为什么要进行时数据的预处理?

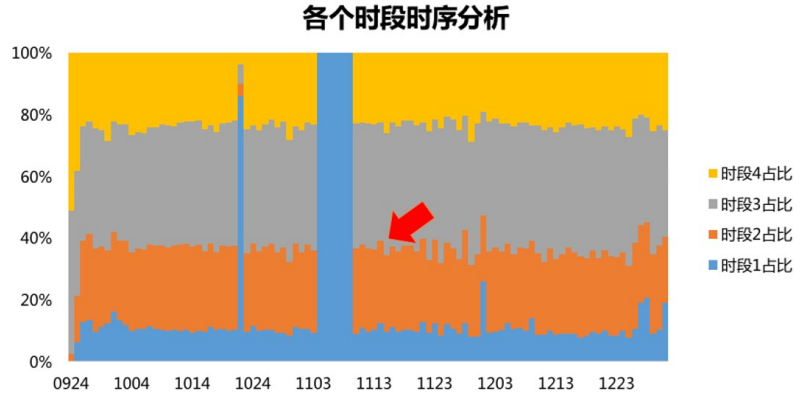
- 原始数据可能存在许多的问题，如：
  - 现实世界中的数据大多是**“脏”数据**：
    - 不完整：缺少属性值或某些感兴趣的属性，或仅包含聚集数据
      - e.g.：occupation=“ ”
    - 含噪声：包含错误或存在偏离期望的离群值
      - e.g.：Salary=“-10”
    - 不一致：例如，用于商品分类的部门编码存在差异
      - e.g.：Age="42" Birthday=“03/07/1997” （根据生日算出来的年龄不是42，一致性不对）
    - 注：数据清洗的过程就是将脏数据变为干净数据的过程。
  - **维数灾难**
    - 定义：用来描述当（数学）空间维度增加时，分析和组织高维空间（通常有成百上千维），因体积指数增加而遇到各种问题场景。
      - 取样指数级增长
      - 稀疏数据特征不明显
      - 距离在高维环境下失去意义
    - 注：可能的两种解决方法
      - 降维：
      - subpure
  - **数据的质量要求达不到**
    - 一致性
    - 准确性
    - 完整性
    - 时效性
    - 可信性

    - 可解释性
- 可见，在海量的原始数据中，存在大量杂乱、重复、不完整的数据，严重影响到数据挖掘的算法执行效率，有可能导致挖掘结果的偏差。
    - 在一个完整的数据挖掘过程中，数据**预处理要花费60%左右的时间。**（数据采集→数据预处理→数据挖掘→知识评价→呈现）
- 综上，我们需要在数据挖掘前进行数据预处理工作。

