第三章 数据预处理

内容提要

- 关于数据
- 为什么要预处理数据？
- 描述性数据汇总
- 数据清理
- 数据集成和变换
- 数据规约
- 数据离散化和概念分层产生

什么是数据?

- 数据是对客观世界及对象的一种符号化或数量化的描述与表示。
- 数据是数据对象的集合及其属性
- 属性是对象的性质或者特征
  - 例如：人眼睛的颜色，温度等
  - 属性也可以理解为变量，领域，特征或者特点
- 描述一个对象的属性集合
  - 对象也可以理解为记录，观点，案例，样本，实体或者实例**Objects**

Attributes

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

属性值

- 属性值是分配给一个属性的数字或者符号
- 属性和属性值的区别
  - 相同的属性可以映射到不同的属性值
    - 例如：高度既可以用尺也可以用米衡量
  - 不同的属性可以映射到相同的取值集合
    - 例如：ID和年龄的属性值是整数
    - 但是属性值的特性可以不同
    - 例如：ID没有限制但是年龄有最大值和最小值

属性的类型

- 属性有很多不同的类型
  - Nominal（名称型号）
    - 例如：身份号码，眼睛颜色，邮政编码
  - Ordinal（顺序型）
    - 例如：排名（例如，薯片味道从1-10级），分数，身高{高、中、低}
  - Interval（间隔型）
    - 例如：日期间隔，摄氏温度或华氏温度
  - Ratio（比率型）
    - 例如：百分比，人口比例

属性值的内容

- 属性的类型取决于它有下列的哪一个特征：
  - 区别性： = ≠
  - 顺序性： < >
  - 可加性： + −
  - 乘除性： * /

  - 名称型属性：区别性
  - 顺序型属性：区别性&顺序性
  - 间隔的属性：区别性，顺序性&加法性
  - 比率的属性：所有四种特征

离散和连续属性

- **离散属性**
  - 只有一个有限集或可数的属性值集
  - 例如：邮政编码，颜色，或者是一个文档集合的词集
  - 通常表示为整数变量
  - 注：二进制属性是离散属性的特殊情况
- **连续属性**
  - 实数作为属性值
  - 例如：温度，高度，或者重量.
  - 特别的，实际值只能用有限位数的数字测量和表示
  - 连续性属性通常用浮点变量表示

数据集的类型

- 根据数据的组织方式和相对关系
- 记录数据——一条条记录组成
  - 数据矩阵
  - 文本数据
  - 交易数据
- 图数据——由记录和记录直接的联系组成
  - 互联网
  - 化学分子结构
- 有序数据——记录之间存在时间/空间上的有序关系
  - 空间的数据
  - 时间的数据
  - 连续的数据
  - 基因序列数据

记录数据

- 由记录集合组成的数据，每一个记录又由一个固定的属性集组成

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

记录数据—数据矩阵

- 如果数据对象具有相同的一套固定的数值属性，那么数据对象可以被认为是一个多维空间中的点，其中每个维度代表了不同的属性

- 这样的数据集可以用m*n的矩阵表示，m行，每行代表一个对象，n列，每列代表一个属性

Projection of x Load	Projection of y load	Distance	Load	Thickness
10.23	5.27	15.22	2.7	1.2
12.65	6.25	16.22	2.2	1.1

- 每个文档都成为一个“term”向量
  - 每个term都是向量的一个分量,
  - 每个分量的值就是对应的term在文档中出现的次数.

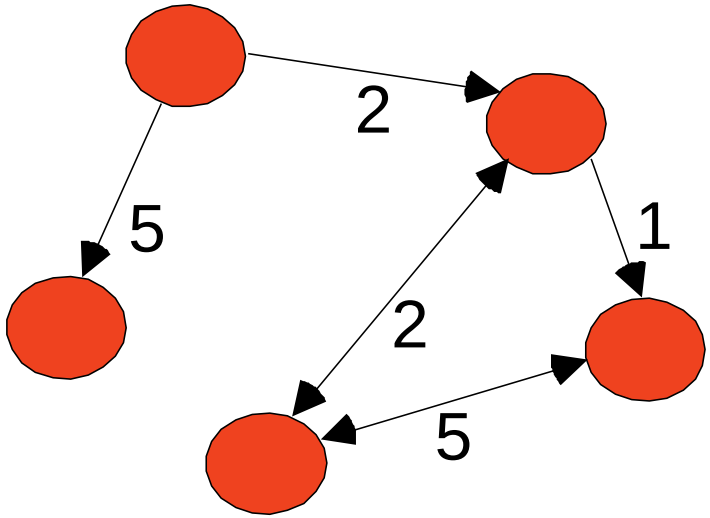
	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

记录数据-交易数据

- 一组特殊类型的记录数据
  - 每个交易记录都涉及一组项目
  - 例如：考虑一个杂货店，一个顾客一次购物所买的一组商品就构成一次交易，这些购买的商品就是项目

TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

- 例如：一般的图片和网页链接



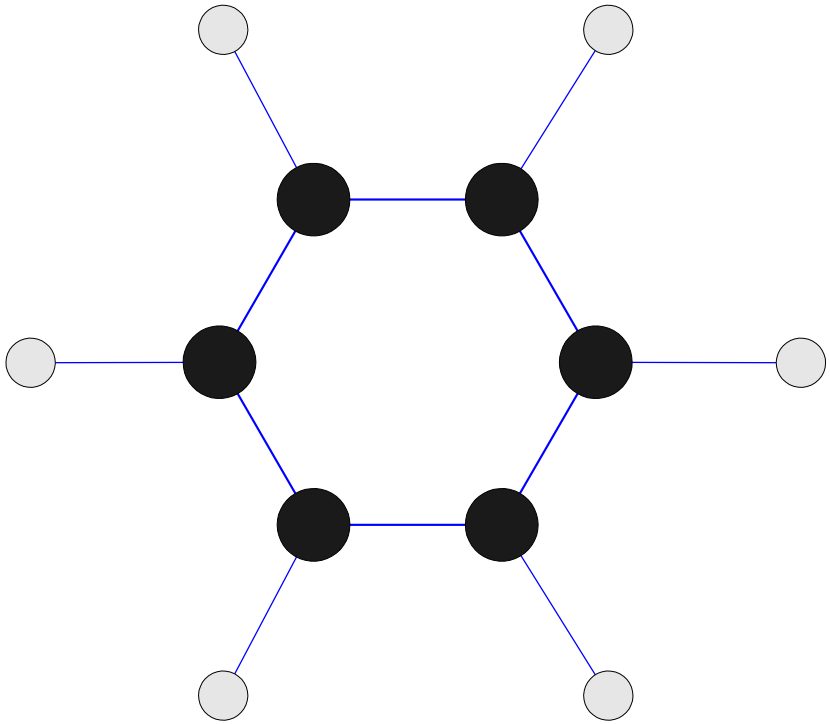
Data Mining

Graph Partitioning

Parallel Solution of Sparse Linear System of Equations

<a href="papers/papers.html#ffff">
N-Body Computation and Dense Linear System Solvers

- Benzene Molecule (苯分子): C_6H_6



有序数据

- 交易序列
- 染色体序列数据

Items/Events

(A B) (D) (C E)
(B D) (C) (E)
(C D) (B) (A E)

