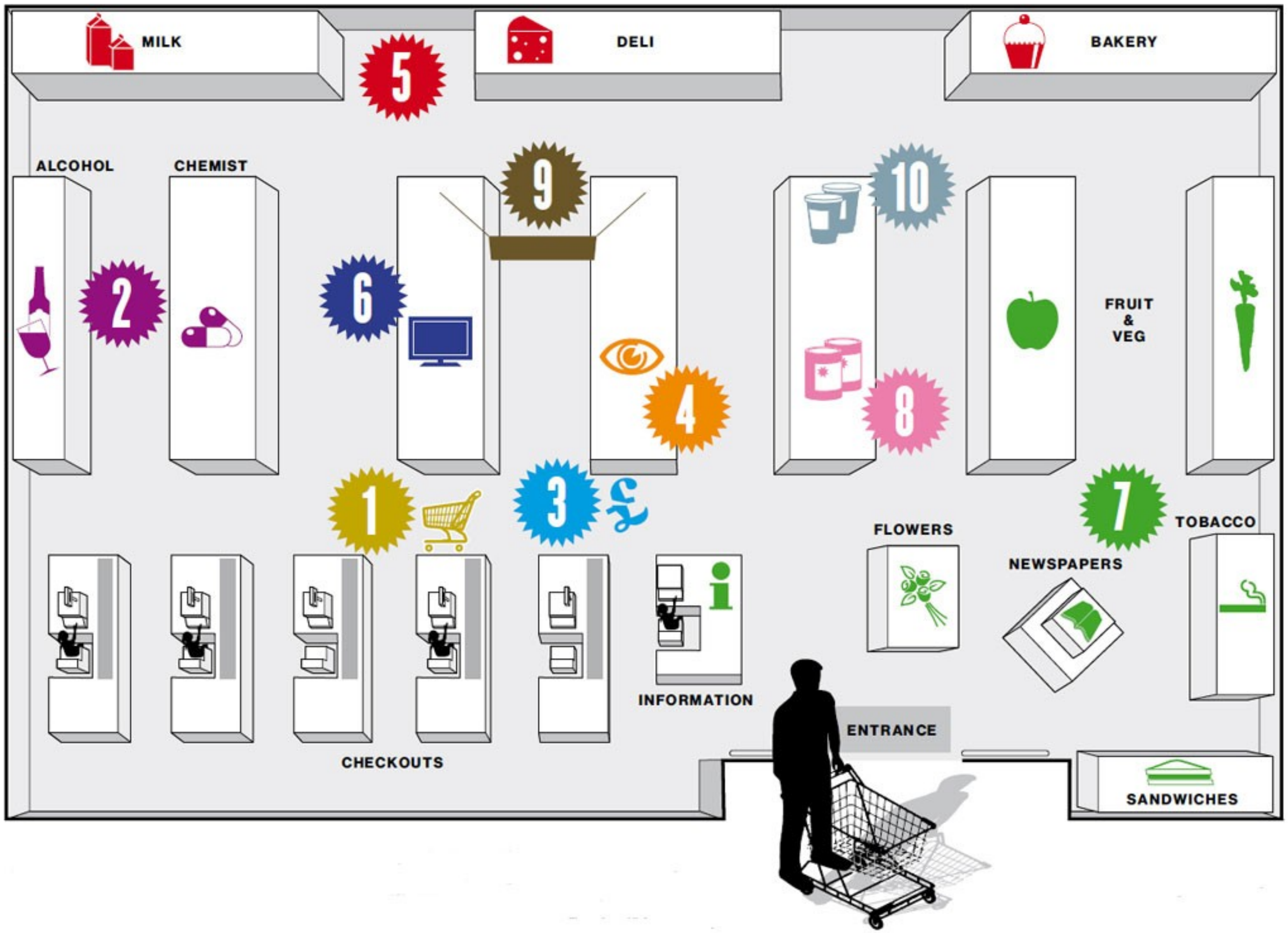
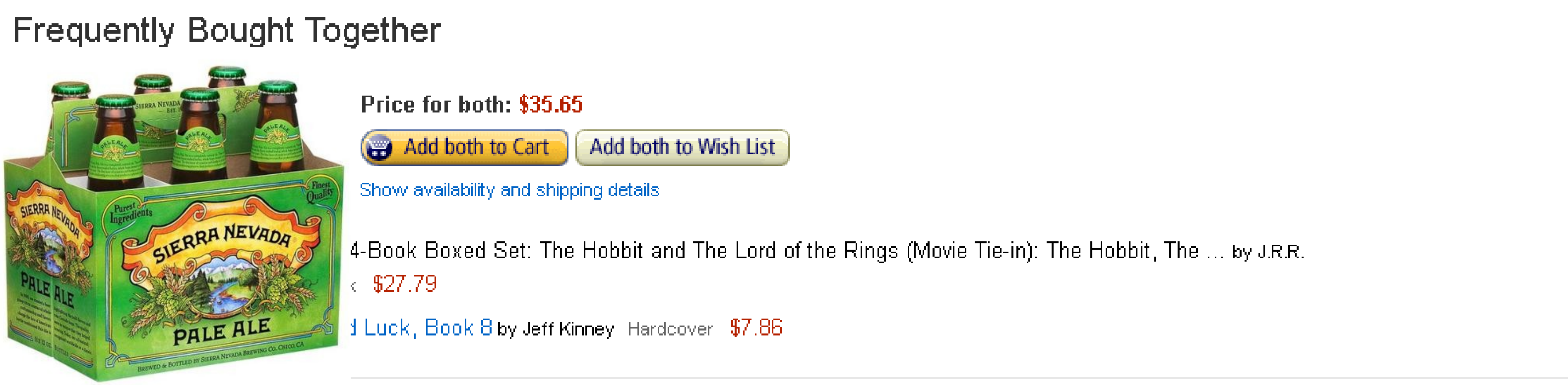
# MINING FREQUENT PATTERNS, ASSOCIATION AND CORRELATION

# Motivation (Brick and Mortar)



# Motivation (Online)

- Frequently Bought Together
- Customers who bought this item also bought

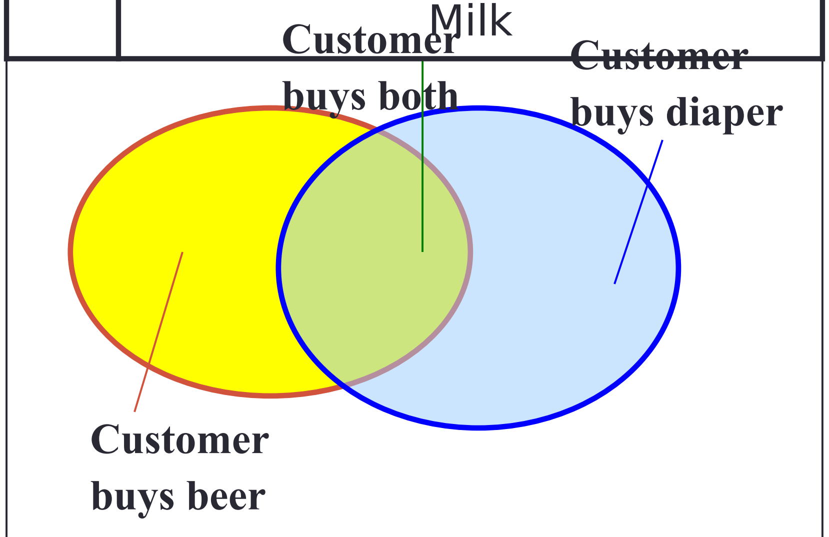



# Frequent Pattern Analysis

- **Frequent pattern**: a pattern (a set of items, subsequences, substructures, etc.) that occurs frequently in a data set
- First proposed by Agrawal, Imielinski, and Swami [AIS93] in the context of **frequent itemsets** and **association rule mining**
  - Motivation: Finding inherent regularities in data
    - What products were often purchased together?
    - What are the subsequent purchases after buying a PC?
    - What kinds of DNA are sensitive to this new drug?
    - Can we automatically classify web documents?
- Applications
  - Market basket analysis, cross-marketing, catalog design, sale campaign analysis, Web log (click stream) analysis, and DNA sequence analysis, etc.

# Basic Concepts or "The Urban Legend of Beer and Diapers"

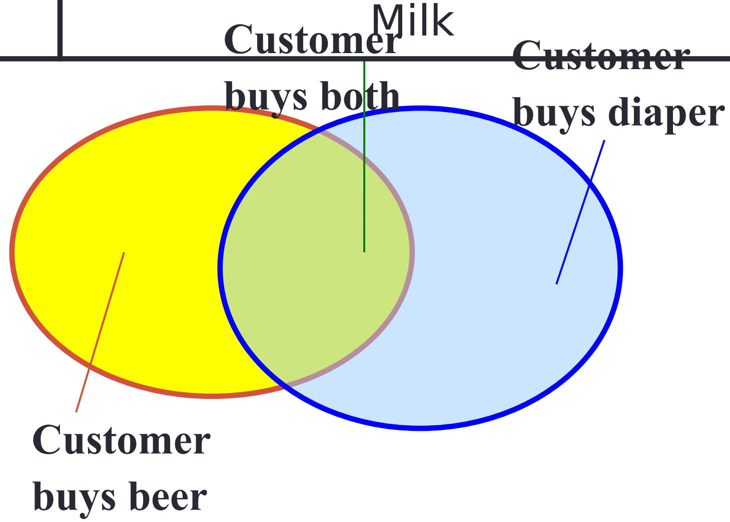
| Tid | Items bought |
|---|---|
| 10 | Beer, Nuts, Diaper |
| 20 | Beer, Coffee, Diaper |
| 30 | Beer, Diaper, Eggs |
| 40 | Nuts, Eggs, Milk |
| 50 | Nuts, Coffee, Diaper, Eggs, Milk |




- **itemset**: A set of one or more items
- **k-itemset** X = $\{x_1, \ldots, x_k\}$
- ***(absolute) support***, or, ***support count*** of X: Frequency or occurrence of an itemset X
- ***(relative) support***, *s*, is the fraction of transactions that contains X (i.e., the probability that a transaction contains X)
- An itemset X is ***frequent*** if X's support is no less than a *minsup* threshold

# Association Rules

| Tid | Items bought |
|---|---|
| 10 | Beer, Nuts, Diaper |
| 20 | Beer, Coffee, Diaper |
| 30 | Beer, Diaper, Eggs |
| 40 | Nuts, Eggs, Milk |
| 50 | Nuts, Coffee, Diaper, Eggs, Milk |




- Find all the rules *X* ☾ *Y* with minimum support and confidence
  - **support**, *s*, probability that a transaction contains X ∪ Y
  - **confidence**, *c*, conditional probability that a transaction having X also contains *Y* *P*(*Y*|*X*)

*Let minsup = 50%, minconf = 50%*

*Freq. Pat.:* Beer:3, Nuts:3, Diaper:4, Eggs:3, {Beer, Diaper}:3

- Association rules: (many more!)
  - *Beer* ☾ *Diaper* (60%, 100%)
  - *Diaper* ☾ *Beer* (60%, 75%)

# Mining Association Rules

**Two-step** process

1. **Find all frequent itemsets**, where itemset frequency is beyond min_sup;

