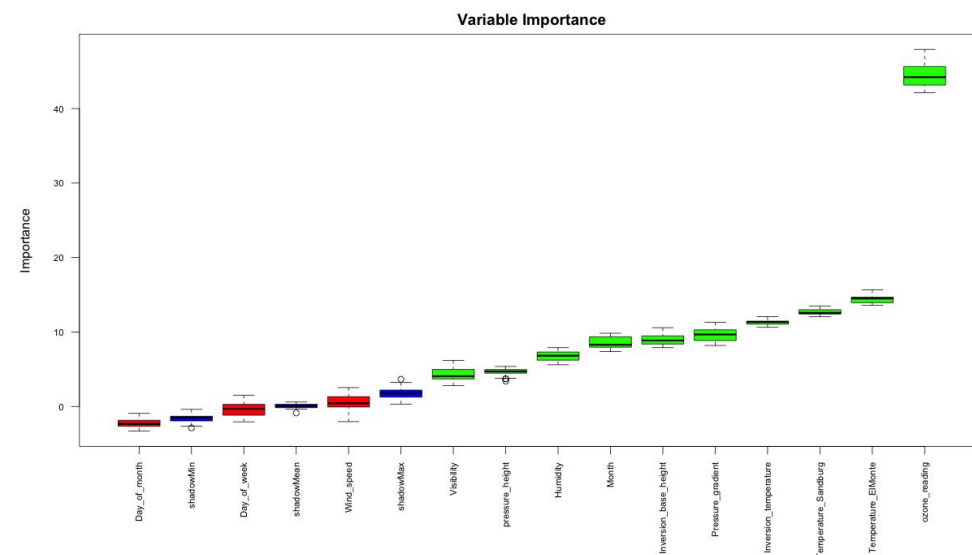


维基百科:主成分分析



# 属性子集选择

- 另一种降低数据维度的方法
- 冗长的属性
  - 重复一个或多个其他属性中包含的大部分或全部信息
    - 例如，产品的购买价格和支付的销售税金额
- 不相关的属性
  - 不包含对手头的数据挖掘任务有用的信息
    - 例子。一个学生的ID通常与预测他/她的GPA无关。



# 属性选择中的启发式搜索

- 有可能 $2^d$ 的属性组合是 $d$ 个属性
- 典型的启发式属性选择方法。
  - 属性独立假设下的最佳单一属性：通过显著性检验选择
  - 最佳阶梯式特征选择。
    - 首先挑选出最好的单一属性
    - 然后，下一个最佳属性条件是第一个，.....。
  - 循序渐进的属性消除。
    - 反复消除最差的属性
  - 最佳组合属性选择和消除
  - 最佳的分支和边界。
    - 使用属性消除和回溯法

# 属性创建（特征生成）

- 创建新的属性（特征），可以比原来的属性更有效地捕捉数据集中的重要信息
- 三个一般的方法学
  - 属性提取
    - 特定领域
  - 将数据映射到新的空间（见：数据还原）。
    - 例如，傅里叶变换、小波变换、流形方法（不包括）。
  - 属性构建
    - 结合特征（见：“高级分类”一章中的判别性频繁模式）。
    - 数据离散化

# 摘要

---

- **数据质量**：准确性、完整性、一致性、及时性、可信度、可解释性
- **数据清理**：如缺失/噪声值、异常值
- 来自多个来源的**数据集成**。
  - 实体识别问题；消除冗余；检测不一致的地方
- **数据归约、数据变换和数据离散化**
  - 数值归约；数据压缩
  - 归一化；概念层次的生成
- **降低维度**

# References

---

- E. Rahm and H. H. Do. Data Cleaning: Problems and Current Approaches. *IEEE Bulletin of the Technical Committee on Data Engineering*. Vol.23, No.4
- D. P. Ballou and G. K. Tayi. Enhancing data quality in data warehouse environments. *Communications of ACM*, 42:73-78, 1999.
- H.V. Jagadish et al., Special Issue on Data Reduction Techniques. *Bulletin of the Technical Committee on Data Engineering*, 20(4), December 1997.
- A. Maydanchik, Challenges of Efficient Data Cleansing (DM Review - Data Quality resource portal)
- D. Pyle. *Data Preparation for Data Mining*. Morgan Kaufmann, 1999.
- D. Quass. A Framework for research in Data Cleaning. (Draft 1999)
- V. Raman and J. Hellerstein. Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, VLDB'2001.
- T. Redman. *Data Quality: Management and Technology*. Bantam Books, New York, 1992.
- Y. Wand and R. Wang. Anchoring data quality dimensions ontological foundations. *Communications of ACM*, 39:86-95, 1996.
- R. Wang, V. Storey, and C. Firth. A framework for analysis of data quality research. *IEEE Trans. Knowledge and Data Engineering*, 7:623-640, 1995.
- <http://www.cs.ucla.edu/classes/spring01/cs240b/notes/data-integration1.pdf>