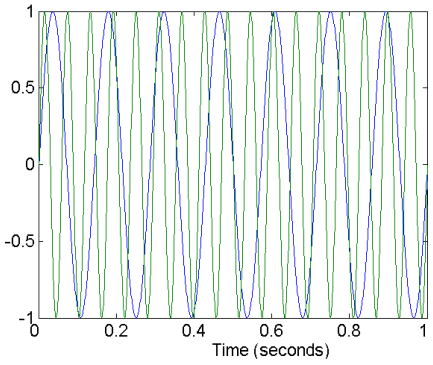
序列中的一个元素

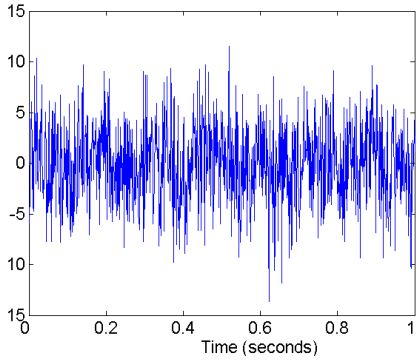
```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

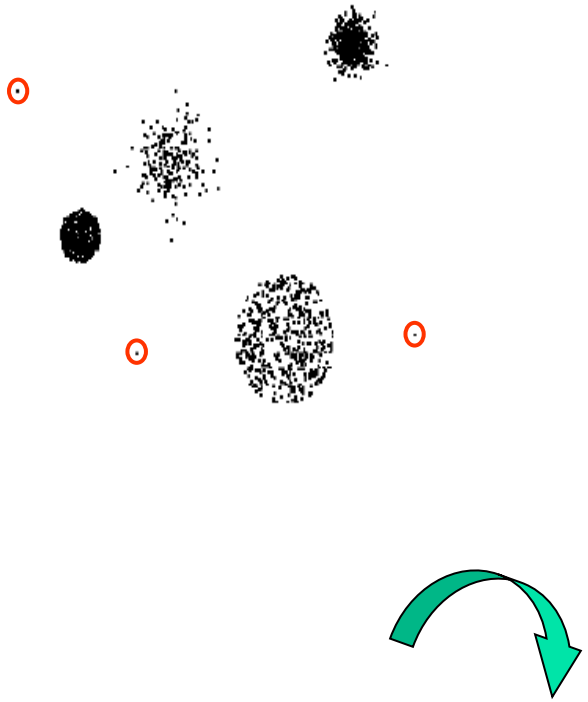
数据质量

- 什么类型的数据质量有问题？
- 如何从数据中发现问题？
- 怎么处理这些问题？

- 数据质量问题的例子：
  - 噪音和离群点
  - 缺失值
  - 重复数据







数据预处理

转：某团队去年用核磁共振扫描了狗的大脑，发现狗是用左脑对语言进行处理的，并在Science上发了论文。然后今年才意识到，人是躺着进去的，而狗是趴着进去的。所以左右脑搞反了...
搞！反！了！！！


Erratum for the Report "Neural mechanisms for lexical processing in dogs" by A. Andics, A. Gábor, M. Gácsi, T. Faragó, D. Szabó, Á. Miklósi

2小时前

令人无语的分析与数据

为什么要进行数据预处理（一）

- 真实世界的数据太杂乱
  - Incomplete(不完整):不完整：缺少属性值，缺少感兴趣对象的确切属性，或者只有汇总数据
    - e.g., occupation= “”
  - Noisy（有噪音）:有错误或者是离散点
    - e.g., Salary= “-10”
  - Inconsistent（不一致）:编码或者名称有冲突:
    - 外部冲突
      - e.g., Age=“42” Birthday=“03/07/1997”
      - e.g., Was rating “1,2,3”, now rating “A, B, C”
      - e.g.,重复记录中的冲突
    - 内部冲突
      - e.g., IngrA(10)+IngrB(3)+IngrC(4) -> Germ(70%)
      - IngrA(13)+IngrB(2)+IngrC(4) -> Germ(65%)

为什么要进行数据预处理（二）

- 不完整的数据来自：
  - 对数据收集时间和分析时间的不同考虑
  - 人为/硬件/软件 因素
- 有噪音的数据来自数据处理的过程
  - 采集
  - 进入
  - 传播
  - 与常识的冲突
- 不一致数据来自：
  - 不同的数据来源
  - 实际试验设备
  - 不同的环境条件

为什么要进行数据预处理（三）

- 需要转换数据类型
  - 需要集成不同的数据
    - e.g. In Table A: Age = “”；
    - e.g. In Table B: Weight= “”
  - 需要转换不同的数据
    - e.g.问卷调查的数据
  - 不同的数据需要离散化
  - 不同的数据需要规约

为什么数据预处理很重要？

- 没有高质量的数据，就没有高质量的挖掘结果
  - 有质量的决定必须基于有质量的数据
    - e.g.,重复的或者遗漏的数据可能导致不真实的甚至误导性的统计结果
  - 数据仓库需要一致的高质量的数据集成
- 数据抽取，清理和转换构成了建造数据仓库的大部分工作

数据质量的多维度评估

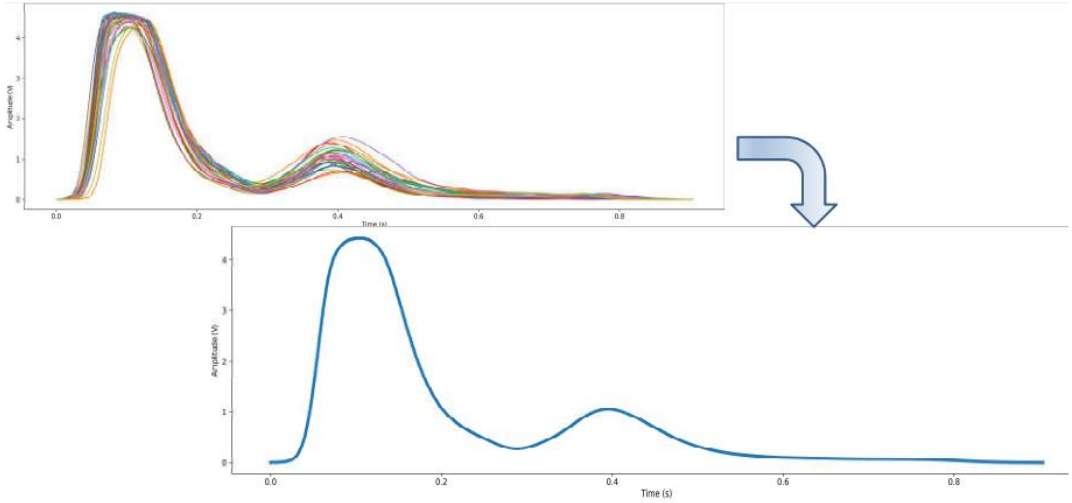
- 一个被大家广泛接受的多维度观点：
  - Accuracy（准确的）
  - Completeness（完整的）
  - Consistency（一致的）
  - Timelines（合时的）
  - Believability（可信的）
  - Value added（有附加价值的）
  - Interpretability（可解释的）
  - Accessibility（可存取的）

数据预处理的主要任务

- **数据清理**
  - 填充缺失数据，平滑有噪音的数据，确认或者去除离散点，解决不一致问题
- **数据集成**
  - 多个数据库，多维数据，或者是文档的整合
- **数据转换**
  - 归一化与聚合
- **数据规约**
  - 获得容量上简化的表示法，但是产生相同或者相似的分析结果
- **数据离散化**
  - 数据规约的一部分但却有相当的重要性，特别是对于数值的数据

- **动机**
  - 更好的理解数据
