

Notes on CSC4008: Data Mining

By Renzhao Li

Spring 2019 in The Chinese University of Hong Kong, Shenzhen

Chapter 2: Getting to Know your data

Attributes	Example
Norminal attributes	red, black, brown
Binary attributes	0, 1
Ordinal attributes	A+, A, A-, B+
Numerical attributes	10Kg, 20Kg

2.1.1 “Measuring the Central Tendency: Mean, Median, and Mode”

<1> mean: 平均值

<2> median: 中位数, 如果有偶数个样本则是中间两个数的平均值。由于样本数量太大的时候计算中位数很麻烦, 可以用以下公式估算:

$$median = L_1 + \left(\frac{N/2 - (\sum freq)_l}{freq_{median}} \right) width$$

where L_1 is the lower boundary of the median interval, N is the number of values in the entire data set, $\sum freq$ is the sum of the frequencies of all of the intervals that are lower than the median interval, $freq_{median}$ is the frequency of the median interval, and $width$ is the width of the median interval.

<3> mode: 在样本中出现次数最多的数据就是mode, 如果两个数据值都出现最多次就是bimodal, 三个就是trimodal, 依此类推。如果每个数据值都只出现一次则没有mode存在。

对于unimodal (只有一个mode的数据集) 存在以下empirical relation:

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

<4> midrange: 样本中最大值和最小值的平均值

“In a unimodal frequency curve with perfect symmetric data distribution, the mean, median, and mode are all at the same center value”

2.22 “Measuring the Dispersion of Data: Range, Quartiles, Variance, Standard Deviation, and Interquartile Range”

<1> Range: 最大值与最小值之差

<2> Quantile: 分位点。 “The k th q -quantile for a given data distribution is the value x such that at most k/q of the data values are less than x and at most $(q - k)/q$ of the data values are more than x , where k is an integer such that $0 < k < q$. There are $q - 1$ q -quantiles.”

<3> Quartiles: 4-quantiles。

first quartile: denoted by Q_1 , 25%

second quartile: denoted by Q_2 , 50%

<4> Percentiles: 100-quantiles。

<5> interquartile range (IQR): $IQR = Q_3 - Q_1$

<6> Five-number summary: Minimum, Q_1 , Median, Q_3 , Maximum

“The five-number summary of a distribution consists of the median (Q_2), the quartiles Q_1 and Q_3 , and the smallest and largest individual observations, written in the order of Minimum, Q_1 , Median, Q_3 , Maximum”

<7> Boxplot: 箱型图

1. “Typically, the ends of the box are at the quartiles so that the box length is the interquartile range.”
2. “The median is marked by a line within the box.”
3. “Two lines (called whiskers) outside the box extend to the smallest (Minimum) and largest (Maximum) observations”

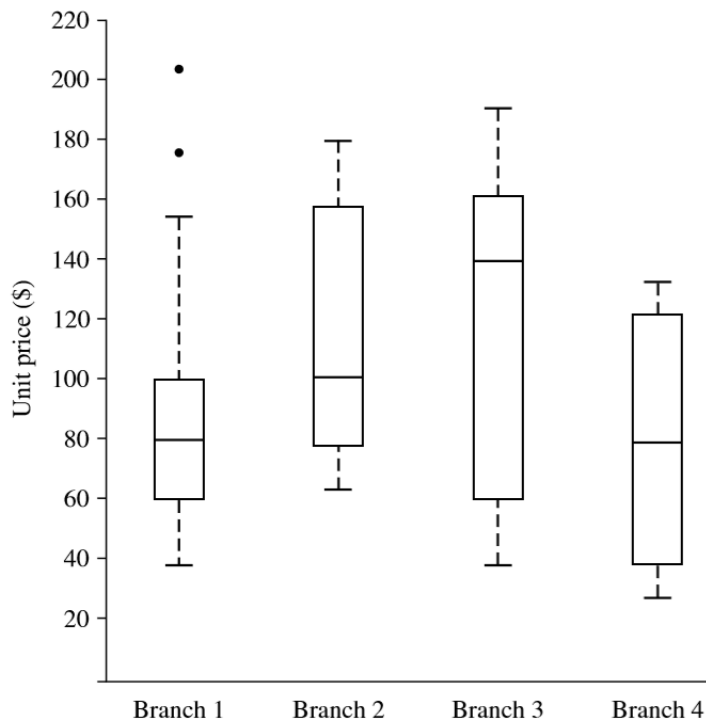


Figure 2.3 Boxplot for the unit price data for items sold at four branches of *AllElectronics* during a given time period.

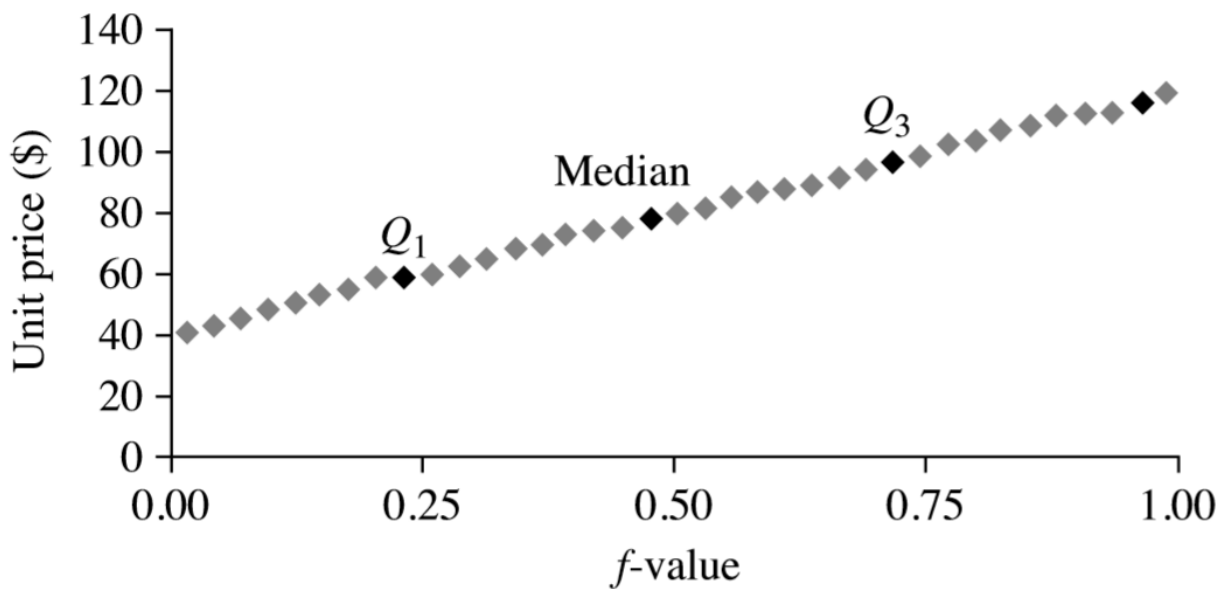
<8> Variance:

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 = \left(\frac{1}{N} \sum_{i=1}^N x_i^2 \right) - \bar{x}^2$$

Graphic Displays of Basic Statistical Descriptions of Data

<1> Quantile plot

纵坐标是样本的数值，横坐标是“有百分之多少的样本值在此数值之下”



其中对于纵坐标 x_i , 横坐标 f_i 是:

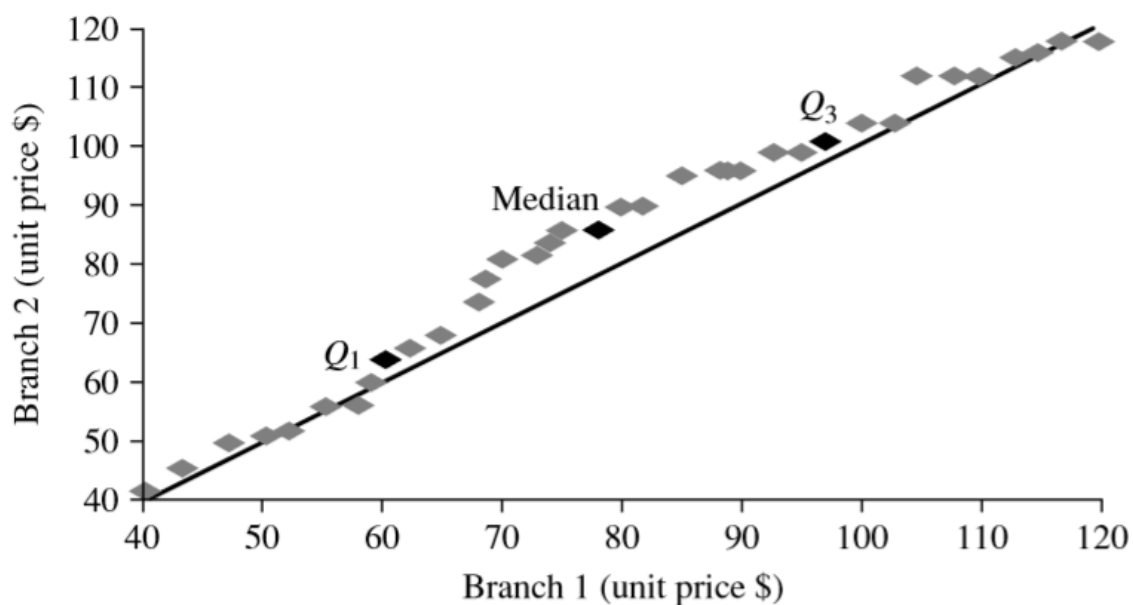
$$f_i = \frac{i - 0.5}{N}$$

f_i 的值从 $\frac{1}{2N}$ 到 $1 - \frac{1}{2N}$.

<2> Quantile-Quantile Plot

假如有两组数据: x_1, x_2, \dots, x_n 和 y_1, y_2, \dots, y_m , 如果 $M = N$,也就是两组数据一样多, 直接把 (x_i, y_i) 画在图上就行, 因为 x_i 和 y_i 对应一样的quantile。

如果 $M < N$ (第二组数据少一些), 就只在图里画 M 个点。



<3> Histogram 柱状图

2.3 Data Visualization

这里就略过吧，因为我觉得不太可能考。

2.4 Measuring Data Similarity and Dissimilarity

proximity: refers to similarity or dissimilarity

<1> Data matrix

n objects \times p attributes

$$\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

<2> Dissimilarity Matrix

$$\begin{bmatrix} 0 & & & & & \\ d(2,1) & 0 & & & & \\ d(3,1) & d(3,2) & 0 & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ d(n,1) & d(n,2) & d(n,3) & \cdots & \cdots & 0 \end{bmatrix}$$

Where $d(i, j)$ is the measured **dissimilarity** or "difference" between objects i and j

<3> Proximity Measures for Nominal Attributes

$$d(i, j) = \frac{p - m}{p}$$

一共p个attribute，有m个一样，p-m个不一样。

$$sim(i, j) = 1 - d(i, j) = \frac{m}{p}$$

<4> Proximity Measures for Binary Attributes

要比较Object i和Object j的proximity。假如一共有p个binary attributes，其中q个在i和j都是1；r个在i是1，在j是0；s个在i是0在j是1；t个在i和j都是0，那么：

Object j

Object i	1	0	sum
1	q	r	q + r
0	s	t	s + t
sum	q + s	r + t	p

1. symmetric binary dissimilarity

$$d(i, j) = \frac{r + s}{q + r + s + t}$$

2. asymmetric binary dissimilarity

$$d(i, j) = \frac{r + s}{q + r + s}$$

适用于：当positive(1)和negative(0)不一样重要的时候，t就不重要，可以被忽略了。

所以：

$$sim(i, j) = 1 - d(i, j) = \frac{q}{q + r + s}$$

sim(i, j)也叫 Jaccard coefficient 。

<5> Dissimilarity of Numeric Data: Minkowski Distance

Let $i = (x_{i1}, x_{i2}, \dots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \dots, x_{jp})$ be two objects described by p numeric attributes.

1. Euclidean distance:

$$d_{Euclidean}(i, j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + \dots + (x_{ip} - x_{jp})^2}$$

2. Manhattan distance:

$$d_{Manhattan}(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}|$$

3. Minkowski distance:

$$d_{Minkowski}(i, j) = \sqrt[h]{|x_{i1} - x_{j1}|^h + |x_{i2} - x_{j2}|^h + \dots + |x_{ip} - x_{jp}|^h}$$

闵科夫斯基距离是前两种距离的generalization。

当 $h \rightarrow \infty$ 时，Minkowski distance变成supremum distance 或者叫Chebyshev distance（切比雪夫距离）。

<6> Proximity Measures for Ordinal Attributes

假设第*i*个Object的attribute*f*的值是 x_{if} ，*f*一共有 M_f 个ordered state，把 x_{if} 换成对应的rank，也就是 r_{if} ，其中 $r_{if} \in \{1, \dots, M_f\}$

因为不同的attribute往往states数量不一样多，所以要映射到[0, 1]之间：

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

然后用<5>里的方法计算proximity。

<7> Cosine Similarity

$$sim(x, y) = \frac{x \cdot y}{||x|| ||y||}$$

Chapter 3: Data Preprocessing

3.2 Data Cleaning

<1> Missing Values

- Ignore
- Fill in manually
- Fill in automatically with: 1. global constant 2. attribute mean 3. attribute mean for all samples belonging to the same class 4. the most probable value