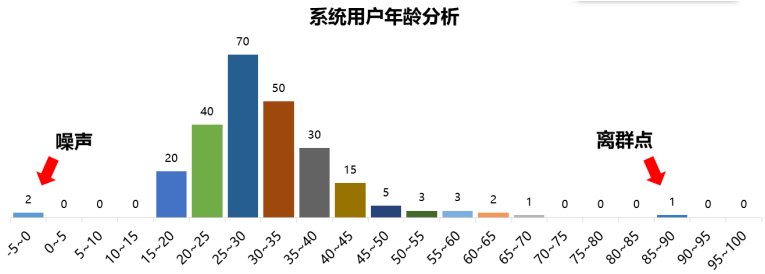
数据预处理的工作包括哪些？这些工作分别是干什么的，怎么做的？

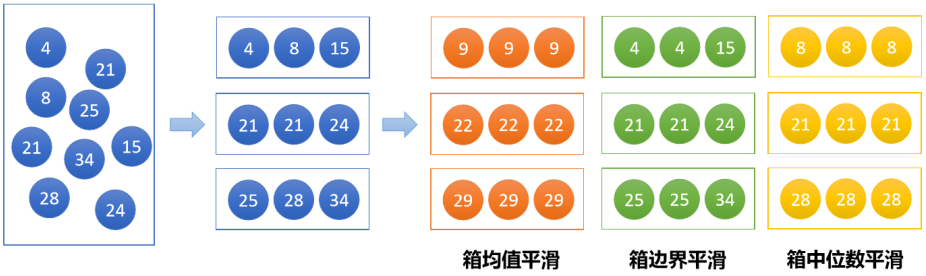
有五大主要工作：数据清洗、数据集成、数据归约、数据变换、数据离散化。

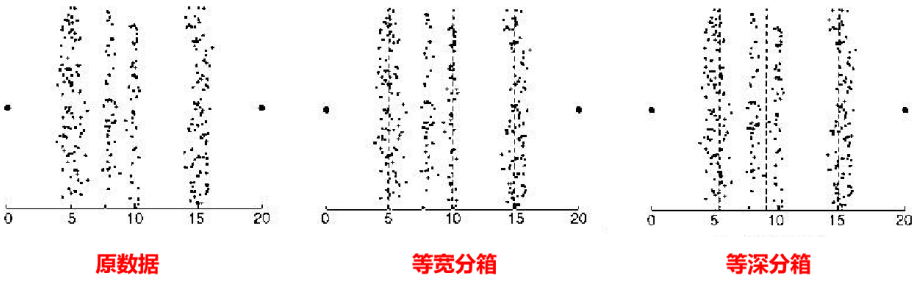
一、数据清洗（将脏数据变为干净数据）

- **数据缺失**
    - 

    - **缺失值处理方法**
        - **忽略元组**：（若数量少、占比低、样本空间足够）元组有多个属性缺少值时，或者该元组剩余属性值使用价值较小时
        - **人工填写**：很费时，当数据集很大、缺失值很多时，人工填写行不通
        - **全局常量填充**：方法简单，但挖掘程序可能误以为他们形成了一个有趣的概念，故并不十分可靠
        - **属性中心度量填充**：（均值/中位数/众数）对于正常的（对称的）数据分布而言可以使用**均值**，而倾斜数据分布应使用**中位数**
        - **使用最可能的值填充**：使用**回归**、基于推理的工具或者**决策树归纳**确定

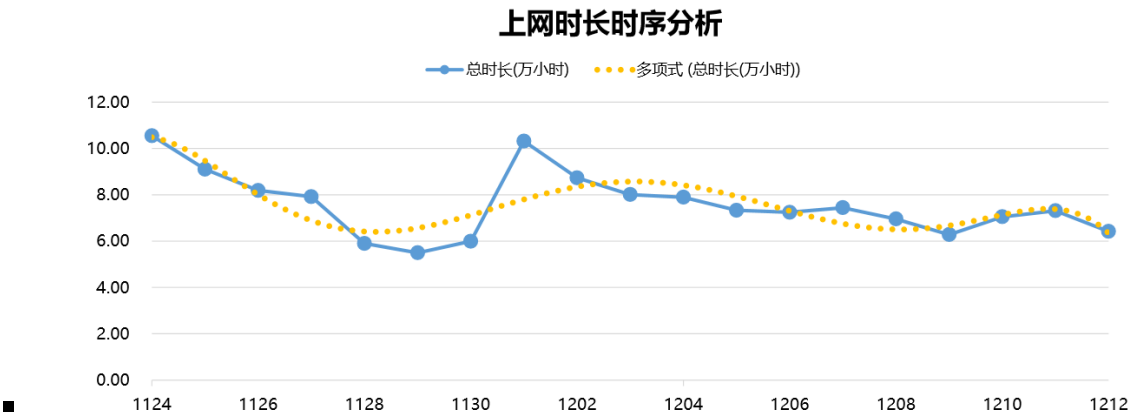
    - ……
- **噪声数据与离群点**
    - 噪声：被测量的变量的随机误差或方差。
    - 离群点：数据集中包含一些数据对象，它们与数据的一般行为或模型不一致。
    - 

- **分箱（binning）**
    - **定义：**通过考察数据的“近邻”（即周围的值）来光滑有序数据值。这些有序的值被分布到一些“桶”或箱中。由于分箱方法考察近邻的值，因此是**局部光滑**。
        - 箱均值平滑、箱边界平滑、箱中位数平滑
            - 边界平滑：判断离边界哪边近：8-4=4，15-8=7，4<7，令8=4；28-25=3，34-28=6，6>3，令28=25
        - 

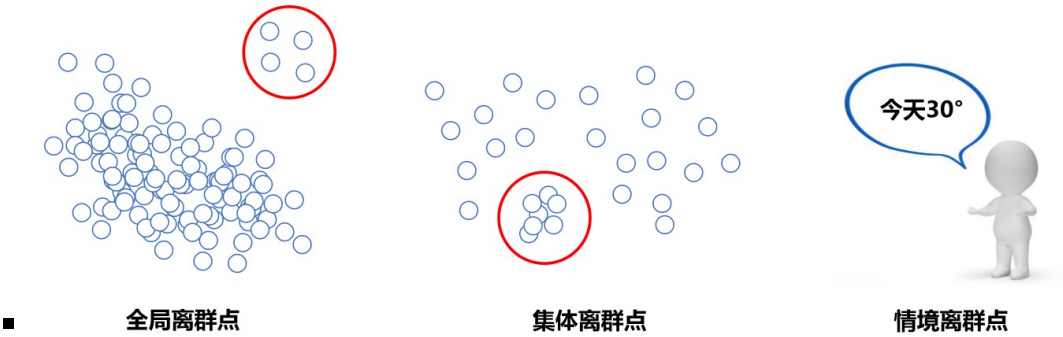
    - **分箱方法：**
        - **等宽分箱：**每个“桶”的**区间宽度相同**
        - **等深分箱：**每个“桶”内的**样本个数相同**
        - 

