# Association Analysis: Basic Concepts and Algorithms


Lecture Notes for Chapter 6

Slides by Tan, Steinbach, Kumar adapted by Michael Hahsler

# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

# 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!

# 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 ($\sigma$)**
  - Frequency of occurrence of an itemset
  - E.g. $\sigma$({Milk, Bread,Diaper}) = 2
- **Support**
  - Fraction of transactions that contain an itemset
  - E.g. s({Milk, Bread, Diaper}) = $\sigma$({Milk, Bread,Diaper}) / |T| = 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 |

$$s(X) = \frac{\sigma(X)}{|T|}$$

# 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:

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

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

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

$$c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} = \frac{s(X \cup Y)}{s(X)}$$

# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

# 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

  $\Rightarrow$ 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} $\rightarrow$ {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)

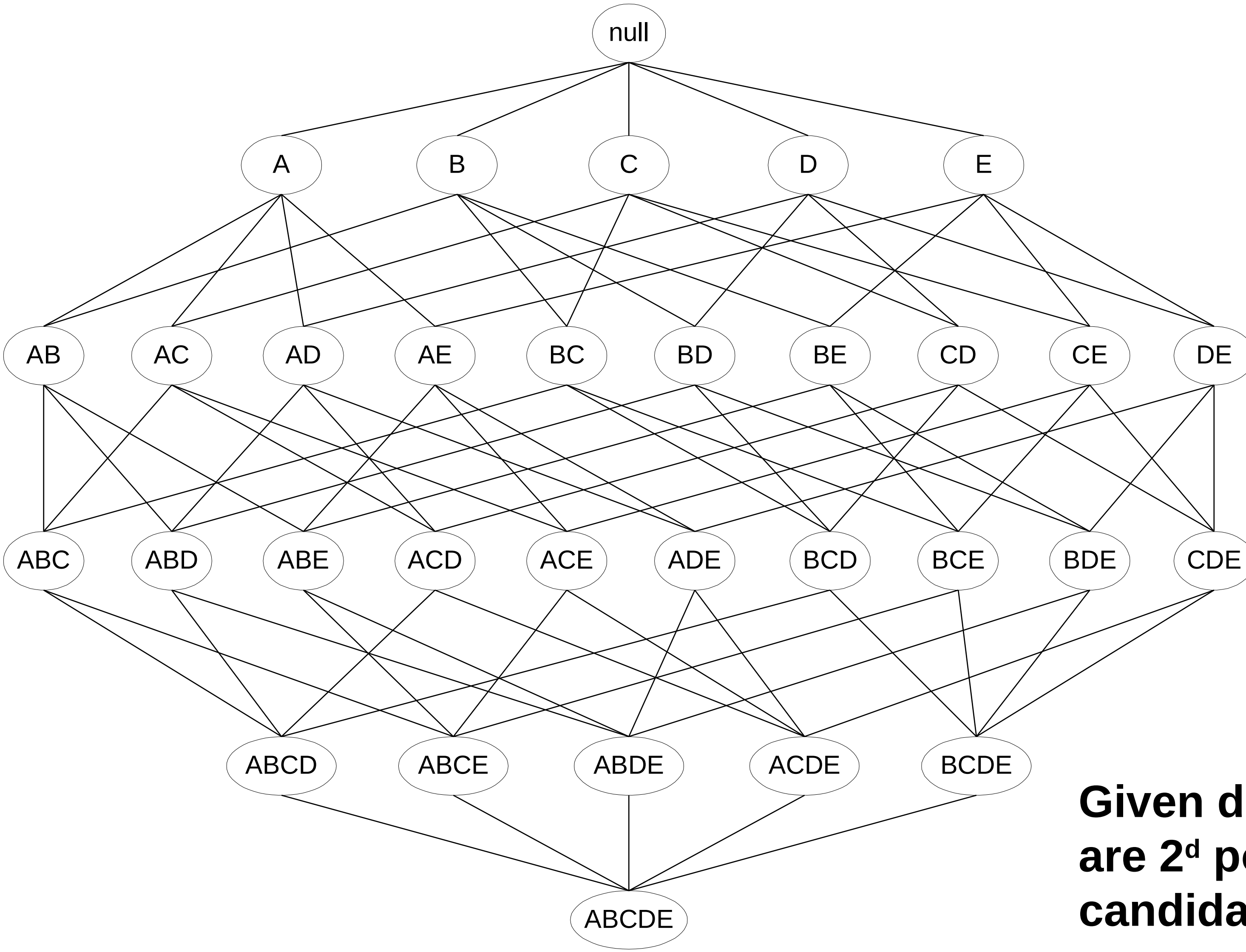
Observations:

- 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

# 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

# Frequent Itemset Generation

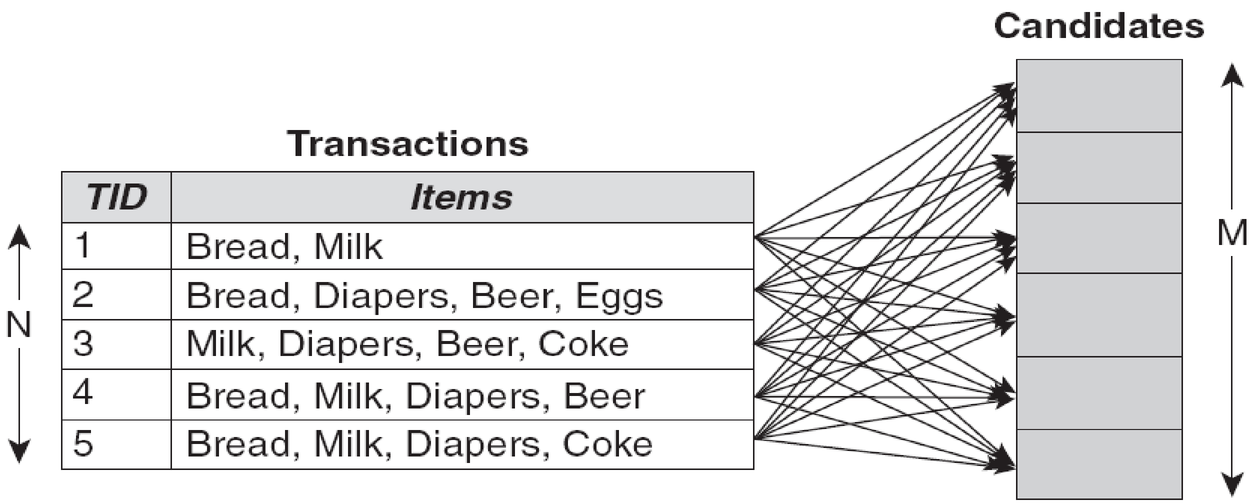



**Given d items, there are $2^d$ possible candidate itemsets**

# 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



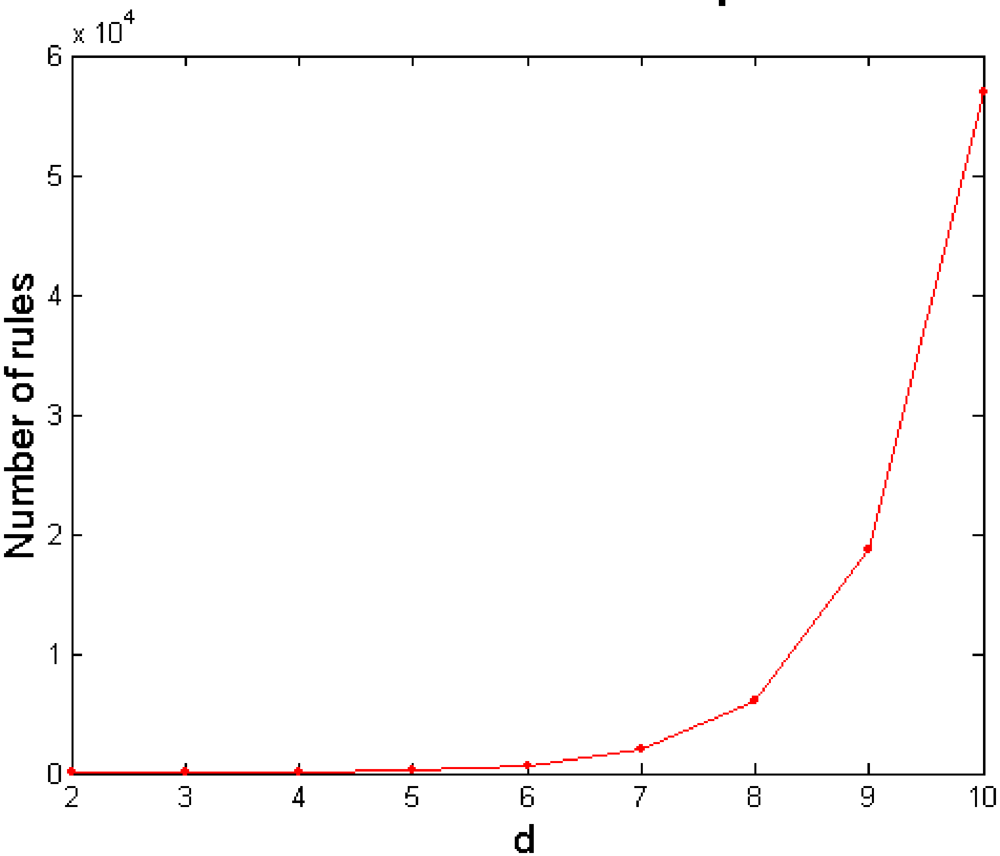

| TID | Items |
|---|---|
| 1 | Bread, Milk |
| 2 | Bread, Diapers, Beer, Eggs |
| 3 | Milk, Diapers, Beer, Coke |
| 4 | Bread, Milk, Diapers, Beer |
| 5 | Bread, Milk, Diapers, Coke |

