Data Mining and Knowledge Discovery

Knowledge Discovery and Knowledge Management in e-Science

Petra Kralj Novak

Petra.Kralj.Novak@ijs.si

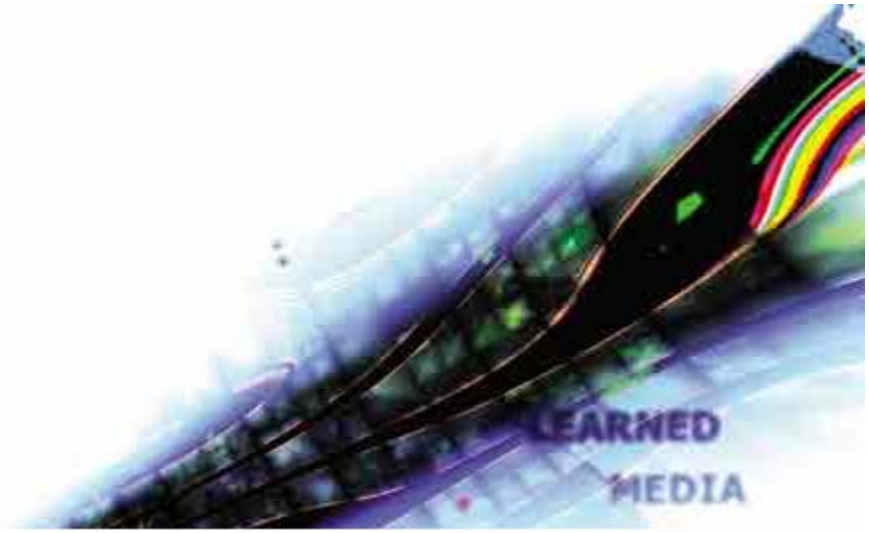
Practice, 2008/11/12

DEPARTMENT OF KNOWLEDGE TECHNOLOGIES
Jožef Stefan Institute

Practice plan

- 2008/10/22: Predictive data mining
  - Decision trees
  - Naïve Bayes classifier
  - Evaluating classifiers (separate test set, cross validation, confusion matrix, classification accuracy)
  - Predictive data mining in Weka
- 2008/11/11: Numeric prediction and descriptive data mining
  - Regression models
  - Regression models and evaluation in Weka
- 2008/11/12: Descriptive data mining
  - Association rules
  - Descriptive data mining in Weka
  - Discussion about seminars and exam
- 2008/12/1: Written exam
- 2008/12/8: Seminar proposals presentations
- 2009/01/14: Seminar presentations

Association Rules



Association rules

- Rules **X → Y**, X, Y conjunction of items
- Task: Find **all** association rules that satisfy minimum support and minimum confidence constraints

- **Support**:

  Sup(X → Y) = #XY/#D ≅ p(XY)

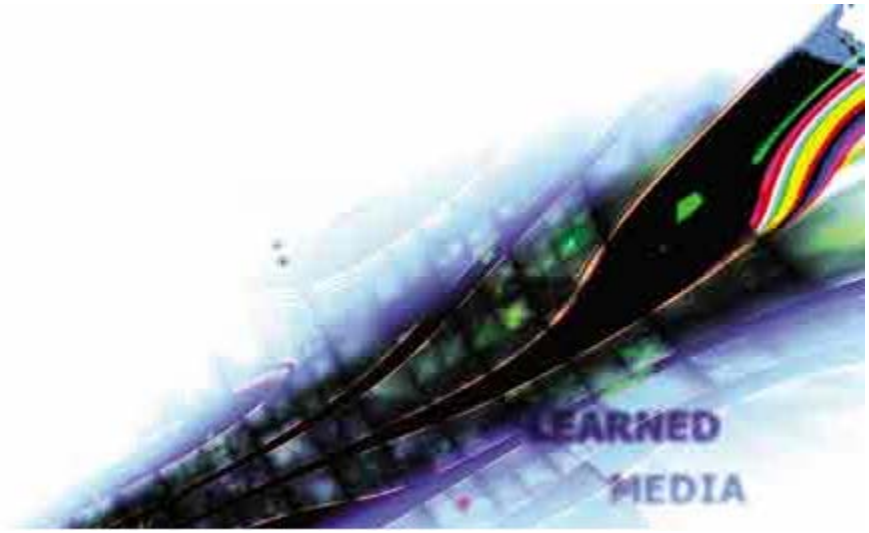
- **Confidence**:

