VSB TECHNICAL UNIVERSITY OF OSTRAVA | FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE | DEPARTMENT OF COMPUTER SCIENCE

Machine Learning

Associative Pattern Mining

Jan Platoš

November 22, 2023

Department of Computer Science
Faculty of Electrical Engineering and Computer Science
VŠB - Technical University of Ostrava

Associative Pattern Mining

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Pattern?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association?

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Mining?

Association Pattern Mining - Pattern Examples

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association Pattern Mining - Pattern Examples

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association Pattern Mining - Pattern Examples

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association Pattern Mining - Pattern Examples

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

- Bread - 7/10

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

- Bread - 7/10
- Milk - 4/10

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

- Bread - 7/10
- Milk - 4/10
- Fruit - 5/10

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

- Bread - 7/10
- Milk - 4/10
- Fruit - 5/10
- Yogurt - 4/10

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

- Bread - 7/10
- Milk - 4/10
- Fruit - 5/10
- Yogurt - 4/10
- Cereals - 7/10

Association Pattern Mining

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

Simple statistics:

- Bread - 7/10
- Milk - 4/10
- Fruit - 5/10
- Yogurt - 4/10
- Cereals - 7/10

How to continue?

Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

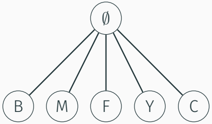
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

$\emptyset$

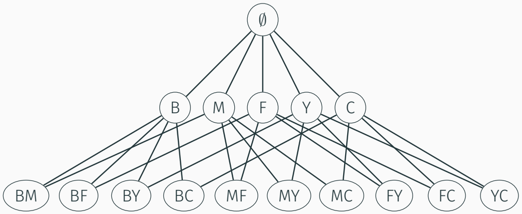
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



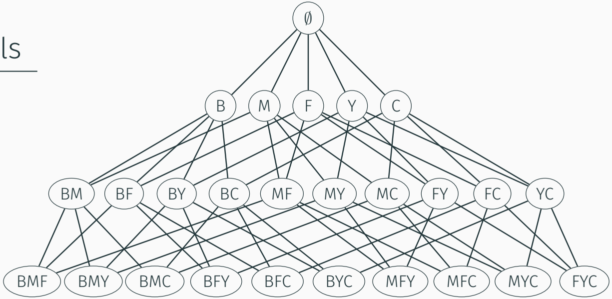
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



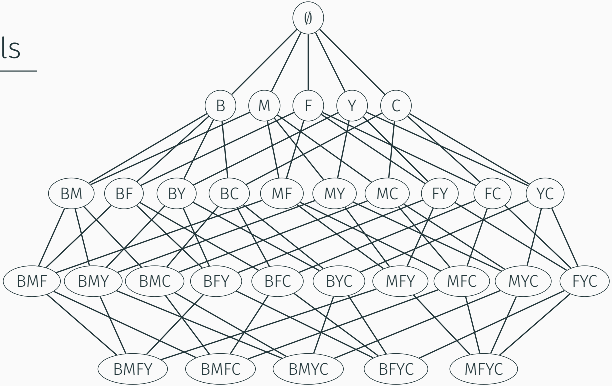
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



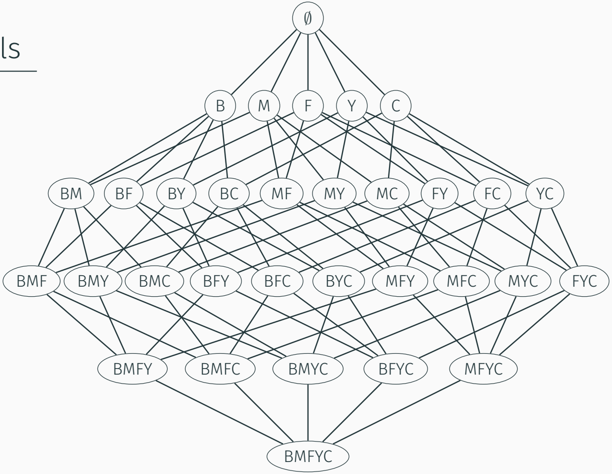
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



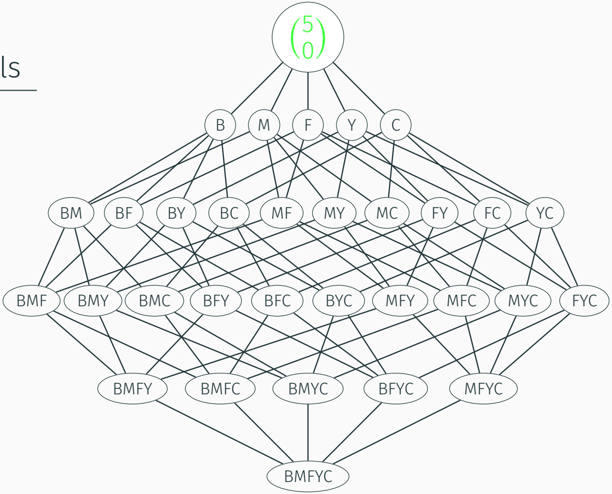
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



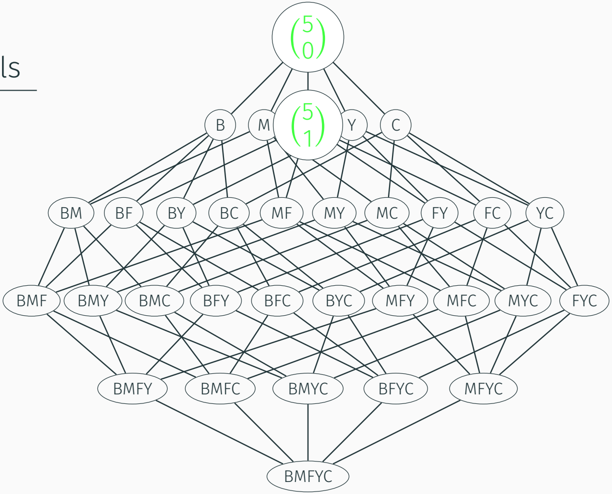
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



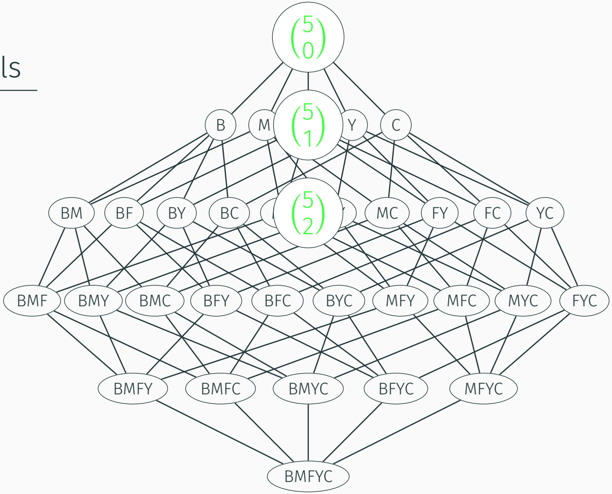
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



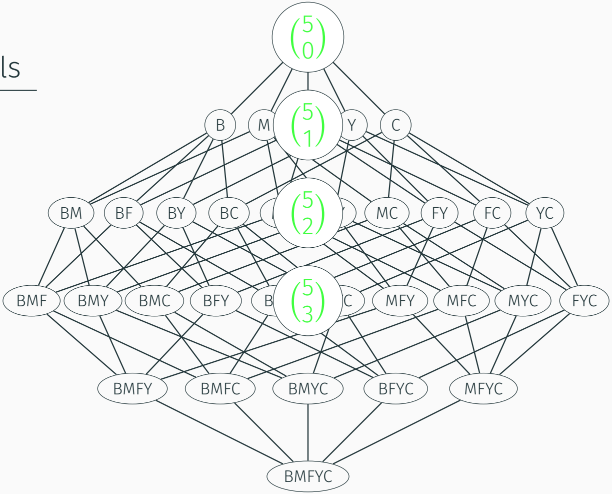
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



