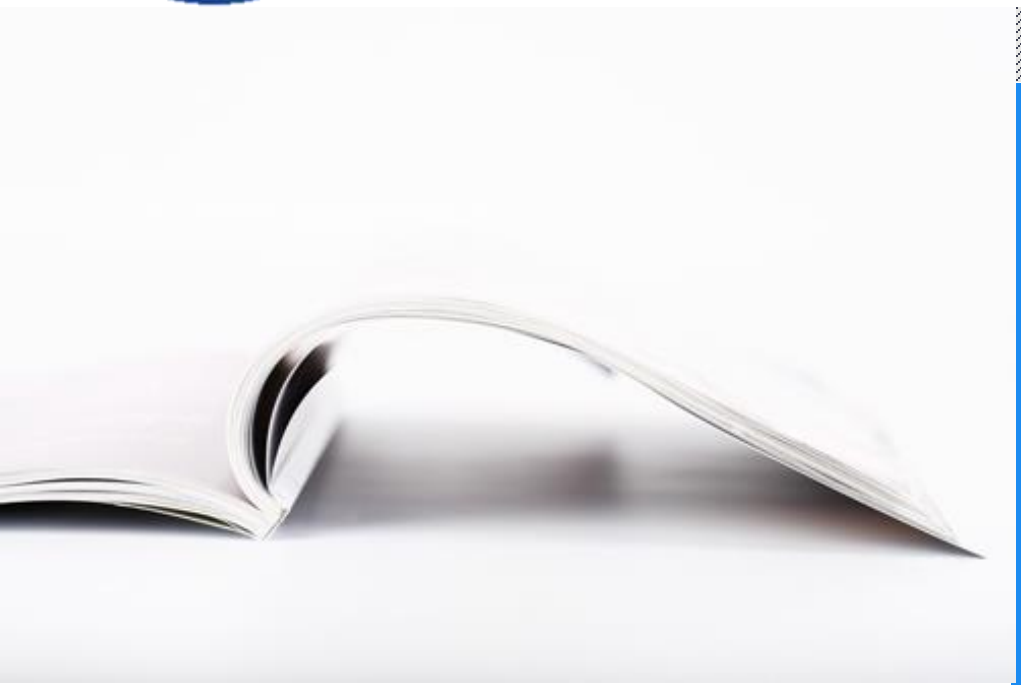


Web信息处理与应用

第十一节

数据准备

徐童

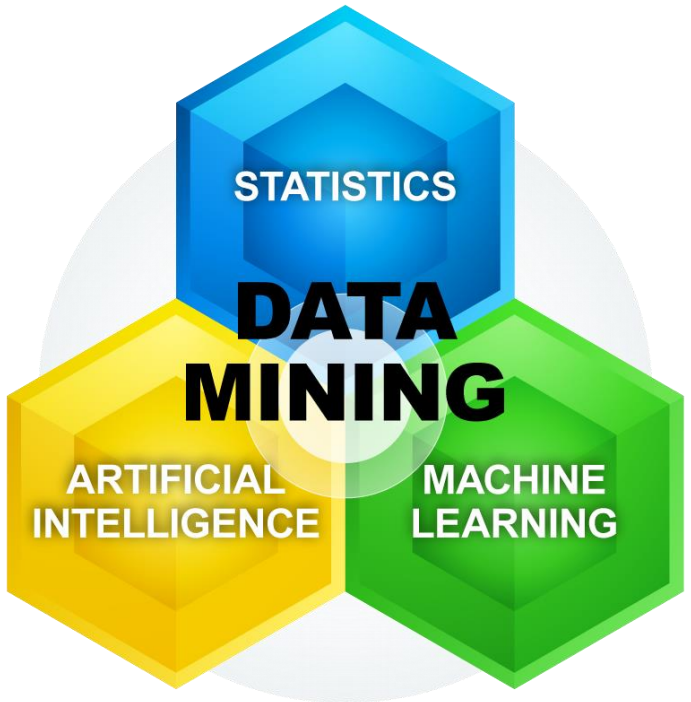
2021.11.22

- 数据挖掘（Data Mining）

- 基本含义：从海量数据中提取或挖掘潜在的知识和规律，用于支持当前的判断或未来的决策
  - 数据准备：筛选、清理并整合有待挖掘的数据
  - 数据建模：使用智能方法建模数据并提取规律
  - 知识表示：以用户能理解的方式展示所得知识

- 随着数据捕获、传输和存储技术的快速发展，用户将更多地需要采用新技术来挖掘数据的价值

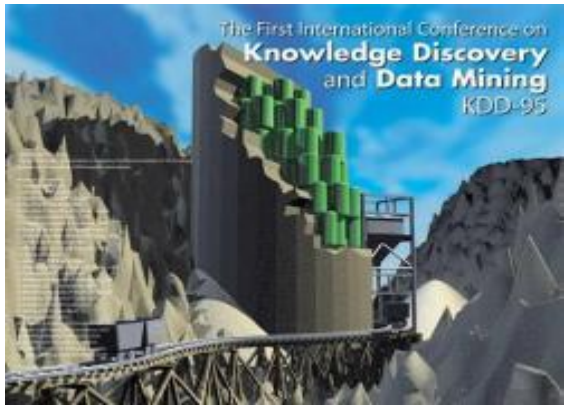
—— Gartner Group，2011



- 数据挖掘的发展历史

- 1989年8月，第11届国际人工智能联合会议（IJCAI）组织了知识发现专题讨论会（IJCAI-1989 Workshop on KDD）
  - 首次出现了KDD这一术语，第二个D仍指Database
  - 本次讨论会参与人数30人
  - 1990年第二届讨论，参与人数46人
- 1995年，首届知识发现与数据挖掘国际会议召开于加拿大蒙特利尔召开
  - 第二个D已由Database改为Data Mining
- 1993年，国家自然科学基金首次资助KDD相关研究项目

