

Откриване на информация

Асоциации

Процеси

- Събиране, анализ и подбор на данни
- Предварително описание и оценка със статистически параметри
- Визуализация и избор на зависимости за изследване
- Построяване на прогнозен модел
- Тестване на модела
- Верификация на модела

Data Mining

- Същност
 - автоматизирано откриване на знания, чрез разкриване на скрити схеми
 - конструиране на аналитични модели, асоцииране, класифициране
- Задачи
 - Описателни
 - Прогнозни
- Методи
 - асоциации
 - класификация
 - клъстеризация
 - регресия
 - и др.

Откриване на асоциации

Pattern Mining

Откриване на асоциации

- Търсене на регулярни повторения в данните
- Примери
 - Какви продукти се купуват най-често заедно?
 - Какво се купува след купуване на принтер?
 - Кои видове ДНК са чувствителни относно новото лекарство?
 - и др.

{Сирене} → {Бира}

{Мляко, Хляб} → {Яйца, Безалкохолни}

{Бира, Хляб} → {Мляко}

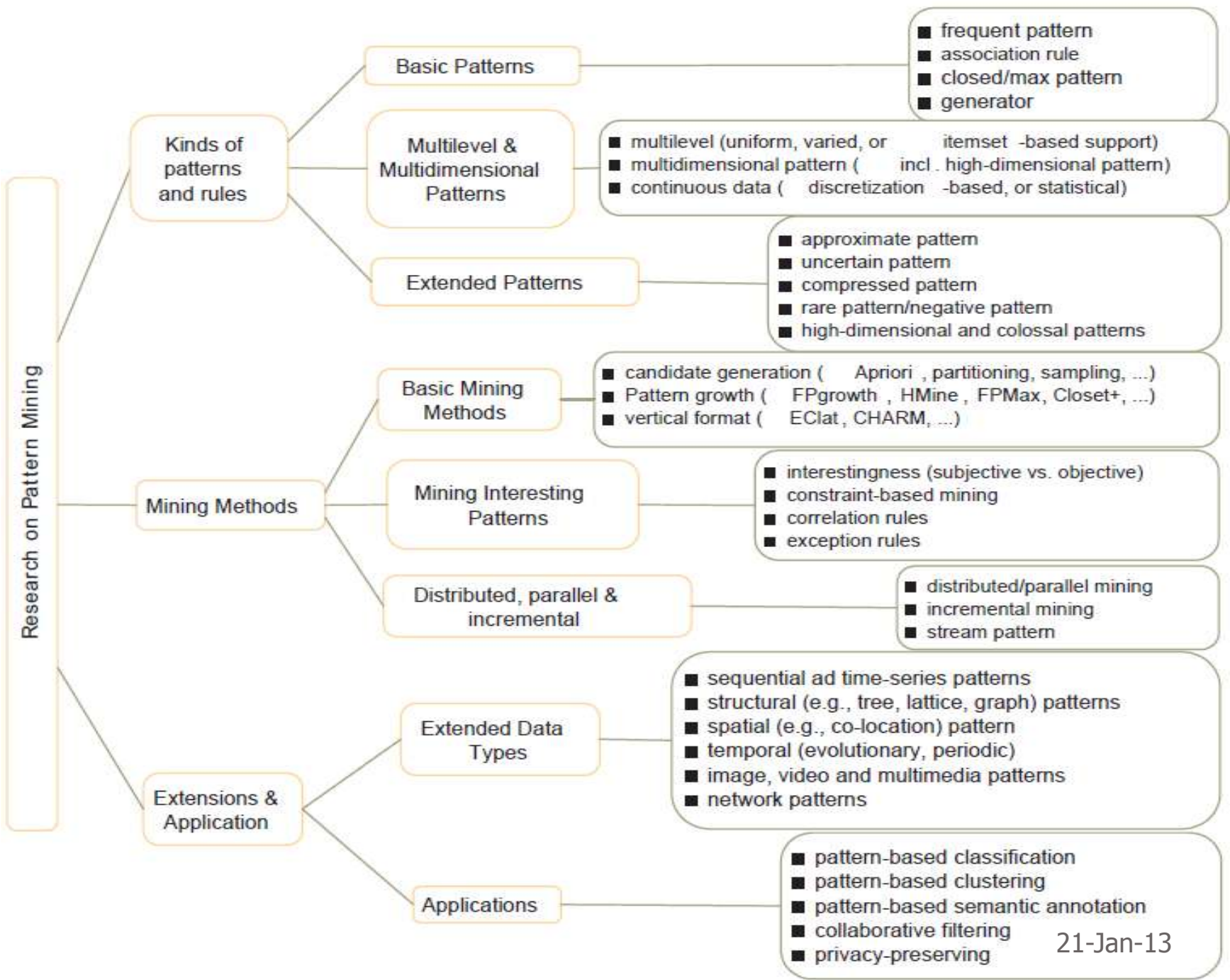
- **импликация**
- не е причинно-следствена връзка

<i>TID</i>	<i>Items</i>
1	Хляб, Мляко
2	Хляб, Сирене, Бира, Яйца
3	Мляко, Сирене, Бира, Кола
4	Хляб, Мляко, Сирене, Бира
5	Хляб, Мляко, Сирене, Кола

Приложения

- Анализ на потребителска кошница
- Дизайн на каталози
- Разпродажби
- Web анализ
- и др.

Видове изследвания

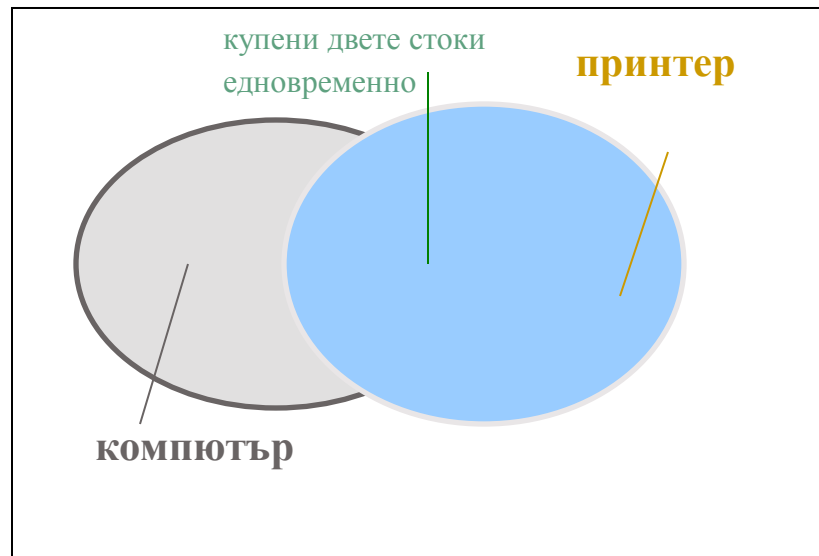


ОСНОВНИ ПОНЯТИЯ

- **Pattern:** шаблон, подмножество от елементи, последователност, повтаряща се в едно множество
 - **Frequency:** брой на повторенията, честота на срещане
 - **Frequent pattern:** шаблон, който често се повтаря, важно качество на множествата
 - **Association Rule:** като се знае множество от транзакции, да се намерят правила, които биха предвидили появяването на елементи, на основание на появяването на други елементи
- Agrawal, Imielinski, and Swami [AIS93]

ОСНОВНИ ПОНЯТИЯ

TID	Купени стоки
10	компютър, принтер, мишка
20	компютър, софтуер
30	мишка, слушалки
40	компютър, слушалки
50	компютър, принтер, мишка, диск



- **транзакции:** регистрирани случаи (T)
- **itemset:** множеството от участващи елементи
{компютър, принтер, мишка, слушалки, софтуер, диск}
- **k-itemset:** подмножество от k-елементи
 $X = \{x_1, \dots, x_k\}$
- **(absolute) support** на множеството X: честотата на появяване на множеството сред всички транзакции [брой]
 $\sigma(\{\text{компютър, принтер}\}) = 2$
- **(relative) support** : вероятността една транзакция да съдържа множеството X [%]
- Множеството X е **frequent** (често срещано), ако неговият коефициент на **support** не е по-малък от минималния зададен

Асоциационни правила

- **Association Rule:** $X \rightarrow Y$
 - **support S**
 - вероятност $P(X \rightarrow Y) = P(X \cup Y) = 0.14$
 - **confidence C**
 - условна вероятност $(X \rightarrow Y) = P(Y | X) = 14/23 = 0.61$
 - **confidence**
 - условна вероятност $(Y \rightarrow X) = P(X | Y) = 14/37 = 0.38$
- минимум **support** и **confidence** $\{X, Y\}$: **minsup, minconf**
- много подмножества го поддържат
- Примери

