# 数据科学导论
# Introduction to Data Science

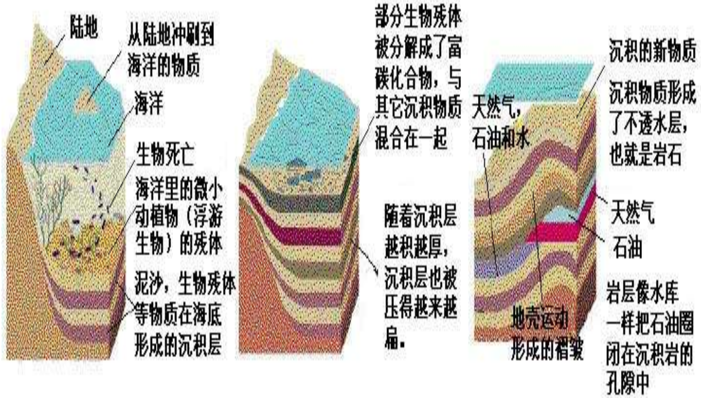
## 第四章 数据挖掘基础

刘 淇
Email: qiliuql@ustc.edu.cn
课程主页：
http://staff.ustc.edu.cn/~qiliuql/DS2017.html

# 数据挖掘基础



- Clustering——面临挑战：Class Imbalance Problem(类不平衡问题)


实际生活中存在很多非平衡问题

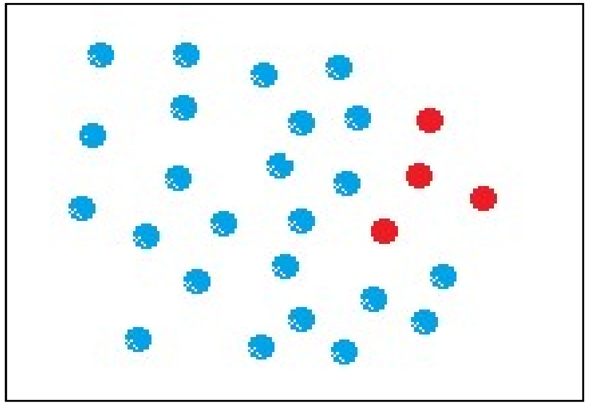
SVM  KNN

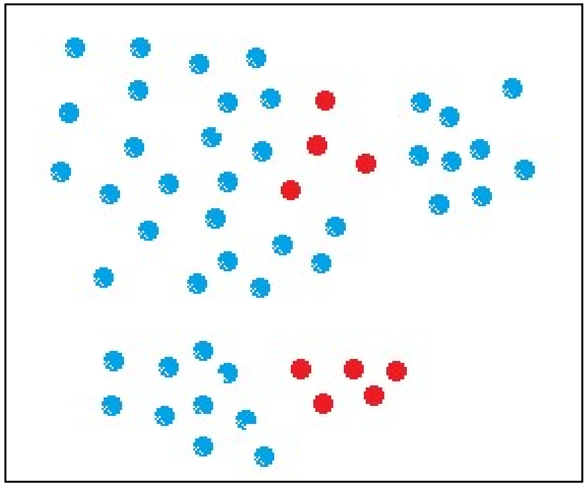
传统方法在非平衡数据集上性能不好

# 数据挖掘基础



- Clustering——面临挑战：Class Imbalance Problem(类不平衡问题)
- 非平衡数据集
  - 绝对稀少
    - 比例1:99， 1000个数据
  - 相对稀少
    - 比例1:99， 100,000个数据
  - 类间不平衡
  - 类内不平衡

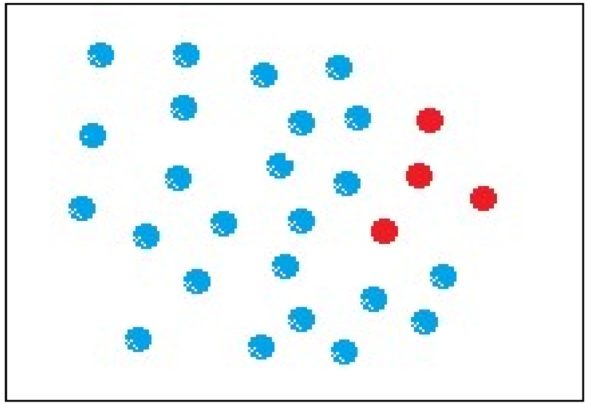


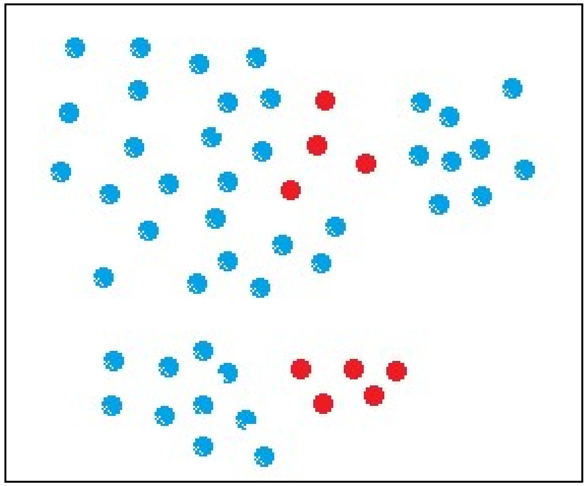


# 数据挖掘基础



- Clustering——面临挑战：Class Imbalance Problem(类不平衡问题)
  - 如何解决？
  - 预处理
    - 上采样
    - 下采样
    - 混合采样
  - 算法改进
    - 单类学习分类
    - 集成学习
    - **代价敏感学习**
    - 基于聚类
    - ……
  - 混合



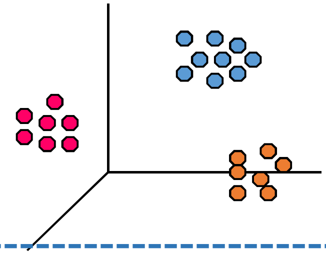


# 数据挖掘基础

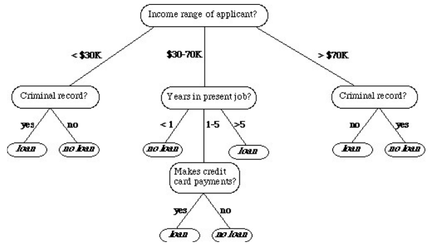


- 常用方法——关于四个任务有哪些常用方法？

**Clustering**



**Classification**




**Data**

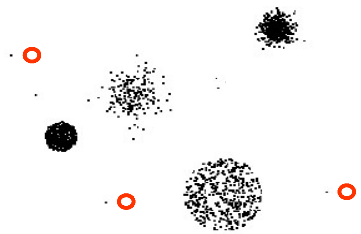
| | T | | H | | P | |
|---|---|---|---|---|---|---|
| | L | H | L | H | L | H |
| J | -6.0 | 8.8 | 60 | 100 | 986 | 1044 |
| F | -2.8 | 10.9 | 48 | 100 | 973 | 1025 |
| M | -5.6 | 17.7 | 34 | 100 | 976 | 1037 |
| A | -1.2 | 22.2 | 27 | 100 | 996 | 1036 |
| M | -0.8 | 27.8 | 25 | 100 | 1003 | 1034 |
| J | 5.2 | 29.1 | 26 | 100 | 998 | 1030 |
| J | 9.8 | 30.6 | 23 | 99 | 997 | 1027 |
| A | 5.6 | 26.1 | 31 | 100 | 992 | 1029 |
| S | 5.2 | 24.8 | 35 | 100 | 998 | 1028 |
| O | -0.4 | 21.3 | 42 | 100 | 990 | 1031 |
| N | -7.6 | 17.3 | 55 | 100 | 963 | 1023 |
| D | -10.4 | 9.2 | 53 | 100 | 987 | 1039 |

*table 17a*

2010 monthly weather variation, Cambridge (UK)

**Association Analysis**


**Anomaly Detection**



# 数据挖掘基础



- 常用方法—— Association Rule Mining（关联规则挖掘）
  - 给出事务的集合,能够发现一些规则：当事务中某些子项出现时，预测其他子项也出现

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

$\{Diaper\} \rightarrow \{Beer\}$,
$\{Milk, Bread\} \rightarrow \{Eggs,Coke\}$,
$\{Beer, Bread\} \rightarrow \{Milk\}$,

