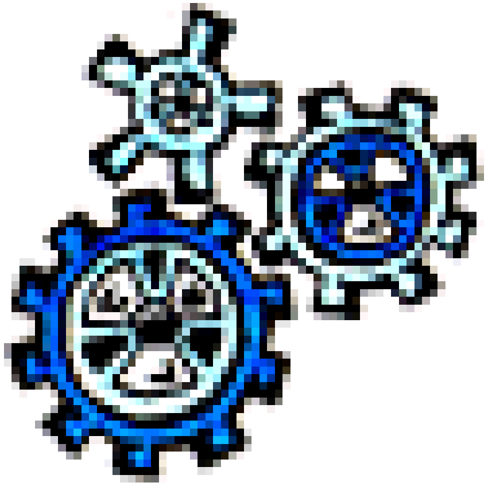
Association rules and market basket analysis



Association rules - module outline

- **What are association rules (AR) and what are they used for:**
  - The paradigmatic application: Market Basket Analysis
  - The single dimensional AR (intra-attribute)
- **How to compute AR**
  - Basic Apriori Algorithm and its optimizations
  - Multi-Dimension AR (inter-attribute)
  - Quantitative AR

Market Basket Analysis: the context

Customer buying habits by finding associations and correlations between the different items that customers place in their “shopping basket”

Milk, eggs, sugar, bread

Customer1

Milk, eggs, cereal, bread

Customer2

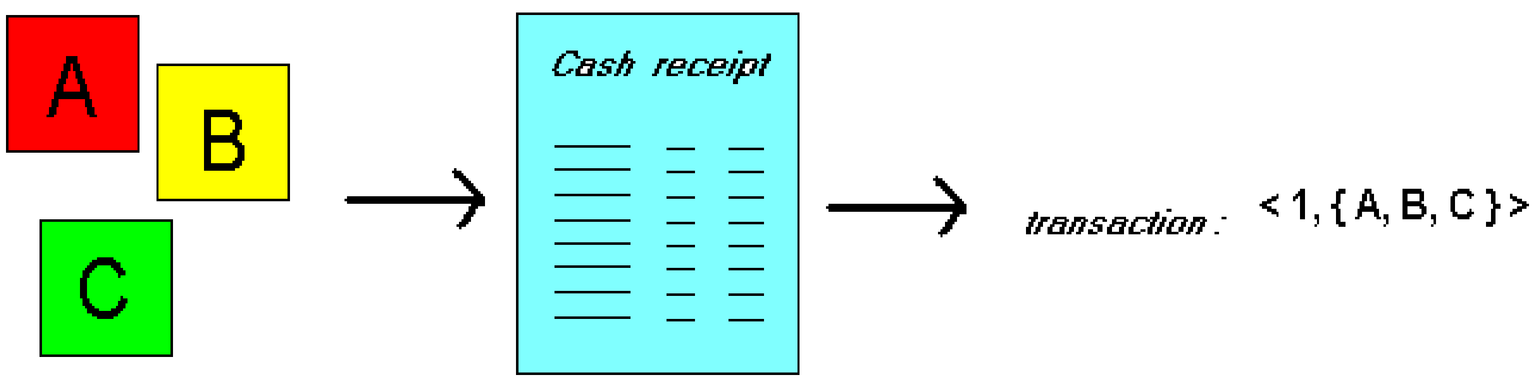
Eggs, sugar

Customer3

Market Basket Analysis: the context

Given: a database of customer **transactions**, where each transaction is a **set of items**

- Find groups of items which are **frequently purchased together**



Goal of MBA

- Extract information on purchasing behavior
- Actionable information: can suggest
  - new store layouts
  - new product assortments
  - which products to put on promotion
- MBA applicable whenever a customer purchases multiple things in proximity
  - credit cards
  - services of telecommunication companies
  - banking services
  - medical treatments

MBA: applicable to many other contexts

Telecommunication:

Each customer is a transaction containing the set of customer's phone calls

Atmospheric phenomena:

Each time interval (e.g. a day) is a transaction containing the set of observed event (rains, wind, etc.)

Etc.

Association Rules

- Express how product/services relate to each other, and tend to group together
- "if a customer purchases three-way calling, then will also purchase call-waiting"
- simple to understand
- actionable information: bundle three-way calling and call-waiting in a single package
- Examples.
  - Rule form: "Body → Head [support, confidence]".
  - buys(x, "diapers") → buys(x, "beers") [0.5%, 60%]
  - major(x, "CS") and takes(x, "DB") → grade(x, "A") [1%, 75%]

Useful, trivial, unexplicable

- Useful: "On Thursdays, grocery store consumers often purchase diapers and beer together".
- Trivial: "Customers who purchase maintenance agreements are very likely to purchase large appliances".
- Unexplicable: "When a new hardaware store opens, one of the most sold items is toilet rings."