Пример

- Ако броят на транзакциите е
 - $T = 100$ и
- X и Y са две подмножества от транзакции (напр. тези, при които са купени компютри - 23, и тези, с купени принтери-37), то
 - $P(X) = 23/100 = 0.23$
 - $P(Y) = 37/100 = 0.37$
- $X \cup Y$ – сечение на двете множества, т.е. брой транзакции, в които са купени X и Y едновременно – 14
 - $P(X \cup Y) = 14/100 = 0.14$

Пример

{Мляко, Сирене} → {Бира} (s=0.4, c=0.67)

{Мляко, Бира} → {Сирене} (s=0.4, c=1.0)

{Сирене, Бира} → {Мляко} (s=0.4, c=0.67)

{Бира} → {Мляко, Сирене} (s=0.4, c=0.67)

{Сирене} → {Мляко, Бира} (s=0.4, c=0.5)

{Мляко} → {Сирене, Бира} (s=0.4, c=0.5)

Задача на DM

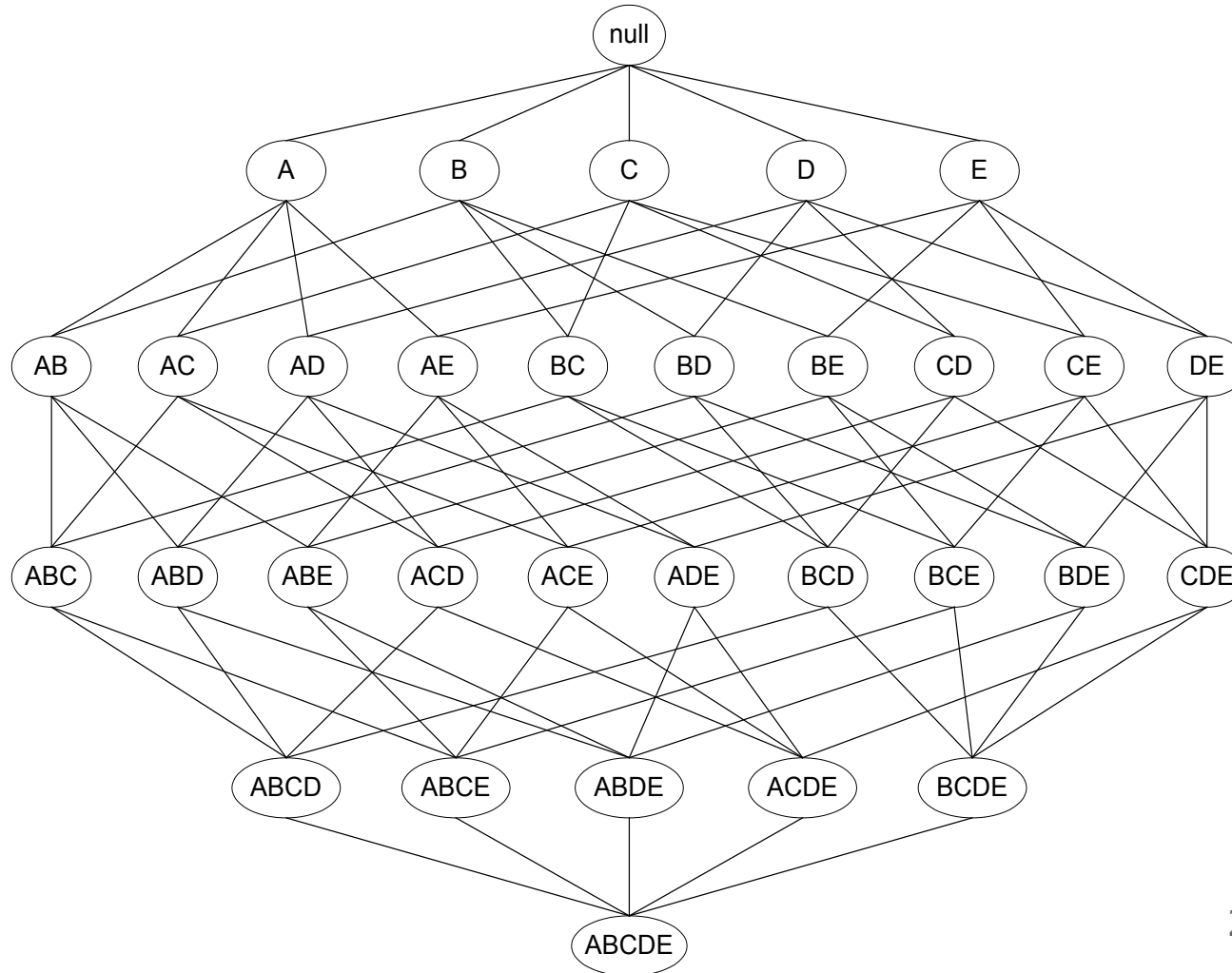
- Цел:
 - за дадено множество от транзакции T да се намерят всички правила / импликации, за които
 - $\text{support} \geq \text{minsup}$
 - $\text{confidence} \geq \text{minconf}$

Методи за решаване

- **Brute-force**
 - СПИСЪК НА ВСИЧКИ ВЪЗМОЖНИ
 - изчисляване на **support** и **confidence** за всяко
 - отхвърляне на неотговарящите на условията
- **Frequent Itemset Generation**
 - генериране на множество, за което $s \geq \text{minsup}$
- **Rule Generation**
 - генерират се правила с висока стойност на **c** за всяко множество, като всяко правило е бинарно подмножество на горното множество

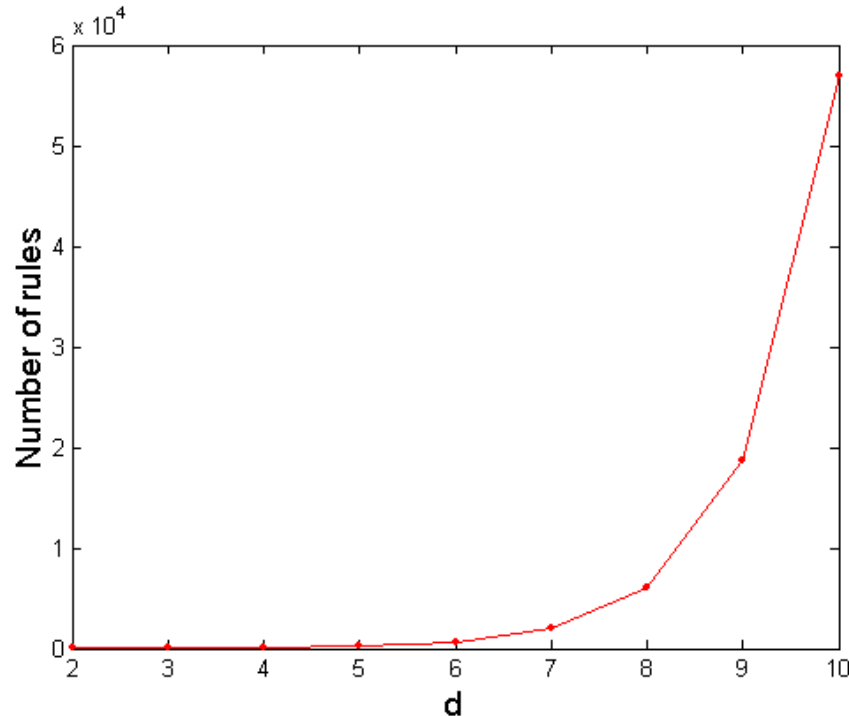
Пример

- за D елементи – 2^D кандидати



Изчислителна сложност

- За d елементи
 - брой кандидати множества $= 2^d$
 - брой възможни правила



$$R = \sum_{k=1}^{d-1} \left[\binom{d}{k} \times \sum_{j=1}^{d-k} \binom{d-k}{j} \right]$$
$$= 3^d - 2^{d+1} + 1$$

Ако $d=6$, $R = 602$ правила

Изчислителна сложност

- Колкото по-нисък е коефициентът **support**, толкова повече подмножества M^N участват
 - M - брой на елементите
 - N - max дължина на транзакциите
- Очаквана вероятност
 - ако $M=10^4$
 - вероятността да се вземе определен елемент = 10^{-4}
 - вероятността да се вземат 10 определени елемента = $\sim 10^{-40}$
 - вероятността точно тези елементи да се вземат 10^3 пъти в 10^9 транзакции?