- 回归（Regression）
    - 用一个函数拟合数据来光滑数据
        - 线性回归找出拟合两个属性（或变量）的最佳直线；多元线性回归涉及多个属

性，将数据拟合到多维曲面。

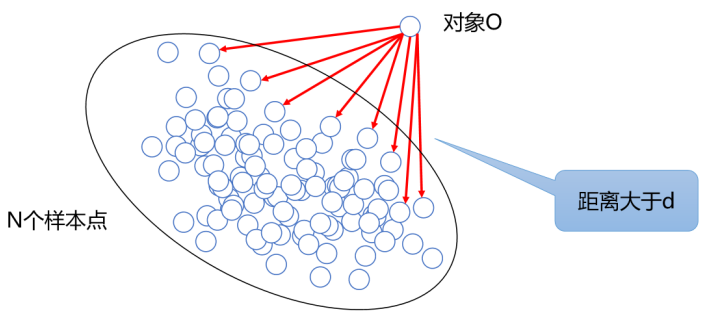


- **离群点**
  - **分类**：全局离群点、集体离群点、情境离群点

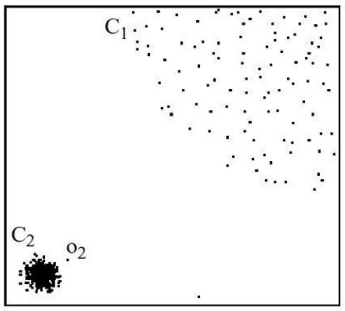


全局离群点　　集体离群点　　情境离群点

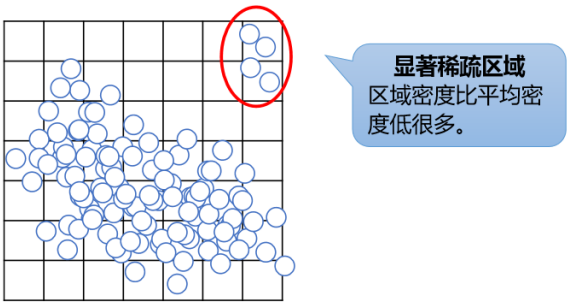
  - **传统离群点的几个预测方法：**
    - **基于统计的离群点预测**：假设给定的数据集服从某一随机分布（如正态分布等），用**不一致性测试（discordancy test）**识别异常
      - 如果某个样本点不符合**工作假设**，那么认为它是离群点；
      - 如果它符合**备选假设**，则认为它是符合某一备选建设分布的离群点。
    - **基于距离的离群点预测**：如果样本空间D中至少有**N个样本点**与对象O的**距离大于d**，那么称对象O是以至少N个样本点和距离d为参数的基于距离的离群点。



    - **基于偏差的离群点预测**：通过检查一组对象的主要特征来识别离群点，那些不符合这种特征的数 据对象被判定为离群点。

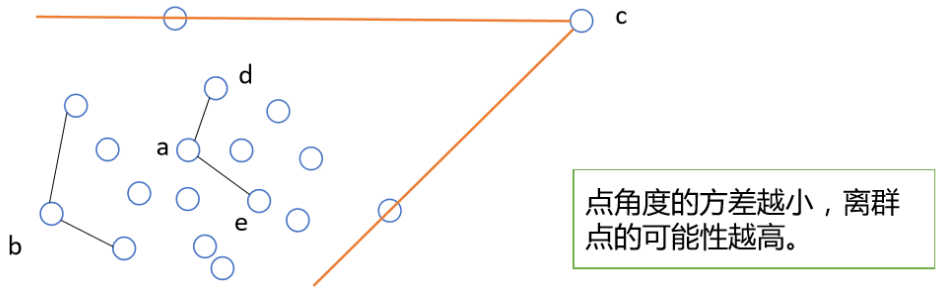
        - 顺序异常技术与OLAP数据立方体技术
    - **基于密度的离群点预测：**
        - 通过基于密度的局部离群点检测，就能在样本空间数据分布不均匀的情况下也可以准确发现离群点
        - 