KDD 1995



会议
论文集

- 数据挖掘 vs. 数据库 vs. 信息检索
- 以一家大型超市为例

数据库：

进货单、价目表……

信息检索：

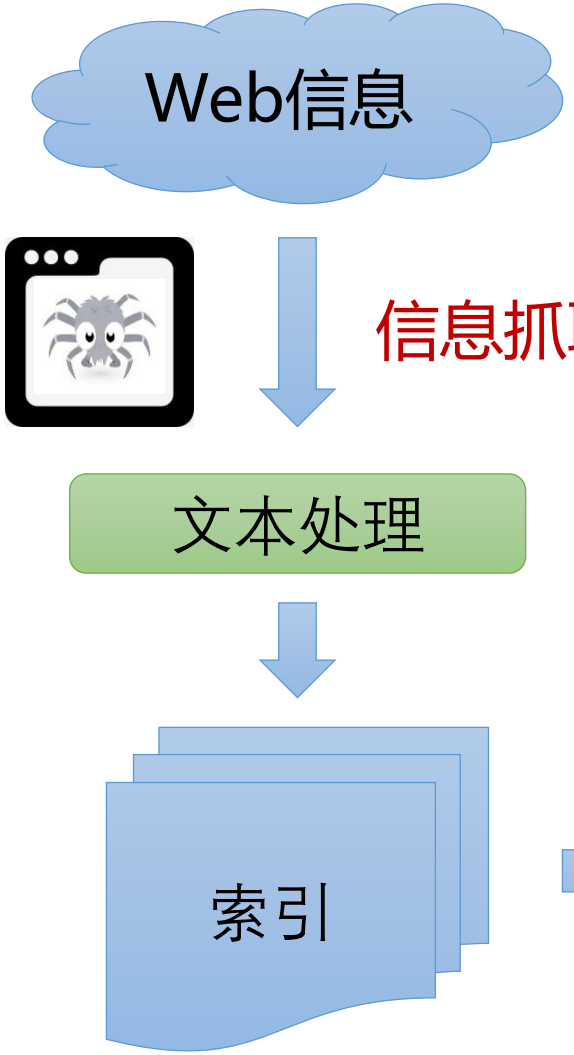
最符合顾客需求的是？

数据挖掘：

啊！啤酒和尿布！

本节主题

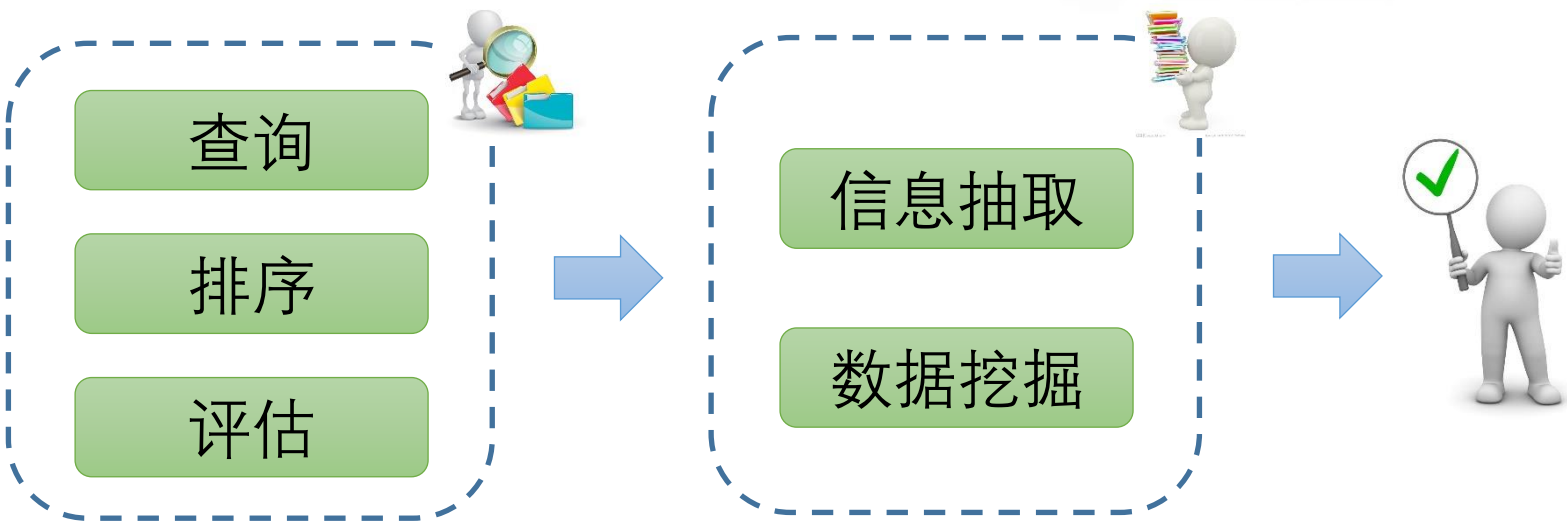
- 本课程所要解决的问题



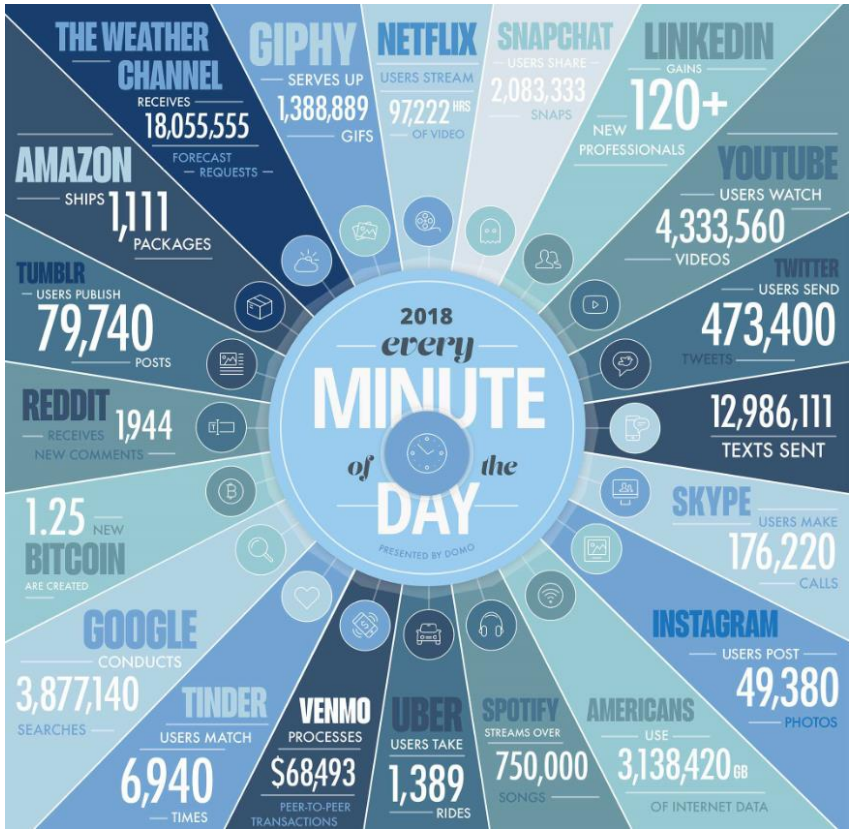
第十个问题：

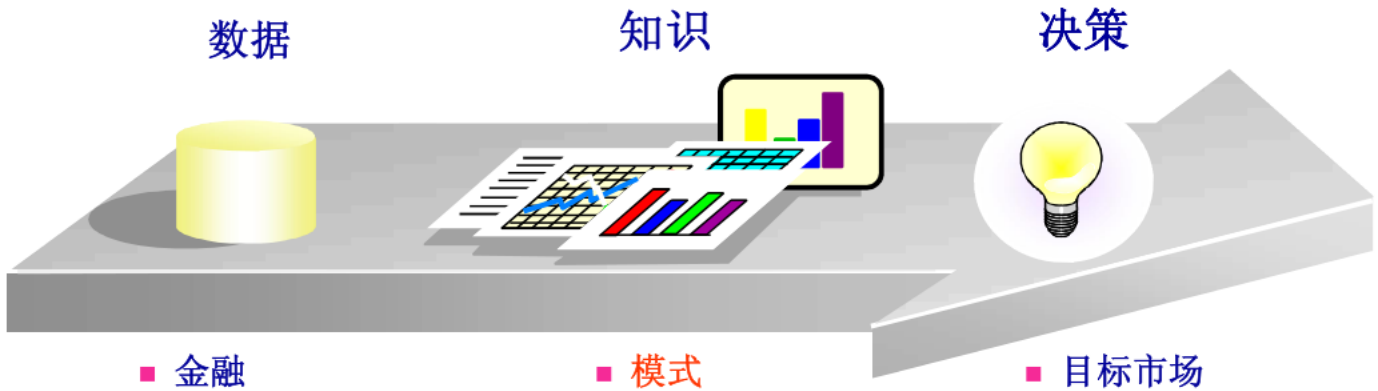
如何从数据中挖掘潜在规律，

指导决策行为？



- **海量数据催生的数据挖掘**





- 金融
- 经济
- 政府
- POS.
- 人口统计
- 生命周期
- 科学实验和观测
- 邮件
- 图像、视频……

- 模式
- 趋势
- 事实
- 关系
- 模型
- 关联规则
- 序列

- 目标市场
- 资金分配
- 贸易选择
- 在哪儿做广告
- 销售的地理位置
- 电信客户的社会网络
- 恐怖分子的联系网

真相与规律淹没在数据中，难以制定高效且合适的决策

- **我们希望什么样的数据挖掘?**

- 数据挖掘的目的在于揭示满足以下条件的模式（Pattern）或模型（Model）
  - 有效性：在处理新数据时具有足够的可信度
  - 实用性：可以用于实际的项目并产生指导价值
  - 解释性：能够被人们以经验或领域知识所理解
  - **意外性**：所得结果并不是那么直观和显而易见
    - 把握尺度的重要，既要有一定深度，又不能过于颠覆

- 围绕“检索”、“抽取”与“挖掘”三条主线

第三部分：面向Web信息的数据挖掘

第11节 数据准备

基本数据挖掘方法

第12节 分类算法

第13节 聚类算法

Web信息应用

第14节 推荐系统

社会网络分析方法

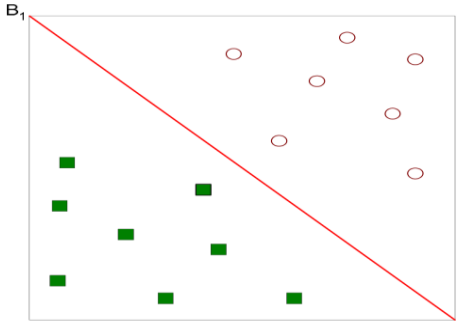
第15节 社会网络

第16节 社会传播

本节目录

- **数据挖掘方法**
  - 关联规则
  - 异常检测
- 数据预处理

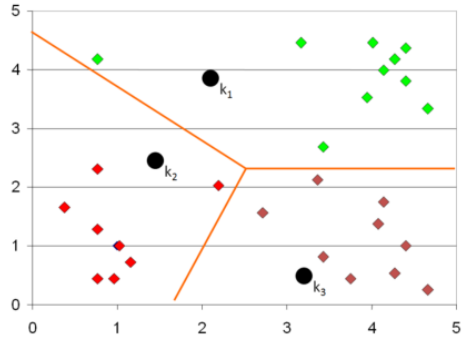
- 数据挖掘的基本方法



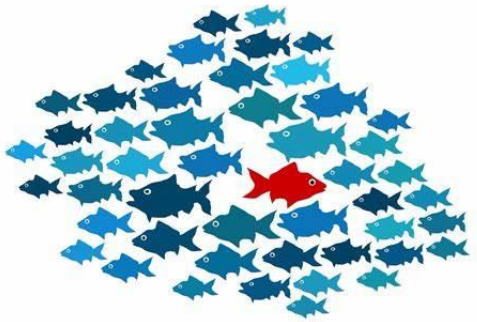
分类

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

关联规则



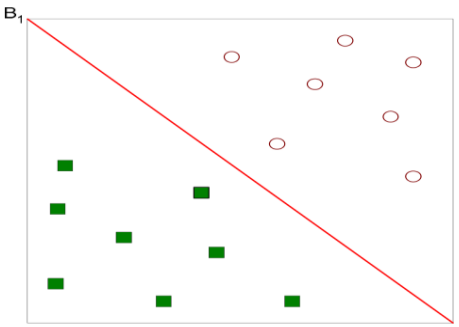
聚类



异常检测

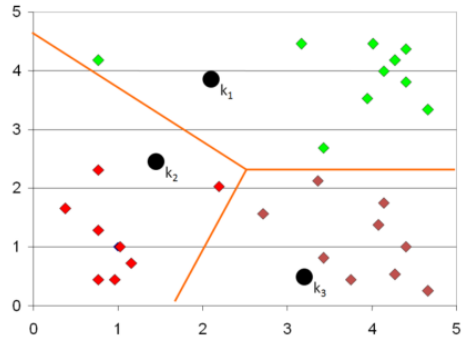
- **常用方法（1）分类**

- 给定一组有标签记录（作为训练集），其目标在于训练一个合适的模型，使其能够有效地区分无标签的新数据，将其归为合适的类别
  - 有监督学习，面向预定义的类别
  - 分类问题是我们的老朋友了（传统艺能）
    - 在各种信息检索问题中广泛存在，因其丰富的解决方案，而成为诸多问题转化的对象
    - 将在第十二节课中详细介绍各种常用方法



• 常用方法（2）聚类

- 与分类问题相对应，作为无监督学习的代表
  - 没有预先定义的类别，而是借助相似性度量自动生成
  - 五花八门的聚类方法，很多来源于不同的相似性度量
  - 聚类问题主要的挑战：
    - 如何确定合适的类别数量？
    - 如何验证结果的合理性？
    - 将在第十三节课中详细介绍各种常用方法



本节目录

- 数据挖掘方法
  - 关联规则
  - 异常检测
- 数据预处理

- **预备知识：事务型数据**

- 事务型数据（Transaction Data）是一类特殊的数据记录
  - 一条记录往往对应着一个项目（Item）的集合（无序）
  - 例如，超市的购物记录，一般同时购买多项商品
    - “啤酒+尿布”小故事的出处

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

• 常用方法（3）关联规则

- 用户需要的信息往往并非孤立存在，而是彼此关联
  - 可能是用途上的搭配，例如买香皂之后还需要买香皂盒
  - 也可能是兴趣上的拓展，例如寻找某部电影主角的其他电影
- 关联规则（Association Rule），旨在分析事务型数据，从而根据一部分项目的存在记录，来判断另一部分项目是否同时存在于事务中

Market-Basket transactions

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

Example of Association Rules

{Diaper} {Beer},
{Milk, Bread} {Eggs,Coke},
{Beer, Bread} {Milk},

Implication means co-occurrence,
not causality!

- **两个基本的指标**

- 关联规则的基本形式：$A \rightarrow B$，A、B均为集合形式
- 如何判定一条好的关联规则？
  - 回顾一下：基于模式的关系抽取中的Snowball算法
  - 两个指标：支持度（Support）、置信度（Confidence）
    - 支持度：{A+B}在全体事务中的比重 $$s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N}$$
    - 置信度：{A+B}占A出现的事务中的比重 $$c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$$

- **两个基本的指标：实例**

- 支持度与置信度的计算实例
  - 给定事务数据集如右表
  - 考虑{Diaper, Milk} → Beer 这一关联规则

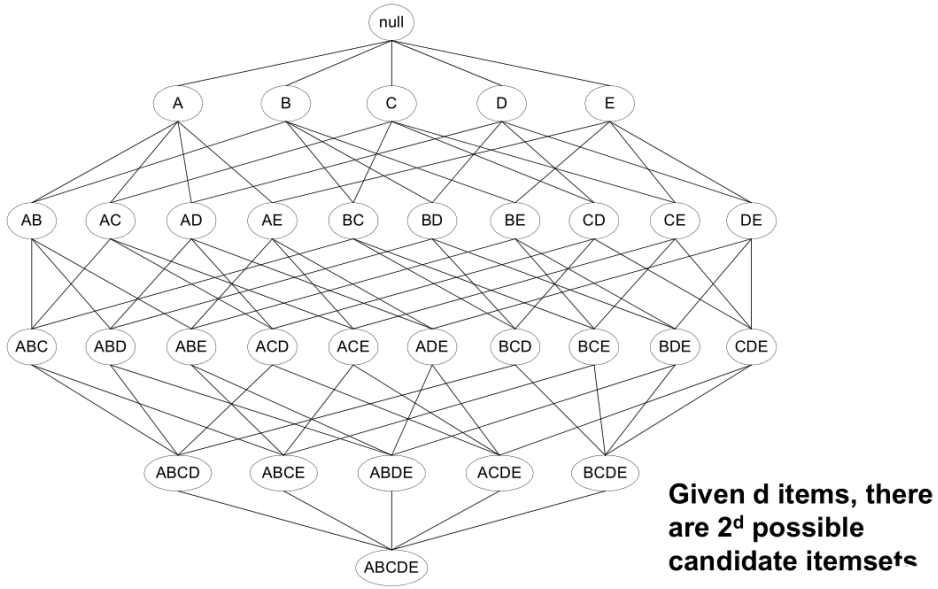
TID	Items
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

$$s = \frac{(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{(\text{Milk}, \text{Diaper}, \text{Beer})}{(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

