Association Analysis
Basic Concepts and Algorithms

INFO 523 - Lecture 8

Dr. Greg Chism

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 (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}) = σ({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:
    $\{Milk, Bread\} \rightarrow \{Diaper\}$

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:

$\{Milk, Bread\} \rightarrow \{Diaper\}$

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

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

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

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

  ⇒ 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

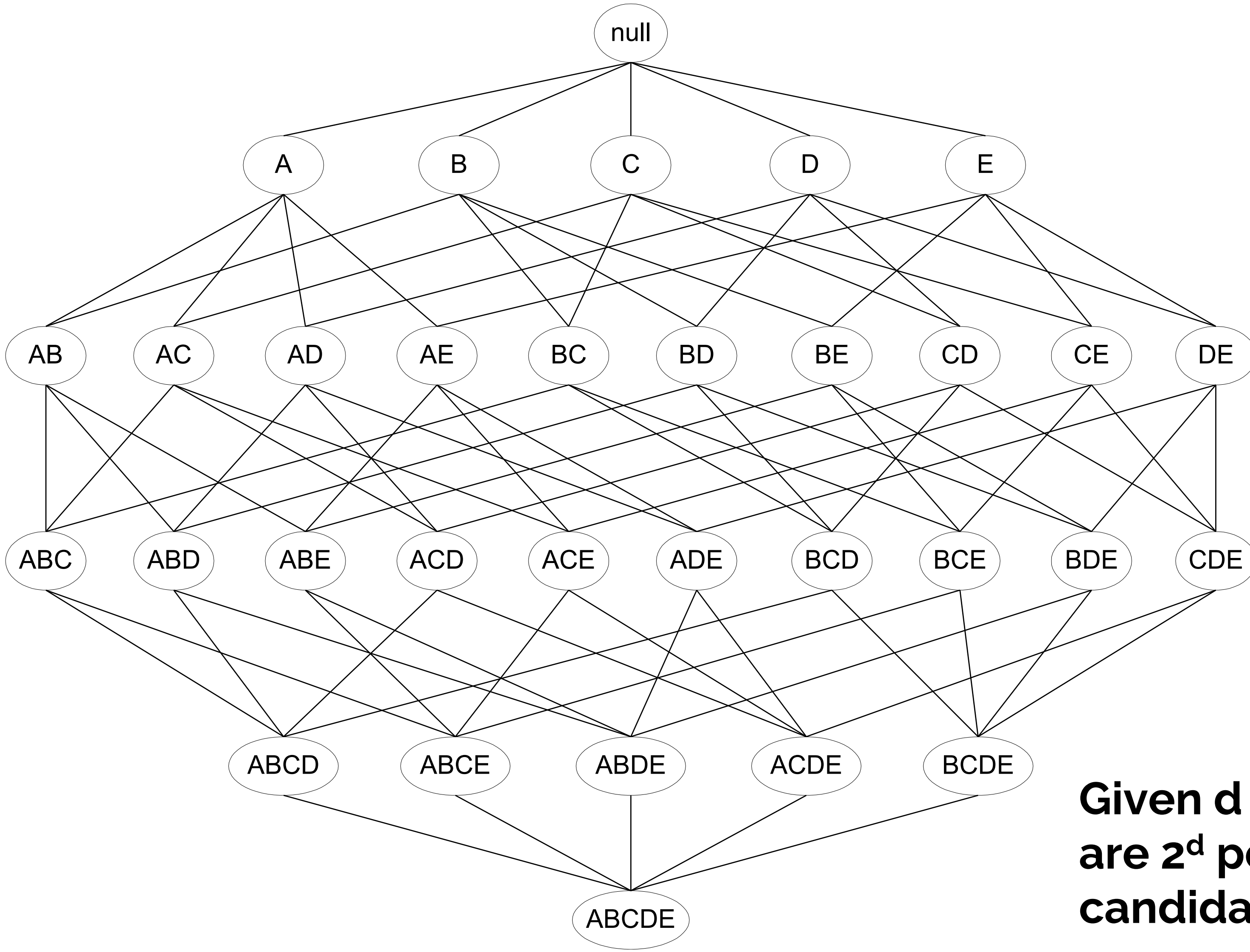
Mining Association Rules

- Two-step approach:
  1. Frequent Itemset Generation
     – Generate all itemsets whose support ≥ 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