Стратегии за опростяване

- Намаляване на **броя на кандидатите** (M)
- Намаляване на **броя на транзакциите** (N)
- Намаляване на **броя на сравненията** (NM)

Затворени и максимални подмножества

- Едно множество $\{a_1, \dots, a_{100}\}$ съдържа подмножества – комбинации от елементите:

$$\binom{100}{1} + \binom{100}{2} + \dots + \binom{100}{0} = 2^{100} - 1 = 1.27 * 10^{30}$$

- Голям брой , *max-patterns* – бавно изследване
- Решение: да не се изследват всички, а само по-важните: *затворени - closed patterns*
- подмножеството X е *closed*, ако X е *frequent* и не съществува друго, по-голямо множество Y, на което X е подмножество, *super-pattern* $Y \supset X$, със същия коефициент на *support* като X (Pasquier, et al. @ ICDT'99)
- подмножеството X е *max-pattern*, ако X е *frequent* и няма друго, по-голямо подмножество, което също е *frequent super-pattern*

$Y \supset X$ (Bayardo @ SIGMOD'98)

- *Closed pattern* е най-малката компресия
- Намалява се броят на подмножествата и правилата

Затворени подмножества

- никое от супер-множествата няма същата стойност на параметъра s

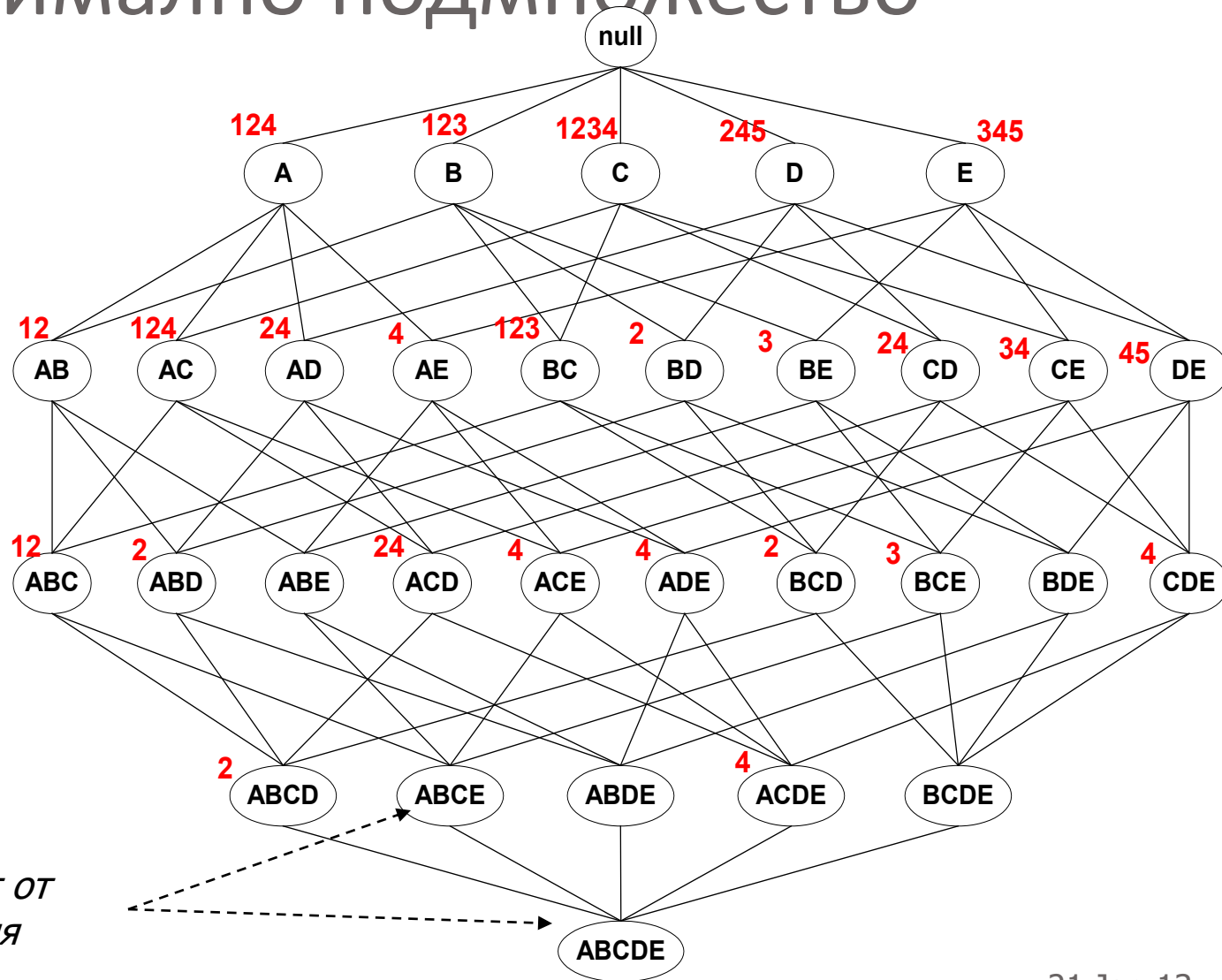
TID	Items
1	{A,B}
2	{B,C,D}
3	{A,B,C,D}
4	{A,B,D}
5	{A,B,C,D}

Itemset	Support
{A}	4
{B}	5
{C}	3
{D}	4
{A,B}	4
{A,C}	2
{A,D}	3
{B,C}	3
{B,D}	4
{C,D}	3

Itemset	Support
{A,B,C}	2
{A,B,D}	3
{A,C,D}	2
{B,C,D}	3
{A,B,C,D}	2

Максимално подмножество

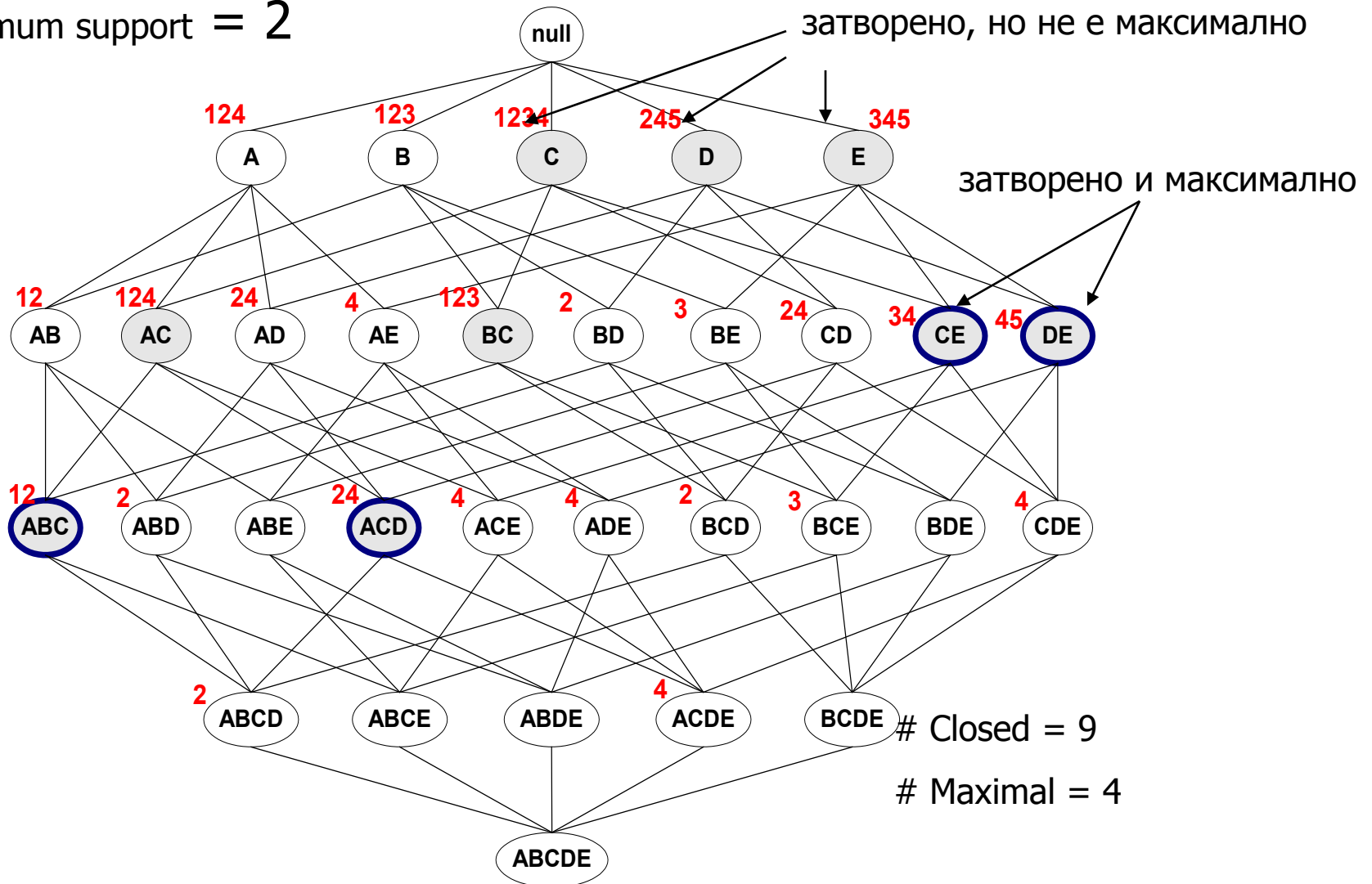
TID	Items
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE

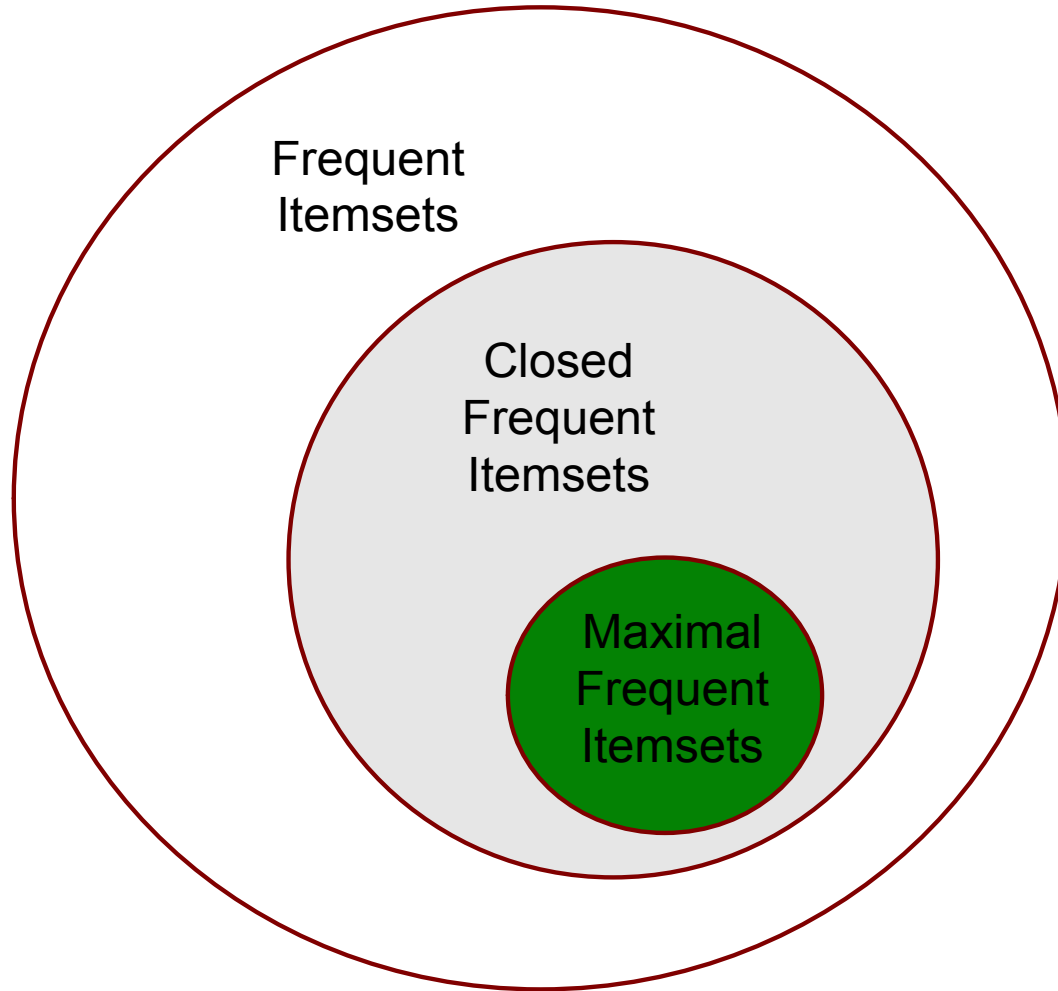


не се поддържат от
никая транзакция

Максимално и затворено подмножество

Minimum support = 2





Свойства и методи

- Свойства **downward closure**
 - Всяко подмножество на frequent множество също е frequent
 - Пример: Ако {компютър, принтер, мишка} е frequent, то {компютър, принтер} също е
 - т.е., всяка транзакция, която съдържа {компютър, принтер, мишка} също съдържа {компютър, мишка}
 - {компютър, мишка} е подмножество на {компютър, принтер, мишка}

Методи за откриване на асоциации

- Три подхода
 - Apriori (Agrawal & Srikant@VLDB'94)
 - Freq pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD'00)
 - Vertical data format approach (Charm—Zaki & Hsiao @SDM'02)

Apriori алгоритъм

- Принцип на отстраняване на кандидатите
 - Ако има не-често срещано множество, не е необходимо да се изчисляват неговите по-старши множества
(Agrawal & Srikant @VLDB'94, Mannila, et al. @ KDD' 94)
 - Apriori property: едно множество участва само ако всички негови подмножества са често срещани ($S > S_{min}$)
- Итеративен подход
 - k itemsets се използват за изчисляване на $k+1$

Метод

- Чете се базата с данни за транзакциите и се разглеждат всички срещаци се елементи, които отговарят на правилата за минимална поддръжка
- Генерира се множество на кандидатите от
 - едноелементни подмножества, $k=1$ frequent itemsets,
 - изчисляват се честотите им на срещане
 - двуелементни множества, $k=2$ frequent itemsets
 - изчисляват се честотите им на срещане
 - триелементни, $k=3$ frequent itemsets
 - изчисляват се честотите им на срещане
 - отстраняват се тези, които нямат frequent подмножества
 - и т.н.
- Процесът продължава до изчерпване на срещаните транзакции

Приложение на Apriori алгоритъм

- Генериране на кандидати
 1. Обединяване на $L_k * L_k$ в по-голямо множество
 2. Отстраняване на неотговарящите на правилата
- Пример:
 - $L_3 = \{abc, abd, acd, ace, bcd\}$
 - обединяване на $L_3 * L_3$
 - $abcd$ от abc и abd
 - $acde$ от acd и ace
 - отстраняване
 - $acde$ се премахва, защото ade не е в L_3
 - $C_4 = \{abcd\}$

Пример за приложение на Apriori алгоритъм

MinSup= 2

D= 4 (бр. транзакции)

Database TDB

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

C_1
 $\kappa=1$

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

L_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3



C_2

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

2nd scan

C_2

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

$\kappa=2$

L_2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2



$\kappa=3$

C_3

Itemset
{B, C, E}

3rd scan

L_3

Itemset	sup
{B, C, E}	2

Генериране на правила за асоциации

1. За всяко често срещано множество itemset l се генерира непразни подмножества s

напр. за $l = \{i1, i2, i5\}$,

подмножествата са $\{i1, i2\}$, $\{i1, i5\}$, $\{i2, i5\}$, $\{i1\}$, $\{i2\}$, $\{i5\}$

2. За всяко непразно подмножество s се изчислява асоциацията

$$s \rightarrow (l-s)$$

като условна вероятност (confidence C)

$$P(s | l-s) = P(l-s) / P(s)$$

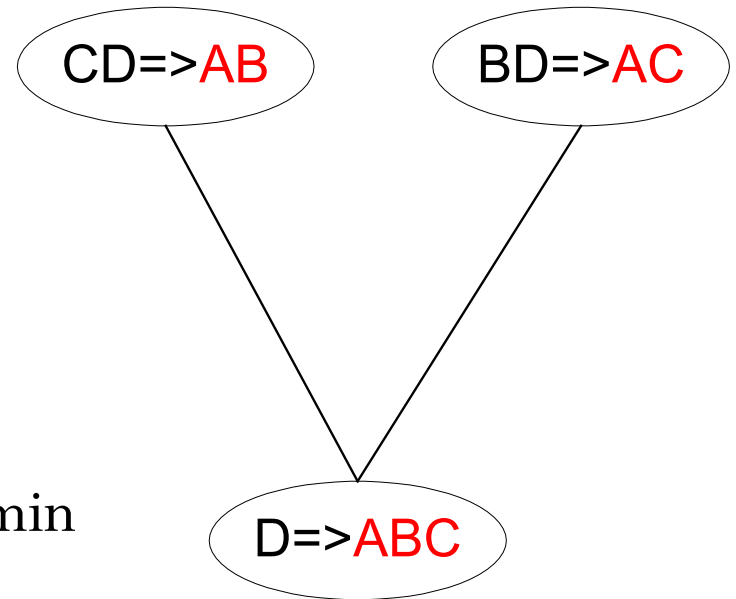
3. Игнорират се всички подмножества, за които

