Mining Frequent Patterns and Association Analysis

- Basic concepts

- Efficient and scalable frequent itemset mining methods
  - Apriori (Agrawal & Srikant@VLDB’94) and variations
  - Frequent pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD’00)
  - Algorithms using vertical format
  - Closed and maximal patterns and their mining method

- Mining various kinds of association rules

- From association mining to correlation analysis

- Constraint-based association mining
Mining Frequent Patterns Without Candidate Generation

- Basic idea: grow long patterns from short ones using local frequent items
  - “abc” is a frequent pattern
  - Get all transactions having “abc”: DB|abc
  - “d” is a local frequent item in DB|abc • abcd is a frequent pattern
- FP-Growth
  - Construct FP-tree
  - Divide compressed database into a set of conditional databases and mines them separately
Construct FP-tree from a Transaction Database

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
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—in frequent 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 count field)
  - For Connect-4 Dataset, compression ratio could be over 100
Mining Frequent Patterns With FP-trees

- **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
Partition Patterns and Databases

- Frequent patterns can be partitioned into subsets according to 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
Set Enumeration Tree of the Patterns

- Depth-first recursive search
- Pruning while building conditional patterns

\[ \Phi (fcabmp) \]

\[ p (fcabm) \quad m (fcab) \quad b (fca) \]

\[ mp (fcab) \quad bp (fca) \quad bm (fca) \]

\[ fmp (cab) \quad \ldots \quad \ldots \quad \ldots \]
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 \textit{transformed prefix paths} of item \( p \) to form \( p \)'s conditional pattern base

**Header Table**

<table>
<thead>
<tr>
<th>Item</th>
<th>Frequency</th>
<th>Header</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>

**Conditional pattern bases**

<table>
<thead>
<tr>
<th>Item</th>
<th>Cond. Pattern Base</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c )</td>
<td>( f:3 )</td>
</tr>
<tr>
<td>( a )</td>
<td>( fc:3 )</td>
</tr>
<tr>
<td>( b )</td>
<td>( fca:1, f:1, c:1 )</td>
</tr>
<tr>
<td>( m )</td>
<td>( fca:2, fcab:1 )</td>
</tr>
<tr>
<td>( p )</td>
<td>( fcam:2, cb:1 )</td>
</tr>
</tbody>
</table>
From Conditional Pattern-bases to Conditional FP-trees

- Accumulate the count for each item in the base
- Construct the FP-tree for the frequent items of the pattern base
- Repeat the process on each newly created conditional FP-tree until the resulting FP-tree is empty, or only one path

**p-conditional pattern base:**

```
fcam:2, cb:1
```

**p-conditional FP-tree (min-support = 3)**

```
\[
\begin{array}{c}
\text{Header Table} \\
\text{Item} & \text{frequency} & \text{head} \\
\hline
f & 4 & \\
c & 4 & a:3, b:3, m:3, p:3 \\
a & 3 & \\
b & 3 & \\
m & 3 & \\
p & 3 & \\
\end{array}
\]
```

All frequent patterns containing \( p \)

```
\[
\begin{array}{c}
\text{All frequent patterns containing } p \\
\{\} \\
c:3 \\
p, cp
\end{array}
\]
```
Finding Patterns Having \( m \)

- Construct \( m \)-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 only one path

\[ \textit{m-conditional pattern base:} \\
\textit{fca:2, fcab:1} \]

### 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>

*All frequent patterns relate to \( m, \ f m, \ cm, \ am, \ f cm, \ fam, \ cam, \ fc am*
FP-Growth vs. Apriori: Scalability With the Support Threshold

Data set T25I20D10K

- D1 FP-Growth runtime
- D1 Apriori runtime
Why Is FP-Growth the Winner?