  Conf(X → Y) = #XY/#X ≅ p(XY)/p(X) = p(Y|X)

Association rules - algorithm

1. generate frequent itemsets with a minimum support constraint
2. generate rules from frequent itemsets with a minimum confidence constraint

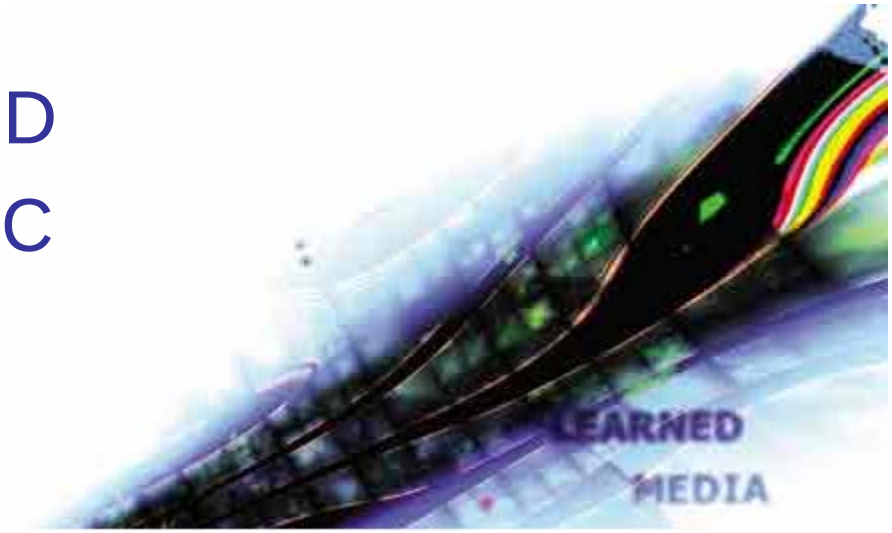
* Data are in a transaction database



Association rules – transaction database

Items: **A**=apple, **B**=banana,
C=coca-cola, **D**=doughnut

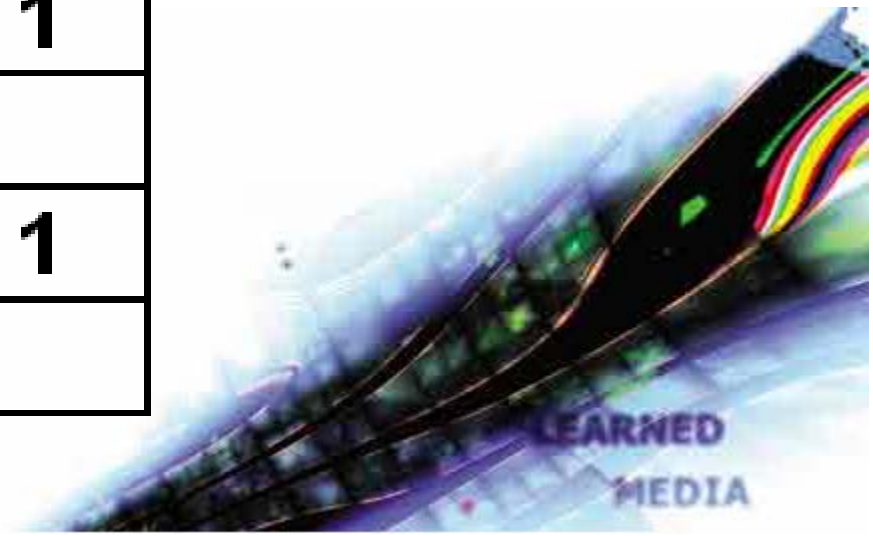
- Client 1 bought: A, B, C, D
- Client 2 bought: B, C
- Client 3 bought: B, D
- Client 4 bought: A, C
- Client 5 bought: A, B, D
- Client 6 bought: A, B, C