$$C < C_{min}$$

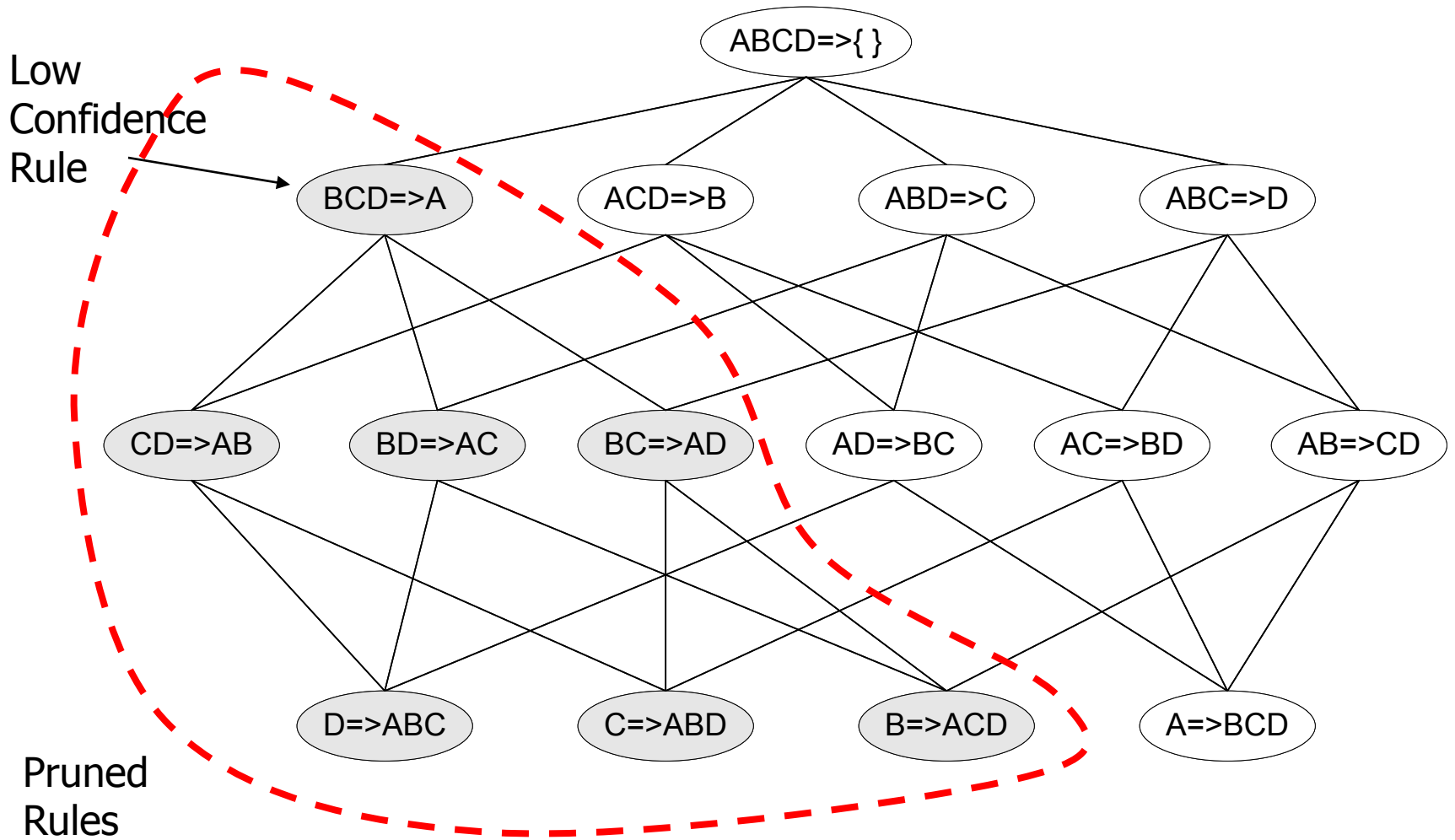
Генериране на правила за асоциации

- join rule
 - $(CD \Rightarrow AB, BD \Rightarrow AC)$
резултат: $D \Rightarrow ABC$

- prune rule
 - $D \Rightarrow ABC$,
подмножеството $AD \Rightarrow BC$
е отстранено, защото $C < C_{min}$



Пример



Оценка на правилата за асоциативност

- Проблеми
 - Броят на кандидатите може да бъде много голям
 - Една транзакция може да съдържа много кандидати
- Методи за повишаване на ефективността
 - хеширане на кандидатите – групиране на кандидатите от едно множество по честотата на срещане
 - намаляване на броя на транзакциите
 - разделяне на транзакциите на отделни части за изследване - partitioning
 - изследване на извадка, а не на цялото множество транзакции - sampling
 - и др.

Вертикално представяне на данните

Horizontal
Data Layout

TID	Items
1	A,B,E
2	B,C,D
3	C,E
4	A,C,D
5	A,B,C,D
6	A,E
7	A,B
8	A,B,C
9	A,C,D
10	B

Vertical Data Layout

A	B	C	D	E
1	1	2	2	1
4	2	3	4	3
5	5	4	5	6
6	7	8	9	
7	8	9		
8	10			
9				

Вертикално представяне на данните

- За всички множества в колоните във вертикалната таблица са изчисляват сечения от подмножества
- Прилага се свойството Apriori

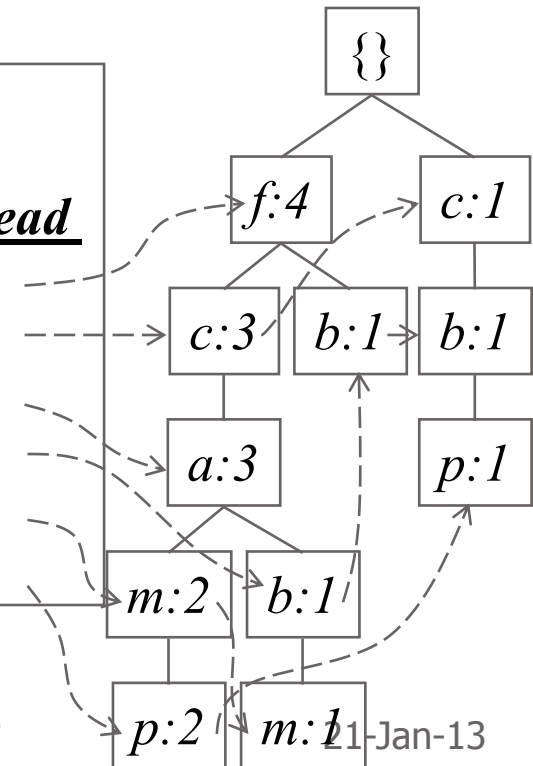
FP-tree

<i>TID</i>	<i>Items bought</i>	<i>(ordered) frequent items</i>
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}
300	{b, f, h, j, o, w}	{f, b}
400	{b, c, k, s, p}	{c, b, p}
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

min_support = 3

1. Откриват се frequent 1-itemset (с един елемент)
2. Сортират се по броя на срещанията
3. Построява се дървото

Header Table	
<u><i>Item frequency head</i></u>	
<i>f</i>	4
<i>c</i>	4
<i>a</i>	3
<i>b</i>	3
<i>m</i>	3
<i>p</i>	3



