Data Mining: Concepts and Techniques

(3rd ed.)

— Chapter 5 —

Jiawei Han, Micheline Kamber, and Jian Pei
University of Illinois at Urbana-Champaign &
Simon Fraser University

©2009 Han, Kamber & Pei. All rights reserved.
Chapter 5: Mining Frequent Patterns, Association and Correlations: Basic Concepts and Methods

- Basic Concepts
- Scalable Frequent Itemset Mining Methods
- Are All Patterns Interesting?—Pattern Evaluation Methods
- Applications of Frequent Patterns and Associations
- Summary
What Is 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?— Beer and diapers?!
  - 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
  - Basket data analysis, cross-marketing, catalog design, sale campaign analysis, Web log (click stream) analysis, and DNA sequence analysis.
Why Is Freq. Pattern Mining Important?

- Freq. pattern: An intrinsic and important property of datasets
- Foundation for many essential data mining tasks
  - Association, correlation, and causality analysis
  - Sequential, structural (e.g., sub-graph) patterns
  - Pattern analysis in spatiotemporal, multimedia, time-series, and stream data
  - Classification: discriminative, frequent pattern analysis
  - Cluster analysis: frequent pattern-based clustering
  - Data warehousing: iceberg cube and cube-gradient
  - Semantic data compression: fascicles
  - Broad applications
Basic Concepts: Frequent Patterns

- **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 $\textit{minsup}$ threshold

<table>
<thead>
<tr>
<th>Tid</th>
<th>Items bought</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Beer, Nuts, Diaper</td>
</tr>
<tr>
<td>20</td>
<td>Beer, Coffee, Diaper</td>
</tr>
<tr>
<td>30</td>
<td>Beer, Diaper, Eggs</td>
</tr>
<tr>
<td>40</td>
<td>Nuts, Eggs, Milk</td>
</tr>
<tr>
<td>50</td>
<td>Nuts, Coffee, Diaper, Eggs, Milk</td>
</tr>
</tbody>
</table>

- **Customer buys both**
- **Customer buys beer**
- **Customer buys diaper**
Basic Concepts: Association Rules

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 $\text{minsup} = 50\%$, $\text{minconf} = 50\%$

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

Association rules: (many more!)
- $\text{Beer} \rightarrow \text{Diaper}$ (60%, 100%)
- $\text{Diaper} \rightarrow \text{Beer}$ (60%, 75%)
Closed Patterns and Max-Patterns

A long pattern contains a combinatorial number of sub-patterns, e.g., \{a_1, ..., a_{100}\} contains \( \binom{100}{1} + \binom{100}{2} + \cdots + \binom{1}{100} = 2^{100} - 1 = 1.27\times10^{30} \) sub-patterns!

Solution: Mine closed patterns and max-patterns instead

An itemset \( X \) is closed if \( X \) is frequent and there exists no super-pattern \( Y \supset X \), with the same support as \( X \) (proposed by Pasquier, et al. @ ICDT’99)

An itemset \( X \) is a max-pattern if \( X \) is frequent and there exists no frequent super-pattern \( Y \supset X \) (proposed by Bayardo @ SIGMOD’98)

Closed pattern is a lossless compression of freq. patterns
  - Reducing the # of patterns and rules
Closed Patterns and Max-Patterns

Exercise. DB = \{<a_{1}, \ldots, a_{100}>, <a_{1}, \ldots, a_{50}>\}
- Min\_sup = 1.

What is the set of closed itemset?
- <a_{1}, \ldots, a_{100}>: 1
- <a_{1}, \ldots, a_{50}>: 2

What is the set of max-pattern?
- <a_{1}, \ldots, a_{100}>: 1

What is the set of all patterns?
- !!
Computational Complexity of Frequent Itemset Mining

- How many itemsets are potentially to be generated in the worst case?
  - The number of frequent itemsets to be generated is sensitive to the minsup threshold
  - When minsup 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: $\sim10^{-40}$
    - What is the chance this particular set of 10 products to be frequent $10^3$ times in $10^9$ transactions?
Chapter 5: Mining Frequent Patterns, Association and Correlations: Basic Concepts and Methods

- Basic Concepts
- Scalable Frequent Itemset Mining Methods
- Are All Patterns Interesting?—Pattern Evaluation Methods
- Applications of Frequent Patterns and Associations
- Summary
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
  - **Any subset of a frequent itemset must be frequent**
  - If \{beer, diaper, nuts\} is frequent, so is \{beer, diaper\}
  - i.e., every transaction having \{beer, diaper, nuts\} also contains \{beer, diaper\}
- 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
  - **Test** the candidates against DB
  - Terminate when no frequent or candidate set can be generated
The Apriori Algorithm—An Example

<table>
<thead>
<tr>
<th>Tid</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>A, C, D</td>
</tr>
<tr>
<td>20</td>
<td>B, C, E</td>
</tr>
<tr>
<td>30</td>
<td>A, B, C, E</td>
</tr>
<tr>
<td>40</td>
<td>B, E</td>
</tr>
</tbody>
</table>