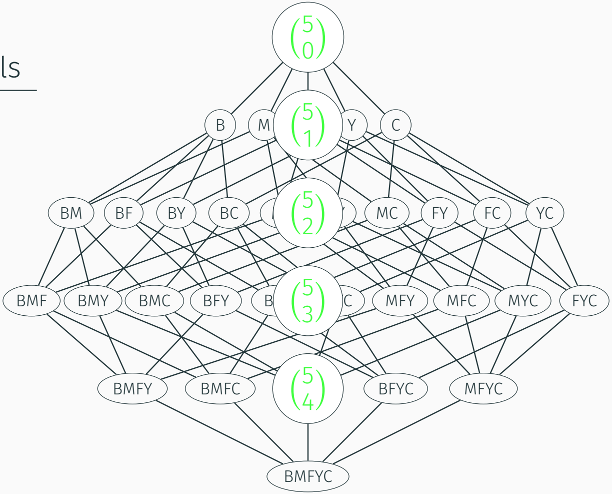
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



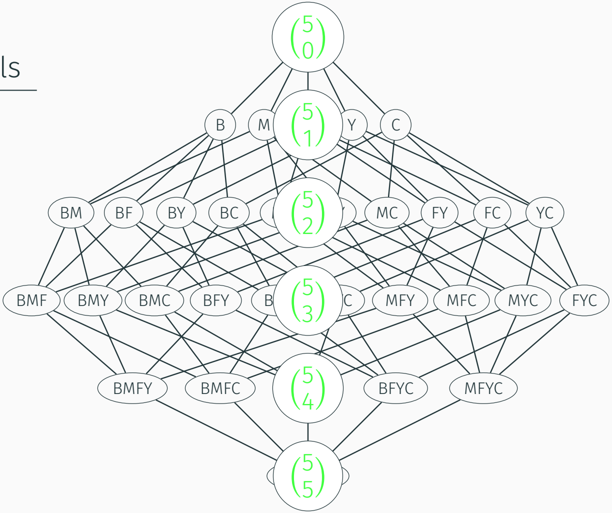
Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



Association Pattern Mining - Test all combinations?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1$$

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1 + 5$$

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10$$

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10 + 10$$

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10 + 10 + 5$$

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10 + 10 + 5 + 1 = 32$$

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10 + 10 + 5 + 1 = 32$$

$$\binom{5}{0} + \binom{5}{1} + \binom{5}{2} + \binom{5}{3} + \binom{5}{4} + \binom{5}{5} = 2^5$$

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10 + 10 + 5 + 1 = 32$$

$$\binom{5}{0} + \binom{5}{1} + \binom{5}{2} + \binom{5}{3} + \binom{5}{4} + \binom{5}{5} = 2^5$$

- How many combination we have to test for n features?

Association Pattern Mining - Test all combinations?

Example

- How many combination we have to test for 5 features?

$$1 + 5 + 10 + 10 + 5 + 1 = 32$$

$$\binom{5}{0} + \binom{5}{1} + \binom{5}{2} + \binom{5}{3} + \binom{5}{4} + \binom{5}{5} = 2^5$$

- How many combination we have to test for n features?

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n$$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$
- $n = 10 \rightarrow 2^{10} = 1,024$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$
- $n = 10 \rightarrow 2^{10} = 1,024$
- $n = 20 \rightarrow 2^{20} = 1,048,576$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$
- $n = 10 \rightarrow 2^{10} = 1,024$
- $n = 20 \rightarrow 2^{20} = 1,048,576$
- $n = 30 \rightarrow 2^{30} = 1,073,741,824$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$
- $n = 10 \rightarrow 2^{10} = 1,024$
- $n = 20 \rightarrow 2^{20} = 1,048,576$
- $n = 30 \rightarrow 2^{30} = 1,073,741,824$
- $n = 40 \rightarrow 2^{40} = 1,099,511,627,775$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$
- $n = 10 \rightarrow 2^{10} = 1,024$
- $n = 20 \rightarrow 2^{20} = 1,048,576$
- $n = 30 \rightarrow 2^{30} = 1,073,741,824$
- $n = 40 \rightarrow 2^{40} = 1,099,511,627,775$
- $n = 272 \rightarrow 2^{272} = 10^{82} = \textit{the number of atoms in Universe}$

Association Pattern Mining - Test all combinations?

Example

- $n = 5 \rightarrow 2^5 = 32$
- $n = 10 \rightarrow 2^{10} = 1,024$
- $n = 20 \rightarrow 2^{20} = 1,048,576$
- $n = 30 \rightarrow 2^{30} = 1,073,741,824$
- $n = 40 \rightarrow 2^{40} = 1,099,511,627,775$
- $n = 272 \rightarrow 2^{272} = 10^{82} = \textit{the number of atoms in Universe}$

Is there a better way to find the important itemsets?

Association Pattern Mining - A better way?

Is every itemset important?

- How to measure the importance of the itemset?
- How to utilize this information in the mining process?

Association Pattern Mining - A better way?

Is every itemset important?

- How to measure the importance of the itemset?
- How to utilize this information in the mining process?

Assumptions

- An itemset is important if it appears frequently.
- Let say that the ”frequently” is when the itemset holds for 20% of rows.

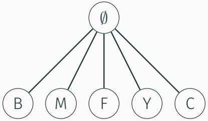
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

$\emptyset$

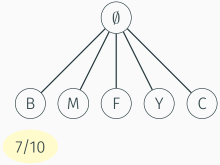
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



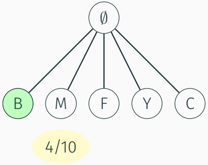
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



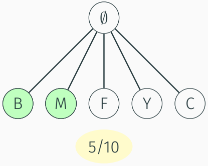
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



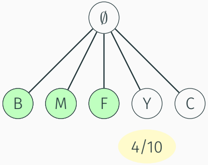
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



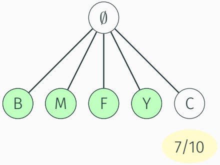
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



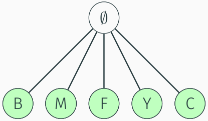
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



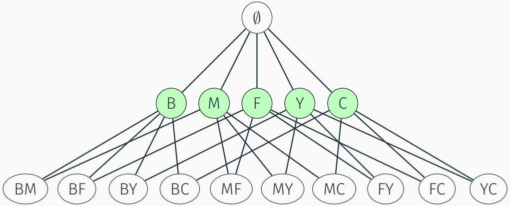
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



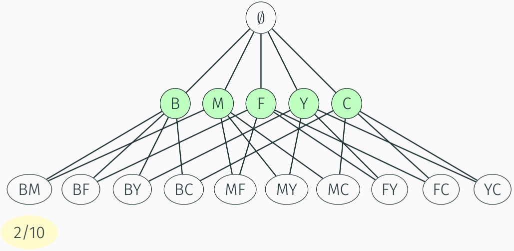
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



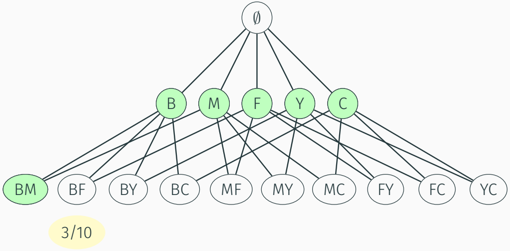
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



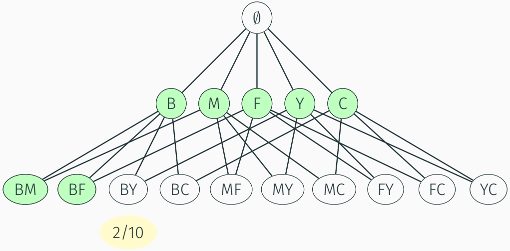
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



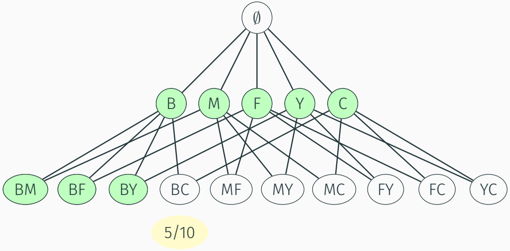
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



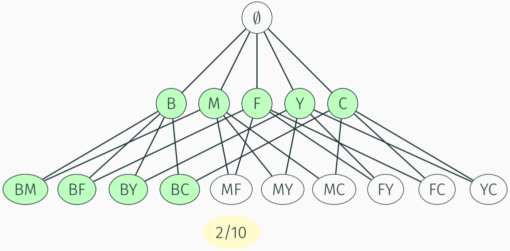
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