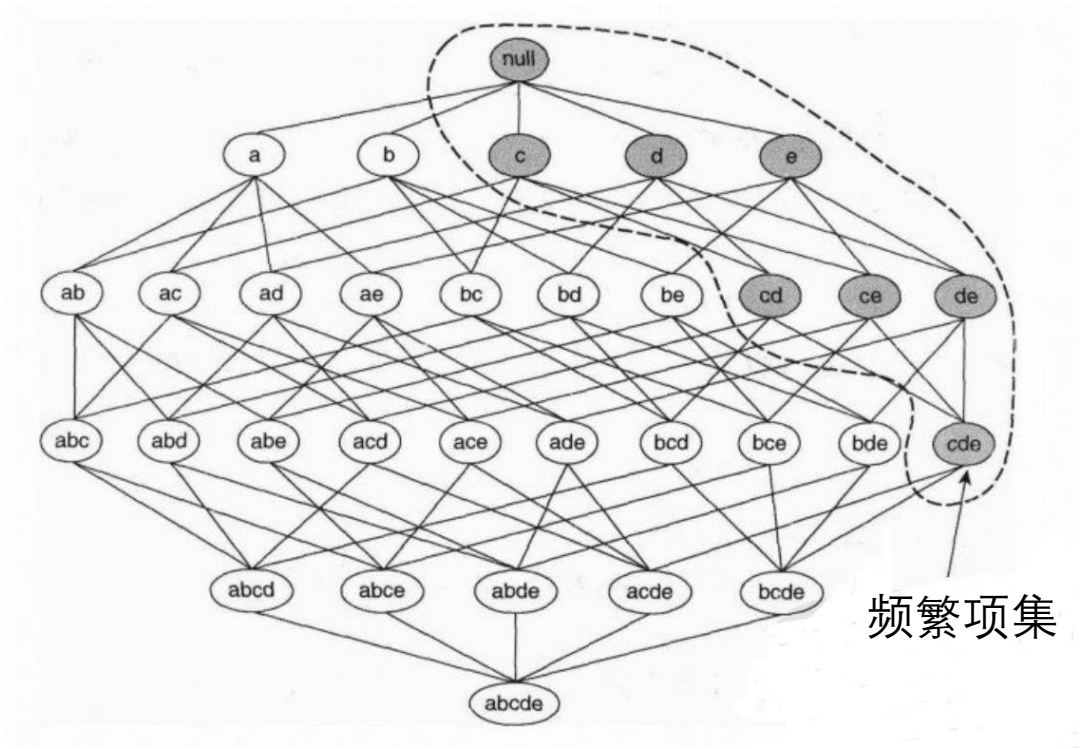
- **基本的频繁项集生成方法**

- 频繁项集（Frequent Itemset），即支持度高于阈值的项目集合A
  - 最基本的办法：穷举所有的可能集合，并计算其支持度
  - d 个项目对应着2^d-1个潜在的频繁项集，复杂度过高！



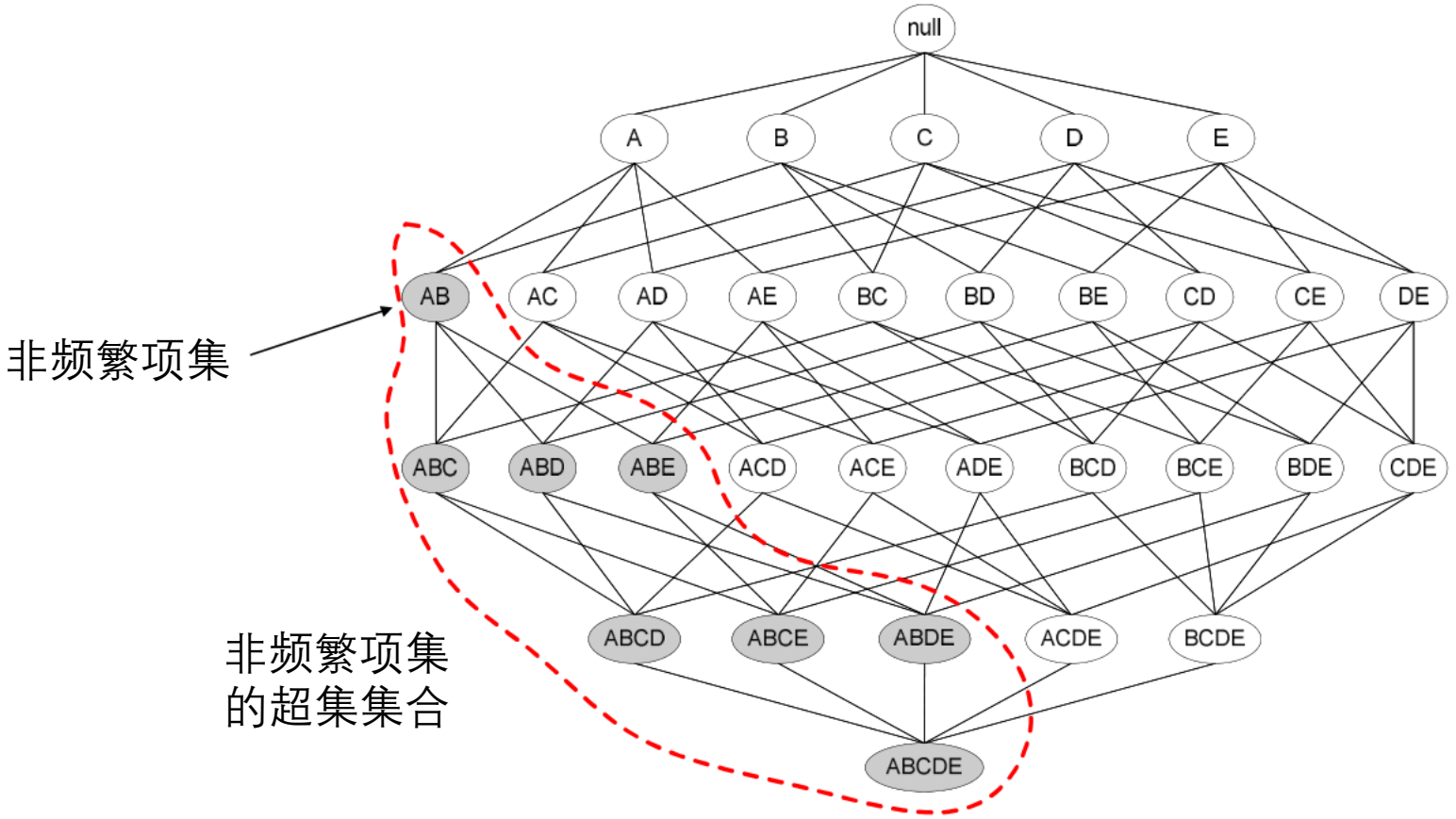
- **改进方法（1）：Apriori**

- 一些原理将有助于我们减少候选的项目集数量
  - 先验原理：如果一个项集是频繁的，那么它的所有子集也是频繁的



- **改进方法（1）：Apriori**

- 反过来理解先验原理：非频繁的项集，其所有超集也是非频繁的



- **改进方法（1）：Apriori**

- Apriori算法：逐步减去所有的非频繁项集，然后基于频繁项集生成其超集

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

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Items (1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Pairs (2-itemsets)

(No need to generate candidates involving Coke or Eggs)

Minimum Support = 3

Triplets (3-itemsets)

Itemset	Count
{Bread,Milk,Diaper}	3

If every subset is considered,
$^{6}C_{1} + {}^{6}C_{2} + {}^{6}C_{3} = 41$
With support-based pruning,
6 + 6 + 1 = 13

• 改进方法（2）：FP-Growth

- Apriori算法减少了对于候选项集的遍历，但仍是一种“生成-测试”的范式
- 如何进一步减少空间开支，同时直接从结构中提取频繁项集？
  - FP-增长（FP-growth）算法由数据挖掘领域泰斗韩家炜教授提出

    J Han, et al., Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach, DMKD, 2004

  - 本质是一种输入数据的压缩表示，通过逐个读入事务，并将事务映射到FP树中的某条路径来构造。

- **改进方法（2）：FP-Growth**

- FP-Growth算法的建树过程
  - 首先，对各个项（Item）按照支持度进行排序（为什么要排序？）
  - 其次，将排序后的项集逐步读入并建立树状结构（假设顺序为abcde）

TID	项
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

图1

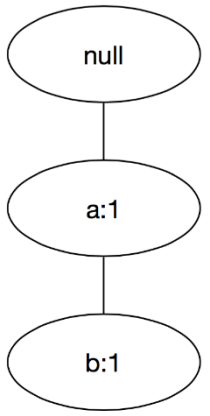
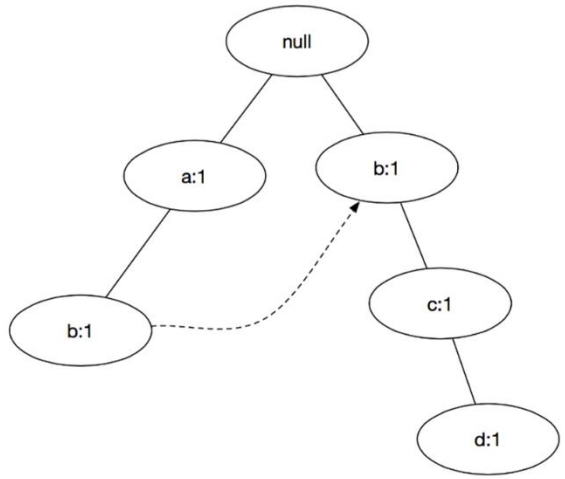


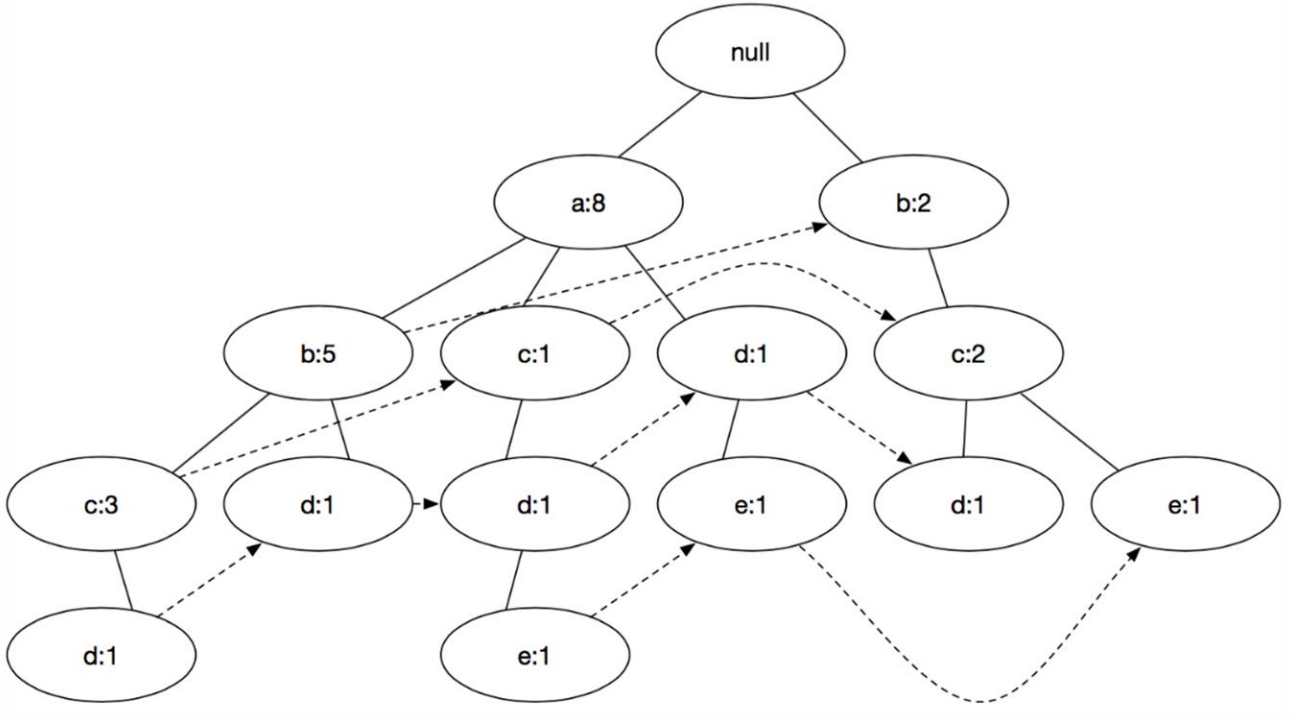
图2



- **改进方法（2）：FP-Growth**

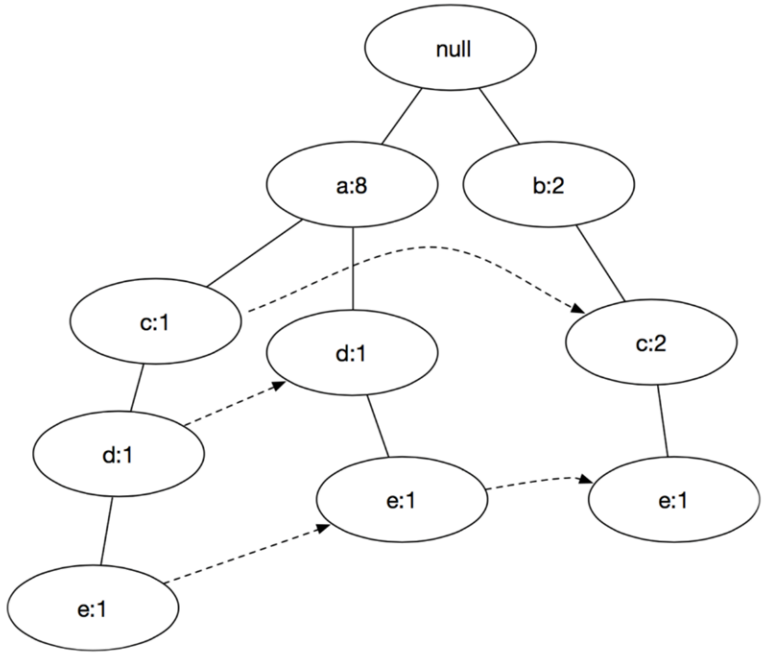
- FP-Growth算法的建树过程
  - 在建树的过程中，对相同项节点采用指针连接，方便快速访问