2. From list of frequent itemsets, **generate association rules** satisfying min_sup and confidence

# Support & Confidence

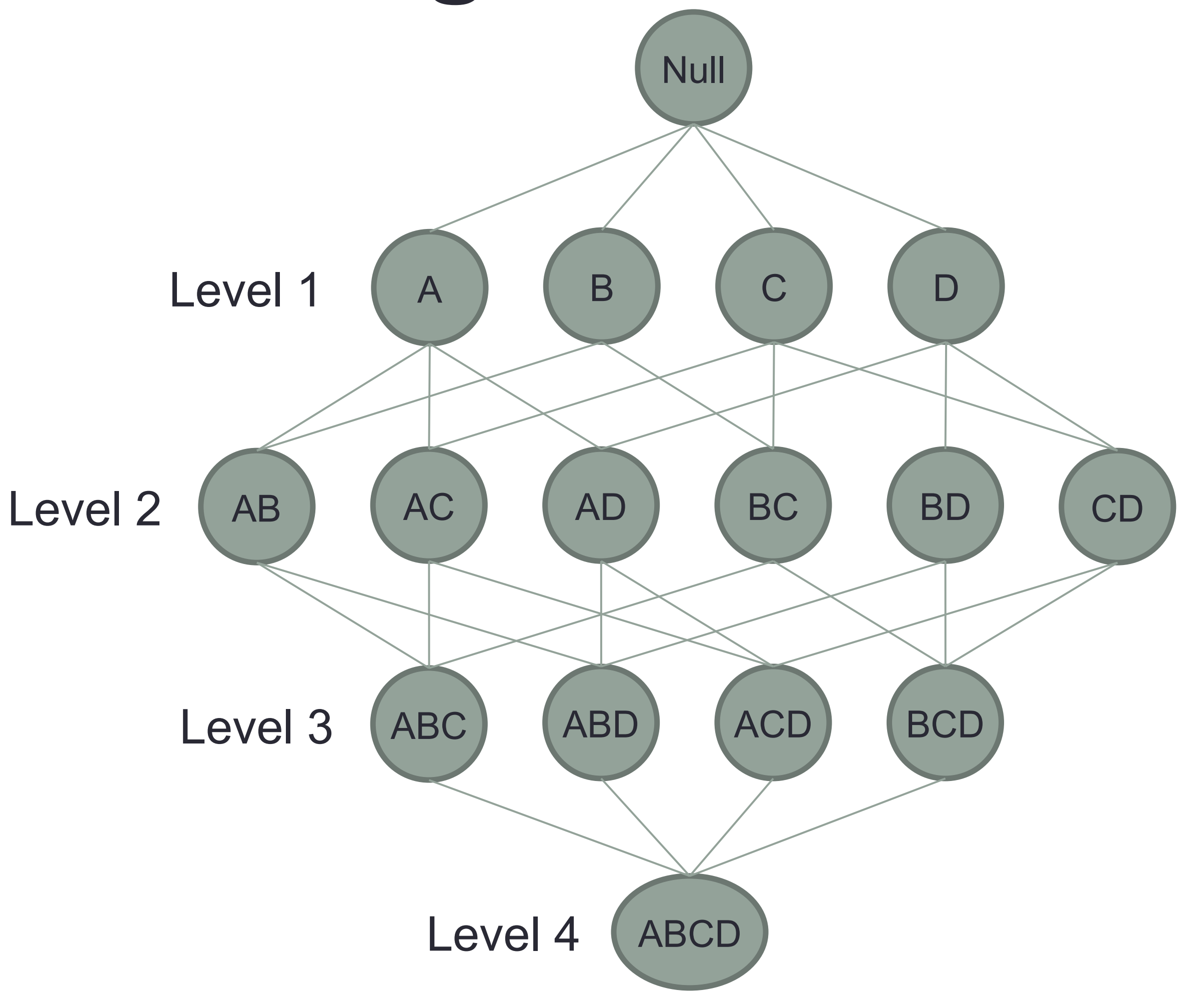
- <u>Support</u> for association rule A ☾ B is proportion of transactions in *D* containing both A and B

$$support = P(A \cap B) = \frac{\text{number of transactions containing both A and B}}{\text{total number of transactions}}$$

- <u>Confidence</u> for association rule A ☾ B measures rule accuracy
- Determined by percentage of transactions in *D* containing A, also containing B

$$\text{confidence} = P(B \mid A) = \frac{P(A \cap B)}{P(A)} = \frac{\text{number of transactions containing both A and B}}{\text{number of transactions containing A}}$$

# The Challenge: Itemset Lattice



# Closed Patterns and Max Patterns

- A long pattern contains a combinatorial number of sub-patterns
- Solution: *Mine* ***closed itemsets*** *and* ***maximal frequent itemsets***
  - An itemset X is ***closed*** if X is *frequent* and there exists *no super-pattern* (Y ⊃ X), *with the same or greater support* than X (proposed by Pasquier, et al. @ ICDT'99)
  - An itemset X is a ***maximal frequent itemset*** if X is frequent and there exists no frequent super-pattern (Y ⊃ X) (proposed by Bayardo @ SIGMOD'98)
- Closed pattern is a lossless compression of freq. patterns
  - Reducing the # of patterns and rules

# Closed Itemset

- Problem with maximal frequent itemsets:
  - Support of their subsets is not known – additional DB scans are needed
- An itemset is closed if none of its immediate supersets has the same support as the itemset

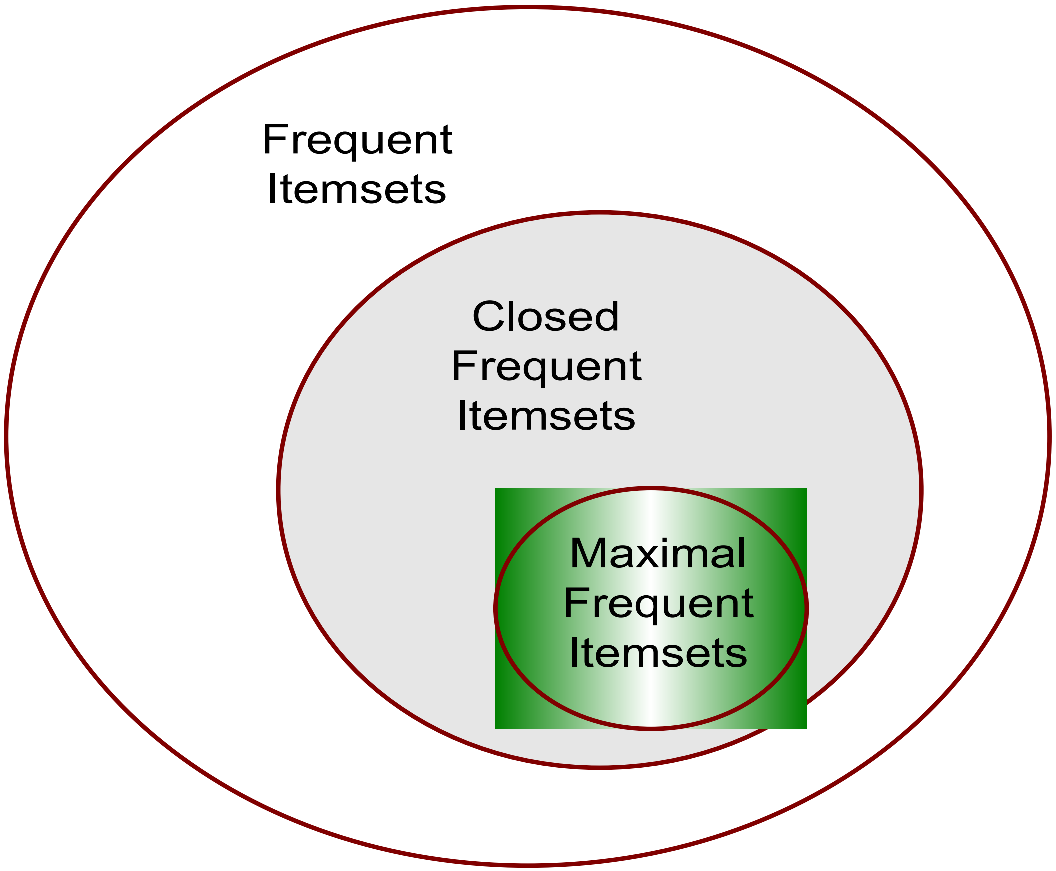
| TID | Items |
|---|---|
| 1 | {A,B} |
| 2 | {B,C,D} |
| 3 | {A,B,C,D} |
| 4 | {A,B,D} |
| 5 | {A,B,C,D} |

| 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 |

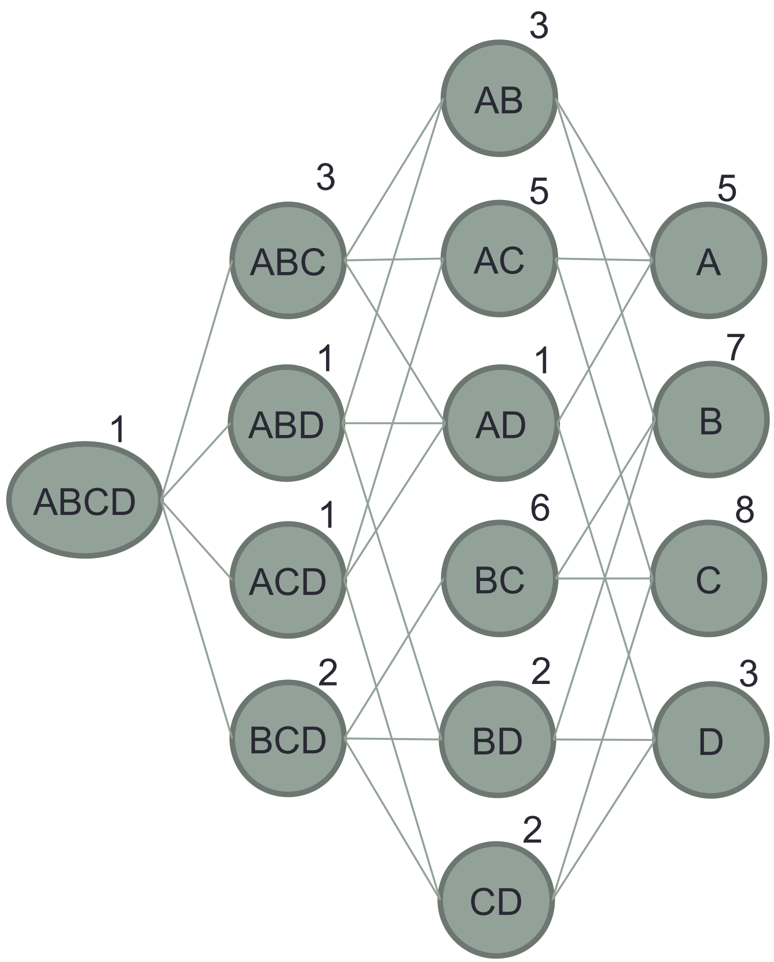
min_sup = 2

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

# Maximal vs. Closed Itemsets



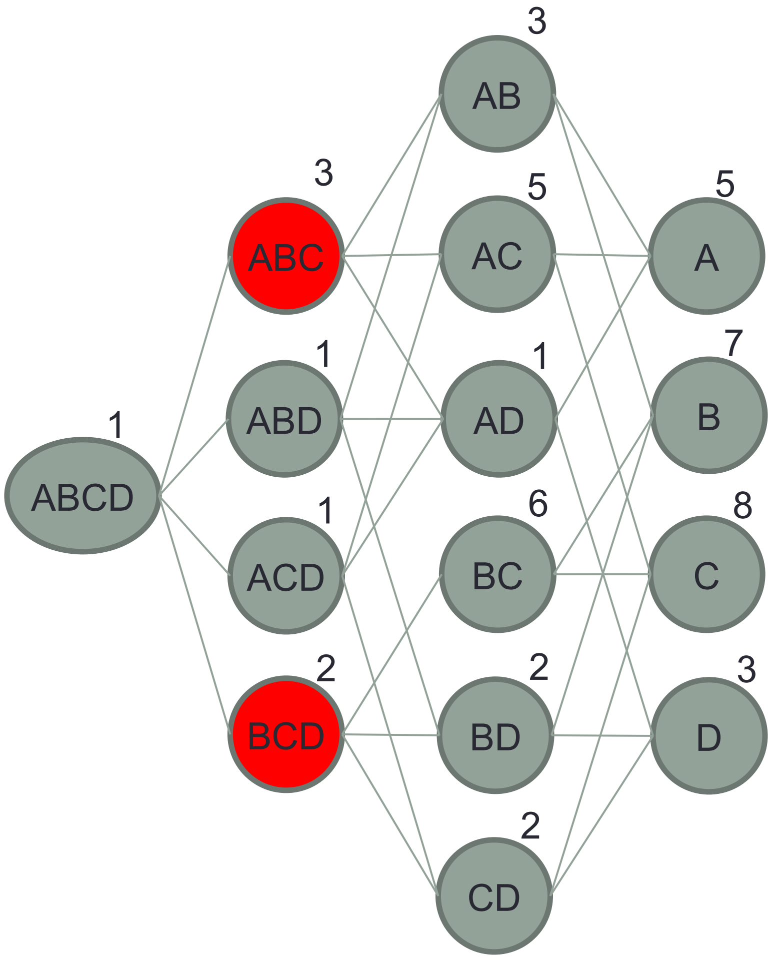
# Maximal Frequent Itemsets




<u>Find the maximal frequent itemsets.</u>
Reminder: A maximal frequent itemset is a frequent itemset with no frequent superset

Minimum Support = 2

# Maximal Frequent Itemsets

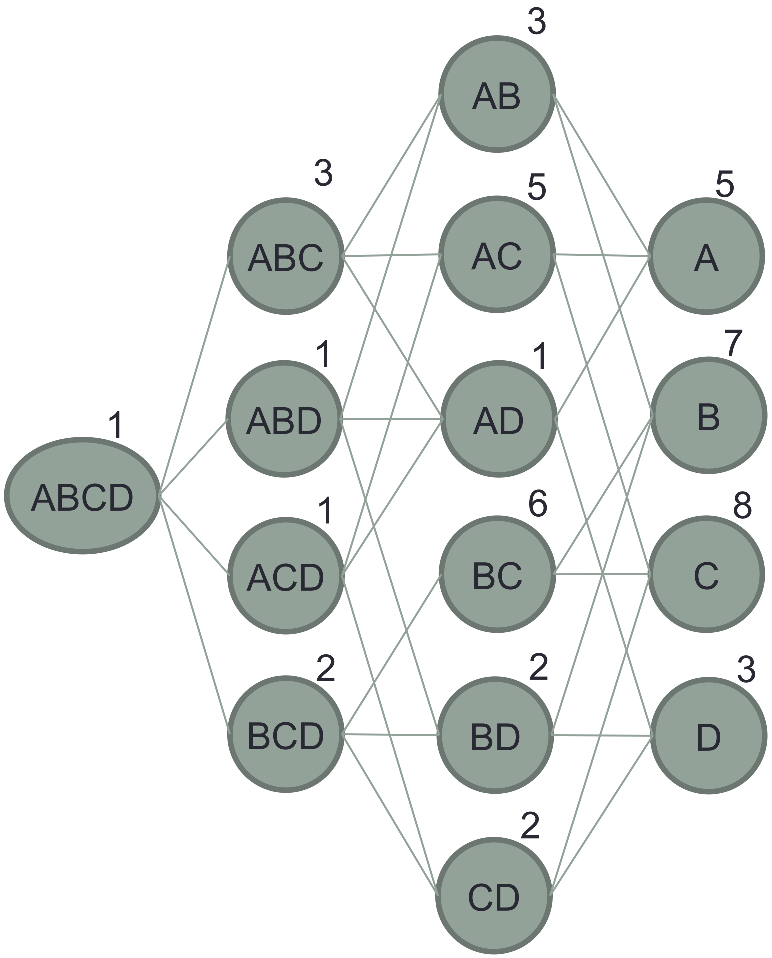



Find the maximal frequent itemsets.