**Figure 6.2.** Counting the support of candidate itemsets.

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

# 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**

# 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

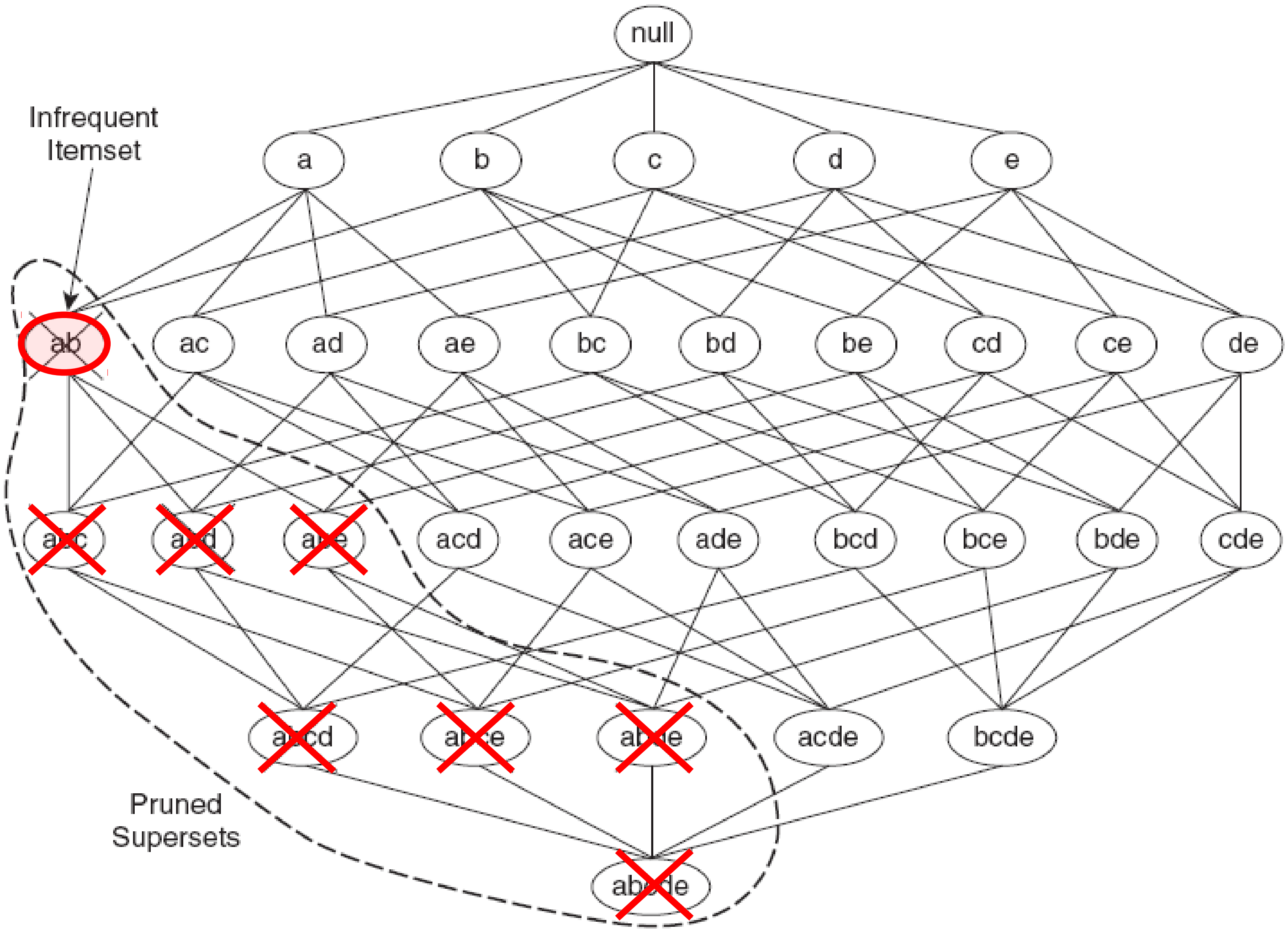



**Figure 6.4.** An illustration of support-based pruning. If $\{a, b\}$ is infrequent, then all supersets of $\{a, b\}$ are infrequent.

# Illustrating Apriori Principle

Items (1-itemsets)

| Item | Count |
|---|---|
| **Bread** | **4** |
| ~~**Coke**~~ | **2** |
| **Milk** | **4** |
| **Beer** | **3** |
| **Diaper** | **4** |
| ~~**Eggs**~~ | **1** |

Pairs (2-itemsets)

(No need to generate candidates involving Coke or Eggs)

| Itemset | Count |
|---|---|
| **{Bread,Milk}** | **3** |
| ~~**{Bread,Beer}**~~ | **2** |
| **{Bread,Diaper}** | **3** |
| ~~**{Milk,Beer}**~~ | **2** |
| **{Milk,Diaper}** | **3** |
| **{Beer,Diaper}** | **3** |

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,

$6 + 6 + 1 = 13$

# Apriori Algorithm

Method:

- Let k=1
- Generate frequent itemsets of length 1
- Repeat until no new frequent itemsets are identified
  - Generate length (k+1) candidate itemsets from length k frequent itemsets
  - Prune candidate itemsets containing subsets of length k that are infrequent
  - Count the support of each candidate by scanning the DB
  - Eliminate candidates that are infrequent, leaving only those that are frequent

# 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)

# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

# Maximal Frequent Itemset

**An itemset is maximal frequent if none of its immediate supersets is frequent**

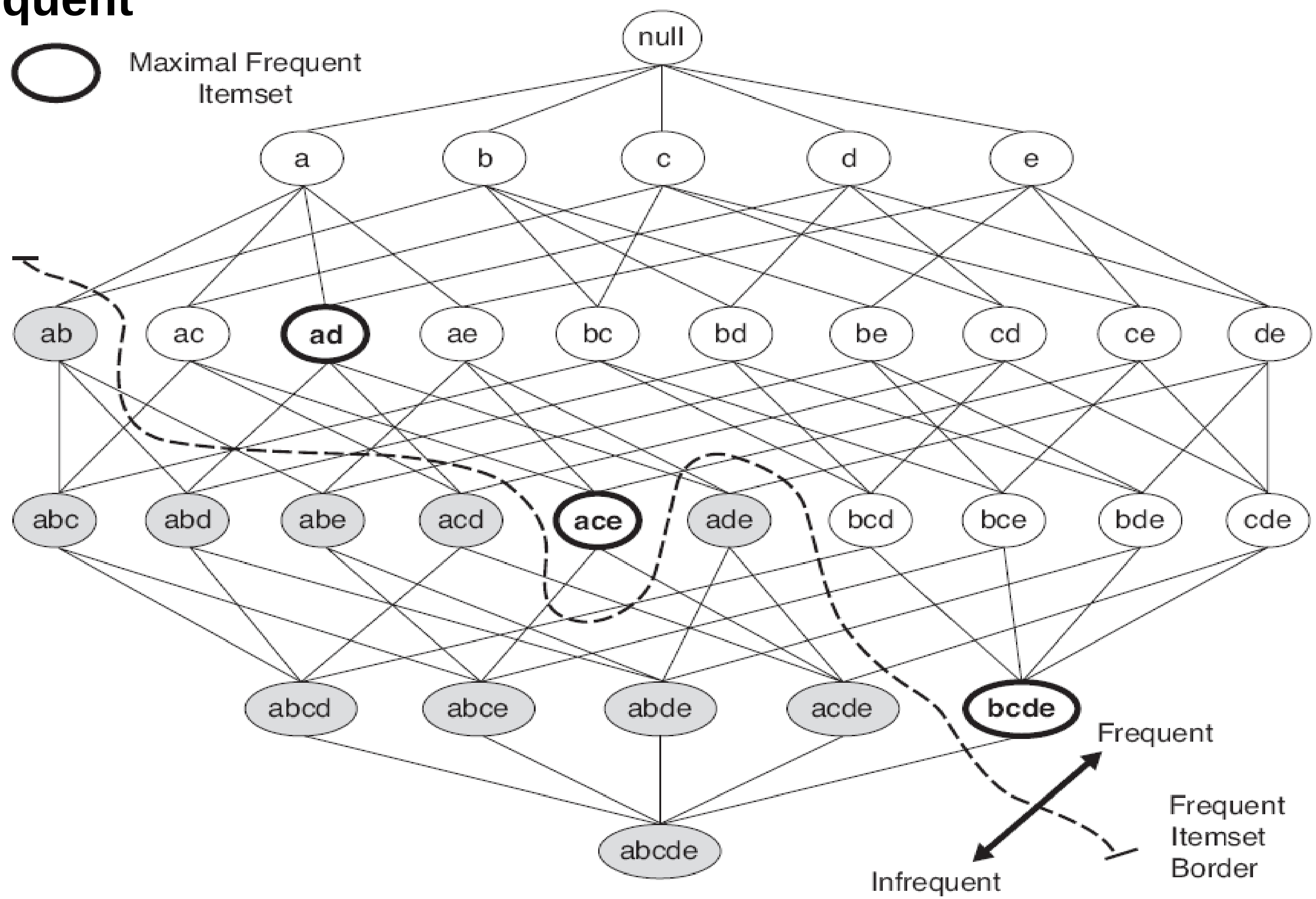



**Figure 6.16.** Maximal frequent itemset.

# Closed Itemset

- An itemset is closed if none of its immediate supersets has the same support as the itemset (can only have smaller support -> see APRIORI principle)