**Implication means co-occurrence, not causality!**

# 数据挖掘基础

- 关联规则挖掘——频繁项集
  - **Itemset（项集）**
    - A collection of one or more items
    - Example: {Milk, Bread, Diaper}
    - k-itemset
    - An itemset that contains k items
  - **Support count (σ)（支持数）**
    - Frequency of occurrence of an itemset
    - E.g. ({Milk, Bread,Diaper}) = 2
  - **Support （支持度）**
    - Fraction of transactions that contain an itemset
    - E.g. s({Milk, Bread, Diaper}) = 2/5
  - **Frequent Itemset （频繁项集）**
    - An itemset whose support is greater than or equal to a minsup threshold

| *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 |

# 数据挖掘基础



- 关联规则挖掘——关联规则
  - **Association Rule**
    - An implication expression of the form
    - X →Y, where X and Y are itemsets
    - Example:
      {Milk, Diaper} → {Beer}
  - Rule Evaluation Metrics
    - **Support (s，支持度)**
      - Fraction of transactions that
      - contain both X and Y
    - **Confidence (c，置信度)**
      - Measures how often items in Y appear in transactions that contain X

支持度是全局角度，置信度只关注于该规则自身

| 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:

$$\{\text{Milk}, \text{Diaper}\} \Rightarrow \text{Beer}$$

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

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

- 关联规则挖掘——关联规则挖掘任务
  - 给定事务集合 T, 关联规则发现是指找出支持度大于等于 minsup 并且置信度大于等于 minconf 的所有规则
    - support ≥ minsup threshold
    - confidence ≥ minconf threshold
  - Brute-force approach:
    - 列出所有可能的关联规则
    - 计算每一个规则的支持度和置信度
    - 修剪不符合minsup and minconf 的规则

    ⇒ Computationally prohibitive!

| *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 Rules:

{Milk,Diaper} $\rightarrow$ {Beer} (s=0.4, c=0.67)
{Milk,Beer} $\rightarrow$ {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} $\rightarrow$ {Milk} (s=0.4, c=0.67)
{Beer} $\rightarrow$ {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} $\rightarrow$ {Milk,Beer} (s=0.4, c=0.5)
{Milk} $\rightarrow$ {Diaper,Beer} (s=0.4, c=0.5)

## Observations:

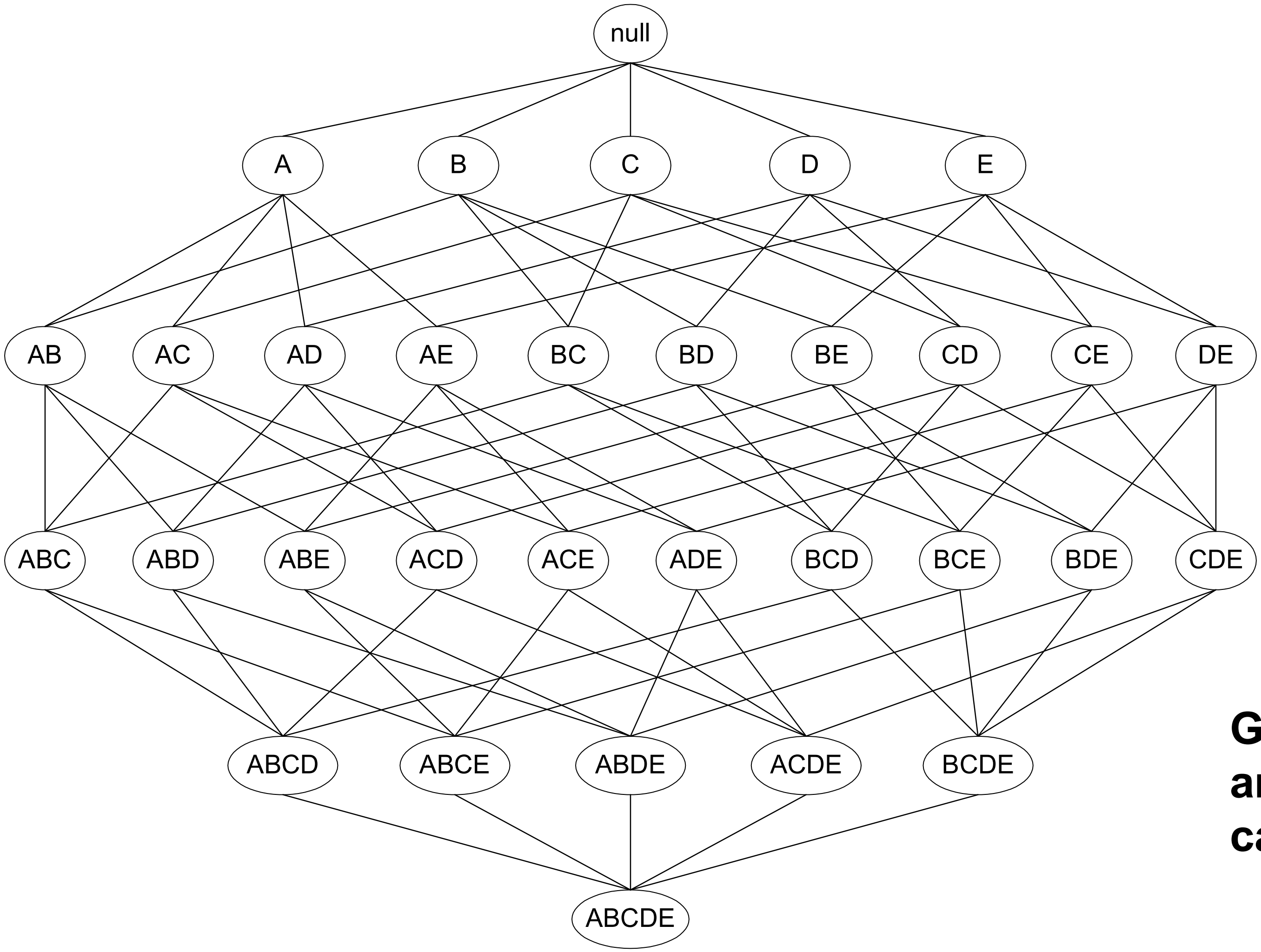
- All the above rules are binary partitions of the same itemset: {Milk, Diaper, Beer}
- Rules originating from the same itemset have identical support but can have different confidence
- **Thus, we may decouple the support and confidence requirements**

## 关联规则挖掘——频繁项集生成策略

- Two-step approach:
  - 1. 频繁项集生成
    - - Generate all itemsets whose support ≥ minsup
  - 2. 规则产生
    - - Generate high confidence rules from each frequent itemset, where each rule is a binary partitioning of a frequent itemset
- **Frequent itemset generation is still computationally expensive**

# 数据挖掘基础






**Given d items, there are $2^d$ possible candidate itemsets**

## 关联规则挖掘——频繁项集生成

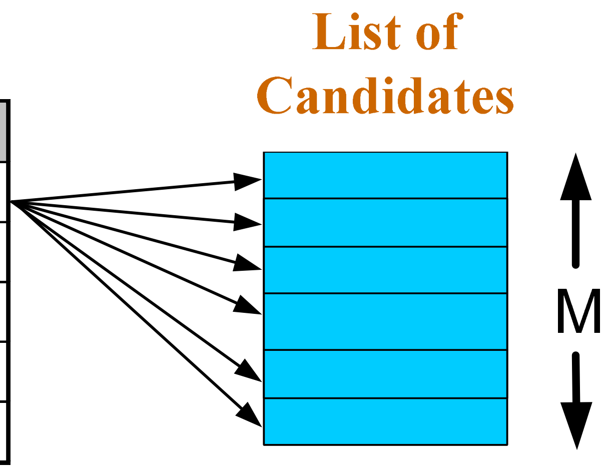
Brute-force approach:
Each itemset in the lattice is a candidate frequent itemset
Count the support of each candidate by scanning the database