Association Rules Road Map

- **Single dimension vs. multiple dimensional AR**
  - E.g., association on items bought vs. linking on different attributes.
  - Intra-Attribute vs. Inter-Attribute
- **Qualitative vs. quantitative AR**
  - Association on categorical vs. numerical attributes
- **Simple vs. constraint-based AR**
  - E.g., small sales (sum < 100) trigger big buys (sum > 1,000)?
- **Single level vs. multiple-level AR**
  - E.g., what brands of beers are associated with what brands of diapers?
- **Association vs. correlation analysis.**
  - Association does not necessarily imply correlation.

Association Rule Mining

- Given a set of transactions, find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction

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

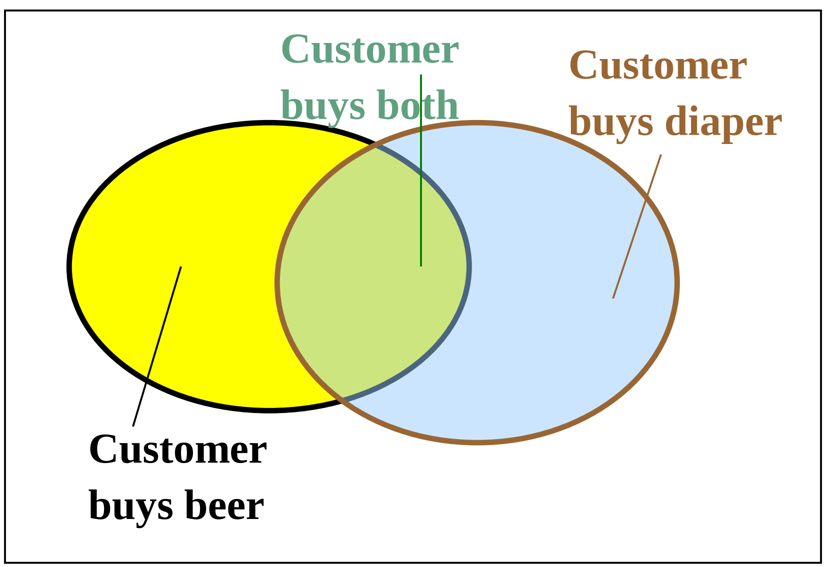
Example of Association Rules

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

Implication means co-occurrence, not causality!

Basic Concepts: Frequent Patterns and Association Rules

Transaction-id	Items bought
10	A, B, D
20	A, C, D
30	A, D, E
40	B, E, F
50	B, C, D, E, F



- Itemset X = $\{x_1, \ldots, x_k\}$
- Find all the rules $X \rightarrow Y$ with minimum support and confidence
  - support, *s*, probability that a transaction contains $X \cup Y$
  - confidence, *c*, conditional probability that a transaction having X also contains *Y*

Let $sup_{min} = 50\%$, $conf_{min} = 50\%$

Freq. Pat.: {A:3, B:3, D:4, E:3, AD:3}

Association rules:

$A \rightarrow D$ (60%, 100%)

$D \rightarrow A$ (60%, 75%)

Definition: Frequent Itemset

- Itemset
  - **A collection of one or more items**
    - ✓ Example: {Milk, Bread, Diaper}
  - **k-itemset**
    - ✓ An itemset that contains k items
- Support count (σ)
  - **Frequency of occurrence of an itemset**
  - **E.g. σ({Milk, Bread,Diaper}) = 2**
- Support
  - **Fraction of transactions that contain an itemset**
  - **E.g. s({Milk, Bread, Diaper}) = 2/5**
- Frequent Itemset
  - **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

Definition: Association Rule

- Association Rule
  - **An implication expression of the form X → Y, where X and Y are itemsets**
  - **Example:**
    {Milk, Diaper} → {Beer}

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

- Rule Evaluation Metrics
  - **Support (s)**
    - ✓ Fraction of transactions that contain both X and Y
  - **Confidence (c)**
    - ✓ Measures how often items in Y appear in transactions that contain X

Example:

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

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

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

Association rules - module outline

- **What are association rules (AR) and what are they used for:**
  - The paradigmatic application: Market Basket Analysis
  - The single dimensional AR (intra-attribute)
- **How to compute AR**
  - Basic Apriori Algorithm and its optimizations
  - Multi-Dimension AR (inter-attribute)
  - Quantitative AR

Association Rule Mining Task

- **Given a set of transactions T, the goal of association rule mining is to find all rules having**
  - support ≥ *minsup* threshold
  - confidence ≥ *minconf* threshold

- **Brute-force approach:**
  - List all possible association rules
  - Compute the support and confidence for each rule
  - Prune rules that fail the *minsup* and *minconf* thresholds

  ⇒ Computationally prohibitive!