TID	项
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}



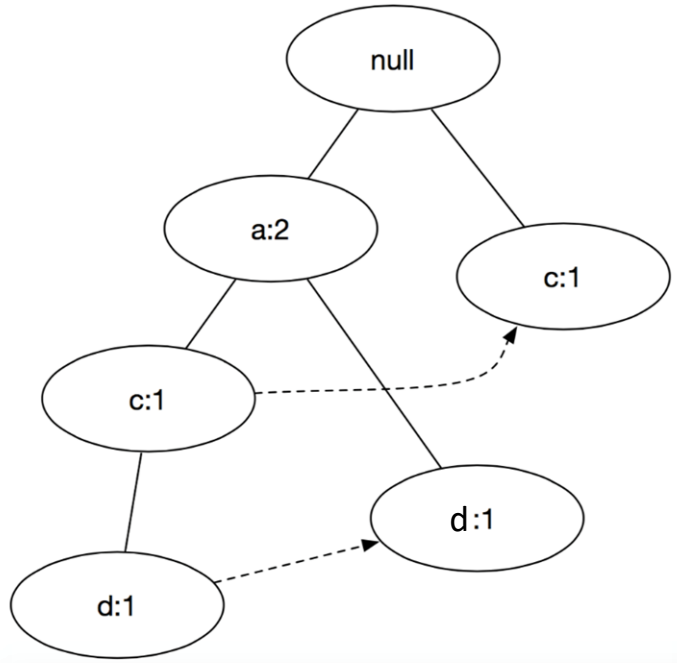
- **改进方法（2）：FP-Growth**

- 基于FP-树，生成频繁项集
  - 本质上说，FP-Growth是一种自底向上的探索方式
    - 首先查找以 e 为结尾的频繁项集为例，其次是d / c / b，最后是 a
  - 以包含 e 的路径为例，如右图



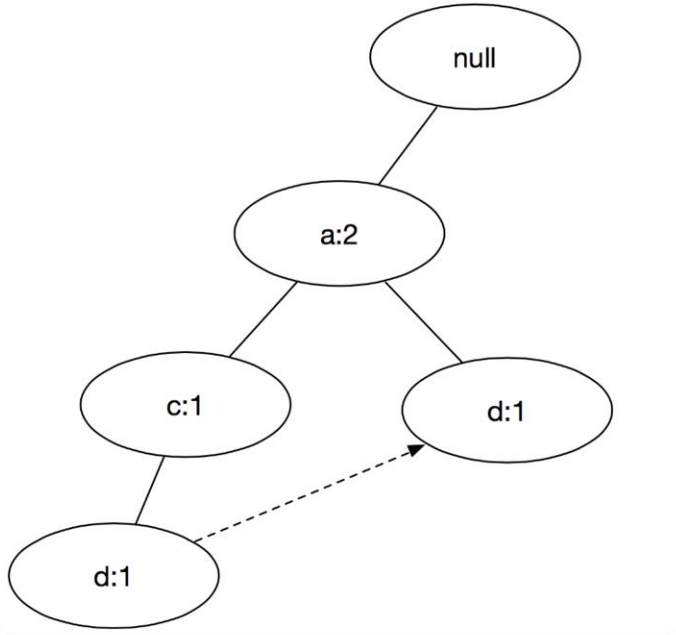
- **改进方法（2）：FP-Growth**

- 基于FP树，生成频繁项集
  - 首先，判定e本身是否为频繁项集（此处设阈值为2，高于阈值）
  - 其次，将 e 的前缀路径转化为条件FP树
    - 其中，需要更新路径上的支持度计数
      - 显然，只有包含 e 的事务会被统计
    - 在这一过程中，删除那些非频繁的项（例如 b）



- **改进方法（2）：FP-Growth**

- 基于FP树，生成频繁项集
  - 在此基础之上，考虑更长结尾的频繁项集的子问题
  - 以 {de}为结尾的频繁项集判定为例
  - 首先，在前一张图上统计与 d 相关的支持度求和
    - 支持度为2，为频繁项集
  - 其次，以de为结尾，得到其前缀路径如右图
    - 通过其条件FP树，发现{ade}支持度为2，也频繁

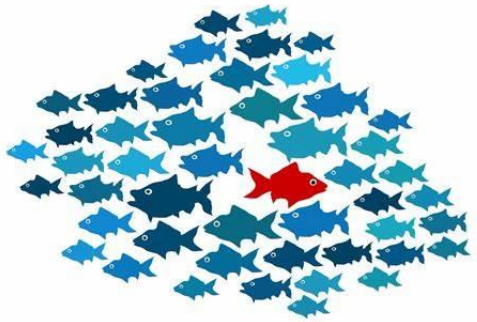


本节目录

- **数据挖掘方法**
  - 关联规则
  - **异常检测**
- 数据预处理

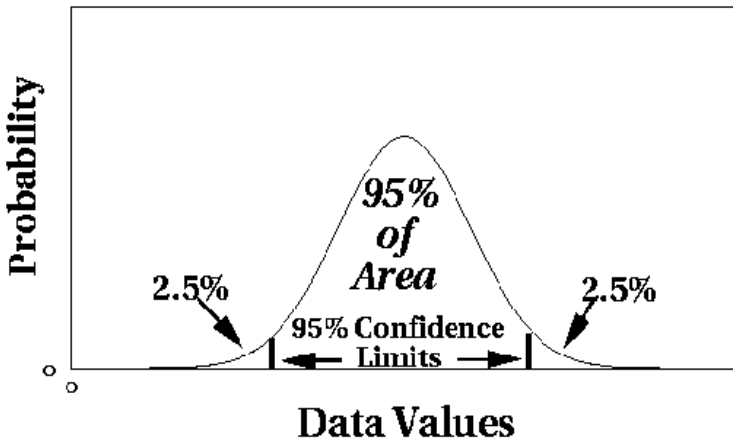
• 常用方法（4）异常检测

- 异常检测（Outlier Detection），旨在发现与大部分其他对象不同的数据
  - 异常数据不等于错误数据！而是包含着不同于寻常规律的数据
  - 异常是相对的：并不意味着数量绝对的小
    - 例如，美国的罕见病比例限制为不高于1/1500，但也可达约20万人
  - 常见的异常检测应用场景
    - 欺诈检测：例如银行中的异常交易
    - 智慧医疗：对于某些疾病的诊断
    - 入侵检测：网络中的异常行为判断



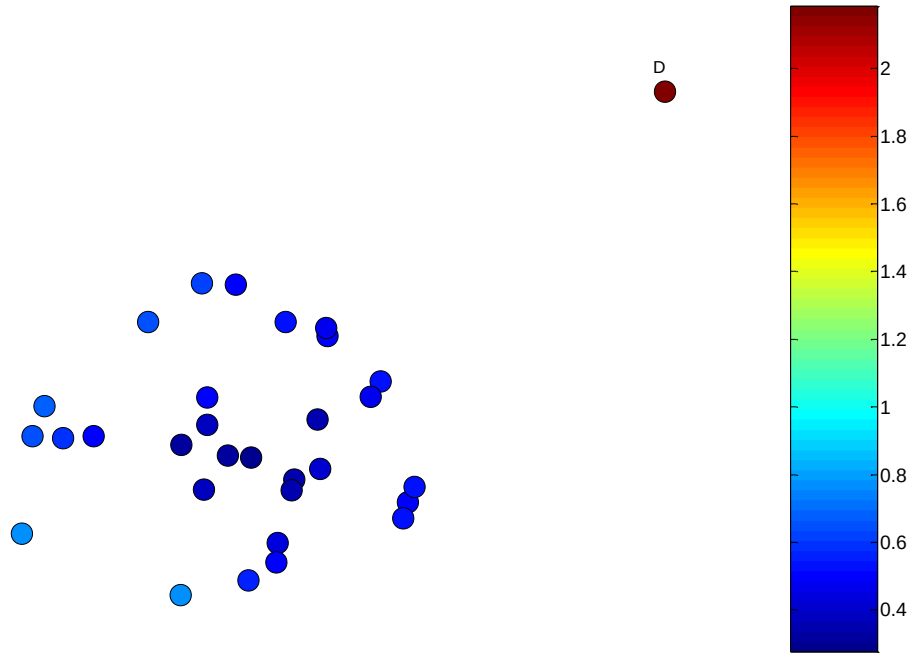
- **基础的异常检测方法：基于分布**

- 统计方法是面向异常检测的最基本方法
- 采用此方法的前提是识别数据集的具体分布，错误识别会导致错误检测
- 基础方法：基于一元正态分布的离群点判定
  - 在已知参数的前提下，可根据正态分布判定离群的概率
  - 进而，可以将一元正态分布拓展至多元正态分布
  - 一个有趣的应用：舆情监测
    - 严重不平衡条件下，通过学习正常值分布进行



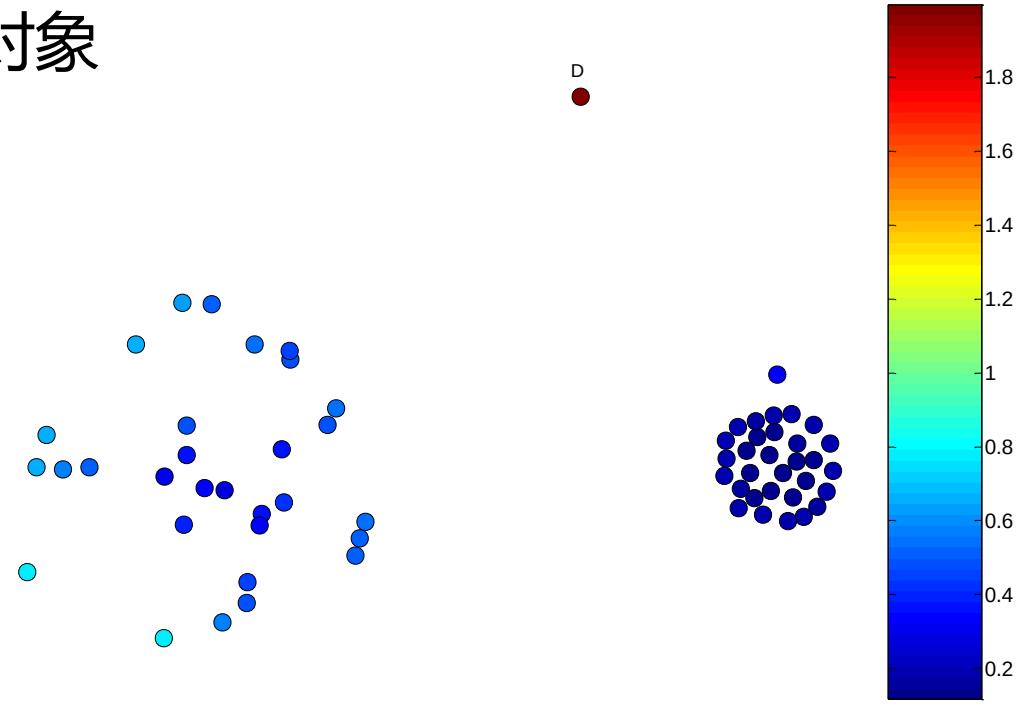
- **进阶的异常检测方法：基于度量**

- 异常点与大部分其他对象不同，意味着在空间中将远离大多数数据点
- 因此，采用基于距离或密度的方式，可以实现离群点检测
- 例如，计算数据点到K最近邻的平均距离
  - 高于阈值则判定为异常点
- 类似地，可以采用基于密度的方式
  - 采用K近邻距离的倒数作为密度
  - 也可采用给定半径内的点的个数