F-list = f-c-a-b-m-p

MaxMiner: Mining Max-Patterns

- Първо сканиране
 - намиране на повтарящите се
A, B, C, D, E

Tid	Items
10	A, B, C, D, E
20	B, C, D, E,
30	A, C, D, F

- Второ сканиране
 - изчисляване на правилата
AB, AC, AD, AE, ABCDE

BC, BD, BE, BCDE

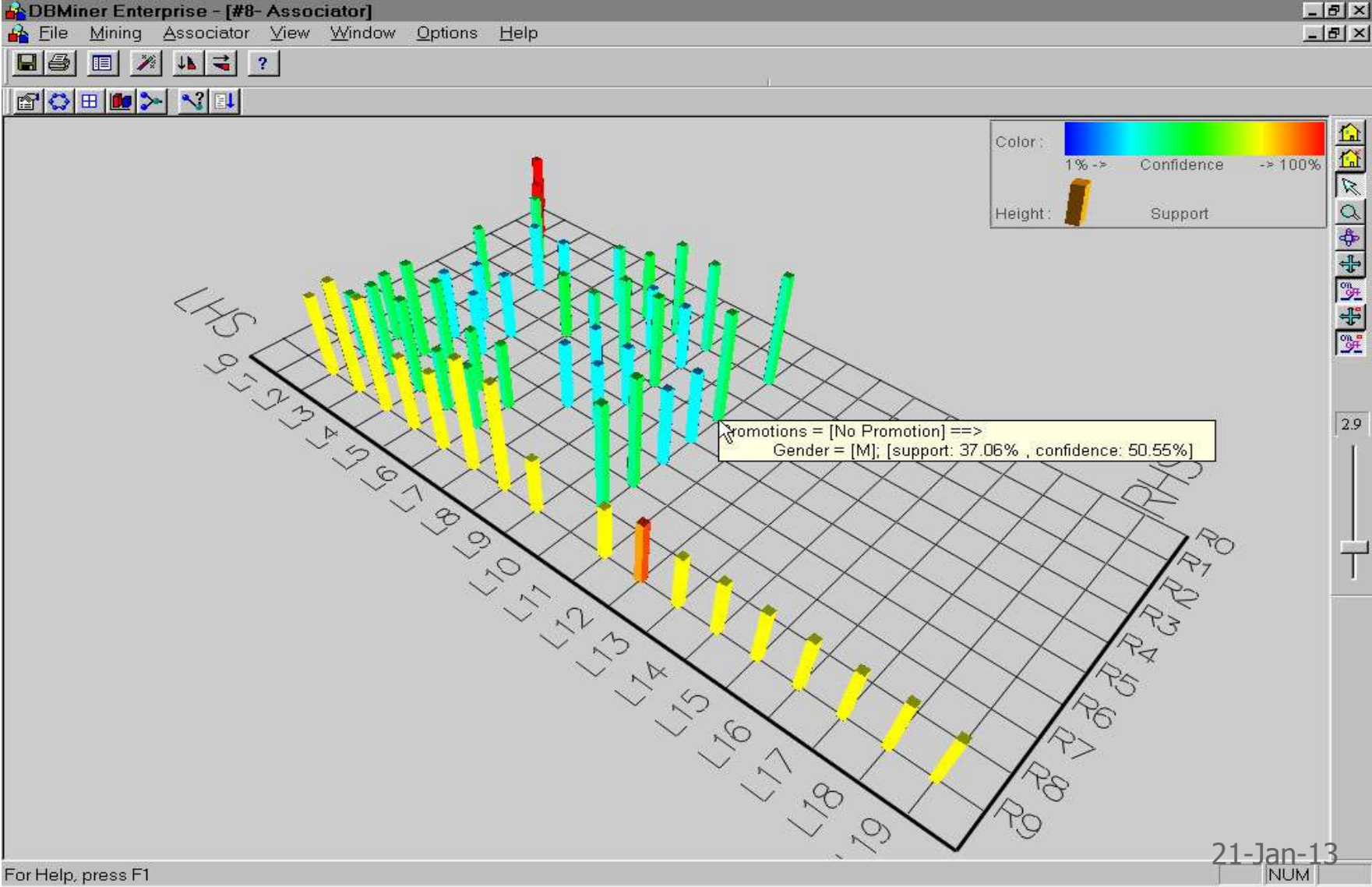
CD, CE, CDE, DE

Potential max-patterns

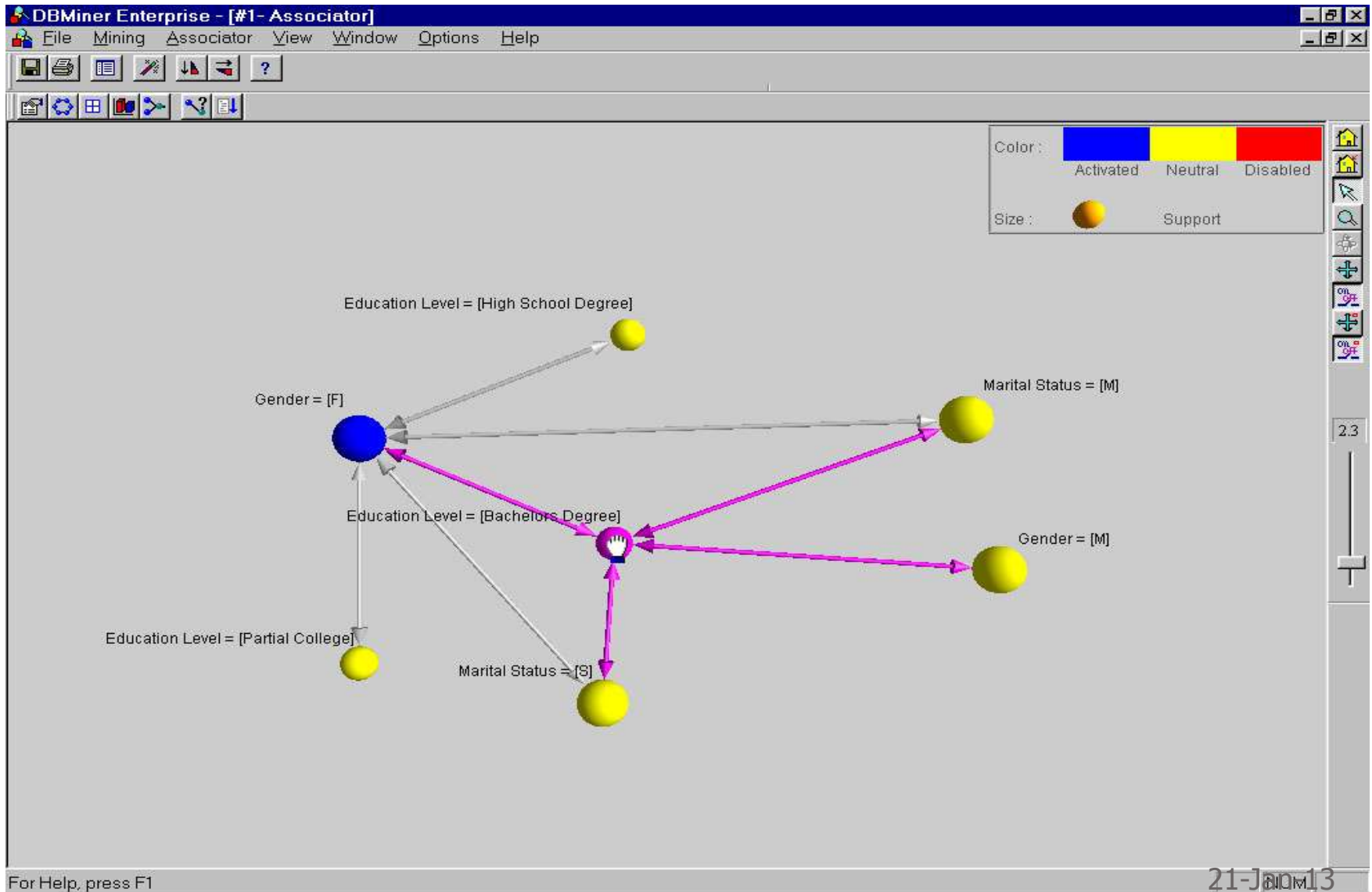


- BCDE е max-pattern, няма нужда да се проверяват BCD, BDE, CDE
- R. Bayardo. Efficiently mining long patterns from databases.