- Decompose both mining task and DB and leads to focused search of smaller databases
- Use least frequent items as suffix (offering good selectivity) and find shorter patterns recursively and concatenate with suffix
Scalable Methods for Mining Frequent Patterns

- Scalable mining methods for frequent patterns
  - Apriori (Agrawal & Srikant@VLDB’94) and variations
  - Frequent pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD’00)
  - Algorithms using vertical format (ECLAT)
- Closed and maximal patterns and their mining methods
ECLAT

- For each item, store a list of transaction ids (tids)

**Horizontal Data Layout**

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A,B,E</td>
</tr>
<tr>
<td>2</td>
<td>B,C,D</td>
</tr>
<tr>
<td>3</td>
<td>C,E</td>
</tr>
<tr>
<td>4</td>
<td>A,C,D</td>
</tr>
<tr>
<td>5</td>
<td>A,B,C,D</td>
</tr>
<tr>
<td>6</td>
<td>A,E</td>
</tr>
<tr>
<td>7</td>
<td>A,B</td>
</tr>
<tr>
<td>8</td>
<td>A,B,C</td>
</tr>
<tr>
<td>9</td>
<td>A,C,D</td>
</tr>
<tr>
<td>10</td>
<td>B</td>
</tr>
</tbody>
</table>

**Vertical Data Layout**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TID-list**
ECLAT

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

\[
\begin{array}{|c|}
\hline
A \\
1 \\
4 \\
5 \\
6 \\
7 \\
8 \\
9 \\
\hline
\end{array}
\quad \quad
\begin{array}{|c|}
\hline
B \\
1 \\
2 \\
5 \\
7 \\
8 \\
10 \\
\hline
\end{array}
\end{array}
\quad \quad
\begin{array}{|c|}
\hline
AB \\
1 \\
5 \\
7 \\
8 \\
\hline
\end{array}
\]

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

- Scalable mining methods for frequent patterns
  - Apriori (Agrawal & Srikant@VLDB’94) and variations
  - Frequent pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD’00)
  - Algorithms using vertical data format (ECLAT)
- Closed and maximal patterns and their mining methods
  - Concepts
  - Max-patterns: MaxMiner, MAFIA
  - Closed patterns: CLOSEST, CLOSEST+, CARPENTER
Closed Patterns and Max-Patterns

- A long pattern contains a combinatorial number of sub-patterns, e.g., \{a_1, ..., a_{100}\} contains $2^{100} - 1$ sub-patterns!
- Solution: *Mine “boundary” patterns*
- An itemset $X$ is **closed** if $X$ is frequent and there exists *no super-pattern* $Y \supset X$, *with the same support* as $X$ (Pasquier, et al. @ ICDT’99)
- An itemset $X$ is a **max-pattern** if $X$ is frequent and there exists no frequent super-pattern $Y \supset X$ (Bayardo @ SIGMOD’98)
- Closed pattern is a lossless compression of freq. patterns and support counts
Max-patterns

- Frequent patterns without frequent super patterns
  - BCDE, ACD are max-patterns
  - E.g. BCD, AD, CD is not a max-pattern

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

Min_sup=2
Max-Patterns Illustration

An itemset is maximal frequent if none of its immediate supersets is frequent
Closed Patterns

- An itemset is closed if none of its immediate supersets has the same support as the itemset

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>{A,B}</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>{B,C,D}</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>{A,B,C,D}</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>{A,B,D}</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>{A,B,C,D}</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Itemset</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>{A}</td>
<td>4</td>
</tr>
<tr>
<td>{B}</td>
<td>5</td>
</tr>
<tr>
<td>{C}</td>
<td>3</td>
</tr>
<tr>
<td>{D}</td>
<td>4</td>
</tr>
<tr>
<td>{A,B}</td>
<td>4</td>
</tr>
<tr>
<td>{A,C}</td>
<td>2</td>
</tr>
<tr>
<td>{A,D}</td>
<td>3</td>
</tr>
<tr>
<td>{B,C}</td>
<td>3</td>
</tr>
<tr>
<td>{B,D}</td>
<td>4</td>
</tr>
<tr>
<td>{C,D}</td>
<td>3</td>
</tr>
</tbody>
</table>

- Closed patterns: B: 5, \{A,B\}: 4, \{B,D\}: 4, \{A,B,D\}: 3, \{B,C,D\}: 3, \{A,B,C,D\}: 2
Maximal vs Closed Itemsets
Example: Closed Patterns and Max-Patterns

- DB = \{a_1, \ldots, a_{100}, a_1, \ldots, a_{50}\}
  
  \text{Min	extunderscore sup} = 1

- What is the set of closed itemset?
  
  \langle a_1, \ldots, a_{100} \rangle: 1
  \langle a_1, \ldots, a_{50} \rangle: 2

- What is the set of max-pattern?
  
