LINKÖPING
UNIVERSITY

# 723A75 Advanced Data Mining
# TDDD41 Data Mining - Clustering and Association Analysis

## Lecture 9: Summary and Exercise

Johan Alenlöv
IDA, Linköping University, Sweden

# Outline

- Content
  - Summary
  - Exercise

# Association Rules

- **Goal:** Given a **transactional database** find association rules on the form

$$\underbrace{X_1, \ldots, X_n}_{(antecedent)} \rightarrow \underbrace{Y_1, \ldots, Y_m}_{(consequent)}$$

  with a user-specified minimum **support** and **confidence**.
- **Support**: The fraction of transactions that contains the **full rule** $X \cup Y$. ($p(X \cup Y)$)
- **Confidence**: The fraction of transactions that contains $X$ that also contains $Y$. ($p(Y \mid X)$)
- **Why?** Help with decision making.
- Note that association **is not** causality.
- Two step solution:
  1. Generate **all** itemsets with a given minimum support.
  2. Generate **all** rules from these itemsets with minimum confidence.

# Apriori Property

- For generating frequent itemsets the following **apriori property** is important.
  - Every subset of a frequent itemset is frequent.
  - Alternatively every superset of an infrequent itemset is infrequent.
- Two algorithms for generating frequent itemsets

  **Apriori algorithm** Using the apriori property to generate candidate sets that are tested.
  Use the sets of length $k$ to generate and test candidates of length $k + 1$.

  **FP Grow** Construct an FP-tree and find the itemsets by looking at the conditional databases.
  Constructs itemsets by building the chains with specific suffixes first.

# Rule Generation

- Given a frequent itemset $L$ we wish to find a subset $X \subseteq L$ such that the rule $X \to L \setminus X$ has minimum confidence.
- Using the following property, if $X' \subseteq X$ then

  $$\mathrm{Conf}(X \to L \setminus X) \geq \mathrm{Conf}(X' \to L \setminus X'),$$

  we can reduce the number of sets to check.
- The algorithm goes over each subset (starting with maximal size) and then checking all subsets to find rules with minimum support.

## Constraints

- Other constraints can be added, such as minimum price, range of prices, sum of prices, etc.
- Constraints can be,

  **Monotone** If it is true for a set $X$ then it is true for every superset $X'$. $(X \subseteq X')$

  **Antimonotone** If it is true for a set $X$ then it is true for every subset $X'$. $(X \supseteq X')$

  **Convertible Monotone** If the items are sorted (in some way) then it is **monotone**.

  **Convertible Antimonotone** If the items are sorted (in some way) then it is **antimonotone**.

  **Strongly convertible** If it is both **convertible monotone** and **convertible antimonotone**.

  **Inconvertible** Can't be converted.

- Depending on the type of constraint different modifications to the algorithms are made.

# Exercise

- Run the Apriori algorithm on the following transactional database with minimum support equal to one transaction. Explain step by step the execution.

| Tid | Items |
|---|---|
| 1 | A, B, C |
| 2 | X, Y, Z |
| 3 | A, Y, C |
| 4 | X, B, Z |

- Repeat with the constraint that the itemsets has to contain A. Make it clear when the constraint is used, don't just run the algorithm and consider the constraint at the end.
- Let the items A, B, C, X, Y, and Z, have the price of respectively $-3, -2, -1, 1, 2,$ and 3 units. Repeat the exercise with the constraint: Find the frequent itemsets with range less than 3. Make it clear when the constraint is used, don't just run the algorithm and consider the constraint at the end.
- Repeat the exercises above with the FP grow algorithm
- Apply the rule generation algorithm to the frequent itemset XBZ on the database above in order to find association rules with confidence 0.5

# Solutions

For the Apriori algorithm we get the following sets, where the number indicates the support,

$$
\begin{aligned}
L_1 &= \{A, B, C, X, Y, Z\} \\
C_2 &= \{AB : 1, AC : 2, AX : 0, AY : 1, AZ : 0, BC : 1, BX : 1, BY : 0, BZ : 1, \\
&\qquad\qquad CX : 0, CY : 1, CZ : 0, XY : 1, XZ : 2, YZ : 1\} \\
L_2 &= \{AB, AC, AY, BC, BX, BZ, CY, XY, XZ, YZ\} \\
C_3 &= \{ABC : 1, ACY : 1, BXZ : 1, XYZ : 1\} \\
L_3 &= \{ABC, ACY, BXZ, XYZ\}
\end{aligned}
$$

Since “set includes $A$” is monotone we check the condition when output. We output

$$
\begin{aligned}
L_1 &= \{A\} \\
L_2 &= \{AB, AC, AY\} \\
L_3 &= \{ABC, ACY\}
\end{aligned}
$$

# Solutions

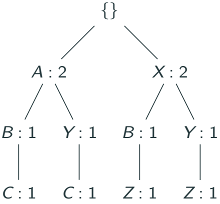
The condition range $< 3$ is antimonotone and we get the following sets during the Apriori algorithm, the numbers now indicate the range, (support is same as previous)

$$\begin{aligned}
L_1 &= \{A, B, C, X, Y, Z\} \\
C_2 &= \{AB: 1, AC: 2, AX: 4, AY: 5, AZ: 6, BC: 1, BX: 3, BY: 4, BZ: 5, \\
&\qquad CX: 2, CY: 3, CZ: 4, XY: 1, XZ: 2, YZ: 1\} \\
L_2 &= \{AB, AC, BC, XY, XZ, YZ\} \\
C_3 &= \{ABC: 2, XYZ: 2\} \\
L_3 &= \{ABC, XYZ\}
\end{aligned}$$

Note that range of a single item is 0, so all 1-itemsets should be included.