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

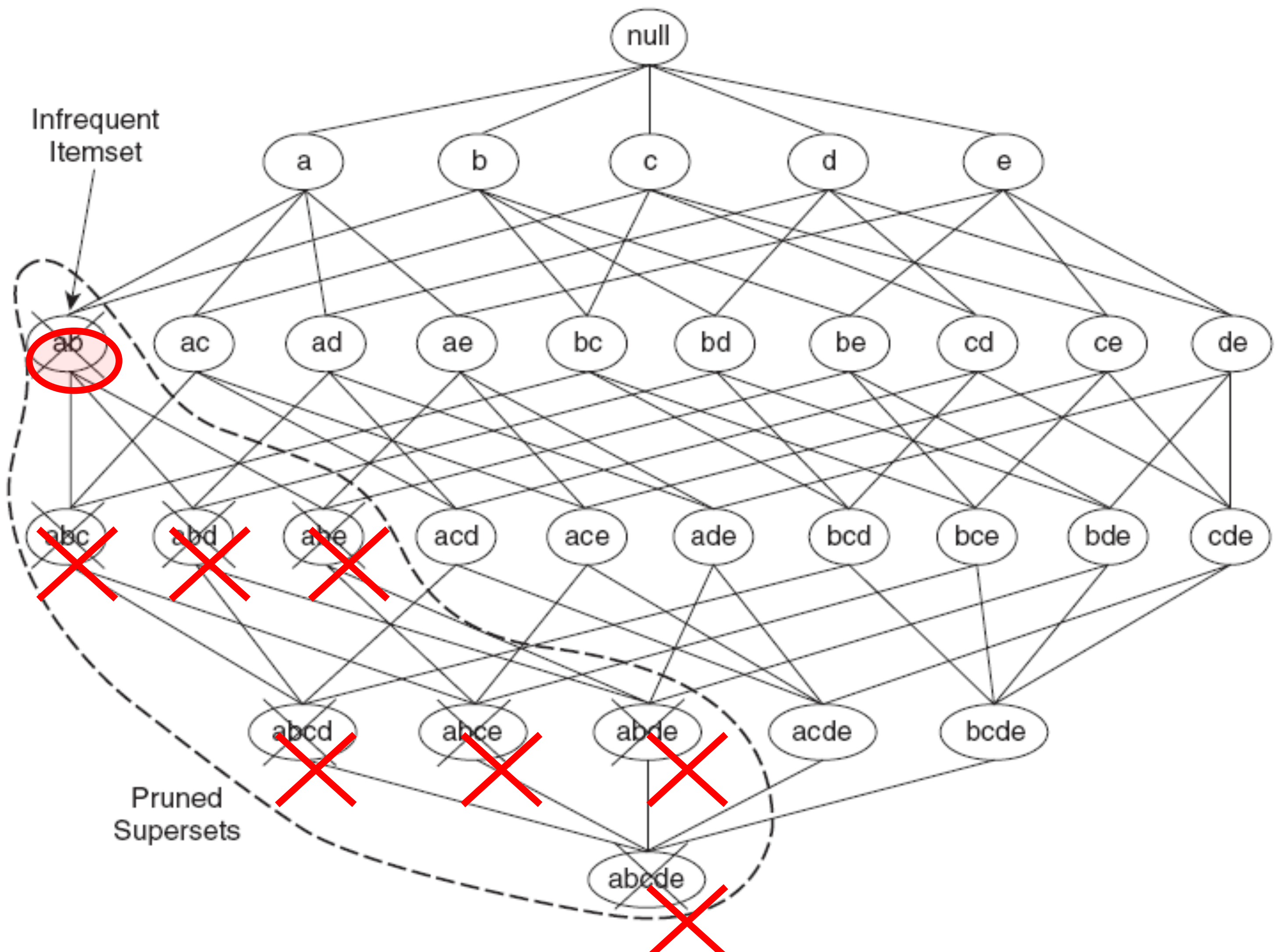


Figure 6.4. An illustration of support-based pruning. If $\{a, b\}$ is infrequent, then all supersets of $\{a, b\}$ are infrequent.