  \langle a_1, \ldots, a_{100} \rangle: 1

- What is the set of all patterns?
Scalable Methods for Mining Frequent Patterns

- Scalable mining methods for frequent patterns
  - Apriori (Agrawal & Srikant@VLDB’94) and variations
  - Frequent pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD’00)
  - Algorithms using vertical data format (ECLAT)
- Closed and maximal patterns and their mining methods
  - Concepts
  - Max-pattern mining: MaxMiner, MAFIA
  - Closed pattern mining: CLOSET, CLOSET+, CARPENTER
MaxMiner: Mining Max-patterns

- R. Bayardo. Efficiently mining long patterns from databases. In *SIGMOD’98*

- Idea: generate the complete set-enumeration tree one level at a time (breadth-first search), while pruning if applicable.
Algorithm MaxMiner

- Initially, generate one node $N = \Phi (ABCD)$, where $h(N) = \Phi$ and $t(N) = \{A, B, C, D\}$.
- Recursively expanding $N$
  - Local pruning
    - If $h(N) \cup t(N)$ is frequent, do not expand $N$.
    - If for some $i \in t(N)$, $h(N) \cup \{i\}$ is NOT frequent, remove $i$ from $t(N)$ before expanding $N$.
  - Global pruning
Local Pruning Techniques (e.g. at node A)

Check the frequency of ABCD and AB, AC, AD.
- If ABCD is frequent, prune the whole sub-tree.
- If AC is NOT frequent, remove C from the parenthesis before expanding.
Global Pruning Technique (across sub-trees)

- When a max pattern is identified (e.g. ABCD), prune all nodes (e.g. B, C and D) where \( h(N) \cup t(N) \) is a sub-set of it (e.g. ABCD).
Example

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

Min_sup=2

Max patterns:

<table>
<thead>
<tr>
<th>Items</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABCDEF</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
</tr>
<tr>
<td>D</td>
<td>3</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
</tr>
<tr>
<td>F</td>
<td>1</td>
</tr>
</tbody>
</table>
Example

\[ \Phi \ (ABCDEF) \]

\[ A \ (BCDE) \quad B \ (CDE) \quad C \ (DE) \quad D \ (E) \quad E \ () \]

\[ AC \ (D) \quad AD \ () \]

Min_sup = 2

Max patterns:

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

Node A

<table>
<thead>
<tr>
<th>Items</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABCDE</td>
<td>1</td>
</tr>
<tr>
<td>AB</td>
<td>1</td>
</tr>
<tr>
<td>AC</td>
<td>2</td>
</tr>
<tr>
<td>AD</td>
<td>2</td>
</tr>
<tr>
<td>AE</td>
<td>1</td>
</tr>
</tbody>
</table>
Example

Node B

<table>
<thead>
<tr>
<th>Items</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>BCDE</td>
<td>2</td>
</tr>
<tr>
<td>BC</td>
<td></td>
</tr>
<tr>
<td>BD</td>
<td></td>
</tr>
<tr>
<td>BE</td>
<td></td>
</tr>
</tbody>
</table>

Max patterns:

BCDE

Min_sup=2

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

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

- \( \Phi (ABCDEF) \)

- \( A (BCDE) \) \( B (CDE) \) \( C (DE) \) \( D (E) \) \( E () \)

- Node AC
  - Items | Frequency
  - ACD   | 2

- Max patterns:
  - BCDE
  - ACD

- Min_sup = 2
Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
- Summary
Mining Various Kinds of Association Rules

- Mining multilevel association
- Mining multidimensional association
- Mining quantitative association
- Mining other interesting associations
Mining Multiple-Level Association Rules

- Items often form hierarchies
- Multi-level association rules
  - Top down mining for different levels
  - Support threshold for each level
    - Uniform support vs. reduced support vs. group based support
- Apriori property

Uniform support

<table>
<thead>
<tr>
<th>Level 1</th>
<th>min_sup = 5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Milk</td>
<td>[support = 10%]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 2</th>
<th>min_sup = 5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>2% Milk</td>
<td>[support = 6%]</td>
</tr>
<tr>
<td>Skim Milk</td>
<td>[support = 4%]</td>
</tr>
</tbody>
</table>

Reduced support

<table>
<thead>
<tr>
<th>Level 1</th>
<th>min_sup = 5%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Milk</td>
<td>[support = 10%]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 2</th>
<th>min_sup = 3%</th>
</tr>
</thead>
<tbody>
<tr>
<td>2% Milk</td>
<td>[support = 6%]</td>
</tr>
<tr>
<td>Skim Milk</td>
<td>[support = 4%]</td>
</tr>
</tbody>
</table>
Multi-level Association Rules: Redundancy

- Some rules may be redundant due to “ancestor” relationships between items.