# Solutions

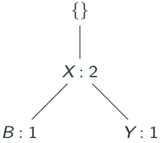
Step 1: Count and sort the items in supportdecending order and output all 1-itemsets. Support of all items are 2, we keep the order and output $\{A, B, C, X, Y, Z\}$. We construct the FP-tree and the conditional databases.



| Item | Conditional Database |
|---|---|
| A | - |
| X | - |
| B | A:1, X:1 |
| Y | A:1, X:1 |
| C | AB:1, AY:1 |
| Z | XB:1, XY:1 |

We look at the Z-suffix, all others are done in the same way. The database is $\{XB:1, XY:1\}$, after counting the support we output $\{XZ, BZ, YZ\}$ and sort the transactions.

| Tid | Items |
|---|---|
| 1 | X, B |
| 2 | X, Y |



| Item | Conditional Database |
|---|---|
| X | - |
| B | X:1 |
| Y | X:1 |

# Solutions

Next we look at the BZ-suffix. The database is $\{X : 1\}$, after counting the support we output $\{XBZ\}$.

Finally (for the Z-suxxif path) we look at the YZ-suffix with database $\{X : 1\}$ we count the support and output $\{XYZ\}$.

Now we need to do the C-suffix, Y-suffix, and B-suffix. After this joining all the outputs will give the same set as $L_1 \cup L_2 \cup L_3$ from the Apriori algorithm.

In the end the algorithm will output the same sets as the first Apriori algorithm

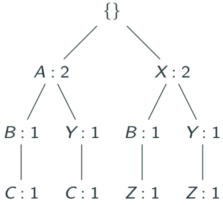
When applying monotone criterion, like $A$ is in the set. Check when outputting that the sets have both minimum support and satisfies the criterion. If the suffix satisfies the condition, the condition can be changed to an always true condition for that branch.

Again the algorithm should output the same sets as the Apriori algorithm.

# Solutions

For an antimonotone condition, such as range($S$) $< 3$, we proceed as follows.

Step 1: We count the support, check the conditionm, and order all transactions. We find that all items satisfy the condition and have minimum support and output $\{A, B, C, X, Y, Z\}$. We construct the FP-tree and the conditional databases.

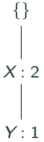


| Item | Conditional Database |
|---|---|
| A | - |
| X | - |
| B | A:1, X:1 |
| Y | A:1, X:1 |
| C | AB:1, AY:1 |
| Z | XB:1, XY:1 |

# Solutions

We look at the Z-suffix, all others are done in the same way. The database is $\{XB : 1, XY : 1\}$. First check if all items in the tree satisfies the condition range($\{Z, X, B, Y\}$) $= 5 > 3$, it does not so we continue by calculating the support of each item in the conditional database and checking the condition with the suffix Z. We find that range($\{B, Z\}$) $= 4 > 3$, range($\{X, Z\}$) $= 2 < 3$, and range($\{Y, Z\}$) $= 1 < 3$, we can thus remove B from this database. We output $\{XZ, YZ\}$, sort the transactions and build the tree

| Tid | Items |
|---|---|
| 1 | X |
| 2 | X, Y |



| Item | Conditional Database |
|---|---|
| X | - |
| Y | X:1 |

Finally we look at the YZ-suffix. We check that range($\{X, Y, Z\}$) $= 2 < 3$ and output $\{XYZ\}$.

We do the same with the other suffixes. Combining all the outputs should give the same answer as the Apriori algorithm.

# Solutions

Last exercise we wish to generate all rules from the frequent itemset $XBZ$ on the database with confidence 0.5.

We begin by checking the subsets $\{XB, XZ, BZ\}$ as the antecedent in the rules.

The rule $XB \rightarrow Z$ has confidence 1 so we output it and check for rules with antecedent $X$ or $B$.

For $X \rightarrow BZ$ we have confidence 0.5 so we output it.

For $B \rightarrow XZ$ we have confidence 0.5 so we output it.

Next we look at the rule $XZ \rightarrow B$ which has confidence 1, we output it and check for rules with antecedent $X$ or $Z$.

$X$ is already done so we check $Z \rightarrow BZ$ which has confidence 0.5 so we output it.

Finally we look at the rule $BZ \rightarrow X$ which has confidence 1, we output it and check for rules with antecedent $B$ or $Z$.

We have already checked those and are done.