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

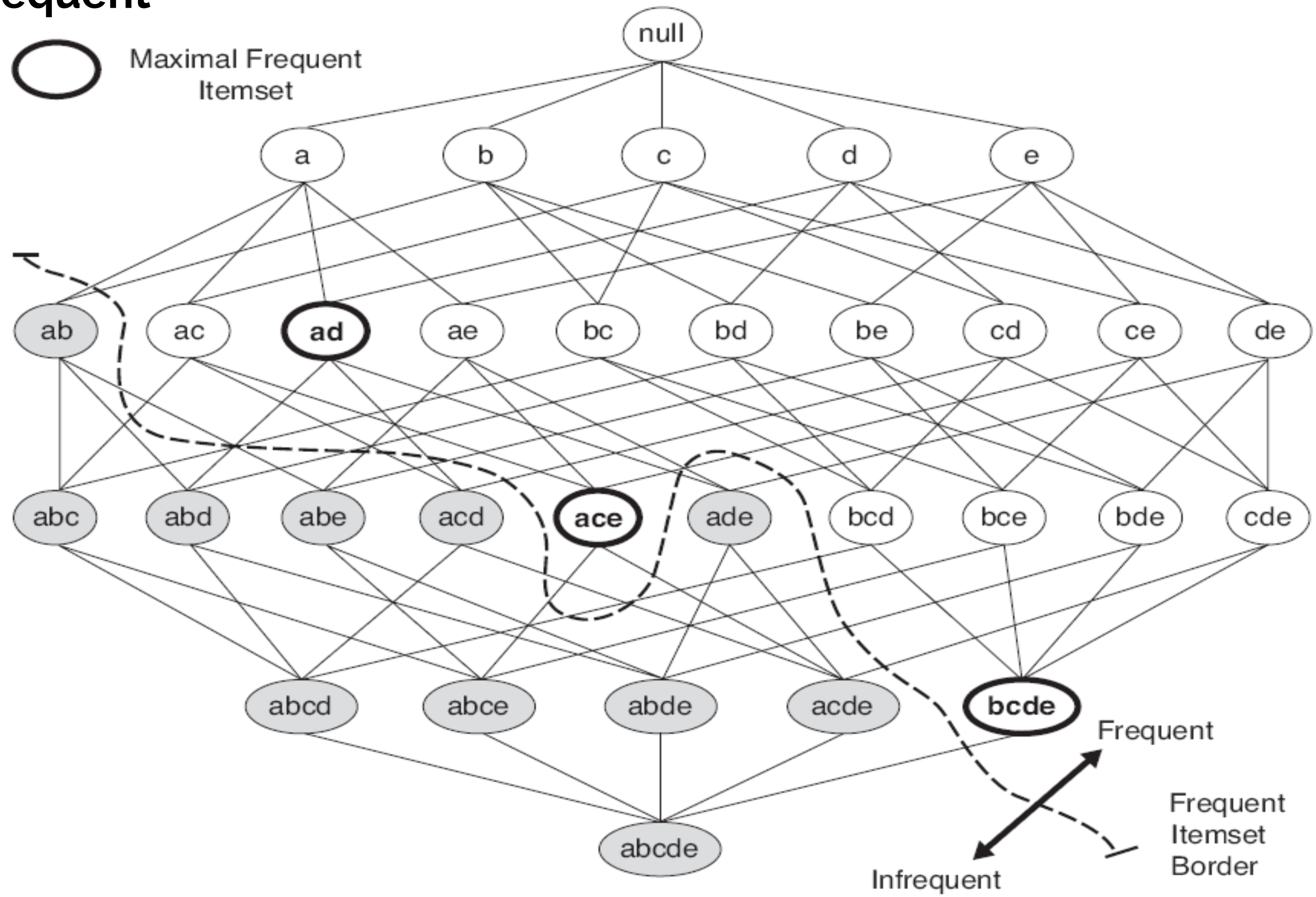
- Choice of minimum support threshold
  - lowering support threshold results in more frequent itemsets
  - this may increase number of candidates and max length of frequent itemsets
- Dimensionality (number of items) of the data set
  - more space is needed to store support count of each item
  - if 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



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)