**Transactions**

N

| *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 |

w



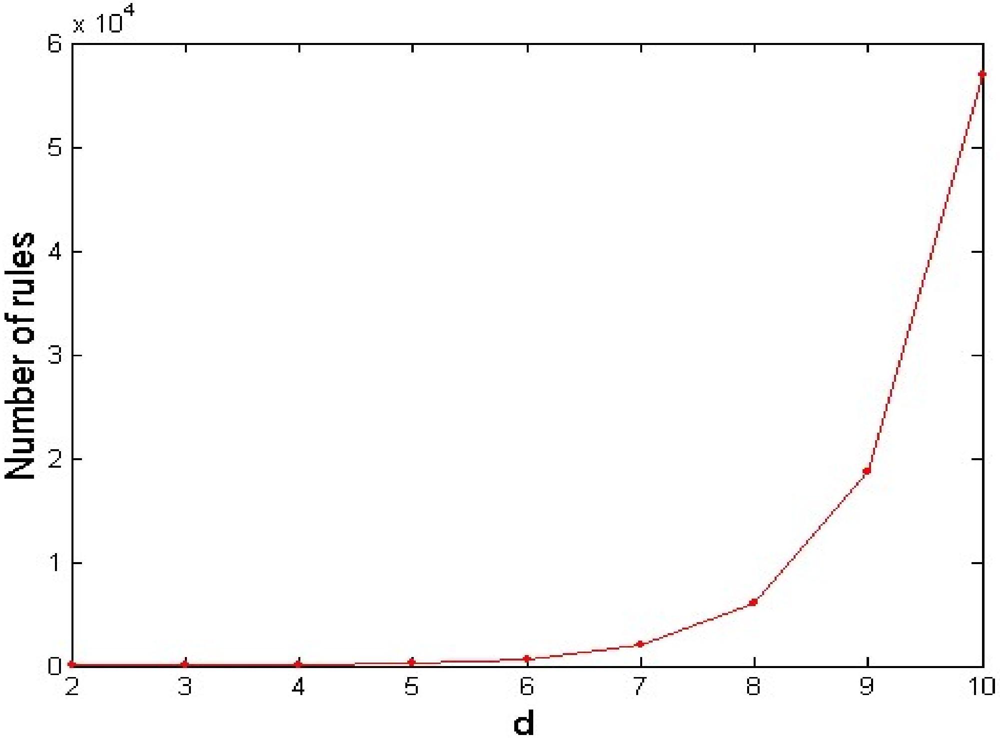

Match each transaction against every candidate
Complexity ~ O(NMw) => Expensive since $M = 2^d$ !!!

- 关联规则挖掘——计算复杂度

Given d unique items:

Total number of itemsets = $2^d$

Total number of possible association rules:




$$R = \sum_{k=1}^{d-1}\left[\binom{d}{k} \times \sum_{j=1}^{d-k}\binom{d-k}{j}\right] = 3^d - 2^{d+1} + 1$$

**If d=6, R = 602 rules**

# 关联规则挖掘——频繁项集生成策略

- 减少候选项集数目(M)
  - Complete search: $M=2^d$
  - Use pruning techniques to reduce M
- 减少事务数目 (N)
  - Reduce size of N as the size of itemset increases
  - Used by DHP and vertical-based mining algorithms
- 减少比较次数 (NM)
  - Use efficient data structures to store the candidates or transactions
  - No need to match every candidate against every transaction

# 数据挖掘基础



## 关联规则挖掘——减少候选项集数目

- Apriori 原理:
  - 如果一个项集是频繁的，则它的所有子集一定也是频繁的

- Apriori principle holds due to the following property of the support measure:

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

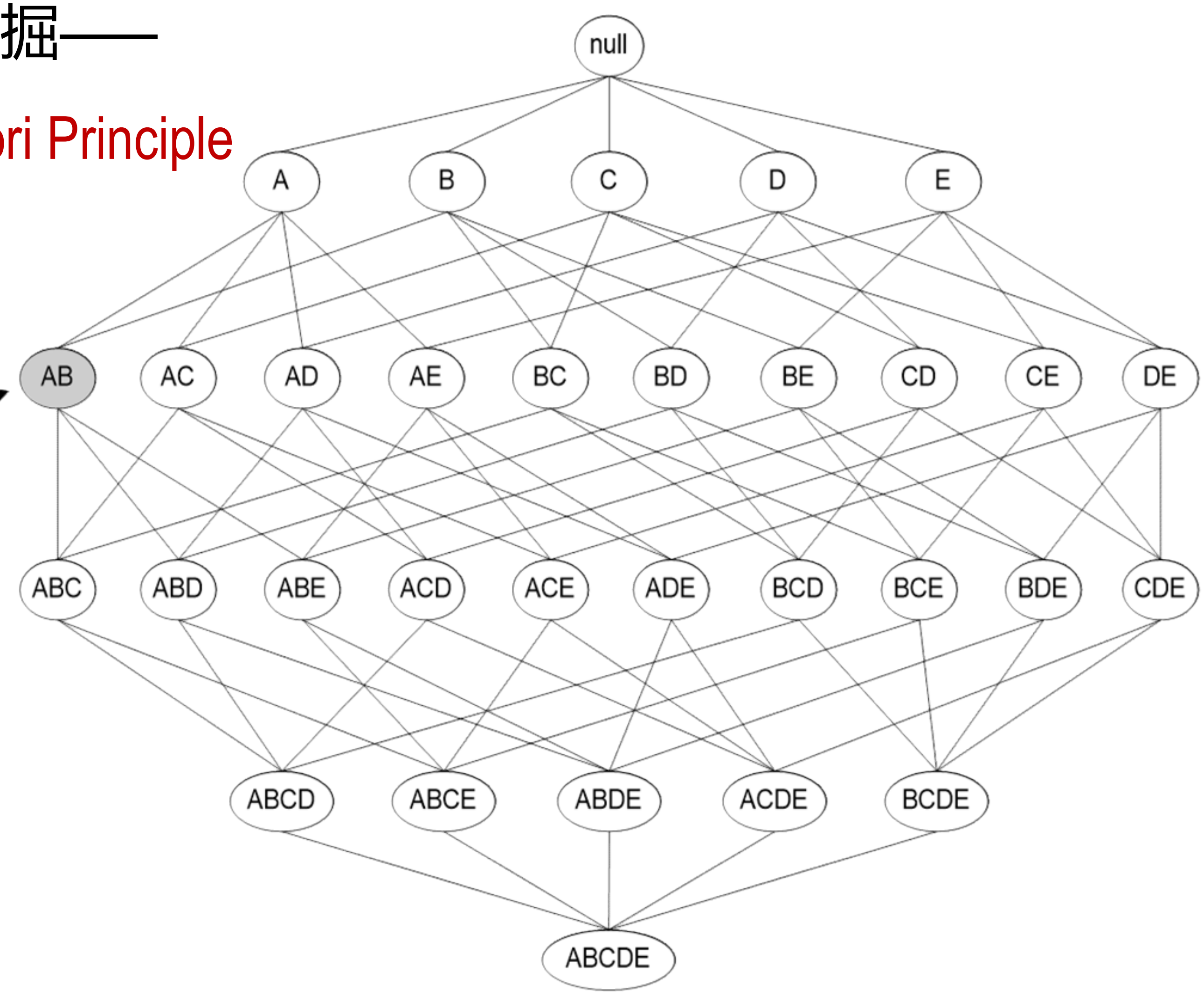
  - Support of an itemset never exceeds the support of its subsets
  - This is known as the anti-monotone（反单调性） property of support

# 数据挖掘基础

- 关联规则挖掘——

Illustrating Apriori Principle

Found to be Infrequent



# 数据挖掘基础



- 关联规则挖掘——使用 Apriori原理

| 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,
$^6C_1 + {}^6C_2 + {}^6C_3 = 41$
With support-based pruning,
$6 + 6 + 1 = 13$

- 关联规则挖掘—— Apriori Algorithm
  - Method:
    - Let k=1
    - 生成长度为 1 的频繁项集
    - 重复下述步骤直到没有新的频繁项集
      - 从长度为 k 的频繁项集生成长度为 (k+1) 候选项集
      - Prune candidate itemsets containing subsets of length k that are infrequent （修剪不频繁的项集）
      - Count the support of each candidate by scanning the DB
      - Eliminate candidates that are infrequent, leaving only those that are frequent （只留下频繁项集）

## 关联规则挖掘——生成候选项集合

- Apriori算法是根据有关频繁项集性质的先验知识Apriori Principal命名的。该算法使用一种逐层搜索的迭代方法，利用k-项集产生(k+1)-项集。

- 具体做法：首先找出频繁1-项集的集合，记为L1；再用L1找频繁2-项集的集合L2；再用L2找L3 ...如此下去，直到不能找到频繁k-项集为止。找每个Lk需要一次数据库扫描。

## 关联规则挖掘——生成候选项集合

- 整个过程由连接和剪枝两步组成，即：
  - (1)连接步
    - 为找Lk，可通过Lk-1与自己连接，产生一个候选k-项集的集合，该候选项集的集合记作Ck
      - 设 l1 和 l2 是 Lk-1 中的项集，记号 li[j] 表示 li 的第 j 项。为方便计，假定事务或项集中的项按字典次序排序。
      - 执行连接 $L_{k-1} \bowtie L_{k-1}$，其中 Lk-1 的元素是可连接的，如果它们前(k-2)个项相同。
        - 即， Lk-1 的元素 l1 和 l2 是可连接的，若：（ l1[1] = l2[1] ∧ l1 [2] = l2 [2] ∧ … ∧ l1[k-2] = l2[k-2] ∧ l1[k-1] < l2[k-1] ）而条件（ l1[k-1] < l2[k-1] ）可确保不产生重复的项集。