Reminder: A maximal frequent itemset is a frequent itemset with no frequent superset

Minimum Support = 2

# Closed Frequent Itemsets

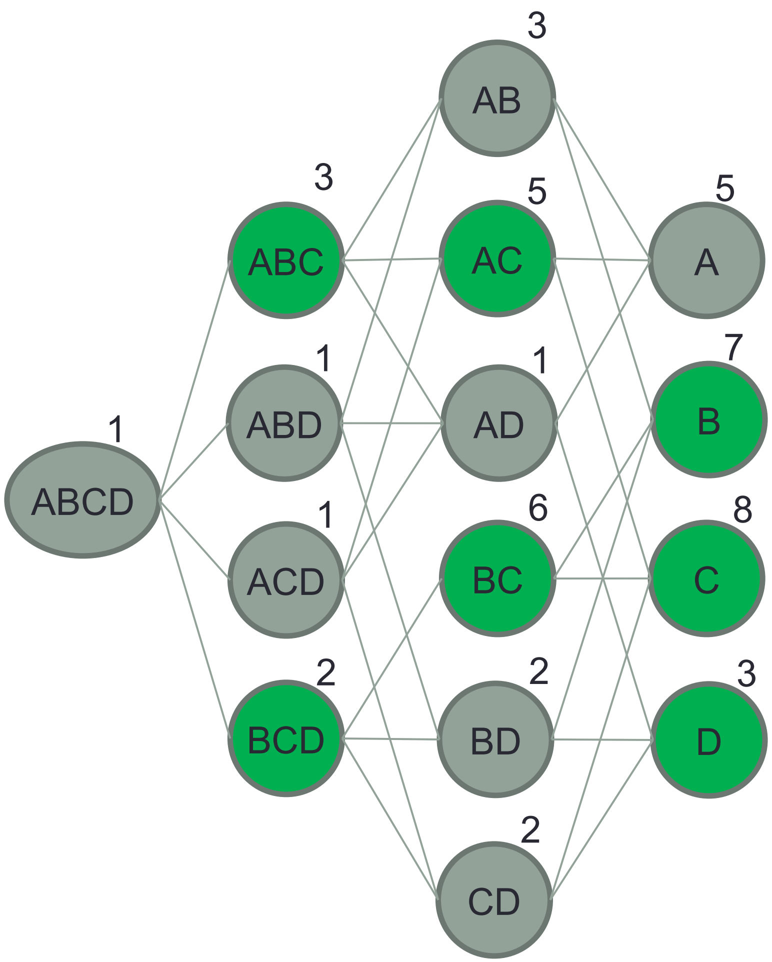



<u>Find the closed frequent itemsets.</u>

Reminder: A closed frequent itemset is a frequent itemset with superset with the smaller support

Minimum Support = 2

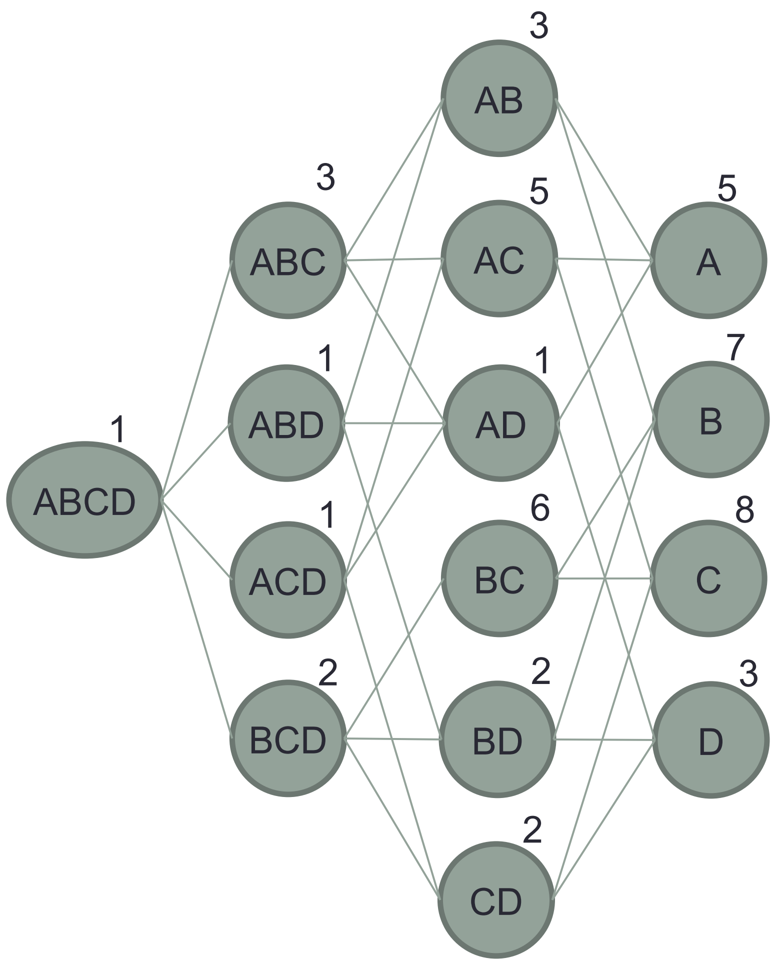
# Closed Frequent Itemsets




<u>Find the closed frequent itemsets.</u>
Reminder: A closed frequent itemset is a frequent itemset with superset with the **smaller** support

Minimum Support = 2

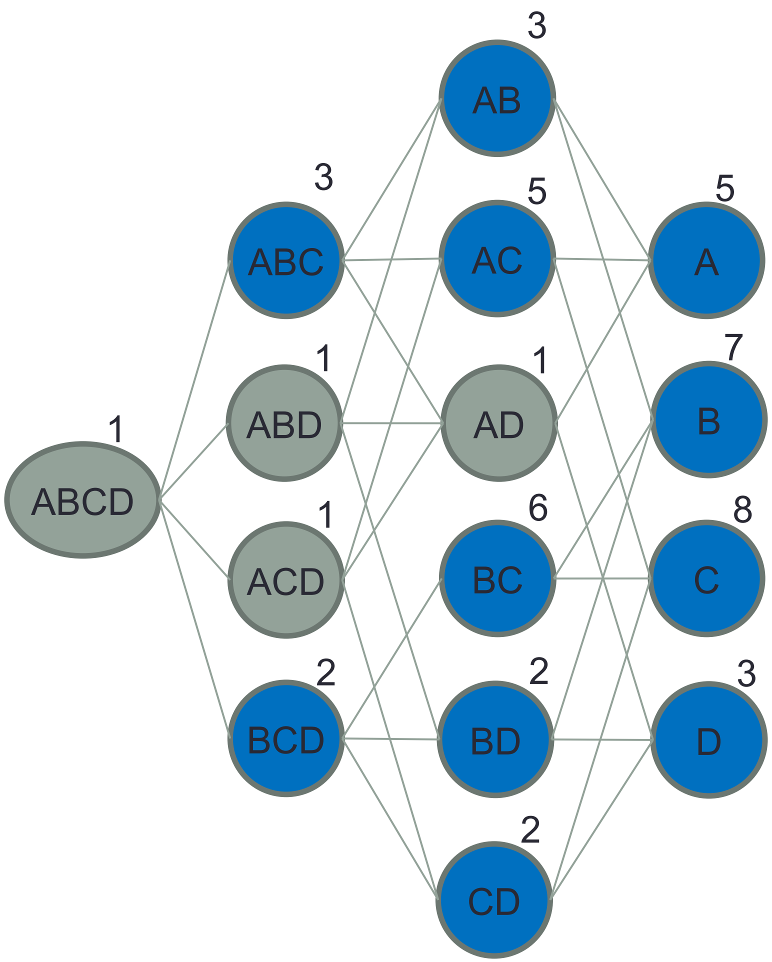
# Frequent Itemsets




Find the frequent itemsets.

Minimum Support = 2

# Frequent Itemsets




Find the frequent itemsets.

Minimum Support = 2

# Computational Complexity of Frequent Itemset Mining

- How many itemsets are potentially to be generated in the worst case?
  - Senstive to the min_sup threshold
  - When min_sup is low, there exist potentially an exponential number of frequent itemsets
  - The worst case: $M^N$ where M = # distinct items, and N = max length of transactions
- The worst case complexity vs. the expected probability
  - Ex. Suppose Walmart has $10^4$ kinds of products
    - The chance to pick up one product $10^{-4}$
    - The chance to pick up a particular set of 10 products: $\sim 10^{-40}$
    - What is the chance this particular set of 10 products to be frequent $10^3$ times in $10^9$ transactions?

## Scalable Frequent Itemset Mining Methods

- Apriori: A Candidate Generation-and-Test Approach

- Improving the Efficiency of Apriori
  - FPGrowth:  A Frequent Pattern-Growth Approach
  - ECLAT: Frequent Pattern Mining with Vertical Data Format

# The Downward Closure Property and Scalable Mining Methods

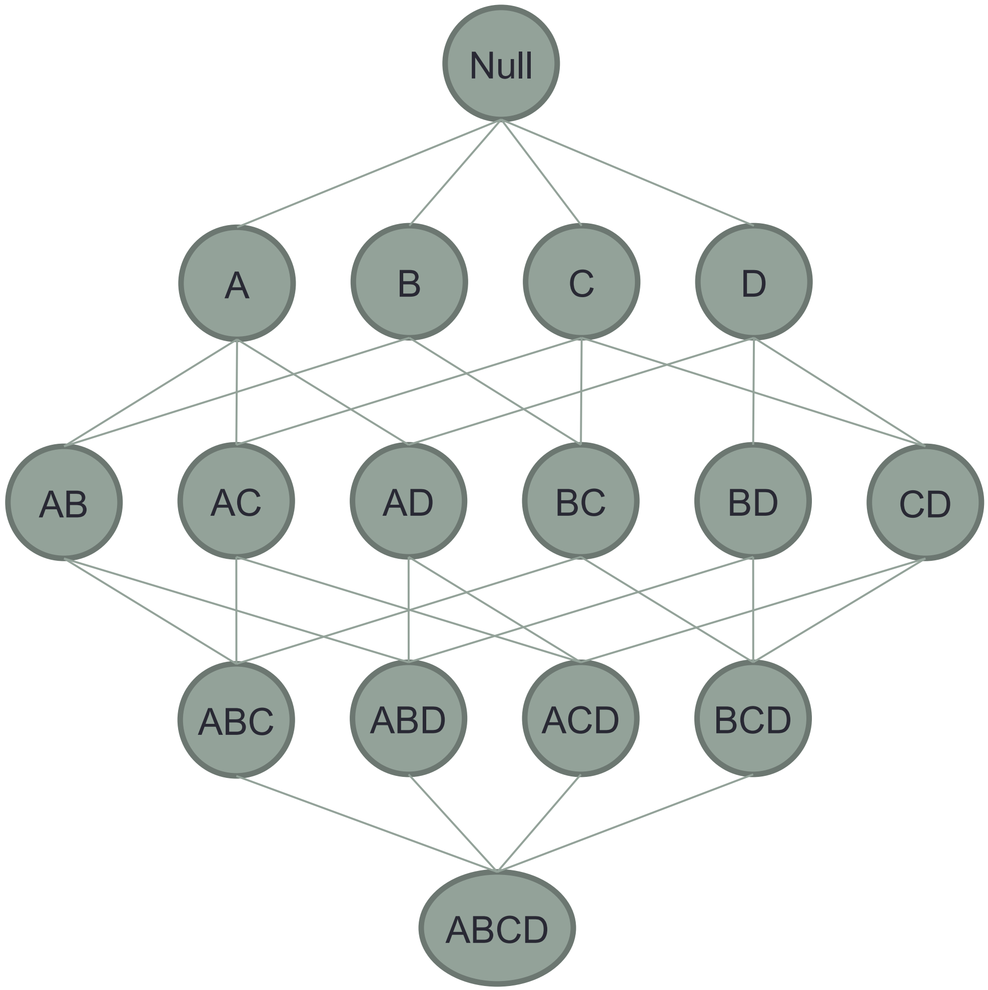
- The downward closure property of frequent patterns
  - <u>Any subset of a frequent itemset must be frequent</u>
  - If {beer, diaper, chips} is frequent, so is {beer, diaper} and {beer, chips}
- Scalable mining methods: Three major approaches
  - Apriori (Agrawal & Srikant@VLDB'94)
  - Freq. pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD'00)
  - Vertical data format approach (Charm—Zaki & Hsiao @SDM'02)