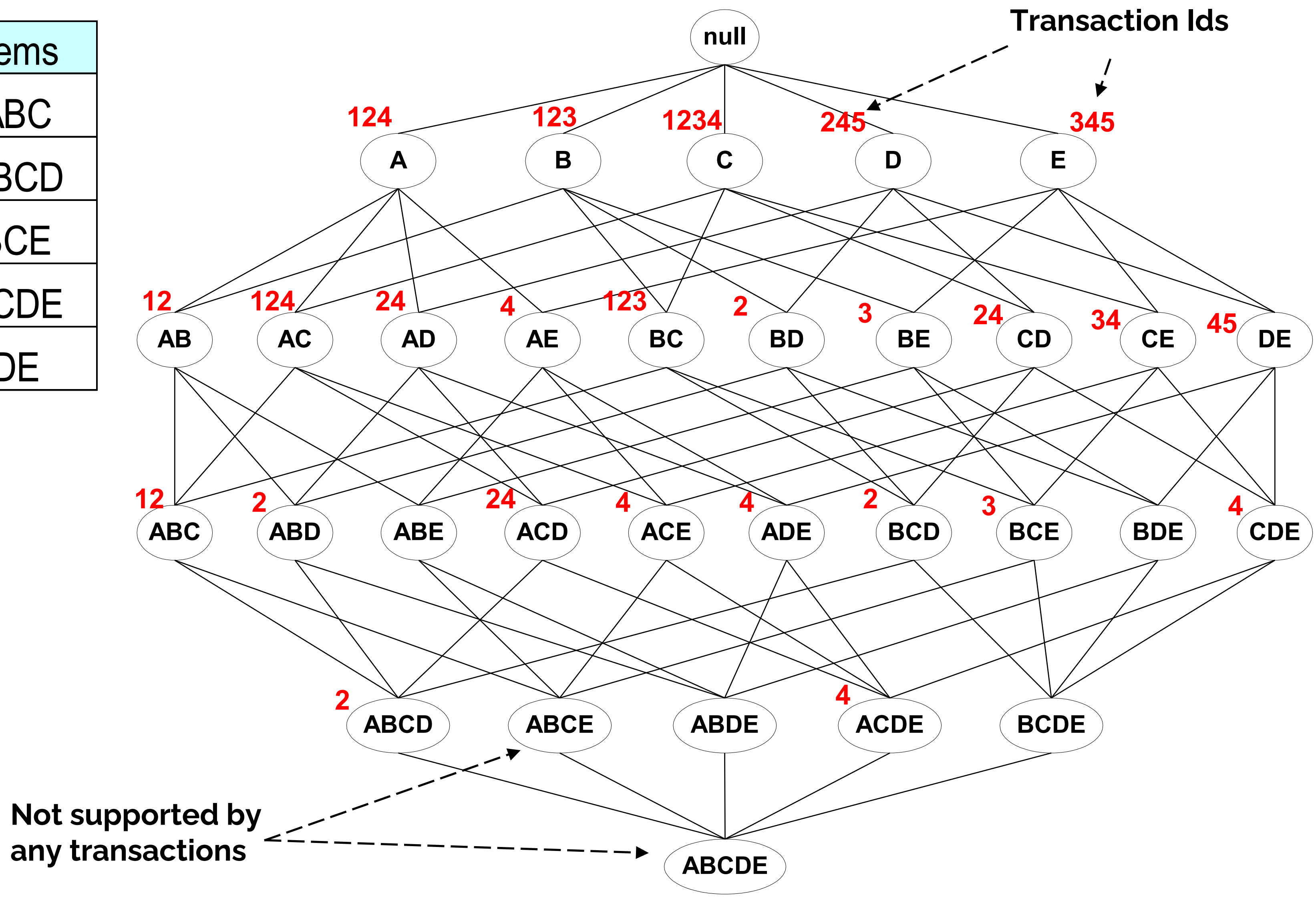
TID	Items
1	{A,B}
2	{B,C,D}
3	{A,B,C,D}
4	{A,B,D}
5	{A,B,C,D}