- **更进阶的异常检测方法：基于聚类**

- 聚类分析可以发现强相关的数据集合，而与其他数据不相关即为离群点
- 一种方法是：抛弃远离其他簇的小簇，但簇的个数将影响结果
- 另一种方法是，先聚类所有对象，再评估对象属于簇的程度，往往采用以下两个指标
  - 点到簇中心的距离
  - 点到簇中心的相对距离
    - 与所有点到中心距离的中位数之比
    - 从而调整簇密度对距离造成的影响



本节目录

- 数据挖掘方法
- **数据预处理**
  - 数据聚合
  - 数据采样
  - 数据归约
  - 数据离散

- **为什么要进行数据处理?**

- 糟糕的数据质量将给数据挖掘过程造成严重的负面影响

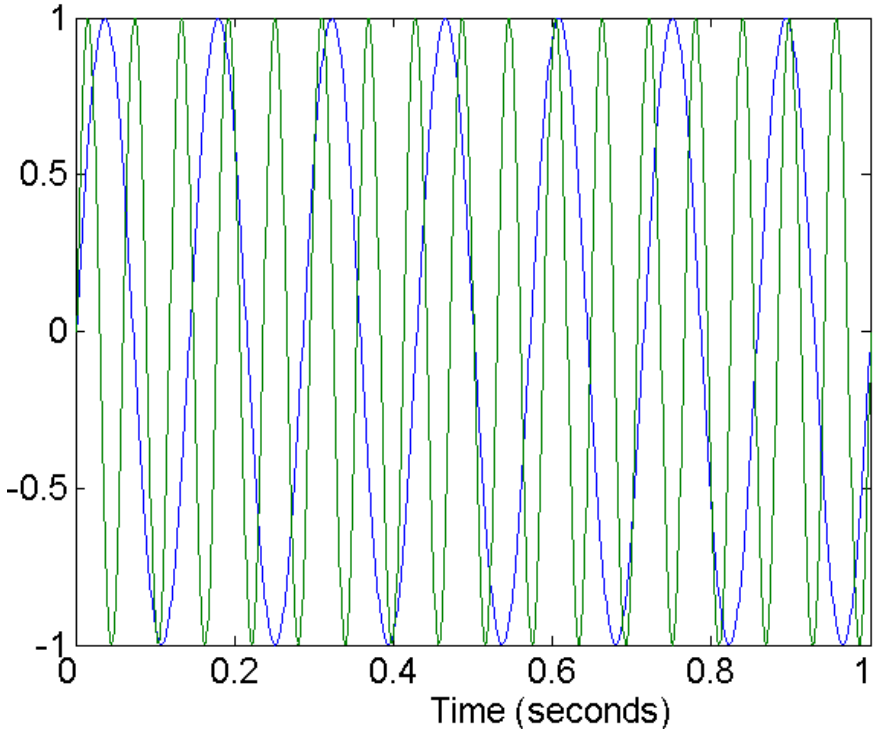
“The most important point is that poor data quality is an unfolding disaster. Poor data quality costs the typical company at least ten percent (10%) of revenue; twenty percent (20%) is probably a better estimate.”

—— Thomas C. Redman, DM Review, August 2004

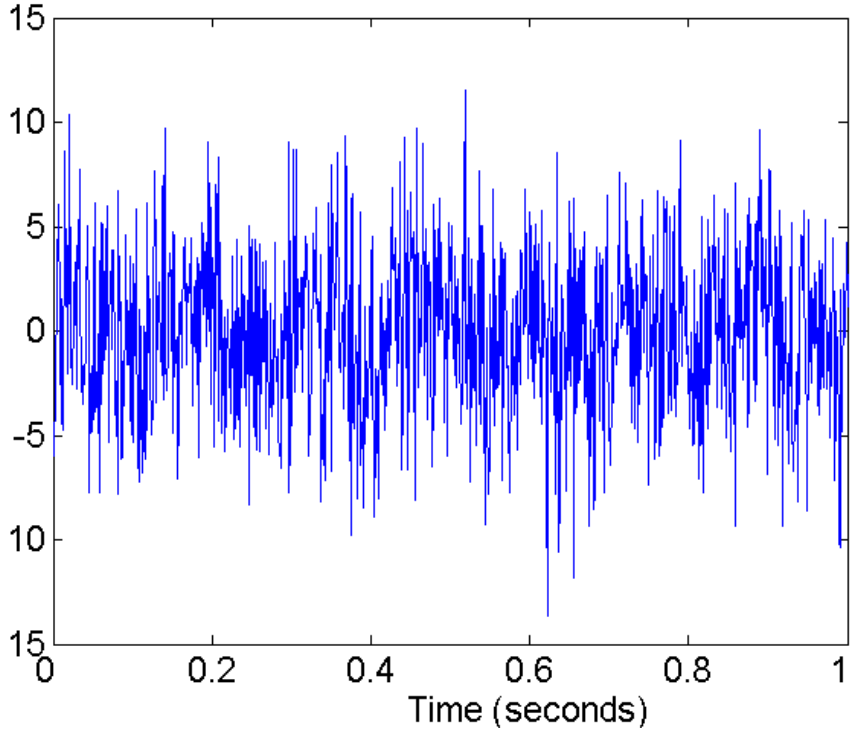
- 常见的数据质量问题
  - 数据测量、采集等过程中出现的错误
  - 噪声、离群点、缺失值等数据问题
  - 重复数据

- **常见数据问题（1）：噪声数据**

- 噪声数据往往表现为原始数据的微调或者背景信号的堆叠
  - 例如，嘈杂环境下的通话声音，各种震动干扰下的地震波形



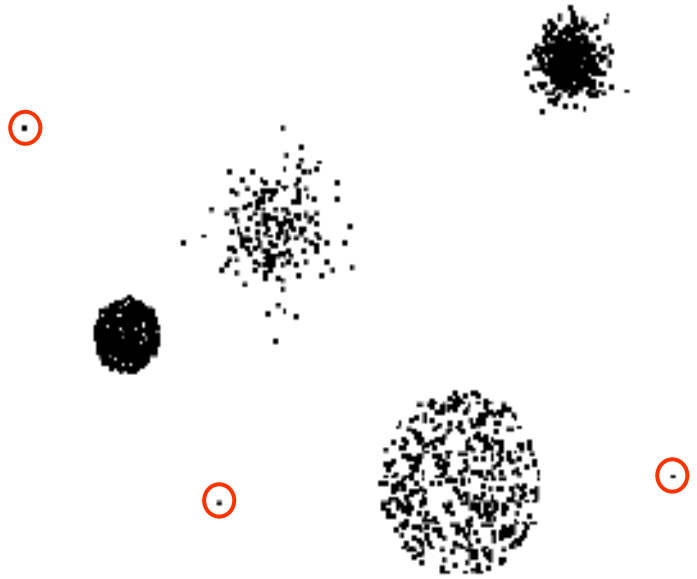
Two Sine Waves



Two Sine Waves + Noise

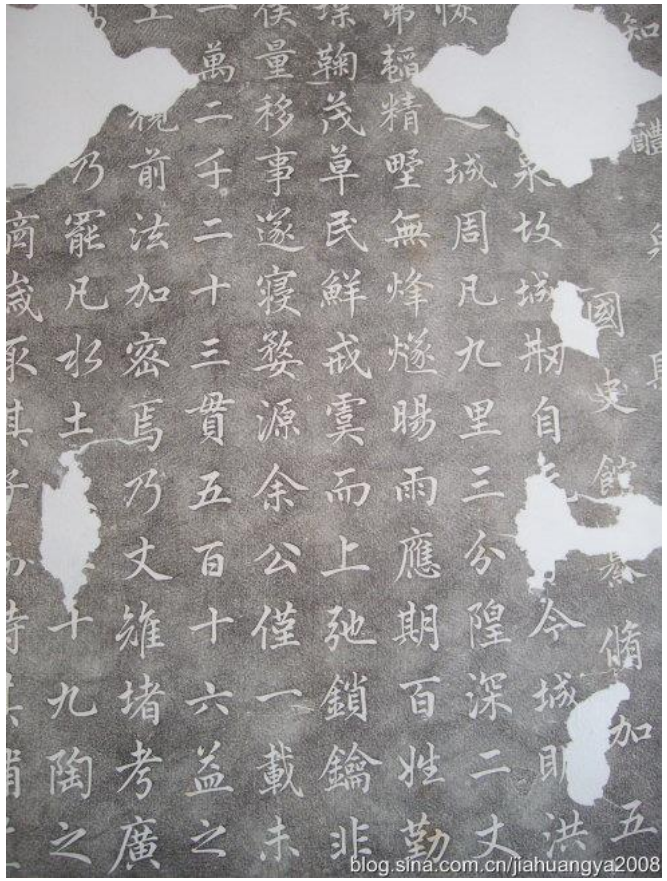
- **常见数据问题（2）：离群数据**

- 如前所述，异常点即与大部分其他对象不同的数据
  - 异常点既可能是我们研究的目标，也可能对我们的研究造成干扰



- **常见数据问题（3）：缺失数据**

- 一方面，数据采集的不完整可能导致数据缺失
  - 例如，室内无法利用GPS采集位置坐标
- 另一方面，部分属性值在部分数据中不适用
  - 例如，儿童往往无法采集其收入信息
- 对待缺失值，往往采取删除和填补并重的方法
  - 一方面，基于已有数据填补缺失值
  - 另一方面，抛弃无法补全的数据记录



- **常见数据问题（4）：重复数据**

- 数据集中可能存在重复或近似重复的数据，这往往是由于多源数据归并所导致的
  - 例如，实体中的歧义现象，利君沙（琥乙红霉素片）
  - 又如，一个社交网络中用户的多个马甲账号
- 往往基于数据整合的方法加以解决
  - 例如，识别两个实体是否是含义相同，并合并相应的数据

- **常见预处理方法**

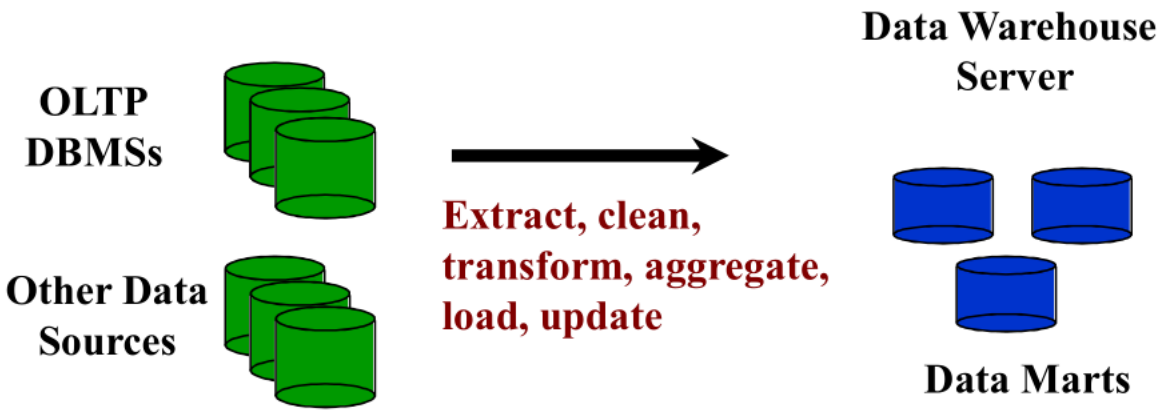
- 针对上述问题，需要通过数据预处理的方式提升数据质量
  - 数据整合（Aggregation）
  - 数据采样（Sampling）
  - 维度归约（Dimensionality Reduction）
  - 数据离散（Discretization & Binarization）

本节目录

- 数据挖掘常用方法
- **数据预处理**
  - **数据聚合**
  - 数据采样
  - 数据归约
  - 数据离散

- **数据聚合的概念和目的**

- 将两个或多个对象合并成为单个对象
  - 往往目的在于归并多个数据源的数据到统一格式下
  - 可以在一定程度上解决前述重复数据的问题
- 数据聚合的动机
  - 减少空间和时间开支
  - 对象群的属性比个体更稳定



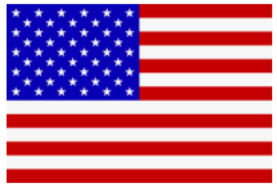
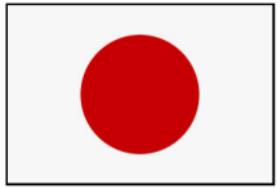
- **数据聚合的问题**