# Apriori: A Candidate Generation & Test Approach

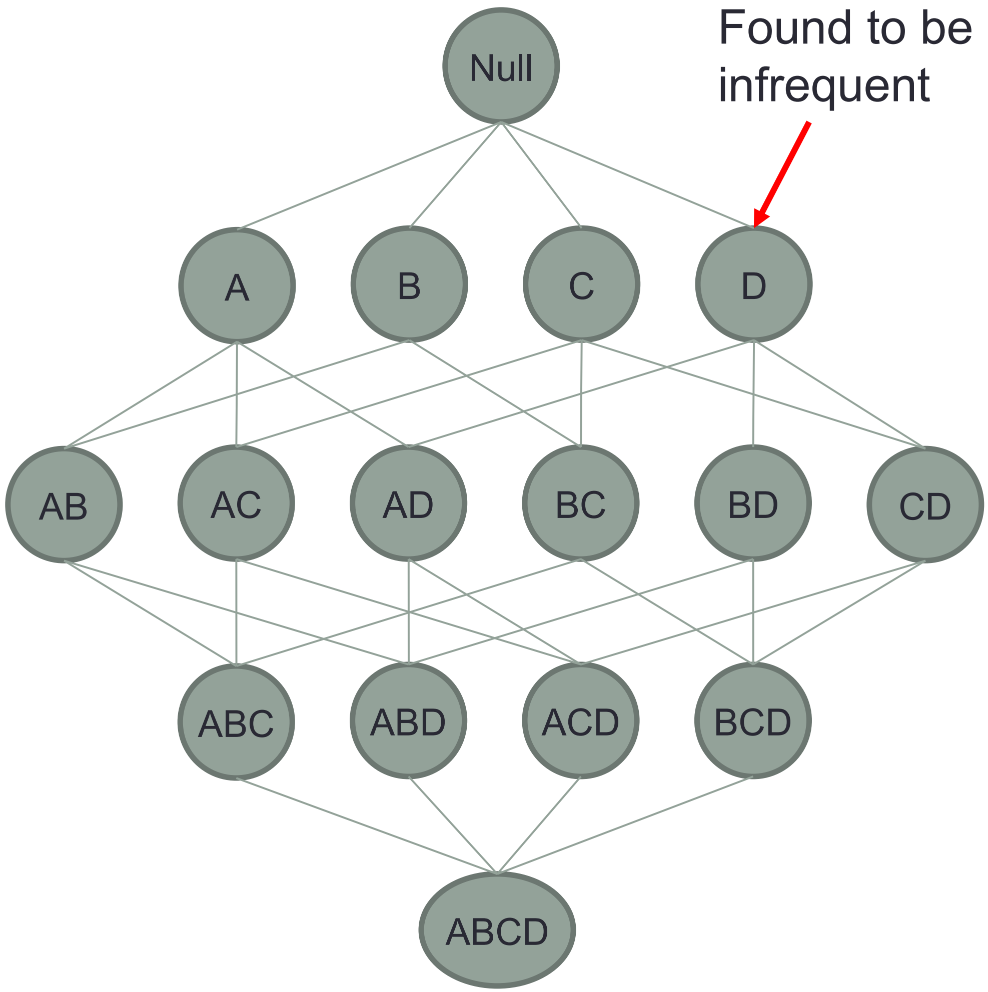
- Apriori pruning principle: If there is any itemset which is infrequent, its superset should not be generated/tested! (Agrawal & Srikant @VLDB'94, Mannila, et al. @ KDD' 94)

- Method:
  - Initially, scan DB once to get frequent 1-itemset
  - Generate length (k+1) candidate itemsets from length k frequent itemsets (self join)
  - Test the candidates against DB
  - Terminate when no frequent or candidate set can be generated

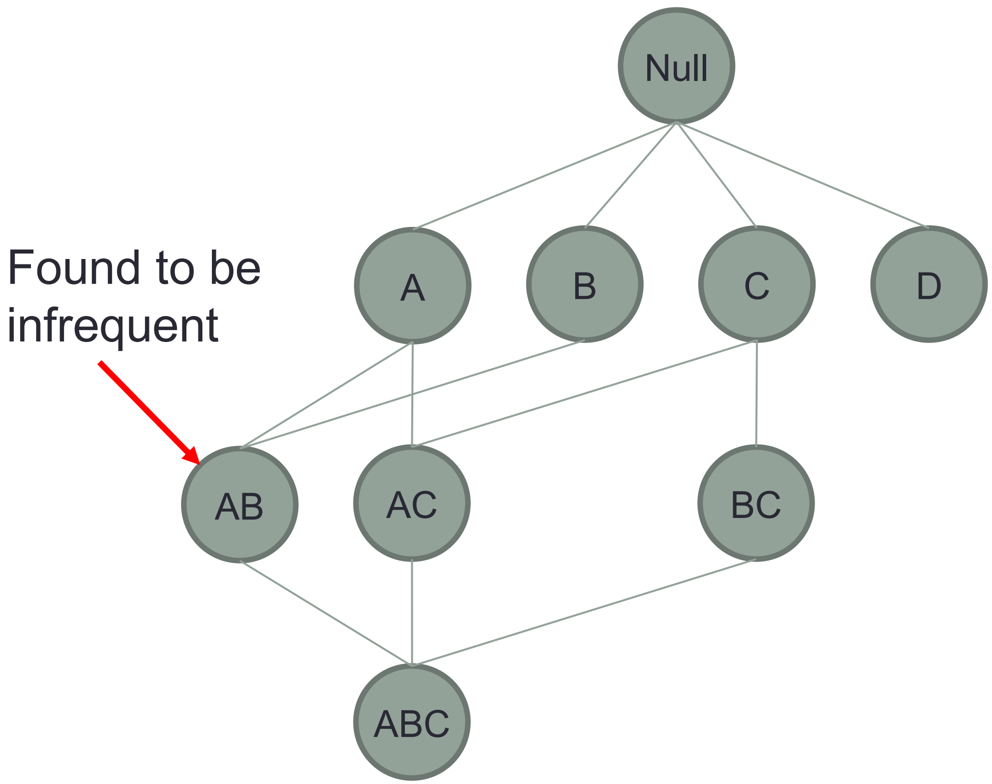
# Illustration of Apriori Principle



# Illustration of Apriori Principle



# Illustration of Apriori Principle



# Apriori Principle Itemset Example

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

Items (1-itemsets)

Minimum Support = 3

| 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 Cola or Eggs)

Triplets (3-itemsets)

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

If every subset is considered,

$${}^{6}C_{1} + {}^{6}C_{2} + {}^{6}C_{3} = 41$$

With support-based pruning,

$$6 + 6 + 1 = 13$$

# The Apriori Algorithm

$C_k$: Candidate itemset of size k

$L_k$ : frequent itemset of size k

$L_1$ = {frequent items};

**for** ($k$ = 1; $L_k$ !=$\emptyset$; $k$++) **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** $\cup_k L_k$;

# Implementation of Apriori

- How to generate candidates?
  - Step 1: self-joining $L_k$
  - Step 2: pruning
- Example of Candidate-generation
  - $L_3$={*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*}

# The Apriori Algorithm Illustration

$Sup_{min} = 2$

Database TDB

| Tid | Items |
|---|---|
| 10 | A, C, D |
| 20 | B, C, E |
| 30 | A, B, C, E |
| 40 | B, E |

1st scan →

$C_1$

| Itemset | sup |
|---|---|
| {A} | 2 |
| {B} | 3 |
| {C} | 3 |
| {D} | 1 |
| {E} | 3 |

→ $L_1$

| Itemset | sup |
|---|---|
| {A} | 2 |
| {B} | 3 |
| {C} | 3 |
| {E} | 3 |

Self Join

$C_2$

| Itemset |
|---|
| {A, B} |
| {A, C} |
| {A, E} |
| {B, C} |
| {B, E} |
| {C, E} |

2nd scan ←

$C_2$

| Itemset | sup |
|---|---|
| {A, B} | 1 |
| {A, C} | 2 |
| {A, E} | 1 |
| {B, C} | 2 |
| {B, E} | 3 |
| {C, E} | 2 |

Self Join ←

$L_2$

| Itemset | sup |
|---|---|
| {A, C} | 2 |
| {B, C} | 2 |
| {B, E} | 3 |
| {C, E} | 2 |

$C_3$

| Itemset |
|---|
| {B, C, E} |

3rd scan →

$L_3$

| Itemset | sup |
|---|---|
| {B, C, E} | 2 |

# The Apriori Algorithm (Exercise)

- Demonstrate the Apriori algorithm on the following dataset to find frequent itemsets
- Absolute Min_sup = 3

Mango – M
Onion – O
Nintendo – N
Key-chain – K
Eggs – E
Yo-yo – Y
Doll – D
Apple – A
Umbrella – U
Corn – C
Ice-cream - I

| Transaction ID | Items Bought |
|---|---|
| T1 | {M, O, N, K, E, Y } |
| T2 | {D, O, N, K, E, Y } |
| T3 | {M, A, K, E} |
| T4 | {M, U, C, K, Y } |
| T5 | {C, O, O, K, I, E} |

# Bottlenecks of Apriori

- Candidate generation can result in huge candidate sets:
  - $10^4$ frequent 1-itemset will generate $10^7$ candidate 2-itemsets
  - To discover a frequent pattern of size 100, e.g., $\{a_1, a_2, \ldots, a_{100}\}$, one needs to generate $2^{100} \sim 10^{30}$ candidates.

