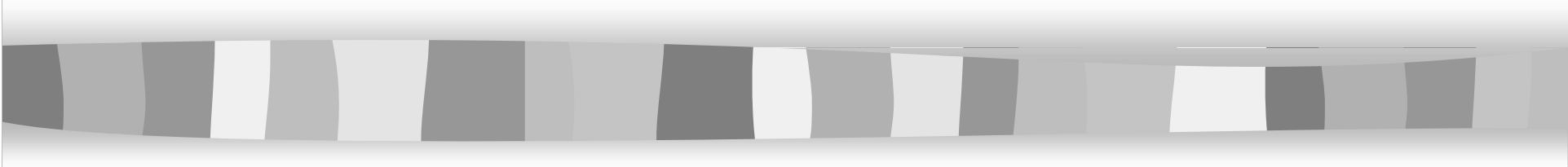


Pruning and Grouping Discovered Association Rules



H Toivonen, M Klemettinen,
P Ronkainen, K Hätönen and
H Mannila



Overview

- Association rules - review
- Why pruning
- Definitions
- Association rule covers
- Example
- Why grouping
- Grouping rules
- Conclusion and future work



Association rules (review)

- Association rule is an expression $X \Rightarrow Y$ where X and Y are sets of attributes
- Rows of the database where the attributes in X have value true, attributes in Y tend to have the value true
- **Support:** Fraction of database D containing X
- **Confidence:** Fraction containing Y given X



Why Pruning

- Large database result in large number of association rules
 - 2000 rules from course enrollment database
 - 30000 rules from telecommunications alarm database
- Not all rules with high support and confidence are interesting
 - rules corresponding to prior knowledge
 - refer to uninteresting attributes
 - present redundant information



Definitions

- X and Y are disjoint sets of attributes
- $m(X)$: Set of rows matched by attribute set X or number of elements in database containing X
- $\Gamma = \{X_i \Rightarrow Y / i=1, \dots, n\}$ is the collection of rules with the same attribute set Y

Association rule covers

- Method to reduce number of rules by eliminating redundancy

- $\Delta \subseteq \Gamma$ is a rule cover if

$$\bigcup_{(X \Rightarrow Y) \in \Gamma} m(XY) = \bigcup_{(X \Rightarrow Y) \in \Delta} m(XY)$$

- High confidence rules only
 - for lossless pruning: data set should be monotonic i.e.. If there is a matching rule for Y with a high confidence, there should not exist a more special rule with lower confidence.



Structural rule cover

- For all attribute sets X , Y and Z ,

$$m(XYZ) \subseteq m(XZ)$$

- Removal of rules like $XY \Rightarrow Z$ still results in a rule cover
- Set of rules $\Delta \subseteq \Gamma$ is a structural rule cover for Γ , if for all rules $(X \Rightarrow Y) \in \Delta$ there is no rule $(X' \Rightarrow Y) \in \Gamma$ such that $X' \subset X$



Example

- Rule 1: Programming in C, Object data Bases \Rightarrow Data Communications
- Rule 2: Programming in C, Object data bases, Computed-Supported Cooperative Work \Rightarrow Data Communications
- Rule 2 pruned from structural rule cover as redundant



Rule cover algorithm

- Input: original rule set Γ and sets of rows matched by each of these rules
- Rule cover Δ initialized to empty set
- Variable s' used to store database rows not matched by rules in Δ
- Sets s_i contain those rows in s' that are matched by the rule $X_i \Rightarrow Y$
- Iteratively the rule in Γ that matches most of the rows in s' is moved from rule set Γ to rule cover
- Process repeated until all rows matched by original rule set matched by rule cover



Why grouping

- Cover for the set of rules can still be quite large
- Set of rules in cover can be made more understandable by ordering and grouping the rules



Grouping rules

- Rules can be ordered based on interestingness
 - confidence and support factors of rules
 - derived from user-specified information
- Clustering: Grouping rules together that make statements about the same database rows
 - Clustering distance between 2 rules $X \Rightarrow Y$ and $Y \Rightarrow Z$ is the number of rows where the rules differ



Conclusion and Future Work

- Considered problem of pruning and grouping rules in order to improve the understandability of the collection of association rules
- Rule covers produced useful short descriptions of large sets of rules
- Efficient methods
- Open problem:
 - Pruning within those association rules that are not very strong
 - How to combine rules with the same consequent