  - 对数据有一个全局了解

- **描述性数据汇总**
  - Central tendency（集中趋势）
  - Dispersion（散布性）



三种度量方法

- Distributive（分布式度量）：把函数应用到n聚合值得到的结果，与把函数应用到没有分割的所有数据所得到的结果相同
  - count(), sum(), min(), max()
- Algebraic（代数度量）：可以用一个有M个参数的代数函数计算，其中，M是一个有界整数，每个参数都用一个分布函数得到
  - avg() 均值——聚集函数, min_N(), standard_deviation()——标准偏差
- Holistic（整体度量）：用来描述一个子集的存储大小没有固定约束
  - median()——中位数, mode()——众数, rank()

度量中心趋势（一）

- Mean (中值，algebraic measure):
  - 算术平均数:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

  - 加权算术平均数:

$$\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

  - 截断均值：去掉极值

    e. g. 薪水和分数

- Median (中位数, holistic measure)
  - 奇数个数值的中间那个值，或者是偶数个数值的中间两个值的平均值
    - *Data* 57 55 85 24 33 49 94 2 8 51 71 30 91 6 47 50 65 43 41 7
    - *Ordered Data*
      - 2 6 7 8 24 30 33 41 43 *47 49* 50 51 55 57 65 71 85 91 94
    - Median 48

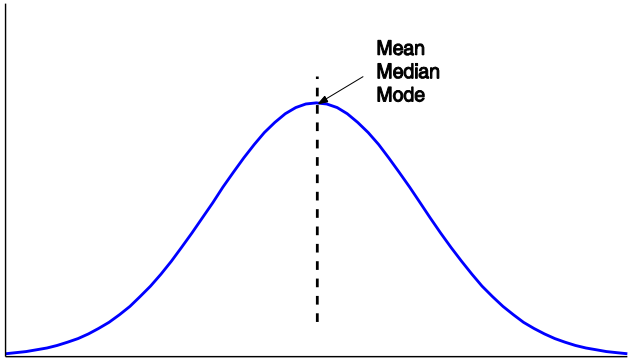
度量中心趋势（三）

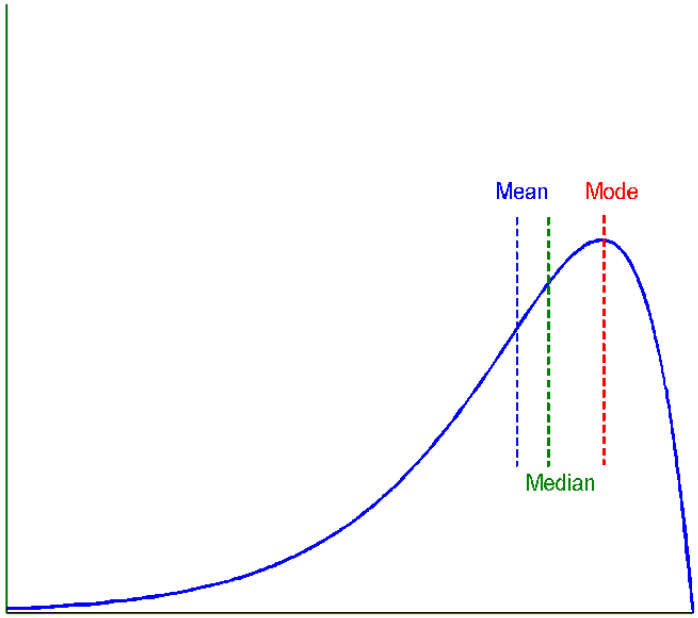
- Mode(众数， Holistic)
  - 数据中出现频率最高的数值
  - Unimodal（单峰）, bimodal（双峰）, trimodal（三峰）
  - 经验公式:
    - For unimodal（单峰） frequency:

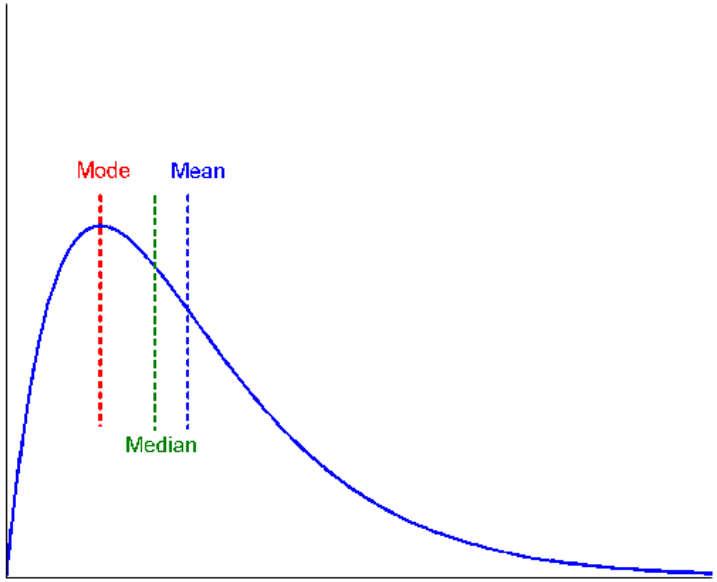
$$mean - mode = 3 \times (mean - median)$$

对称的数据V. S有偏数据

- 对称数据、左偏数据和右偏数据的中位数、均值和众数



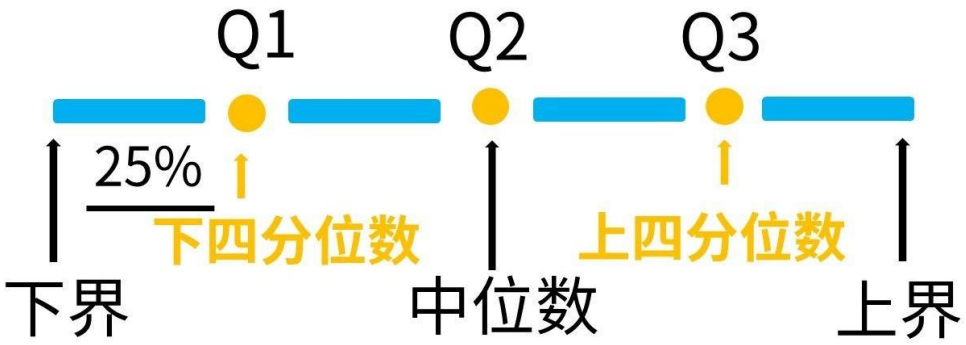




度量数据的离散程度(一)

四分位数，离散点和箱线图

- Quartiles(4分位数)：Q_1 (25th percentile)，Q_3 (75th percentile)
- Inter-quartile range（中间四分位数）：$IQR = Q_3 - Q_1$
- Five number summary（五数概括）：min，Q_1，M，Q_3，max
- Boxplot（箱线图）:两端是四分位数，中位数被标记出来，外边界，并且分别绘制出离散点
- Outlier:通常，比1.5倍的IQR的值高/低的值



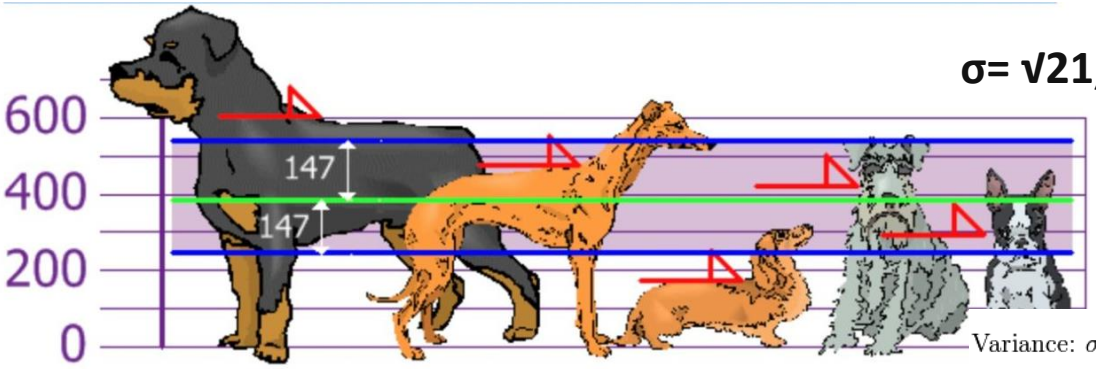
假设数列一共有n个数
Q1第在（n+1）/4位
Q2第 (n+1)/2位
Q3第(n+1)/4*3位

度量数据的离散程度(二)

- Variance（方差） and standard deviation（标准差）
  - Variance s^2:（代数的，可伸缩的计量）

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}[\sum_{i=1}^{n} x_i^{\,2} - \frac{1}{n}(\sum_{i=1}^{n} x_i)^2]$$

  - Standard deviation s 方差s^2的平方根



σ= √21,704= 147.32……=147

身高（到肩膀）是：
600mm、470mm、170mm、430mm 和 300mm

$$\text{Variance: } \sigma^2 = \frac{206^2 + 76^2 + (-224)^2 + 36^2 + (-94)^2}{5}$$