Database TDB

\[ \text{Sup}_{\min} = 2 \]

1\text{st scan}

\[ C_1 \]

\begin{itemize}
  \item \{A\} \quad 2
  \item \{B\} \quad 3
  \item \{C\} \quad 3
  \item \{D\} \quad 1
  \item \{E\} \quad 3
\end{itemize}

\[ L_1 \]

\begin{itemize}
  \item \{A\} \quad 2
  \item \{B\} \quad 3
  \item \{C\} \quad 3
  \item \{E\} \quad 3
\end{itemize}

2\text{nd scan}

\[ C_2 \]

\begin{itemize}
  \item \{A, B\} \quad 1
  \item \{A, C\} \quad 2
  \item \{A, E\} \quad 1
  \item \{B, C\} \quad 2
  \item \{B, E\} \quad 3
  \item \{C, E\} \quad 2
\end{itemize}

\[ L_2 \]

\begin{itemize}
  \item \{A, B\}
  \item \{A, C\}
  \item \{A, E\}
  \item \{B, C\}
  \item \{B, E\}
  \item \{C, E\}
\end{itemize}

3\text{rd scan}

\[ C_3 \]

\{B, C, E\}

\[ L_3 \]

\{B, C, E\} \quad 2
The Apriori Algorithm (Pseudo-Code)

\[ C_k: \text{Candidate itemset of size } k \]
\[ L_k: \text{frequent itemset of size } k \]

\[ L_1 = \{\text{frequent items}\}; \]
\[ \textbf{for} \ (k = 1; \ L_k \neq \emptyset; \ k++) \ \textbf{do begin} \]
\[ C_{k+1} = \text{candidates generated from } L_k; \]
\[ \textbf{for each} \ \text{transaction } t \ \text{in database do} \]
\[ \quad \text{increment the count of all candidates in } C_{k+1} \ \text{that} \]
\[ \quad \text{are contained in } t \]
\[ L_{k+1} = \text{candidates in } C_{k+1} \ \text{with min\_support} \]
\[ \textbf{end} \]
\[ \textbf{return} \ \bigcup_k L_k; \]

Input:
- \( D \), a database of transactions;
- \( \min\_sup \), the minimum support count threshold.

Output: \( L \), frequent itemsets in \( D \).

Method:

1. \( L_1 = \text{find\_frequent\_1\_itemsets}(D); \)
2. \( \text{for } (k = 2; L_{k-1} \neq \emptyset; k++ ) \{ \)
3. \( C_k = \text{apriori\_gen}(L_{k-1}); \)
4. \( \text{for each transaction } t \in D \{ // \text{scan } D \text{ for counts} \)
5. \( C_t = \text{subset}(C_k, t); // \text{get the subsets of } t \text{ that are candidates} \)
6. \( \text{for each candidate } c \in C_t \)
7. \( \quad c.\text{count}++; \)
8. \( \} \)
9. \( L_k = \{ c \in C_k | c.\text{count} \geq \min\_sup \} \)
10. \( \}\)
11. \( \text{return } L = \cup_k L_k; \)

procedure apriori\_gen\((L_{k-1});\text{frequent } (k-1)\text{-itemsets}\)

1. \( \text{for each itemset } l_1 \in L_{k-1} \)
2. \( \quad \text{for each itemset } l_2 \in L_{k-1} \)
3. \( \quad \quad \text{if } (l_1[1] = l_2[1]) \land (l_1[2] = l_2[2]) \land \ldots \land (l_1[k-2] = l_2[k-2]) \land (l_1[k-1] < l_2[k-1]) \) then \{ \)
4. \( \quad \quad \quad c = l_1 \uplus l_2; // \text{join step: generate candidates} \)
5. \( \quad \quad \quad \text{if has\_infrequent\_subset}(c, L_{k-1}) \) then \{ \)
6. \( \quad \quad \quad \quad \text{delete } c; // \text{prune step: remove unfruitful candidate} \)
7. \( \quad \quad \quad \text{else add } c \text{ to } C_k; \)
8. \( \quad \}\)
9. \( \text{return } C_k; \)

procedure has\_infrequent\_subset\((c; \text{candidate } k\text{-itemset};\)

\( L_{k-1}; \text{frequent } (k-1)\text{-itemsets}; // \text{use prior knowledge} \)

1. \( \text{for each } (k-1)\text{-subset } s \text{ of } c \)
2. \( \quad \text{if } s \not\in L_{k-1} \) then \{ \)
3. \( \quad \quad \text{return } \text{TRUE}; \)
4. \( \quad \}\)
\( \text{return } \text{FALSE}; \)
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
Further Improvement of the Apriori Method

- Major computational challenges
  - Multiple scans of transaction database
  - Huge number of candidates
  - Tedious workload of support counting for candidates