Frequent itemsets

- Generate frequent itemsets with support at least 2/6

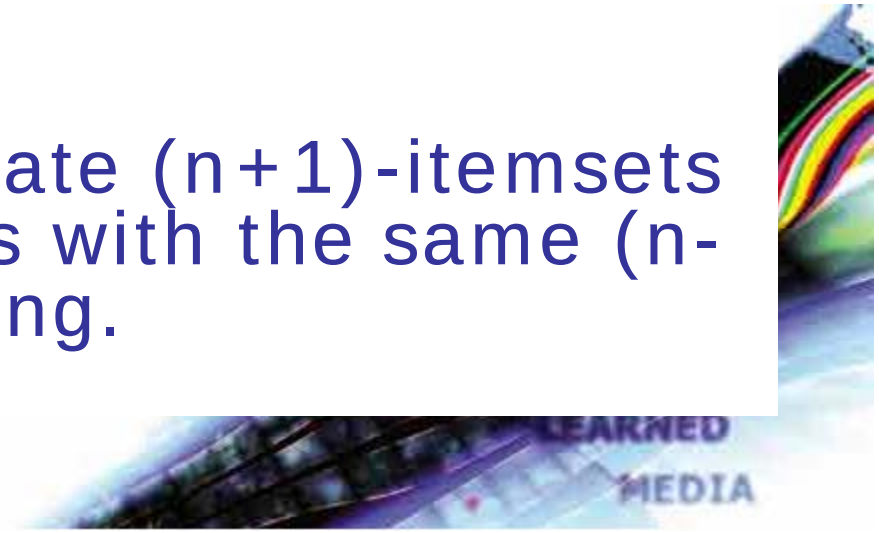
A	B	C	D
1	1	1	1
	1	1	
	1		1
1		1	
1	1		1
1	1	1	