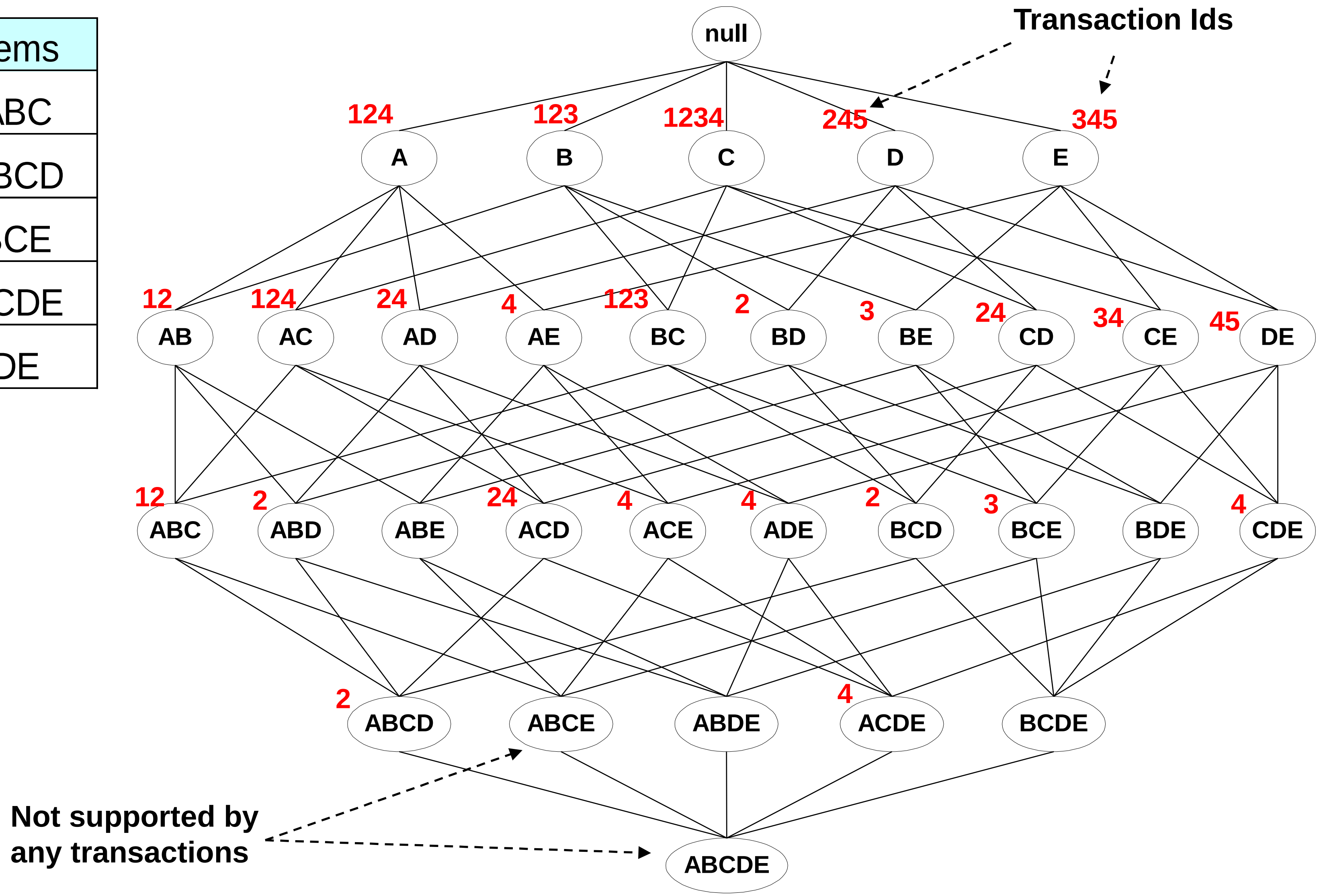
| TID | Items |
|---|---|
| 1 | {A,B} |
| 2 | {B,C,D} |
| 3 | {A,B,C,D} |
| 4 | {A,B,D} |
| 5 | {A,B,C,D} |

| Itemset | Support |
|---|---|
| {A} | 4 |
| {B} | 5 |
| {C} | 3 |
| {D} | 4 |
| {A,B} | 4 |
| {A,C} | 2 |
| {A,D} | 3 |
| {B,C} | 3 |
| {B,D} | 4 |
| {C,D} | 3 |

| Itemset | Support |
|---|---|
| {A,B,C} | 2 |
| {A,B,D} | 3 |
| {A,C,D} | 2 |
| {B,C,D} | 3 |
| {A,B,C,D} | 2 |