- Improving Apriori: general ideas
  - Reduce passes of transaction database scans (giảm số lần phải scan database)
  - Shrink number of candidates (thu nhỏ số lượng candidate)
  - Facilitate support counting of candidates (de dằng thực hiện việc tính support)
Partition: Scan Database Only Twice

- Any itemset that is potentially frequent in DB must be frequent in at least one of the partitions of DB
  - Scan 1: partition database and find local frequent patterns
  - Scan 2: consolidate global frequent patterns
- A. Savasere, E. Omiecinski and S. Navathe, *VLDB’95*

\[
\begin{align*}
\text{DB}_1 & \quad + \quad \text{DB}_2 & \quad + & \quad \cdots & \quad + \quad \text{DB}_k \\
\text{sup}_1(i) & < \sigma_{\text{DB}_1} & \quad \text{sup}_2(i) & < \sigma_{\text{DB}_2} & \quad \text{sup}_k(i) & < \sigma_{\text{DB}_k} \\
\text{sup}(i) & < \sigma_{\text{DB}}
\end{align*}
\]
A $k$-itemset whose corresponding hashing bucket count is below the threshold cannot be frequent

- Candidates: a, b, c, d, e
- Hash entries
  - \{ab, ad, ae\}
  - \{bd, be, de\}
  - ...
- Frequent 1-itemset: a, b, d, e
- ab is not a candidate 2-itemset if the sum of count of \{ab, ad, ae\} is below support threshold

J. Park, M. Chen, and P. Yu. *An effective hash-based algorithm for mining association rules*. *SIGMOD’95*
Sampling for Frequent Patterns

- Select a sample of original database, mine frequent patterns within sample using Apriori
- Scan database once to verify frequent itemsets found in sample, only *borders* of closure of frequent patterns are checked
  - Example: check *abcd* instead of *ab, ac, ..., etc.*
- Scan database again to find missed frequent patterns
- H. Toivonen. *Sampling large databases for association rules*. In *VLDB’96*
DIC: Reduce Number of Scans

Once both A and D are determined frequent, the counting of AD begins.

Once all length-2 subsets of BCD are determined frequent, the counting of BCD begins.

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
Pattern-Growth Approach: Mining Frequent Patterns Without Candidate Generation

- Bottlenecks of the Apriori approach
  - Breadth-first (i.e., level-wise) search
  - Candidate generation and test
    - Often generates a huge number of candidates
- The FPGrowth Approach (J. Han, J. Pei, and Y. Yin, SIGMOD’ 00)
  - Depth-first search
  - Avoid explicit candidate generation
- Major philosophy: Grow long patterns from short ones using local frequent items only
  - “abc” is a frequent pattern
  - Get all transactions having “abc”, i.e., project DB on abc: DB|abc
  - “d” is a local frequent item in DB|abc → abcd is a frequent pattern
Construct FP-tree from a Transaction Database

<table>
<thead>
<tr>
<th>TID</th>
<th>Items bought</th>
<th>(ordered) frequent items</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>{f, a, c, d, g, i, m, p}</td>
<td>{f, c, a, m, p}</td>
</tr>
<tr>
<td>200</td>
<td>{a, b, c, f, l, m, o}</td>
<td>{f, c, a, b, m}</td>
</tr>
<tr>
<td>300</td>
<td>{b, f, h, j, o, w}</td>
<td>{f, b}</td>
</tr>
<tr>
<td>400</td>
<td>{b, c, k, s, p}</td>
<td>{c, b, p}</td>
</tr>
<tr>
<td>500</td>
<td>{a, f, c, e, l, p, m, n}</td>
<td>{f, c, a, m, p}</td>
</tr>
</tbody>
</table>

1. Scan DB once, find frequent 1-itemset (single item pattern)
2. Sort frequent items in frequency descending order, f-list
3. Scan DB again, construct FP-tree

F-list = f-c-a-b-m-p
Partition Patterns and Databases

- Frequent patterns can be partitioned into subsets according to f-list
  - F-list = f-c-a-b-m-p
  - Patterns containing p
  - Patterns having m but no p
  - ...
  - Patterns having c but no a nor b, m, p
  - Pattern f
- Completeness and non-redundency
Find Patterns Having P From P-conditional Database

- Starting at the frequent item header table in the FP-tree
- Traverse the FP-tree by following the link of each frequent item $p$
- Accumulate all of transformed prefix paths of item $p$ to form $p$'s conditional pattern base
From Conditional Pattern-bases to Conditional FP-trees

- For each pattern-base
  - Accumulate the count for each item in the base
  - Construct the FP-tree for the frequent items of the pattern base

---

**Header Table**

<table>
<thead>
<tr>
<th>Item</th>
<th>Frequency</th>
<th>Head</th>
</tr>
</thead>
<tbody>
<tr>
<td>f</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>m</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