$$= \frac{42,436 + 5,776 + 50,176 + 1,296 + 8,836}{5}$$
$$= \frac{108,520}{5} = 21,704$$

平均 = （*600 + 470 + 170 + 430 + 300*）/**5** = *1970*/**5** = 394

度量数据的离散程度(三)

- **正态分布曲线**

μ（miu） σ（sigma）

  - 从μ－σ 到 μ+σ：

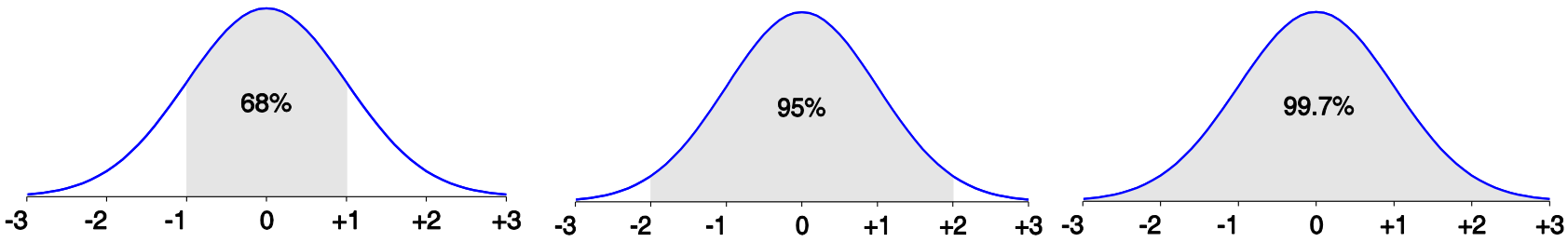
包含了68%的测量值（μ：均值，σ：标准差）

  - 从μ－2σ 到 μ+2σ：

包含了95%的测量值

  - 从μ－3σ 到μ+3σ：

包含了99.7%的测量值



度量数据的离散程度(四)

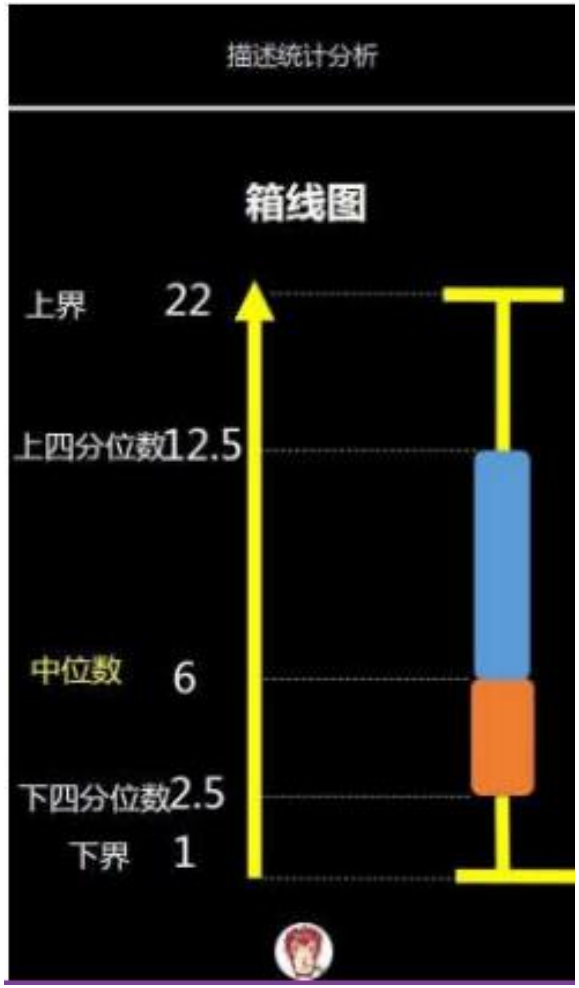
- 箱线图分析
- 一个分布的五数概括：

  Minimum, Q1, M, Q3, Maximum

- 箱线图
  - 数据用一个盒子来表示
  - 盒子两端是第一和第三四分位数，

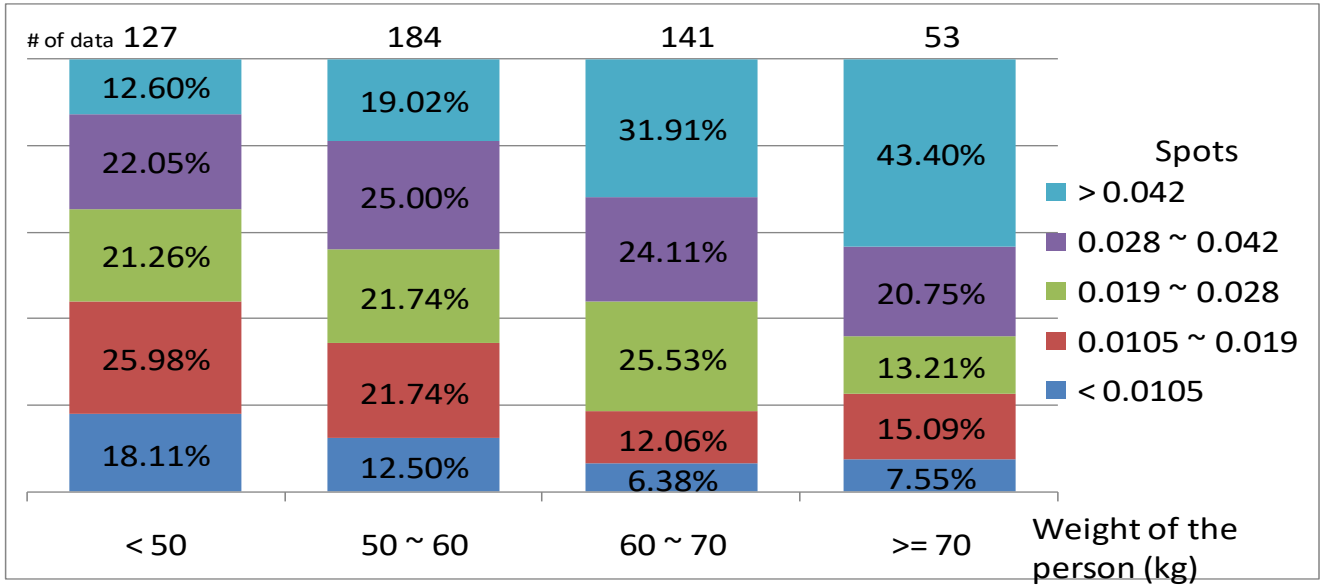
  也就是说，盒子的高度是IRQ
  - 中位数在盒子里用一条线标记出来
  - 外边界：

  盒子外面延伸到最大值和最小值的两条线



描述数据的其他方式（可视化）

- 直方图分析
- 图表展示了基本统计的类描述
  - Frequency histograms（频率直方图）
    - 一个单变量图解法
    - 由一组矩形组成，这些矩形反映了给定数据中所呈现的类别的计数或者是频率



描述数据的其他方式（可视化）

- Quantile Plot（分位数图）
- 展示所有的数据（允许用户同时评估整体行为和不寻常事件）
- 绘制分位数信息
  - 对于一个数据x_i，数据被升序排列，f_i代表小于或等于x_i的数据在全部数据中所占的百分比

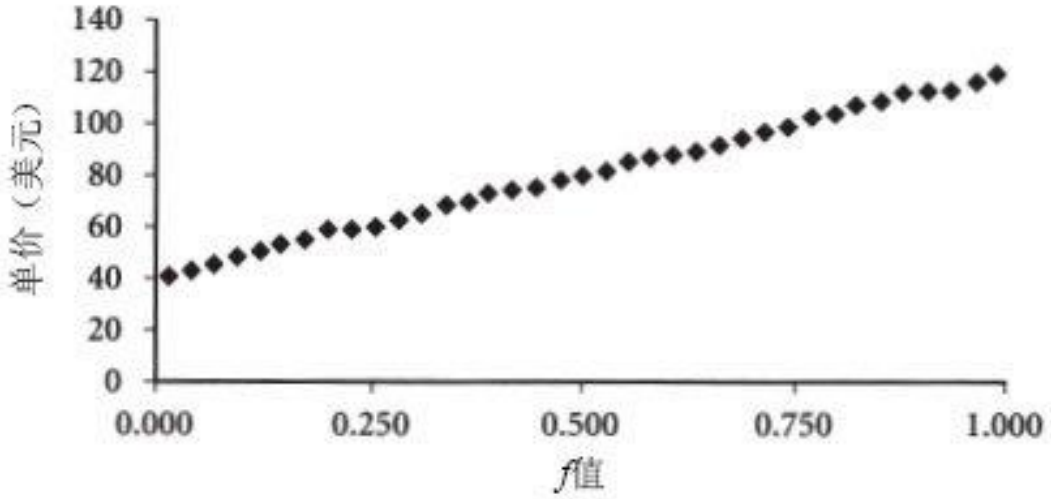


图2-5　表2-1单价数据的分位数图cnrepair.com

描述数据的其他方式（可视化）

- 分位数-分位数图
- 绘制一个单变量分布的分位数和另外一个单变量分布的对应的分位数
- 允许用户观察是否有从一个分布到另一个分布的转变

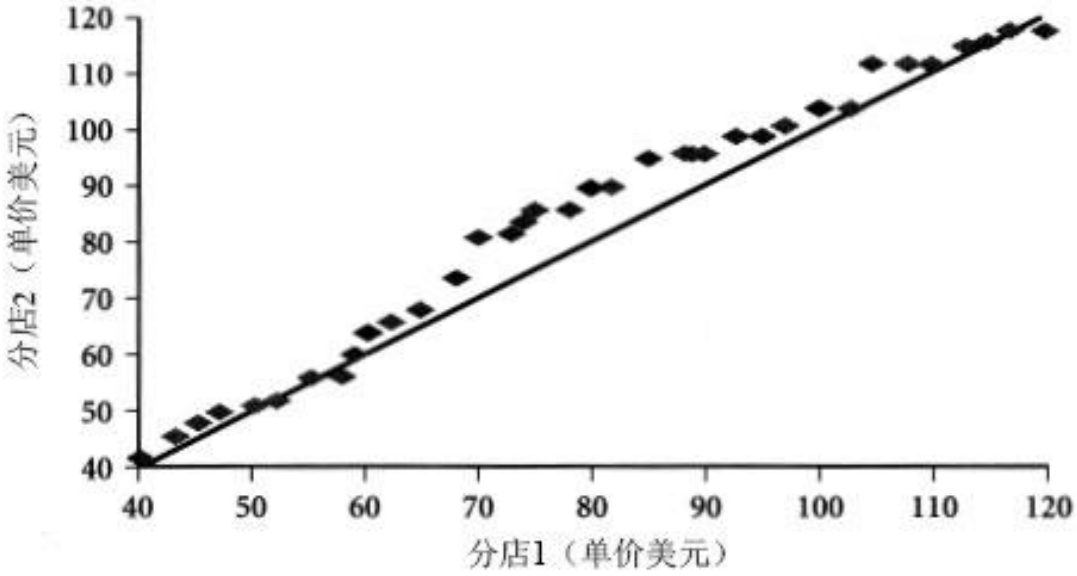


图2-6 两个不同分店的单价数据的分位数-分位数图

描述数据的其他方式（可视化）

- Scatter plot(散点图)
- 提供了一个先看看二元数据的群集和离群点等的途径
- 每对值都被当作一对坐标并在平面上用点绘出来

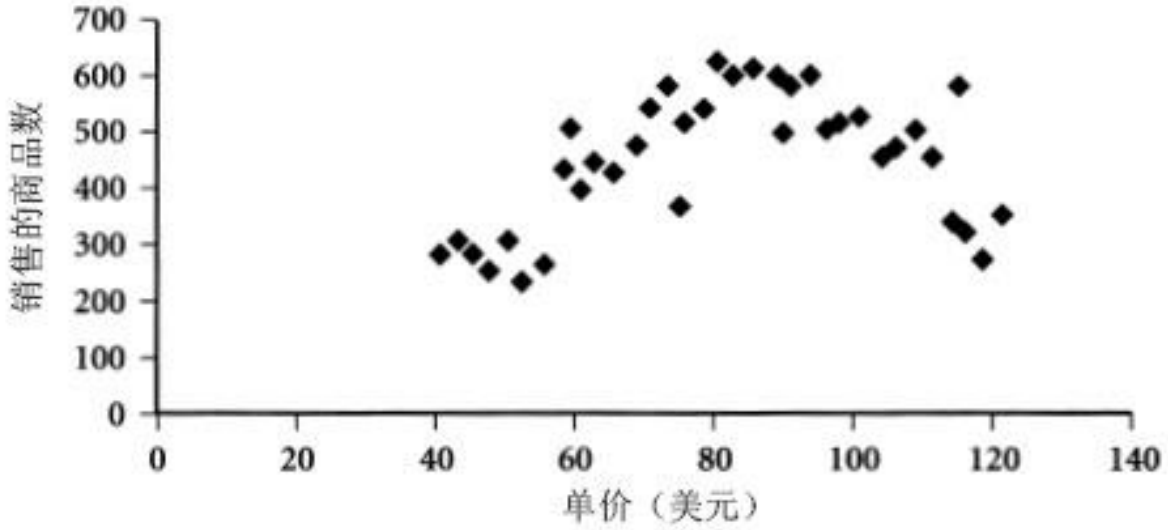
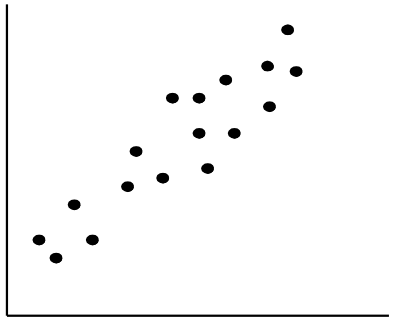
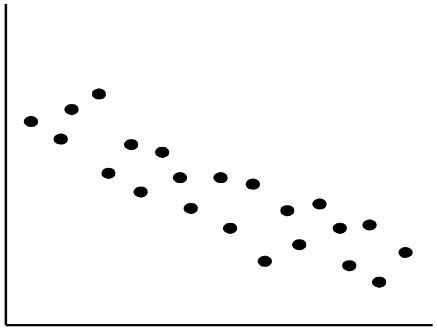


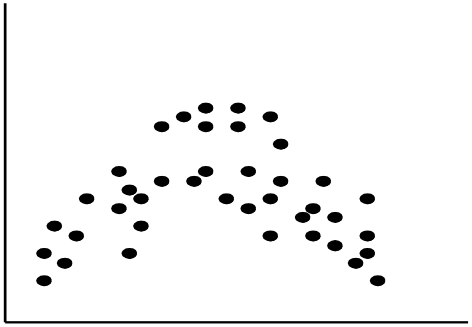
图2-7　表2-1中数据的散布图

散点图-相关数据

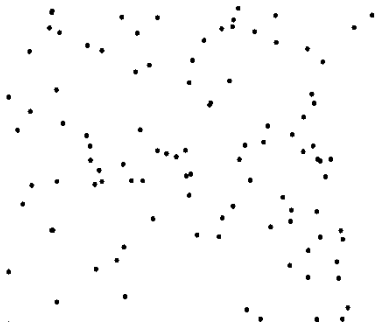
- 相关数据





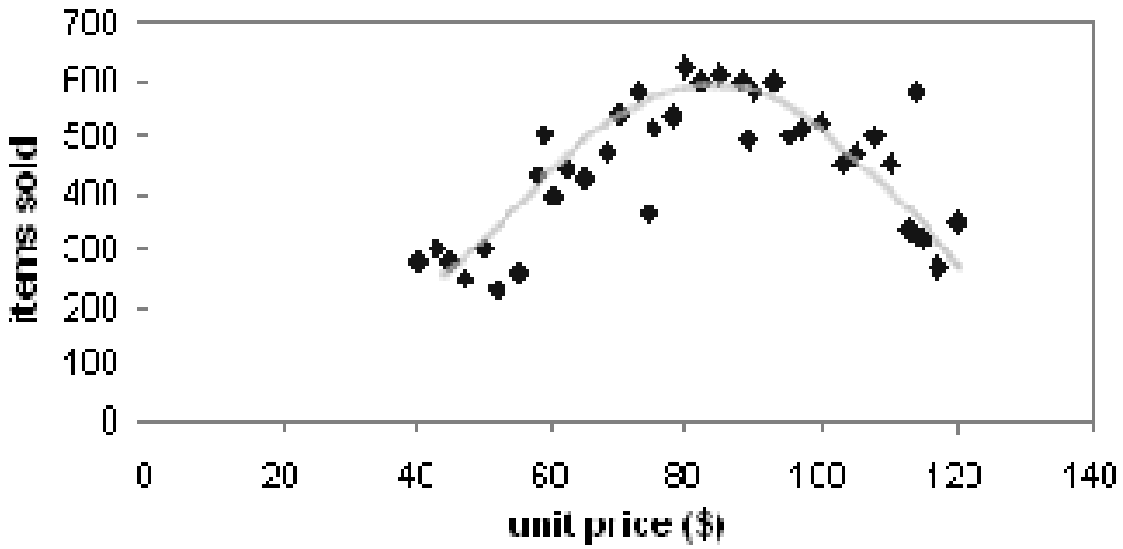


- 不相关数据



描述数据的其他方式（可视化）

- Loess Curve（局部回归曲线）
- 给散点图添加一条平滑的曲线，为模式的依赖性提供更好的展示
- 局部回归曲线通过设置两个参数拟合：一个平滑参数，和回归拟合的多项式程度



数据清洗

- **重要性**
  - “数据清理是数据仓库中最重要的三个问题之一”
    —Ralph Kimball（数据仓库和商务智能领域的权威专家）
  - “数据清理是数据仓库中第一位的问题”—DCI调查
- **数据清理任务**
  - 填充遗漏值
  - 标记离散点和平滑噪音数据
  - 修正不连续数据
  - 解决数据整合引起的冗余

缺失值处理

- **数据并不是总是可用**
  - 例如，许多元组没有多个属性的记录值，比如销售值中的客户收入
- **数据遗漏可能是由于**
  - 设备故障
  - 与其他记录的数据不一致因此删掉了
  - 由于误解没有将数据输入