<2> Noisy Data

- Binning

把数据分成几个一组的，然后在每一组里进行一些操作。

1. smoothing by bin means
2. smoothing by bin medians
3. smoothing by bin boundary: 一个bin里面的最大值和最小值是这个bin的boundary，然后对于这个bin里的所有数值，都用和它最接近的boundary来代替。

举个栗子：

Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15

Bin 2: 21, 21, 24

Bin 3: 25, 28, 34

Smoothing by bin means:

Bin 1: 9, 9, 9

Bin 2: 22, 22, 22

Bin 3: 29, 29, 29

Smoothing by bin boundaries:

Bin 1: 4, 4, 15

Bin 2: 21, 21, 24

Bin 3: 25, 25, 34

- Regression
- Outlier analysis(异常值分析/聚类): 掉到cluster外面的点就是outlier(异常值)

3.3 Data Integration

Data Integration: Combines data from multiple sources into a coherent store.

<1> Entity Identification Problem

从不同的data source里找出对应的entity, 比如有一个source里叫customer_id的和另外一个source里叫cust_number的可能是同一个attribute。

<2> Redundancy and Correlation Analysis

1. χ^2 Correlation Test for Nominal Data

假设有A和B两个attribute, A有c种可能的值 a_1, a_2, \dots, a_c , B有r种可能的值 b_1, b_2, \dots, b_r 。

(A_i, B_j) 表示 $A = a_i, B = b_j$, o_{ij} 是observed frequency也就是观察到了多少个这样取值的个体, e_{ij} 是expected frequency表示按照比例分配这样取值的个体应该有多少个。

$$e_{ij} = \frac{\text{count}(A = a_i) \times \text{count}(B = b_j)}{n}$$

χ^2 Correlation的计算方式:

$$\chi^2 = \sum_{i=1}^c \sum_{j=1}^r \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

举个栗子:

Table 3.1 Example 2.1's 2×2 Contingency Table Data

	male	female	Total
<i>fiction</i>	250 (90)	200 (360)	450
<i>non-fiction</i>	50 (210)	1000 (840)	1050
Total	300	1200	1500

Note: Are gender and preferred_reading correlated?

Using Eq. (3.1) for χ^2 computation, we get

$$\begin{aligned} \chi^2 &= \frac{(250 - 90)^2}{90} + \frac{(50 - 210)^2}{210} + \frac{(200 - 360)^2}{360} + \frac{(1000 - 840)^2}{840} \\ &= 284.44 + 121.90 + 71.11 + 30.48 = 507.93. \end{aligned}$$

括号里的是expected frequency

2. Correlation Coefficient for Numeric Data

$$r_{A,B} = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{n\sigma_A\sigma_B} = \frac{\sum_{i=1}^n (a_i b_i - n\bar{A}\bar{B})}{n\sigma_A\sigma_B}$$

Covariance的计算方法:

$$\begin{aligned} Cov(A, B) &= E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{n} = E(A \cdot B) - \bar{A}\bar{B} \\ \therefore r_{A,B} &= \frac{Cov(A, B)}{\sigma_A\sigma_B} \end{aligned}$$

3.4 Data Reduction

还没完全看懂说实话，知道概念但还不知道怎么用。

快速傅里叶变换（FFT）的缺点是它只能获取一段信号总体上包含哪些频率的成分，但是对各成分出现的时刻并无所知。由时域相差很大的两个信号可能得出一样的频谱图。

小波变换就是把无限长的三角函数基换成了有限长的会衰减的小波基。

<2> Principal Component Analysis

PCA主要用来降低数据的维度。<https://www.matongxue.com/madocs/1025/>, 我觉得这个博客写得很清楚了。

Step 1: The input data are normalized, so that each attribute falls within the same range. This step helps ensure that attributes with large domains will not dominate attributes with smaller domains.

Step 2: PCA computes k orthonormal vectors that provide a basis for the normalized input data. These are unit vectors that provide a basis for the normalized input data. These vectors are referred to as the principal components.

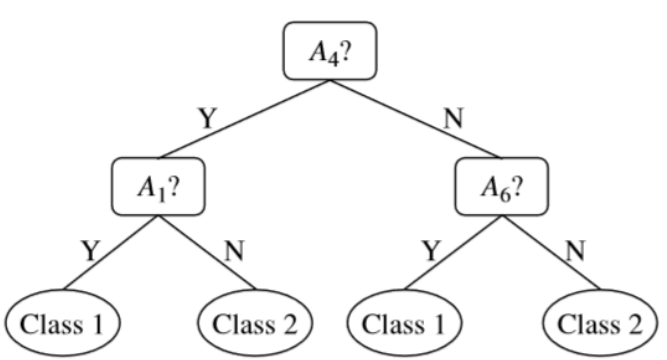
Step 3: The principal components are sorted in order of decreasing "significance" or strength. The principal components essentially serve as a new set of axes for the data. The sorted axes are such that the first axis shows the most variance among the data.

Step 4: Because the components are sorted in decreasing order of "significance", the data size can be reduced by eliminating the weaker components.

<3> Attribute Subset Selection

从原来的attribute里选出一些来。

1. Stepwise forward selection: The procedure starts with an empty set of attributes as the reduced set. The best of the original attributes is determined and added to the reduced set. At each subsequent iteration or step, the best of the remaining original attributes is added to the set.
2. Stepwise backward elimination: The procedure starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set.
3. Combination of forward selection and backward elimination: The stepwise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.
4. Decision tree induction: Decision tree algorithms (e.g., ID3, C4.5, and CART) were originally intended for classification. Decision tree induction constructs a flowchart-like structure where each internal (nonleaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the "best" attribute to partition the data into individual classes. When decision tree induction is used for attribute subset selection, a tree is constructed from the given data. All attributes that do not appear in the tree are assumed to be irrelevant. The set of attributes appearing in the tree form the reduced subset of attributes.

Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ $\Rightarrow \{A_1\}$ $\Rightarrow \{A_1, A_4\}$ \Rightarrow Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ $\Rightarrow \{A_1, A_3, A_4, A_5, A_6\}$ $\Rightarrow \{A_1, A_4, A_5, A_6\}$ \Rightarrow Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  <pre> graph TD A4["A4?"] -- Y --> A1["A1?"] A4 -- N --> A6["A6?"] A1 -- Y --> C1_1((Class 1)) A1 -- N --> C2_1((Class 2)) A6 -- Y --> C1_2((Class 1)) A6 -- N --> C2_2((Class 2)) </pre> \Rightarrow Reduced attribute set: $\{A_1, A_4, A_6\}$

<4> Regression and Log-Linear Models

linear regression/multiple linear regression/log-linear model

<5> Histogram

equal-width/equal-frequency

<6> Clustering

Partition data set into clusters based on similarity, and store cluster representation (比如cluster之后的圆心和半径).