- 关联规则挖掘——生成候选项集合
  - 整个过程由连接和剪枝两步组成，即：
    - (2)剪枝步
      - Ck是Lk的超集，即它的成员不一定都是频繁项集，但所有的频繁k-项集都包含在Ck中。
      - 扫描数据库，确定Ck中每个候选项集的计数，从而确定Lk。然而， Ck可能很大，这样所涉及的计算量就很大。
      - 为了压缩Ck，可利用Apriori性质：**任何非频繁的(k-1)-项集都不可能是频繁k-项集的子集**。因此，若一个候选k-项集的(k-1)-项子集不在Lk-1中，则该候选也不可能是频繁的，从而可以从Ck中删除。

- 关联规则挖掘——候选项目集的生成函数
  - 以Lk-1作为输入，输出全部频繁k-项目集的一个超集。该函数包含两个操作，连接(join)与修剪(prune)。连接操作将Lk-1中的频繁项目集按如下方式进行拼接：

```
insert into Ck
select p.item1, p.item2, …, p.itemk-1, q. itemk-1
from Lk-1  p,  Lk-1  q
where p. item1=q. item1,…, p. itemk-2=q. itemk-2, p. itemk-1 < q. itemk-1;
```

## 关联规则挖掘——修剪操作

- 对Ck中任一候选项集c，若c的某个大小为k-1的子集不属于Lk-1，则将其从Ck中删除。
- forall itemsets c$\in$ Ck do

  forall (k-1)-subsets s of c do

  if (s$\notin$Lk-1) then

  delete c from Ck ;

# 数据挖掘基础



- 关联规则挖掘—— Apriori Algorithm

【例】一个Apriori的具体例子，该例基于右图某商店的事务DB。DB中有9个事务，Apriori假定事务中的项按字典次序存放。

| TID | 项ID的列表 |
|---|---|
| T100 | I1，I2，I5 |
| T200 | I2，I4 |
| T300 | I2，I3 |
| T400 | I1，I2，I4 |
| T500 | I1，I3 |
| T600 | I2，I3 |
| T700 | I1，I3 |
| T800 | I1，I2，I3，I5 |
| T900 | I1，I2，I3 |

- 关联规则挖掘—— Apriori Algorithm

(1) 在算法的第一次迭代，每个项都是*候选1-项集*的集合$C_1$的成员。算法简单地扫描所有的事务，对每个项的出现次数计数。

扫描D，对每个候选计数 →

$C_1$

| 项集 | 支持度计数 |
|---|---|
| {I1} | 6 |
| {I2} | 7 |
| {I3} | 6 |
| {I4} | 2 |
| {I5} | 2 |

## 关联规则挖掘—— Apriori Algorithm

（2）设最小支持计数为2，可以确定*频繁*1-*项集*的集合$L_1$ 。它由具有最小支持度的*候选*1-*项集*组成。

比较候选支持度计数与最小支持度计数

$L_1$

| 项集 | 支持度计数 |
|---|---|
| {I1} | 6 |
| {I2} | 7 |
| {I3} | 6 |
| {I4} | 2 |
| {I5} | 2 |

# 数据挖掘基础



- 关联规则挖掘—— Apriori Algorithm

（3）为发现频繁2-项集的集合$L_2$，算法使用 $L_1 \bowtie L_1$ 产生候选2-项集集合$C_2$ 。

由$L_1$产生候选$C_2$ →

$C_2$

| 项集 |
| --- |
| {I1，I2} |
| {I1，I3} |
| {I1，I4} |
| {I1，I5} |
| {I2，I3} |
| {I2，I4} |
| {I2，I5} |
| {I3，I4} |
| {I3，I5} |
| {I4，I5} |

- 关联规则挖掘—— Apriori Algorithm

（4）扫描D中事务，计算$C_2$中每个候选项集的支持计数。

$C_2$

扫描D，对每个候选计数

| 项集 | 支持度计数 |
|---|---|
| {I1，I2} | 4 |
| {I1，I3} | 4 |
| {I1，I4} | 1 |
| {I1，I5} | 2 |
| {I2，I3} | 4 |
| {I2，I4} | 2 |
| {I2，I5} | 2 |
| {I3，I4} | 0 |
| {I3，I5} | 1 |
| {I4，I5} | 0 |

- 关联规则挖掘——Apriori Algorithm

（5）确定*频繁2-项集*的集合$L_2$，它由具有最小支持度的$C_2$中的*候选2-项集*组成。

*比较候选支持度计数*
*与最小支持度计数*

→

$L_2$

| 项集 | 支持度计数 |
|---|---|
| {I1，I2} | 4 |
| {I1，I3} | 4 |
| {I1，I5} | 2 |
| {I2，I3} | 4 |
| {I2，I4} | 2 |
| {I2，I5} | 2 |

- 关联规则挖掘—— Apriori Algorithm

（6）候选3-*项集*的集合$C_3$的产生如下：

① *连接*: $C_3 = L_2 \bowtie L_2$

= {{I1, I2}, {I1, I3}, {I1, I5}, {I2, I3}, {I2, I4}, {I2, I5}}

$\bowtie$

{{I1, I2}, {I1, I3}, {I1, I5}, {I2, I3}, {I2, I4}, {I2,I5}}

= {{I1, I2, I3}, {I1, I2, I5}, {I1, I3, I5}, {I2, I3, I4}, {I2, I3, I5}, {I2, I4, I5}}

# 数据挖掘基础



## 关联规则挖掘—— Apriori Algorithm

$L_2$ {{I1, I2}，{I1, I3}，{I1, I5}，{I2, I3}，{I2, I4}，{I2, I5}}

$C_3$ {{I1, I2, I3}，{I1, I2, I5}，{I1, I3, I5}，{I2, I3, I4}，{I2, I3, I5}，{I2, I4, I5}}

② 利用Apriori性质剪枝：频繁项集的所有子集必须是频繁的。存在候选项集，判断其子集是否频繁。

- {I1,I2,I3}的2-*项子集*是{I1,I2}，{I1,I3}和{I2,I3},
  它们都是$L_2$的元素。因此保留{I1,I2,I3}在$C_3$中。
- {I1,I2,I5}的2-*项子集*是{I1,I2}，{I1,I5}和{I2,I5},
  它们都是$L_2$的元素。因此保留{I1,I2,I5}在$C_3$中。
- {I1,I3,I5}的2-*项子集*是{I1,I3}，{I1,I5}和{I3,I5},

  {I3,I5}不是$L_2$的元素，因而不是频繁的，由$C_3$中删除{I1,I3,I5}。

....

# 数据挖掘基础



- 关联规则挖掘—— Apriori Algorithm

③ 这样，剪枝后$C_3$ = {{I1,I2,I3}，{I1,I2,I5}}。

（7）扫描D中事务，以确定$L_3$，它由$C_3$中具有最小支持度的的*候选3-项集*组成。

**由$L_2$产生候选$C_3$**

| 项集 C3 |
|---|
| {I1，I2，I3} |
| {I1，I2，I5} |

**扫描D，对每个候选计数**

| 项集C3 | 支持度计数 |
|---|---|
| {I1，I2，I3} | 2 |
| {I1，I2，I5} | 2 |

- 关联规则挖掘—— Apriori Algorithm

（8）算法使用$L_3 \bowtie L_3$产生候选4-项集的集合$C_4$。尽管连接产生结果 {{I1,I2,I3,I5}},这个项集被剪去，因为它的子集{I2,I3,I5}不是频繁的。则 $C_4 = \psi$, 因此算法终止，找出了所有的频繁项集。

$L_3$

比较候选支持度计数与最小支持度计数

| 项集 | 支持度计数 |
|---|---|
| {I1，I2，I3} | 2 |
| {I1，I2，I5} | 2 |

# 练习



- 设min_sup = 50%求出右图事务列表中所有的频繁项集
- （包括1-频繁，2-频繁，3-频繁等，给出求解过程）

| TID | Item |
|---|---|
| 100 | 1 2 3 4 |
| 200 | 1 2 5 |
| 300 | 1 2 3 5 |
| 400 | 2 4 5 |
| 500 | 1 2 3 |

# 练习

D

| TID | Item |
| --- | --- |
| 100 | 1 2 3 4 |
| 200 | 1 2 5 |
| 300 | 1 2 3 5 |
| 400 | 2 4 5 |
| 500 | 1 2 3 |

$L_1$

| Itemset | Support |
| --- | --- |
| {1} | 4 |
| {2} | 5 |
| {3} | 3 |
| {5} | 3 |