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

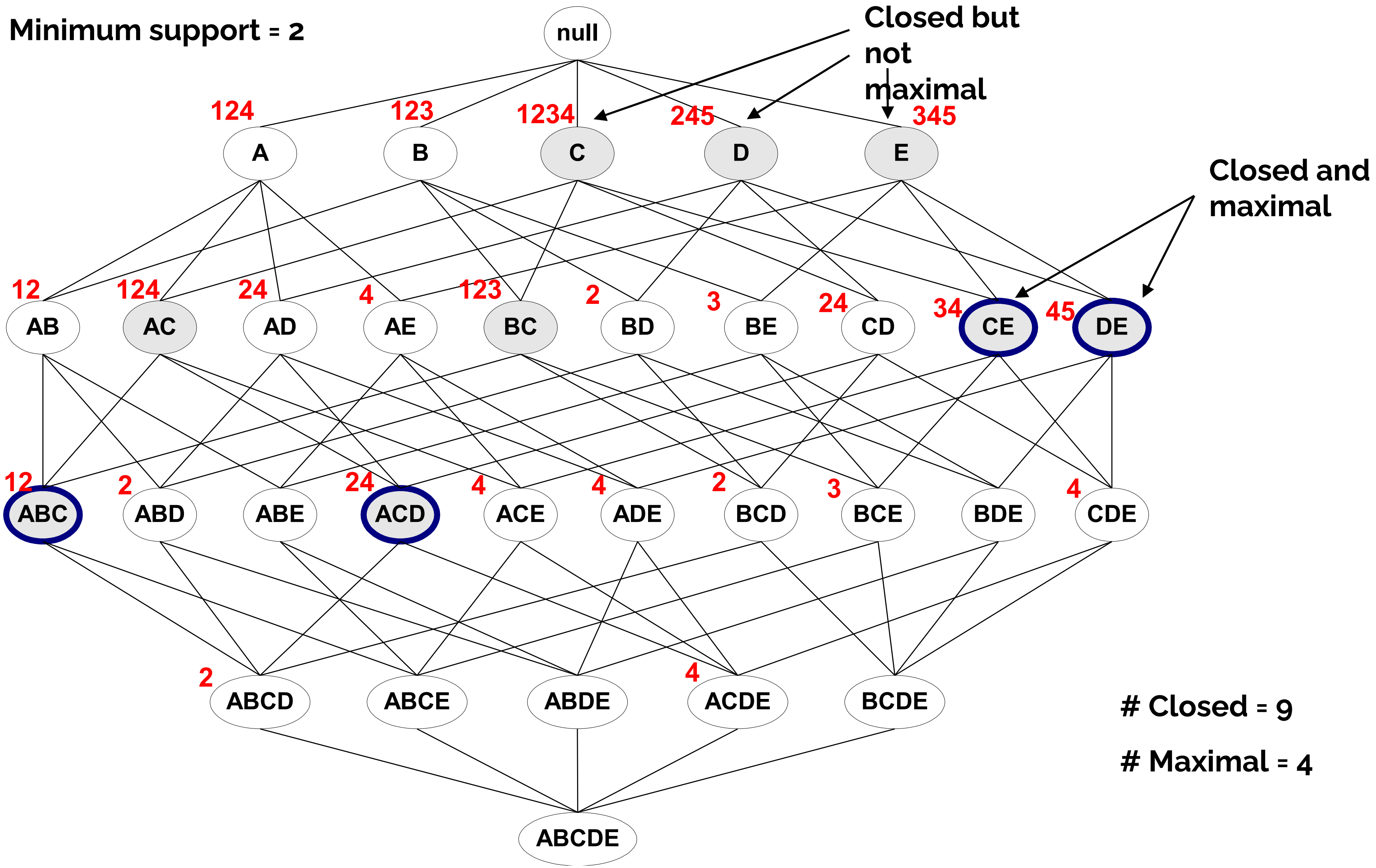
Itemset	Support
{A,B,C}	2
{A,B,D}	3
{A,C,D}	2
{B,C,D}	3
{A,B,C,D}	2