  - 没有记录确切的数据，在输入数据的时候没有被慎重的考虑
  - 数据的历史记录或改变
- **丢失的数据可能需要推测**

怎么处理丢失的数据?

- 忽略这个元组:

  通常当类标签丢失时这样做(假定是分类任务—当每个属性的丢失数据的百分比变化比较大时就无效)

- 人为填充遗失数据: tedious(冗余) + infeasible（不可行）?

- 自动填充
  - 一个全局变量 :

  例如, “未知”, 一个新的类别?!
  - 属性的均值
  - 属于同一类别的所有样本的均值: 更灵活
  - 最大可能值:

  推理为基础的，如贝叶斯公式或决策树

名字	公司	工资
Amy	A	13000
Alice	A	?
Mike	A	9000
Joey	B	5000
Tom	B	?
Zelda	B	3000

有噪音数据

- 噪音:被测变量中的随机错误或者不一致
- 不正确的属性值可能是由于
  - 有缺陷的数据收集设备（Ex.1：摄像机）
  - 数据输入错误
  - 数据传输问题（Ex.2：监控电视内容）
  - 技术限制
  - 命名约定不一致（命名约定）
- 其他需要数据清理的数据问题
  - 重复的记录
  - 不完整的数据
  - 不连续的数据

怎么处理有噪音的数据?

- Binning（分箱）
  - 先对数据排序并（等频率）分割成箱
  - 然后可以通过箱均值，箱中位数，箱边界值等平滑.
- 聚类
  - 检测和移除
- 结合计算机和人工检查
  - 检测可疑值并人工核准（例如，处理可能的离散点）
- 回归
  - 对数据进行回归函数拟合进行平滑

噪声数据-分箱

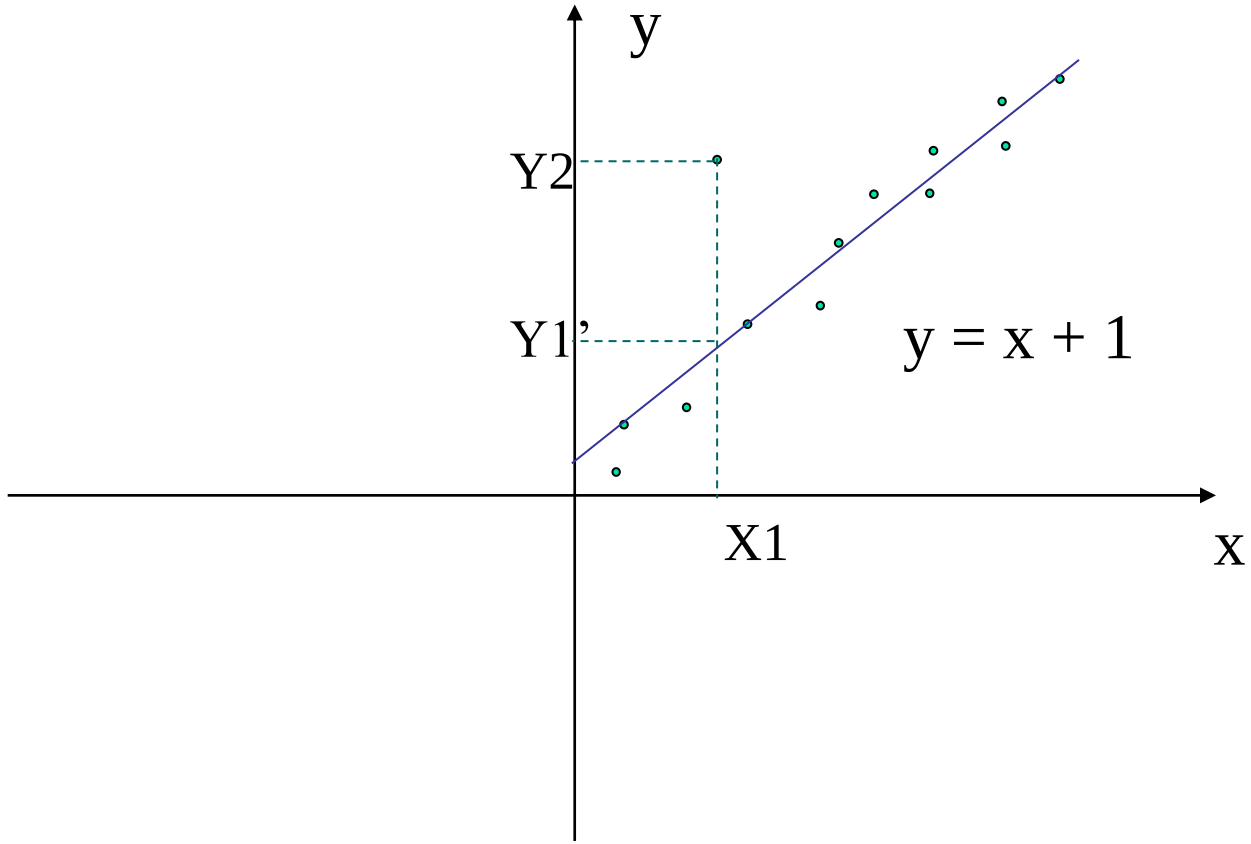
- **等距分区：**
  - 分成N个大小相等的区间：均匀网格
  - 如果A和B是属性值的最小值和最大值，间隔的宽度就是 ： $W = (B - A)/N.$
  - 最简单的，但是离散值可能主导呈现的形式
  - 处理有偏的数据不是很好.
- **等深（频率）分区：**
  - 分成N个区间，每个区包含近似相同数量的样本
  - 好的数据尺度
  - 管理类别的属性可能会非常棘手.

平滑数据的分箱方法

- 按价格对数据排序(以美元计): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

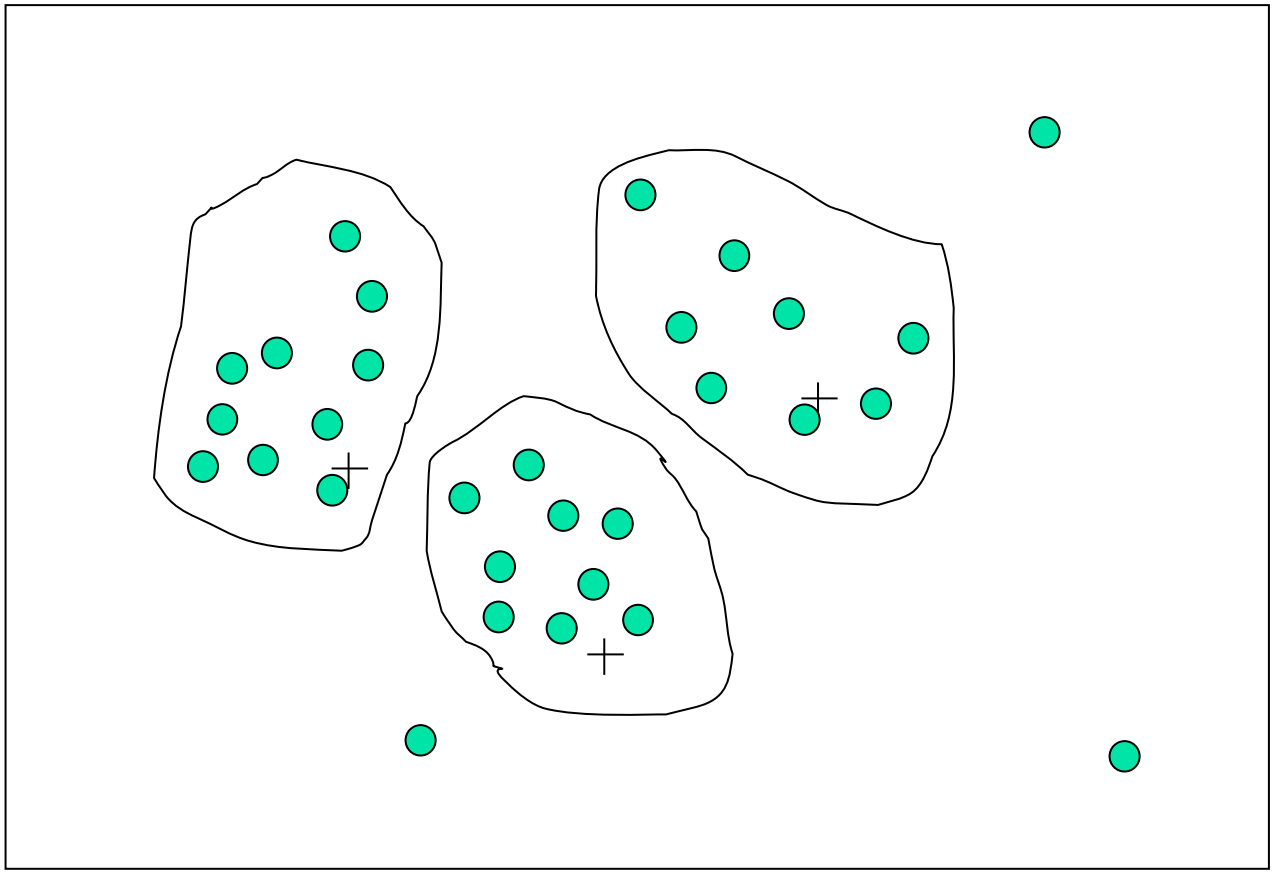
* 分成等频率的箱:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34
* 用箱均值平滑数据:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29
* 用边界值平滑数据:
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

- 一种确定变量依赖的定量关系的分析方法。



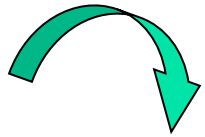
噪声数据-聚类分析

- 将相似的数据聚在一起，不相似的数据分开



数据清洗的过程

- 是两个过程的交替迭代
- 异常数据的检测
  - 用元数据（例如，领域，范围，依赖性，分布）
  - 检查领域超载
  - 检查唯一性原则，连续规则和空规则
  - 使用商用工具
    - Data scrubbing(数据清洗): 使用简单的领域知识 (例如，邮政编码，拼写检查s) 检测错误并修正
    - Data auditing(数据审查): 通过分析数据去发现规则和关系以检测违规 (例如, 相关性和聚类去寻找离散点)
- 数据迁移和整合
  - 数据迁移工具：允许特定的转换
  - ETL（抽取/转换/下载）工具:通过图形用户界面允许用户指定转换



数据集成

- **数据集成:**
  - 把多个数据源的数据合并到一个一致的数据存储(如数据仓库)中
- **模式集成**
  - 整合来自不同来源的元数据
  - 实体识别问题:识别来自多个数据来源的真实世界实体，例如，A.cust-id ≡ B.cust-#
- **检测和解决数值冲突**
  - 对于同一个真实世界实体，来自不同来源的属性值不同
  - 可能的原因：不同的表示，不同的范围，e.g.，十进制 vs. 英式单位
- **两个主要问题：数据冗余和数据转换**

处理数据整合中的冗余问题

- 当整合多个数据库的数据时，多余的数据经常出现