$C_2$

| Itemset | Support |
| --- | --- |
| {1 2} | 4 |
| {1 3} | 3 |
| {1 5} | 2 |
| {2 3} | 3 |
| {2 5} | 3 |
| {3 5} | 1 |

$L_2$

| Itemset | Support |
| --- | --- |
| {1 2} | 3 |
| {1 3} | 4 |
| {2 3} | 3 |
| {2 5} | 3 |

$C_3$

| Itemset | Support |
| --- | --- |
| {1 2 3} | 3 |
| {2 3 5} | 1 |

$L_3$

| Itemset | Support |
| --- | --- |
| {1 2 3} | 3 |

- 关联规则挖掘—— 规则生成(Rule Generation)
- Given a frequent itemset L, find all non-empty subsets $f \subset L$ such that $f \rightarrow L - f$ satisfies the minimum confidence requirement
  - If {A,B,C,D} is a frequent itemset, candidate rules:

    ABC $\rightarrow$D, ABD $\rightarrow$C, ACD $\rightarrow$B, BCD $\rightarrow$A,
    A $\rightarrow$BCD, B $\rightarrow$ACD, C $\rightarrow$ABD, D $\rightarrow$ABC
    AB $\rightarrow$CD, AC $\rightarrow$ BD, AD $\rightarrow$ BC, BC $\rightarrow$AD,
    BD $\rightarrow$AC, CD $\rightarrow$AB,

- **If |L| = k, then there are $2^k - 2$ candidate association rules** (ignoring $L \rightarrow \varnothing$ and $\varnothing \rightarrow L$)

- 关联规则挖掘—— 规则生成(Rule Generation)
- How to efficiently generate rules from frequent itemsets?
  - In general, **confidence** does not have an **anti-monotone** property

    $c(ABC \rightarrow D)$ can be larger or smaller than $c(AB \rightarrow D)$

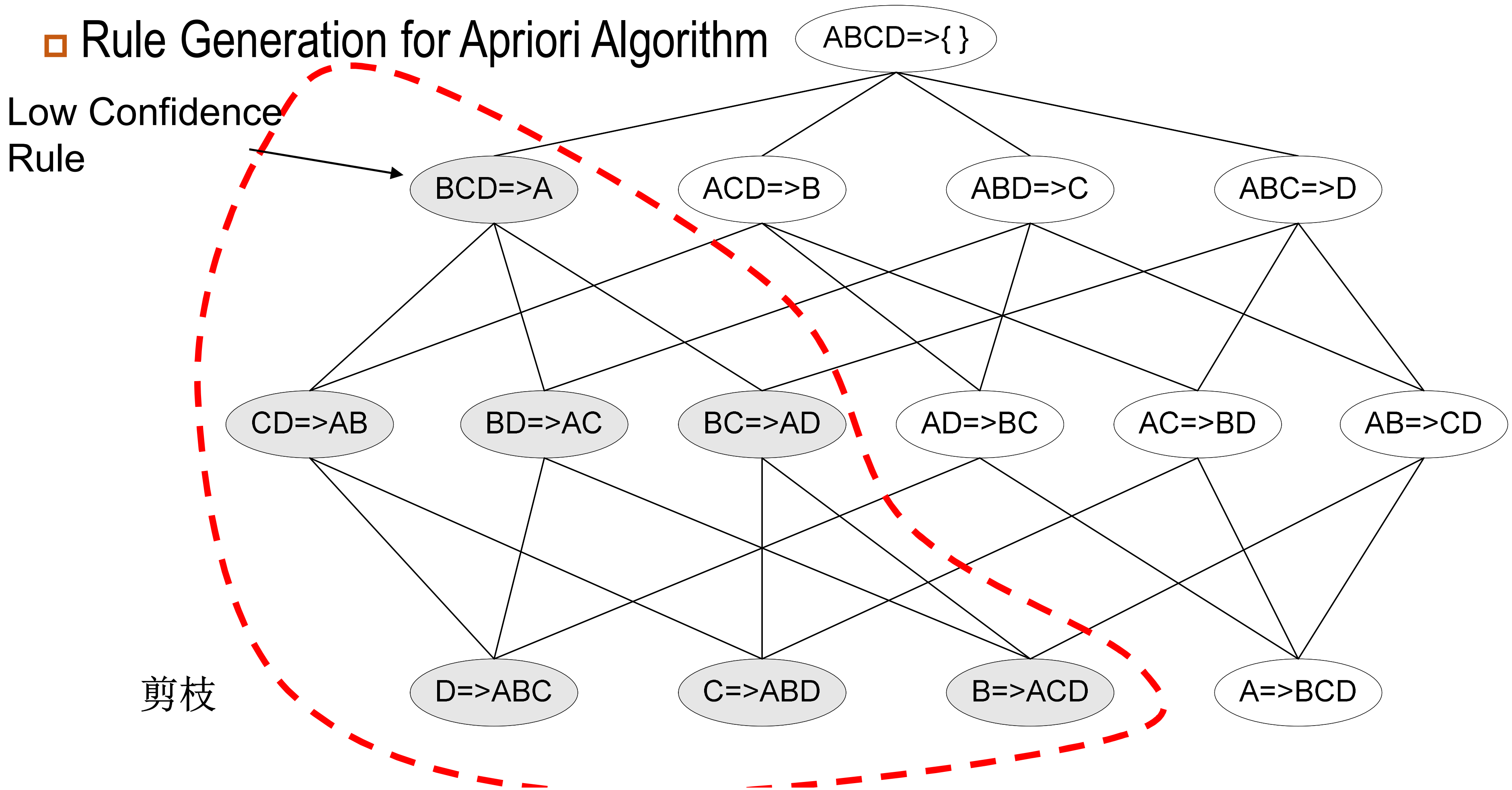
  - But confidence of rules generated from the same itemset has an anti-monotone property
  - e.g., L = {A,B,C,D}:

$$c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$$

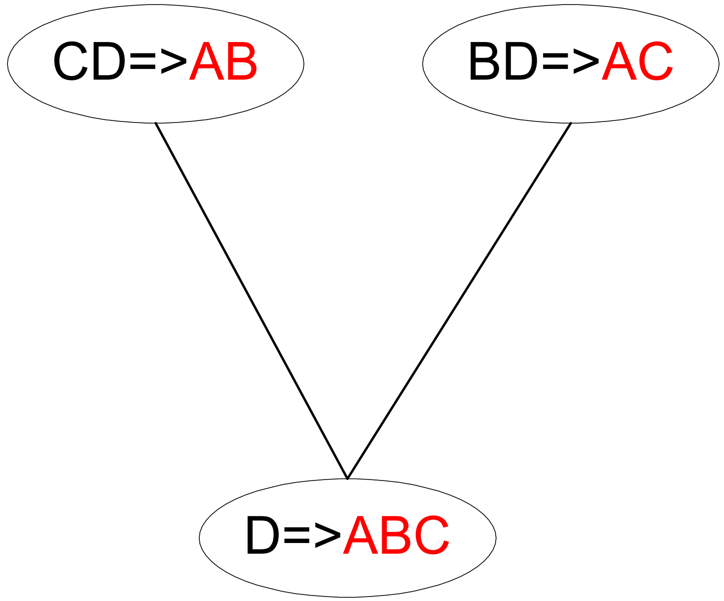
- Confidence is anti-monotone w.r.t. number of items on the RHS of the rule

# 数据挖掘基础

- 关联规则挖掘—— 规则生成(Rule Generation)
- Rule Generation for Apriori Algorithm



- 关联规则挖掘—— 规则生成(Rule Generation)
- Rule Generation for Apriori Algorithm
  - Candidate rule is generated by merging two rules that share the same prefix in the rule consequent
  - join(CD=>AB,BD=>AC) would produce the candidate rule D => ABC
  - Prune rule D=>ABC if its subset AD=>BC does not have high confidence



- 关联规则挖掘—— 如何处理连续属性？

| Gender | ··· | Age | Annual Income | No of hours spent online per week | No of email accounts | Privacy Concern |
|---|---|---|---|---|---|---|
| Female | ··· | 26 | 90K | 20 | 4 | Yes |
| Male | ··· | 51 | 135K | 10 | 2 | No |
| Male | ··· | 29 | 80K | 10 | 3 | Yes |
| Female | ··· | 45 | 120K | 15 | 3 | Yes |
| Female | ··· | 31 | 95K | 20 | 5 | Yes |
| Male | ··· | 25 | 55K | 25 | 5 | Yes |
| Male | ··· | 37 | 100K | 10 | 1 | No |
| Male | ··· | 41 | 65K | 8 | 2 | No |
| Female | ··· | 26 | 85K | 12 | 1 | No |
| ··· | ··· | ··· | ··· | ··· | ··· | ··· |