Maximal vs Closed Itemsets

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



Maximal vs Closed Frequent Itemsets



Maximal vs Closed Itemsets

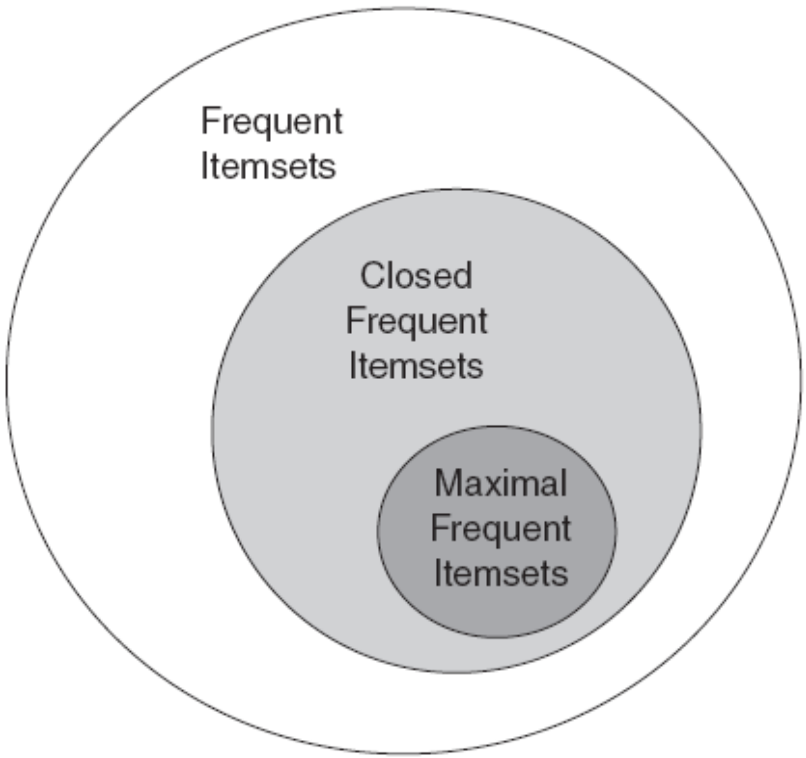


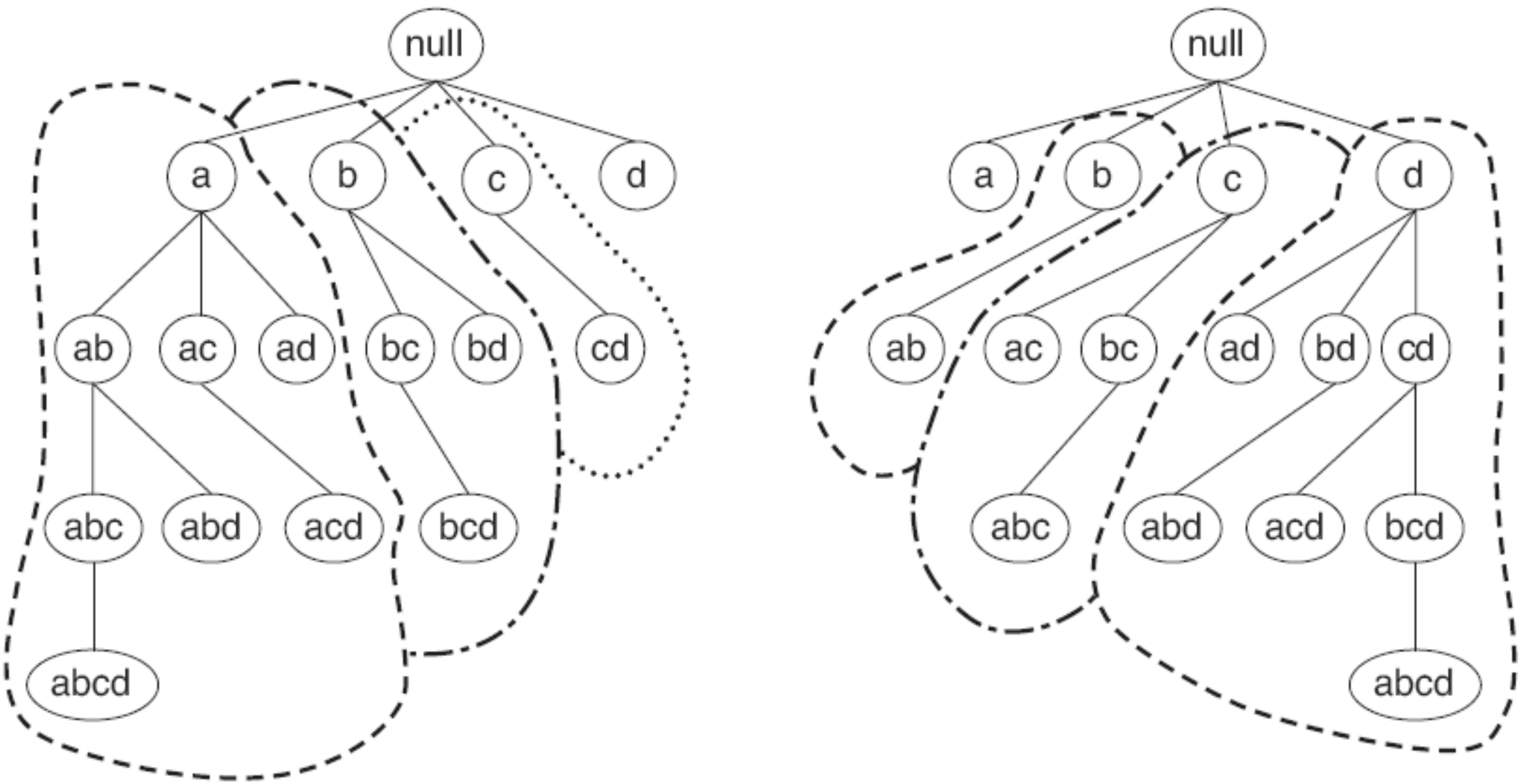
Figure 6.18. Relationships among frequent, maximal frequent, and closed frequent itemsets.