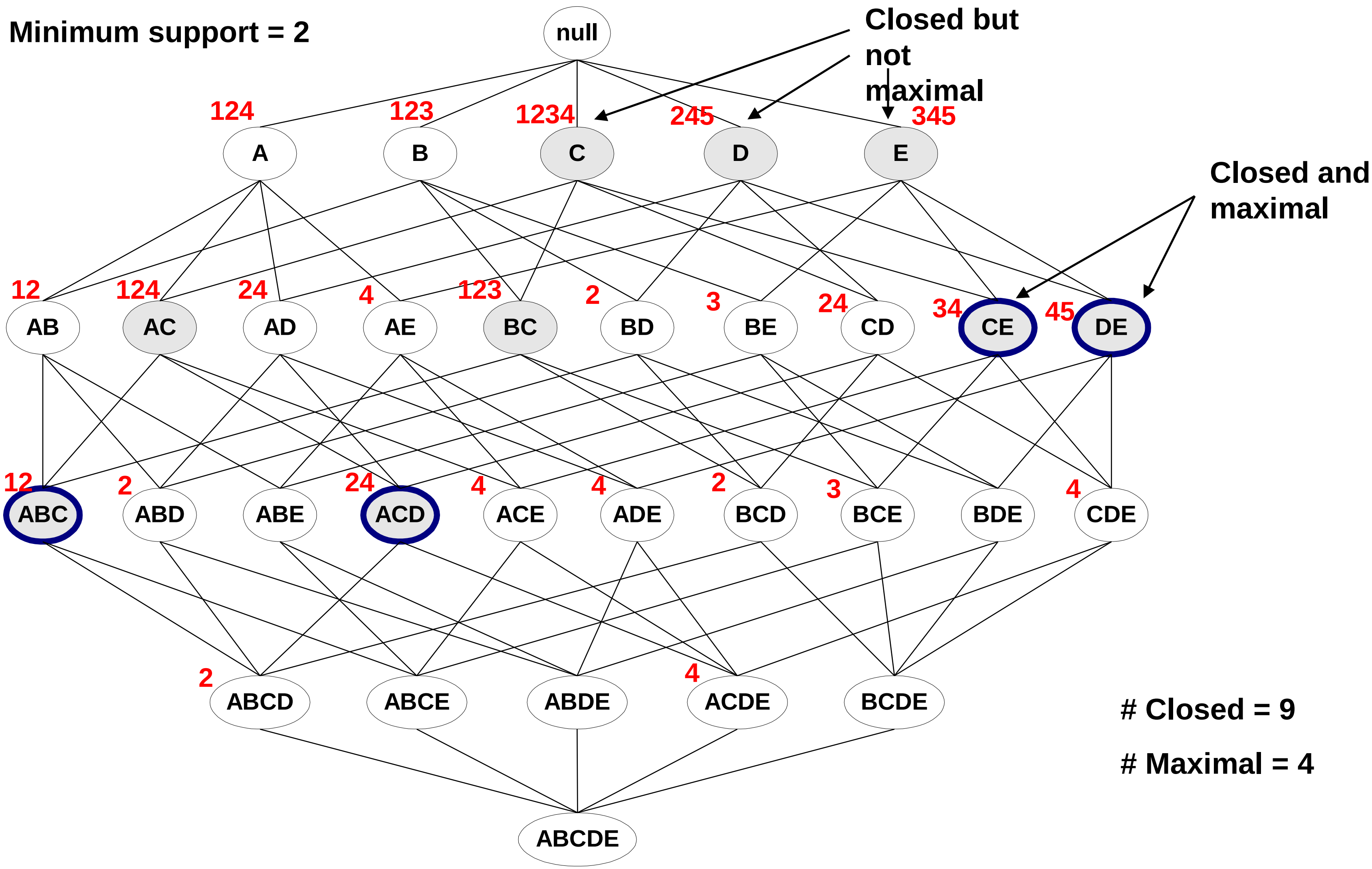
# Maximal vs Closed Itemsets

| TID | Items |
|---|---|
| 1 | ABC |
| 2 | ABCD |
| 3 | BCE |
| 4 | ACDE |
| 5 | DE |



# Maximal vs Closed Frequent Itemsets

Minimum support = 2




# Closed = 9

# Maximal = 4

# Maximal vs Closed Itemsets

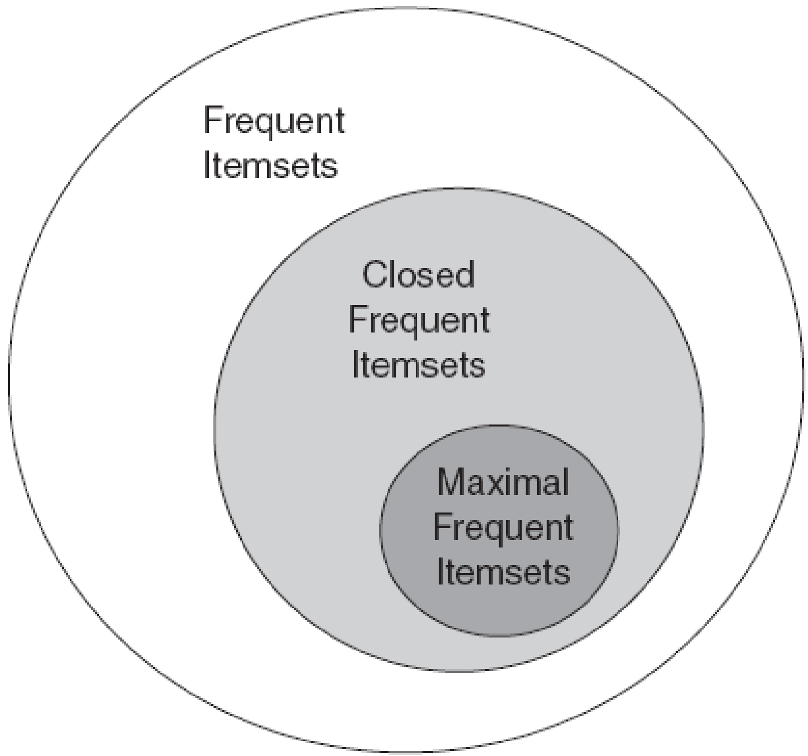


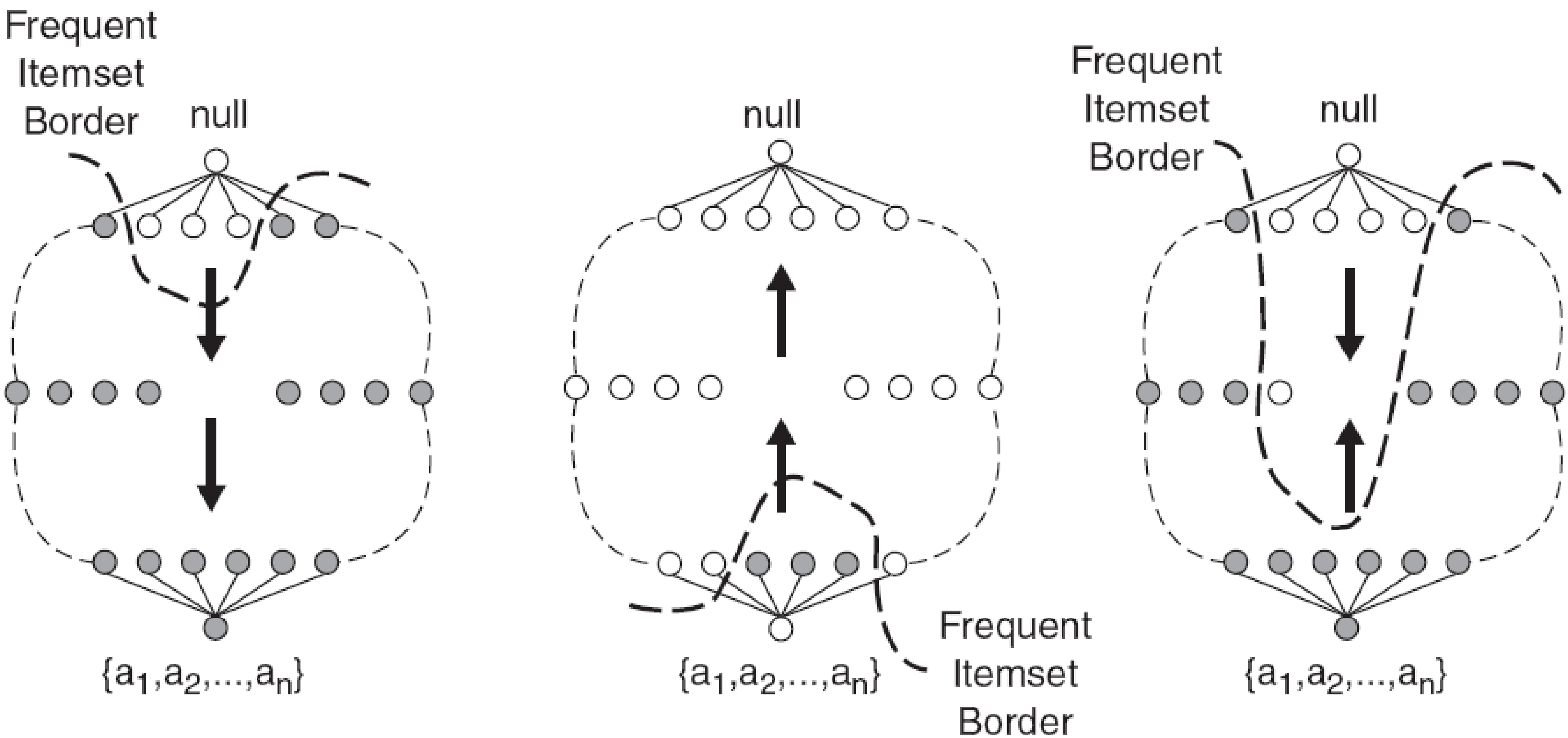

**Figure 6.18.** Relationships among frequent, maximal frequent, and closed frequent itemsets.

# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

# Alternative Methods for Frequent Itemset Generation

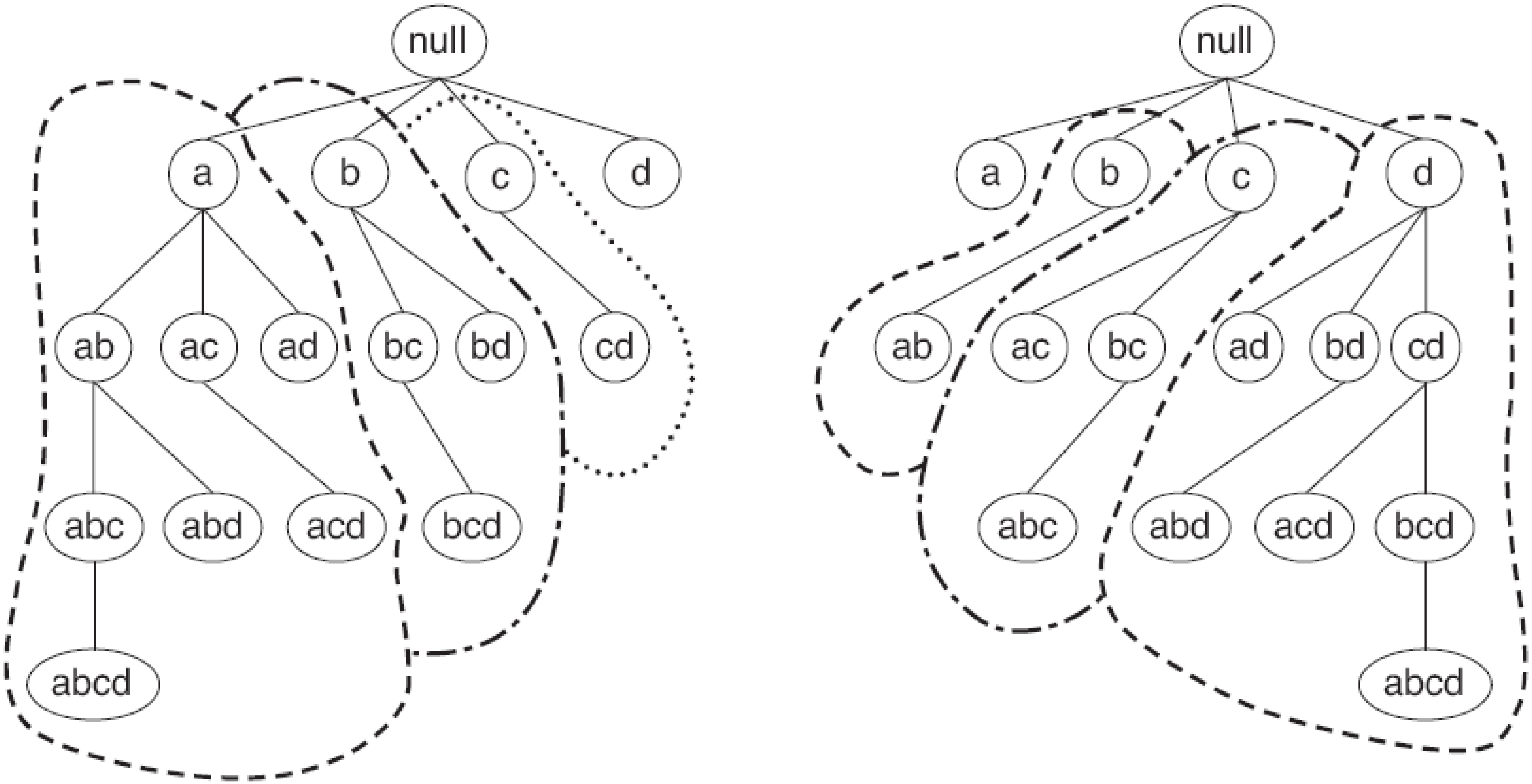
- Traversal of Itemset Lattice
  - General-to-specific vs Specific-to-general




