# Association Analysis (5)

# Evaluation of Association Patterns

- Association analysis algorithms have the potential to generate a large number of patterns.
  - In real commercial databases we could easily end up with thousands or even millions of patterns, many of which might not be interesting.

- Very important to establish a set of well accepted criteria for evaluating the quality of association patterns.

- **First set** of criteria can be established through statistical arguments.
  - Patterns involving mutually independent items or cover very few transactions are considered uninteresting because they may capture spurious relationships in the data [**confidence, support**].
  - Will talk also for **interest factor**.

- **Second set** of criteria can be established through subjective arguments.

# Subjective Arguments

- A pattern is considered subjectively uninteresting unless it reveals unexpected information about the data.

- E.g., the rule {Butter} → {Bread} isn't interesting, despite having high support and confidence values.

- On the other hand, the rule {Diapers} → {Beer} is interesting because the relationship is quite unexpected and may suggest a new crossselling opportunity for retailers.

- **Drawback**: Incorporating subjective knowledge into pattern evaluation is a difficult task because it requires a considerable amount of prior information from the domain experts.

# Computing Interestingness Measures

- Given a rule $X \rightarrow Y$, the information needed to compute rule **interestingness** can be obtained from a **contingency table**

Contingency table for $X \rightarrow Y$

| | $Y$ | $\overline{Y}$ | |
|---|---|---|---|
| $X$ | $f_{11}$ | $f_{10}$ | $f_{1+}$ |
| $\overline{X}$ | $f_{01}$ | $f_{00}$ | $f_{0+}$ |
| | $f_{+1}$ | $f_{+0}$ | $\|T\|$ |

$f_{11}$: support of X and Y
$f_{10}$: support of X and $\overline{Y}$
$f_{01}$: support of $\overline{X}$ and Y
$f_{00}$: support of $\overline{X}$ and $\overline{Y}$

# Pitfall of Confidence

| | **Coffee** | **¬Coffee** | |
|---|---|---|---|
| **Tea** | 150 | 50 | 200 |
| **¬Tea** | 750 | 150 | 900 |
| | 900 | 200 | 1100 |

*The pitfall of confidence can be traced to the fact that the measure ignores the support of the itemset in the rule consequent.*

Consider association rule: Tea $\rightarrow$ Coffee

Confidence=

P(Coffee,Tea)/P(Tea) = P(Coffee|Tea) = 150/200 = 0.75 (seems quite high)

But, P(Coffee) = 0.9

Thus knowing that a person is a tea drinker actually decreases his/her probability of being a coffee drinker from **90%** to **75%**!

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

In fact P(Coffee|¬Tea) =

P(Coffee, ¬Tea)/P(¬Tea) = 750/900 = 0.83

# Statistical Independence

- Population of 1000 students
- 600 students know how to swim (S)
- 700 students know how to bike (B)
- 420 students know how to swim and bike (S,B)

- $P(S|B) = P(S)$ ( $P(S \wedge B)/P(B) = .42 / .7 = .6 = P(S)$ )
- $P(S \wedge B)/P(B) = P(S)$

- $P(S \wedge B) = P(S) \times P(B)$ => Statistical independence
- $P(S \wedge B) > P(S) \times P(B)$ => Positively correlated
  - i.e. if someone knows how to swim, then it is more probable he knows how to bike, and vice versa
- $P(S \wedge B) < P(S) \times P(B)$ => Negatively correlated
  - i.e. if someone knows how to swim, then it is less probable he/she knows how to bike, and vice versa

# Interest Factor

- Measure that takes into account statistical dependence

$$Interest = \frac{P(X,Y)}{P(X)P(Y)} = \boxed{\frac{f_{11}/N}{\left(f_{1+}/N\right)\times\left(f_{+1}/N\right)} = \frac{N \times f_{11}}{f_{1+} \times f_{+1}}}$$

- Interest factor compares the frequency of a pattern against a baseline frequency computed under the statistical independence assumption.
- The **baseline** frequency for a pair of mutually independent variables is:

$$\frac{f_{11}}{N} = \frac{f_{1+}}{N} \times \frac{f_{+1}}{N}$$ Or equivalently $$f_{11} = \frac{f_{1+} \times f_{+1}}{N}$$

# Interest Equation

- Previous equation follows from the standard approach of using simple fractions as estimates for probabilities.
- The fraction $f_{11}/N$ is an estimate for the joint probability P(A,B), while $f_{1+}/N$ and $f_{+1}/N$ are the estimates for P(A) and P(B), respectively.
- If A and B are statistically independent, then $P(A,B)=P(A)\times P(B)$, thus the Interest is 1.