  - *对象识别*: 同一个属性或者对象在不同的数据库或许有不同的名字
  - *导出性数据*: 一个属性可能是另一个表中的派生属性，e.g.，annual revenue（年度税收）
- 多余的属性可以通过相关性分析检测，两种相关性分析工具：皮尔森相关系数和卡方检验
- 仔细的对多个来源的数据进行整合，或许可以帮助减少冗余和不一致，提高挖掘速度和质量

相关性分析（数字数据）

- 相关系数（也叫皮尔逊积矩系数）

$$r_{A,B} = \frac{\sum (A-\bar{A})(B-\bar{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum (AB) - n\bar{A}\bar{B}}{(n-1)\sigma_A \sigma_B}$$

其中， n 是元组的个数， $\bar{A}$ $\bar{B}$ 分别是A和B的均值， σ_A 和 σ_B e分别是A和B的标准差， $\Sigma(AB)$ 是 AB 叉积的和。

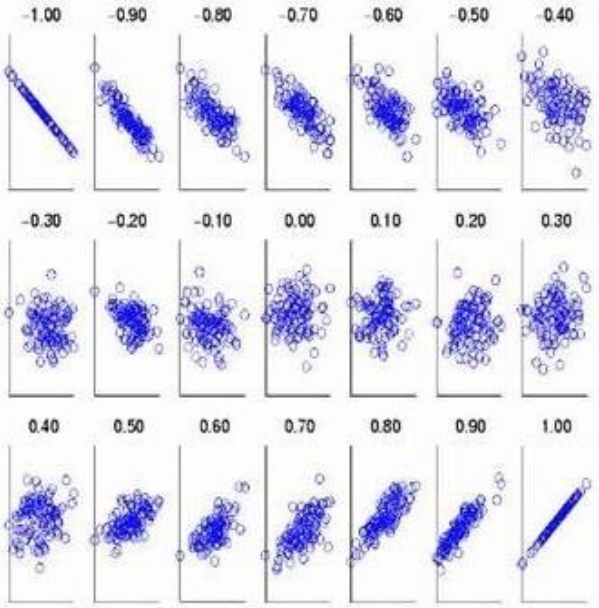
- 如果 $r_{A,B} > 0$

A 和 B正相关

A的值随B的值增加而增加

值越大， 相关性越强

- $r_{A,B} = 0$: 独立;
- $r_{A,B} < 0$: 负相关



Scatter plots showing the similarity from –1 to 1.

相关性分析(类别数据)

- X^2 卡方检验

观测概率　　估计概率

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

- X^2 的值越大，变量相关性越高
- 对 X^2 的值贡献最高的是那些与期望值相差甚远的实际计数
- 相关性并不意味着因果关系
  - # of hospitals and # of car-theft in a city are correlated
  - 两者都与第三个变量相关：人口

卡方计算：一个例子

	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

- 估计概率，假设变量彼此独立：

P(Play chess& Like science fiction)

= P(Play chess)*P(Like science fiction)

= 300/1500 * 450/1500 = 3/50

估计Play chess& Like science fiction的人数：

3/50 * 1500 = 90 人

实际 250人

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

$$\frac{(250-90)^2}{90}$$

求和一共有几项？

卡方计算：一个例子

	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

- X^2 计算(括号中的数字是基于两类数据分布计算出的期望频率)

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

- 它显示在这个小组中，喜欢科幻小说和下象棋有相关性

数据转换

- Smoothing（平滑）：除去数据的噪音
- Aggregation（聚合）:概括统计，构造数据立方体
- Generalization（泛化）：概念层次攀升
- Normalization（规范化）：数据范围变换在一个特定的小的范围内
  - 最小值最大值规范化
  - z-score规范化
  - 小数定标规范化
- 属性构造
  - 对指定的对象构造属性

数据转换：规范化

- 最小值最大值规范化

$$v' = \frac{v - min_A}{max_A - min_A}$$

数据映射到[0,1]的区间

- z-score规范化（μ：均值，σ：标准差）

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

规范化后数据均值为0，标准差为1

- 小数定标规范化

$$v' = \frac{v}{10^j}$$

其中，j 是最小的整数，因此Max(|v'|) < 1

数据归约策略

- **一个数据仓库可能存储了兆兆字节的数据**
  - 复杂的数据挖掘，在完整的数据集上运行，可能会花很长的时间
- **数据归约**
