Requirements/Challenges in Data Mining (Con’t)

- Data source issues:
  - Diversity of data types
    - Handling complex types of data
    - Mining information from heterogeneous databases and global information systems.
    - Is it possible to expect a DM system to perform well on all kinds of data? (distinct algorithms for distinct data sources)
  - Data glut
    - Are we collecting the right data with the right amount?
    - Distinguish between the data that is important and the data that is not.

• Other issues
  - Integration of the discovered knowledge with existing knowledge: A knowledge fusion problem.

Data Mining

- Needing More than just Information Retrieval
- Elementary Concepts
- Patterns and Rules to be Discovered
- Requirements and Challenges

Association Rule Mining
- Classification
- Clustering

Basic Concepts

A transaction is a set of items: \( T=\{i_a, i_b, \ldots, i_t\} \)

\( T \subseteq I \), where \( I \) is the set of all possible items \( \{i_1, i_2, \ldots, i_n\} \)

\( D \), the task relevant data, is a set of transactions.

An association rule is of the form:
\( P \Rightarrow Q \), where \( P \subseteq I \), \( Q \subseteq I \), and \( P \cap Q = \emptyset \)
Basic Concepts (con’t)

P⇒Q holds in D with support s
and
P⇒Q has a confidence c in the transaction set D.

Support(P⇒Q) = Probability(P ∪ Q)
Confidence(P⇒Q) = Probability(Q / P)

Itemsets

A set of items is referred to as itemset.

An itemset containing k items is called k-itemset.

An items set can also be seen as a conjunction of items (or a predicate)

Rule Measures: Support and Confidence

• Support of a rule P → Q
  = Support of (P ∪ Q) in D
  - s_D(P → Q) = s_D(P ∪ Q): percentage of transactions in D containing P and Q.
  (#transactions containing P and Q divided by cardinality of D).

• Confidence of a rule P → Q
  - c_D(P → Q) = s_D(P ∪ Q) / s_D(P): percentage of transactions that contain both P and Q in the subset of transactions that contain already P.

Strong Rules

• Thresholds:
  - minimum support: minsup
  - minimum confidence: minconf

• Frequent itemset P
  - support of P larger than minimum support,

• Strong rule P → Q (c%)
  - (P ∪ Q) frequent,
  - c is larger than minimum confidence.
### Mining Association Rules

<table>
<thead>
<tr>
<th>Transaction ID</th>
<th>Items Bought</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>A,B,C</td>
</tr>
<tr>
<td>1000</td>
<td>A,C</td>
</tr>
<tr>
<td>4000</td>
<td>A,D</td>
</tr>
<tr>
<td>5000</td>
<td>B,E,F</td>
</tr>
</tbody>
</table>

#### Frequent Itemset Support

- \{A\} 75%
- \{B\} 50%
- \{C\} 50%
- \{A,C\} 50%

Min. support 50%
Min. confidence 50%

For rule \{A\} \rightarrow \{C\}:
- support = support(\{A, C\}) = 50%
- confidence = support(\{A, C\})/support(\{A\}) = 66.6%

For rule \{C\} \rightarrow \{A\}:
- support = support(\{A, C\}) = 50%
- confidence = support(\{A, C\})/support(\{C\}) = 100.0%

### How do we Mine Association Rules?

- **Input**
  - A database of transactions
  - Each transaction is a list of items (Ex. purchased by a customer in a visit)
  - Find all strong rules that associate the presence of one set of items with that of another set of items.

  - Example: **98% of people who purchase tires and auto accessories also get automotive services done**
  - There are no restrictions on the number of items in the head or body of the rule.

### Mining Frequent Itemsets: the Key Step

- Iteratively find the frequent itemsets, i.e. sets of items that have minimum support, with cardinality from 1 to \(k\) \((k\text{-itemsets})\)
- Based on the **Apriori principle**: **Any subset of a frequent itemset must also be frequent.**
  - E.g., if \{AB\} is a frequent itemset, both \{A\} and \{B\} must be frequent itemsets.
- Use the frequent itemsets to generate association rules.

