Association Rules Fundamentals

Elena Baralis, Tania Cerquitelli, Silvia Chiusano
Politecnico di Torino

Association rules

Objective
- extraction of frequent correlations or pattern from a transactional database

Tickets at a supermarket counter

<table>
<thead>
<tr>
<th>TID</th>
<th>Items</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bread, Coke, Milk</td>
</tr>
<tr>
<td>2</td>
<td>Coke, Bread</td>
</tr>
<tr>
<td>3</td>
<td>Coke, Diapers, Milk</td>
</tr>
<tr>
<td>4</td>
<td>Coke, Bread, Diapers, Milk</td>
</tr>
<tr>
<td>5</td>
<td>Coke, Diapers, Milk</td>
</tr>
</tbody>
</table>

Association rule: diapers \(\Rightarrow\) beer
- 2% of transactions contain both items
- 30% of transactions containing diapers also contain beer

Association rule mining

- A collection of transactions is given
  - a transaction is a set of items
  - items in a transaction are not ordered
- Association rule: \(A, B \Rightarrow C\)
  - \(A, B\) = items in the rule body
  - \(C\) = item in the rule head
- The \(\Rightarrow\) means co-occurrence
  - not causality
- Example: coke, diapers \(\Rightarrow\) milk

Transactional formats

- Association rule extraction is an exploratory technique that can be applied to any data type
- A transaction can be any set of items
  - Market basket data
  - Textual data
  - Structured data
  - ...

Transactional formats

Textual data
- A document is a transaction
- Words in a document are items in the transaction

Data example
- Doc1: algorithm analysis customer data mining relationship
- Doc2: customer data management relationship
- Doc3: analysis customer data mining relationship social

Rule example
- customer, relationship \(\Rightarrow\) data, mining

Example from: Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006
**Definitions**

- **Itemset** is a set including one or more items
  - Example: \{Beer, Diapers\}

- **k-itemset** is an itemset that contains \(k\) items
  - Example: \(\#(\{\text{Beer, Diapers}\}) = 2\)

- **Support** is the fraction of transactions that contain an itemset
  - Example: \(\text{sup}(\{\text{Beer, Diapers}\}) = 2/5\)

- **Frequent itemset** is an itemset whose support is greater than or equal to a \(\minsup\) threshold

**Rule quality metrics**

- Given the association rule \(A \Rightarrow B\)
  - \(A, B\) are itemsets
  - **Support** is the fraction of transactions containing both \(A\) and \(B\)
    \[ \frac{\#(A, B)}{|T|} \]
  - \(|T|\) is the cardinality of the transactional database
  - a priori probability of itemset \(AB\)
  - rule frequency in the database

- **Confidence** is the frequency of \(B\) in transactions containing \(A\)
  \[ \frac{\text{sup}(A, B)}{\text{sup}(A)} \]

**Rule quality metrics: example**

- From itemset \{Milk, Diapers\} the following rules may be derived
  - Rule: \(\text{Milk} \Rightarrow \text{Diapers}\)
    - support \(\text{sup} = 3/5 = 60\%
    - confidence \(\text{conf} = 3/4 = 75\%
  - Rule: \(\text{Diapers} \Rightarrow \text{Milk}\)
    - same support \(s = 60\%
    - confidence \(\text{conf} = 3/3 = 100\%

**Association rule extraction**

- Given a set of transactions \(T\), association rule mining is the extraction of the rules satisfying the constraints
  - support \(\geq \minsup\) threshold
  - confidence \(\geq \minconf\) threshold

- The result is
  - complete (all rules satisfying both constraints)
  - correct (only the rules satisfying both constraints)

- May add other more complex constraints

**Association rule extraction**

1. **Extraction of frequent itemsets**
   - many different techniques
     - level-wise approaches (Apriori, ...)
     - approaches without candidate generation (FP-growth, ...)
     - other approaches
     - most computationally expensive step
     - limit extraction time by means of support threshold

2. **Extraction of association rules**
   - generation of all possible binary partitioning of each frequent itemset
     - possibly enforcing a confidence threshold
Given d items, there are $2^d$ possible candidate itemsets.