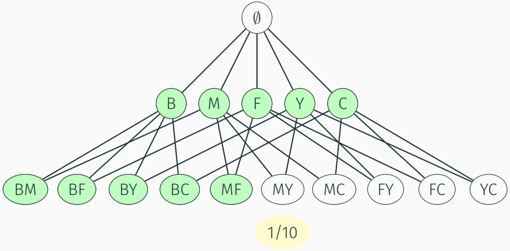
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



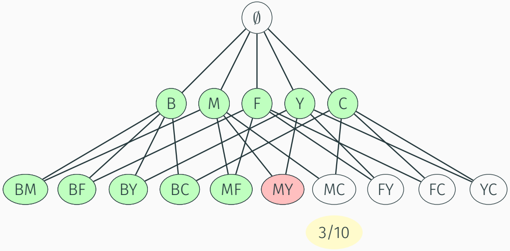
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



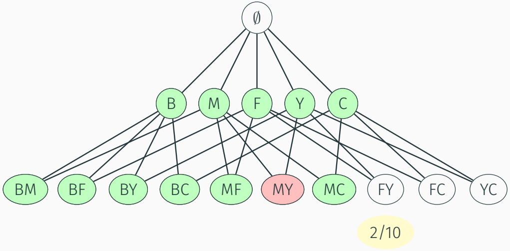
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



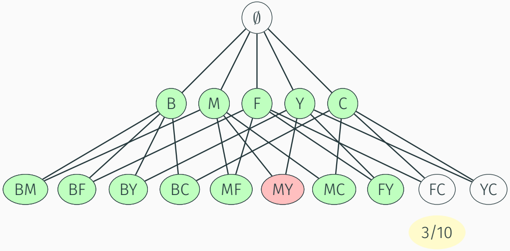
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



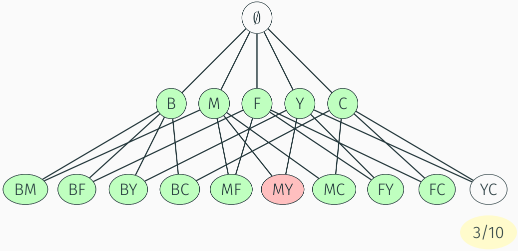
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



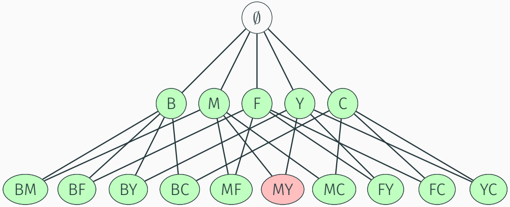
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



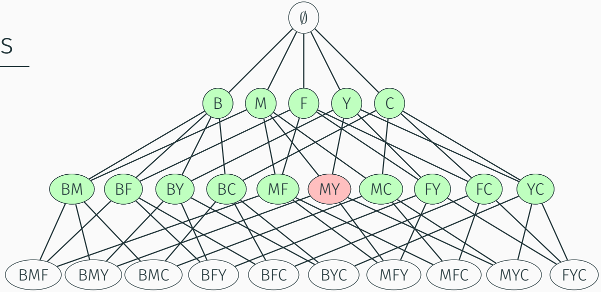
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



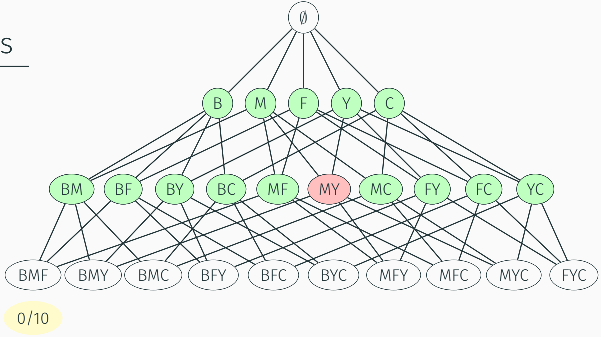
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



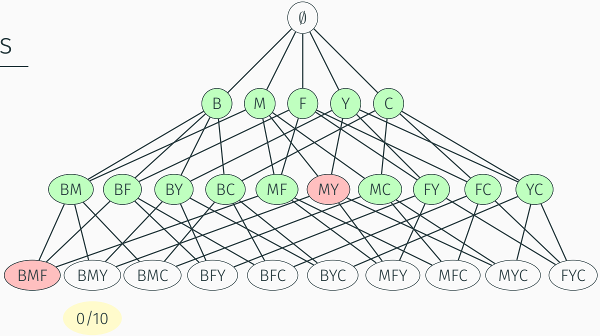
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



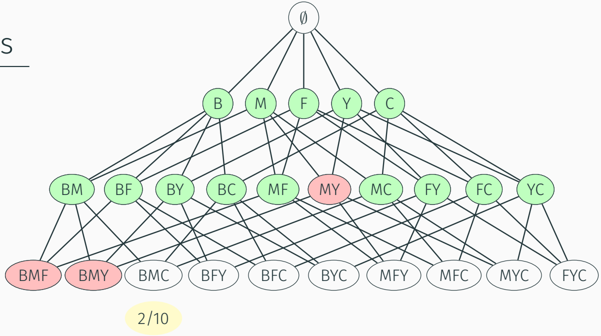
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



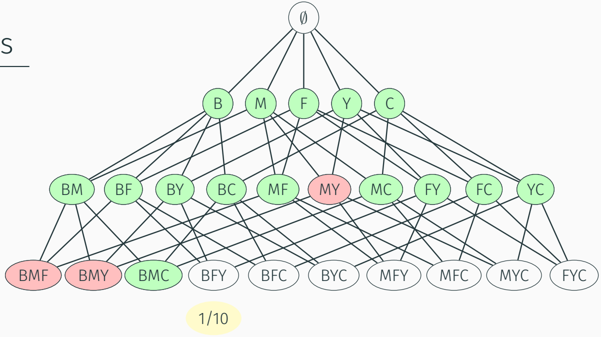
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



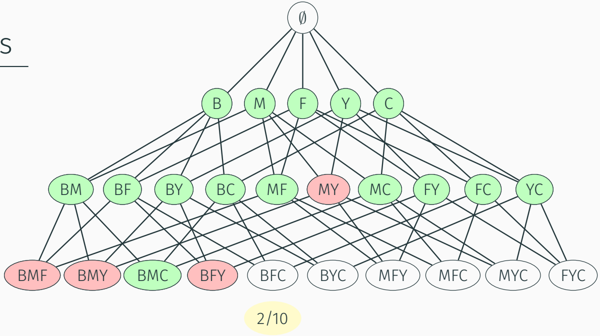
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



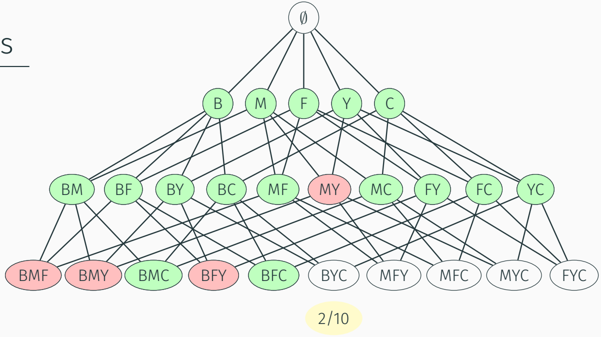
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



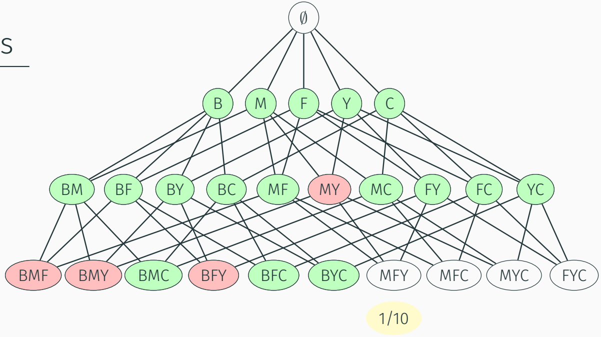
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