### The Apriori Algorithm

\(C_k\): Candidate itemset of size \(k\)

\(L_k\): frequent itemset of size \(k\)

\(L_j = \{\text{frequent items}\};\)

\(\text{for } (k = 1; L_k \neq \emptyset; k++) \text{ do begin}\)

\(C_{k+1} = \text{candidates generated from } L_k;\)

\(\text{for each transaction } t \text{ in database do}\)

\(\text{increment the count of all candidates in } C_{k+1} \text{ that are contained in } t;\)

\(L_{k+1} = \text{candidates in } C_{k+1} \text{ with min_support}\)

\(\text{end}\)

\(\text{return } \bigcup_k L_k;\)
The Apriori Algorithm -- Example

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>1 3 4</td>
</tr>
<tr>
<td>200</td>
<td>2 3 5</td>
</tr>
<tr>
<td>300</td>
<td>1 2 3 5</td>
</tr>
<tr>
<td>400</td>
<td>2 5</td>
</tr>
</tbody>
</table>

Scan D

\[ C_1 \]

<table>
<thead>
<tr>
<th>itemset</th>
<th>sup.</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1}</td>
<td>2</td>
</tr>
<tr>
<td>{2}</td>
<td>3</td>
</tr>
<tr>
<td>{3}</td>
<td>3</td>
</tr>
<tr>
<td>{4}</td>
<td>1</td>
</tr>
<tr>
<td>{5}</td>
<td>3</td>
</tr>
</tbody>
</table>

\[ L_1 \]

<table>
<thead>
<tr>
<th>itemset</th>
<th>sup.</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1}</td>
<td>2</td>
</tr>
<tr>
<td>{2}</td>
<td>3</td>
</tr>
<tr>
<td>{3}</td>
<td>3</td>
</tr>
<tr>
<td>{5}</td>
<td>3</td>
</tr>
</tbody>
</table>

Scan D

\[ C_2 \]

<table>
<thead>
<tr>
<th>itemset</th>
<th>sup.</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1 2}</td>
<td>1</td>
</tr>
<tr>
<td>{1 3}</td>
<td>2</td>
</tr>
<tr>
<td>{1 5}</td>
<td>1</td>
</tr>
<tr>
<td>{2 3}</td>
<td>2</td>
</tr>
<tr>
<td>{2 5}</td>
<td>3</td>
</tr>
<tr>
<td>{3 5}</td>
<td>2</td>
</tr>
</tbody>
</table>

Scan D

\[ C_3 \]

<table>
<thead>
<tr>
<th>itemset</th>
<th>sup.</th>
</tr>
</thead>
<tbody>
<tr>
<td>{2 3 5}</td>
<td>2</td>
</tr>
</tbody>
</table>

Note: \{1, 2, 3\} \{1, 2, 5\} and \{1, 3, 5\} not in \[ C_3 \]

Generating Association Rules from Frequent Itemsets

- Only strong association rules are generated.
- Frequent itemsets satisfy minimum support threshold.
- Strong AR satisfy minimum confidence threshold.

\[ \text{Confidence}(P \Rightarrow Q) = \frac{\text{Prob}(Q|P)}{\text{Prob}(P)} = \frac{\text{Support}(P \cup Q)}{\text{Support}(P)} \]

For each frequent itemset, \( f \), generate all non-empty subsets of \( f \). For every non-empty subset \( s \) of \( f \) do:

\[ \text{output rule } s \Rightarrow (f-s) \text{ if support}(f)/\text{support}(s) \geq \text{min\_confidence} \]

end

Data Mining

- Needing More than just Information Retrieval
- Elementary Concepts
- Patterns and Rules to be Discovered
- Requirements and Challenges
- Association Rule Mining
  - Classification
  - Clustering

What is Classification?

The goal of data classification is to organize and categorize data in distinct classes.

- A model is first created based on the data distribution.
- The model is then used to classify new data.
- Given the model, a class can be predicted for new data.