**Brute-force approach**
- each itemset in the lattice is a candidate frequent itemset
- scan the database to count the support of each candidate
- match each transaction against every candidate
- Complexity $\sim O(|T| \cdot 2^d \cdot w)$
  - $|T|$ is number of transactions
  - $d$ is number of items
  - $w$ is transaction length

**Improving Efficiency**
- Reduce the number of candidates
  - Prune the search space
    - complete set of candidates is $2^d$
- Reduce the number of transactions
  - Prune transactions as the size of itemsets increases
  - reduce $|T|$
- Reduce the number of comparisons
  - Equal to $|T| \cdot 2^d$
  - Use efficient data structures to store the candidates or transactions

"If an itemset is frequent, then all of its subsets must also be frequent"  
- The support of an itemset can never exceed the support of any of its subsets
- It holds due to the antimonotone property of the support measure
  - Given two arbitrary itemsets $A$ and $B$
    - if $A \subseteq B$ then $\sup(A) \geq \sup(B)$
- It reduces the number of candidates

**Factors Affecting Performance**
- Minimum support threshold
  - lower support threshold increases number of frequent itemsets
  - larger number of candidates
  - larger (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 in dense data sets
  - may increase max length of frequent itemsets and traversals of hash tree
  - number of subsets in a transaction increases with its width
FP-growth Algorithm [Han00]
- Exploits a main memory compressed representation of the database, the FP-tree
  - high compression for dense data distributions
  - less so for sparse data distributions
  - complete representation for frequent pattern mining
  - enforces support constraint
- Frequent pattern mining by means of FP-growth
  - recursive visit of FP-tree
  - applies divide-and-conquer approach
  - decomposes mining task into smaller subtasks
- Only two database scans
  - count item supports + build FP-tree

Other approaches
- Many other approaches to frequent itemset extraction
  - some covered later
- May exploit a different database representation
  - represent the tidset of each item [Zak00]

Maximal vs Closed Itemsets

Effect of Support Threshold
- Selection of the appropriate minsup threshold is not obvious
  - If minsup is too high
    - itemsets including rare but interesting items may be lost
      - example: pieces of jewellery (or other expensive products)
  - If minsup is too low
    - it may become computationally very expensive
    - the number of frequent itemsets becomes very large

Interestingness Measures
- A large number of pattern may be extracted
  - rank patterns by their interestingness
- Objective measures
  - rank patterns based on statistics computed from data
    - initial framework [Agr94] only considered support and confidence
    - other statistical measures available
- Subjective measures
  - rank patterns according to user interpretation [Silb98]
    - interesting if it contradicts the expectation of a user
    - interesting if it is actionable

Confidence measure: always reliable?
- 5000 high school students are given
  - 3750 eat cereals
  - 3000 play basket
  - 2000 eat cereals and play basket
- Rule
  - play basket ⇒ eat cereals
  - sup = 40%, conf = 66.7%
- Problem caused by high frequency of rule head
  - negative correlation

Confidence measure: always reliable?
- 5000 high school students are given
  - 3750 eat cereals
  - 3000 play basket
  - 2000 eat cereals and play basket
- Rule
  - play basket ⇒ eat cereals
  - sup = 40%, conf = 66.7%
- Problem caused by high frequency of rule head
  - negative correlation
Correlation or lift

\[ r: A \Rightarrow B \]

Correlation = \[
\frac{P(A, B)}{P(A)P(B)} = \frac{\text{conf}(r)}{\text{sup}(B)}
\]

- Statistical independence
  - Correlation = 1
- Positive correlation
  - Correlation > 1
- Negative correlation
  - Correlation < 1

Example

- Association rule
  
  play basket \Rightarrow eat cereals

  has corr = 0.89
  - negative correlation
  - but rule
  
  play basket \Rightarrow not (eat cereals)

  has corr = 1.34