- **上述传统离群点检测算法的缺点：**
    - 基于统计的算法：不适合多维度空间；预先需要知道样本空间中数据集的分布特征。
    - 基于距离的算法：参数的选取非常敏感；受时间复杂度限制，不适用于高维稀疏数据集。
    - 基于偏差的算法：实际应用少；在数据量大、数据维度高的数据集中，很难获得该数据集的主要特征。
    - 基于密度的算法：时间复杂度高，不适合运用在高维空间上检测离群点。
- **高维数据中的离群点检测**
    - 高维数据的离群点检测方法面临众多挑战。
    - **离群点的解释：**高维数据涉及多个维度，因此离群点较难解释，可能是揭示离群点的特定子空间，或者关于对象的“离群点性”的评估
    - **数据的稀疏性：**离群点检测方法应该能够处理高维空间的稀疏性。随着维度增加，对象之间的据类严重被噪声所左右
    - **数据子空间：**离群点检测方法应该以合适的方式对离群点建模，例如，自适应显示离群点的子空间和捕获数据的局部变化
    - **维度可伸缩性：**随着维度的增加，子空间数量指数增加，包含所有可能的子空间的穷举组合探索不是可伸缩的选择
- **发现子空间中的离群点：**
    - 如果发现一个对象是很低维度的子空间中的离群点，则该子空间提供了重要信息，解释该对象为什么和在何种程度上是离群点。（**可解释性**）



- **基于网格的子空间离群点检测方法**

- **高维离群点建模**
  - 直接为高维离群点建立一个新模型，方法通常避免邻近性度量，而采用**新的启发式方法**来检测离群点，方法不会在高维数据数据中退化



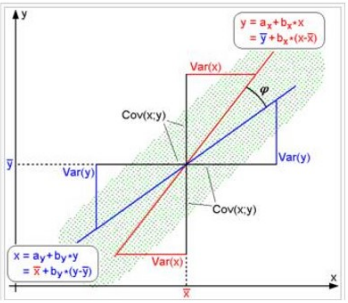
  - **基于角度的离群点检测**

二、 数据集成

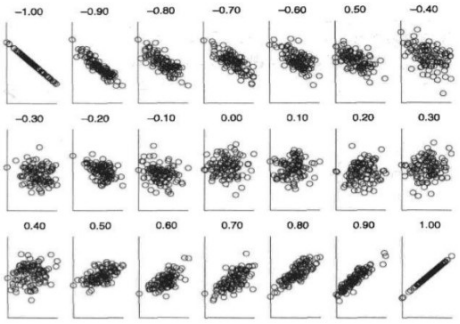
- **数据集成做了什么？**把不同来源、格式、特点性质的数据在逻辑上或物理上有机地集中，从而为企业提供全面的数据贡献。
- 数据属性：

标称属性	属性值是一些符号或事物的名称。
	经常看做分类属性，每个值代表某种类别、编码或状态。
	头发颜色：黑色、棕色、黄色、红色.....
二元属性	是一种标称属性，只有两个类别：0或1，0通常表示该属性不出现，1表示出现。
	布尔属性：true和false
	对称的、非对称的二元属性。（两类属性权重是否相同）
序数属性	其可能的值之间具有有意义的序或秩评定，但是相继值之间的差是未知的。
	客户满意度：0—很不满意，1—不太满意，2—中性，3—满意，4—很满意。
数值属性	定量的，可度量的量，用整数或实数值表示。
	区间标度属性：2017年与2000年相差17年。
	比率标度属性：文档以字数计，雇员的工作年限等。

    - **离散属性：**具有有限或无限可数个值；可以是数值属性。如：头发颜色、性别、员工号……
    - **连续属性：**如果属性不是离散的，则它是连续的。连续属性一般用浮点变量表示。例如：结算金额。
- **实体识别问题**
    - 数据集成时，模式集成和对象匹配非常重要，如何将来自多个信息源的等价实体进行匹配，此即**实体识别问题**。
        - 如：如何将百度百科的数据挖掘定义与Wiki的数据挖掘定义进行合并?
- **数据冗余和相关性分析**
    - 数据冗余
        - 同一数据在系统中**多次重复出现**，**文件系统**中文件之间没有联系，有时数据在多个文件中出现；而**数据库系统**则克服了文件系统的这种缺陷，但仍然存在数据冗余问题。
        - 因此需要消除数据冗余，**以便保持数据的一致性**
    - 相关性分析
        - 网站分析中经常使用的分析方法之一
        - 通过对**不同特征或数据间的关系**进行分析，可以发现业务运营中的**关键影响及驱动因素**，并对业务的发展进行预测
- **皮尔逊相关系数**
    - 用于度量两个变量X和Y之间的相关（**线性相关**），其**值介于-1与1之间。**