(a) General-to-specific (b) Specific-to-general (c) Bidirectional

# Alternative Methods for Frequent Itemset Generation

- Traversal of Itemset Lattice
  - Equivalent Classes

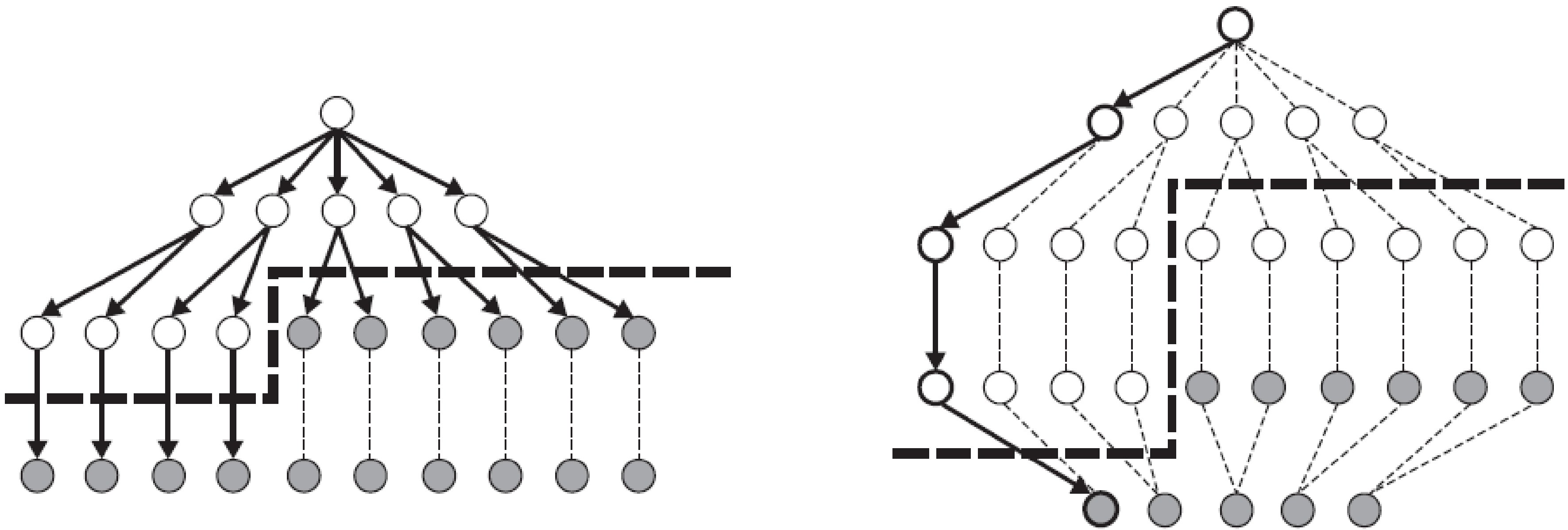



(a) Prefix tree.

(b) Suffix tree.

# Alternative Methods for Frequent Itemset Generation

- Traversal of Itemset Lattice
  - Breadth-first vs Depth-first



(a) Breadth first

(b) Depth first

**Figure 6.21.** Breadth-first and depth-first traversals.

# Alternative Methods for Frequent Itemset Generation

Representation of Database: horizontal vs vertical data layout

Horizontal
Data Layout

| TID | Items |
|---|---|
| 1 | a,b,e |
| 2 | b,c,d |
| 3 | c,e |
| 4 | a,c,d |
| 5 | a,b,c,d |
| 6 | a,e |
| 7 | a,b |
| 8 | a,b,c |
| 9 | a,c,d |
| 10 | b |

Vertical Data Layout

| a | b | c | d | e |
|---|---|---|---|---|
| 1 | 1 | 2 | 2 | 1 |
| 4 | 2 | 3 | 4 | 3 |
| 5 | 5 | 4 | 5 | 6 |
| 6 | 7 | 8 | 9 | |
| 7 | 8 | 9 | | |
| 8 | 10 | | | |
| 9 | | | | |

**Figure 6.23.** Horizontal and vertical data format.

# Alternative Algorithms

- FP-growth
  - Use a compressed representation of the database using an FP-tree
  - Once an FP-tree has been constructed, it uses a recursive divide-and-conquer approach to mine the frequent itemsets
- ECLAT
  - Store transaction id-lists (vertical data layout).
  - Performs fast tid-list intersection (bit-wise XOR) to count itemset frequencies

# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

# Rule Generation

Given a frequent itemset L, find all non-empty subsets X=f $\subset$ L and Y=L – f such that X $\rightarrow$ Y satisfies the minimum confidence requirement

$$c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$$

- 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 $\rightarrow \emptyset$ and $\emptyset \rightarrow$ L)

# Rule Generation

How to efficiently generate rules from frequent itemsets?

- In general, confidence does not have an anti-monotone property

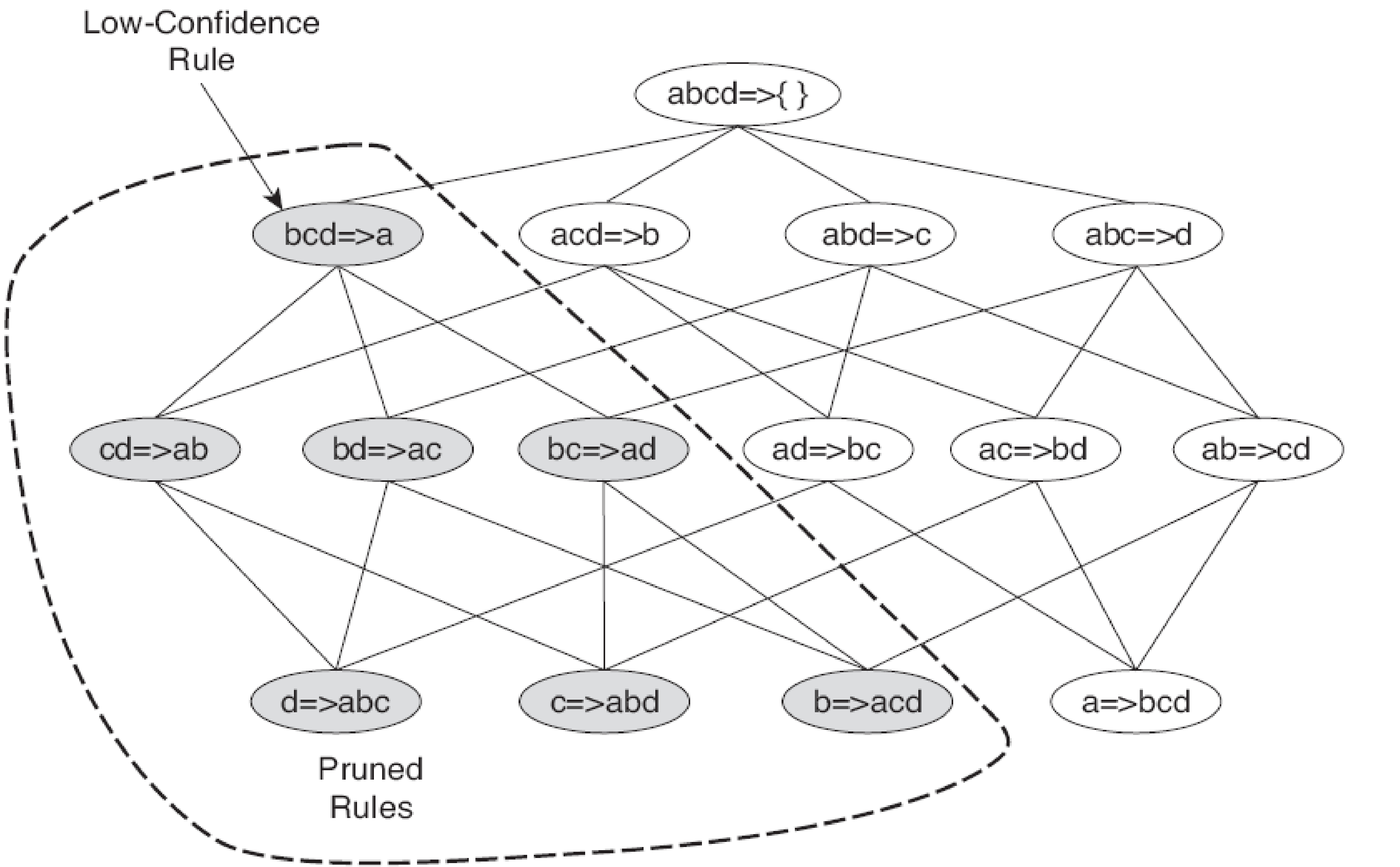
  $c(ABC \rightarrow D)$ can be larger or smaller than $c(AB \rightarrow D)$