- 多源数据归并时可能存在各种问题
  - 可能存在不同的属性名称

cid	name	byear
1	Jones	1960
2	Smith	1974
3	Smith	1950

Customer-ID	state
1	NY
2	CA
3	NY

  - 可能存在不同的单位或尺度


- **数据聚合的问题**

- 多源数据归并时可能存在各种问题
  - 可能存在不同的属性统计方式

cid	monthlySalary
1	5000
2	2400
3	3000

cid	Salary
6	50,000
7	100,000
8	40,000

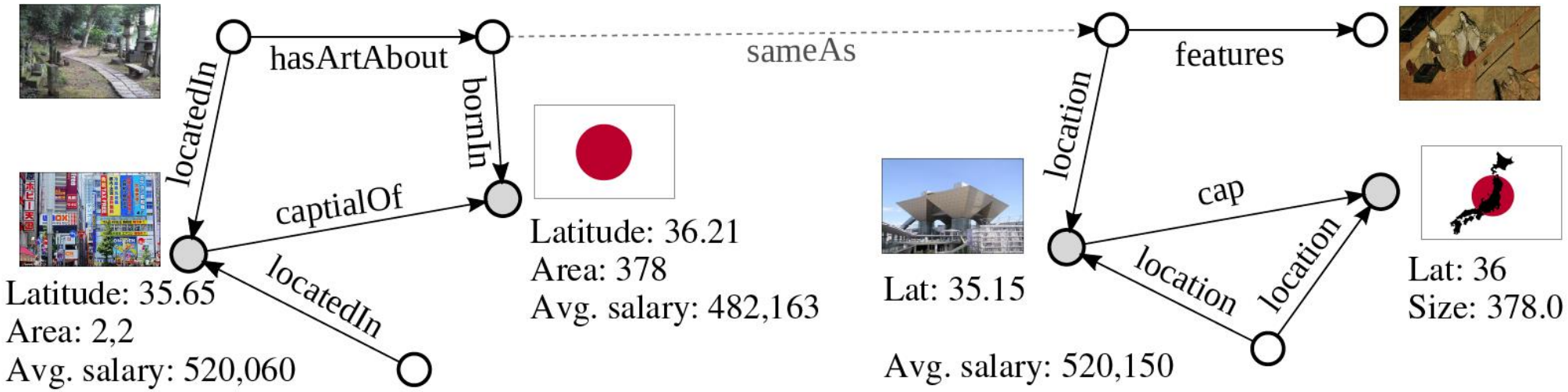
  - 不同数据源可能存在统计上的不一致性

cid	monthlySalary
1	5000
2	6000

cid	Salary
1	60,000
2	80,000

- **数据聚合的实现**

- 根据待归并的数据形式不同采取不同的方案
  - 可能采用简单的换算和汇总即可
  - 也可能需要复杂的算法，如实体对齐，且存在一定的错误率

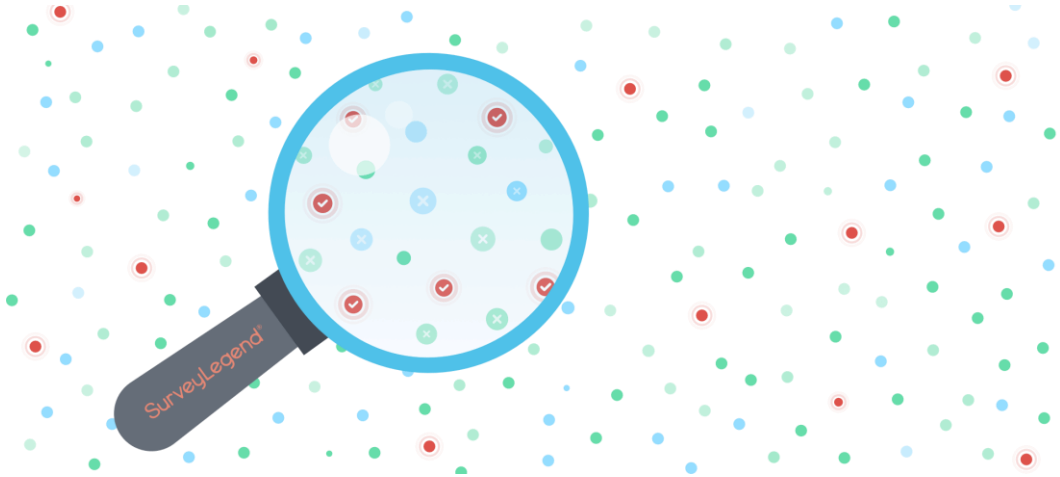


本节目录

- 数据挖掘常用方法
- **数据预处理**
  - 数据聚合
  - **数据采样**
  - 数据归约
  - 数据离散

- **数据采样的概念和目的**

- 数据采样，指选择一部分数据对象的子集进行分析的常用方法
  - 数据规模的急剧增加带来了计算能力的巨大负担
  - 通过采取小规模样本，可以起到近似的效果，同时降低开支
  - 即使在要求精确的场合，通过采取小规模样本进行初步分析，了解数据特性，也是有效的手段



- **数据采样的代表性问题**

- 采样缺乏代表性，将影响对于原数据集的还原程度，进而产生误导
  - 采样数据应至少在统计指标上近似原数据集，例如均值和方差

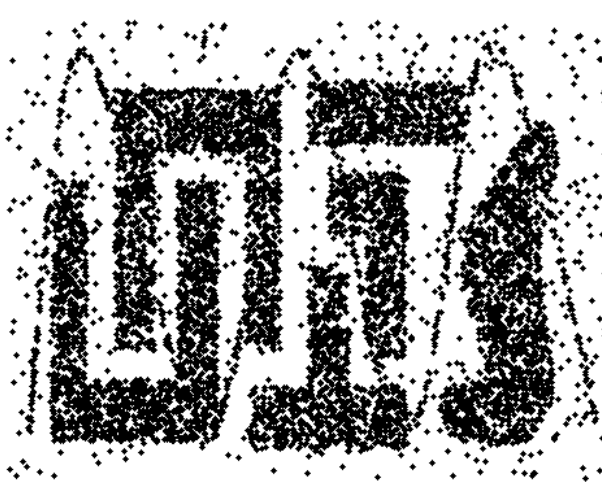
How the Media can manipulate our viewpoint

- **常用的数据采样方法**

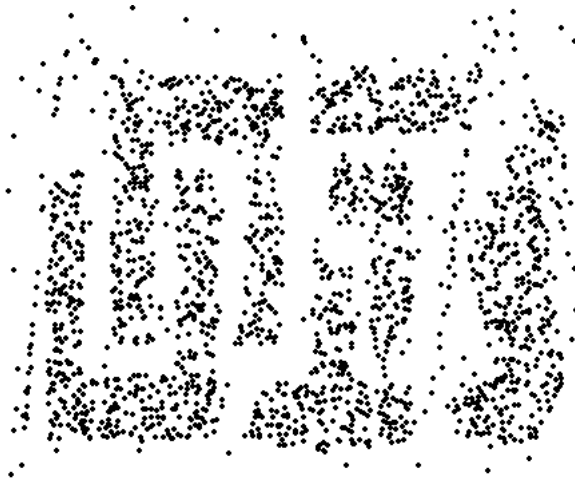
- 最基本的采样技术为简单随机采用（Simple Random Sampling）
  - 对于所有对象，采用简单的等概率方式进行采样
  - 一般采用两种方式进行
    - 无放回采样：被采中的对象从整体中删除，仅可选中一次
    - 有放回采样：被选中的对象不会从整体中删除，可再次被选中
  - 两种方式无本质区别，但有放回采样较为简单（概率不变）
  - 更为复杂的方法包括分层采样（Stratified Sampling）
    - 对数据进行分组，从预先指定的组里进行采样

- **采样规模与信息损失**

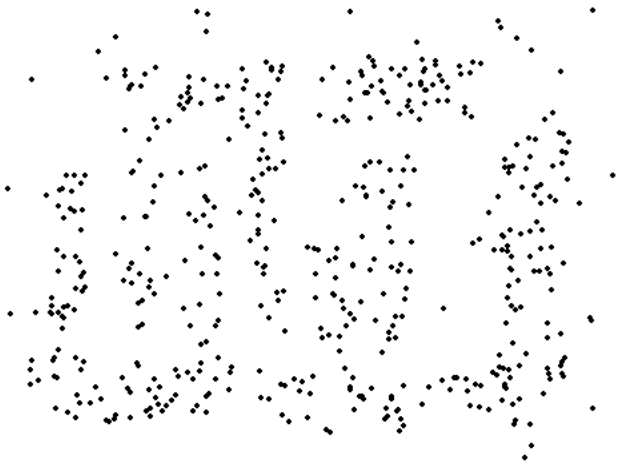
- 影响采样效果的重要因素之一是采样的样本容量
  - 较大的样本容量更能完整代表数据，但降低了采样的收益
  - 较小的样本容量在采样收益上更高，但可能造成信息的损失



8000 points



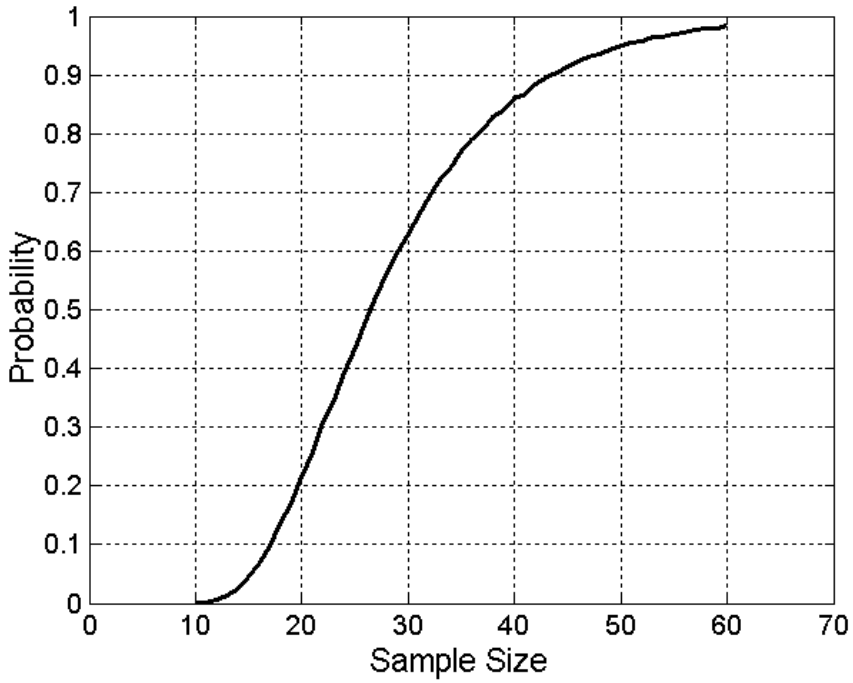
2000 Points



500 Points

- **启发式的采样规模确定方法**

- 通过分组采样的方式，可以近似确定适当的样本容量
  - 例如，将一个数据集分为10组，每组数量大致相等
  - 组内的数据高度相似，而不同组的对象差异性较大
  - 右图展示了一定的样本容量下，能够从每个组中至少取到一个数据点的概率
    - 可见，至少40个点，才能保证10组都取到的概率接近90%



- **一些题外话**

- 数据采样不能成为数据造假的帮凶！
  - 一种不良的倾向：将数据采样等同于数据“筛选”
  - 采样不应该具有任何的倾向性
    - 所有算法的数据标准应该一视同仁
    - 不能以结果作为采样依据

本节目录

- 数据挖掘常用方法
- **数据预处理**
  - 数据聚合
  - 数据采样
  - **数据归约**
  - 数据离散

- **维度归约的必要性**

- 在数据集中，用于描述对象可能涉及大量的特征
  - 然而，并不是所有的特征都具有显著的区分作用

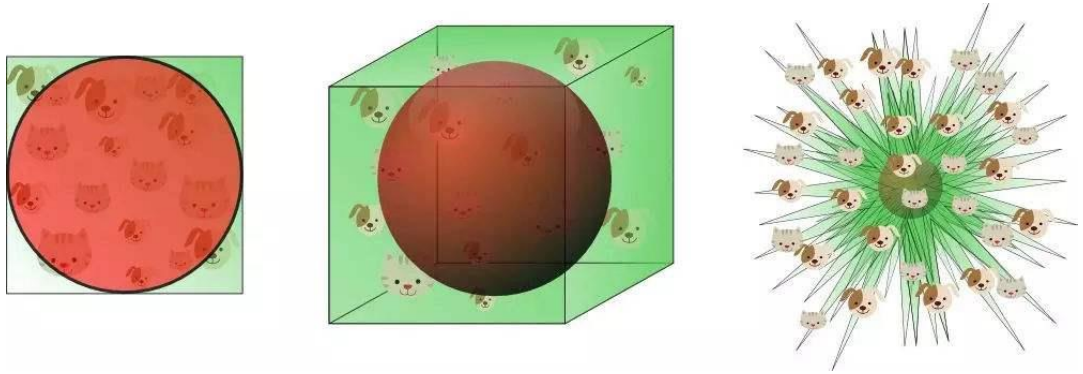
表 1

学生编号	语文	数学	物理	化学
1	90	140	99	100
2	90	97	88	92
3	90	110	79	83
. . .	. . .	. . .	. . .	. . .