- Example
  - milk ⇒ wheat bread [support = 8%, confidence = 70%]
  - 2% milk ⇒ wheat bread [support = 2%, confidence = 72%]

- We say the first rule is an ancestor of the second rule.

- A rule is redundant if its support is close to the “expected” value, based on the rule’s ancestor.
Mining Multi-Dimensional Association

- Single-dimensional rules:
  \[ \text{buys}(X, \text{“milk”}) \Rightarrow \text{buys}(X, \text{“bread”}) \]

- Multi-dimensional rules: \( \geq 2 \) dimensions or predicates
  - Inter-dimension assoc. rules (no repeated predicates)
    \[ \text{age}(X,\text{“19-25”}) \land \text{occupation}(X,\text{“student”}) \Rightarrow \text{buys}(X, \text{“coke”}) \]
  - hybrid-dimension assoc. rules (repeated predicates)
    \[ \text{age}(X,\text{“19-25”}) \land \text{buys}(X, \text{“popcorn”}) \Rightarrow \text{buys}(X, \text{“coke”}) \]

- Frequent itemset -> frequent predicate set

- Treating quantitative attributes: discretization
Mining Other Interesting Patterns

- Flexible support constraints (Wang et al. @ VLDB’02)
  - Some items (e.g., diamond) may occur rarely but are valuable
  - Customized sup$_{\text{min}}$ specification and application
- Top-K closed frequent patterns (Han, et al. @ ICDM’02)
  - Hard to specify sup$_{\text{min}}$, but top-k with length$_{\text{min}}$ is more desirable
  - Dynamically raise sup$_{\text{min}}$ in FP-tree construction and mining, and select most promising path to mine
Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
- Summary
Correlation Analysis

- Association rules with strong support and confidence can be still uninteresting or even misleading
  - Buy walnuts $\Rightarrow$ buy milk [1%, 80%] misleading - 85% of customers buy milk

- Additional interestingness and correlation measures indicates the strength (and direction) of the (linear) relationship between two random variables.
  - Lift, all-confidence, coherence
  - Chi-square
  - Pearson correlation

- Correlation analysis discussed in dimension reduction
Correlation Measure: Lift

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

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

- Independent or correlated?

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

\[
lift(B, \neg C) = \frac{1000 / 5000}{3000 / 5000 \times 1250 / 5000} = 1.33
\]
Correlation Measures: All_confidence and Coherence

- Tan, Kumar, Sritastava @KDD’02

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

\[
\text{all\_conf} = \frac{\sup(X)}{\max_{\text{item}} \sup(X)} = \frac{P(A \cup B)}{\max(P(A), P(B))}
\]

\[
\text{coh} = \frac{\sup(X)}{|\text{universe}(X)|} = \frac{P(A \cup B)}{P(A) + P(B) - P(A \cup B)}
\]

- Both all-confidence and coherence have the downward closure property
Are Lift and Chi-Square Good Measures?