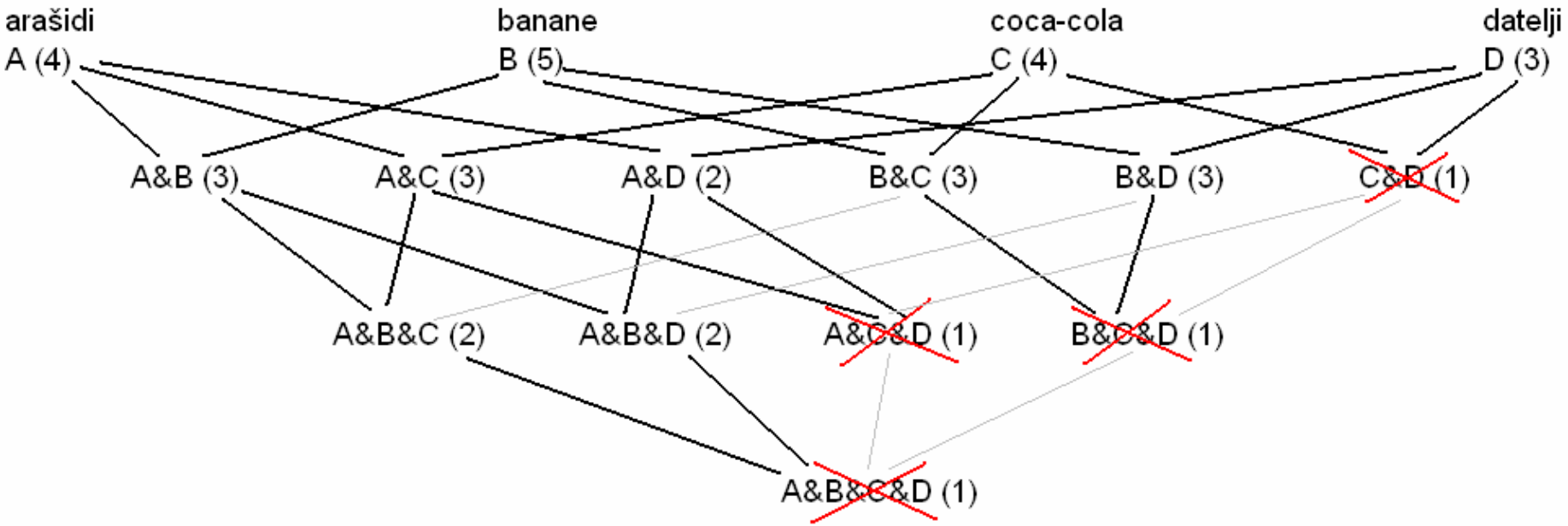
Frequent itemsets algorithm

Items in an itemset should be sorted alphabetically.

- Generate all 1-itemsets with the given minimum support.
- Use 1-itemsets to generate 2-itemsets with the given minimum support.
- From 2-itemsets generate 3-itemsets with the given minimum support as unions of 2-itemsets with the same item at the beginning.
- ...
- From n-itemsets generate (n+1)-itemsets as unions of n-itemsets with the same (n-1) items at the beginning.

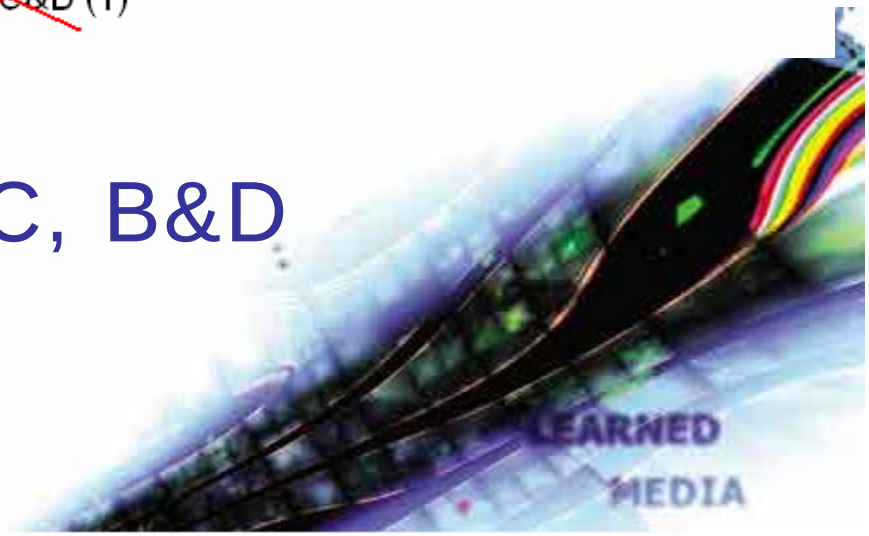


Frequent itemsets lattice



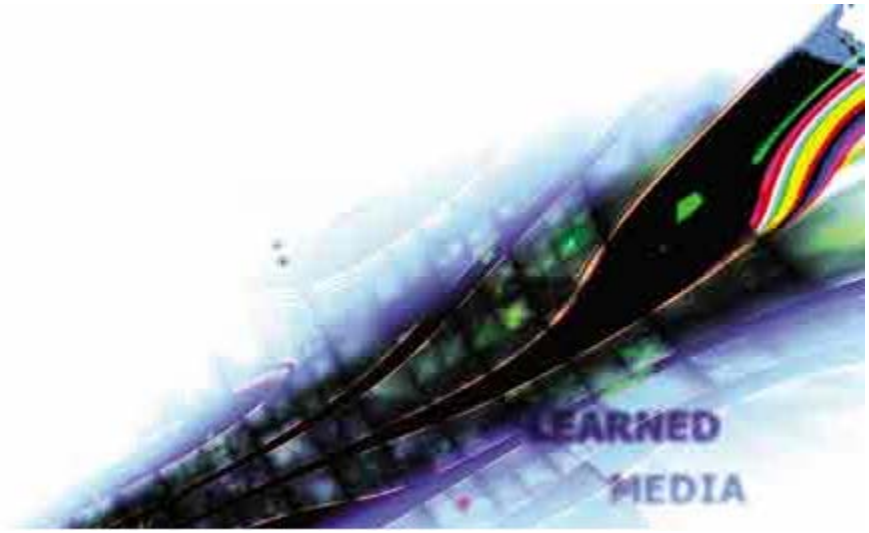
Frequent itemsets:

- A&B, A&C, A&D, B&C, B&D
- A&B&C, A&B&D



Rules from itemsets

- A&B is a frequent itemset with support 3/6
- Two possible rules
  - A→B confidence = #(A&B)/#A = 3/4
  - B→A confidence = #(A&B)/#B = 3/5
- All the counts are in the itemset lattice!



Discussion

- Transformation of an attribute-value dataset to a transaction dataset.
- What would be the association rules for a dataset with two items A and B, each of them with support 80% and appearing in the same transactions as rarely as possible?
  - minSupport = 50%, min conf = 70%
  - minSupport = 20%, min conf = 70%
- What if we had 4 items: A, ¬A, B, ¬ B
- Compare decision trees and association rules regarding handling an attribute like “PersonID”. What about attributes that have many values (eg. Month of year)

Quality of association rules

Support(X) = #X / #D ..... P(X)

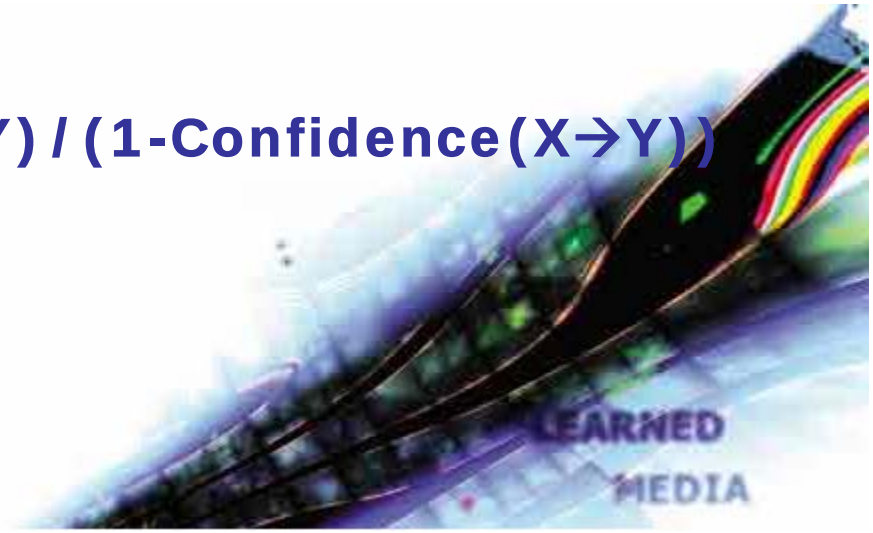
Support(X→Y) = Support (XY) #XY / #D ..... P(XY)

Confidence(X→Y) = #XY / #X ..... P(Y|X)

Lift(X→Y) = Support(X→Y) / (Support (X)*Support(Y))

Leverage(X→Y) = Support(X→Y) – Support(X)*Support(Y)

Conviction(X → Y) = 1-Support(Y)/(1-Confidence(X→Y))



Quality of association rules

Support(X) = #X / #D ..... P(X)

Support(X→Y) = Support (XY) #XY / #D ..... P(XY)

Confidence(X→Y) = #XY / #X ..... P(Y|X)

Lift(X→Y) = Support(X→Y) / (Support (X)*Support(Y))

How many more times the items in X and Y occur together then it would be expected if the itemsets were statistically independent.

Leverage(X→Y) = Support(X→Y) – Support(X)*Support(Y)

Similar to lift, difference instead of ratio.

Conviction(X → Y) = 1-Support(Y)/(1-Confidence(X→Y))

Degree of implication of a rule.

Sensitive to rule direction.