  - 获得数据的简化表达方式，这些数据在容量上更小，但是会产生相同（或几乎相同）的分析结果
- **数据归约策略**
  - 数据立方体聚集
  - 维度缩减—去掉不重要的属性
  - 数据压缩—小波变换
  - 数值归约—将数据拟合到模型中
  - 离散化和观念分层生成

数据立方体聚集

- 数据立方体的最低水平 (base cuboid基数长方体)
  - 为感兴趣的单个实体汇总的数据
  - 例如，分析工资和学历的关系
  - 数据立方体中的多层次聚集
  - 进一步归约要处理的数据
- 参考适当的水平
  - 用最精简的表示方式就足以解决任务
- 在可能的情况下，使用数据立方体回答有关汇总信息的问题

名字	公司	学历	月工资
Amy	A	硕士	13000
Alice	A	本科	10000
Mike	A	本科	9000
Joey	B	硕士	5000
Tom	B	本科	4000
Zelda	B	本科	3000

月平均工资		学历	
		硕士	本科
公司	A	13000	9500
	B	5000	3500

维度规约

- **特征选择**

  选择特征的一个最小集，这样的话，对于给定这些特征的值的不同类别的可能分布，就会与给定所有特征的值的原始分布尽可能的接近

  - reduce # of patterns in the patterns, easier to understand

- **启发式方法**
  - 前向特征选择
  - 后向特征消除
  - 决策树归纳

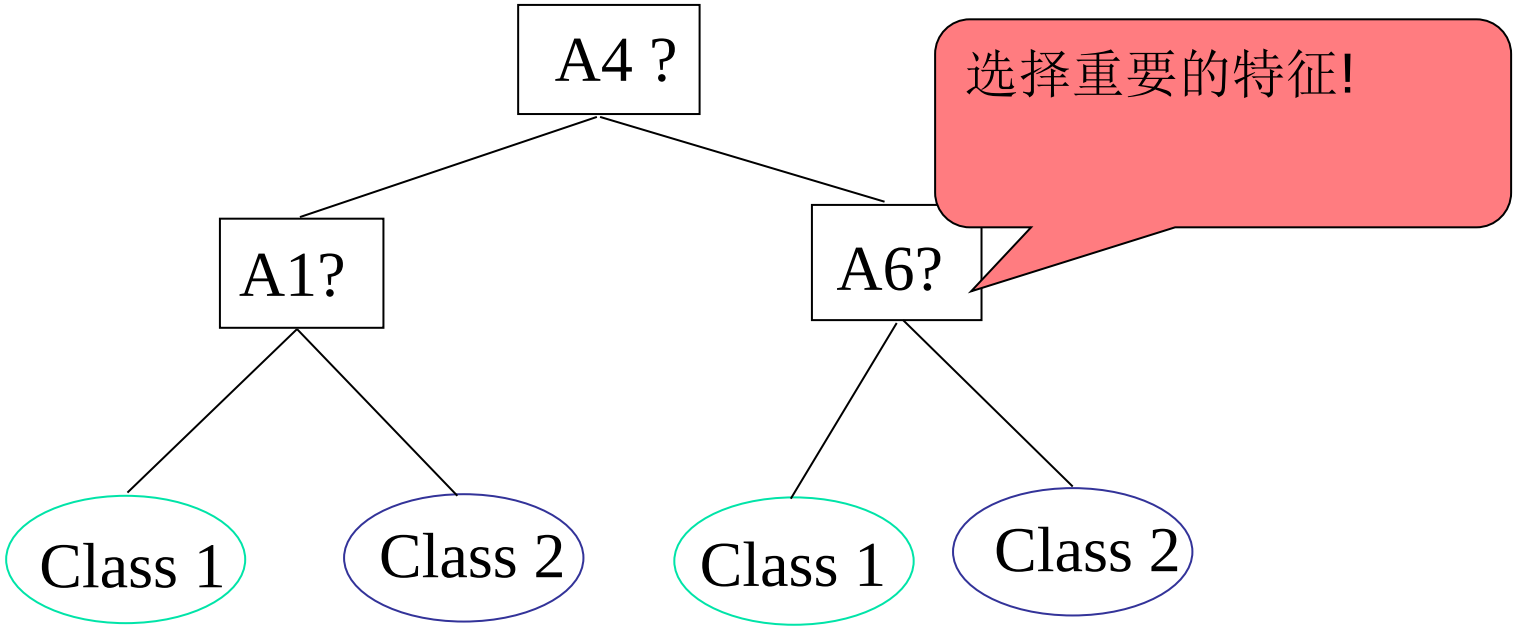
启发式的特征选择方法

- d个特征有2^d个可能的子特征
- 几种启发式的特征选择方法：
  - 特征独立性假设下的最好单特征：通过显著性检验选择
  - 最好分步进行特征选择：
    - 先挑选出来最好的单特征
    - 然后是次好的特征 ...
  - 分步特征清除：
    - 反复清除最糟糕的特征
  - 最好是将特征选择和清除结合起来

决策树归纳的例子

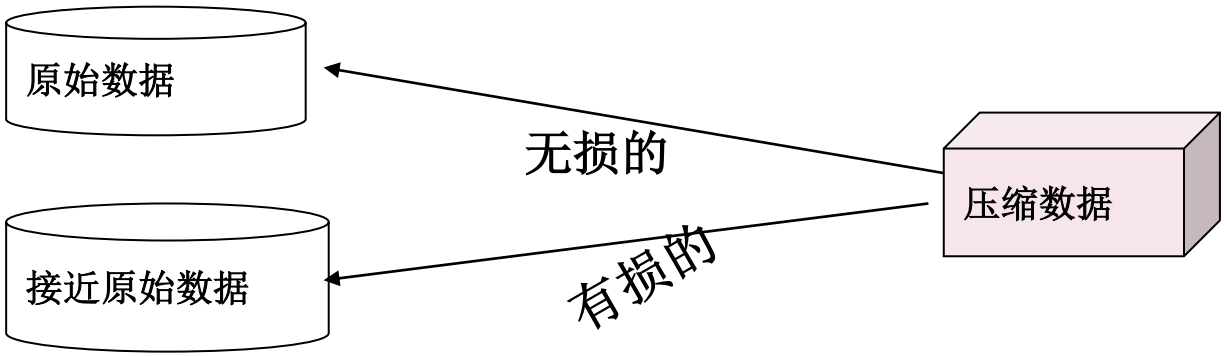
- 初始属性集:

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



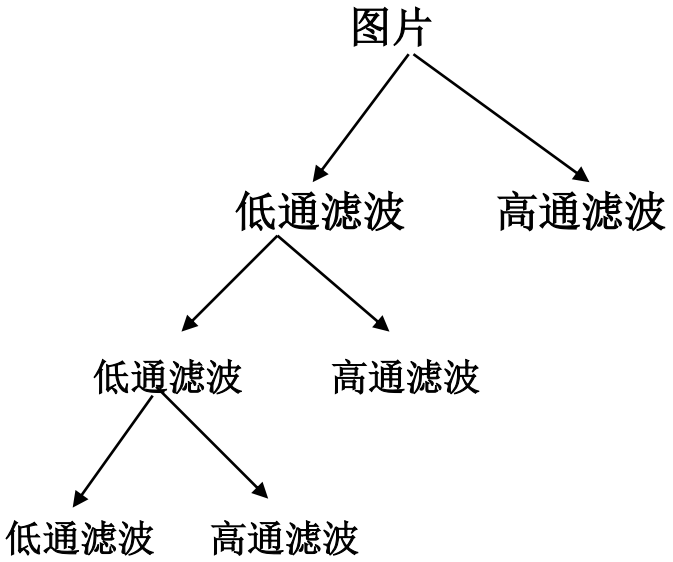
→ 归约的属性集: {A1, A4, A6}

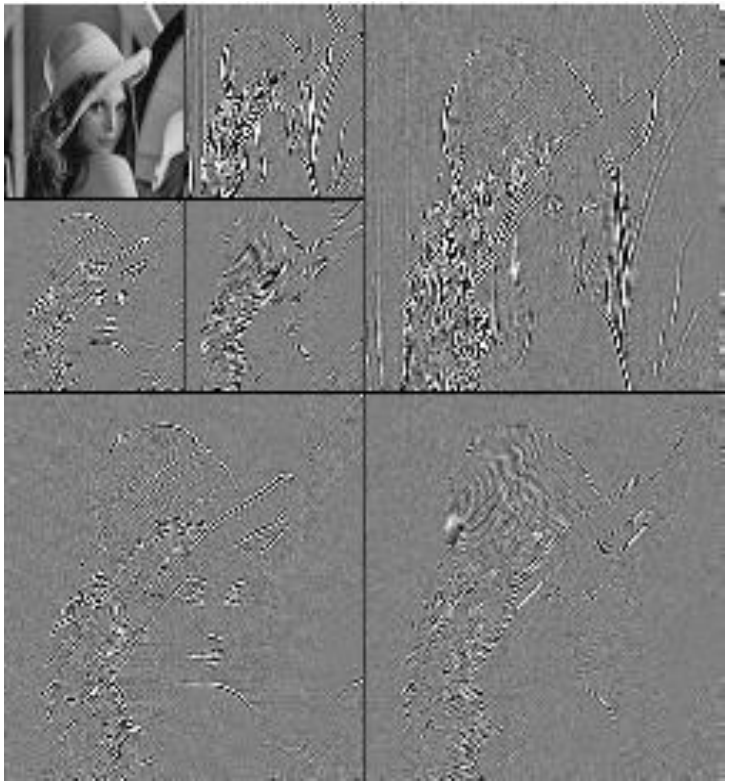
- 有损压缩和无损压缩
- 字符串压缩
  - 有大量的理论和调整好的算法
  - 通常无损的
- 音频/视频压缩
  - 通常是有损压缩, 逐步细化
  - 有时，小片段的信号可以重建，而不用重建整个信号



离散小波变换

- 针对图像压缩
- 压缩逼近:只存储一小部分的最强的小波系数
- 图像质量无损，获得高压缩率





主成分分析

- 给定n维空间中的N个数据向量，找出能最好的表示这些数据的k（$\leq n$）个正交向量
- 步骤
  - 规范化输入数据:每个属性都落在同一范围内
  - 计算 k 个正交向量，也就是，主成分
  - 每个输入的数据都是k个主成分向量的一个线性组成部分
  - 主成分按重要性或者强度降序排列储存
  - 既然成分已经储存好，就可以通过清除弱成分（也就是那些低方差的成分）来减小数据的大小（也就是说,用最强的主成分，可能会重建一个更接近原始数据的数据）
- 只适用于数字数据
- 当维数很大的时候用

数值归约

- 通过选择替代的，较小的数据表示形式来减少数据量
- 参数方法
  - 假设数据符合某个模型，估计模型的参数，仅存储参数，并丢弃数据（除了可能的离散点）
  - 例如： 对数线性模型—obtain value at a point in m-D space as the product on appropriate marginal subspaces
- 非参数方法
  - 不假设模型
  - 主要成员:直方图，聚类，抽样

数值归约-回归和对数线性模型

- 线性回归：将数据拟合成一条直线
  - 线性回归：Y = w X + b
  - 两个回归系数，w 和 b，用手头的数据估计和标记直线
  - 对已知的值Y_1，Y_2，…，X_1，X_2，….用最小二乘法标准
- 多元回归：允许响应变量Y作为一个多维特征向量的线性函数模型
  - 多元回归：Y = b0 + b1 X1 + b2 X2.
- 对数线性模型：近似离散的多维概率分布

