Поиск сочетаний

- Характерные наборы – группы признаков, которые часто встречаются совместно в транзакциях
- Варианты названий: Market Basket Analysis
- Критерий характерности набора задается пользователем Supp* (например, 10 – 30 % от общего объема данных)
- Алгоритмы поиска сочетаний

  ~~- Полный перебор возможных комбинаций~~

  - Алгоритм Apriori
  - Алгоритм FPG-дерева
  - eclat
  - ..
  - АФП
- Варианты задачи: поиск обобщенных правил, обработка интервальных показателей, поиск временных последовательностей

Характеристики наборов

- Поддержка набора (support) – доля объектов, содержащих признаки набора, от общего числа объектов

$$Supp(a,b,c) = \frac{N(a,b,c)}{N}$$

- Порог характерности Supp* - поддержка, которая принимается минимально допустимой для набора

Алгоритм поиска характерных комбинаций

Базовый алгоритм:

Для всех k = 1 .. m

1. Генерация наборов-кандидатов длины k
2. Подсчет поддержки для каждого набора-кандидата
3. Отбор набор, удовлетворяющих заданному порогу *Supp**

Алгоритм обладает экспоненциальной сложностью.

Для m = 10 (признаки) число наборов-кандидатов равно:

$$2^{10} = C_{10}^{1} + C_{10}^{2} + .. + C_{10}^{10} = 10 + 45 + 120 + 210 + 252 + 210 + 120 + 45 + 10 + 1 = 1023$$

Для m=15 число наборов-кандидатов равно $2^{15} = 32768$

Свойство антимонотонности

- Поддержка набора из p-признаков не превышает минимальной поддержки по всем поднаборам из (p-k) признаков, где k = 1 .. (p-1)

A	B	C
1	0	1
0	1	1
1	1	1
1	0	1

$$Supp_{AB} = 1 \qquad Supp_{AC} = 3$$

$$Supp_{BC} = 2$$

$$Supp_{ABC} \le \min(Supp_{AB}, Supp_{BC}, Supp_{AC})$$

Алгоритм Apriori

- Свойство антимонотонности позволяет уменьшить число кандидатов на каждой k-итерации
- Алгоритм Apriori учитывает свойство антимонотонности: на каждой k-итерации при формировании набора-кандидата *set* учитывается поддержка всех поднаборов длины k-1
- Если множество характерных наборов, полученных на предыдущей k-1 итерации, не содержит какой-либо поднабор набора *set*, значит набор *set* является заведомо нехарактерным – поддержку для него не считаем.

Agrawal & Srikant @VLDB'94,
Mannila, et al. @ KDD' 94

Генерации наборов

K=1

A B C D E F

K=2

AB AD BD AE BE DE

K=3

ABD ADE

A	B	C	D	E	F
1	0	1	1	1	0
0	1	0	1	0	0
1	1	0	1	0	1
1	0	1	0	0	0
1	1	0	1	1	1
1	1	0	1	1	0

$$Supp^{*} = 3/6$$

Алгоритм поиска характерных комбинаций Apriori

$APRIORI(D)$

// Вход: D – таблица данных $N \times M$

// Выход: $L = \{L_i\}$ множество характерных комбинаций

$L_1 = find1L(D);$

$for\ (k = 2;\ L_{k-1} \neq \emptyset; k++)$

$C_k = Candidates(L_{k-1});$

$for\ each\ c \in C_k$

$supp_C = CalcSupport(D, c);$

$if\ supp_C \geq supp^*$

$L_k = L_k \cup c$

$CANDIDATES\ (L_{k-1})$

// Вход: L_{k-1} - множество характерных наборов длины k

// Выход: C_k - множество кандидатов длины k

$foreach\ l_i \in L_{k-1}$

$foreach\ l_j \in L_{k-1}$

$if\ |l_i \cap l_j| = k-2$

$l = l_i \cup l_j$

$if\ CheckSubsets(l, L_{k-1})$

$C_k = C_k \cup l$

FPG-алгоритм

- Алгоритм Frequent Pattern-Growth Strategy (FPG)
- Предложен в 2000-м году J. Han, J. Pei, and Y. Yin, SIGMOD’ 00
- В основе метода лежит предобработка базы транзакций и построение компактной древовидной структуры, называемой Frequent-Pattern Tree – *дерево популярных наборов*.
- Позволяет избежать затратной процедуры генерации кандидатов, характерной для алгоритма Apriori.
- Отсутствие интенсивной работы с исходной системой данных

Формирование FP-дерева

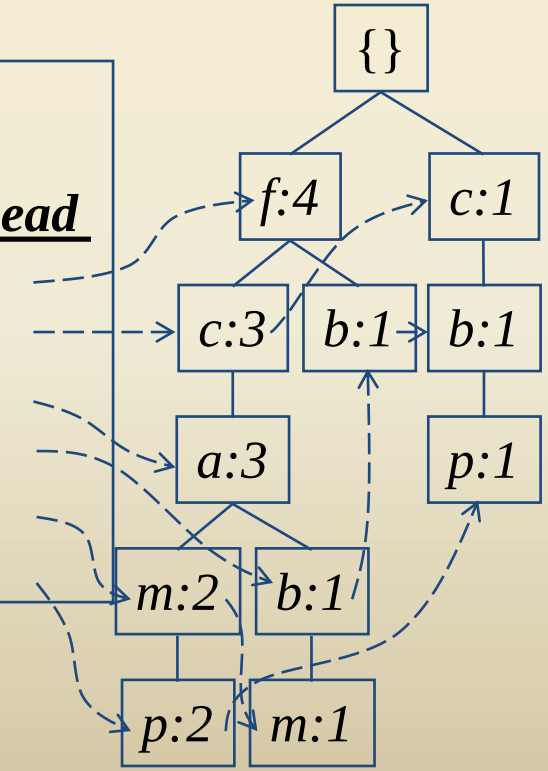
TID	Items bought	(ordered) frequent items
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}
300	{b, f, h, j, o, w}	{f, b}
400	{b, c, k, s, p}	{c, b, p}
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}

min_support = 3

1. Найти частоты отдельных признаков
2. Сортировка признаков в транзакциях
3. Построение дерева с последовательным обходом транзакций

Header Table

Item	frequency	head
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	



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

• Сканирование БД транзакций и отбор часто встречающихся признаков (в примере Supp* = 3).
• Упорядочивание наборов в порядке убывания поддержки:

(c, 6), (b, 5), (d, 5), (e, 5), (a, 3).