**m-conditional pattern base:**

- \(fca:2, fcab:1\)

**All frequent patterns relate to \(m\):**

- \(m, \ fcm, cm, am, fcam\)

**m-conditional FP-tree**
Recursion: Mining Each Conditional FP-tree

Cond. pattern base of “am”: (fc:3)

```
{}                  f:3
\--|\--                    c:3
  |      \--
  \--|\--
    f:3
    \--
    c:3
    \--
    a:3
```

am-conditional FP-tree

Cond. pattern base of “cm”: (f:3)

```
{}                  f:3
\--|\--                    c:3
  |      \--
  \--|\--
    f:3
    \--
```

cm-conditional FP-tree

Cond. pattern base of “cam”: (f:3)

```
{}                  f:3
\--|\--
  \--
```

cam-conditional FP-tree
A Special Case: Single Prefix Path in FP-tree

- Suppose a (conditional) FP-tree T has a shared single prefix-path P
- Mining can be decomposed into two parts
  - Reduction of the single prefix path into one node
  - Concatenation of the mining results of the two parts

```
    {}                  b1:m1
    \        \       /  \      /  \     \    \    \    r1   =   \    \    \    a1:n1
  a1:n1          C1:k1   a2:n2          a3:n3
        \                        \       /  \      /  \     \    \    \    ...
      C2:k2         C3:k3     C2:k2         C3:k3
```

October 21, 2010
Data Mining: Concepts and Techniques
Benefits of the FP-tree Structure

- Completeness
  - Preserve complete information for frequent pattern mining
  - Never break a long pattern of any transaction
- Compactness
  - Reduce irrelevant info—infrequent items are gone
  - Items in frequency descending order: the more frequently occurring, the more likely to be shared
  - Never be larger than the original database (not count node-links and the \textit{count} field)
The Frequent Pattern Growth Mining Method

- Idea: Frequent pattern growth
  - Recursively grow frequent patterns by pattern and database partition

- Method
  - For each frequent item, construct its conditional pattern-base, and then its conditional FP-tree
  - Repeat the process on each newly created conditional FP-tree
  - Until the resulting FP-tree is empty, or it contains only one path—single path will generate all the combinations of its sub-paths, each of which is a frequent pattern
Scaling FP-growth by Database Projection

- What about if FP-tree cannot fit in memory?
  - DB projection
  - First partition a database into a set of projected DBs
  - Then construct and mine FP-tree for each projected DB
  - Parallel projection vs. partition projection techniques
    - Parallel projection
      - Project the DB in parallel for each frequent item
      - Parallel projection is space costly
      - All the partitions can be processed in parallel
    - Partition projection
      - Partition the DB based on the ordered frequent items
      - Passing the unprocessed parts to the subsequent partitions
Partition-Based Projection

- Parallel projection needs a lot of disk space
- Partition projection saves it
FP-Growth vs. Apriori: Scalability With the Support Threshold

Data set T25I20D10K

- D1 FP-growth runtime
- D1 Apriori runtime

Run time (sec.) vs. Support threshold (%)

Data set T25I20D10K
FP-Growth vs. Tree-Projection: Scalability with the Support Threshold

Data set T25I20D100K

- **D2 FP-growth**
- **D2 TreeProjection**
Advantages of the Pattern Growth Approach

- **Divide-and-conquer:**
  - Decompose both the mining task and DB according to the frequent patterns obtained so far
  - Lead to focused search of smaller databases
- **Other factors**
  - No candidate generation, no candidate test
  - Compressed database: FP-tree structure
  - No repeated scan of entire database
  - Basic ops: counting local freq items and building sub FP-tree, no pattern search and matching
- **A good open-source implementation and refinement of FPGrowth**
  - FPGrowth+ (Grahne and J. Zhu, FIMI'03)
Further Improvements of Mining Methods

- AFOPT (Liu, et al. @ KDD’03)
  - A “push-right” method for mining condensed frequent pattern (CFP) tree
- Carpenter (Pan, et al. @ KDD’03)
  - Mine data sets with small rows but numerous columns
  - Construct a row-enumeration tree for efficient mining
- FPgrowth+ (Grahne and Zhu, FIMI’03)
  - Efficiently Using Prefix-Trees in Mining Frequent Itemsets, Proc. ICDM'03 Int. Workshop on Frequent Itemset Mining Implementations (FIMI'03), Melbourne, FL, Nov. 2003
- TD-Close (Liu, et al, SDM’06)
Extension of Pattern Growth Mining Methodology