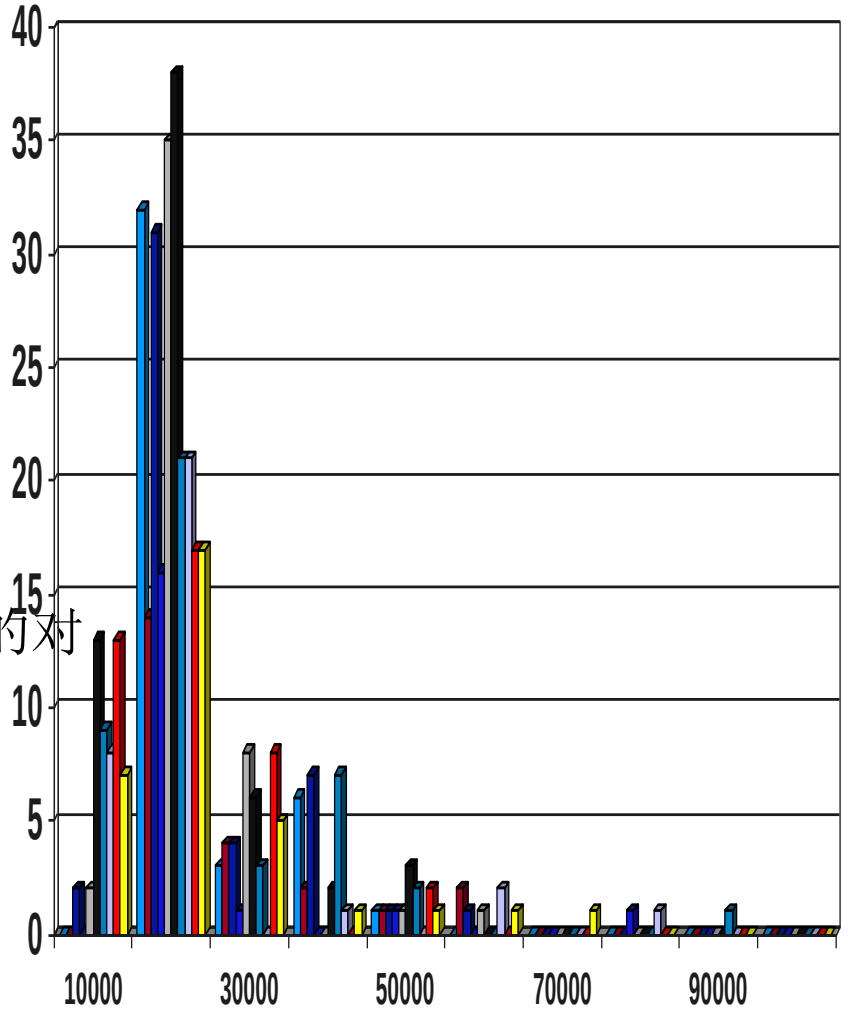
数值归约-直方图

- 将数据分割，存储每一部分的平均值t
- 划分规则：
  - 等宽：相等的间距
  - 等频（或者等深）
  - V-最优：

  最小直方图方差

  (每个篮子代表原始值的加权和)
  - MaxDiff：

  桶的边界是具有β－1个最大差的对



数值归约-聚类

- 在相似性基础上将分区数据设置成聚类，并只储存聚类的表示方式(例如，质心和直径)
- 如果数据是聚类的而不是杂乱的，会非常有效
- 可以有层次聚类和多维索引树结构存储
- 聚类的定义和聚类算法有很多选择
- 在以后的章节中将对聚类分析进行深入的学习.

数值归约-取样

- 允许用数据的小得多的随机样本（子集）表示大型数据集
- 选取数据的一个有代表性的子集
  - 简单随机抽样，可能在存在偏差的情况下效果很差
- 放回简单随机抽样,不放回简单随机抽样，聚类抽样
- 自适应采样方法
  - 分层采样:
    - 整个数据库中每类的近似百分比
    - 在有偏数据的结合中使用

离散化和概念分层

- **离散化**
  - 通过把属性的范围划分成间隔，减少给定的一个连续属性值的数目，然后用间隔标签去代替真实的数据值，例如，薪水，价格，年龄
- **概念分层**
  - 通过用更高水平的概念代替低水平的概念来减少数据，地点-街道-城市-国家

数字数据的离散化和概念层次生成

- 分箱 (见以前章节内容)
- 直方图分析 (见以前章节内容)
- 聚类分析(见以前章节内容)
- 基于熵的离散化
- 自然分区分割

基于熵的离散化

- 样本S，S被边界T分割成两个区间S_1和S_2，分区后的信息增益是

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

- 基于样本在集合里的类分布进行计算。给定m个类，S_1的熵是

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

其中，p_i 是i类在S_1中出现的概率

- 在所有可能的边界中使得信息增益最大的边界被选定为一个划分
- 这个过程是递归应用到所得到的每个分划，直到满足某个终止标准
- 这样的边界或许可以减少数据大小并且提高分类精度

基于卡方检验的离散化

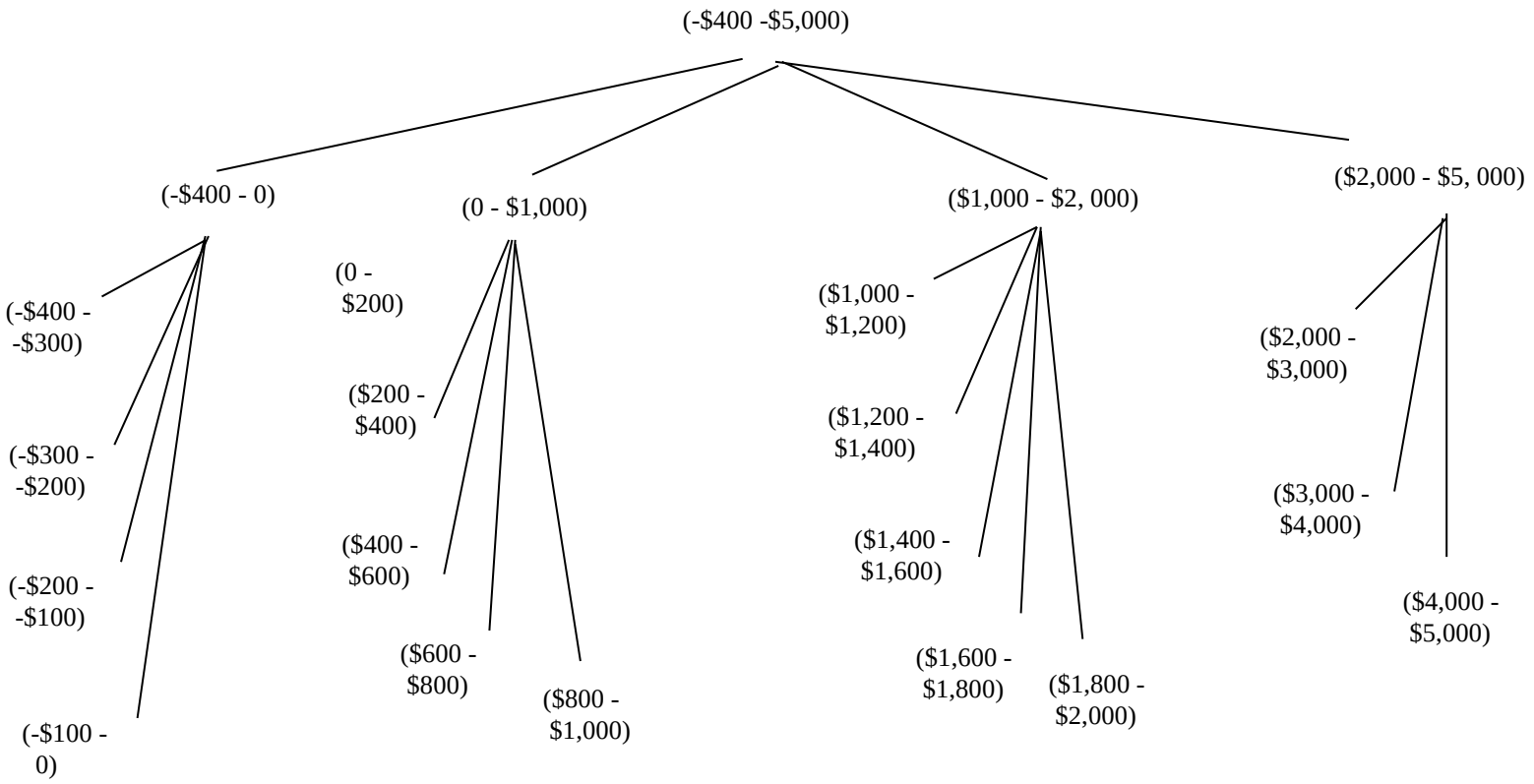
- 合并型vs. 分裂型 方法
- 合并:找到最好的相邻间隔，并把它们递归的合并形成较大的区间
- ChiMerge
  - 最初，每一个数值的属性值被认为是一个区间
  - χ^2 检验适用于每对相邻的间隔
  - 有最小χ^2 值的相邻间隔合并
  - 这个合并过程递归进行，直到满足一个预定义的停止准则（例如，显著性水平，最大间隔，最大不一致，等）

自然分区分割

- 可以用一个简单的3-4-5规则将数字数据分割成相对正式的，“自然的”间隔.
  - 如果一个间隔在最高有效位包含了3, 6, 7或者9个不同的值，将这个范围分成3个等宽的间隔。
  - 如果一个间隔在最高有效位包含了2, 4或者8个不同的值, 将这个范围分成4个间隔
  - 如果一个间隔在最高有效位包含了1, 5或者10个不同的值, 将这个范围分成5个间隔

自然分区分割

- 可以用一个简单的3-4-5规则将数字数据分割成相对正式的，“自然的”间隔.

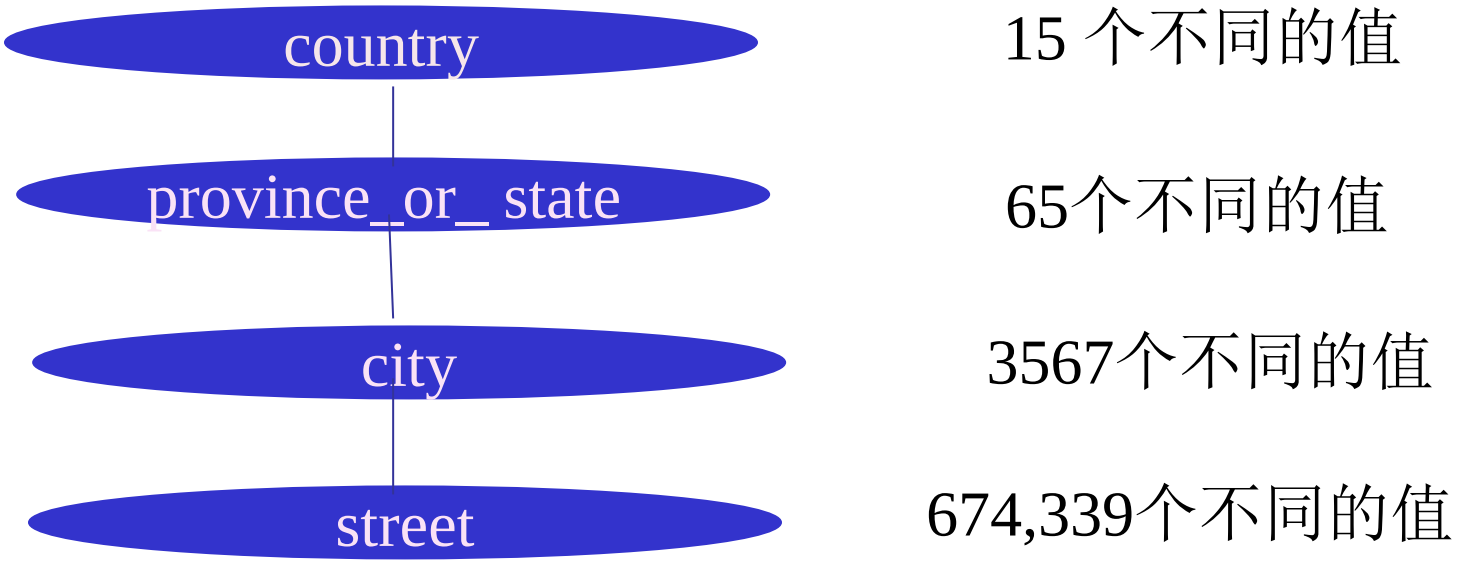


分类数据的概念分层产生

- 属性的偏序由用户或专家在模式级显示的说明
  - 街道<城市<州<国家
- 用显式的数据分组来说明分层结构的一部分
  - {南京，苏州，南通}<江苏
- 一组属性规范.
  - 通过不同值的数量分析，系统会自动生成偏序
  - 例如，街道<城市<州<国家
- 只有部分属性值的规范
  - 例如，只有　街道 <城市，　没有其他属性

自动概念分层生成

- 在给定的数据集的每个属性的不同值的数量分析的基础上，可以自动生成一些概念层次
  - 有最多不同值的属性被放置在最底层
  - 注意：例外—工作日，月，季，年



小结

- 对于数据仓库和数据挖掘，数据准备或者数据预处理都是一个大问题
- 描述性的数据汇总对于高质量的数据预处理是必需的
- 数据预处理包含
  - 数据清理和数据整合
  - 数据简化和特征筛选
  - 离散化
- 虽然已经开发了很多方法，但数据预处理仍然是一个热门的研究领域

Thank you !!!