<7> Sampling

1. Simple random sample without replacement (SRSWOR) of size s
2. Simple random sample with replacement (SRSWR) of size s
3. Cluster sample
4. Stratified sample: 原来的数据已经分成了几组，以对应的比例从几组中抽取样本。

<8> Data Cube Aggregation

Use the smallest representation which is enough to solve the task.

3.5 Data Transformation and Data Discretization

1. Smoothing: remove noise from data (binning, regression, clustering)
2. Attribute construction: 从已有的attribute构建新的attribute
3. Aggregation: summary or aggregation operations applied to the data. 比如：销售数据计算每月或每周的金额。
4. Normalization: scaled to fall within a smaller, specified range
5. Discretization

<1> Normalization

- Min-max normalization: 从 $[min, max]$ 到 $[newMin, newMax]$

$$v' = (v - min) \frac{newMax - newMin}{max - min} + newMin$$

- Z-score normalization (μ : mean, σ : standard deviation)

$$v' = \frac{v - \mu}{\sigma}$$

- Normalization by decimal scaling

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

Where j is the smallest integer such that $Max(|v'|) < 1$

<2> Discretization

所谓discretization, 就是把连续的attribute分成不同的区间, 然后用interval label代替原来的数值。

方法一: Binning

Equal-width (distance) partitioning: divide the range into N intervals of equal size

Equal-depth (frequency) partitioning: divide the range into N intervals, each containing approximately the same number of samples

举个栗子:

1. Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

2. 然后分到equal-frequency bins:

Bin 1: 4, 8, 9, 15

Bin 2: 21, 21, 24, 25

Bin 3: 26, 28, 29, 34

3. Smoothing by bin means:

Bin 1: 9, 9, 9, 9

Bin 2: 23, 23, 23, 23

Bin 3: 29, 29, 29, 29

4. Smoothing by bin boundaries:

Bin 1: 4, 4, 4, 15

Bin 2: 21, 21, 25, 25

Bin 3: 26, 26, 34, 34

方法二: Classification & Correlation Analysis

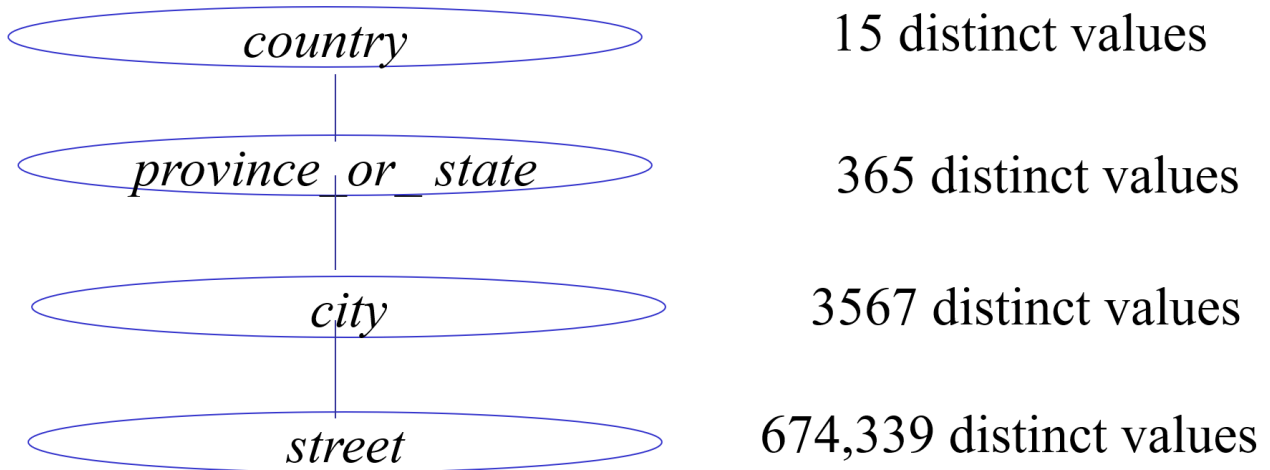
Classification: decision tree analysis/Correlation Analysis: X^2 -based discretization

<3> Concept Hierarchy Generation for Nominal Data

Concept hierarchy organizes concepts (i.e., attribute values) hierarchically and is usually associated with each dimension in a data warehouse.

例子：街道<城市<省份<国家

Automatic Concept Hierarchy Generation: 值最少的attribute被认为是等级最高的，比如：



Chapter 4: Data Warehousing and Online Analytical Processing

4.1 Data Warehouse: Basic Concepts

4.1.1 What is a Data Warehouse

Data Warehouse也就是数据仓库，顾名思义就是一个很大的数据储存集合，它的创建是为了企业的分析性报告和决策支持目的。

Key features of DW:

1. Subject-oriented(**主题性**): 比如对于滴滴出行，“司机行为分析”就是一个主题，而DW的所有数据都围绕某一主题来组织。
2. Integrated(**集成性**): DW中存储的数据来源于多个数据源的集成，需要整合成为最终的数据集合。
3. Time-variant(**时变性**): DW会定期接收新的继承数据。
4. Nonvolatile(**稳定性**): DW中的数据不允许被修改，用户只能查询和分析

4.1.2 Operational Database和Data Warehouse的区别：

Table 4.1 Comparison of OLTP and OLAP Systems

<i>Feature</i>	<i>OLTP</i>	<i>OLAP</i>
Characteristic	operational processing	informational processing
Orientation	transaction	analysis
User	clerk, DBA, database professional	knowledge worker (e.g., manager, executive, analyst)
Function	day-to-day operations	long-term informational requirements decision support
DB design	ER-based, application-oriented	star/snowflake, subject-oriented
Data	current, guaranteed up-to-date	historic, accuracy maintained over time
Summarization	primitive, highly detailed	summarized, consolidated
View	detailed, flat relational	summarized, multidimensional
Unit of work	short, simple transaction	complex query
Access	read/write	mostly read
Focus	data in	information out
Operations	index/hash on primary key	lots of scans
Number of records accessed	tens	millions
Number of users	thousands	hundreds
DB size	GB to high-order GB	\geq TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