  - 通过维度归约，可以删除不具有区分度的特征，同时可能降低噪声
  - 在避免维度灾难的同时，模型更容易理解，也更易于可视化

- **维度灾难的概念**

- 维度灾难，又称维度诅咒（Curse of Dimensionality）
  - 指随着数据维度的增加，数据分析困难程度大幅上升的现象
  - 可能由于以下两点原因导致
    - 计算量呈指数级增长，难以处理
    - 数据稀疏，没有足够数据可建模



- **如何进行维度归约?**

- 在先前的例子中，缺乏区分度的维度可以直观地发现，因此人工删除即可
- 然而，更多时候，需要归约的维度难以通过简单的人工判断加以区分

表 2

学生编号	数学	物理	化学	语文	历史	英语
1	65	61	72	84	81	79
2	77	77	76	64	70	55
3	67	63	49	65	67	57
4	80	69	75	74	74	63
5	74	70	80	84	82	74
6	78	84	75	62	72	64
7	66	71	67	52	65	57
8	77	71	57	72	86	71
9	83	100	79	41	67	50
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .

- **维度归约的代表性方法：主成分分析**

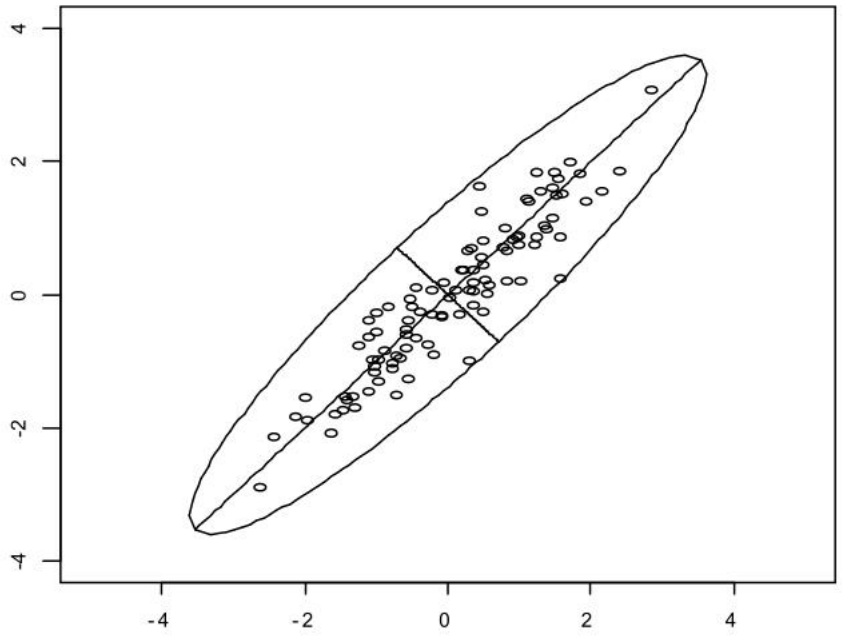
- 对于成绩而言，优秀的学生尚可用总分来加以衡量
- 然而，更多时候，属性更复杂、维度更多，且无法简单加和
  - 假定你是一个公司的财务经理，掌握了公司的所有数据，如固定资产、流动资金、借贷的数额和期限、工资支出、原料消耗、产值等信息。
  - 当你需要向上级汇报公司状况的时候，你可以原封不动地罗列所有指标和数字吗？当然不能。
  - 你必须对各个方面进行高度概括，进而用少数指标简单明了地介绍。

- **维度归约的代表性方法：主成分分析**

- 针对这一需求，我们希望能够从纷乱的属性中找到一些具有代表性、综合性的指标，从而包含丰富的信息量，帮助我们抓住主要矛盾
- 主成分分析（Principal Component Analysis，PCA）应运而生
- PCA的思路是通过正交变换将一组可能存在相关性的变量转换为一组线性不相关的变量，转换后的这组变量叫主成分。
  - 通过这种方式，可以采用较少的综合指标综合先前存在于各个属性（且相关）中的各类信息，而综合指标之间彼此不相关。

- **维度归约的代表性方法：主成分分析**

- 获取主成分的过程，同时也是降维的过程。如何进行?
- 先看一个二维的实例，如果只有两维特征，我们可以将其表示为一个椭圆
  - 椭圆有长轴短轴，相比之下，短轴方向的数据变化较少，区分度偏低
  - 在这种情况下，我们删去短轴，再将坐标轴变换与长轴平行
- 对于高维椭球而言，思路是类似的，即找出主轴与几个最长的轴作为新维度



- **维度归约的代表性方法：主成分分析**

- 现在的问题在于，如何选定轴（坐标系）？
- 回顾先前的过程，我们意识到，选择轴的标准是轴上的投影方差尽可能大
- 而一个有趣的现象是：最大特征值对应的特征向量可以最大化投影方差
  - 证明将作为思考题在下次作业中出现，资料很好找啊随便一搜就能找到
- 因此，我们只要求得数据样本的最大的K个特征值，其特征向量所对应的线性组合就可以形成K个新的综合指标
  - K个特征值的比重反应了主成分的信息量，一般应大于0.85

参考资料：https://blog.csdn.net/zhongkelee/article/details/44064401

- **维度归约的代表性方法：主成分分析**

- 主成分分析的一个实例，以先前的考试成绩为例
  - 可见，头两个成分特征值累积占了总方差的81.142%，而后面的特征值的贡献越来越少

Total Variance Explained

Component	Initial Eigenvalues			Extraction Sums of Squared Loadings		
	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	3.735	62.254	62.254	3.735	62.254	62.254
2	1.133	18.887	81.142	1.133	18.887	81.142
3	.457	7.619	88.761			
4	.323	5.376	94.137			
5	.199	3.320	97.457			
6	.153	2.543	100.000			

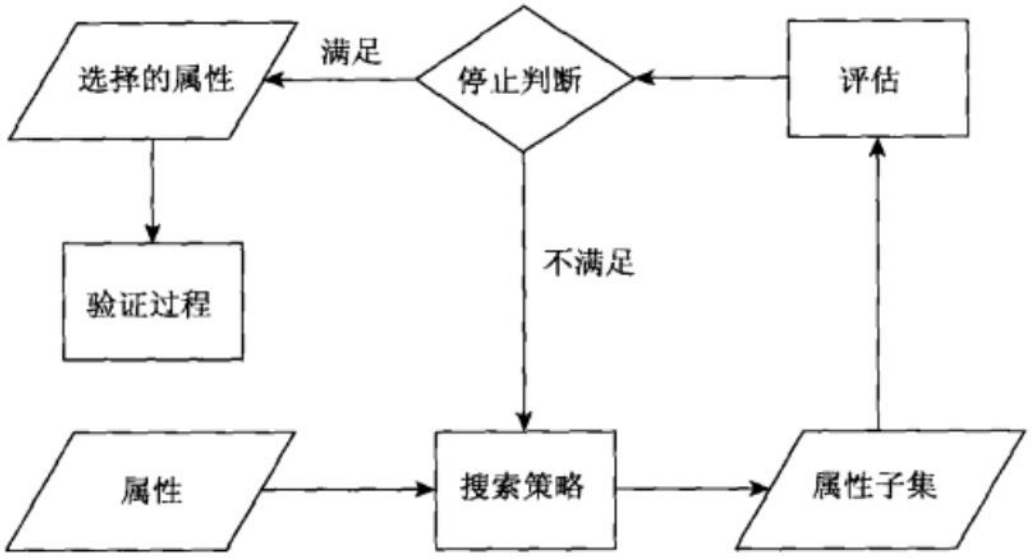
Extraction Method: Principal Component Analysis.

- **维度归约的代表性方法：主成分分析**

- 主成分分析有哪些需要注意之处?
  - 主成分分析依赖于原始变量，也只能反映原始变量的信息。因此，原始变量的选择很重要（对于原来的问题本身也很重要）
  - 主成分分析的内在假设之一是原始变量直接存在一定的关联性
    - 相应的，如果原始变量本质上相互独立，那么降维就有可能失败
    - 很难将独立变量用少数综合变量概况，数据越相关，降维效果越好
  - PCA的结果未必清晰可解释，与选取的原始变量及数据质量等都有关

- **维度归约的另一种思路：特征子集选择**

- 降低维度的另一个思路是仅使用特征的一个子集（而不是归纳新特征）
  - 其目的在于去除冗余特征（重复信息，例如商品价格与商品税）和不相关特征（例如学生成绩与学生学号往往无关）
  - 除了直接删去多余特征外，为特征赋予不同权值也是一个可选择的方案