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
  - Equivalent Classes



(a) Prefix tree. (b) Suffix tree.

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				

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

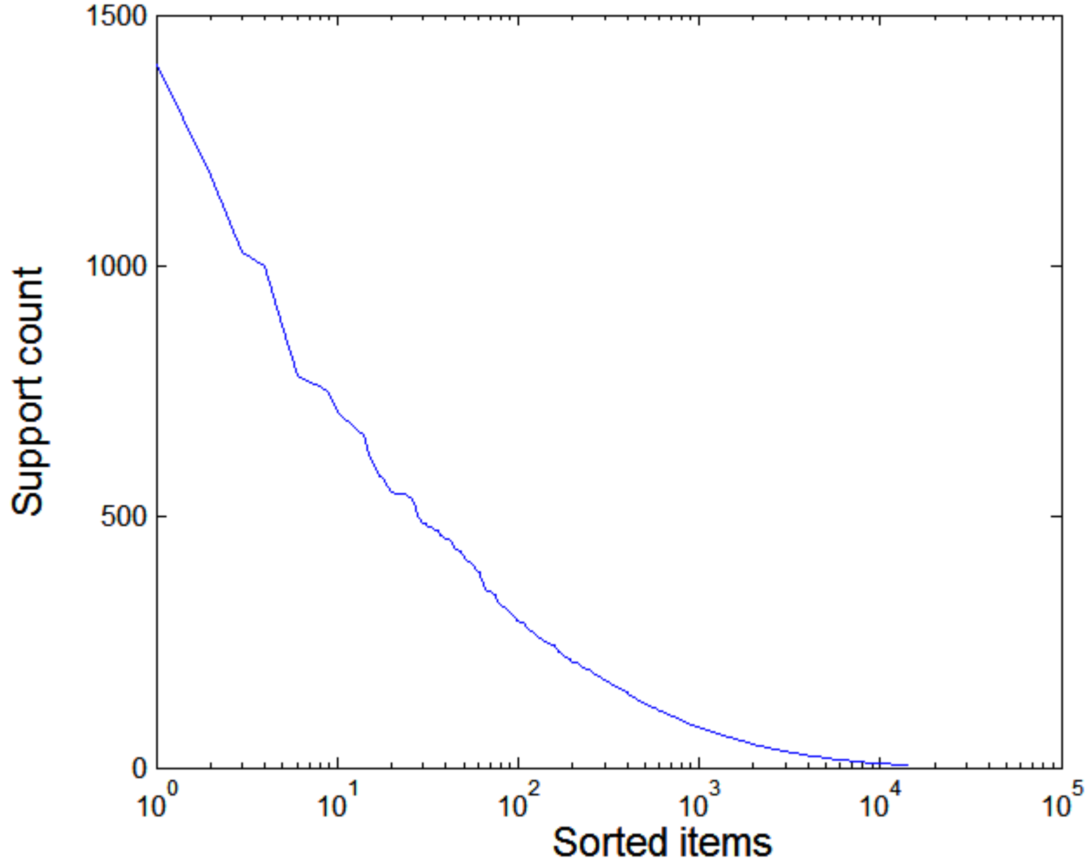
Topics

- 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

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

- A rule $\{A,B,C\} \rightarrow \{D\}$ can be considered **redundant** if $\{A,B\} \rightarrow \{D\}$ has the same or higher confidence.

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)