Визуализация на правилата



Визуализация на правилата



Измерване на полезността на асоциациите

- Някои асоциации са безинтересни, не полезни или подвеждащи
- Напр.
 - *play basketball* \Rightarrow *eat cereal* [40%, 66.7%] **подвеждаща, защото**
 - общо % студенти, които ядат мюсли е 75%, т.е. $> 66.7\%$
- *play basketball* \Rightarrow *not eat cereal* [20%, 33.3%] **е по-точно, въпреки, че има по-слаби коефициенти**
- Необходима е и друга мярка
- **Корелация**
между множествата

	Basketball	Not basketball	Sum (row)
Cereal	2000	1750	3750
Not cereal	1000	250	1250
Sum(col.)	3000	2000	5000

Видове мерки на полезността

#	Measure	Formula
1	ϕ -coefficient	$\frac{P(A,B) - P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
2	Goodman-Kruskal's (λ)	$\frac{\sum_j \max_k P(A_j, B_k) + \sum_k \max_j P(A_j, B_k) - \max_j P(A_j) - \max_k P(B_k)}{2 - \max_j P(A_j) - \max_k P(B_k)}$
3	Odds ratio (α)	$\frac{P(A,B)P(\bar{A},\bar{B})}{P(A,\bar{B})P(\bar{A},B)}$
4	Yule's Q	$\frac{P(A,B)P(\bar{A}\bar{B}) - P(A,\bar{B})P(\bar{A},B)}{P(A,B)P(\bar{A}\bar{B}) + P(A,\bar{B})P(\bar{A},B)} = \frac{\alpha - 1}{\alpha + 1}$
5	Yule's Y	$\frac{\sqrt{P(A,B)P(\bar{A}\bar{B})} - \sqrt{P(A,\bar{B})P(\bar{A},B)}}{\sqrt{P(A,B)P(\bar{A}\bar{B})} + \sqrt{P(A,\bar{B})P(\bar{A},B)}} = \frac{\sqrt{\alpha} - 1}{\sqrt{\alpha} + 1}$
6	Kappa (κ)	$\frac{P(A,B) + P(\bar{A},\bar{B}) - P(A)P(B) - P(\bar{A})P(\bar{B})}{1 - P(A)P(B) - P(\bar{A})P(\bar{B})}$
7	Mutual Information (M)	$\frac{\sum_i \sum_j P(A_i, B_j) \log \frac{P(A_i, B_j)}{P(A_i)P(B_j)}}{\min(-\sum_i P(A_i) \log P(A_i), -\sum_j P(B_j) \log P(B_j))}$
8	J-Measure (J)	$\max \left(P(A, B) \log \left(\frac{P(B A)}{P(B)} \right) + P(\bar{A}\bar{B}) \log \left(\frac{P(\bar{B} \bar{A})}{P(\bar{B})} \right), \right. \\ \left. P(A, B) \log \left(\frac{P(A B)}{P(A)} \right) + P(\bar{A}\bar{B}) \log \left(\frac{P(\bar{A} \bar{B})}{P(\bar{A})} \right) \right)$
9	Gini index (G)	$\max \left(P(A)[P(B A)^2 + P(\bar{B} A)^2] + P(\bar{A})[P(B \bar{A})^2 + P(\bar{B} \bar{A})^2] \right. \\ \left. - P(B)^2 - P(\bar{B})^2, \right. \\ \left. P(B)[P(A B)^2 + P(\bar{A} B)^2] + P(\bar{B})[P(A \bar{B})^2 + P(\bar{A} \bar{B})^2] \right. \\ \left. - P(A)^2 - P(\bar{A})^2 \right)$
10	Support (s)	$P(A, B)$
11	Confidence (c)	$\max(P(B A), P(A B))$
12	Laplace (L)	$\max \left(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2} \right)$
13	Conviction (V)	$\max \left(\frac{P(A)P(\bar{B})}{P(\bar{A}B)}, \frac{P(B)P(\bar{A})}{P(\bar{B}A)} \right)$
14	Interest (I)	$\frac{P(A,B)}{P(A)P(B)}$
15	cosine (IS)	$\frac{P(A,B)}{\sqrt{P(A)P(B)}}$
16	Piatetsky-Shapiro's (PS)	$P(A, B) - P(A)P(B)$
17	Certainty factor (F)	$\max \left(\frac{P(B A) - P(B)}{1 - P(B)}, \frac{P(A B) - P(A)}{1 - P(A)} \right)$
18	Added Value (AV)	$\max(P(B A) - P(B), P(A B) - P(A))$
19	Collective strength (S)	$\frac{P(A,B) + P(\bar{A}\bar{B})}{P(A)P(B) + P(\bar{A})P(\bar{B})} \times \frac{1 - P(A)P(B) - P(\bar{A})P(\bar{B})}{1 - P(A,B) - P(\bar{A}\bar{B})}$
20	Jaccard (ζ)	$\frac{P(A,B)}{P(A) + P(B) - P(A,B)}$
21	Klogsen (K)	$\sqrt{P(A, B)} \max(P(B A) - P(B), P(A B) - P(A))$

Корелация

- Методи

- lift

$$lift = \frac{P(A \cup B)}{P(A)P(B)}$$

- за lift=1
 - няма корелация
 - lift>1
 - положителна
 - lift<1
 - отрицателна

- χ^2

$$\chi^2 = \sum \frac{(observed - expected)^2}{expected}$$

$$lift(B, C) = \frac{2000 / 5000}{3000 / 5000 * 3750 / 5000} = 0.89$$

$$lift(B, \neg C) = \frac{1000 / 5000}{3000 / 5000 * 1250 / 5000} = 1.33$$

- Други методи

- all_confidence
 - cosine

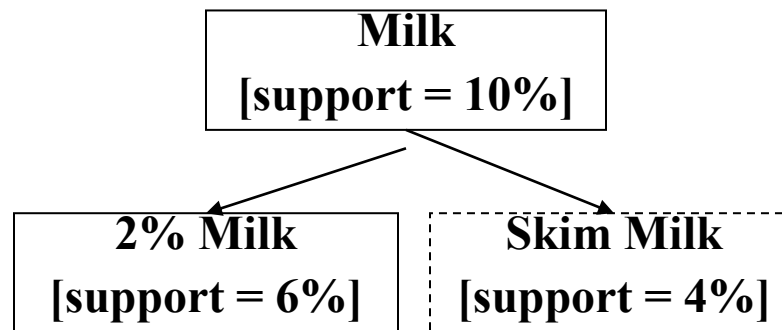
Multiple-Level Association Rules

- Елементите съставят йерархии
 - напр. мляко + мляко 2%
- По-ниско ниво – по слаба поддръжка
- Exploration of *shared* multi-level mining
(Agrawal & Srikant@VLB'95, Han & Fu@VLDB'95)