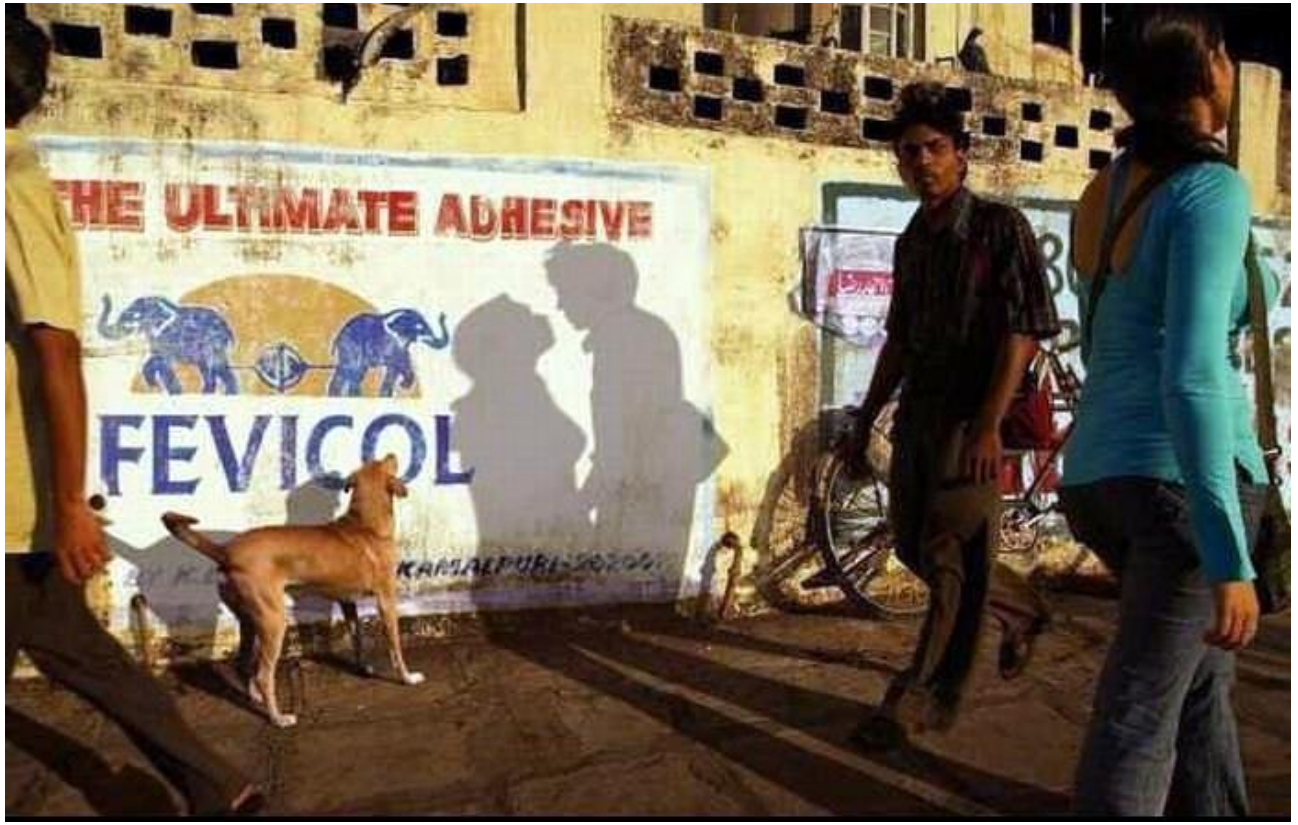


- **维度归约中的信息损失**

- 需要注意的是，维度归约同样可能造成信息损失，甚至产生误导效果

- **维度归约中的信息损失**

- 需要注意的是，维度归约同样可能造成信息损失，甚至产生误导效果

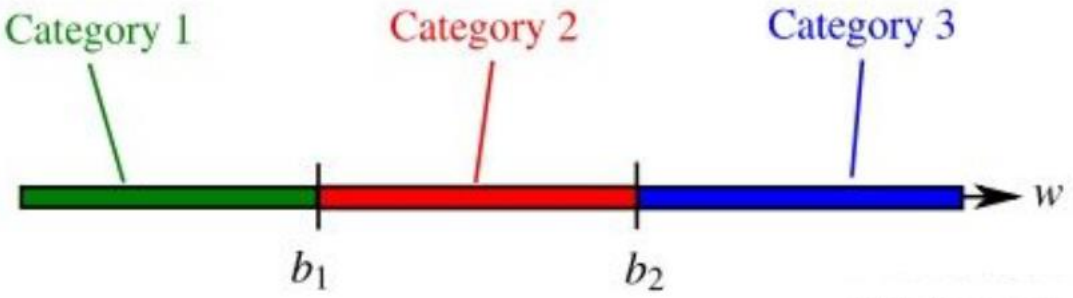


本节目录

- 数据挖掘常用方法
- **数据预处理**
  - 数据聚合
  - 数据采样
  - 数据归约
  - **数据离散**

- **Pointwise类排序算法**

- 第三类方法：有序回归方法（Ordinal Regression）
  - 某种意义上，相当于利用回归方法求解有序多分类问题。



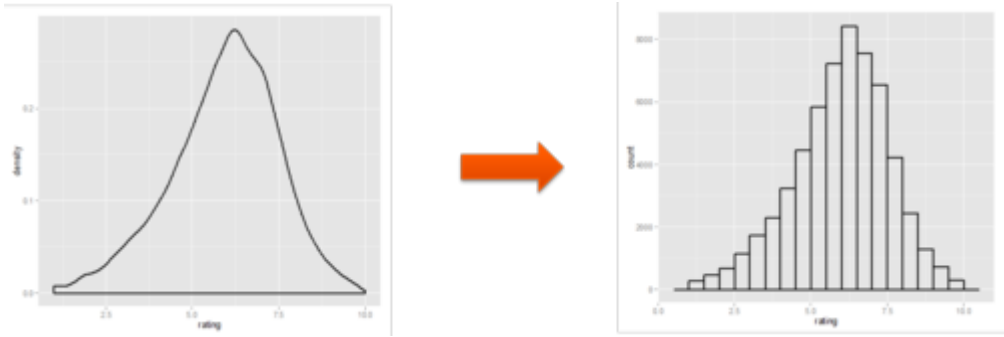
  - 其中，文档的标签可通过如下公式推测：

$$\hat{y}_j = \arg\min_k \{w^T x_j - b_k < 0\}.$$

  - 投影后，位于数轴上的区间决定了文档的标签。

- **数据离散化的概念**

- 数据离散化（Discretization），旨在将连续属性变换为分类属性
  - 例如，病人的年龄是一个连续值，但是在实际治疗中，往往仅需要一些年龄段的信息即可（如成年/未成年，儿童/成人/老人等）
  - 也可用于分类属性值过多的情况，例如大学的专业较多，可以用文理工农加以概括
- 对于特定数据挖掘问题，特别分类问题，通过合并减少类别数目是有益的



- **最基本的离散化：二元化**

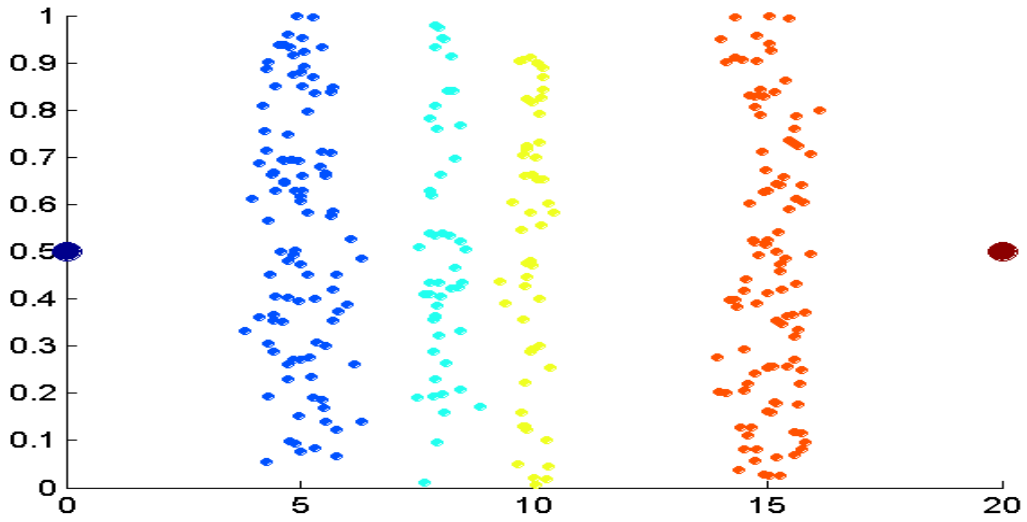
- 二元化（Binarization）是离散化的基本形式
  - 其目的在于将连续或离散属性转化为一个或多个二元属性
    - 不同类型问题采用不同二元化（如关联问题倾向于第二种方式）

分类值	整数值	x_1	x_2	x_3
awful	0	0	0	0
poor	1	0	0	1
OK	2	0	1	0
good	3	0	1	1
great	4	1	0	0

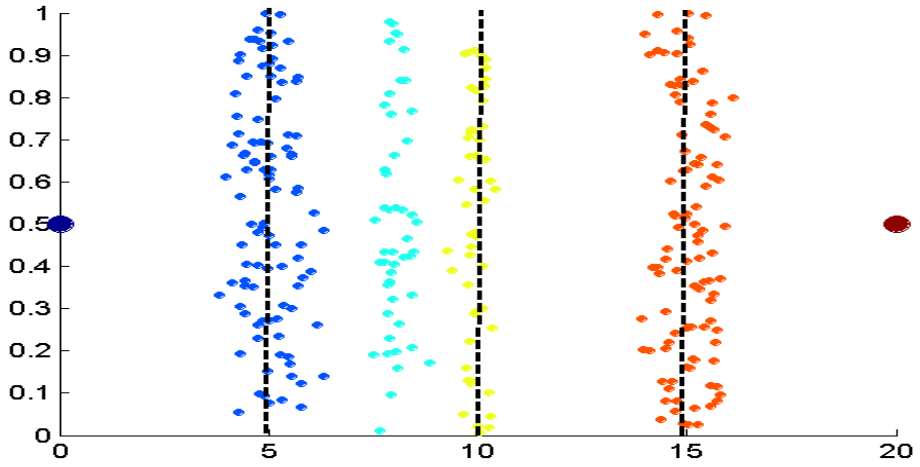
分类值	整数值	x_1	x_2	x_3	x_4	x_5
awful	0	1	0	0	0	0
poor	1	0	1	0	0	0
OK	2	0	0	1	0	0
good	3	0	0	0	1	0
great	4	0	0	0	0	1

- **非监督离散化**

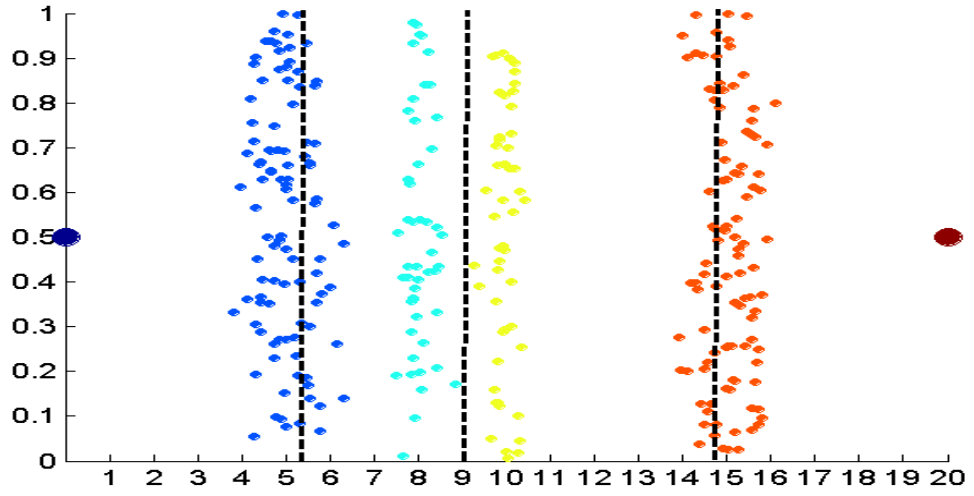
- 用于分类的离散化方法，最根本的区别在于其离散化过程是否有监督
  - 即是否使用类别信息（Supervision）
  - 对于不使用类别信息的非监督离散化方法，往往根据数据本身的特性进行离散，常见的方法包括等宽、等频率、等深等



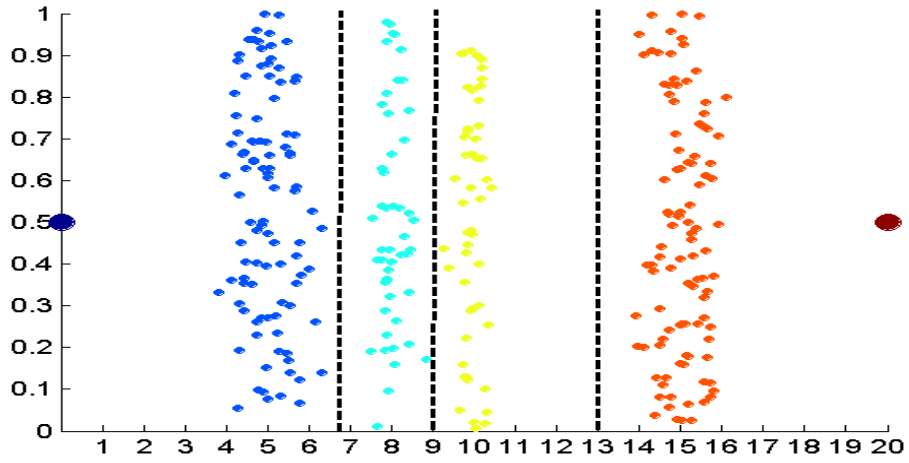
• 非监督离散化



等宽离散化



等频率离散化



K均值离散化

此处仅依赖 x 轴进行离散化
Y 轴主要用于取不同值方便可视化

- **有监督离散化**

- 有监督离散化更注重问题导向，其目的在于取得更好的结果
- 基于熵（Entropy）的方法是最重要的有监督离散化方法之一
  - 采用如下方法定义某个区间的熵

$$e_i = -\sum_{j=1}^{k} p_{ij} log_2 p_{ij}$$

  - 总的熵可以定义为加权和

$$e = \sum_{i=1}^{n} w_i e_i$$

  - 显然，熵越小，区间内的纯度越高（标签越一致），越符合我们的要求
  - 因此，一种做法是先进行二分，选择熵最小的点进行分割。进而，对其中具有较大熵（即纯度不高，信息较混乱）的部分再进行下一轮分割，以此类推。

本章小结

数据准备

- 数据挖掘基本方法
  - 关联规则
  - 异常检测
- 常用的数据预处理方法
  - 数据聚合、数据采样
  - 维度归约
  - 数据离散化/二元化

tongxu@ustc.edu.cn