Mining Association Rules

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} → {Beer} (s=0.4, c=0.67)
{Milk,Beer} → {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} → {Milk} (s=0.4, c=0.67)
{Beer} → {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} → {Milk,Beer} (s=0.4, c=0.5)
{Milk} → {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

Mining Association Rules

- **Two-step approach:**
  1. **Frequent Itemset Generation**
     – Generate all itemsets whose support $\geq$ minsup
  2. **Rule Generation**
     – 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**

Basic Apriori Algorithm

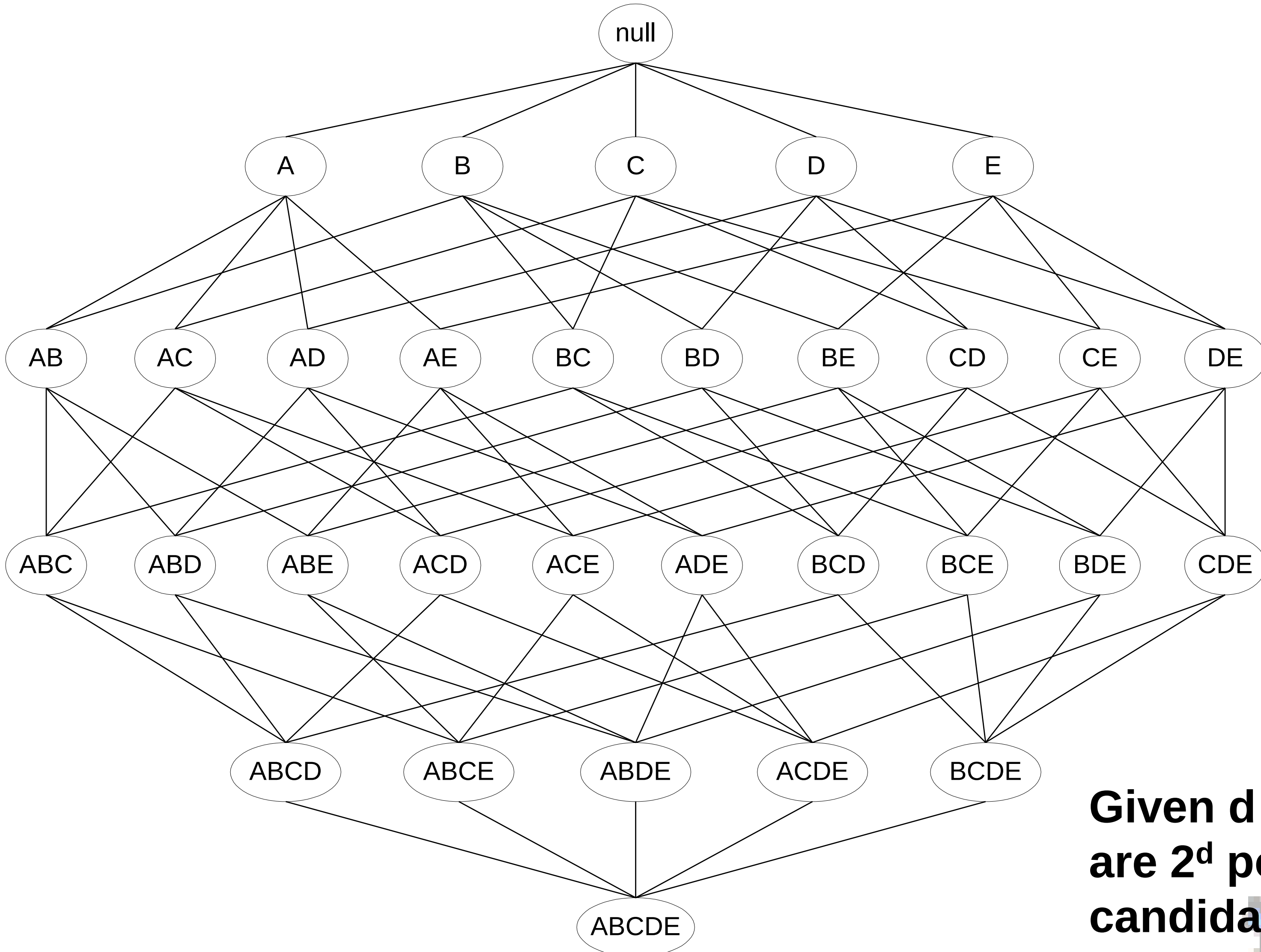
Problem Decomposition

① **Find the *frequent itemsets*: the sets of items that satisfy the support constraint**

- *A subset of a frequent itemset is also a frequent itemset*, i.e., if {*A*,*B*} is a frequent itemset, both {*A*} and {*B*} should be a frequent itemset
- Iteratively find frequent itemsets with cardinality from 1 to *k* (*k*-itemset)