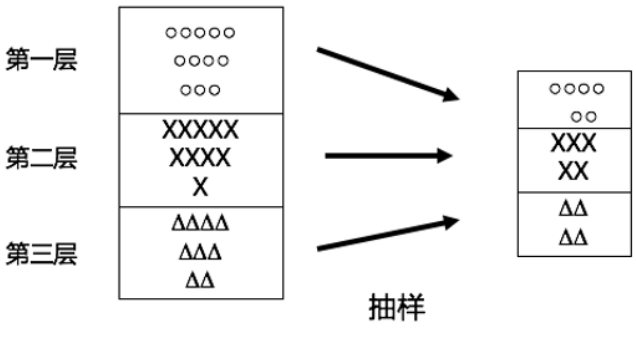
- 两组数据向量夹角的余弦



相关度从-1到1的散布图

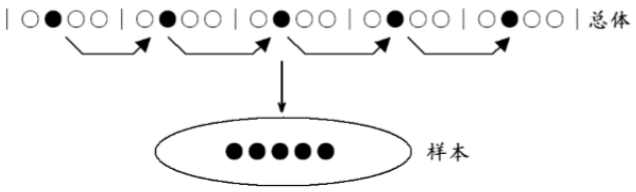
三、 数据归约

- **数据归约策略**
  - **维归约**：减少所考虑的随机变量或属性的个数；把原数据变换或投影到较小的空间；包括小波变换、主成分分析等方法
  - **数量归约**：用替代的、较小的数据表示形式替换原数据；包括有抽样和数据立方体聚集
  - **数据压缩**：使用变换，以便得到原数据的规约或“压缩”表示；
    - **无损压缩**：原数据能够从压缩后的数据重构，而不损失信息
    - **有损压缩**：只能近似重构原数据
- **抽样**
  - **分层抽样**
    - 第一层：5+4+3=12—4+2=6；二：5+4+1=10—3+2=5；三：4+3+2=9—2+2=4