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

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

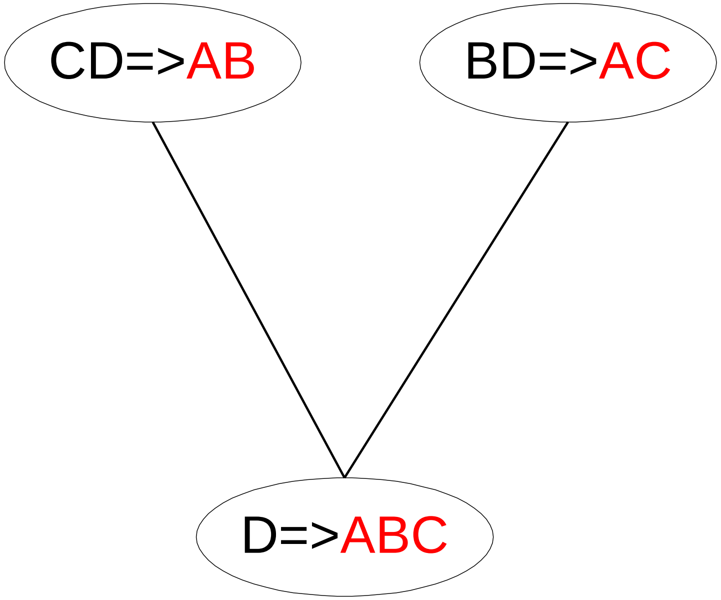
  - Confidence is anti-monotone w.r.t. number of items on the RHS of the rule

# Rule Generation for Apriori Algorithm



# 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

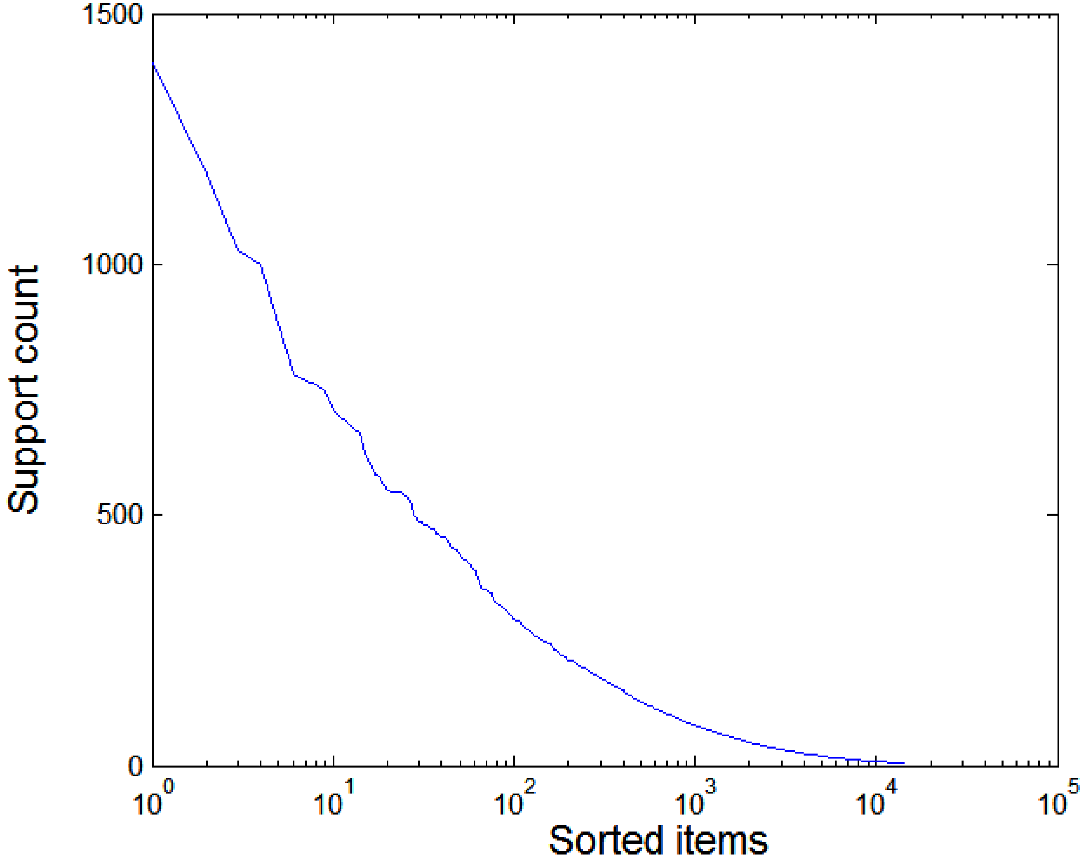
# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

# Effect of Support Distribution

- Many real data sets have skewed support distribution

**Support distribution of a retail data set**



# Effect of Support Distribution

- How to set the appropriate *minsup* threshold?
  - If *minsup* is set too high, we could miss itemsets involving interesting rare items (e.g., expensive products)
  - If *minsup* is set too low, it is computationally expensive and the number of itemsets is very large
- Using a single minimum support threshold may not be effective

# Topics

- Definition
- Mining Frequent Itemsets (APRIORI)
- Concise Itemset Representation
- Alternative Methods to Find Frequent Itemsets
- Association Rule Generation
- Support Distribution
- Pattern Evaluation

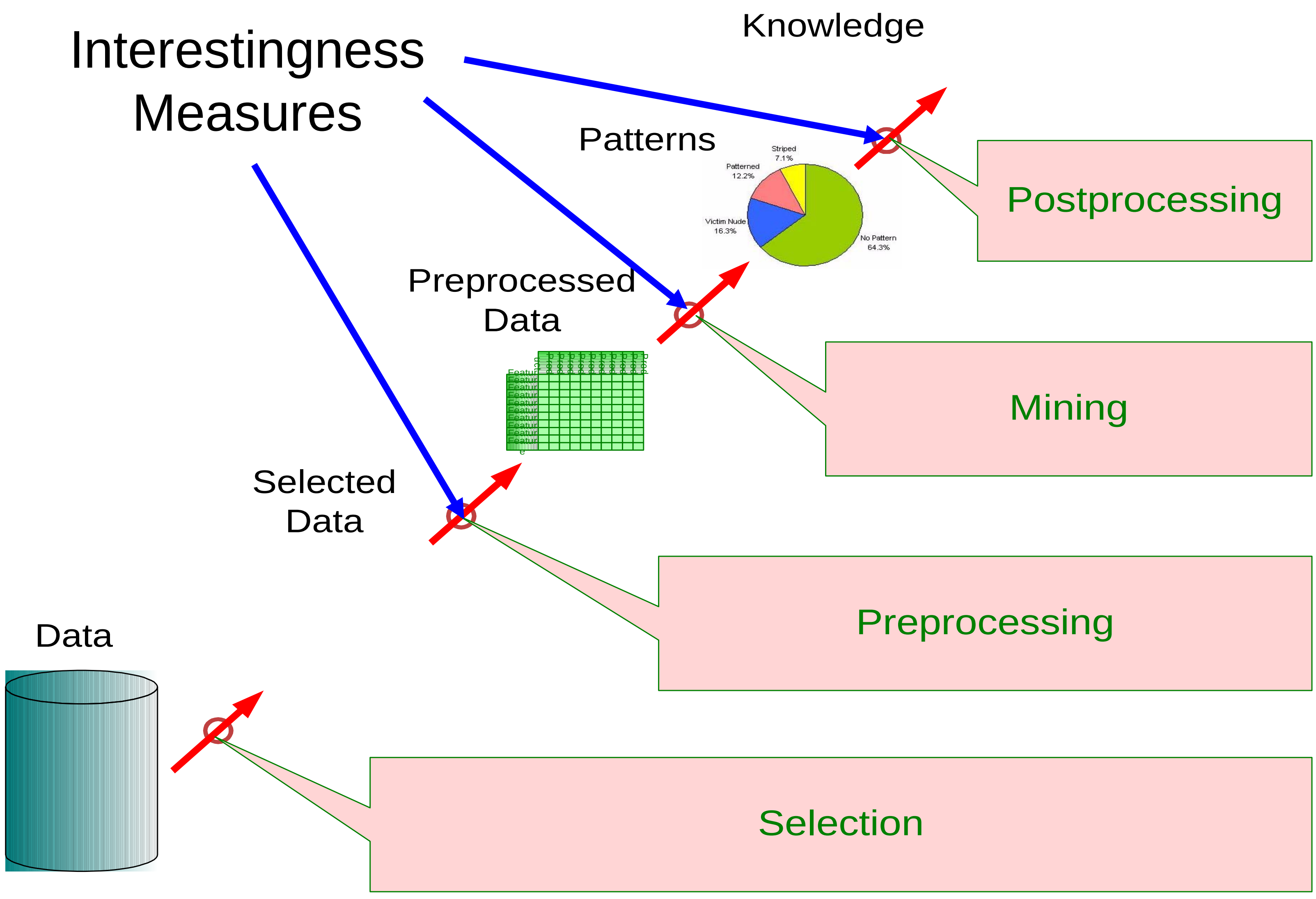
# Pattern Evaluation

- Association rule algorithms tend to produce too many rules
  - many of them are uninteresting or redundant
  - Redundant if $\{A,B,C\} \rightarrow \{D\}$ and $\{A,B\} \rightarrow \{D\}$ have same support & confidence