② **Use the frequent itemsets to generate association rules.**

Frequent Itemset Generation



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

Frequent Itemset Generation

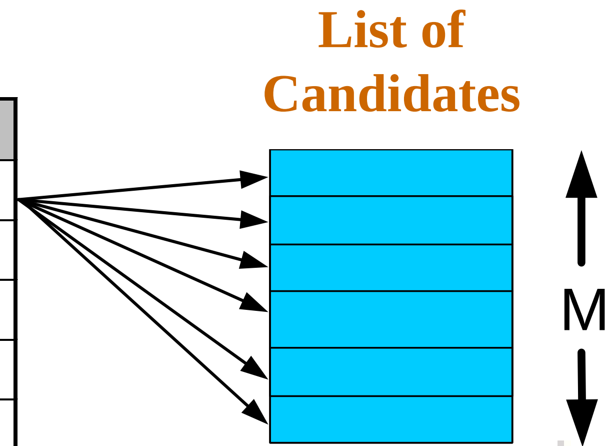
- **Brute-force approach:**
  - Each itemset in the lattice is a candidate frequent itemset
  - Count the support of each candidate by scanning the database

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

N

w



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

Frequent Itemset Generation Strategies

- **Reduce the number of candidates (M)**
  - Complete search: $M=2^d$
  - Use pruning techniques to reduce M
- **Reduce the number of transactions (N)**
  - Reduce size of N as the size of itemset increases
  - Used by DHP and vertical-based mining algorithms
- **Reduce the number of comparisons (NM)**
  - Use efficient data structures to store the candidates or transactions
  - No need to match every candidate against every transaction

Reducing Number of Candidates

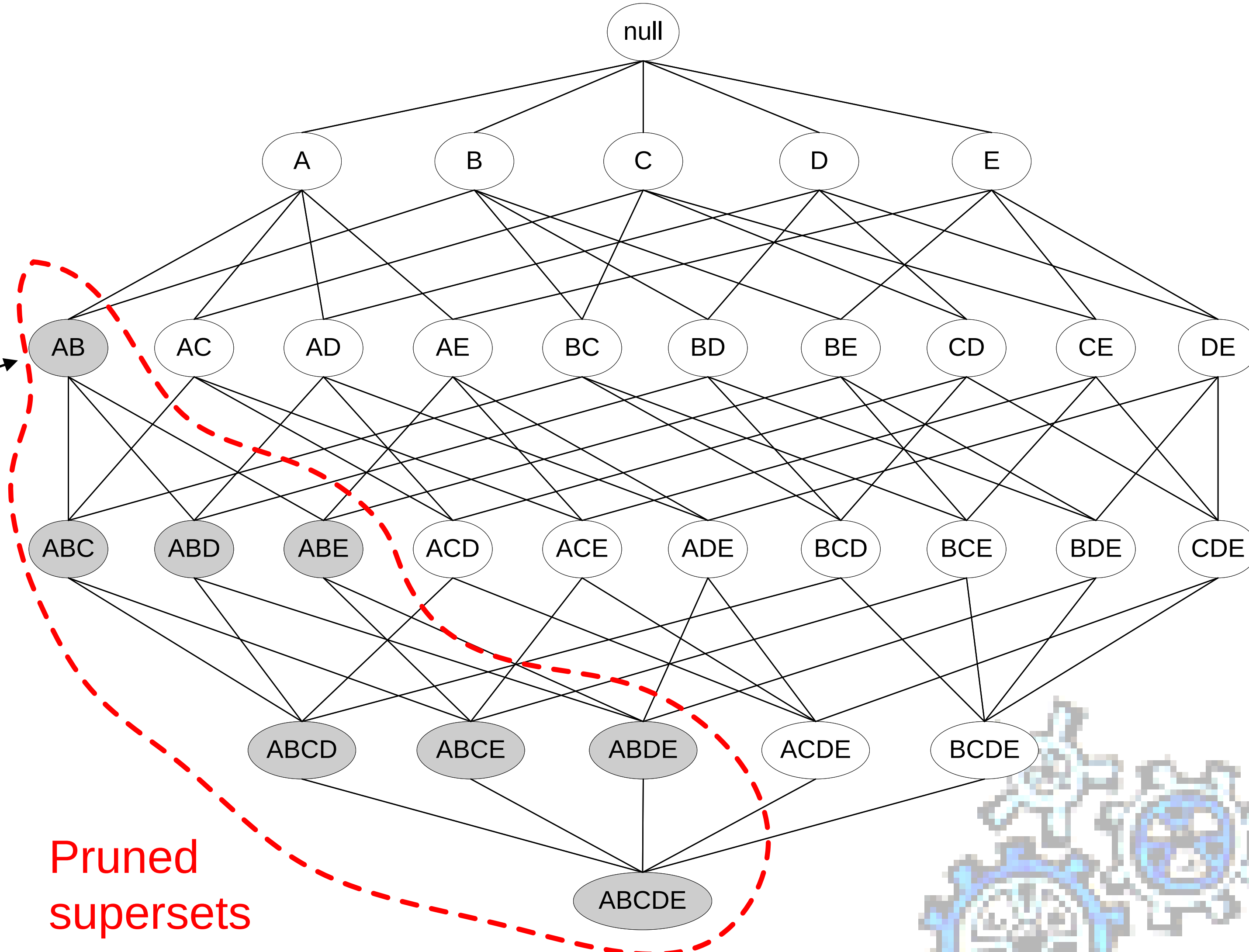
- **Apriori principle**:
  - If an itemset is frequent, then all of its subsets must also be frequent