4.1.3 Reasons for having a separate Data Warehouse:

原因一： High performance for both systems

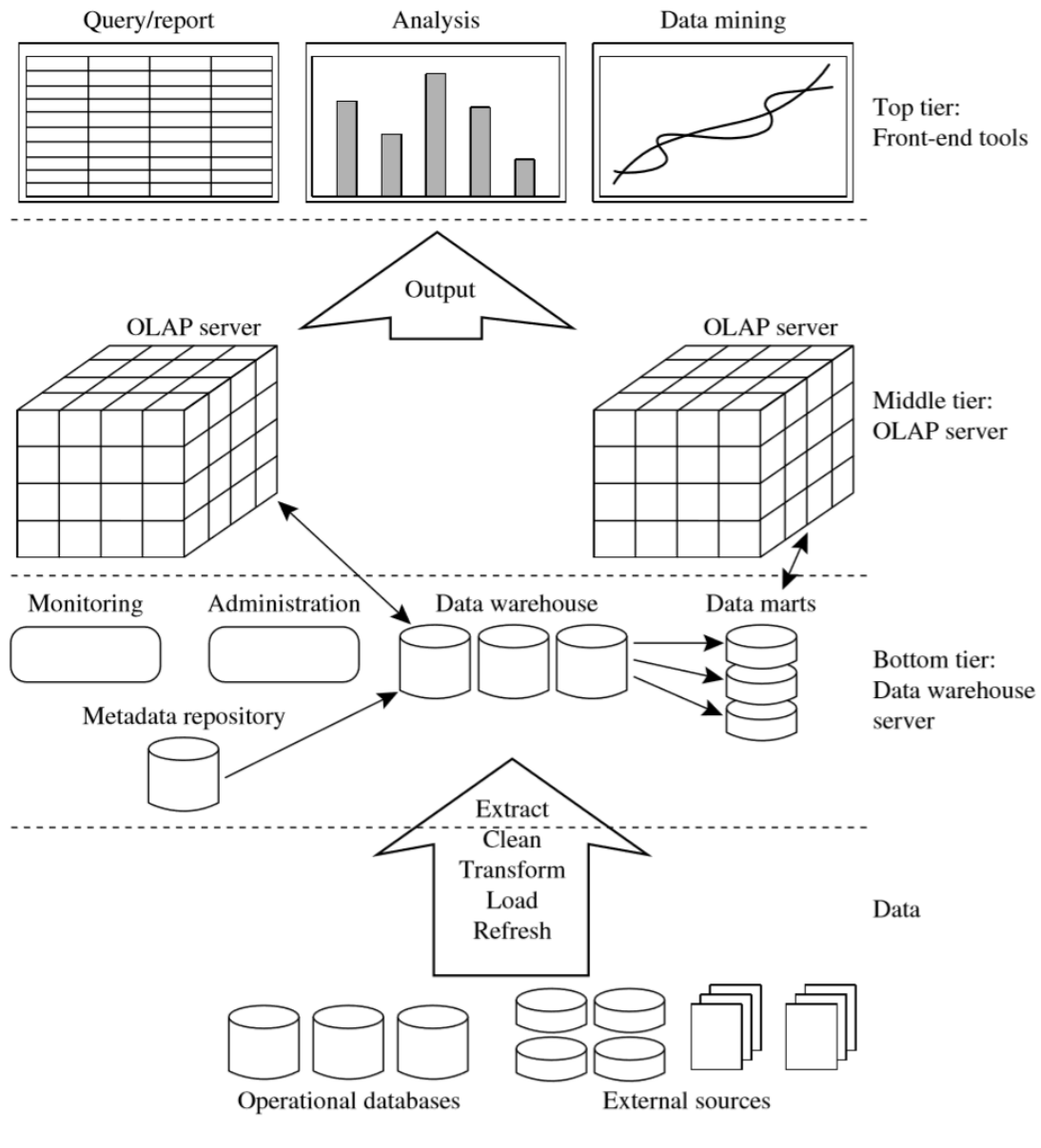
原因二： Different functions and different data:

Decision support requires historical data which operational DBs do not typically maintain;

DS requires consolidation (aggregation, summarization) of data from heterogeneous sources;

different sources typically use inconsistent data representations, codes and formats which have to be reconciled

4.1.4 数据仓库的多层结构:



第一层：The bottom tier is a warehouse database server that is almost always a relational database system.

第二层：The middle tier is an OLAP server

第三层：The top tier is a front-end client layer, which contains query and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on).

4.1.5 三种数据仓库的模型

<1> Enterprise warehouse: collects all of the information about subjects spanning the entire organization.

<2> Data Mart: a subset of corporate-wide data that is of value to a specific groups of users.

<3> Virtual warehouse: a set of views over operational databases.

4.1.6 Extraction, Transformation, and Loading (ETL)

<1> Data extraction, which typically gathers data from multiple, heterogeneous, and external sources.

<2> Data cleaning, which detects errors in the data and rectifies them when possible.

<3> Data transformation, which converts data from legacy or host format to warehouse format.

<4> Load, which sorts, summarizes, consolidates, computes views, checks integrity, and builds indices and partitions.

<5> Refresh, which propagates the updates from the data sources to the warehouse.

4.1.7 Metadata Repository

Meta data is the data defining warehouse objects.

- Description of the structure of the data warehouse: schema, view, dimensions, hierarchies, derived data definitions, data mart locations and contents.
- Operational metadata, which include data lineage (history of migrated data and the sequence of transformations applied to it), currency of data (active, archived, or purged), and monitoring information (warehouse usage statistics, error reports, and audit trails).
- The algorithms used for summarization, which include measure and dimension definition algorithms, data on granularity, partitions, subject areas, aggregation, summarization, and predefined queries and reports.
- Mapping from the operational environment to the data warehouse, which includes source databases and their contents, gateway descriptions, data partitions, data extraction, cleaning, transformation rules and defaults, data refresh and purging rules, and security (user authorization and access control).
- Data related to system performance, which include indices and profiles that improve data access and retrieval performance, in addition to rules for the timing and scheduling of refresh, update, and replication cycles.
- Business metadata, which include business terms and definitions, data ownership information, and charging policies.

4.2 Data Warehouse Modeling: Data Cube and OLAP

Chapter 6: Mining Frequent Patterns, Associations, and Correlations: Basic Concepts and Methods

6.1 Basic Concepts

<1> Frequent itemsets, closed itemsets and association rules

Let $I = \{I_1, I_2, \dots, I_m\}$ be an itemset. Let D , the task-relevant data, be a set of database transactions where each transaction T is a nonempty itemset such that $T \subset I$. Let A and B be set of items.

- support: percentage of transactions in D that contain $A \cup B$
- confidence: percentage of transactions in D containing A that also contains B

$$\text{support}(A \Rightarrow B) = P(A \cup B)$$
$$\text{confidence}(A \Rightarrow B) = P(B|A)$$

上式中的support是relative support.