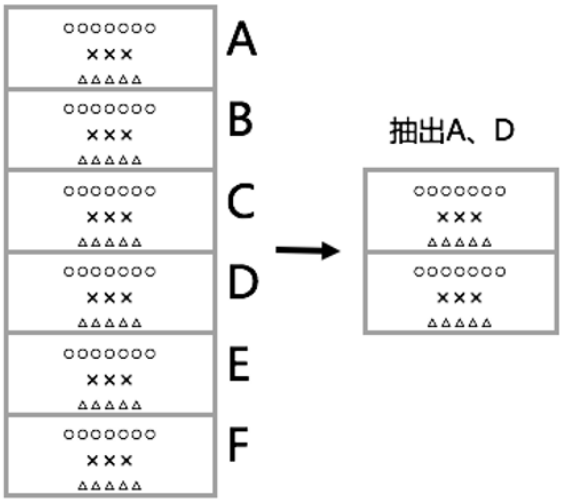
分层抽样（stratified sampling）

  - **系统抽样**



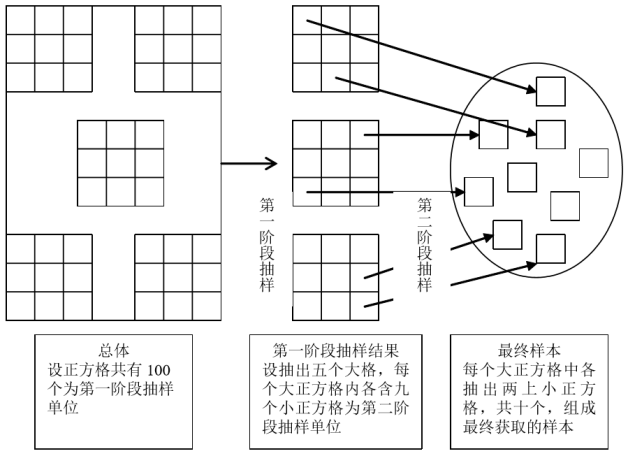
系统抽样（systematic sampling）

- 整群抽样

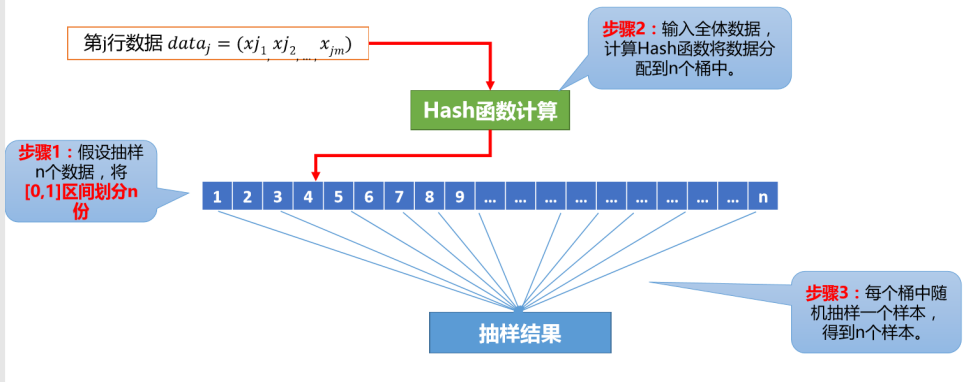


整群抽样（cluster sampling）

- 多阶段抽样

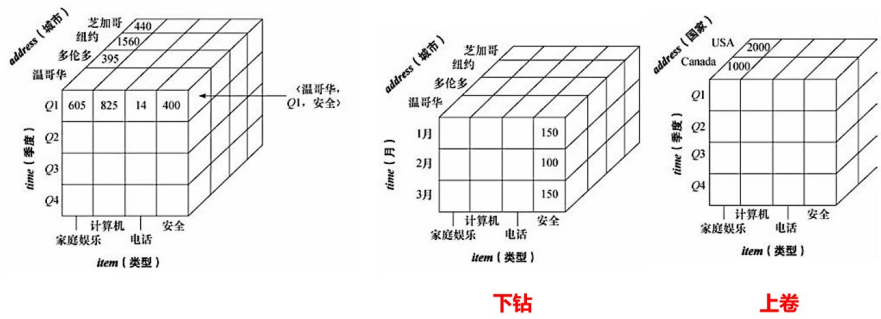


- 基于Hash函数的取样技术SHF



- **数据立方体聚集**

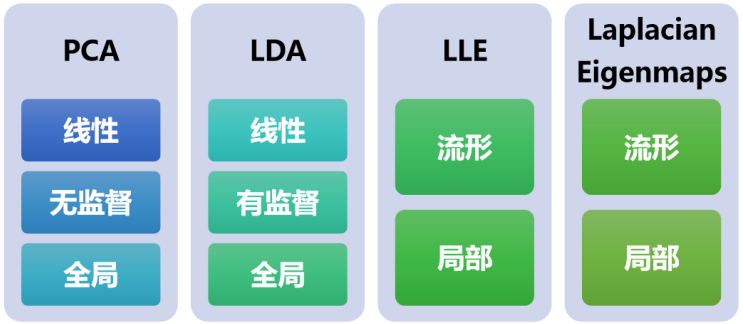
  ■多维数据模型是为了满足用户从多角度多层次进行数据查询和分析的需要而建立起来的基于事实和维的数据库模型，其基本的应用是为了实现**OLAP（Online Analytical Processing）**。