- **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 property of support

Illustrating Apriori Principle



Found to be Infrequent

Pruned supersets

Apriori Execution Example (min_sup = 2)

Database TDB

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

Scan TDB →

C_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

→

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

← Scan TDB

C_2

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

←

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_3

itemset
{2 3 5}

Scan TDB →

L_3

itemset	sup
{2 3 5}	2

The Apriori Algorithm

- **Join Step**: **C_k is generated by joining L_{k-1} with itself**
- **Prune Step**: **Any (k-1)-itemset that is not frequent cannot be a subset of a frequent k-itemset**
- **Pseudo-code**:

C_k: Candidate itemset of size k
L_k : frequent itemset of size k

```
L_1 = {frequent items};
for (k = 1; L_k !=∅; k++) do begin
    C_k+1 = candidates generated from L_k;
    for each transaction t in database do
        increment the count of all candidates in C_k+1
        that are contained in t
    L_k+1 = candidates in C_k+1 with min_support
    end
return ∪_k L_k;
```

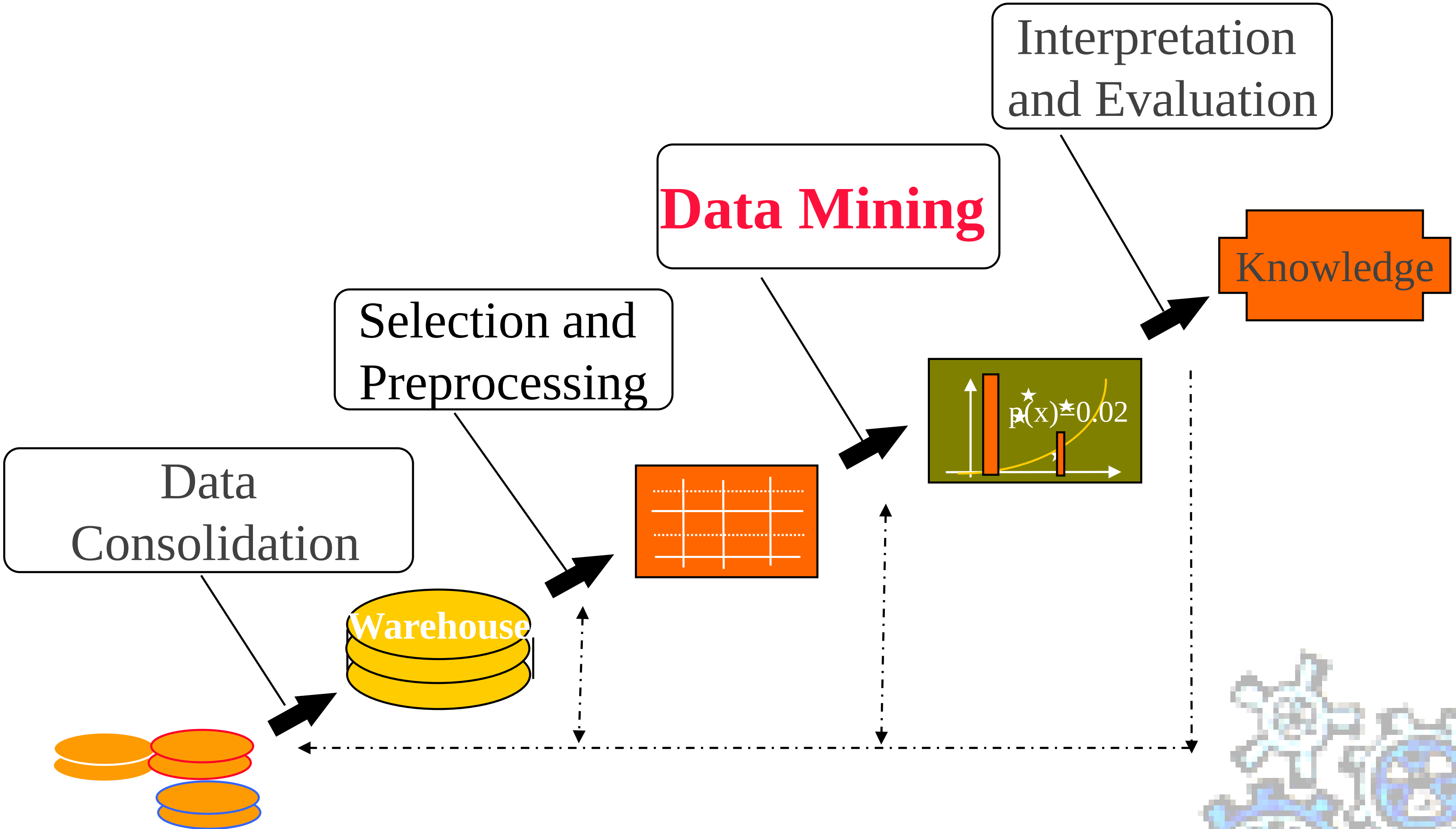
Example of Generating Candidates

- L_3={*abc, abd, acd, ace, bcd*}
- Self-joining: L_3*L_3
  - *abcd* from *abc* and *abd*
  - *acde* from *acd* and *ace*
- Pruning:
  - *acde* is removed because *ade* is not in L_3
- C_4={*abcd*}

Factors Affecting Complexity

- Choice of minimum support threshold
  - lowering support threshold results in more frequent itemsets
  - this may increase number of candidates and max length of frequent itemsets
- Dimensionality (number of items) of the data set
  - more space is needed to store support count of each item
  - if number of frequent items also increases, both computation and I/O costs may also increase
- Size of database
  - since Apriori makes multiple passes, run time of algorithm may increase with number of transactions
- Average transaction width
  - transaction width increases with denser data sets
  - This may increase max length of frequent itemsets and traversals of hash tree (number of subsets in a transaction increases with its width)

The KDD process



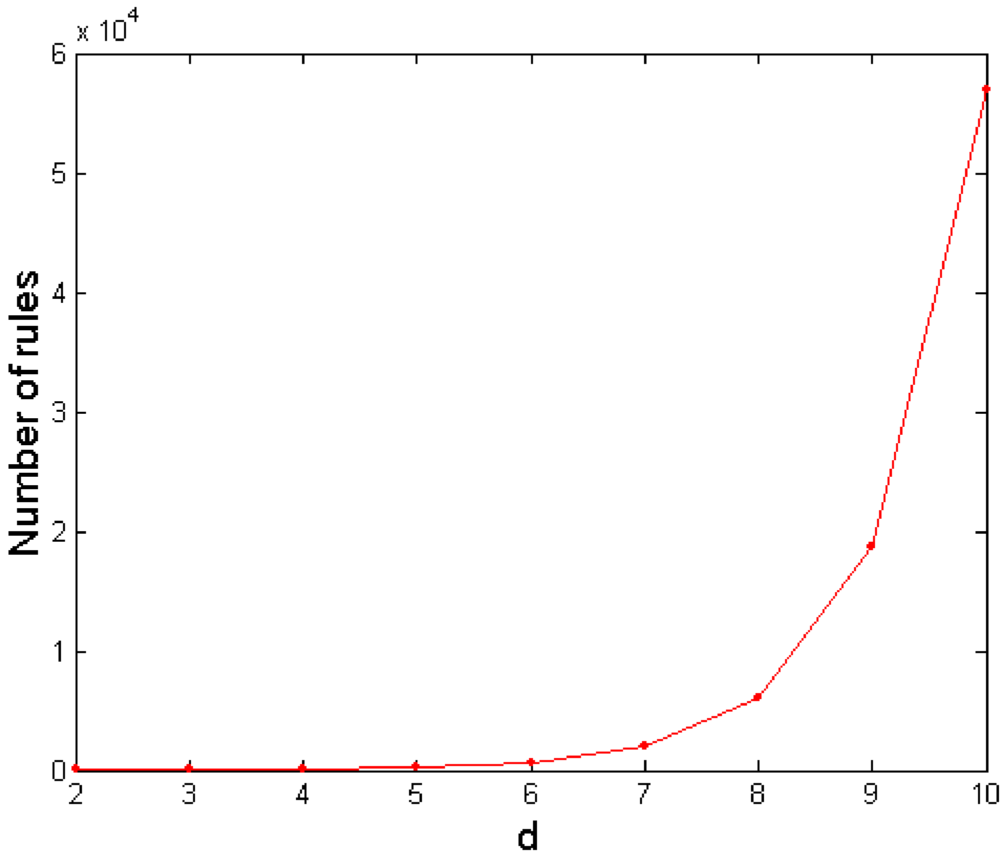
Generating Association Rules from Frequent Itemsets