**Example of Association Rule:**

{Gender=Male, Age $\in$ [21,30)} $\rightarrow$ {No. of hours online $\geq$ 10}

## 关联规则挖掘—— 产生分类、异常检测的效果

- Example: Internet Usage Data

| Gender | Level of Education | State | Computer at Home | Online Auction | Chat Online | Online Banking | Privacy Concerns |
|---|---|---|---|---|---|---|---|
| Female | Graduate | Illinois | Yes | Yes | Daily | Yes | Yes |
| Male | College | California | No | No | Never | No | No |
| Male | Graduate | Michigan | Yes | Yes | Monthly | Yes | Yes |
| Female | College | Virginia | No | Yes | Never | Yes | Yes |
| Female | Graduate | California | Yes | No | Never | No | Yes |
| Male | College | Minnesota | Yes | Yes | Weekly | Yes | Yes |
| Male | College | Alaska | Yes | Yes | Daily | Yes | No |
| Male | High School | Oregon | Yes | No | Never | No | No |
| Female | Graduate | Texas | No | No | Monthly | No | No |
| ... | ... | ... | ... | ... | ... | ... | ... |

{Level of Education=Graduate, Online Banking=Yes}

$\rightarrow$ {Privacy Concerns = Yes}

- 关联规则挖掘—— Pattern Evaluation（规则评估）
  - 关联规则往往会生成很多规则
    - many of them are uninteresting or redundant（冗余）
    - Redundant if {A,B,C} → {D} and {A,B} → {D} have same support & confidence
  - Interestingness measures（兴趣度量） can be used to prune（修剪）/rank the derived patterns 衍生模式
  - In the original formulation of association rules, **support & confidence** are the only measures used

## 关联规则挖掘—— Drawback of Confidence

| | Coffee | $\overline{\text{Coffee}}$ | |
|---|---|---|---|
| Tea | 15 | 5 | 20 |
| $\overline{\text{Tea}}$ | 75 | 5 | 80 |
| | 90 | 10 | 100 |

Association Rule: Tea $\rightarrow$ Coffee

Support = P(Coffee,Tea| ALL) =**?** ; Confidence= P(Coffee|Tea) = ?

but P(Coffee) = ?

$\Rightarrow$ Although confidence is high, rule is misleading

$\Rightarrow$ P(Coffee|$\overline{\text{Tea}}$) =**?**

# 数据挖掘基础



- 关联规则挖掘——计算兴趣度量
  - Given a rule X → Y, information needed to compute rule interestingness can be obtained from a contingency table(相依表)

Contingency table for X → Y

| | $Y$ | $\overline{Y}$ | |
|---|---|---|---|
| $X$ | $f_{11}$ | $f_{10}$ | $f_{1+}$ |
| $\overline{X}$ | $f_{01}$ | $f_{00}$ | $f_{o+}$ |
| | $f_{+1}$ | $f_{+0}$ | $\|T\|$ |

$f_{11}$: support of X and Y
$f_{10}$: support of X and $\overline{Y}$
$f_{01}$: support of $\overline{X}$ and Y
$f_{00}$: support of $\overline{X}$ and $\overline{Y}$

Used to define various measures

- support, confidence, lift, Gini, J-measure, etc.

# 数据挖掘基础



- 关联规则挖掘—— Statistical Independence
  - **Population of 1000 students**
    - 600 students know how to swim (S)
    - 700 students know how to bike (B)
    - 420 students know how to swim and bike (S,B)

    - $P(S \wedge B) = 420/1000 = 0.42$
    - $P(S) \times P(B) = 0.6 \times 0.7 = 0.42$

    - $P(S \wedge B) = P(S) \times P(B)$ => **Statistical independence**
    - $P(S \wedge B) > P(S) \times P(B)$ => **Positively correlated**
    - $P(S \wedge B) < P(S) \times P(B)$ => **Negatively correlated**

- 关联规则挖掘—— Statistical-based Measures
  - Measures that take into account statistical dependence

$$Lift = \frac{P(Y \mid X)}{P(Y)}$$

$$Interest = \frac{P(X,Y)}{P(X)P(Y)}$$

$$PS = P(X,Y) - P(X)P(Y)$$

$$\phi - coefficient = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}}$$

There are lots of measures proposed in the literature

Some measures are good for certain applications, but not for others

What criteria should we use to determine whether a measure is good or bad?

What about Apriori-style support based pruning? How does it affect these measures?

| # | Measure | Formula |
|---|---|---|
| 1 | $\phi$-coefficient | $\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$ |
| 2 | Goodman-Kruskal's ($\lambda$) | $\frac{\sum_j \max_k P(A_j,B_k)+\sum_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-\max_k P(B_k)}$ |
| 3 | Odds ratio ($\alpha$) | $\frac{P(A,B)P(\overline{A},\overline{B})}{P(A,\overline{B})P(\overline{A},B)}$ |
| 4 | Yule's $Q$ | $\frac{P(A,B)P(\overline{AB})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{AB})+P(A,\overline{B})P(\overline{A},B)} = \frac{\alpha-1}{\alpha+1}$ |
| 5 | Yule's $Y$ | $\frac{\sqrt{P(A,B)P(\overline{AB})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{AB})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}} = \frac{\sqrt{\alpha}-1}{\sqrt{\alpha}+1}$ |
| 6 | Kappa ($\kappa$) | $\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$ |
| 7 | Mutual Information ($M$) | $\frac{\sum_i \sum_j P(A_i,B_j)\log\frac{P(A_i,B_j)}{P(A_i)P(B_j)}}{\min(-\sum_i P(A_i)\log P(A_i), -\sum_j P(B_j)\log P(B_j))}$ |
| 8 | J-Measure ($J$) | $\max\Big(P(A,B)\log(\frac{P(B\|A)}{P(B)})+P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}),$ $P(A,B)\log(\frac{P(A\|B)}{P(A)})+P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})\Big)$ |
| 9 | Gini index ($G$) | $\max\Big(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A})[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]$ $-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B})[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]$ $-P(A)^2-P(\overline{A})^2\Big)$ |
| 10 | Support ($s$) | $P(A,B)$ |
| 11 | Confidence ($c$) | $\max(P(B\|A),P(A\|B))$ |
| 12 | Laplace ($L$) | $\max\Big(\frac{NP(A,B)+1}{NP(A)+2},\frac{NP(A,B)+1}{NP(B)+2}\Big)$ |
| 13 | Conviction ($V$) | $\max\Big(\frac{P(A)P(\overline{B})}{P(A\overline{B})},\frac{P(B)P(\overline{A})}{P(B\overline{A})}\Big)$ |
| 14 | Interest ($I$) | $\frac{P(A,B)}{P(A)P(B)}$ |
| 15 | cosine ($IS$) | $\frac{P(A,B)}{\sqrt{P(A)P(B)}}$ |
| 16 | Piatetsky-Shapiro's ($PS$) | $P(A,B)-P(A)P(B)$ |
| 17 | Certainty factor ($F$) | $\max\Big(\frac{P(B\|A)-P(B)}{1-P(B)},\frac{P(A\|B)-P(A)}{1-P(A)}\Big)$ |
| 18 | Added Value ($AV$) | $\max(P(B\|A)-P(B),P(A\|B)-P(A))$ |
| 19 | Collective strength ($S$) | $\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})}\times\frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$ |
| 20 | Jaccard ($\zeta$) | $\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$ |
| 21 | Klosgen ($K$) | $\sqrt{P(A,B)}\max(P(B\|A)-P(B),P(A\|B)-P(A))$ |

# 辛普森悖论 Simpson's Paradox

- Observed relationship in data may be influenced by the presence of other confounding factors (hidden variables)
  - Hidden variables may cause the observed relationship to disappear or reverse its direction!