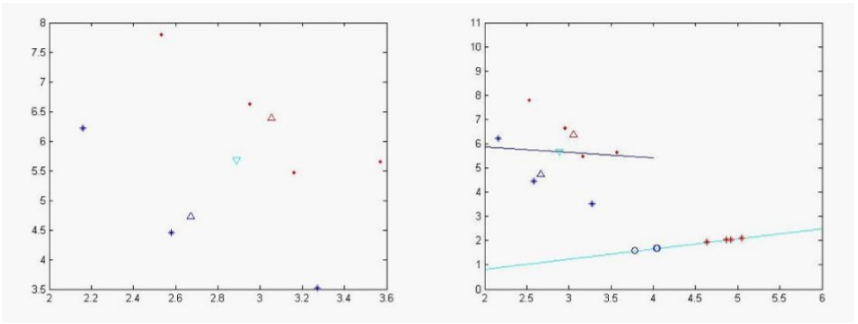
下钻　　上卷

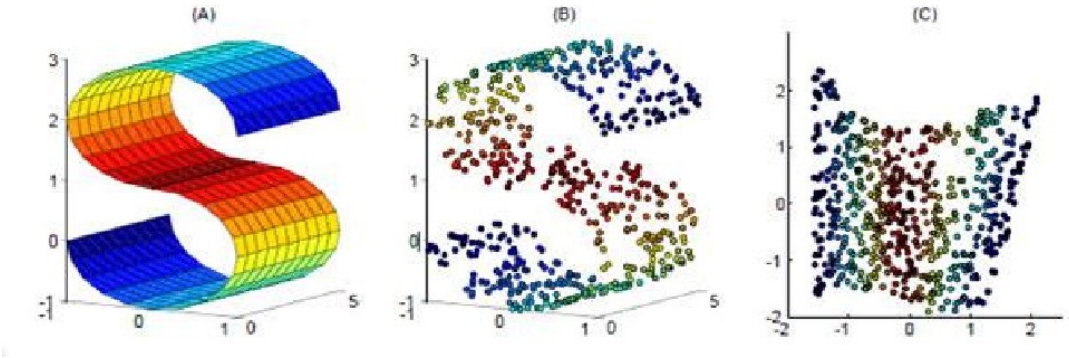
  - 按季度汇总—>按月汇总：下钻——列的扩展
  - 按城市汇总—>按国家汇总：上卷——行的归约
  - 钻：改变维的层次，变换分析的粒度
  - 上卷：是沿着维的层次向上聚集汇总数据。下钻：上卷的逆操作，沿着维的层次向下，查看更详细的数据
  - 行归约：抽样、据类
  - 列归约：降维等
  - 归约时，行列分开

- **机器学习中的降维算法：**四种：PCA/LDA/LLE/Laplacian Eigenmaps



  - **主成分分析PCA**
    - 主成分分析（Principal Component Analysis，PCA）是最常用的线性降维方法。
    - 核心思想：**在降维之后能够最大化保持数据的内在信息**，通过衡量在投影方向上的**数据方差的大小**来衡量该方向的重要性。
    - 

- **线性判别分析LDA**
  - 线性判别分析（Linear Discriminant Analysis，LDA），有监督的线性降维方法。
  - 核心思想：**数据在降维后能够很容易地被区分开来。**将高维的模式样本投影到最佳鉴别矢量空间，保证模式样本在新子空间有**最大的类间距离和最小的类内距离**，即模式在该空间中有最佳的**可分离性。**
  - 

- **局部线性嵌入LLE**
  - 局部线性嵌入（Locally linear embedding，LLE）是一种非线性降维算法，它能够使降维后的数 据较好地保持原有流形结构。
  - 

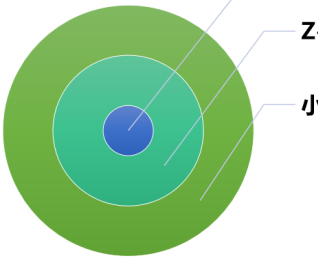
  - 如果数据分布在**整个封闭的球面**上，LLE则不能将它映射到二维空间，且不能保持原有的数据流形。在 处理数据中，首先假设数据不是分布在闭合的球面或者椭球面上。

四、 数据变换

- 数据变换策略
  - 光滑：去掉数据中的噪声。包括分箱、回归和聚类。

- 属性构造：由给定的属性构造新的属性，并添加到属性集中，以帮助挖掘过程。
- 聚集：对数据进行汇总或聚集。通常，这 一步用来为多个抽象层的数据分析构造数据立方体。
- 规范化：把属性数据按比例缩放，使之落入一个特定的小区间。
- 离散化：数值属性的原始值用区间标签或概念标签替换。这些标签可以递归地组成更高层概念，导致数值属性的概念分层。
- 由标称数据产生概念分层：有些属性可以泛化到较高的概念层。 许多标称属性的概念分层都蕴含在数据库的模式中，可以在模式定义级自动定义。