- Mining closed frequent itemsets and max-patterns
  - CLOSET (DMKD’00), FPclose, and FPMax (Grahne & Zhu, Fimi’03)
- Mining sequential patterns
  - PrefixSpan (ICDE’01), CloSpan (SDM’03), BIDE (ICDE’04)
- Mining graph patterns
  - gSpan (ICDM’02), CloseGraph (KDD’03)
- Constraint-based mining of frequent patterns
  - Convertible constraints (ICDE’01), gPrune (PAKDD’03)
- Computing iceberg data cubes with complex measures
  - H-tree, H-cubing, and Star-cubing (SIGMOD’01, VLDB’03)
- Pattern-growth-based Clustering
  - MaPle (Pei, et al., ICDM’03)
- Pattern-Growth-Based Classification
  - Mining frequent and discriminative patterns (Cheng, et al, ICDE’07)
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
ECLAT: Mining by Exploring Vertical Data Format

- Vertical format: \( t(AB) = \{T_{11}, T_{25}, \ldots\} \)
  - tid-list: list of trans.-ids containing an itemset
- Deriving frequent patterns based on vertical intersections
  - \( t(X) = t(Y) \): X and Y always happen together
  - \( t(X) \subseteq t(Y) \): transaction having X always has Y
- Using **diffset** to accelerate mining
  - Only keep track of differences of tids
  - \( t(X) = \{T_1, T_2, T_3\}, \ t(XY) = \{T_1, T_3\} \)
  - Diffset \((XY, X) = \{T_2\}\)
- Eclat (Zaki et al. @KDD’97)
- Mining Closed patterns using vertical format: CHARM (Zaki & Hsiao@SDM’02)
Chapter 5: Mining Frequent Patterns, Association and Correlations: Basic Concepts and Methods

- Basic Concepts
- Scalable Frequent Itemset Mining Methods
- Are All Patterns Interesting?—Pattern Evaluation Methods
- Applications of Frequent Patterns and Associations
- Summary
Interestingness Measure: Correlations (Lift)

- *play basketball ⇒ eat cereal* [40%, 66.7%] is misleading
  - The overall % of students eating cereal is 75% > 66.7%.
- *play basketball ⇒ not eat cereal* [20%, 33.3%] is more accurate, although with lower support and confidence
- Measure of dependent/correlated events: *lift*

\[
\text{lift} = \frac{P(A \cup B)}{P(A)P(B)}
\]

\[
lift(B, C) = \frac{2000/5000}{3000/5000 \times 3750/5000} = 0.89
\]

\[
lift(B, \neg C) = \frac{1000/5000}{3000/5000 \times 1250/5000} = 1.33
\]
"Buy walnuts ⇒ buy milk [1%, 80%]" is misleading if 85% of customers buy milk.

Support and confidence are not good to indicate correlations.

Over 20 interestingness measures have been proposed (see Tan, Kumar, Sritastava @KDD’02).

Which are good ones?

<table>
<thead>
<tr>
<th>symbol</th>
<th>measure</th>
<th>range</th>
<th>formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\phi$</td>
<td>$\phi$-coefficient</td>
<td>-1...1</td>
<td>$\frac{P(A \cap B) - P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$</td>
</tr>
<tr>
<td>$Q$</td>
<td>Yule’s $Q$</td>
<td>-1...1</td>
<td>$\frac{P(A \cap B) - P(A)P(B)}{P(A,B)P(A) + P(A,B)P(B)}$</td>
</tr>
<tr>
<td>$Y$</td>
<td>Yule’s $Y$</td>
<td>-1...1</td>
<td>$\sqrt{P(A,B)P(A,B) - P(A)P(B)}$</td>
</tr>
<tr>
<td>$k$</td>
<td>Cohen’s</td>
<td>-1...1</td>
<td>$\sqrt{\frac{P(A,B)P(A,B) - P(A)P(B)}{1 - (P(A)P(B) + P(A,B)P(B))}}$</td>
</tr>
<tr>
<td>$PS$</td>
<td>Piatetsky-Shapiro’s</td>
<td>-0.25 to 0.25</td>
<td>$\chi^2 = \frac{\sum_{i=1}^{k}(O_i - E_i)^2}{E_i}$</td>
</tr>
<tr>
<td>$F$</td>
<td>Certainty factor</td>
<td>-1...1</td>
<td>$\max(\frac{P(B</td>
</tr>
<tr>
<td>$AV$</td>
<td>added value</td>
<td>-0.5...1</td>
<td>$\max(P(B</td>
</tr>
<tr>
<td>$K$</td>
<td>Klosgen’s $Q$</td>
<td>0.33...0.38</td>
<td>$\frac{\Sigma \max_i P(A_j,B_k) - \Sigma \max_j P(A_j,B_k) - \max_i P(A_j) + \max_k P(B_k)}{2}$</td>
</tr>
<tr>
<td>$g$</td>
<td>Goodman-Kruskal’s</td>
<td>0...1</td>
<td>$\frac{P(B</td>
</tr>
<tr>
<td>$M$</td>
<td>Mutual Information</td>
<td>0...1</td>
<td>$\max(\frac{P(B</td>
</tr>
<tr>
<td>$J$</td>
<td>J-Measure</td>
<td>0...1</td>
<td>$\max(P(B</td>
</tr>
<tr>
<td>$G$</td>
<td>Gini index</td>
<td>0...1</td>
<td>$\max(P(B</td>
</tr>
<tr>
<td>$s$</td>
<td>support</td>
<td>0...1</td>
<td>$\max(NP(A</td>
</tr>
<tr>
<td>$c$</td>
<td>confidence</td>
<td>0...1</td>
<td>$\max(P(B</td>
</tr>
<tr>
<td>$L$</td>
<td>Laplace</td>
<td>0...1</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$IS$</td>
<td>Cosine</td>
<td>0...1</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>coherence (Jaccard)</td>
<td>0...1</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>all-confidence</td>
<td>0...1</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$o$</td>
<td>odds ratio</td>
<td>0...∞</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$V$</td>
<td>Conviction</td>
<td>0.5...∞</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>lift</td>
<td>0...∞</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$S$</td>
<td>Collective strength</td>
<td>0...∞</td>
<td>$\max(\frac{P(A</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>$\chi^2$</td>
<td>0...∞</td>
<td>$\Sigma_i (O_i - E_i)^2$</td>
</tr>
</tbody>
</table>