- Proper stratification(分层) is needed to avoid generating spurious(虚假) patterns

# 辛普森悖论 Simpson’s Paradox



- Association patterns may behave differently at the local level from the global level

| Global Observation | Local Observation | Pitfalls(陷阱) |
| --- | --- | --- |
| Significant | Insignificant | False Positive |
| Insignificant | Significant | False Negative |

- Simpson’s Paradox
  - **The (global) pattern differs from each local segment**
  - Direction of the correlation might be **reversed**

# Simpson's Paradox: An Example*

- UC Berkeley was sued for bias against women applying to graduate school.

| Men | | Women | |
|---|---|---|---|
| #Applicants | %Admitted | #Applicants | %Admitted |
| 832 | 44% | 366 | 11% |

- In fact, most departments had a small bias against men

| Major | Men | | Women | |
|---|---|---|---|---|
| | #Applicants | %Admitted | #Applicants | %Admitted |
| B | 560 | 63% | 25 | 68% |
| F | 272 | 6% | 341 | 7% |

**适当的数据分层有助于避免辛普森悖论**

- **Adapted from the example at [http://en.wikipedia.org/wiki/Simpson's_paradox](http://en.wikipedia.org/wiki/Simpson's_paradox) . See the following paper for more details: P.J. Bickel, E.A. Hammel and J.W. O'Connell (1975). "Sex Bias in Graduate Admissions: Data From Berkeley". Science 187 (4175): 398–404.**
- **https://en.wikipedia.org/wiki/Simpson%27s_paradox**

# 练习

- 请依据下表计算出关于早餐的关联规则{面包}->{豆浆}的支持度与置信度。结合此例说明支持度与置信度的不足，同时给出一个可以更准确的评价关联规则的方法（也可以是多个）。

| | 买豆浆 | 不买豆浆 | |
|---|---|---|---|
| 买面包 | 90 | 30 | 120 |
| 不买面包 | 390 | 90 | 480 |
| | 480 | 120 | 600 |

# 数据挖掘基础

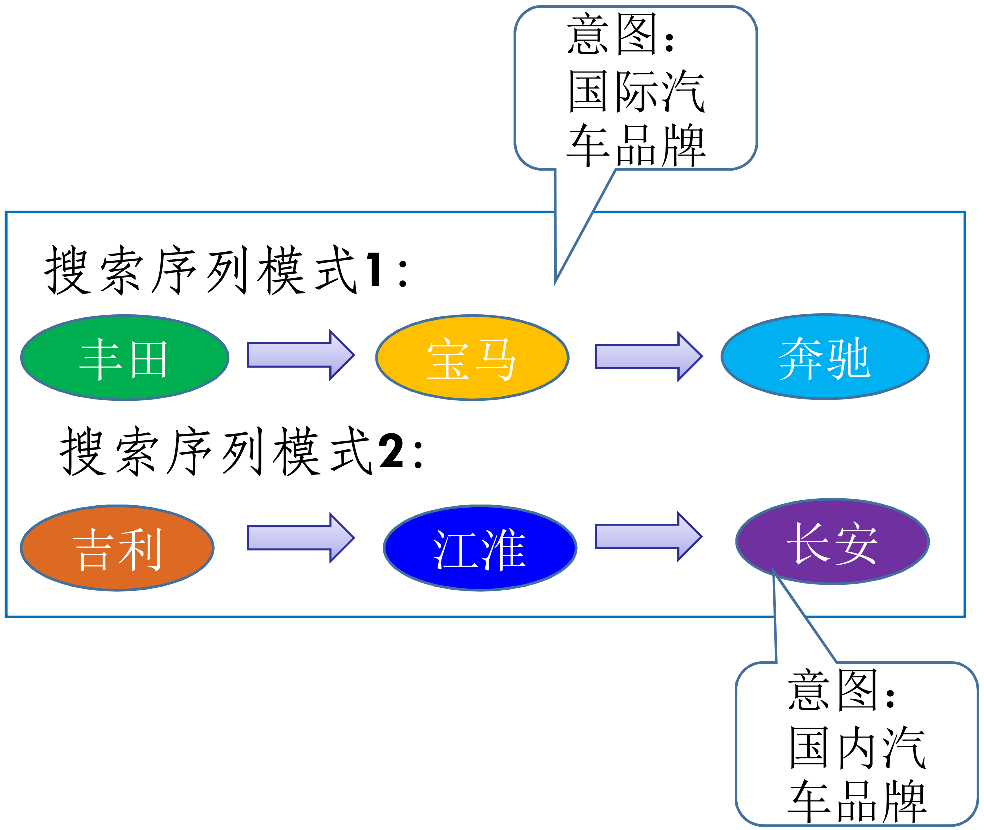


- 关联规则挖掘—— 扩展算法：序列模式挖掘
  - 序列模式的概念最早是由Agrawal和Srikant 提出的。
  - 动机：大型连锁超市的交易数据有一系列的用户事务数据库，每一条记录包括用户的ID，事务发生的时间和事务涉及的项目。如果能在其中挖掘涉及事务间关联关系的模式，即用户几次购买行为间的联系，可以采取更有针对性的营销措施。
  - 应用领域：
    - 客户购买行为模式预测
    - Web访问模式预测
    - 疾病诊断
    - 自然灾害预测DNA序列分析

## 关联规则挖掘—— 扩展算法：序列模式挖掘

- 查询扩展是搜索领域一个重要的问题。用户提交的查询往往不能完全反映其信息需求。一些研究工作尝试用用户的查询序列模式来辅助原始查询

| SID | Search Session |
|---|---|
| 1 | 丰田➔雷诺➔宝马➔奔驰 |
| 2 | 宝马➔奔驰➔法拉利 |
| 3 | 本田➔丰田➔通用➔宝马 |
| 4 | 吉利➔奇瑞➔长城➔江淮 |
| 5 | 比亚迪➔吉利➔江淮➔长安 |
| 6 | 长城➔江淮➔华泰➔长安 |



- 序列模式挖掘算法
  - 类Apriori算法
    - 序列模式的任一子序列也是序列模式。算法首先自底向上的根据较短的序列模式生成较长的候选序列模式，然后计算候选序列模式的支持度。典型的代表有GSP算法, spade算法等。
  - 基于划分的模式生长算法
    - 基于分治的思想，迭代的将原始数据集进行划分，减少数据规模，同时在划分的过程中动态的挖掘序列模式，并将新发现的序列模式作为新的划分元。典型的代表有FreeSpan算法和prefixSpan算法。
  - 基于序列比较的算法
    - 首先定义序列的大小度量，接着从小到大的枚举原始序列数据库中包含的所有k-序列，理论上所有的k-序列模式都能被找到。算法制定特定的规则加快这种枚举过程。典型的代表为Disc-all算法。
- Lei Zhang, Ping Luo, Linpeng Tang, Enhong Chen, Qi Liu, Min Wang, Hui Xiong, Occupancy-based Frequent Pattern Mining, ACM Transactions on Knowledge Discovery from Data.

# 数据挖掘基础



## 序列模式挖掘算法扩展—根据序列模式预测未来

- 时间序列分析方法
  - 序列自回归模型(AR)、移动平均模型(MA)、自回归移动平均模型(ARMA)

如果时间序列 $X_t$ 是它的前期值和随机项的线性函数，即可表示为

$$X_t = \varphi_1 X_{t-1} + \varphi_2 X_{t-2} + \cdots + \varphi_p X_{t-p} + u_t \quad 【1】$$

【1】式称为 $p$ 阶自回归模型，记为AR（ $p$ ）

- 隐马尔可夫模型(Hiden Markov Model, HMM) ---CRF

| SID | Search Session |
|---|---|
| 1 | 丰田➔雷诺➔宝马➔奔驰 |
| 2 | 宝马➔奔驰➔法拉利 |
| 3 | 本田➔丰田➔通用➔宝马 |
| 4 | 吉利➔奇瑞➔长城➔江淮 |
| 5 | 比亚迪➔吉利➔江淮➔长安 |
| 6 | 长城➔江淮➔华泰➔长安 |

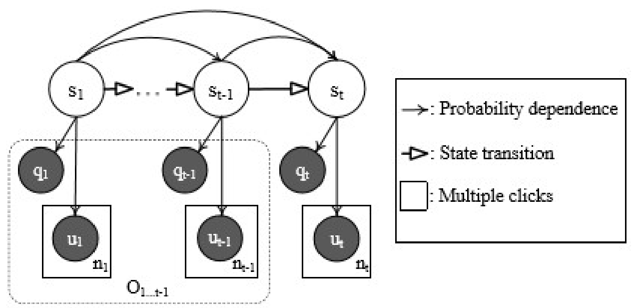



Figure 1: Graphical structure of the vlHMM.