- **Only strong association rules are generated**
- **Frequent itemsets satisfy minimum support threshold**
- **Strong rules are those that satisfy minimum confidence threshold**
- $\textit{confidence}(A \Longrightarrow B) = \Pr(B \mid A) = \boxed{\dfrac{support(A \cup B)}{support(A)}}$

For each frequent itemset, **f**, generate all non-empty subsets of **f**
For every non-empty subset **s** of **f** **do**
    if support(**f**)/support(**s**) ≥ min_confidence **then**
        output rule **s** ==> **(f-s)**
end

Computational Complexity

- 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

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 $\varnothing \rightarrow L$)

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}:

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

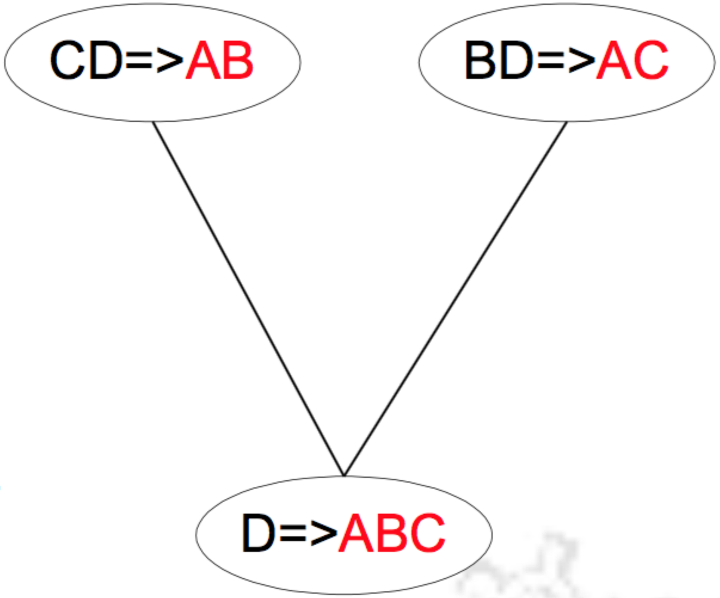
✓ Confidence is anti-monotone w.r.t. number of items on the RHS of the rule

Rule Generation for Apriori Algorithm

- Candidate rule is generated by merging two rules that share the same prefix in the rule consequent

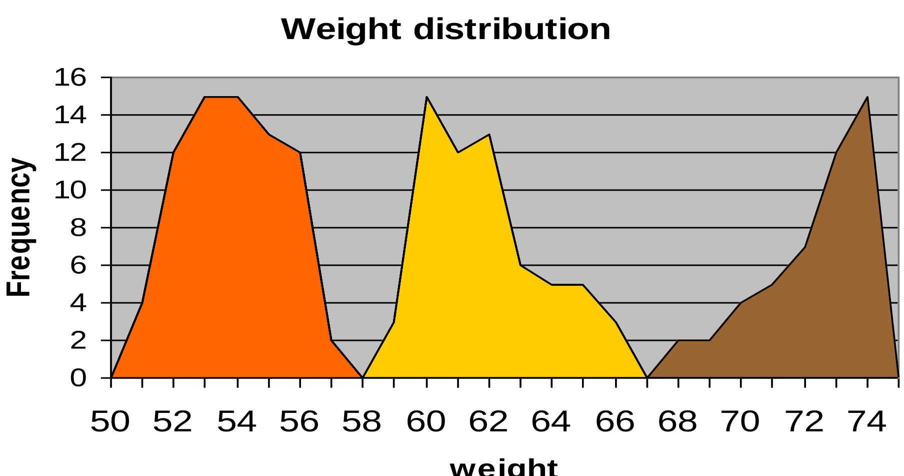
- join(CD=>AB,BD=>AC) would produce the candidate rule D => ABC

- Prune rule D=>ABC if its subset AD=>BC does not have high confidence



How to choose intervals?

1. Interval with a fixed "reasonable" granularity
   Ex. intervals of 10 cm for height.
2. Interval size is defined by some domain dependent criterion
   Ex.: 0-20ML, 21-22ML, 23-24ML, 25-26ML, >26ML
3. Interval size determined by analyzing data, studying the distribution or using clustering



50 - 58 kg
59-67 kg
> 68 kg

Discretization of quantitative attributes

1. Quantitative attributes are statically discretized by using predefined concept hierarchies:
   - elementary use of background knowledge

Loose interaction between Apriori and discretizer

2. Quantitative attributes are dynamically discretized
   - into “bins” based on the distribution of the data.
   - considering the distance between data points.