Are lift and $\chi^2$ Good Measures of Correlation?
Null-Invariant Measures

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

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Measure</th>
<th>Range</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>O1</th>
<th>O2</th>
<th>O3</th>
<th>O3’</th>
<th>O4</th>
</tr>
</thead>
<tbody>
<tr>
<td>φ</td>
<td>φ-coefficient</td>
<td>−1 . . . 0 . . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>λ</td>
<td>Goodman-Kruskal’s</td>
<td>0 . . 1</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No*</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>α</td>
<td>odds ratio</td>
<td>0 . . 1 . . . ∞</td>
<td>Yes*</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes*</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Q</td>
<td>Yule’s Q</td>
<td>−1 . . . 0 . . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Y</td>
<td>Yule’s Y</td>
<td>−1 . . . 0 . . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>κ</td>
<td>Cohen’s</td>
<td>−1 . . . 0 . . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>M</td>
<td>Mutual Information</td>
<td>0 . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No**</td>
<td>No*</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>J-Measure</td>
<td>0 . . 1</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>G</td>
<td>Gini index</td>
<td>0 . . 1</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>s</td>
<td>Support</td>
<td>0 . . 1</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>c</td>
<td>Confidence</td>
<td>0 . . 1</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>L</td>
<td>Laplace</td>
<td>0 . . 1</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>V</td>
<td>Conviction</td>
<td>0.5 . . 1 . . . ∞</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>I</td>
<td>Interest</td>
<td>0 . . 1 . . . ∞</td>
<td>Yes*</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>IS</td>
<td>Cosine</td>
<td>0 . . √P(A, B) . . 1</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>PS</td>
<td>Piatesky-Shapiro’s</td>
<td>−0.25 . . 0 . . 0.25</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>F</td>
<td>Certainty factor</td>
<td>−1 . . . 0 . . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>AV</td>
<td>Added value</td>
<td>−0.5 . . 0 . . 1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>S</td>
<td>Collective strength</td>
<td>0 . . 1 . . . ∞</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes*</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>ζ</td>
<td>Jaccard</td>
<td>0 . . 1</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>K</td>
<td>Kisosgen’s</td>
<td>(2 - 1)^(1/2)[2 - 3 - (1/3)] · . . . 2 / 3√3</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No**</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

where: P1: \( O(M) = 0 \) if \( \det[M] = 0 \), i.e., whenever \( A \) and \( B \) are statistically independent.

P2: \( O(M_2) > O(M_1) \) if \( M_2 = M_1 + [k \ - k; \ -k \ k] \).

P3: \( O(M_2) < O(M_1) \) if \( M_2 = M_1 + [0 \ k; \ 0 \ -k] \) or \( M_2 = M_1 + [0 \ 0; \ k \ -k] \).

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

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

Kulczynski measure (1927)

Null-invariant

Subtle: They disagree

<table>
<thead>
<tr>
<th>Measure</th>
<th>Definition</th>
<th>Range</th>
<th>Null-Invariant</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\chi^2(a,b)$</td>
<td>$\sum_{i,j=0,1} \frac{(e(a_i,b_j) - o(a_i,b_j))^2}{e(a_i,b_j)}$</td>
<td>$[0, \infty]$</td>
<td>No</td>
</tr>
<tr>
<td>Lift(a, b)</td>
<td>$\frac{P(ab)}{P(a)P(b)}$</td>
<td>$[0, \infty]$</td>
<td>No</td>
</tr>
<tr>
<td>AllConf(a, b)</td>
<td>$\frac{sup(ab)}{\max{sup(a), sup(b)}}$</td>
<td>$[0, 1]$</td>
<td>Yes</td>
</tr>
<tr>
<td>Coherence(a, b)</td>
<td>$\frac{sup(ab)}{sup(a)+sup(b)-sup(ab)}$</td>
<td>$[0, 1]$</td>
<td>Yes</td>
</tr>
<tr>
<td>Cosine(a, b)</td>
<td>$\frac{sup(ab)}{\sqrt{sup(a)sup(b)}}$</td>
<td>$[0, 1]$</td>
<td>Yes</td>
</tr>
<tr>
<td>Kulc(a, b)</td>
<td>$\frac{sup(ab)}{2} \left( \frac{1}{sup(a)} + \frac{1}{sup(b)} \right)$</td>
<td>$[0, 1]$</td>
<td>Yes</td>
</tr>
<tr>
<td>MaxConf(a, b)</td>
<td>$\max\left{ \frac{sup(ab)}{sup(a)}, \frac{sup(ab)}{sup(b)} \right}$</td>
<td>$[0, 1]$</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 3. Interestingness measure definitions.