# 数据挖掘基础



## 序列模式挖掘算法扩展—根据序列模式预测未来

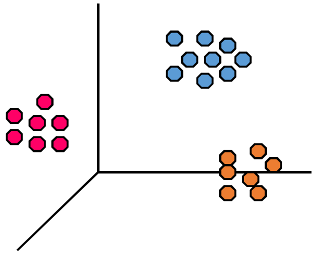
- Huanhuan Cao, Derek Hao Hu, Dou Shen, Daxin Jiang, Jian-Tao Sun, Enhong Chen, Qiang Yang, **Context-Aware Query Classification**, ***SIGIR'2009***.
- Huanhuan Cao, Daxin Jiang, Jian Pei, Enhong Chen, Hang Li, **Towards Context-Aware Search by Learning A Very Large Variable Length Hidden Markov Model from Search Logs**, ***WWW'2009***.
- Huanhuan Cao, Daxin Jiang, Jian Pei, Qi He, Zhen liao, Enhong Chen, Hang Li, **Context-Aware Query Suggestion by Mining Click-Through and Session Data**, ***KDD'2008***.**(Best Application Paper Award)**.
- Biao Chang, Hengshu Zhu, Yong Ge, Enhong Chen*, Hui Xiong, Chang Tan, **Predicting the Popularity of Online Serials with Autoregressive Models**, ***CIKM'2014***.
- Hengshu Zhu, Chuanren Liu, Yong Ge, Hui Xiong, Enhong Chen, **Popularity Modeling for Mobile Apps: A Sequential Approach**, *IEEE Transactions on Cybernetics* ***(IEEE TC)***, 45(7): 1303-1314, July 2015.
- Hongke Zhao, Qi Liu*, Hengshu Zhu, Yong Ge, Enhong Chen, Yan Zhu, Junping Du, **A Sequential Approach to Market State Modeling and Analysis in Online P2P Lending**, *IEEE Transactions on Systems, Man, and Cybernetics: Systems* ***(IEEE TSMC-S)***, accepted, 2017.
- Le Wu, Yong Ge, Qi Liu, Enhong Chen, Richang Hong, Meng Wang, Junping Du, **Modeling the Evolution of Users' Preferences and Social Links in Social Networking Services**, *IEEE Transactions on Knowledge and Data Engineering* ***(IEEE TKDE)***, 29(6): 1240-1253, June 2017.
- Le Wu, Qi Liu, Enhong Chen, Xing Xie and Chang Tan, **Product Adoption Rate Prediction: A Multi-factor View**, ***SDM'2015***.

# 数据挖掘基础



- 常用方法——关于四个任务有哪些常用方法？

**Clustering**



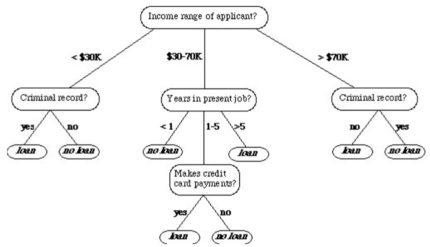
**Data**

| | T | | H | | P | |
|---|---|---|---|---|---|---|
| | L | H | L | H | L | H |
| J | -6.0 | 8.8 | 60 | 100 | 986 | 1044 |
| F | -2.8 | 10.9 | 48 | 100 | 973 | 1025 |
| M | -5.6 | 17.7 | 34 | 100 | 976 | 1037 |
| A | -1.2 | 22.2 | 27 | 100 | 996 | 1036 |
| M | -0.8 | 27.8 | 25 | 100 | 1003 | 1034 |
| J | 5.2 | 29.1 | 26 | 100 | 998 | 1030 |
| J | 9.8 | 30.6 | 23 | 99 | 997 | 1027 |
| A | 5.6 | 26.1 | 31 | 100 | 992 | 1029 |
| S | 5.2 | 24.8 | 35 | 100 | 998 | 1028 |
| O | -0.4 | 21.3 | 42 | 100 | 990 | 1031 |
| N | -7.6 | 17.3 | 55 | 100 | 963 | 1023 |
| D | -10.4 | 9.2 | 53 | 100 | 987 | 1039 |

*table 17a*

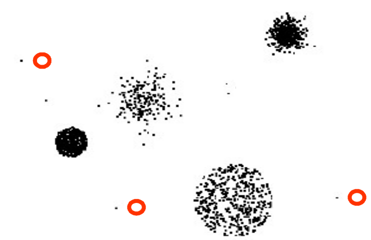
2010 monthly weather variation, Cambridge (UK)

**Classification**




**Association Analysis**

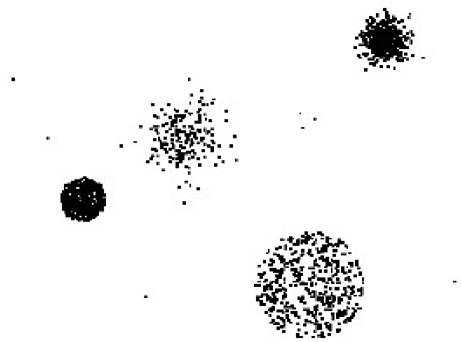

**Anomaly Detection**



# 数据挖掘基础



- Anomaly Detection——异常/离群点 检测
  - 什么是异常/离群点?
    - The set of data points that are considerably different than the remainder of the data
  - 通常情况下异常是罕见的
    - One in a thousand occurs often if you have lots of data
    - Context is important, e.g., freezing temps in July
  - 可能是重要的也可能是有害的
    - 10 foot tall 2 year old
    - Unusually high blood pressure



# 数据挖掘基础



- Anomaly Detection——异常/离群点 检测
- 给定数据集 D, 发现所有点 **x** ∈ D， 其异常得分大于阈值t
- 给定数据集 D, 发现前 top-n 得分的点 **x** ∈ D
- 给定数据集 D, 包含一般 (未标记) 数据点, 对于测试点 t **x**, 根据 D 计算它的异常得分

# 数据挖掘基础



- Anomaly Detection——异常检测模型
- Build a model for the data and see
  - Unsupervised
    - 异常是那些不能拟合的点
    - 异常是那些扭曲模型的点
    - Examples:
      - Statistical distribution
      - Clusters
      - Regression
      - Geometric
      - Graph
  - Supervised
    - Anomalies are regarded as a rare class
    - Need to have training data

- Anomaly Detection——异常检测的方法
  - 基于邻近度的离群点检测
    - Anomalies are points far away from other points
    - Can detect this graphically in some cases
  - 基于密度的离群点检测
    - Low density points are outliers
  - 模式匹配
    - Create profiles or templates of atypical but important events or objects
    - Algorithms to detect these patterns are usually simple and efficient

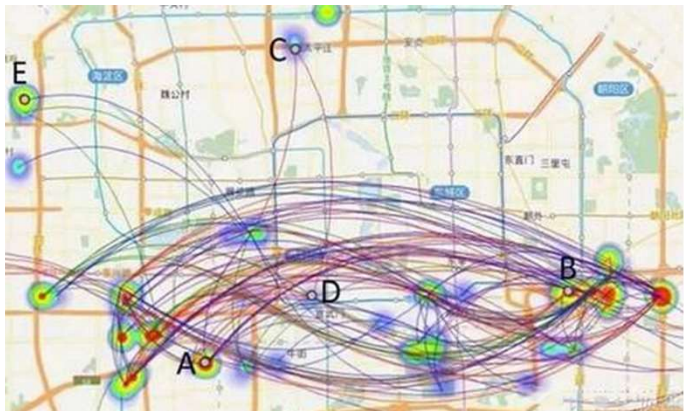
[1]Xue Bai, Yun Xiong, Yangyong Zhu, Qi Liu and Zhiyuan Chen. Co-anomaly Event Detection in Multiple Temperature Series. Springer KSEM 2013, pages: 1-14,2013. **(Best Paper Award).**

[2]Hengshu Zhu, Hui Xiong, Yong Ge, and Enhong Chen, Discovery of Ranking Fraud for Mobile Apps, IEEE Transactions on Knowledge and Data Engineering (IEEE TKDE).

# 数据挖掘基础



- Anomaly Detection——异常检测
- 大数据告诉你：公交车上谁是小偷！
  - （a）正常出行者，主要在居住地、工作地、途经区域活动
  - （b）旅游者，频繁访问圆明园、天安门、南锣鼓巷等景点区域。
  - （c）购物者，主要访问王府井、西单等购物区域。
  - （d）扒手，他们是一种流浪的模式，没有清晰的目的地，他们频繁地换乘，随机的停留，经常进行短途的出行。他们还（一段时间内）频繁地访问多种功能区：交通枢纽（例如西直门）、购物区（例如王府井）、景点（例如鼓楼）


# 数据挖掘基础

- 数据挖掘定义、四类任务及其应用场景
- 分类任务
  - 有监督：决策树、朴素贝叶斯、K近邻、SVM、集成分类器、评估方法
- 聚类任务
  - 无监督：K-Means、DBSCAN、评估方法
- 关联规则挖掘
  - 支持度和置信度、Apriori、序列模式挖掘、评估方法
- 异常检测

**Tips:** 在设计针对大数据与小数据的挖掘方法时，所用的思想在本质上是一致的。

# THANK YOU!