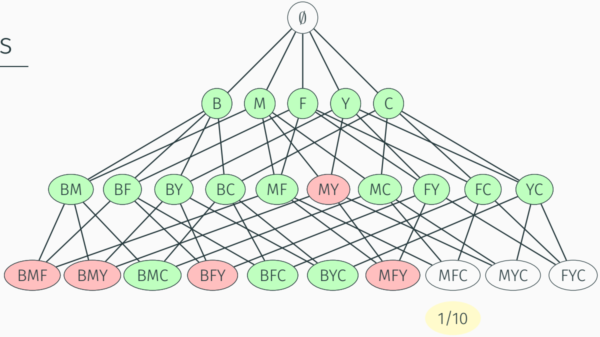
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



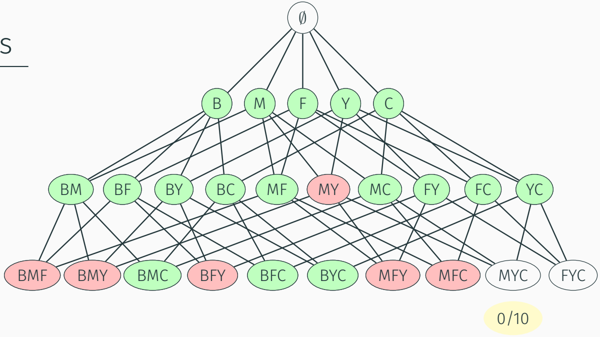
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



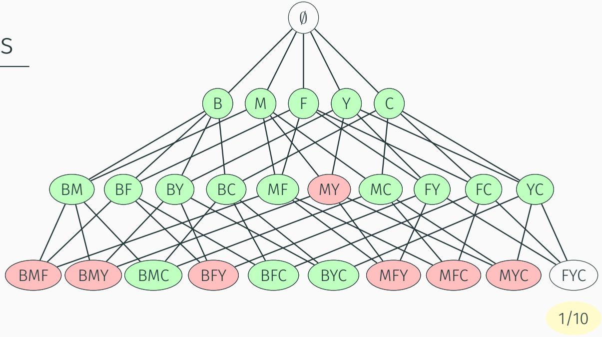
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



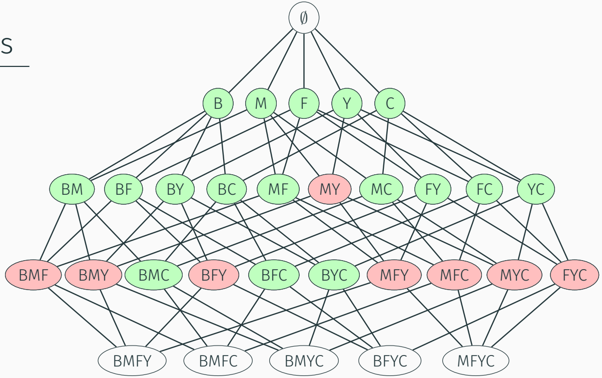
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



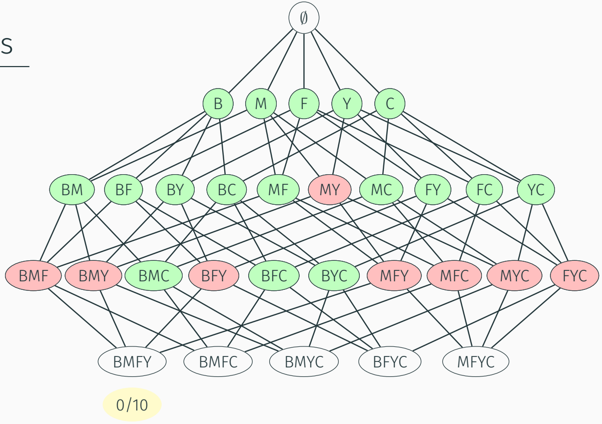
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



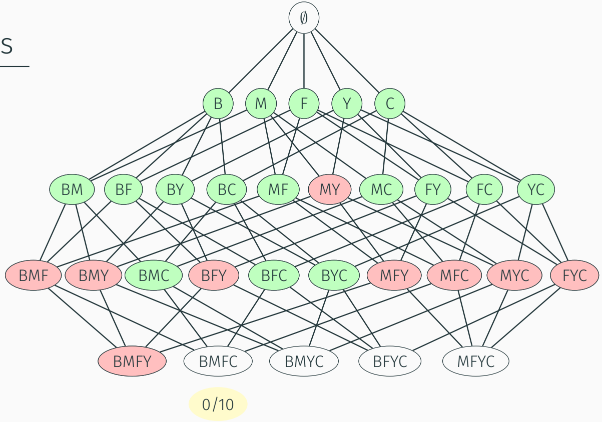
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



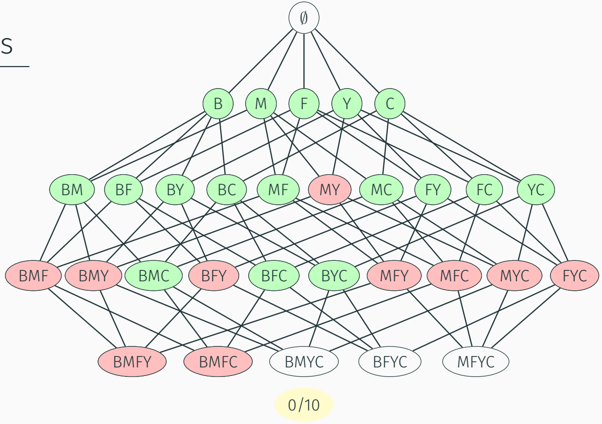
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



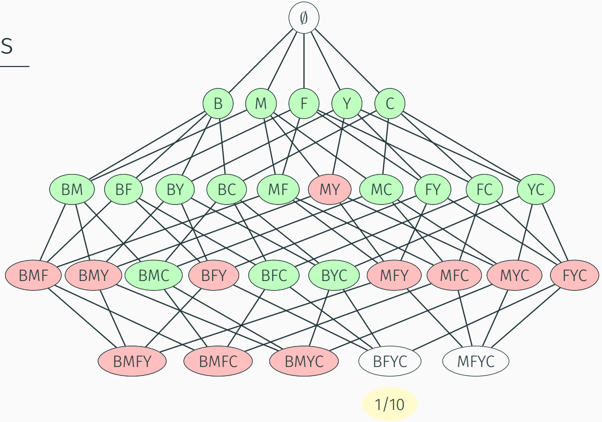
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



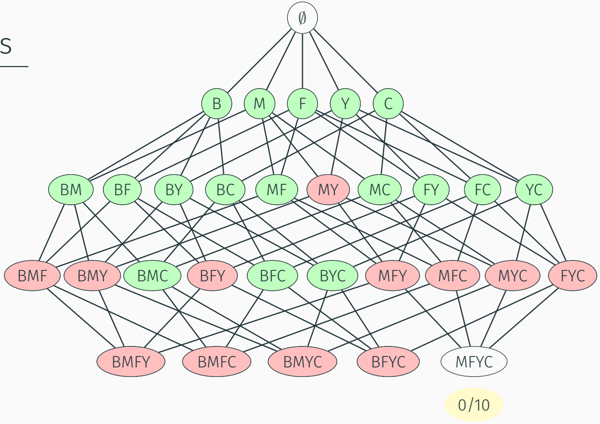
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



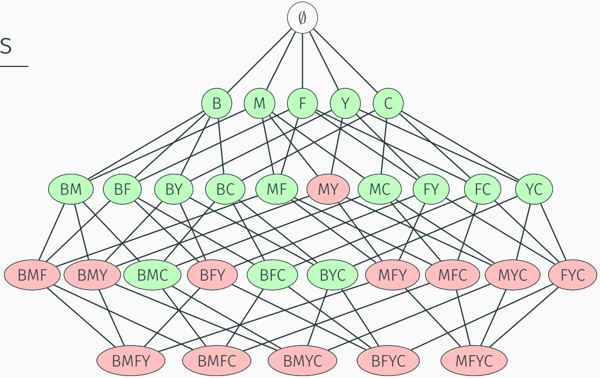
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