Table 2. Example data sets.
Analysis of DBLP Coauthor Relationships

Recent DB conferences, removing balanced associations, low sup, etc.

<table>
<thead>
<tr>
<th>ID</th>
<th>Author a</th>
<th>Author b</th>
<th>sup(ab)</th>
<th>sup(a)</th>
<th>sup(b)</th>
<th>Coherence</th>
<th>Cosine</th>
<th>Kulc</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hans-Peter Kriegel</td>
<td>Martin Ester</td>
<td>28</td>
<td>146</td>
<td>54</td>
<td>0.163 (2)</td>
<td>0.315 (7)</td>
<td>0.355 (9)</td>
</tr>
<tr>
<td>2</td>
<td>Michael Carey</td>
<td>Miron Livny</td>
<td>26</td>
<td>104</td>
<td>58</td>
<td>0.191 (1)</td>
<td>0.335 (4)</td>
<td>0.349 (10)</td>
</tr>
<tr>
<td>3</td>
<td>Hans-Peter Kriegel</td>
<td>Joerg Sander</td>
<td>24</td>
<td>146</td>
<td>36</td>
<td>0.152 (3)</td>
<td>0.331 (5)</td>
<td>0.416 (8)</td>
</tr>
<tr>
<td>4</td>
<td>Christos Faloutsos</td>
<td>Spiros Papadimitriou</td>
<td>20</td>
<td>162</td>
<td>26</td>
<td>0.119 (7)</td>
<td>0.308 (10)</td>
<td>0.446 (7)</td>
</tr>
<tr>
<td>5</td>
<td>Hans-Peter Kriegel</td>
<td>Martin Pfeifle</td>
<td>18</td>
<td>146</td>
<td>18</td>
<td>0.123 (6)</td>
<td>0.351 (2)</td>
<td>0.562 (2)</td>
</tr>
<tr>
<td>6</td>
<td>Hector Garcia-Molina</td>
<td>Wilbur Labio</td>
<td>16</td>
<td>144</td>
<td>18</td>
<td>0.110 (9)</td>
<td>0.314 (8)</td>
<td>0.500 (4)</td>
</tr>
<tr>
<td>7</td>
<td>Divyakant Agrawal</td>
<td>Wang Hsiung</td>
<td>16</td>
<td>120</td>
<td>16</td>
<td>0.133 (5)</td>
<td>0.365 (1)</td>
<td>0.567 (1)</td>
</tr>
<tr>
<td>8</td>
<td>Elke Rundensteiner</td>
<td>Murali Mani</td>
<td>16</td>
<td>104</td>
<td>20</td>
<td>0.148 (4)</td>
<td>0.351 (3)</td>
<td>0.477 (6)</td>
</tr>
<tr>
<td>9</td>
<td>Divyakant Agrawal</td>
<td>Oliver Po</td>
<td>12</td>
<td>120</td>
<td>12</td>
<td>0.100 (10)</td>
<td>0.316 (6)</td>
<td>0.550 (3)</td>
</tr>
<tr>
<td>10</td>
<td>Gerhard Weikum</td>
<td>Martin Theobald</td>
<td>12</td>
<td>106</td>
<td>14</td>
<td>0.111 (8)</td>
<td>0.312 (9)</td>
<td>0.485 (5)</td>
</tr>
</tbody>
</table>

Table 5. Experiment on DBLP data set.

- Advisor-advisee relation: Kulc: high, coherence: low, cosine: middle

- Tianyi Wu, Yuguo Chen and Jiawei Han, “Association Mining in Large Databases: A Re-Examination of Its Measures”, Proc. 2007 Int. Conf. Principles and Practice of Knowledge Discovery in Databases (PKDD'07), Sept. 2007
Which Null-Invariant Measure Is Better?