- Tan, Kumar, Sritastava @KDD’02, Omiecinski@TKDE’03

<table>
<thead>
<tr>
<th></th>
<th>Milk</th>
<th>No Milk</th>
<th>Sum (row)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td>m, c</td>
<td>~m, c</td>
<td>c</td>
</tr>
<tr>
<td>No Coffee</td>
<td>m, ~c</td>
<td>~m, ~c</td>
<td>~c</td>
</tr>
<tr>
<td>Sum(col.)</td>
<td>m</td>
<td>~m</td>
<td>Σ</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>m, c</th>
<th>~m, c</th>
<th>m~c</th>
<th><del>m</del>c</th>
<th>lift</th>
<th>all-conf</th>
<th>coh</th>
<th>χ²</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>1000</td>
<td>100</td>
<td>100</td>
<td>10,000</td>
<td>9.26</td>
<td>0.91</td>
<td>0.83</td>
<td>9055</td>
</tr>
<tr>
<td>A2</td>
<td>100</td>
<td>1000</td>
<td>1000</td>
<td>100,000</td>
<td>8.44</td>
<td>0.09</td>
<td>0.05</td>
<td>670</td>
</tr>
<tr>
<td>A3</td>
<td>1000</td>
<td>100</td>
<td>1000</td>
<td>100,000</td>
<td>9.18</td>
<td>0.09</td>
<td>0.09</td>
<td>8172</td>
</tr>
<tr>
<td>A4</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1</td>
<td>0.5</td>
<td>0.33</td>
<td>0</td>
</tr>
</tbody>
</table>

- **lift** and **χ²** are not good measures for large transactional DBs
- **all-confidence** or **coherence** could be good measures because they are **null-invariant** – free of influence of null transactions (~m~c)
# More Correlation Measures

<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>$\sqrt{P(A)P(B)(1-P(A))(1-P(B))}$</td>
</tr>
<tr>
<td>$Q$</td>
<td>Yule's Q</td>
<td>-1 ... 1</td>
<td>$\frac{P(A,B)P(A\bar{B})-P(A\bar{B})P(A,B)}{P(A,B)P(A\bar{B})+P(A,B)P(A\bar{B})}$</td>
</tr>
<tr>
<td>$Y$</td>
<td>Yule’s Y</td>
<td>-1 ... 1</td>
<td>$\frac{\sqrt{P(A,B)P(A\bar{B})}-\sqrt{P(A,B)P(A\bar{B})}}{P(A,B)+P(A,B)-P(A)P(\bar{B})}$</td>
</tr>
<tr>
<td>$k$</td>
<td>Cohen’s</td>
<td>-1 ... 1</td>
<td>$\frac{P(A,B)-P(A)P(B)}{1-P(A)P(B)-P(A)P(\bar{B})}$</td>
</tr>
<tr>
<td>$PS$</td>
<td>Piatetsky-Shapiro’s</td>
<td>-0.25 ... 0.25</td>
<td>$\max\left{ \frac{P(B\mid A)-P(B)}{1-P(B)}, \frac{P(B\mid \bar{A})-P(B)}{1-P(B)} \right}$</td>
</tr>
<tr>
<td>$F$</td>
<td>Certainty factor</td>
<td>-1 ... 1</td>
<td>$\frac{\sqrt{P(A,B)\max{P(B\mid A)-P(B), P(B\mid B)-P(A)}}}{\Sigma_j \max_k P(A_j,B_k)+\Sigma_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}$</td>
</tr>
<tr>
<td>$AV$</td>
<td>added value</td>
<td>-0.5 ... 1</td>
<td>$\min{-\Sigma_i P(A_i)\log P(A_i)\log P(\bar{A}_i)}$</td>
</tr>
<tr>
<td>$K$</td>
<td>Klosgen’s Q</td>
<td>-0.33 ... 0.38</td>
<td>$\max{P(B\mid A)-P(B), P(A\mid B)-P(A)}$</td>
</tr>
<tr>
<td>$g$</td>
<td>Goodman-kurskal’s</td>
<td>0 ... 1</td>
<td>$\frac{2-\max_j P(A_j)-\max_k P(B_k)}{P(A,B)}$</td>
</tr>
<tr>
<td>$M$</td>
<td>Mutual Information</td>
<td>0 ... 1</td>
<td>$\frac{\min{-\Sigma_i P(A_i)\log P(A_i)\log P(\bar{A}_i)}}{\max{P(B\mid A)-P(B), P(A\mid B)-P(A)}}$</td>
</tr>
<tr>
<td>$J$</td>
<td>J-Measure</td>
<td>0 ... 1</td>
<td>$\frac{P(A,B)\log\frac{P(B\mid A)}{P(A)}}{P(B)\log\frac{P(B\mid A)}{P(A)}}$</td>
</tr>
<tr>
<td>$G$</td>
<td>Gini index</td>
<td>0 ... 1</td>
<td>$\max{P(B\mid A)^2 + P(\bar{A}\mid A)^2}$</td>
</tr>
<tr>
<td>$s$</td>
<td>support</td>
<td>0 ... 1</td>
<td>$\max\left{ \frac{NP(A,B)+1}{NP(A,B)+2}, \frac{NP(A,B)+1}{NP(B)+2} \right}$</td>
</tr>
<tr>
<td>$c$</td>
<td>confidence</td>
<td>0 ... 1</td>
<td>$\frac{P(A,B)}{P(A,B)+P(\bar{A}\mid B)}$</td>
</tr>
<tr>
<td>$L$</td>
<td>Laplace</td>
<td>0 ... 1</td>
<td>$\frac{\max{P(A,B), P(\bar{A}\mid B)}}{P(A,B)}$</td>
</tr>
<tr>
<td>$IS$</td>
<td>Cosine</td>
<td>0 ... 1</td>
<td>$\frac{P(A,B)}{P(A,B)+P(\bar{A}\mid B)}$</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>coherence (Jaccard)</td>
<td>0 ... 1</td>
<td>$\frac{P(A)+P(\bar{B})-P(A,B)}{P(A)+P(\bar{B})}$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>all-confidence</td>
<td>0 ... 1</td>
<td>$\max{P(A,B), P(\bar{A}\mid B)}$</td>
</tr>
<tr>
<td>$\phi$</td>
<td>odds ratio</td>
<td>0 ... $\infty$</td>
<td>$\frac{\max{P(A,B), P(\bar{A}\mid B)}}{P(A)}$</td>
</tr>
<tr>
<td>$V$</td>
<td>Conviction</td>
<td>0.5 ... $\infty$</td>
<td>$\max\left{\frac{P(\bar{A}\mid B)}{P(A)}, \frac{P(B\mid \bar{A})}{P(B)}\right}$</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>lift</td>
<td>0 ... $\infty$</td>
<td>$\frac{P(A,B)}{P(A)P(B)}$</td>
</tr>
<tr>
<td>$S$</td>
<td>Collective strength</td>
<td>0 ... $\infty$</td>
<td>$\frac{P(A,B)+P(\bar{A}\mid B)\times (1-P(A)P(B)-P(A)P(\bar{B})}{1-P(A,B)-P(AB)}$</td>
</tr>
<tr>
<td>$\chi^2$</td>
<td>$\chi^2$</td>
<td>0 ... $\infty$</td>
<td>$\sum_i \frac{(P(A_i)-E_i)^2}{E_i}$</td>
</tr>
</tbody>
</table>
Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
Constraint-based (Query-Directed) Mining