uniform support

Level 1
min_sup = 5%

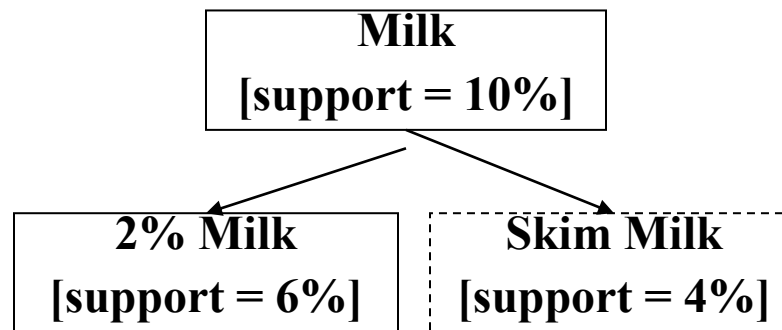
Level 2
min_sup = 5%



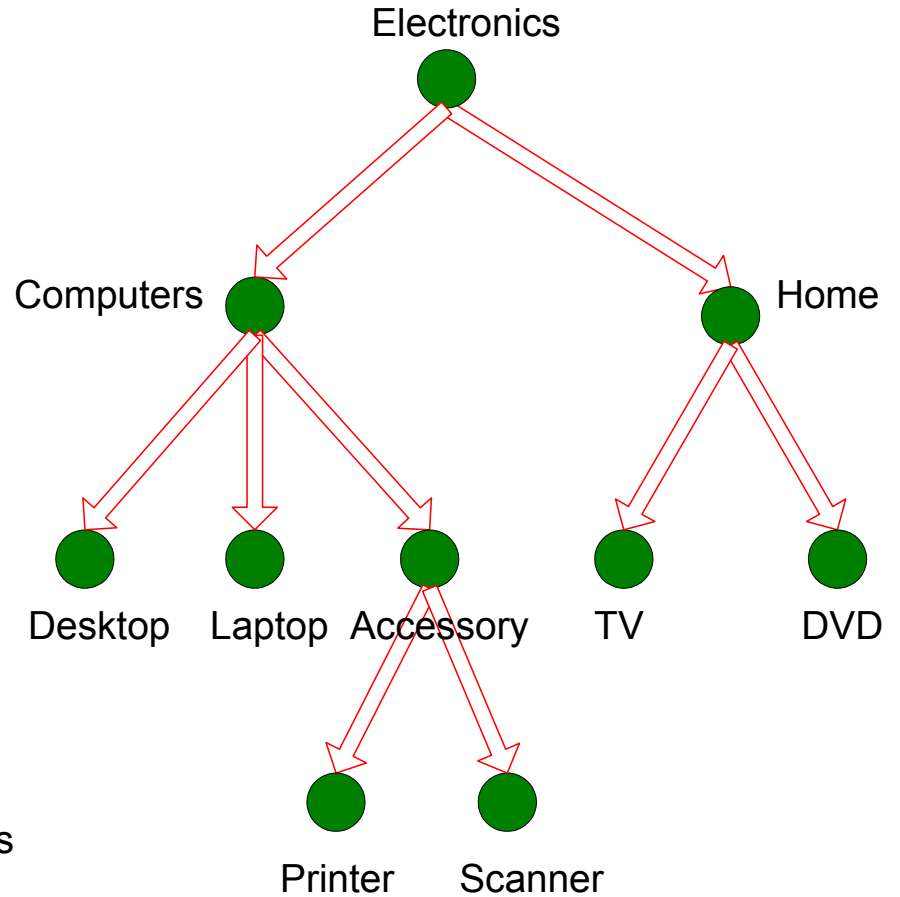
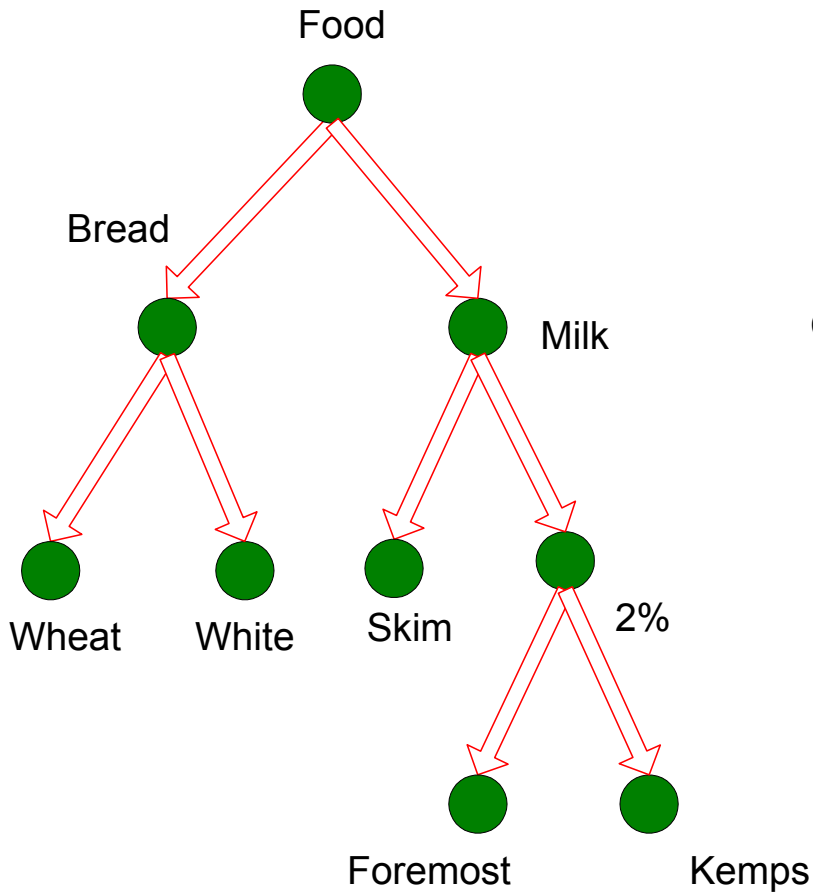
reduced support

Level 1
min_sup = 5%

Level 2
min_sup = 3%



Multi-level Association Rules



Multi-Dimensional Association

- До тук: едномерни асоциации
 - само един предикат: покупки
 - $\text{buys}(X, \text{"milk"}) \Rightarrow \text{buys}(X, \text{"bread"})$
- Многомерни правила
 - ≥ 2 дименсии или правила
- Интер-дименсионални асоциации
 - многомерни, без повторение на предикатите (*no repeated predicates*)
 - $\text{age}(X, \text{"19-25"}) \wedge \text{occupation}(X, \text{"student"}) \Rightarrow \text{buys}(X, \text{"coke"})$
- Хибридни
 - многомерни, с повторение на предикатите (*repeated predicates - buys*)
 - $\text{age}(X, \text{"19-25"}) \wedge \text{buys}(X, \text{"popcorn"}) \Rightarrow \text{buys}(X, \text{"coke"})$
- Categorical Attributes - data cube
 - краен брой възможни стойности
 - без подредба
- Quantitative Attributes - discretization, clustering, gradient approaches
 - числени атрибути
 - подреждане по скала

Количествени и качествени атрибути

Session Id	Country	Session Length (sec)	Number of Web Pages viewed	Gender	Browser Type	Buy
1	USA	982	8	Male	IE	No
2	China	811	10	Female	Netscape	No
3	USA	2125	45	Female	Mozilla	Yes
4	Germany	596	4	Male	IE	Yes
5	Australia	123	9	Male	Mozilla	No
...

Пример за правило:

$\{\text{Number of Pages} \in [5,10) \wedge (\text{Browser}=\text{Mozilla})\} \rightarrow \{\text{Buy} = \text{No}\}$

Количествени атрибути

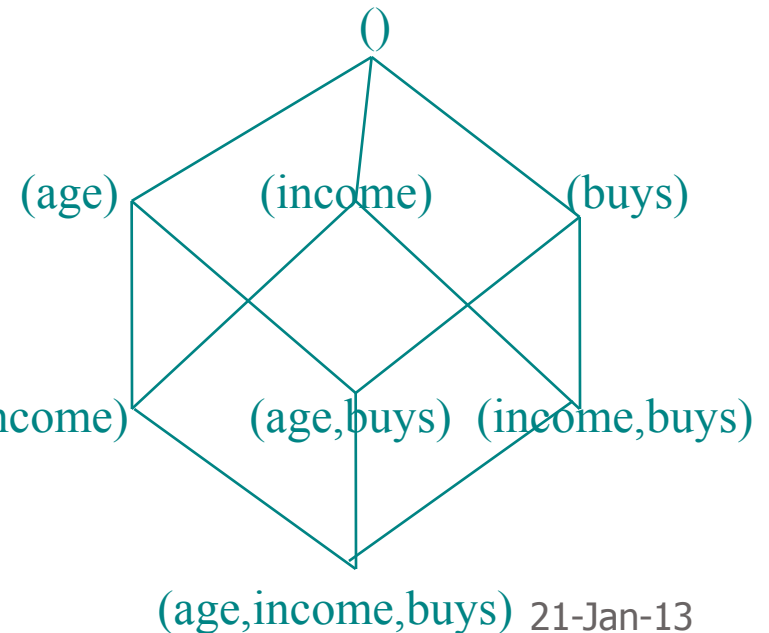
- Числови атрибути, като `age` или `salary`
- Ако се вземат всички възрасти и всички заплати – твърде много комбинации за изследване
- Необходимо е дискретизиране на стойностите
- Видове методи за дискретизиране
 - Статично дискретизиране - предварително дефинирани йерархии (`data cube methods`)
 - Динамично дискретизиране, според наблюденията и разпределението на данните - `data bins` (quantitative rules, e.g., Agrawal & Srikant@SIGMOD96)
- Clustering: Distance-based association (Yang & Miller@SIGMOD97)
 - One dimensional clustering, then association
- Deviation: (Aumann and Lindell@KDD99)
 - Sex = female => Wage: mean=\$7/hr (overall mean = \$9)

Статично дискретизиране

- Предварително
- Числените стойности се заменят с интервални
- Намирането на всички frequent k -predicate sets изисква k или $k+1$ сканирания на таблицата в базата

- Data cube
- Клетките на n -dimensional cuboid кореспондират с множеството предикати

- По-бързо изследване



Отрицателни или рядко срещани шаблони

- Rare patterns
 - Слаби стойности на коефициентите, но интересни шаблони
 - напр. кой купува Rolex
 - Mining: Setting individual-based or special group-based support threshold for valuable items
- Negative patterns
 - несъвместими
 - напр. никой не си купува два различни джипа едновременно
 - негативно корелираните множества са по-интересни за изследване

Други видове изследвания

- Constraint-based mining
 - интерактивен процес, насочван от потребителя
- Meta-Rule Guided Mining
- Mining Compressed Patterns: δ -clustering

Apriori + Constraint

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

C_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_2

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

C_3

itemset
{2 3 5}

L_3

itemset	sup
{2 3 5}	2

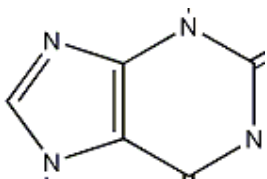
Constraint:
 21-Jan-13
 $\text{Sum}\{S.\text{price}\} < 5$

Интерпретиране на резултатите

хляб

мляко

- Смислени?
- Значение?
- Употреба?



морфологична информация и статистика



семантична информация

не всички са полезни



Да се аотират със семантична информация