- **IR (Imbalance Ratio):** measure 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 D₄ through D₆
  - D₄ is balanced & neutral
  - D₅ is imbalanced & neutral
  - D₆ is very imbalanced & neutral

<table>
<thead>
<tr>
<th>Data</th>
<th>mc</th>
<th>( \overline{mc} )</th>
<th>( \overline{mc} )</th>
<th>( \overline{mc} )</th>
<th>all_conf.</th>
<th>max_conf.</th>
<th>Kulc.</th>
<th>cosine</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>D₁</td>
<td>10,000</td>
<td>1,000</td>
<td>1,000</td>
<td>100,000</td>
<td>0.91</td>
<td>0.91</td>
<td>0.91</td>
<td>0.91</td>
<td>0.0</td>
</tr>
<tr>
<td>D₂</td>
<td>10,000</td>
<td>1,000</td>
<td>1,000</td>
<td>100</td>
<td>0.91</td>
<td>0.91</td>
<td>0.91</td>
<td>0.91</td>
<td>0.0</td>
</tr>
<tr>
<td>D₃</td>
<td>100</td>
<td>1,000</td>
<td>1,000</td>
<td>100,000</td>
<td>0.09</td>
<td>0.09</td>
<td>0.09</td>
<td>0.09</td>
<td>0.0</td>
</tr>
<tr>
<td>D₄</td>
<td>1,000</td>
<td>1,000</td>
<td>1,000</td>
<td>100,000</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.0</td>
</tr>
<tr>
<td>D₅</td>
<td>1,000</td>
<td>100</td>
<td>10,000</td>
<td>100,000</td>
<td>0.09</td>
<td>0.91</td>
<td>0.5</td>
<td>0.29</td>
<td>0.89</td>
</tr>
<tr>
<td>D₆</td>
<td>1,000</td>
<td>10</td>
<td>100,000</td>
<td>100,000</td>
<td>0.01</td>
<td>0.99</td>
<td>0.5</td>
<td>0.10</td>
<td>0.99</td>
</tr>
</tbody>
</table>
Chapter 5: Mining Frequent Patterns, Association and Correlations: Basic Concepts and Methods

- Basic Concepts
- Scalable Frequent Itemset Mining Methods
- Are All Patterns Interesting?—Pattern Evaluation Methods
- Applications of Frequent Patterns and Associations
- Summary
Applications of Frequent Patterns and Associations

- Weblog mining
- Collaborative filtering
- Bioinformatics
Chapter 5: Mining Frequent Patterns, Association and Correlations: Basic Concepts and Methods

- Basic Concepts
- Scalable Frequent Itemset Mining Methods
- Are All Patterns Interesting?—Pattern Evaluation Methods
- Applications of Frequent Patterns and Associations
- Summary
Frequent-Pattern Mining: Summary

- Basic concepts: association rules, support-confident framework, closed and max-patterns
- Scalable frequent pattern mining methods
  - Apriori (Candidate generation & test)
  - Projection-based (FPgrowth, CLOSEST+, ...)
  - Vertical format approach (ECLAT, CHARM, ...)
- Are all patterns interesting? Pattern evaluation
- Frequent pattern mining applications
Ref: Basic Concepts of Frequent Pattern Mining


- **(Max-pattern)** R. J. Bayardo. Efficiently mining long patterns from databases. SIGMOD'98.


- **(Sequential pattern)** R. Agrawal and R. Srikant. Mining sequential patterns. ICDE'95
Ref: Apriori and Its Improvements

Ref: Depth-First, Projection-Based FP Mining

- J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. SIGMOD’ 00.
- J. Liu, Y. Pan, K. Wang, and J. Han. Mining Frequent Item Sets by Opportunistic Projection. KDD'02.
- J. Han, J. Wang, Y. Lu, and P. Tzvetkov. Mining Top-K Frequent Closed Patterns without Minimum Support. ICDM'02.
- J. Wang, J. Han, and J. Pei. CLOSET+: Searching for the Best Strategies for Mining Frequent Closed Itemsets. KDD'03.
Ref: Vertical Format and Row Enumeration Methods

- Zaki and Hsiao. CHARM: An Efficient Algorithm for Closed Itemset Mining, SDM'02.
- H. Liu, J. Han, D. Xin, and Z. Shao, Mining Interesting Patterns from Very High Dimensional Data: A Top-Down Row Enumeration Approach, SDM'06.
Ref: Mining Correlations and Interesting Rules


- P.-N. Tan, V. Kumar, and J. Srivastava. Selecting the Right Interestingness Measure for Association Patterns. KDD'02.

- E. Omiecinski. Alternative Interest Measures for Mining Associations. TKDE’03.

- T. Wu, Y. Chen and J. Han, “Association Mining in Large Databases: A Re-Examination of Its Measures”, PKDD'07
Ref: Freq. Pattern Mining Applications


- T. Dasu, T. Johnson, S. Muthukrishnan, and V. Shkapenyuk. Mining Database Structure; or How to Build a Data Quality Browser. SIGMOD'02.

- K. Wang, S. Zhou, J. Han. Profit Mining: From Patterns to Actions. EDBT’02.