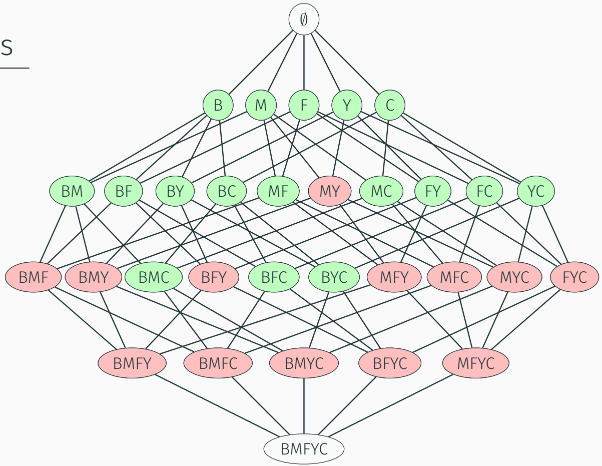
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



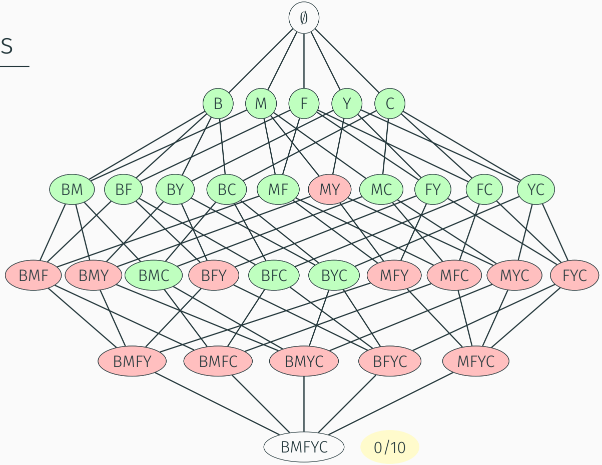
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0



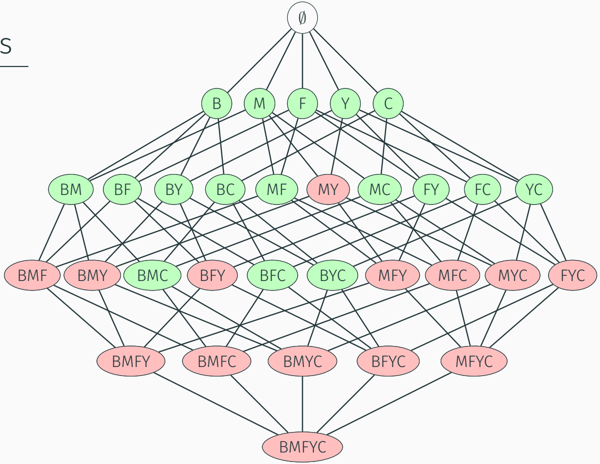
Association Pattern Mining - A better way?

Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

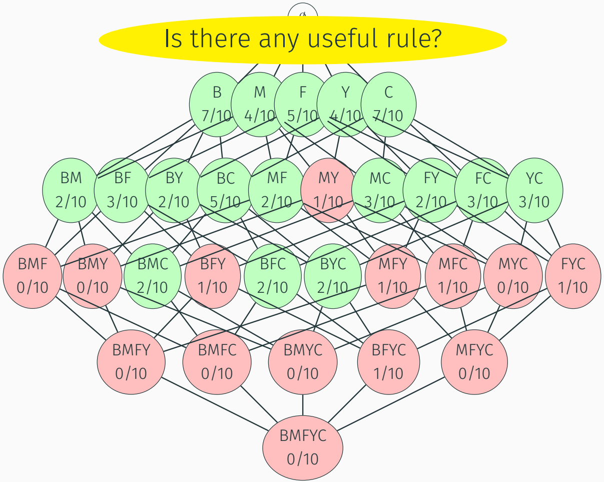


Association Pattern Mining - A better way?

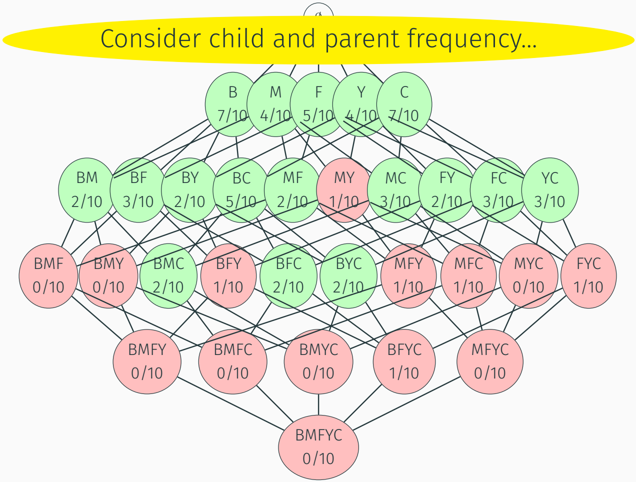
Bread	Milk	Fruit	Yogurt	Cereals
1	1	0	0	1
1	0	0	1	1
1	0	1	0	0
0	1	1	1	0
0	1	1	0	1
0	0	0	1	1
1	0	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	0	0

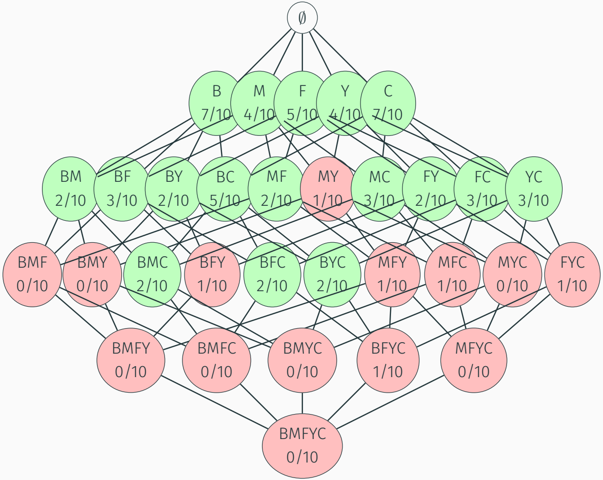


Association Pattern Mining - A better way?

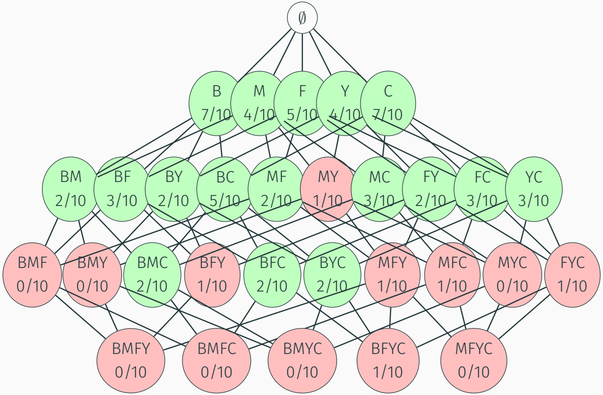


Association Pattern Mining - A better way?

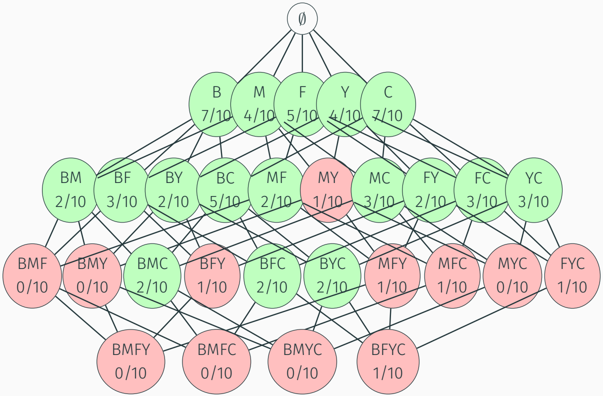



∅
B 7/10
M 4/10
F 5/10
Y 4/10
C 7/10
BM 2/10
BF 3/10
BY 2/10
BC 5/10
MF 2/10
MY 1/10
MC 3/10
FY 2/10
FC 3/10
YC 3/10
BMF 0/10
BMY 0/10
BMC 2/10
BFY 1/10
BFC 2/10
BYC 2/10
MFY 1/10
MFC 1/10
MYC 0/10
FYC 1/10
BMFY 0/10
BMFC 0/10
BMYC 0/10
BFYC 1/10
MFYC 0/10
BMFYC 0/10

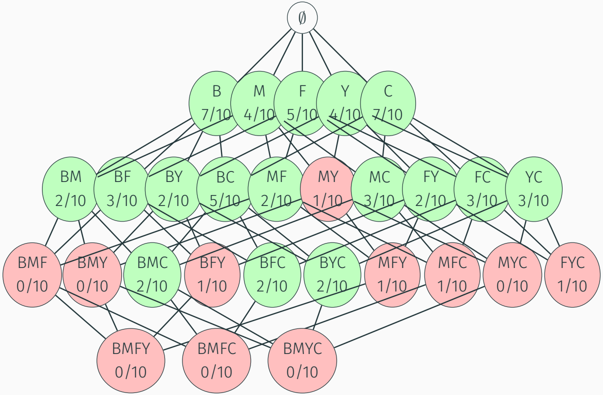
Association Pattern Mining - A better way?