- 数据规范化



最小最大规范化 $to\ [new\_min_A, new\_max_A]$

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

Z-Score规范化

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

$(\mu: mean, \sigma: standard\ deviation)$

小数定标

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

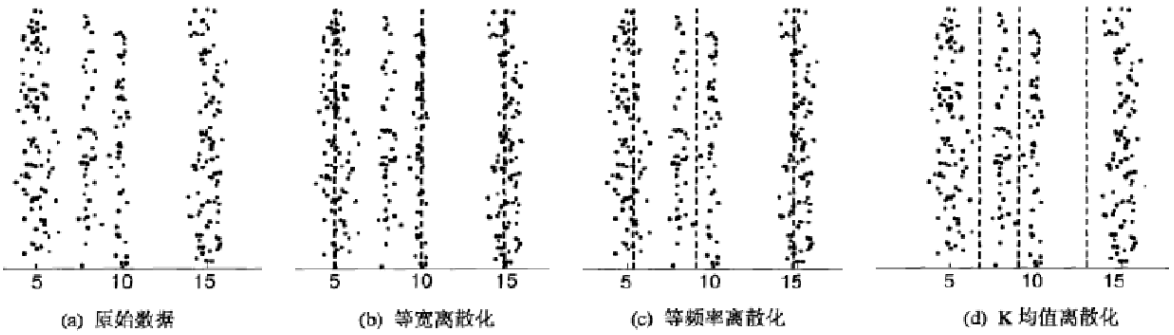
$where\ j\ is\ the\ smallest\ integer\ such\ that\ max(|v'|) < 1$

- 推荐最小最大规范化：将数据映射到区间中

  为了避免量纲的不同产生的影响

五、 数据离散化

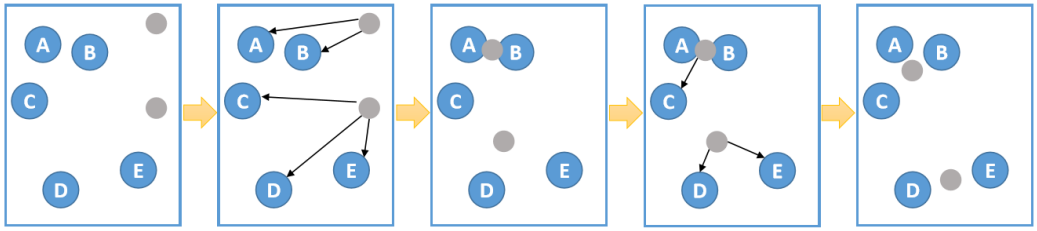
（大数据与数据的差别：多态数据的融合）



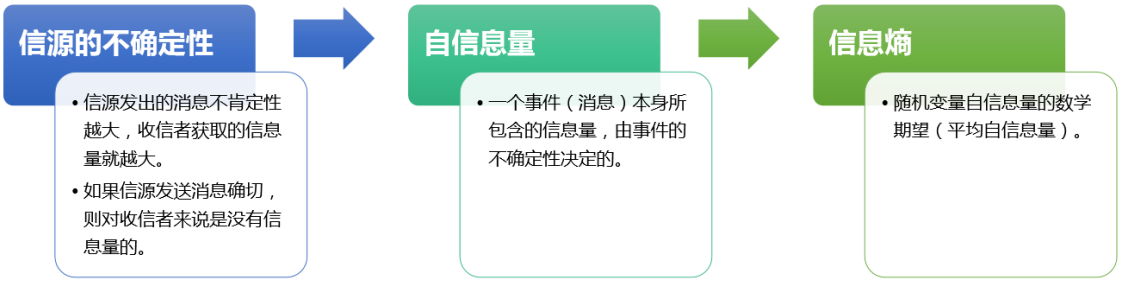
- **非监督离散化：**无监督离散化方法在离散过程中不考虑类别属性，其输入数据集仅含有待离散化属性的值

- **K均值离散化**

  预先指定目标离散化区间数：K



- **监督离散化**：监督离散化方法的输入数据包含类别信息（类标号），效果比无监督离散化要好。
  - **基于熵的离散化**
    - 使用类分布信息用于计算和确定划分点（划分属性区间的数据值）。
    - 主要思想：选择划分点使得一个给定的结果分区包含尽可能多的同类元组。
    - 基于熵的离散化：为了离散化数值属性A，该方法选择最小化熵的A的值作为划分点，并递归地划分结果 区间，得到分层离散化。这种离散化形成A的概念分层。
    -



额外内容

大数据预处理

大数据时代：海量数据、多格式数据、产生速度快

集群的威力：Hadoop

2008年，Hadoop打破了297s的世界纪录，成为最快的TB级数据排序系统，仅用时209s。

“Hadoop是一个提供分布式存储和计算的软件框架，它具有无共享、高可用、弹性可扩展的特点，非常适合处理海量数据。”

——Hadoop电梯演讲

海量数据的摇篮——HDFS

i. HDFS（Hadoop Distributed File System），是Hadoop的基石。

ii. HDFS处于Hadoop生态圈的最下层，存储所有数据，支持Hadoop所有服务。

iii. HDFS设计理念： 以流式数据访问模式，存储超大文件，运行于廉价硬件集群上，而且是一个具有高度容错性的文件系统。

HDFS能提供高吞吐量的数据访问，非常适合大规模数据集上的应用。

处理海量数据的利器——MapReduce