- Multiple scans of database:
  - Needs (n +1 ) scans, n is the length of the longest pattern

# How to Count Supports of Candidates

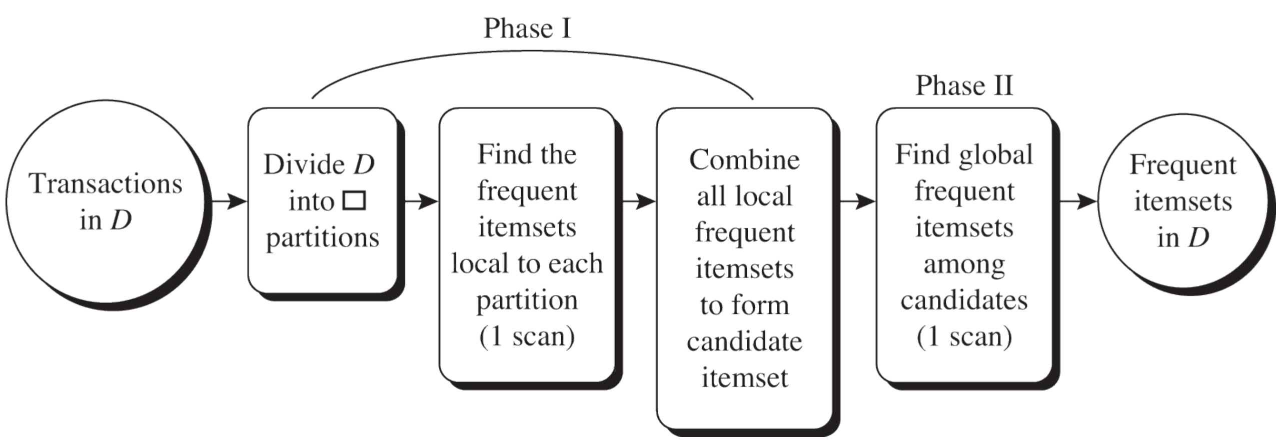
- Why counting supports of candidates a problem?
  - The total number of candidates can be very huge
  - One transaction may contain many candidates
- Method:
  - ✓ Hash Table
  - Remove buckets that are below the support threshold

$H_2$

| bucket address | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| bucket count | 2 | 2 | 4 | 2 | 2 | 4 | 4 |
| bucket contents | {I1, I4}<br>{I3, I5} | {I1, I5}<br>{I1, I5} | {I2, I3}<br>{I2, I3}<br>{I2, I3}<br>{I2, I3} | {I2, I4}<br>{I2, I4} | {I2, I5}<br>{I2, I5} | {I1, I2}<br>{I1, I2}<br>{I1, I2}<br>{I1, I2} | {I1, I3}<br>{I1, I3}<br>{I1, I3}<br>{I1, I3} |

# How to Count Supports of Candidates

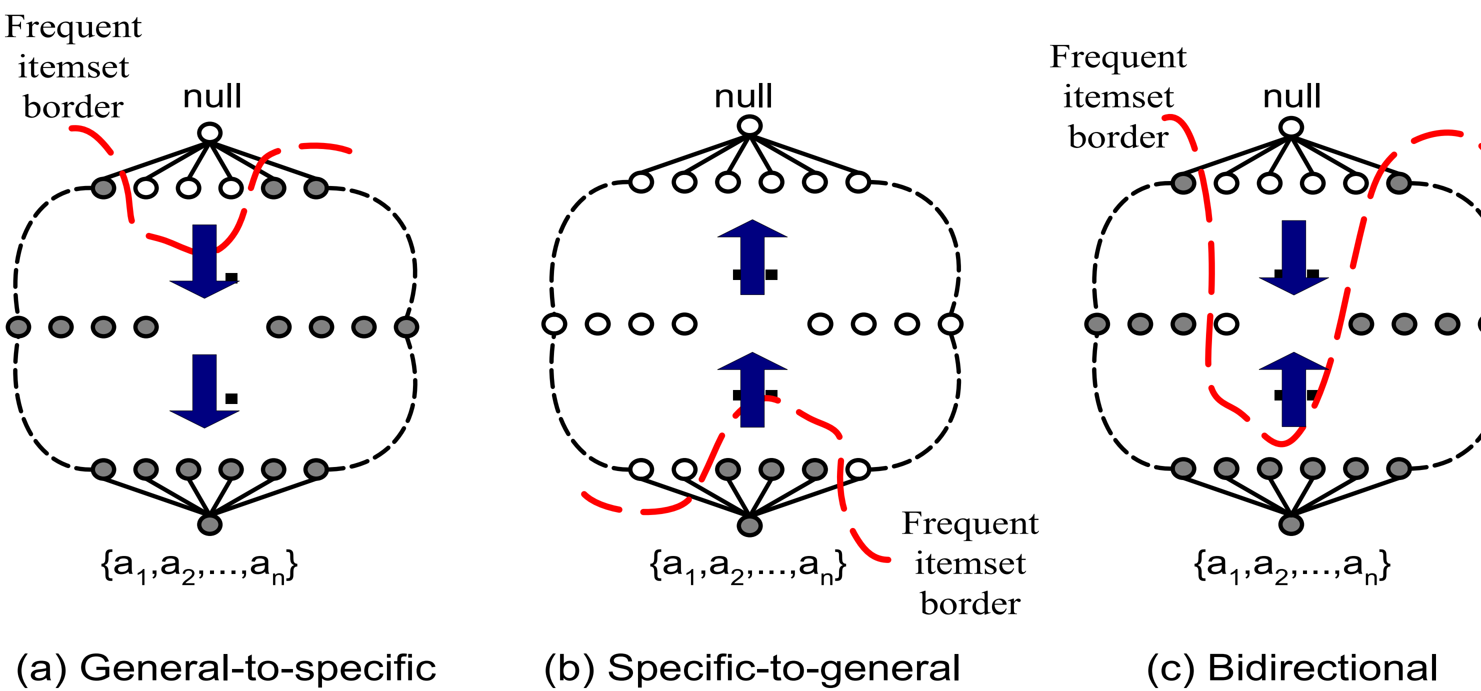
- Transaction Reduction
  - A transaction that does not contain any frequent k-itemsets cannot contain any frequent (k+1)-itemsets
  - Such a transaction can be removed from further consideration
- Partitioning
  - Partition transactions into non-overlapping partitions




- Sampling
- Dynamic Itemset Counting (adds itemsets to frequent itemset count dynamically)

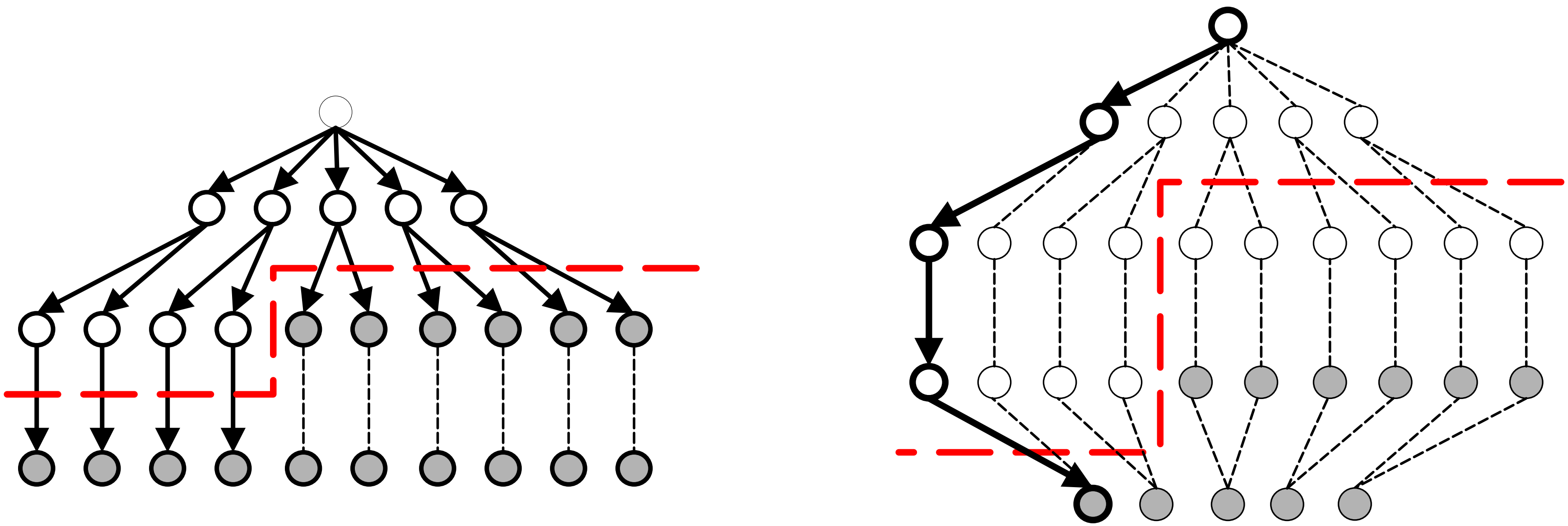
# Apriori: Alternative Search Methods

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



# Apriori: Alternative Search Methods

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



(a) Breadth first

(b) Depth first

# FP-growth

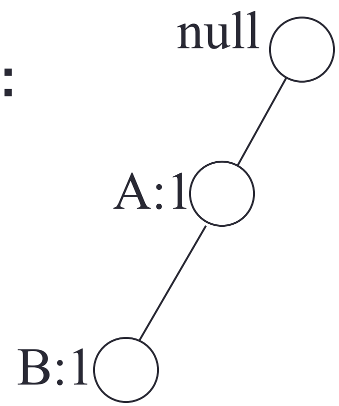
- Use a compressed representation of the database using an FP-tree (FP = Frequent Pattern)

- Once an FP-tree has been constructed, it uses a recursive divide-and-conquer approach to mine the frequent itemsets

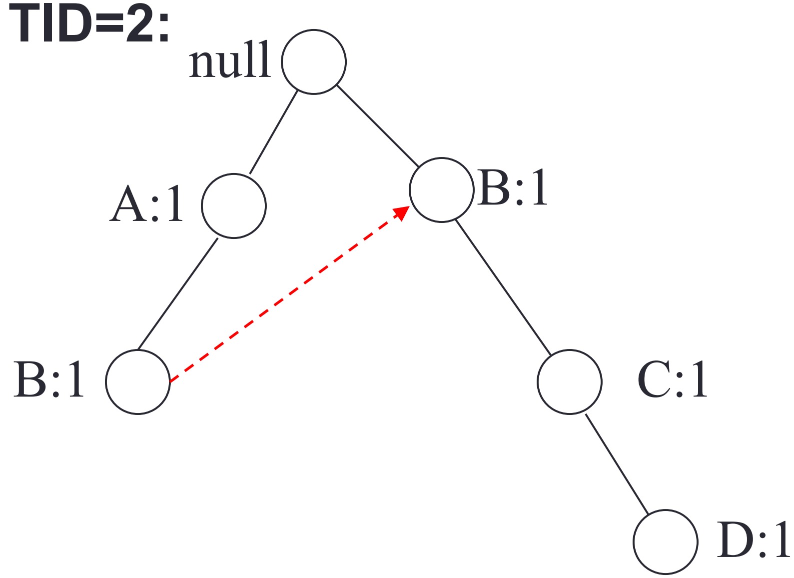
# FP-Tree Construction

| TID | Items |
|---|---|
| 1 | {A,B} |
| 2 | {B,C,D} |
| 3 | {A,C,D,E} |
| 4 | {A,D,E} |
| 5 | {A,B,C} |
| 6 | {A,B,C,D} |
| 7 | {B,C} |
| 8 | {A,B,C} |
| 9 | {A,B,D} |
| 10 | {B,C,E} |

**After reading TID=1:**



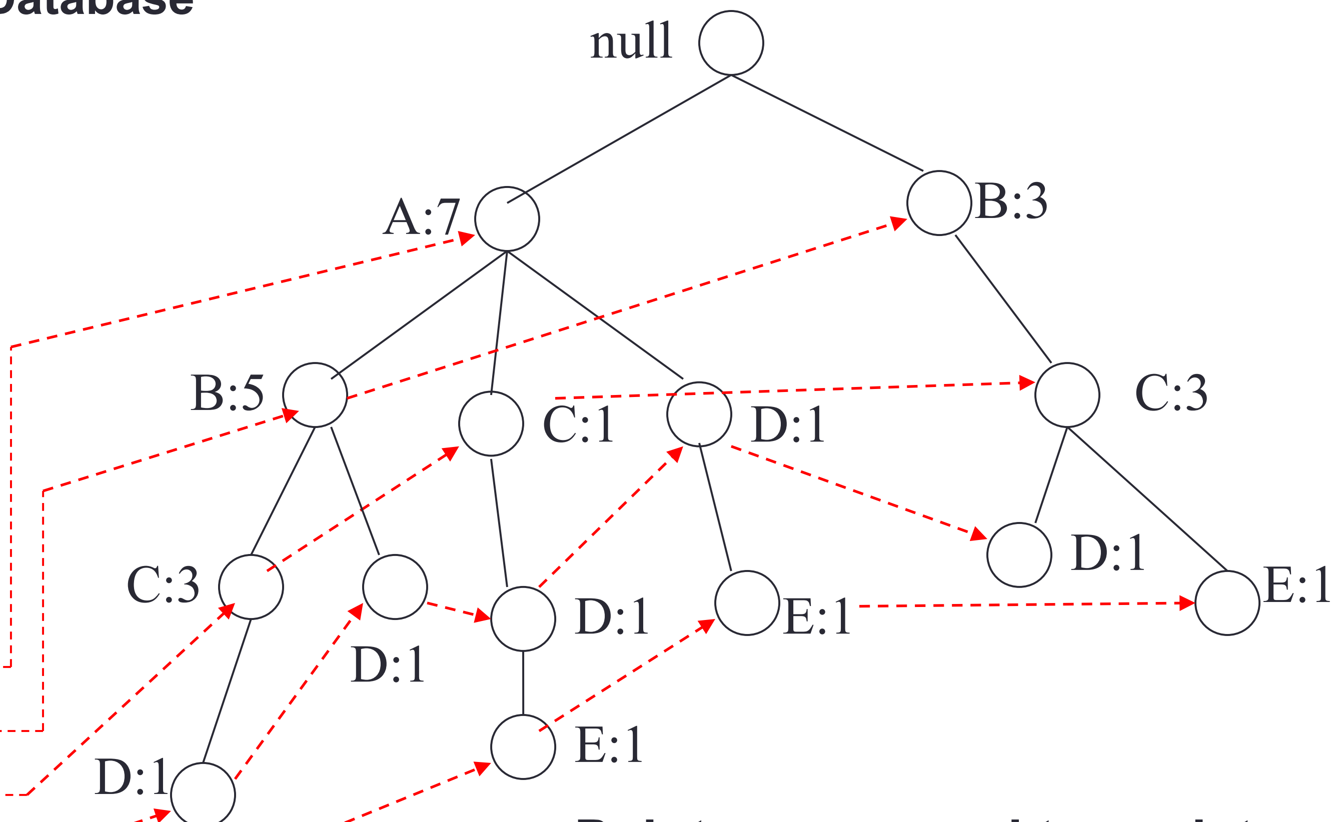

**After reading TID=2:**



# FP-Tree Construction

**Transaction Database**

| TID | Items |
|---|---|
| 1 | {A,B} |
| 2 | {B,C,D} |
| 3 | {A,C,D,E} |
| 4 | {A,D,E} |
| 5 | {A,B,C} |
| 6 | {A,B,C,D} |
| 7 | {B,C} |
| 8 | {A,B,C} |
| 9 | {A,B,D} |
| 10 | {B,C,E} |