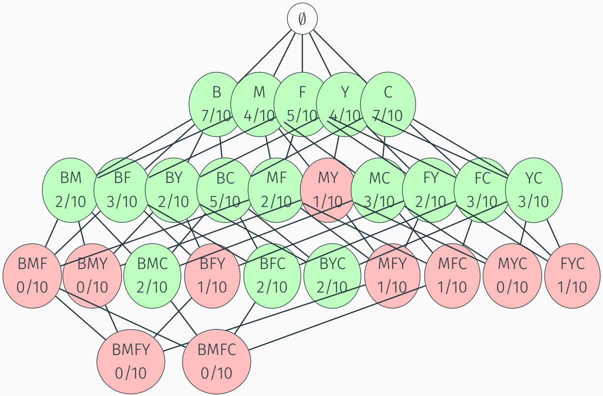
Association Pattern Mining - A better way?



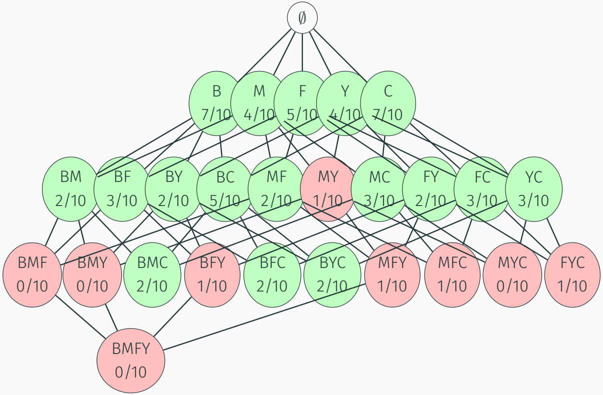
Association Pattern Mining - A better way?



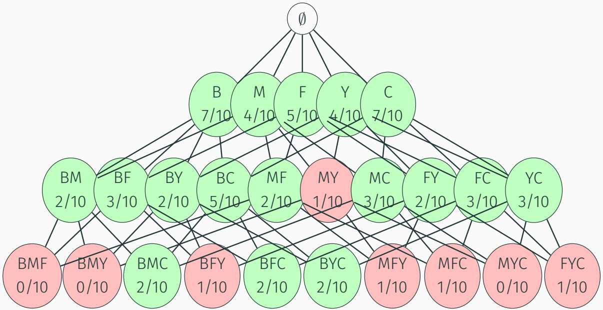
Association Pattern Mining - A better way?



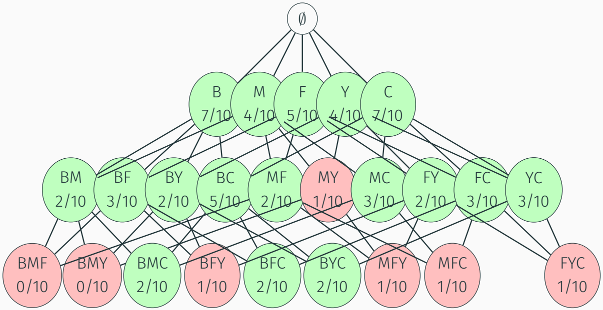
Association Pattern Mining - A better way?



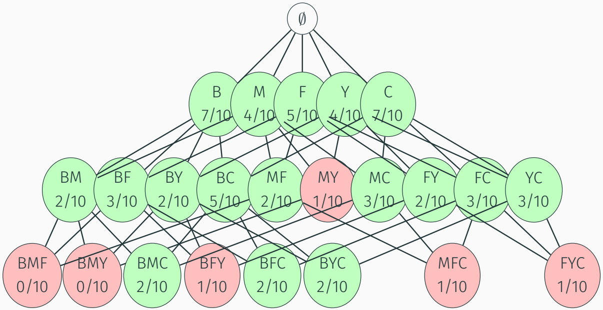
Association Pattern Mining - A better way?



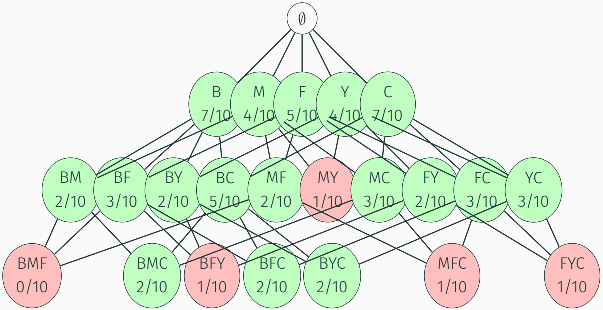
Association Pattern Mining - A better way?



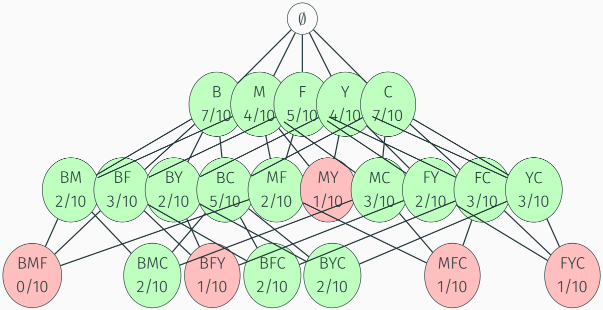
Association Pattern Mining - A better way?



Association Pattern Mining - A better way?



Association Pattern Mining - A better way?



A tested combination reduction
is to 23 from 32 nodes, i.e. to 72%

Association Pattern Mining - A notice about combinations

- How to generate all the combinations needed?

Association Pattern Mining - A notice about combinations

- How to generate all the combinations needed?
- We need to generate then as efficiently as possible (without repetitions).

Association Pattern Mining - A notice about combinations

- How to generate all the combinations needed?
- We need to generate then as efficiently as possible (without repetitions).
- If we can generate them in lexicographical order, it will be even better.

Association Pattern Mining - A notice about combinations

- How to generate all the combinations needed?
- We need to generate then as efficiently as possible (without repetitions).
- If we can generate them in lexicographical order, it will be even better.
- We may use a binary representation, numerical representation, zero-position representation and many other (see Chapter 7 in Volume 4A of Donald Knuth's The Art of Computer Programming series).

Association Pattern Mining - A notice about combinations

- How to generate all the combinations needed?
- We need to generate then as efficiently as possible (without repetitions).
- If we can generate them in lexicographical order, it will be even better.
- We may use a binary representation, numerical representation, zero-position representation and many other (see Chapter 7 in Volume 4A of Donald Knuth's The Art of Computer Programming series).
- We need only the combination of a specified length k of all n items.