- Finding all the patterns in a database autonomously? — unrealistic!
  - The patterns could be too many but not focused!
- Data mining should be an interactive process
  - User directs what to be mined using a data mining query language (or a graphical user interface)
- Constraint-based mining
  - User flexibility: provides constraints on what to be mined
  - System optimization: explores such constraints for efficient mining—constraint-based mining
Constraints in Data Mining

- **Knowledge type constraint:**
  - association, correlation, etc.

- **Data constraint** — using SQL-like queries
  - find product pairs sold together in stores in **Chicago** in Dec.’02

- **Dimension/level constraint**
  - in relevance to **region, price, brand, customer category**

- **Interestingness constraint (support, confidence, correlation)**
  - min_support $\geq$ 3%, min_confidence $\geq$ 60%

- **Rule (or pattern) constraint**
  - small sales (price < $10) triggers big sales (sum > $200)
Constrained Mining

- Rule constraints as metarules specifies the syntactic form of rules
- Constrained mining
  - Finding all patterns satisfying constraints
- Constraint pushing
  - Shares a similar philosophy as pushing selections deeply in query processing
  - What kind of constraints can be pushed?
- Constraints
  - Anti-monotonic
  - Monotonic
  - Succinct
  - Convertible
Frequent-Pattern Mining: Summary

- Frequent pattern mining—an important task in data mining
- Scalable frequent pattern mining methods
  - Apriori (Candidate generation & test)
  - Projection-based (FPgrowth, CLOSET+, ...)
  - Vertical format approach (ECLAT, ...)
  - Max and closed pattern mining
- Mining various kinds of rules
- Correlation analysis
- Constraint-based mining