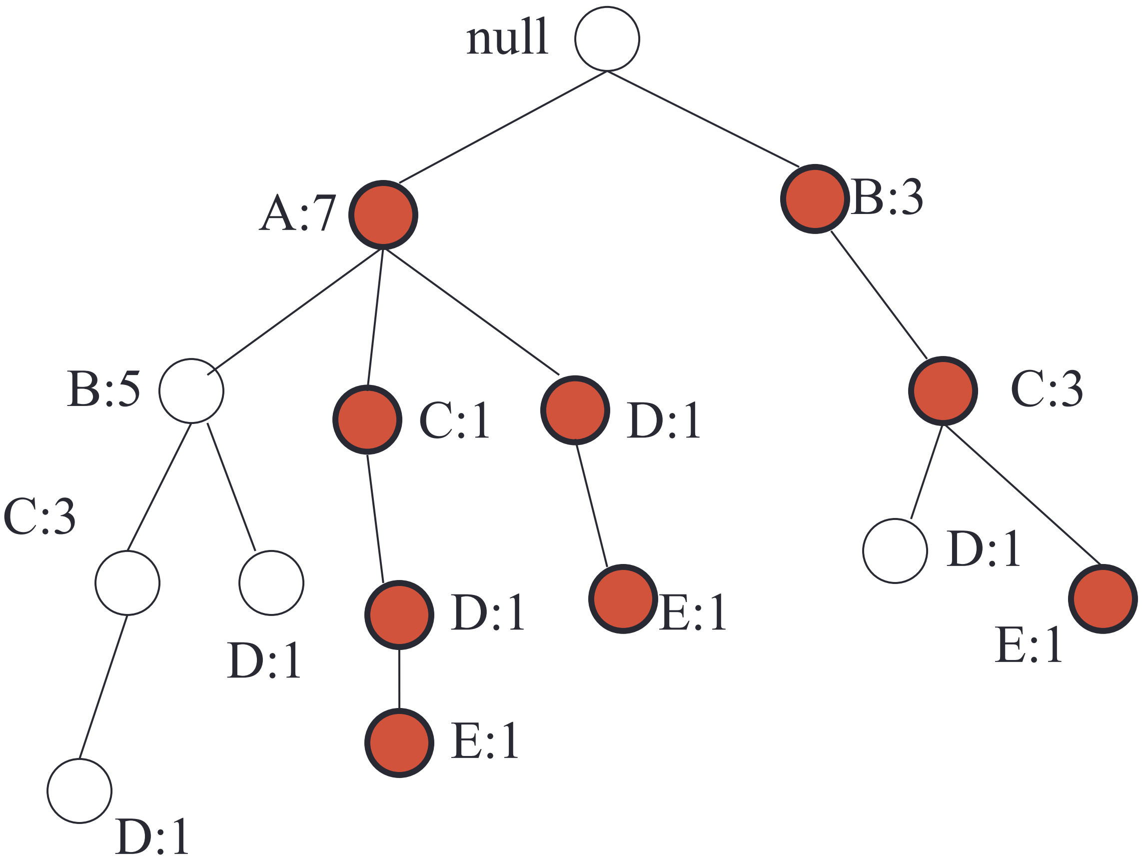



**Header table**

| Item | Pointer |
|---|---|
| A | |
| B | |
| C | |
| D | |
| E | |

**Pointers are used to assist frequent itemset generation**

# FP-Growth



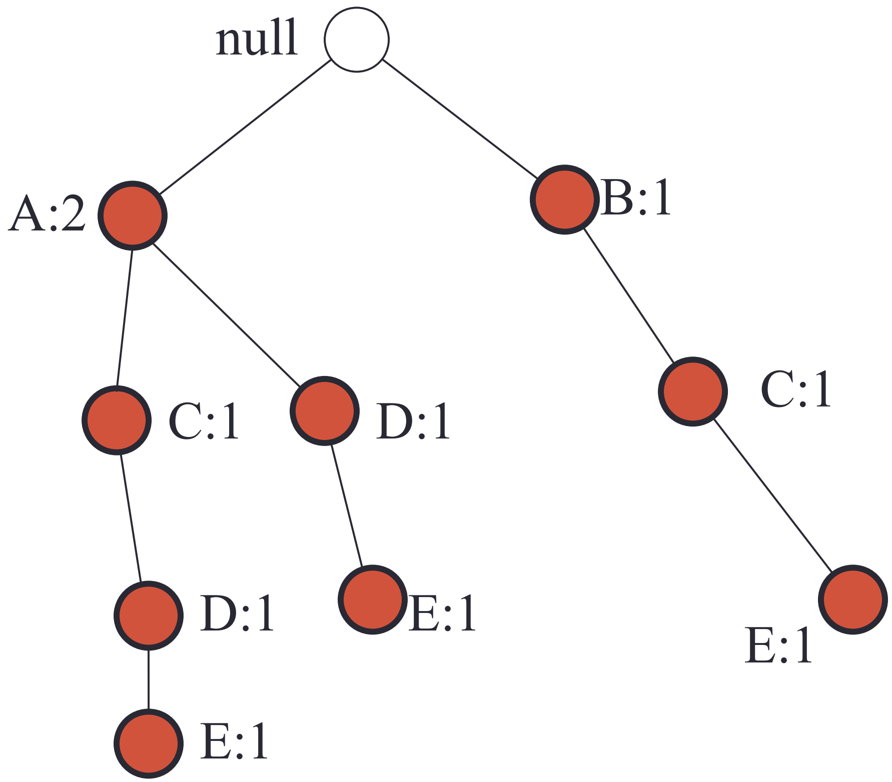

**Build conditional pattern base for E:**

**P = {(A:1,C:1,D:1),**
**(A:1,D:1),**
**(B:1,C:1)}**

**Recursively apply FP-growth on P**

# FP-Growth

<u>**Conditional tree for E:**</u>




**Conditional Pattern base for E:**

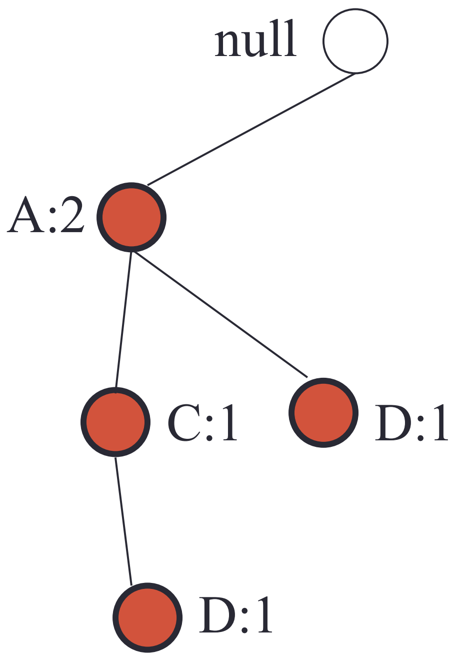
**P = {(A:1,C:1,D:1,E:1),**
**(A:1,D:1,E:1),**
**(B:1,C:1,E:1)}**

**Count for E is 3: {E} is frequent itemset**

**Recursively apply FP-growth on P**

# FP-Growth

**Conditional tree for D within conditional tree for E:**




**Conditional pattern base for D within conditional base for E:**

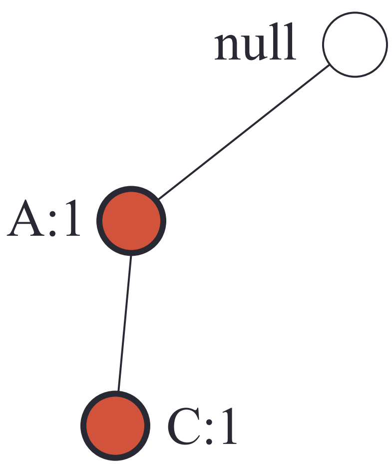
P = {(A:1,C:1,D:1),
(A:1,D:1)}

**Count for D is 2: {D,E} is frequent itemset**

**Recursively apply FP-growth on P**

# FP-Growth

**Conditional tree for C within D within E:**



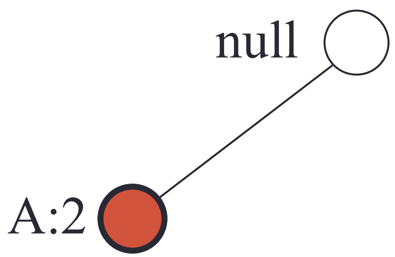

**Conditional pattern base for C within D within E:**

**P = {(A:1,C:1)}**

**Count for C is 1: {C,D,E} is NOT frequent itemset**

# FP-Growth

**Conditional tree for A within D within E:**




**Count for A is 2: {A,D,E} is frequent itemset**

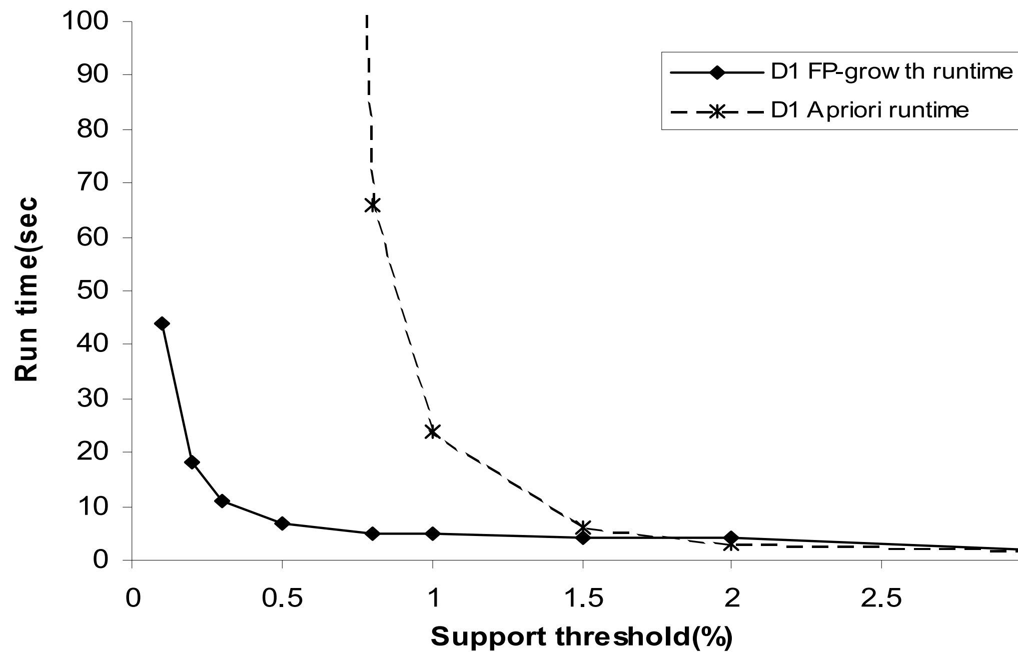
**Next step:**

**Construct conditional tree C within conditional tree E**

**Continue until exploring conditional tree for A (which has only node A)**

# Benefits of the FP-Tree Structure

- Performance study shows
  - FP-growth is an order of magnitude faster than Apriori
- Reasoning
  - No candidate generation, no candidate test
  - Use compact data structure
  - Eliminate repeated database scan
  - Basic operation is counting and FP-tree building



# ECLAT: Another Method for Frequent Itemset Generation

- ECLAT: for each item, store a list of transaction ids (tids); 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 | | | | |

**TID-list**

# ECLAT: Another Method for Frequent Itemset Generation

- Determine support of any k-itemset by intersecting tid-lists of two of its (k-1) subsets.