Example (A simple nested cycle solution for $k = 3$ and $n = 5$)

```
for (int a=1;a<=5; a++)
{
    for (int b=a+1;b<=5; b++)
    {
        for (int c=b+1;c<=5; c++)
        {
            printf("%d %d %d\n", a, b, c);
        }
    }
}
```

a	b	c
1	2	3

Example (A simple nested cycle solution for $k = 3$ and $n = 5$)

```
for (int a=1;a<=5; a++)
{
    for (int b=a+1;b<=5; b++)
    {
        for (int c=b+1;c<=5; c++)
        {
            printf("%d %d %d\n", a, b, c);
        }
    }
}
```

a	b	c
1	2	3
1	2	4

Example (A simple nested cycle solution for $k = 3$ and $n = 5$)

```
for (int a=1;a<=5; a++)
{
    for (int b=a+1;b<=5; b++)
    {
        for (int c=b+1;c<=5; c++)
        {
            printf("%d %d %d\n", a, b, c);
        }
    }
}
```

a	b	c
1	2	3
1	2	4
1	2	5

Association Pattern Mining - A notice about combinations

Example (A simple nested cycle solution for $k = 3$ and $n = 5$)

```
for (int a=1;a<=5; a++)
{
    for (int b=a+1;b<=5; b++)
    {
        for (int c=b+1;c<=5; c++)
        {
            printf("%d %d %d\n", a, b, c);
        }
    }
}
```

a	b	c
1	2	3
1	2	4
1	2	5
1	3	4

Example (A simple nested cycle solution for $k = 3$ and $n = 5$)

```
for (int a=1;a<=5; a++)
{
    for (int b=a+1;b<=5; b++)
    {
        for (int c=b+1;c<=5; c++)
        {
            printf("%d %d %d\n", a, b, c);
        }
    }
}
```

a	b	c
1	2	3
1	2	4
1	2	5
1	3	4
1	3	5
1	4	5
2	3	4
2	3	5
2	4	5
3	4	5

Example (A array based version works universally ($k = 6$, $n = 8$))

	[0]	[1]	[2]	[3]	[4]	[5]
1	1	2	3	4	5	6
2	1	2	3	4	5	7
3	1	2	3	4	5	8
4	1	2	3	4	6	7
5	1	2	3	4	6	8
6	1	2	3	4	7	8
7	1	2	3	5	6	7
8	1	2	3	5	6	8
9	1	2	3	5	7	8
10	1	2	3	6	7	8
11	1	2	4	5	6	7
12	1	2	4	5	6	8
13	1	2	4	5	7	8

	[0]	[1]	[2]	[3]	[4]	[5]
15	1	2	5	6	7	8
16	1	3	4	5	6	7
17	1	3	4	5	6	8
18	1	3	4	5	7	8
19	1	3	4	6	7	8
20	1	3	5	6	7	8
21	1	4	5	6	7	8
22	2	3	4	5	6	7
23	2	3	4	5	6	8
24	2	3	4	5	7	8
25	2	3	4	6	7	8
26	2	3	5	6	7	8
27	2	4	5	6	7	8

Association Pattern Mining - Summary

- Patterns

Association Pattern Mining - Summary

- Patterns
- Frequency

Association Pattern Mining - Summary

- Patterns
- Frequency
- Exhaustive search

Association Pattern Mining - Summary

- Patterns
- Frequency
- Exhaustive search
- Optimized search

Association Pattern Mining - Summary

- Patterns
- Frequency
- Exhaustive search
- Optimized search
- Rules

Association Pattern Mining - Summary

- Patterns
- Frequency
- Exhaustive search
- Optimized search
- Rules
- Combinations

Association Pattern Mining - Formal definition

- The goal is to determine associations between groups of items bought by customers, which can intuitively be viewed as k-way correlations between items.
- The most popular model for association pattern mining uses the frequencies of sets of items as the quantification of the level of association.
- The discovered sets of items are referred to as large itemsets, **frequent itemsets**, or frequent patterns.

Association Itemset Mining - The Frequent Itemset Mining Model

- Set of Transactions $T = T_1, T_2, \ldots, T_n$.
- A transaction $T_i = u_{i1}, u_{i2}, \ldots, u_{in}$.
- A universe of items $U = u_1, u_2, \ldots, u_n$.
- An **itemset** is a set of items from U.
- An k-**itemset** is a set of exactly k items from U.
- Example:
  - A shopping cart from supermarket.
  - A contrast between $|U|$ and average size of a transaction.

Support
The support of an itemset I, $sup(I)$, is defined as the fraction of the transactions in the database $T = T_1, \dots T_n$ that contain I as a subset.

Frequent Itemset Mining
Given a set of transactions $T = T_1, \dots T_n$, where each transaction T_i is a subset of items from U, determine all itemsets I that occur in a subset of at least a predefined fraction *minsup* of the transactions in T.

Frequent Itemset Mining: Set-wise Definition
Given a set of sets $T = T_1, \dots T_n$, where each element of the set T_i is drawn on the universe of elements U, determine all sets I that occur as a subset of at least a predefined fraction *minsup* of the sets in T.

Support Monotonicity Property
The support of every subset J of I is at least equal to that of the support of itemset I.

$$\text{sup}(J) \geq \text{sup}(I) \qquad \forall J \subseteq I$$

Downward Closure Property
Every subset of a frequent itemset is also frequent.

Maximal Frequent Itemset
A frequent itemset is maximal at a given minimum support level *minsup*, if it is frequent, and no superset of it is frequent.

Brute Force Algorithm (Exhaustive search)

- Generate all possible combinations of the input features.

Brute Force Algorithm (Exhaustive search)

- Generate all possible combinations of the input features.
- Test whether they have defined support.

More efficient algorithm

- Generate the combination with increasing length, starting from the length 1.

More efficient algorithm

- Generate the combination with increasing length, starting from the length 1.
- Generate them in non-redundant way and lexicographically ordered.

More efficient algorithm

- Generate the combination with increasing length, starting from the length 1.
- Generate them in non-redundant way and lexicographically ordered.
- Apply the *Downward closure property* to filter the combination.

More efficient algorithm

- Generate the combination with increasing length, starting from the length 1.
- Generate them in non-redundant way and lexicographically ordered.
- Apply the *Downward closure property* to filter the combination.
- Prune the used transactions that are irrelevant for support counting.

More efficient algorithm

- Generate the combination with increasing length, starting from the length 1.
- Generate them in non-redundant way and lexicographically ordered.
- Apply the *Downward closure property* to filter the combination.
- Prune the used transactions that are irrelevant for support counting.
- Use compact data structures for candidate database as well as for transaction database.

Apriori Algorithm

- The first and the basic algorithm for efficient itemset mining.
- Strict using of the downward closure property to prune candidates.
- Level-wise generation of candidates
    - Candidates with length k are generated.
    - A support of these candidates is computed.
    - Candidates with length k+1 are generated.
- Uses lexicographic ordering on itemsets as a helper.
- Only itemsets with $k-1$ common items may be joined.
- A $(k+1)$-itemsets are generated only when all subsets are frequent.

Algorithm 1: Apriori(Transactions: T, Minimum support: *minsup*)

begin
    k = 1;
    F_1 = {All Frequent 1-itemsets};
    while F_k *is not empty* **do**
        Generate C_{k+1} by joining itemset-pairs in F_k;
        Prune itemsets from C_{k+1} that violate downward closure property;
        Perform the support counting operation in (C_{k+1}, T);
        Put all itemsets with support at least *minsup* into F_{k+1};
        k = k+1;
    end
    return $(\bigcup_{i=1}^{k} F_i)$
end

Efficient support counting

- The detection of the presence of a an candidate itemset in a transaction is crucial for support counting.
- The *hash tree* data structure may be efficiently used.
- This structure organize the candidate itemsets in a way that each candidate is in exactly one leaf.
- Each internal node consist of a hash table.
- The interior nodes defines the path from the root the each leaf nodes.
- Required the transactions to be lexicographically sorted.
- Each level of the tree corresponds to the one item in a candidate.

Enumeration-Tree Algorithm

- A useful generalization/abstraction of the most of the frequent itemset mining algorithms.
- Allows systematic exploration of the candidates in a non-repetitive way.
- Enumeration-Tree is defined on the frequent itemsets:
  - A node exists in the tree corresponding to each frequent itemset. The root of the tree corresponds to the null itemset.
  - Let $I = \{i_1, \ldots, i_k\}$ be a frequent itemset, where $i_1, i_2, \ldots, i_k$ are listed in lexicographic order. The parent of the node I is the itemset $\{i_1, \ldots, i_{k1}\}$. Thus, the child of a node can only be extended with items occurring lexicographically after all items occurring in that node. The enumeration tree can also be viewed as a prefix tree on the lexicographically ordered string representation of the itemsets.

Association Pattern Mining - Enumeration-Tree Algorithm

Algorithm 2: GenericEnumerationTree(Transactions: T, Minimum support: *minsup*)

begin

  Initialize enumeration tree $\mathcal{ET}$ to single *Null* root node;

  while *any node in* $\mathcal{ET}$ *has not been examined* **do**

    Select one or more not-examined nodes $\mathcal{P}$ from $\mathcal{ET}$ for examination;

    Generate candidates extensions $C(P)$ of each node $P \in \mathcal{P}$;

    Determine frequent extension $F(P) \subseteq C(P)$ for each $P \in \mathcal{P}$ with support counting ;

    Extend each node $P \in \mathcal{P}$ in $\mathcal{ET}$ with its frequent extension in $F(P)$;

  end

  return *enumeration tree* $\mathcal{ET}$

end

TreeProjection

- A general framework for database projection (a mapping of a set of transaction to the itemset).
- Support many different strategies for construction of an enumeration tree.
- The main idea follows the same properties that are used in Apriori.