- Interestingness measures can be used to prune/rank the derived patterns

- In the original formulation of association rules, support & confidence are the only measures used

# Application of Interestingness Measure



# 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 $\overline{Y}$

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

error

Used to define various measures

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

$$sup(\{X, Y\}) = f_{11} / |T|$$

$$conf(X\text{->}Y) = f_{11} / f_{1+}$$

# 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 $\wedge$ Tea) = 15/100 = 0.15

Confidence= P(Coffee|Tea) = 15/20 = 0.75

but P(Coffee) = 90/100 = 0.9

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

$\Rightarrow$ P(Coffee|$\overline{\text{Tea}}$) = 75/80 = **0.9375**

# Statistical Independence

Population of 1000 students

- 600 students know how to swim (S)
- 700 students know how to bike (B)
- 450 students know how to swim and bike (S,B)

- $P(S \wedge B) = 450/1000 = 0.45$ (observed joint prob.)
- $P(S) \times P(B) = 0.6 \times 0.7 = 0.42$ (expected under indep.)

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

# Statistical-based Measures

Measures that take statistical dependence into account for rule: $X \rightarrow Y$

$$\left.\begin{aligned} Lift = Interest &= \frac{P(Y|X)}{P(Y)} = \frac{P(X,Y)}{P(X)P(Y)} \\ PS &= P(X,Y) - P(X)P(Y) \end{aligned}\right\}$$

Deviation from independence

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

Correlation

# Example: Lift/Interest

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

Association Rule: Tea $\rightarrow$ Coffee

Conf(Tea → Coffee)= P(Coffee|Tea) = P(Coffee,Tea)/P(Tea)
= .15/.2 = 0.75

but P(Coffee) = 0.9

$\Rightarrow$ Lift(Tea → Coffee) = P(Coffee,Tee)/(P(Coffee)P(Tee))
= .15/(.9 x .2) = **0.8333**

**Note:** Lift < 1, therefore Coffee and Tea are negatively associated

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 ($K$) | $\sqrt{P(A,B)}\max(P(B\|A)-P(B),P(A\|B)-P(A))$ |

# Comparing Different Measures

10 examples of contingency tables:

| Example | $f_{11}$ | $f_{10}$ | $f_{01}$ | $f_{00}$ |
|---|---|---|---|---|
| E1 | 8123 | 83 | 424 | 1370 |
| E2 | 8330 | 2 | 622 | 1046 |
| E3 | 9481 | 94 | 127 | 298 |
| E4 | 3954 | 3080 | 5 | 2961 |
| E5 | 2886 | 1363 | 1320 | 4431 |
| E6 | 1500 | 2000 | 500 | 6000 |
| E7 | 4000 | 2000 | 1000 | 3000 |
| E8 | 4000 | 2000 | 2000 | 2000 |
| E9 | 1720 | 7121 | 5 | 1154 |
| E10 | 61 | 2483 | 4 | 7452 |

Rankings of contingency tables using various measures:

support & confidence

| # | $\phi$ | $\lambda$ | $\alpha$ | $Q$ | $Y$ | $\kappa$ | $M$ | $J$ | $G$ | $s$ | $c$ | $L$ | $V$ | $I$ | $IS$ | $PS$ | $F$ | $AV$ | $S$ | $\zeta$ | $K$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| E1 | 1 | 1 | 3 | 3 | 3 | 1 | 2 | 2 | 1 | 3 | 5 | 5 | 4 | 6 | 2 | 2 | 4 | 6 | 1 | 2 | 5 |
| E2 | 2 | 2 | 1 | 1 | 1 | 2 | 1 | 3 | 2 | 2 | 1 | 1 | 1 | 8 | 3 | 5 | 1 | 8 | 2 | 3 | 6 |
| E3 | 3 | 3 | 4 | 4 | 4 | 3 | 3 | 8 | 7 | 1 | 4 | 4 | 6 | 10 | 1 | 8 | 6 | 10 | 3 | 1 | 10 |
| E4 | 4 | 7 | 2 | 2 | 2 | 5 | 4 | 1 | 3 | 6 | 2 | 2 | 2 | 4 | 4 | 1 | 2 | 3 | 4 | 5 | 1 |
| E5 | 5 | 4 | 8 | 8 | 8 | 4 | 7 | 5 | 4 | 7 | 9 | 9 | 9 | 3 | 6 | 3 | 9 | 4 | 5 | 6 | 3 |
| E6 | 6 | 6 | 7 | 7 | 7 | 7 | 6 | 4 | 6 | 9 | 8 | 8 | 7 | 2 | 8 | 6 | 7 | 2 | 7 | 8 | 2 |
| E7 | 7 | 5 | 9 | 9 | 9 | 6 | 8 | 6 | 5 | 4 | 7 | 7 | 8 | 5 | 5 | 4 | 8 | 5 | 6 | 4 | 4 |
| E8 | 8 | 9 | 10 | 10 | 10 | 8 | 10 | 10 | 8 | 4 | 10 | 10 | 10 | 9 | 7 | 7 | 10 | 9 | 8 | 7 | 9 |
| E9 | 9 | 9 | 5 | 5 | 5 | 9 | 9 | 7 | 9 | 8 | 3 | 3 | 3 | 7 | 9 | 9 | 3 | 7 | 9 | 9 | 8 |
| E10 | 10 | 8 | 6 | 6 | 6 | 10 | 5 | 9 | 10 | 10 | 6 | 6 | 5 | 1 | 10 | 10 | 5 | 1 | 10 | 10 | 7 |