$$I(A,B)\begin{cases} =1, & \text{if } A \text{ and } B \text{ are independent;} \\ >1, & \text{if } A \text{ and } B \text{ are positively correlated;} \\ <1, & \text{if } A \text{ and } B \text{ are negatively correlated.} \end{cases}$$

# Example: Interest

| | **Coffee** | **¬Coffee** | |
|---|---|---|---|
| **Tea** | 150 | 50 | 200 |
| **¬Tea** | 750 | 150 | 900 |
| | 900 | 200 | 1100 |

Association Rule: Tea $\rightarrow$ Coffee

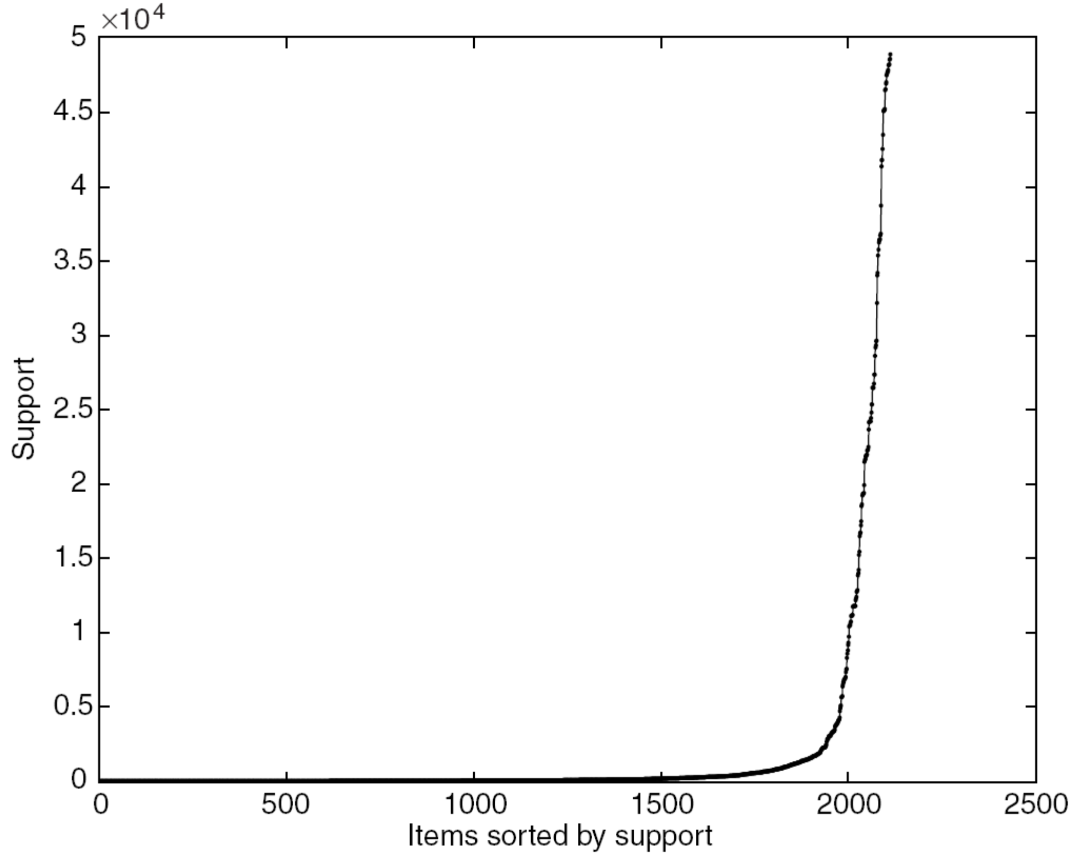
Interest =

150*1100 / (200*900)= 0.92

(< 1, therefore they are negatively correlated)

# Effect of Support Distribution

- Many real data sets have skewed support distribution where most of the items have relatively low to moderate frequencies, but a small number of them have very high frequencies.



# Skewed distribution

| Group | $G_1$ | $G_2$ | $G_3$ |
|---|---|---|---|
| Support | $< 1\%$ | $1\% - 90\%$ | $> 90\%$ |
| Number of Items | 1735 | 358 | 20 |

- Tricky to choose the right support threshold for mining such data sets.
- If we set the threshold too high (e.g., 20%), then we may miss many interesting patterns involving the low support items from G1.
  - Such low support items may correspond to expensive products (such as jewelry) that are seldom bought by customers, but whose patterns are still interesting to retailers.

- Conversely, when the threshold is set too low, there is the risk of generating spurious patterns that relate a high-frequency item such as milk to low-frequency item such as caviar.