An itemset that contains k items is a k-itemset.

absolute support = occurrence frequency = frequency = support count = count = number of transactions that contain the itemset.

If the relative support of an itemset I satisfies a prespecified minimum support threshold, then I is a frequent itemset.

- Closed itemset: there exists no proper super-itemset Y such that Y has the same support count as X in D不存在出现次数比它更多的super-itemset
- closed frequent itemset: 既closed又frequent
- max-itemset (maximal frequent itemset): 本身要frequent, 而且不能有frequent super-itemset

举例: 假如database里只有两个transaction, $\{ \langle a_1, a_2, \dots, a_{100} \rangle; \langle a_1, a_2, \dots, a_{50} \rangle \}$, 如果minimum support count是1, 那么有两个closed frequent itemset: $C = \{ \{a_1, a_2, \dots, a_{100}\} : 1; \{a_1, a_2, \dots, a_{50}\} : 2 \}$

只有一个maximal frequent itemset: $M = \{ \{a_1, a_2, \dots, a_{100}\} : 1 \}$, 而 $\{a_1, a_2, \dots, a_{50}\}$ 不是max-itemset, 因为前者是它的frequent super-itemset.

6.2 Frequent Itemset Mining Methods

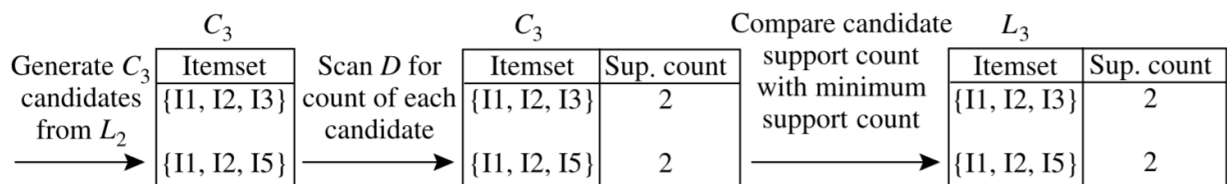
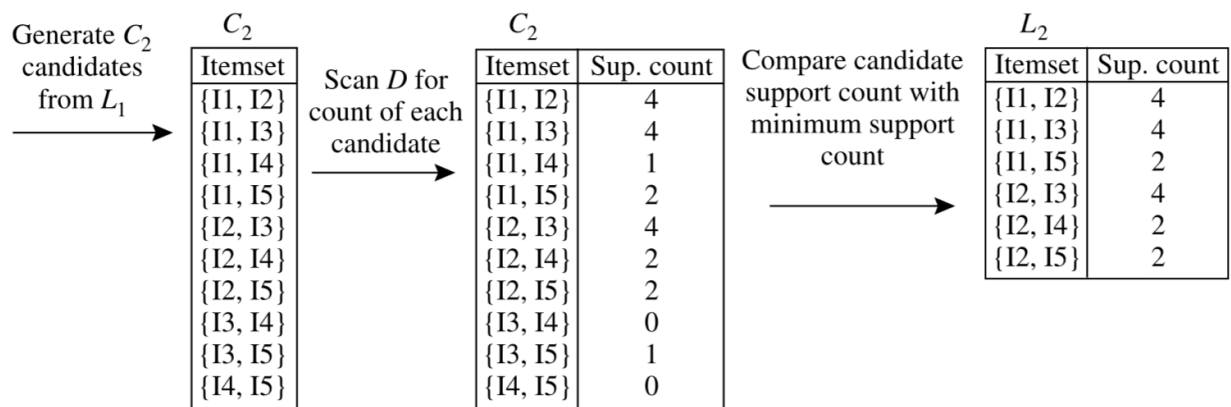
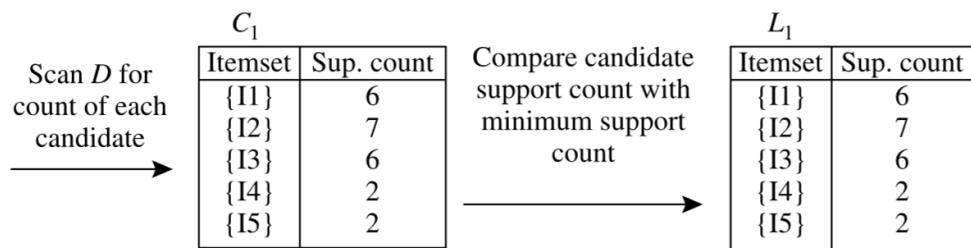
设计算法去找frequent itemset

- The downward closure property of frequent patterns: Any subset of a frequent itemset must be frequent.

6.2.1 Apriori Algorithm

1. Join step
2. Prune step

举个栗子:



每一次的iteration, 先生成 C_n candidate, 然后找“candidate中满足所有子集都在 L_{n-1} 当中的”, 然后筛选掉小于minimum support count的。

比如产生 C_3 candidate的时候, 是这样的:

- (a) Join: $C_3 = L_2 \bowtie L_2 = \{\{I1, I2\}, \{I1, I3\}, \{I1, I5\}, \{I2, I3\}, \{I2, I4\}, \{I2, I5\}\} \bowtie \{\{I1, I2\}, \{I1, I3\}, \{I1, I5\}, \{I2, I3\}, \{I2, I4\}, \{I2, I5\}\} = \{\{I1, I2, I3\}, \{I1, I2, I5\}, \{I1, I3, I5\}, \{I2, I3, I4\}, \{I2, I3, I5\}, \{I2, I4, I5\}\}.$
- (b) Prune using the Apriori property: All nonempty subsets of a frequent itemset must also be frequent. Do any of the candidates have a subset that is not frequent?