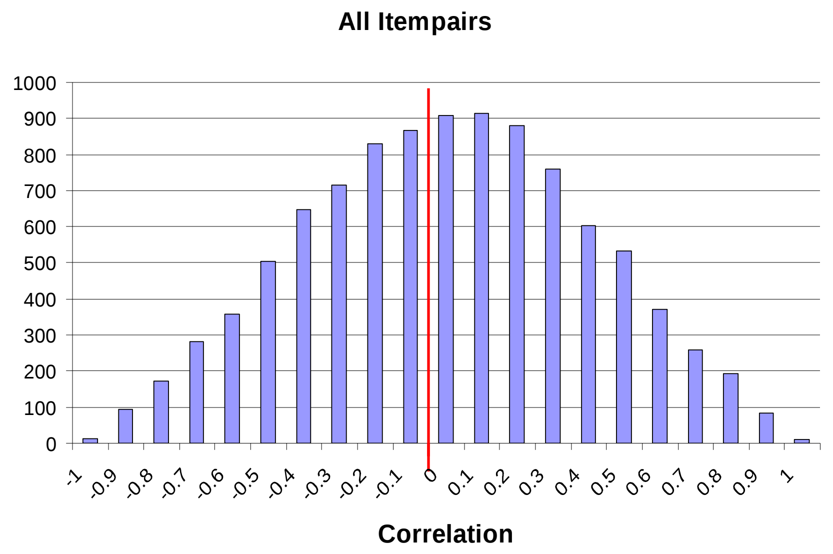
lift

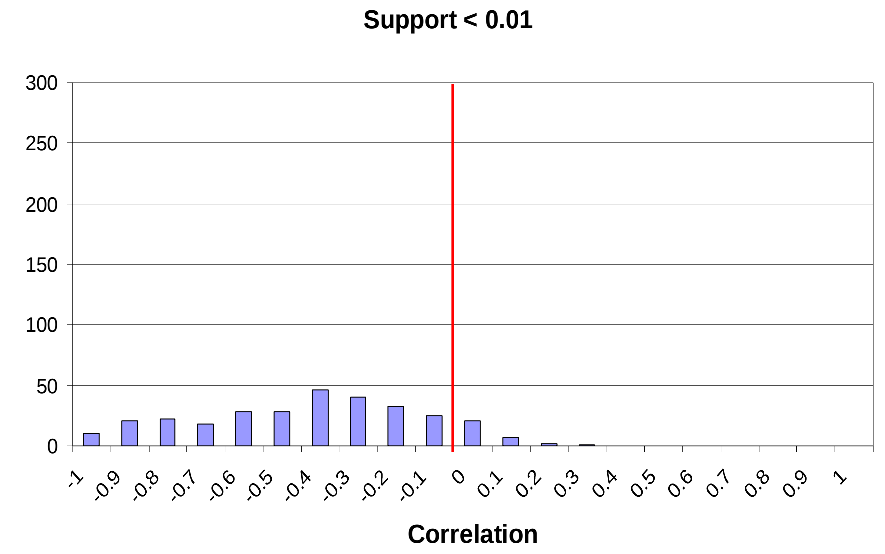
# Support-based Pruning

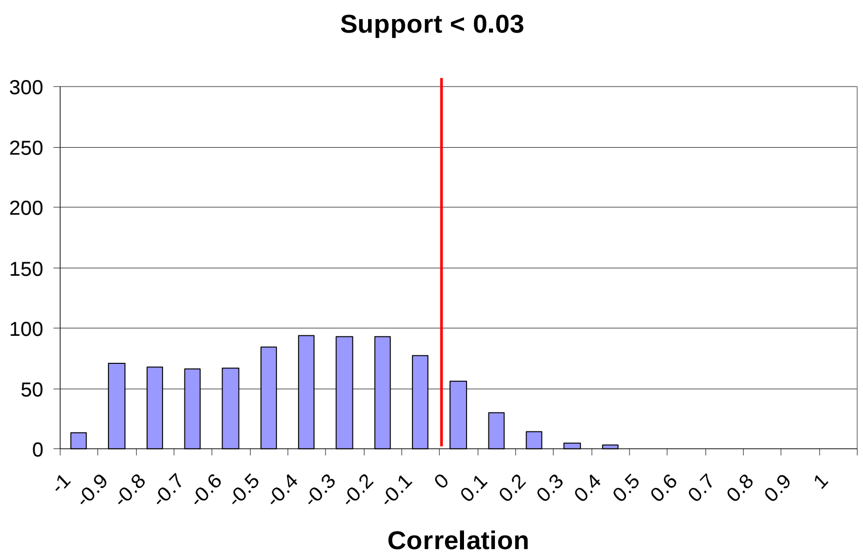
- Most of the association rule mining algorithms use support measure to prune rules and itemsets

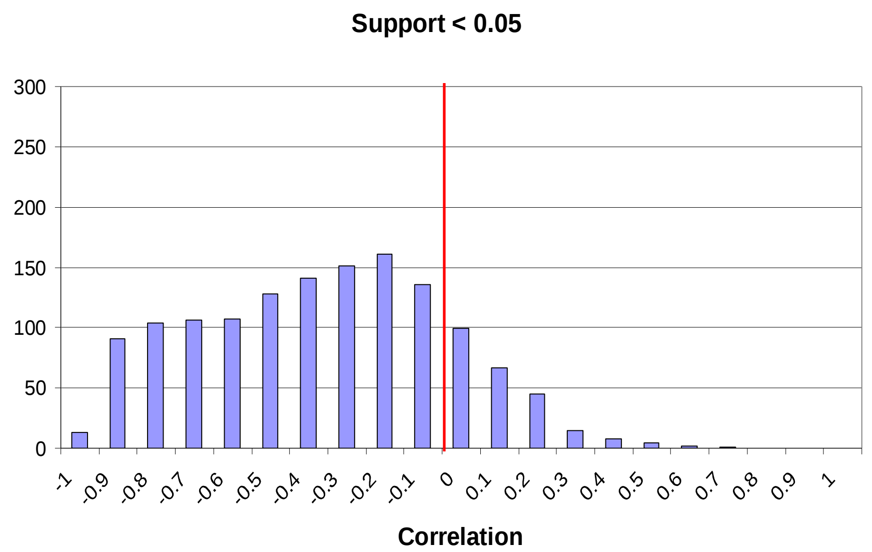
- Study effect of support pruning on correlation of itemsets
  - Generate 10,000 random contingency tables
  - Compute support and pairwise correlation for each table
  - Apply support-based pruning and examine the tables that are removed

# Effect of Support-based Pruning




Support-based pruning eliminates mostly negatively correlated itemsets

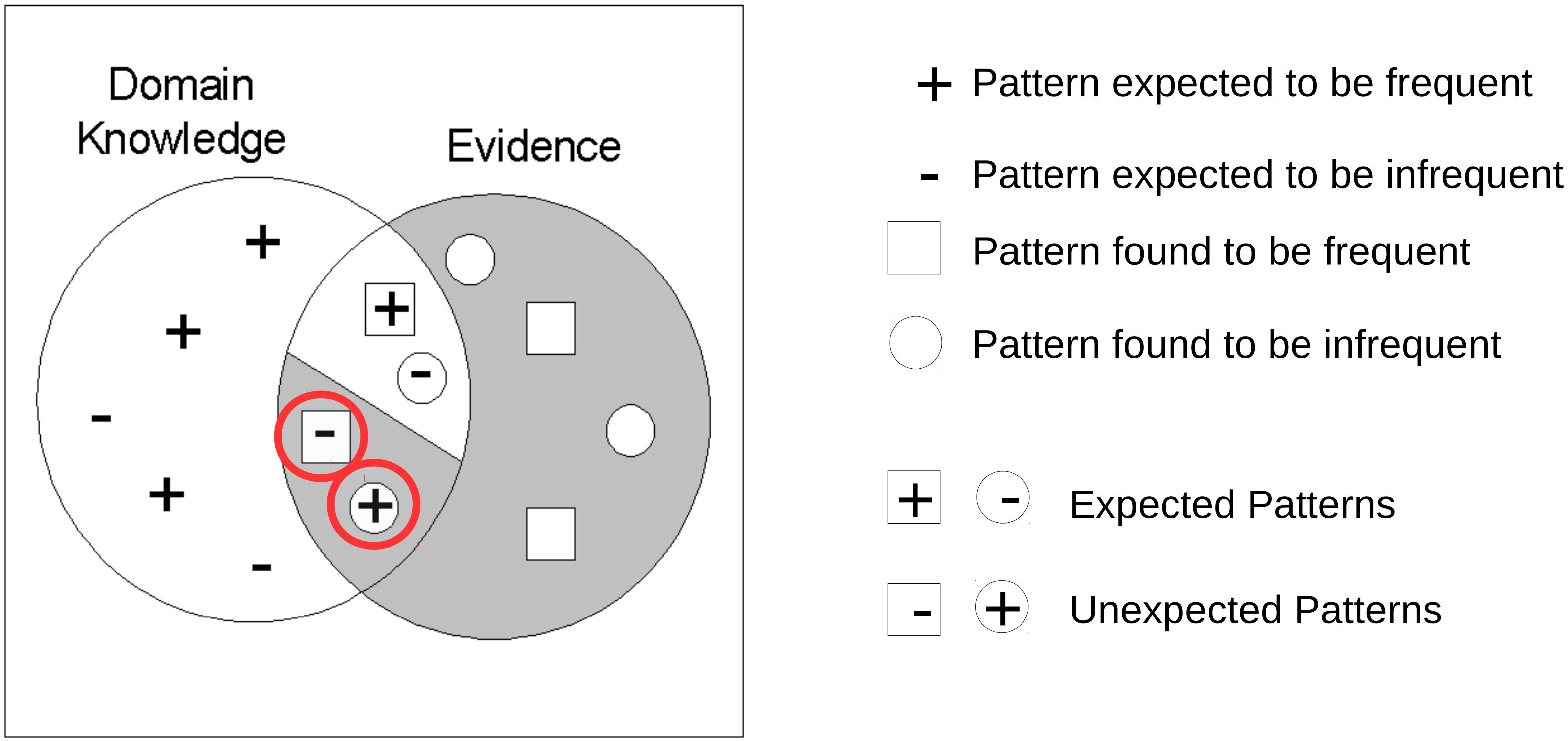
# Subjective Interestingness Measure

- Objective measure:
  - Rank patterns based on statistics computed from data
  - e.g., 21 measures of association (support, confidence, Laplace, Gini, mutual information, Jaccard, etc).

- Subjective measure:
  - Rank patterns according to user's interpretation
    - A pattern is subjectively interesting if it **contradicts the expectation** of a user (Silberschatz & Tuzhilin)
    - A pattern is subjectively interesting if it is **actionable** (Silberschatz & Tuzhilin)

# Interestingness via Unexpectedness

- Need to model expectation of users (domain knowledge)




- Need to combine expectation of users with evidence from data (i.e., extracted patterns)

# Applications for Association Rules

- **Market Basket Analysis**
  Marketing & Retail. E.g., frequent itemsets give information about "other customer who bought this item also bought X"

- **Exploratory Data Analysis**
  Find correlation in very large (= many transactions), high-dimensional (= many items) data

- **Intrusion Detection**
  Rules with low support but very high lift

- **Build Rule-based Classifiers**
  Class association rules (CARs)