If a transaction does not contain itemset that corresponds to the node in enumeration tree, it will not be relevant even for the child nodes of the node.

- The proper selection of the node *P* for extension affect the memory consumption.
- Evaluate the Depth-first and Breath-first approach.
- The counting may be solved differently at deeper levels (such as bit maps).

Association Pattern Mining - TreeProjection

Algorithm 3: ProjectedEnumerationTree(Transactions: T, Minimum support: *minsup*)

begin
    Initialize enumeration tree $\mathcal{ET}$ to single $(Null, T)$ root node;
    while *any node in* $\mathcal{ET}$ *has not been examined* **do**
        Select an not-examined nodes $(P, T(P))$ from $\mathcal{ET}$ for examination;
        Generate candidates extensions $C(P)$ of each node $(P, T(P))$;
        Determine frequent extension $F(P) \subseteq C(P)$ by support counting of individual items in smaller projected database $T(P)$;
        Remove infrequent items in $T(P)$;
        foreach *each frequent item extension* $i \in F(P)$ **do**
            Generate $T(P \cup \{i\})$ from $T(P)$;
            Add $(P \cup \{i\}, T(P \cup \{i\}))$ as child of P in $\mathcal{ET}$;
        end
    end

Vertical Counting Methods

- A transposed transaction database.
- Higher memory consumption.
- Faster due to implicit transaction list.
- Support counting refers to the length of transaction list.
- Merging is a intersection of the list (linear time operation).
- Partitioning of transaction list into chunks reduces memory requirements.
- Algorithms: Partition, Monet, Eclat, VIPER.

Interesting patterns

- Alternative definition to frequent itemset.
- Applies when support and confidence is not ideal measure.
- When we are investigating the relation between set of items, we are focused on the similarity more than on their frequency.
- The Negative pattern mining is also difficult to find and investigate (the downward closure property does not hold).
- New methods for two or more items to be compared have to be defined.
- **Bit symmetry property** hold when the presence and the absence of an item is evaluated in the exactly the same way.

Pearson coefficient of correlation between pair of items

$$\rho_{ij} = \frac{sup(\{i,j\}) - sup(i) \cdot sup(j)}{\sqrt{sup(i) \cdot sup(j) \cdot (1 - sup(i)) \cdot (1 - sup(j))}}$$

- Holds the *bit symmetry property*.
- Measures the correlation between items i and j.
- The results is in $[-1, 1]$ where $+1$ is a maximum positive correlation, and -1 a maximum negative correlation. The values around 0 means weakly correlated data.
- The most robust way of measuring correlation.
- Hard to interpret when the support is low.

χ^2 Measure

$$\chi^2(X) = \sum_{i=0}^{i<2^{|X|}} \frac{(O_i - E_i)^2}{E_i}$$

- Holds the *bit symmetry property*.
- The X is a set of k binary items, the number of possible combination is $2^{|X|}$.
- The E_i is the expected fractional presence, when the items are non-dependent on each other.
- The O_i is the observed presence, i.e. the support of a combination X_i of items.
- The χ^2 close to zero means no dependence between items, and large χ^2 means high dependence but does not discover positive or negative.

Interest ratio

$$I(\{i_1, \ldots, i_k\}) = \frac{sup(\{i_1, \ldots, i_k\})}{\prod_{j=1}^{k} sup(i_j)}$$

- Holds the *bit symmetry property*.
- Simple measure with easy interpretation.
- For the statistically independent items the joint support is equal to the product of the support of separate items.
- The value greater than 1 indicate positive correlation, the value less than 1 negative.
- The extremely rare items confuse the ratio (e.g. single occurrence in large database).

Symmetric Confidence Measures

- The classic confidence measure is asymmetric between antecedent and consequent.
- The support measure is symmetric.
- The symmetric confidence may replace support-confidence with a single measure.
- The measures does not satisfy the downward closure property.

Cosine Coefficient on Columns

$$cosine(i, j) = \frac{sup(\{i, j\})}{\sqrt{sup(\{i\})} \cdot \sqrt{\{j\}}}$$

- Measures the similarity between columns instead of rows.
- It may be viewed as a geometric mean of the confidences of the rules $\{i\} \Rightarrow \{j\}$ and $\{j\} \Rightarrow \{i\}$.

Jaccard Coefficient

$$J(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$$

- Defined over sets.
- The sets are the transactions Ids in single columns.
- Satisfies the downward closure property.

Collective Strength

$$C(I) = \frac{1 - v(i)}{1 - E\left[v(i)\right]} \cdot \frac{E\left[v(i)\right]}{v(i)}$$

- The I is an itemset.
- The measure is defined in terms of its violation rate.
- An itemset I is said to be in violation of a transaction, if some of the items are present in the transaction and others are not.
- The violation rate $v(I)$ is the fraction of violations of the itemset I over all transactions.
- The expected value $E\left[v(i)\right]$ of $v(I)$ assumes the statistical independence.

$$E\left[v(i)\right] = 1 - \prod_{u \in I} p_i - \prod_{u \in I} (1 - p_i)$$

Association Pattern Mining - Generalization to other data types

- Numeric values
  - Division into subranges
  - The subranges may be uniform or based on the number probability density.
  - Adjacent ranges may be merged during mining to provide summarized knowledge.
- Categorical data
  - Binarization into separate columns.
  - Clusters of similar values are also possible.
  - A domain knowledge may be used to process the data.

Association Pattern Mining - Applications

- Classification
  - Rule-based classification.
  - For rules $X \Rightarrow Y$, Y is a class variable.
  - Support and confidence not enough.
  - Rules have to discriminate between class variables.
- Outlier detection
  - Search for transactions that are not covered/violated by patterns.
  - Useful when distance based measures are not working

Association Pattern Mining - Applications

- Collaborative filtering and recommendation
  - Localized pattern mining
  - Grouping users according to their behavior.
- Web log analysis
  - Web logs similar to the baskets.
  - Temporal aspects.
- Bio-informatics
  - Gene-expression data.
  - Very high number of columns (thousands, hundred thousands).
  - Maximal and closed patterns.

Association Rule Generation Framework

Association Rule Generation Framework

Association Rule Generation Framework
Association rules are if/then statements that help uncover relationships between seemingly unrelated data in a relational database or other information repository. An example of an association rule would be ”If a customer buys a dozen eggs, he is 80% likely to also purchase milk.”

$$X \Rightarrow Y$$

Confidence
Let X and Y be two sets of items. The confidence $conf(X \cup Y)$ of the rule $X \cup Y$ is the conditional probability of $X \cup Y$ occurring in a transaction, given that the transaction contains X. Therefore, the confidence $conf(X \Rightarrow Y)$ is defined as follows:

$$conf(X \Rightarrow Y) = \frac{sup(X \cup Y)}{sup(X)}$$

Confidence Monotonicity
Let X_1, X_2 and I be itemsets such that $X_1 \subset X_2 \subset I$. Then the confidence of $X_2 \Rightarrow I - X_2$ is at least that of $X_1 \Rightarrow I - X_1$.

$$conf(X_2 \Rightarrow I - X_2) \geq conf(X_1 \Rightarrow I - X_1)$$

Association Rule Generation Framework

Association Rules
Let X and Y be two sets of items. Then, the rule $X \Rightarrow Y$ is said to be an association rule at a minimum support of *minsup* and minimum confidence of *minconf*, if it satisfies both the following criteria:

1. The support of the itemset $X \cup Y$ is at least *minsup*.
2. The confidence of the rule $X \Rightarrow Y$ is at least *minconf*.

Association Rule Generation Framework

Phase 1

- Generate all the frequent itemsets at the minimum support of *minsup*.
  - Very computationally expensive.
  - The Apriori of similar algorithm may be used.

Phase 2

- Generate all the association rules from the frequent itemsets at the minimim confidence of *minconf*.
  - A much simple phase when all frequent itemsets F are generated.
  - For each itemset $I \in F$ generate all possible combinations X and Y and compute confidence.

Questions?