# Cross support patterns

- They are patterns that relate a high frequency item such as milk to a low frequency item such as caviar.
  - Likely to be spurious because their correlations tend to be weak.
  - Large number of weakly correlated cross support patterns can be generated when the support threshold is sufficiently low.

- E.g. the confidence of {caviar}→{milk} is likely to be high, but still the pattern is spurious, since there isn't probably any correlation between caviar and milk.

- So, we want to **detect** cross-support patterns by looking at some antimonotone property (such as APRIORI).
  - We don't want to use "interest" as a measure because it doesn't have an antimonotone property; it's rather used a post processing evaluation measure.
  - Towards this, a definition comes next.

# Crosssupport patterns

**Definition**

A crosssupport pattern is an itemset $X = \{i_1, i_2, \ldots, i_k\}$ whose support ratio

$$r(X) = \frac{\min\{s(i_1), s(i_2), \ldots, s(i_k)\}}{\max\{s(i_1), s(i_2), \ldots, s(i_k)\}}$$

is less than a user specified threshold $h_c$.

**Example**

Suppose the support for milk is 70%, while the support for sugar is 10% and caviar is 0.04%

Given $h_c = 0.01$, the frequent itemset {milk, sugar, caviar} is a crosssupport pattern because its support ratio is

$$r = \min\{0.7, 0.1, 0.0004\} / \max\{0.7, 0.1, 0.0004\}$$

$$= 0.0004 / 0.7 = 0.00058 < 0.01$$

# Detecting crosssupport patterns

- E.g. assuming that $h_c = 0.3$, the itemsets {p,q}, {p,r}, and {p,q,r} are crosssupport patterns.
  - Because their support ratios, which are equal to 0.2, are less than the threshold $h_c$.

- We can apply a high support threshold, say, 20%, to eliminate the crosssupport patterns…but,

  this may come at the expense of discarding other interesting patterns such as the strongly correlated itemset {q,r} that has support equal to 16.7%.

| p | q | r |
|---|---|---|
| 0 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 1 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |

# Lowest confidence rule

- Notice that the rule $\{p\} \rightarrow \{q\}$ has very low confidence because most of the transactions that contain p do not contain q.

- This observation suggests that:

  Crosssupport patterns can be detected by examining the lowest confidence rule that can be extracted from a given itemset.

# Finding lowest confidence

- Recall the antimonotone property of confidence:
  conf( $\{i_1, i_2\} \rightarrow \{i_3, i_4, \ldots, i_k\}$ ) $\leq$ conf( $\{i_1, i_2, i_3\} \rightarrow \{i_4, \ldots, i_k\}$ )

- This property suggests that confidence never increases as we shift more items from the left to the righthand side of an association rule.

- Hence, the lowest confidence rule that can be extracted from a frequent itemset contains only **one item** on its lefthand side.

# Finding lowest confidence

- Given a frequent itemset $\{i_1,i_2,i_3,i_4,\ldots,i_k\}$, the rule

  $\{i_j\}\rightarrow\{i_1,i_2,i_3,i_{j-1},i_{j+1},i_4,\ldots,i_k\}$

  has the lowest confidence if ?

  $s(i_j) = \max\{s(i_1), s(i_2),\ldots,s(i_k)\}$

- This follows directly from the definition of confidence as the ratio between the rule's support and the support of the rule antecedent.

# Finding lowest confidence

- Summarizing, the lowest confidence attainable from a frequent itemset $\{i_1,i_2,i_3,i_4,\ldots,i_k\}$, is

$$\frac{s(\{i_1, i_2,\ldots, i_k\})}{\max\{s(i_1), s(i_2),\ldots, s(i_k)\}}$$

- This is also known as the **h-confidence** measure or **all-confidence** measure.
- Because of the antimonotone property of support, the numerator of the hconfidence measure is bounded by the minimum support of any item that appears in the frequent itemset. So,

$$\text{h-confidence} = \frac{s(\{i_1, i_2,\ldots, i_k\})}{\max\{s(i_1), s(i_2),\ldots, s(i_k)\}} \le \frac{\min\{s(i_1), s(i_2),\ldots, s(i_k)\}}{\max\{s(i_1), s(i_2),\ldots, s(i_k)\}} = r(\ldots)$$

# hconfidence

- Clearly, crosssupport patterns can be eliminated by ensuring that the hconfidence values for the patterns exceed $h_c$.

- Finally, observe that the measure is also antimonotone, i.e.,

$$\text{hconfidence}(\{i_1, i_2, \ldots, i_k\}) \geq \text{hconfidence}(\{i_1, i_2, \ldots, i_{k+1}\})$$

and thus can be incorporated directly into the mining algorithm.