| A |
|---|
| 1 |
| 4 |
| 5 |
| 6 |
| 7 |
| 8 |
| 9 |

$\cap$

| B |
|---|
| 1 |
| 2 |
| 5 |
| 7 |
| 8 |
| 10 |

$\rightarrow$

| AB |
|---|
| 1 |
| 5 |
| 7 |
| 8 |

- 3 traversal approaches:
  - top-down, bottom-up and hybrid
- Advantage: very fast support counting
- Disadvantage: intermediate tid-lists may become too large for memory

# Complexity of Association Rule Mining

- 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)

# Mining Association Rules

**Two-step** process

1. Find all frequent itemsets, where itemset frequency is beyond min_sup; **done**!

2. From list of frequent itemsets, generate association rules satisfying min_sup and confidence

# 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 {O,K,E} is a frequent itemset, candidate rules:
    {O,K} →{E}, {O,E} →{K}, {K,E} →{O}, {K} → {O,E}, {E} → {O,K}, {O} → {K,E}, {O} → {K}, {O} → {E}, {K} → {O}, {K} → {E}, {E} → {O}, {E} → {K}

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

# Confidence and Association Rules

*IF Body then Consequent*

*Body ==> Consequent* [ *Support* , *Confidence* ]

Confidence (A ==> B) = P(B|A) = *support_count*(AυB)/*support_count*(A)

| Transaction ID | Items Bought |
|---|---|
| T1 | {M, O, N, K, E, Y } |
| T2 | {D, O, N, K, E, Y } |
| T3 | {M, A, K, E} |
| T4 | {M, U, C, K, Y } |
| T5 | {C, O, O, K, I, E} |

Evaluate the following rules based on confidence:

{O,K} →{E}

{O,E} →{K}

{K,E} →{O}

{K} → {O,E}

{E} → {O,K}

{O} → {K,E}

# Rule Generation

- How to efficiently generate rules from frequent itemsets?
  - In general, confidence does not have an anti-monotone property (downward closure)

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

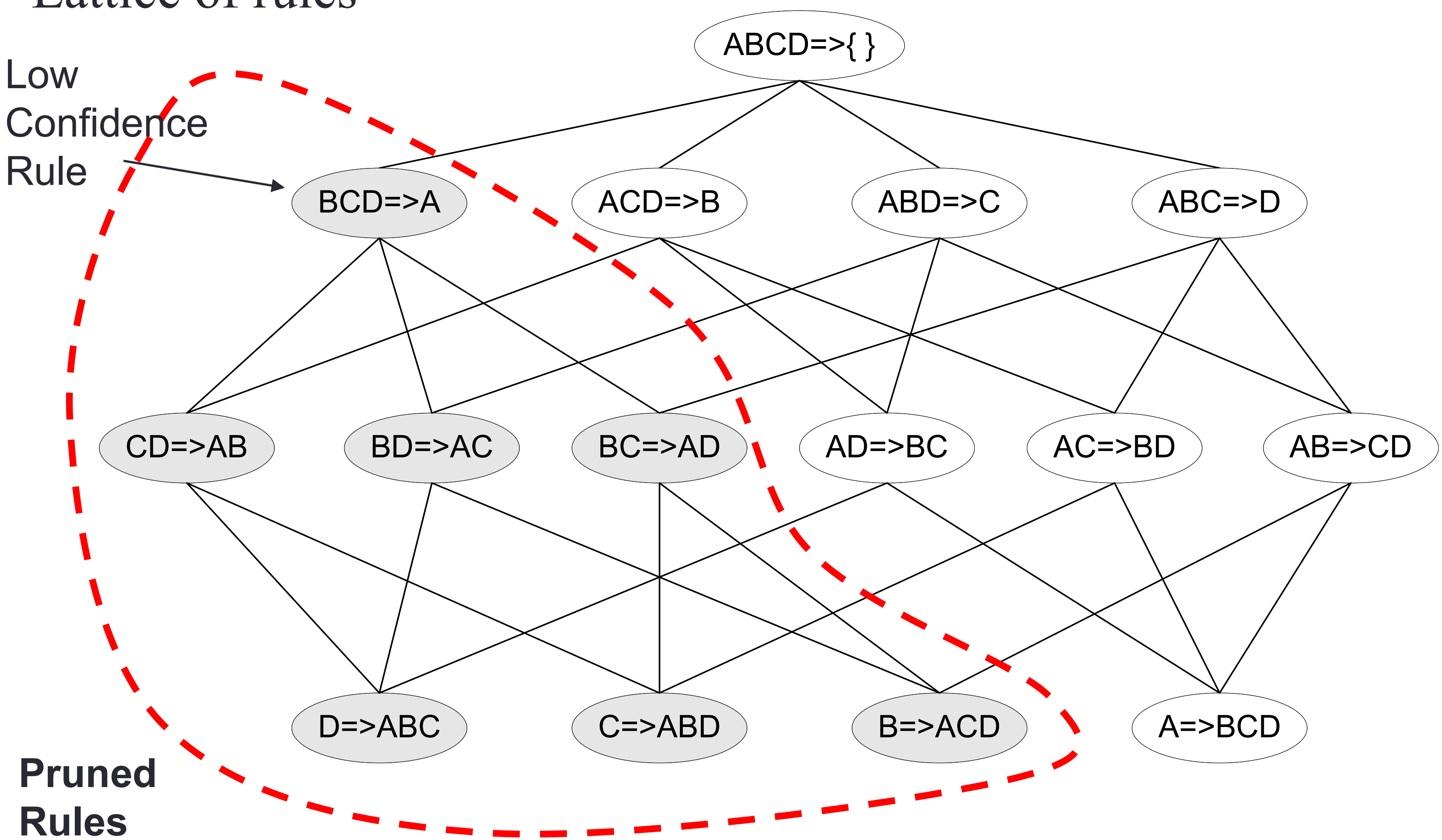
  - But confidence of rules generated from the **same** itemset has an anti-monotone property
  - e.g., L = {A,B,C,D}:

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

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

# Rule Generation

Lattice of rules



# Information-Theoretic Approach: Generalized Rule Induction Method

- Association rules well-suited to using Boolean and categorical attributes
- However, how are numeric attributes handled?
- *Apriori* not equipped to handle numeric inputs
- One possible solution discretizes numeric variables during pre-processing
- For example, *income* discretized to “low”, “medium”, and “high”

- Discretizing variables often leads to loss of information
- Generalized Rule Induction (GRI) offers alternate method for mining association rules
- GRI works with Boolean, categorical, and numeric input

# Information-Theoretic Approach: Generalized Rule Induction Method

- GRI introduced by Smyth and Goodman in 1992
- Method does not use frequent itemsets
- Instead, GRI applies information-theoretic approach
- Method measures “interestingness” of candidate rules

- J-Measure
  - GRI applies J-Measure:

$$J = p(x)\left[ p(y|x)\ln\frac{p(y|x)}{p(y)} + [1 - p(y|x)]\ln\frac{1 - p(y|x)}{1 - p(y)} \right]$$

  - p(*x*) represents probability of the observed value of *x*
  - Measures prevalence of observed value for antecedent
  - Where rules have more than one antecedent, p(*x*) is conjunction of variable values in antecedent

Discovering Knowledge in Data: An Introduction to Da
ta Mining, Second Edition, by Daniel Larose and Chant
al Larose, John Wiley and Sons, Inc., 2014.

# Information-Theoretic Approach: Generalized Rule Induction Method

$$J = p(x)\left[ p(y|x) \ln \frac{p(y|x)}{p(y)} + [1 - p(y|x)] \ln \frac{1 - p(y|x)}{1 - p(y)} \right]$$

- p(*y*) represents prior probability of the observed value of *y*
- Measures prevalence of observed value for consequent

- p(*y*|*x*) equals conditional probability (posterior confidence) of *y*, given *x* has occurred
- Measures probability of the observed value of *y*, given value of *x* has occurred
- Measured directly by confidence of rule

- ln is natural logarithm (log to the base *e*)

# Information-Theoretic Approach: Generalized Rule Induction Method

- As before, analyst specifies <u>minimum confidence</u> and <u>support levels</u>
- Using GRI, number of association rules to report also specified
- Number of rules defines size of <u>Rule Table</u>, referenced internally by algorithm

- First, GRI generates single-antecedent rules
- For each rule, the value for *J* (the *J-Measure*) is computed
- If value of *J* exceeds current minimum *J* in rule table, new rule inserted in rule table, displacing rule associated with current minimum *J*
- In this way, rule table remains at constant size
- Next, rules with multiple antecedents are considered

# Information-Theoretic Approach: Generalized Rule Induction Method

- Behavior of *J-Measure* described:
- Higher values of *J* are associated with higher values of p(*x*)
- Rules favored whose antecedent value more prevalent
- Reflects higher coverage in data set

- *J* tends toward higher values when p(*y*) and p(*y*|*x*) near 0 or 1
- That is, rules favored where p(*y*) or p(*y*|*x*) more extreme

# Information-Theoretic Approach: Generalized Rule Induction Method

- Also, *J-Measure* favors rules with either very high or very low confidence
- Why would rules with very low confidence be mined?
- For example, rule *R* = "If buy beer, then buy fingernail polish", with confidence $p(y|x) = 0.01\%$
- By definition, *J-Measure* favors rule *R* (extremely low confidence)
- Alternately, consider negative form of *R* = "If buy beer, then NOT buy fingernail polish" with confidence = 99.99%
- Although negative rules sometimes interesting, results often not directly actionable

# Interestingness/usefulness

- Not all strong rules are interesting
  - Confidence can be deceiving
  - Does not measure **strength** (or lack of strength) of correlation in a rule

- Measures of ***correlation*** can be used in combination with support and confidence

- Correlation rules:
  - A=>B[support, confidence, correlation]

- Interestingness of the rule is measured based on **support**, **confidence**, and **correlation**

# Correlation Measures

- $\chi^2$
  - $\chi^2$ >1 then correlated
  - If expected value is greater than actual value: negatively correlated
  - If expected value is less than actual value: positively correlated

$$\chi^2 = \sum \frac{(\text{observed} - \text{expected})^2}{\text{expected}}$$

- Lift
  - If the occurrence of A is independent from the occurrence of B, lift = 1
  - Lift < 1 negative correlation
  - Lift > 1 positive correlation

$$lift = \frac{\sup(A \cup B)}{\sup(A) * \sup(B)}$$

# Interestingness Measure: Correlations (Lift and $\chi^2$)

| | Basketball | Not basketball | Sum (row) |
|---|---|---|---|
| Cereal | 2000 | 1750 | 3750 |
| Not cereal | 1000 | 250 | 1250 |
| Sum(col.) | 3000 | 2000 | 5000 |

$$\chi^2 = \sum \frac{(\text{observed} - \text{expected})^2}{\text{expected}} =$$

$$\frac{(2000-2250)^2}{2250} + \frac{(1750-1500)^2}{1500} + \frac{(1000-750)^2}{750} + \frac{(250-500)^2}{500} = 277.78$$

- *Negative correlation between playing basketball and eating cereal as shown by chi-square and lift values*
- *play basketball* $\Rightarrow$ *eat cereal* [40%, 66.7%] is misleading
  - The overall % of students eating cereal is 75% > 66.7%.
- *play basketball* $\Rightarrow$ *not eat cereal* [20%, 33.3%] is more interesting, although with lower support and confidence