Tighter interaction between Apriori and discretizer

Reasoning with AR

- **Significance**:

  Example: <1, {a, b}>
  <2, {a} >
  <3, {a, b, c}>
  <4, {b, d}>

  {b} ⇒ {a} has confidence (66%), but is not significant as support({a}) = 75%.

Beyond Support and Confidence

- Example 1: (Aggarwal & Yu, PODS98)

	coffee	not coffee	sum(row)
tea	20	5	25
not tea	70	5	75
sum(col.)	90	10	100

- {tea} => {coffee} has high support (20%) and confidence (80%)
- However, a priori probability that a customer buys coffee is 90%
  - A customer who is known to buy tea is less likely to buy coffee (by 10%)
  - There is a negative correlation between buying tea and buying coffee
  - {~tea} => {coffee} has higher confidence(93%)

Correlation and Interest

- Two events are independent if $P(A \wedge B) = P(A)*P(B)$, otherwise are correlated.
- Interest = $P(A \wedge B) / P(B)*P(A)$
- Interest expresses measure of correlation
  - = 1 $\Rightarrow$ A and B are independent events
  - less than 1 $\Rightarrow$ A and B negatively correlated,
  - greater than 1 $\Rightarrow$ A and B positively correlated.
  - In our example, I(*buy tea* $\wedge$ *buy coffee*)=0.89 i.e. they are negatively correlated.

Computing Interestingness Measure

- **Given a rule $X \rightarrow Y$, information needed to compute rule interestingness can be obtained from a contingency table**

Contingency table for $X \rightarrow Y$

	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.

Statistical-based Measures

- **Measures that take into account statistical dependence**

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

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

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

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

Example: Lift/Interest

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

Association Rule: Tea $\rightarrow$ Coffee

Confidence= P(Coffee|Tea) = 0.75

but P(Coffee) = 0.9

$\Rightarrow$ Lift = 0.75/0.9= 0.8333 (< 1, therefore is negatively associated)

Drawback of Lift & Interest

	Y	$\overline{Y}$	
X	10	0	10
$\overline{X}$	0	90	90
	10	90	100

	Y	$\overline{Y}$	
X	90	0	90
$\overline{X}$	0	10	10
	90	10	100

$$Lift = \frac{0.1}{(0.1)(0.1)} = 10$$

$$Lift = \frac{0.9}{(0.9)(0.9)} = 1.11$$

Statistical independence:

If P(X,Y)=P(X)P(Y) => Lift = 1

Association rules - module outline

- **What are association rules (AR) and what are they used for:**
  - The paradigmatic application: Market Basket Analysis
  - The single dimensional AR (intra-attribute)
- **How to compute AR**
  - Basic Apriori Algorithm and its optimizations
  - Multi-Dimension AR (inter-attribute)
  - Quantitative AR

Multidimensional AR

Associations between values of different attributes :

CID	**nationality**	**age**	**income**
1	Italian	50	low
2	French	40	high
3	French	30	high
4	Italian	50	medium
5	Italian	45	high
6	French	35	high

RULES:

nationality = French $\Rightarrow$ **income** = high [50%, 100%]

income = high $\Rightarrow$ **nationality** = French [50%, 75%]

age = 50 $\Rightarrow$ **nationality** = Italian [33%, 100%]

Single-dimensional vs Multi-dimensional AR

Multi-dimensional

<1, Italian, 50, low>

<2, French, 45, high>

Single-dimensional

<1, {nat/Ita, age/50, inc/low}>

<2, {nat/Fre, age/45, inc/high}>

Schema: <ID, a?, b?, c?, d?>

<1, yes, yes, no, no>

<2, yes, no, yes, no>

<1, {a, b}>

<2, {a, c}>

Quantitative Attributes

- Quantitative attributes (e.g. age, income)
- Categorical attributes (e.g. color of car)

CID	height	weight	income
1	168	75,4	30,5
2	175	80,0	20,3
3	174	70,3	25,8
4	170	65,2	27,0

Problem: too many distinct values

Solution: transform quantitative attributes in categorical ones via **discretization**.

Quantitative Association Rules

CID	Age	Married	NumCars
1	23	No	1
2	25	Yes	1
3	29	No	0
4	34	Yes	2
5	38	Yes	2

[Age: 30..39] and [Married: Yes] ⇒ [NumCars:2]

support = 40%
confidence = 100%

Discretization of quantitative attributes

Solution: each value is replaced by the interval to which it belongs.

height: 0-150cm, 151-170cm, 171-180cm, >180cm

weight: 0-40kg, 41-60kg, 60-80kg, >80kg

income: 0-10ML, 11-20ML, 20-25ML, 25-30ML, >30ML

CID	**height**	**weight**	**income**
1	151-171	60-80	>30
2	171-180	60-80	20-25
3	171-180	60-80	25-30
4	151-170	60-80	25-30

Problem: the discretization may be useless (see **weight**).