- The 2-item subsets of $\{I1, I2, I3\}$ are $\{I1, I2\}$, $\{I1, I3\}$, and $\{I2, I3\}$. All 2-item subsets of $\{I1, I2, I3\}$ are members of L_2 . Therefore, keep $\{I1, I2, I3\}$ in C_3 .
- The 2-item subsets of $\{I1, I2, I5\}$ are $\{I1, I2\}$, $\{I1, I5\}$, and $\{I2, I5\}$. All 2-item subsets of $\{I1, I2, I5\}$ are members of L_2 . Therefore, keep $\{I1, I2, I5\}$ in C_3 .
- The 2-item subsets of $\{I1, I3, I5\}$ are $\{I1, I3\}$, $\{I1, I5\}$, and $\{I3, I5\}$. $\{I3, I5\}$ is not a member of L_2 , and so it is not frequent. Therefore, remove $\{I1, I3, I5\}$ from C_3 .
- The 2-item subsets of $\{I2, I3, I4\}$ are $\{I2, I3\}$, $\{I2, I4\}$, and $\{I3, I4\}$. $\{I3, I4\}$ is not a member of L_2 , and so it is not frequent. Therefore, remove $\{I2, I3, I4\}$ from C_3 .
- The 2-item subsets of $\{I2, I3, I5\}$ are $\{I2, I3\}$, $\{I2, I5\}$, and $\{I3, I5\}$. $\{I3, I5\}$ is not a member of L_2 , and so it is not frequent. Therefore, remove $\{I2, I3, I5\}$ from C_3 .
- The 2-item subsets of $\{I2, I4, I5\}$ are $\{I2, I4\}$, $\{I2, I5\}$, and $\{I4, I5\}$. $\{I4, I5\}$ is not a member of L_2 , and so it is not frequent. Therefore, remove $\{I2, I4, I5\}$ from C_3 .

(c) Therefore, $C_3 = \{\{I1, I2, I3\}, \{I1, I2, I5\}\}$ after pruning.

- 在产生 C_k candidate的join的过程中，要两个 C_{k-1} itemset的前k-2个item都一样，否则是没有join操作的。比如这里 $\{I_1, I_2\}$ 和 $\{I_2, I_4\}$ 的“前k-2 = 1”个也就是第一个item不一样，所以不join。我也是看了半天发现怎么没有 $\{I_1, I_2, I_4\}$ ，原来之前的理解一直不对。。。

6.2.3 Improving the Efficiency of Apriori

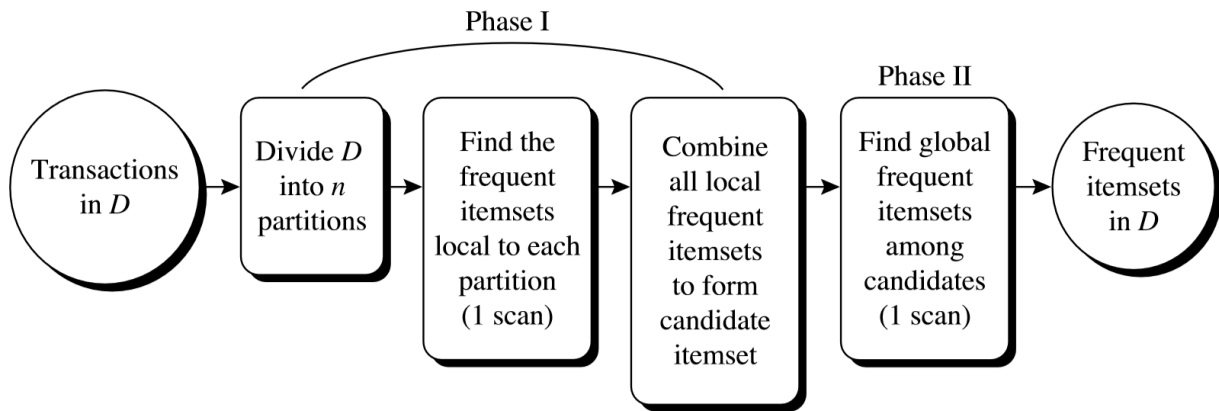
<1> Hash-based technique

把这些itemset都hash到一个hash table里面，然后这个bucket里面itemset总数小于support threshold的话就排除掉了（因为一个bucket里面可能有多种itemset，加在一起还小于support threshold那就可以排除了）

<2> Transaction reduction

A transaction that does not contain any frequent k-itemsets cannot contain any frequent (k+1)-itemsets.

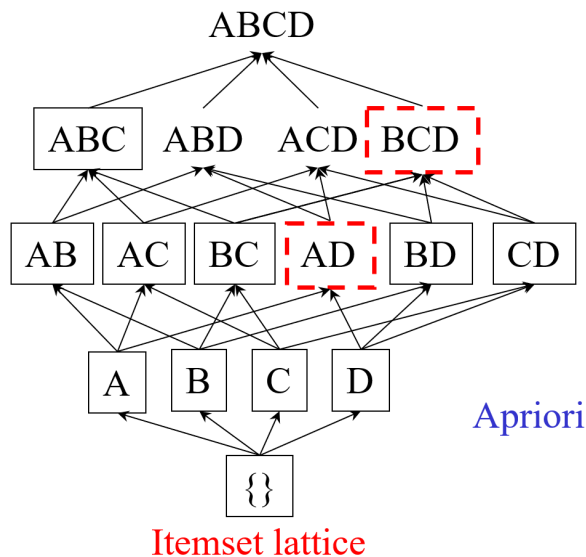
<3> Partitioning



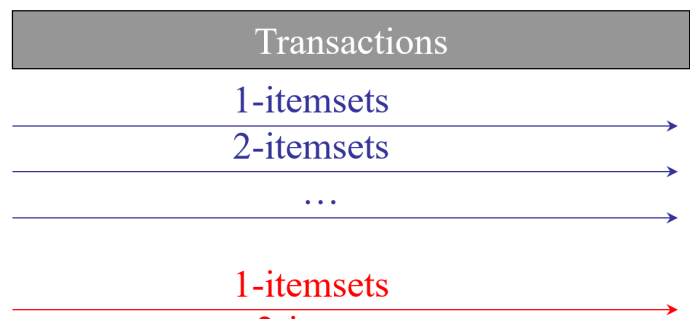
<4> Sampling

Pick a random sample S of the given data D , and then search for frequent itemsets in S instead of D .