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

$$lift(B, C) = \frac{2000/5000}{3000/5000 * 3750/5000} = 0.89$$

$$lift(B, \neg C) = \frac{1000/5000}{3000/5000 * 1250/5000} = 1.33$$

# Measures of Interestingness

| symbol | measure | range | formula |
|---|---|---|---|
| $\phi$ | $\phi$-coefficient | -1...1 | $\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$ |
| $Q$ | Yule's Q | -1 ... 1 | $\frac{P(A,B)P(\overline{A},\overline{B})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{A},\overline{B})+P(A,\overline{B})P(\overline{A},B)}$ |
| $Y$ | Yule's Y | -1 ... 1 | $\frac{\sqrt{P(A,B)P(\overline{A},\overline{B})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{A},\overline{B})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}}$ |
| $k$ | Cohen's | -1 ... 1 | $\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$ |
| $PS$ | Piatetsky-Shapiro's | -0.25 ... 0.25 | $P(A,B)-P(A)P(B)$ |
| $F$ | Certainty factor | -1 ... 1 | $\max(\frac{P(B\mid A)-P(B)}{1-P(B)}, \frac{P(A\mid B)-P(A)}{1-P(A)})$ |
| $AV$ | added value | -0.5 ... 1 | $\max(P(B\mid A)-P(B), P(A\mid B)-P(A))$ |
| $K$ | Klosgen's Q | -0.33 ... 0.38 | $\sqrt{P(A,B)}\max(P(B\mid A)-P(B), P(A\mid B)-P(A))$ |
| $g$ | Goodman-kruskal's | 0 ... 1 | $\frac{\Sigma_j \max_k P(A_j,B_k)+\Sigma_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-max_k P(B_k)}$ |
| $M$ | Mutual Information | 0 ... 1 | $\frac{\Sigma_i\Sigma_j P(A_i,B_j)\log\frac{P(A_i,B_j)}{P(A_i)P(B_J)}}{\min(-\Sigma_i P(A_i)\log P(A_i)\log P(A_i), -\Sigma_i P(B_i)\log P(B_i)\log P(B_i))}$ |
| $J$ | J-Measure | 0 ... 1 | $\max(P(A,B)\log(\frac{P(B\mid A)}{P(B)})+P(A\overline{B})\log(\frac{P(\overline{B}\mid A)}{P(\overline{B})}))$ <br> $P(A,B)\log(\frac{P(A\mid B)}{P(A)})+P(\overline{A}B)\log(\frac{P(\overline{A}\mid B)}{P(\overline{A})})$ |
| $G$ | Gini index | 0 ... 1 | $\max(P(A)[P(B\mid A)^2+P(\overline{B}\mid A)^2]+P(\overline{A}[P(B\mid\overline{A})^2+P(\overline{B}\mid\overline{A})^2]-P(B)^2-P(\overline{B})^2,$ <br> $P(B)[P(A\mid B)^2+P(\overline{A}\mid B)^2]+P(\overline{B}[P(A\mid\overline{B})^2+P(\overline{A}\mid\overline{B})^2]-P(A)^2-P(\overline{A})^2)$ |
| $s$ | support | 0 ... 1 | $P(A,B)$ |
| $c$ | confidence | 0 ... 1 | $max(P(B\mid A), P(A\mid B))$ |
| $L$ | Laplace | 0 ... 1 | $\max(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2})$ |
| $IS$ | Cosine | 0 ... 1 | $\frac{P(A,B)}{\sqrt{P(A)P(B)}}$ |
| $\gamma$ | coherence(Jaccard) | 0 ... 1 | $\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$ |
| $\alpha$ | all_confidence | 0 ... 1 | $\frac{P(A,B)}{\max(P(A),P(B))}$ |
| $o$ | odds ratio | 0 ... $\infty$ | $\frac{P(A,B)P(\overline{A},\overline{B})}{P(\overline{A},B)P(A,\overline{B})}$ |
| $V$ | Conviction | 0.5 ... $\infty$ | $\max(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})})$ |
| $\lambda$ | lift | 0 ... $\infty$ | $\frac{P(A,B)}{P(A)P(B)}$ |
| $S$ | Collective strength | 0 ... $\infty$ | $\frac{P(A,B)+P(\overline{A}\overline{B})}{P(A)P(B)+P(\overline{A})P(\overline{B})}\times\frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{A}\overline{B})}$ |
| $\chi^2$ | $\chi^2$ | 0 ... $\infty$ | $\Sigma_i\frac{(P(A_i)-E_i)^2}{E_i}$ |

# *Lift*, $\chi^2$ and Null Transactions

- Null transactions are transactions that contain none of the items in the rule.
- Null transactions can outweigh the number of individual associations because many transactions may not have any of the items of interest
- *Lift* and $\chi^2$ are both sensitive to null transactions
  - They can't distinguish interesting pattern association relationships because they are both strongly influenced by null transactions
- Both measures are sensitive to *n*
- It is desirable to have a measure that removes the influence of null transactions
- Null-invariant measures

# Null-Invariant Measures

Table 6: Properties of interestingness measures. Note that none of the measures satisfies all the properties.

| Symbol | Measure | Range | P1 | P2 | P3 | O1 | O2 | O3 | O3' | O4 |
|---|---|---|---|---|---|---|---|---|---|---|
| $\phi$ | $\phi$-coefficient | $-1 \cdots 0 \cdots 1$ | Yes | Yes | Yes | Yes | No | Yes | Yes | No |
| $\lambda$ | Goodman-Kruskal's | $0 \cdots 1$ | Yes | No | No | Yes | No | No* | Yes | No |
| $\alpha$ | odds ratio | $0 \cdots 1 \cdots \infty$ | Yes* | Yes | Yes | Yes | Yes | Yes* | Yes | No |
| $Q$ | Yule's $Q$ | $-1 \cdots 0 \cdots 1$ | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No |
| $Y$ | Yule's $Y$ | $-1 \cdots 0 \cdots 1$ | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No |
| $\kappa$ | Cohen's | $-1 \cdots 0 \cdots 1$ | Yes | Yes | Yes | Yes | No | No | Yes | No |
| $M$ | Mutual Information | $0 \cdots 1$ | Yes | Yes | Yes | No** | No | No* | Yes | No |
| $J$ | J-Measure | $0 \cdots 1$ | Yes | No | No | No** | No | No | No | No |
| $G$ | Gini index | $0 \cdots 1$ | Yes | No | No | No** | No | No* | Yes | No |
| $s$ | Support | $0 \cdots 1$ | No | Yes | No | Yes | No | No | No | No |
| $c$ | Confidence | $0 \cdots 1$ | No | Yes | No | No** | No | No | No | Yes |
| $L$ | Laplace | $0 \cdots 1$ | No | Yes | No | No** | No | No | No | No |
| $V$ | Conviction | $0.5 \cdots 1 \cdots \infty$ | No | Yes | No | No** | No | No | Yes | No |
| $I$ | Interest | $0 \cdots 1 \cdots \infty$ | Yes* | Yes | Yes | Yes | No | No | No | No |
| $IS$ | Cosine | $0 \cdots \sqrt{P(A,B)} \cdots 1$ | No | Yes | Yes | Yes | No | No | No | Yes |
| $PS$ | Piatetsky-Shapiro's | $-0.25 \cdots 0 \cdots 0.25$ | Yes | Yes | Yes | Yes | No | Yes | Yes | No |
| $F$ | Certainty factor | $-1 \cdots 0 \cdots 1$ | Yes | Yes | Yes | No** | No | No | Yes | No |
| $AV$ | Added value | $-0.5 \cdots 0 \cdots 1$ | Yes | Yes | Yes | No** | No | No | No | No |
| $S$ | Collective strength | $0 \cdots 1 \cdots \infty$ | No | Yes | Yes | Yes | No | Yes* | Yes | No |
| $\zeta$ | Jaccard | $0 \cdots 1$ | No | Yes | Yes | Yes | No | No | No | Yes |
| $K$ | Klosgen's | $(\frac{2}{\sqrt{3}}-1)^{1/2}[2-\sqrt{3}-\frac{1}{\sqrt{3}}] \cdots 0 \cdots \frac{2}{3\sqrt{3}}$ | Yes | Yes | Yes | No** | No | No | No | No |

where:
- P1: $O(\mathbf{M}) = 0$ if $det(\mathbf{M}) = 0$, *i.e.*, whenever $A$ and $B$ are statistically independent.
- P2: $O(\mathbf{M_2}) > O(\mathbf{M_1})$ if $\mathbf{M_2} = \mathbf{M_1} + [k \ \ -k; \ -k \ \ k]$.
- P3: $O(\mathbf{M_2}) < O(\mathbf{M_1})$ if $\mathbf{M_2} = \mathbf{M_1} + [0 \ k; \ 0 \ -k]$ or $\mathbf{M_2} = \mathbf{M_1} + [0 \ 0; \ k \ -k]$.
- O1: Property 1: Symmetry under variable permutation.
- O2: Property 2: Row and Column scaling invariance.
- O3: Property 3: Antisymmetry under row or column permutation.
- O3': Property 4: Inversion invariance.
- O4: Property 5: Null invariance.
- Yes*: Yes if measure is normalized.
- No*: Symmetry under row or column permutation.
- No**: No unless the measure is symmetrized by taking $\max(M(A,B), M(B,A))$.

# Comparison of Interestingness Measures

- Null-(transaction) invariance is crucial for correlation analysis

| Measure | Definition | Range | Null-Invariant |
|---|---|---|---|
| $\chi^2(a,b)$ | $\sum_{i,j=0,1} \frac{(e(a_i,b_j)-o(a_i,b_j))^2}{e(a_i,b_j)}$ | $[0,\infty]$ | No |
| $Lift(a,b)$ | $\frac{P(ab)}{P(a)P(b)}$ | $[0,\infty]$ | No |
| $AllConf(a,b)$ | $\frac{sup(ab)}{max\{sup(a),sup(b)\}}$ | $[0,1]$ | Yes |
| $Coherence(a,b)$ | $\frac{sup(ab)}{sup(a)+sup(b)-sup(ab)}$ | $[0,1]$ | Yes |
| $Cosine(a,b)$ | $\frac{sup(ab)}{\sqrt{sup(a)sup(b)}}$ | $[0,1]$ | Yes |
| $Kulc(a,b)$ | $\frac{sup(ab)}{2}(\frac{1}{sup(a)}+\frac{1}{sup(b)})$ | $[0,1]$ | Yes |
| $MaxConf(a,b)$ | $max\{\frac{sup(ab)}{sup(a)},\frac{sup(ab)}{sup(b)}\}$ | $[0,1]$ | Yes |

Table 3. Interestingness measure definitions.

- Lift and $\chi^2$ are not null-invariant
- 5 null-invariant measures

| | Milk | No Milk | Sum (row) |
|---|---|---|---|
| Coffee | m, c | ~m, c | c |
| No Coffee | m, ~c | ~m, ~c | ~c |
| Sum(col.) | m | ~m | Σ |

Null-transactions w.r.t. m and c

Kulczynski measure (1927)

Null-invariant

| Data set | $mc$ | $\overline{m}c$ | $m\overline{c}$ | $\overline{m}\overline{c}$ | $\chi^2$ | $Lift$ | $AllConf$ | $Coherence$ | $Cosine$ | $Kulc$ | $MaxConf$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $D_1$ | 10,000 | 1,000 | 1,000 | 100,000 | 90557 | 9.26 | 0.91 | 0.83 | 0.91 | 0.91 | 0.91 |
| $D_2$ | 10,000 | 1,000 | 1,000 | 100 | 0 | 1 | 0.91 | 0.83 | 0.91 | 0.91 | 0.91 |
| $D_3$ | 100 | 1,000 | 1,000 | 100,000 | 670 | 8.44 | 0.09 | 0.05 | 0.09 | 0.09 | 0.09 |
| $D_4$ | 1,000 | 1,000 | 1,000 | 100,000 | 24740 | 25.75 | 0.5 | 0.33 | 0.5 | 0.5 | 0.5 |
| $D_5$ | 1,000 | 100 | 10,000 | 100,000 | 8173 | 9.18 | 0.09 | 0.09 | 0.29 | 0.5 | 0.91 |
| $D_6$ | 1,000 | 10 | 100,000 | 100,000 | 965 | 1.97 | 0.01 | 0.01 | 0.10 | 0.5 | 0.99 |

Table 2. Example data sets.

Subtle: They disagree

# Which Null Invariant Measure is Better?

- IR (Imbalance Ratio): measures the imbalance of two itemsets A and B in rule implications

$$IR(A,B) = \frac{|sup(A) - sup(B)|}{sup(A) + sup(B) - sup(A \cup B)}$$

- **Kulczynski and Imbalance Ratio (IR) together present a clear picture for all the three datasets $D_4$ through $D_6$**
  - $D_4$ is balanced & neutral
  - $D_5$ is imbalanced & neutral
  - $D_6$ is very imbalanced & neutral

| $Data$ | $mc$ | $\overline{m}c$ | $m\overline{c}$ | $\overline{mc}$ | $all\_conf.$ | $max\_conf.$ | $Kulc.$ | $cosine$ | IR |
|---|---|---|---|---|---|---|---|---|---|
| $D_1$ | 10,000 | 1,000 | 1,000 | 100,000 | 0.91 | 0.91 | 0.91 | 0.91 | 0.0 |
| $D_2$ | 10,000 | 1,000 | 1,000 | 100 | 0.91 | 0.91 | 0.91 | 0.91 | 0.0 |
| $D_3$ | 100 | 1,000 | 1,000 | 100,000 | 0.09 | 0.09 | 0.09 | 0.09 | 0.0 |
| $D_4$ | 1,000 | 1,000 | 1,000 | 100,000 | 0.5 | 0.5 | 0.5 | 0.5 | 0.0 |
| $D_5$ | 1,000 | 100 | 10,000 | 100,000 | 0.09 | 0.91 | 0.5 | 0.29 | 0.89 |
| $D_6$ | 1,000 | 10 | 100,000 | 100,000 | 0.01 | 0.99 | 0.5 | 0.10 | 0.99 |