<5> Dynamic itemset counting

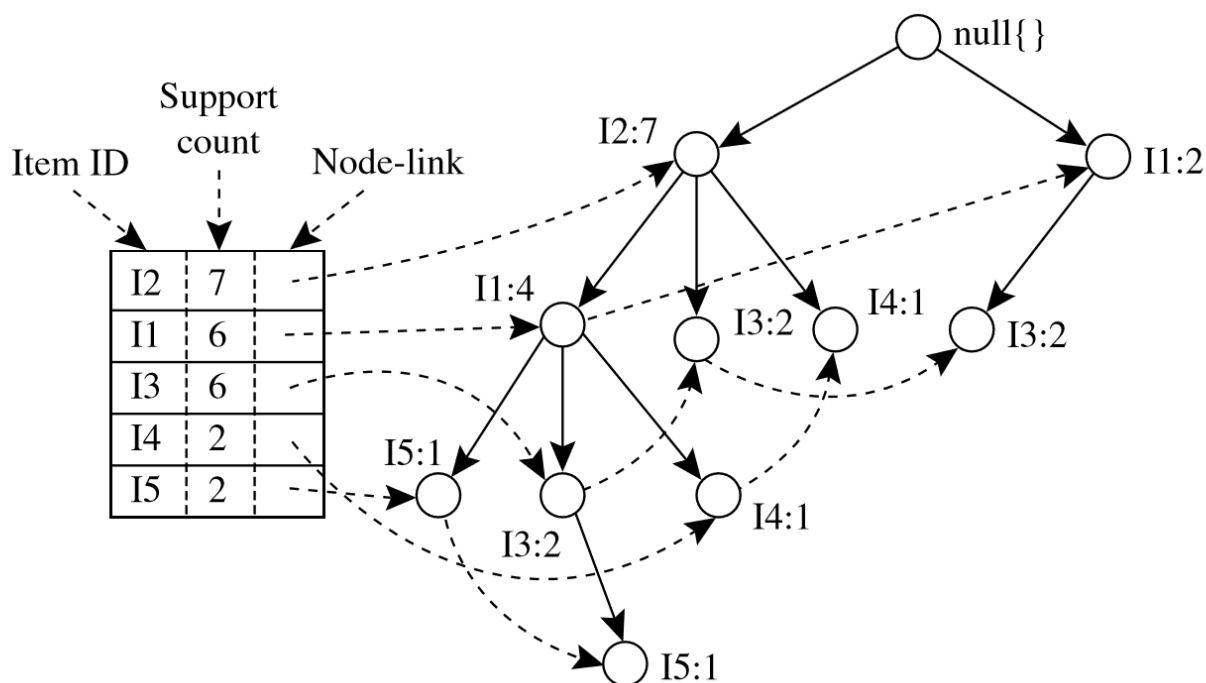


- Once both A and D are determined frequent, the counting of AD begins
- Once all length-2 subsets of BCD are determined frequent, the counting of BCD begins



6.2.4 A Pattern-Growth Approach for Mining Frequent Itemsets

第一步先画出FP-Tree:



按照item出现次数从大到小排列。首先创造根节点null，然后对每个transaction里的item根据support count从大到小排序，构造FP-tree的一个分支，后面的transaction依此类推。

第二步，根据FP-tree找frequent pattern：

Item	Conditional Pattern Base	Conditional FP-tree	Frequent Patterns Generated
I5	{{I2, I1: 1}, {I2, I1, I3: 1}}	$\langle I2: 2, I1: 2 \rangle$	{I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}
I4	{{I2, I1: 1}, {I2: 1}}	$\langle I2: 2 \rangle$	{I2, I4: 2}
I3	{{I2, I1: 2}, {I2: 2}, {I1: 2}}	$\langle I2: 4, I1: 2 \rangle, \langle I1: 2 \rangle$	{I2, I3: 4}, {I1, I3: 4}, {I2, I1, I3: 2}
I1	{{I2: 4}}	$\langle I2: 4 \rangle$	{I2, I1: 4}

比如对于 I_5 ， $\{I_2, I_1\}$ 和 $\{I_2, I_1, I_3\}$ 都出现了一次（这里不把 I_5 本身算在里面），于是 I_2 累计出现2次， I_1 也是累计出现两次，再和 I_5 组合起来就有三个frequent pattern。

6.2.5 Mining Frequent Itemsets Using the Vertical Data Format

把这样的horizontal format：

Table 6.1 Transactional Data for an *AllElectronics* Branch

<i>TID</i>	<i>List of item_IDs</i>
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

换成这样的vertical format:

Table 6.3 The Vertical Data Format of the Transaction Data Set *D* of Table 6.1

<i>itemset</i>	<i>TID_set</i>
I1	{T100, T400, T500, T700, T800, T900}
I2	{T100, T200, T300, T400, T600, T800, T900}
I3	{T300, T500, T600, T700, T800, T900}
I4	{T200, T400}
I5	{T100, T800}

然后做集合之间的intersection就行了:

Table 6.4 2-Itemsets in Vertical Data Format

<i>itemset</i>	<i>TID_set</i>
{I1, I2}	{T100, T400, T800, T900}
{I1, I3}	{T500, T700, T800, T900}
{I1, I4}	{T400}
{I1, I5}	{T100, T800}
{I2, I3}	{T300, T600, T800, T900}
{I2, I4}	{T200, T400}
{I2, I5}	{T100, T800}
{I3, I5}	{T800}

Table 6.5 3-Itemsets in Vertical Data Format

<i>itemset</i>	<i>TID_set</i>
{I1, I2, I3}	{T800, T900}
{I1, I2, I5}	{T100, T800}

frequent in Table 6.3, there are 10 intersections performed in total, which lead to eight nonempty 2-itemsets, as shown in Table 6.4. Notice that because the itemsets {I1, I4} and {I3, I5} each contain only one transaction, they do not belong to the set of frequent 2-itemsets.

Based on the Apriori property, a given 3-itemset is a candidate 3-itemset only if every one of its 2-itemset subsets is frequent. The candidate generation process here will generate only two 3-itemsets: {I1, I2, I3} and {I1, I2, I5}. By intersecting the TID_sets of any two corresponding 2-itemsets of these candidate 3-itemsets, it derives Table 6.5, where there are only two frequent 3-itemsets: {I1, I2, I3: 2} and {I1, I2, I5: 2}. ■

6.2.6 Mining Closed and Max Patterns

<1> Item Merging

If every transaction containing a frequent itemset X also contains an itemset Y but not any proper superset of Y , then $X \cup Y$ forms a frequent closed itemset and there is no need to search for any itemset containing X but no Y .

<2> Sub-itemset pruning

If a frequent itemset X is a proper subset of an already found frequent closed itemset Y and support count(X)=support count(Y), then X and all of X 's descendants in the set enumeration tree cannot be frequent closed itemsets and thus can be pruned.

<3> Item skipping

In the depth-first mining of closed itemsets, at each level, there will be a prefix itemset X associated with a header table and a projected database. If a local frequent item p has the same support in several header tables at different levels, we can safely prune p from the header tables at higher levels.

6.3 Which Patterns Are Interesting?—Pattern Evaluation Methods

Strong rules are not necessarily interesting.

Lift(提升度):

$$lift(A, B) = \frac{P(A \cup B)}{P(A)P(B)}$$

如果A和B是相互独立的则lift = 1.

If the resulting value of lift is less than 1, then the occurrence of A is negatively correlated with the occurrence of B, meaning that the occurrence of one likely leads to the absence of the other one. If the resulting value is greater than 1, then A and B are positively correlated, meaning that the occurrence of one implies the occurrence of the other. If the resulting value is equal to 1, then A and B are independent and there is no correlation between them.

要考试了先不写了，未完待续。。。。