• Упорядочивание признаков в транзакциях по убыванию поддержки

N	Исходные наборы
1	a b c d e
2	a b c
3	a c d e
4	b c d e
5	b c
6	b d e
7	c d e

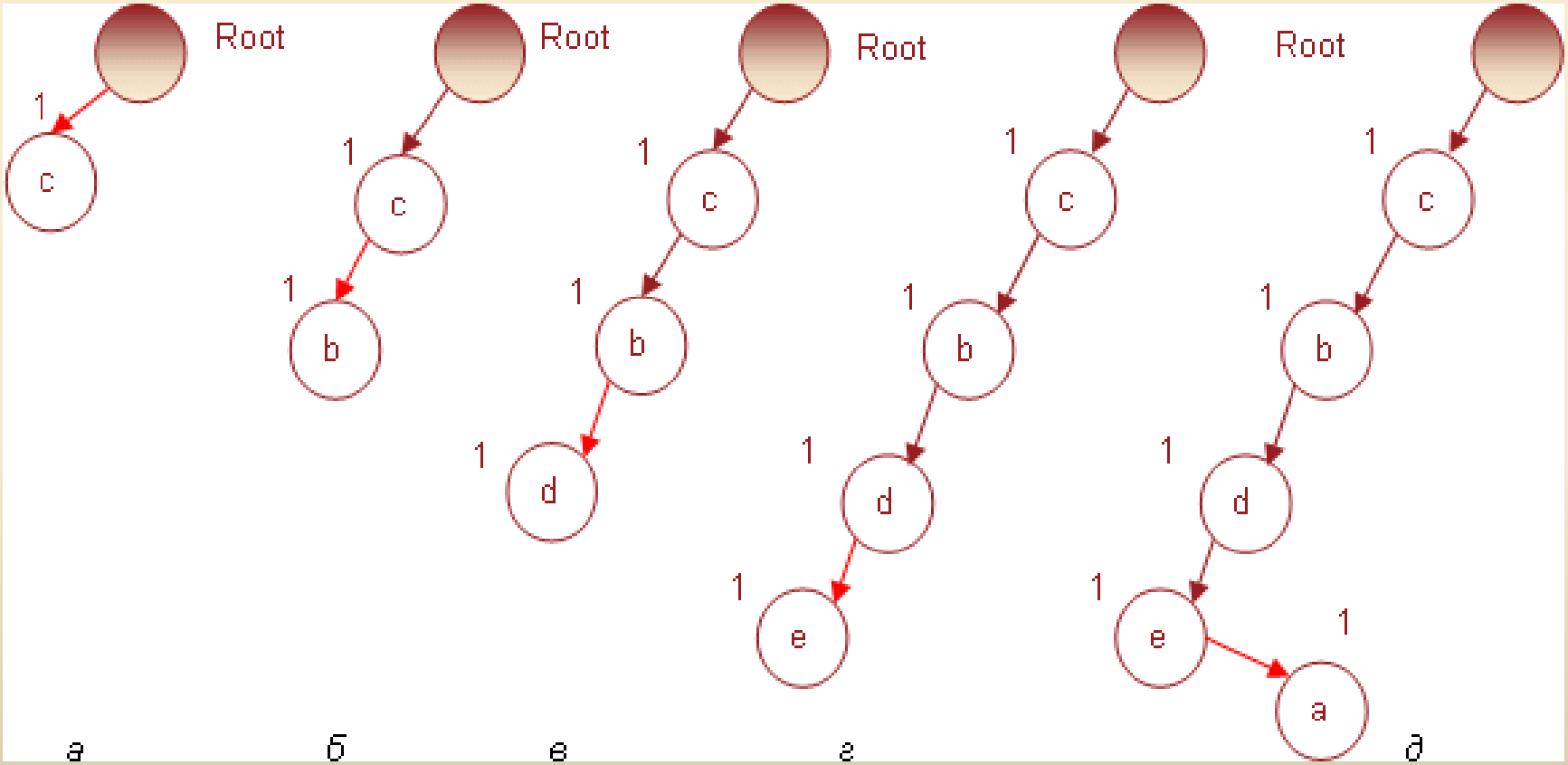
N	Упорядоченные наборы
1	c b d e a
2	c b a
3	c d e a
4	c b d e
5	c b
6	b d e
7	c d e

Построение FPG-дерева

- Каждый узел дерева, кроме корневого, соответствует отдельному признаку; один признак может встречаться несколько раз в дереве
- Узлы образуются при последовательном пропускании транзакций через дерево, начиная с корневого узла
- Признаки транзакции последовательно сверяются с текущим узлом дерева
- Если для очередного признака транзакции в дереве встречается одноименный узел, то индекс этого узла инкрементируется.
- Следующий признак транзакции начинает проверку со следующего дочернего узла дерева
- Если признак транзакции не соответствует узлу, то формируется альтернативная ветка дерева; индекс нового узла равен 1.

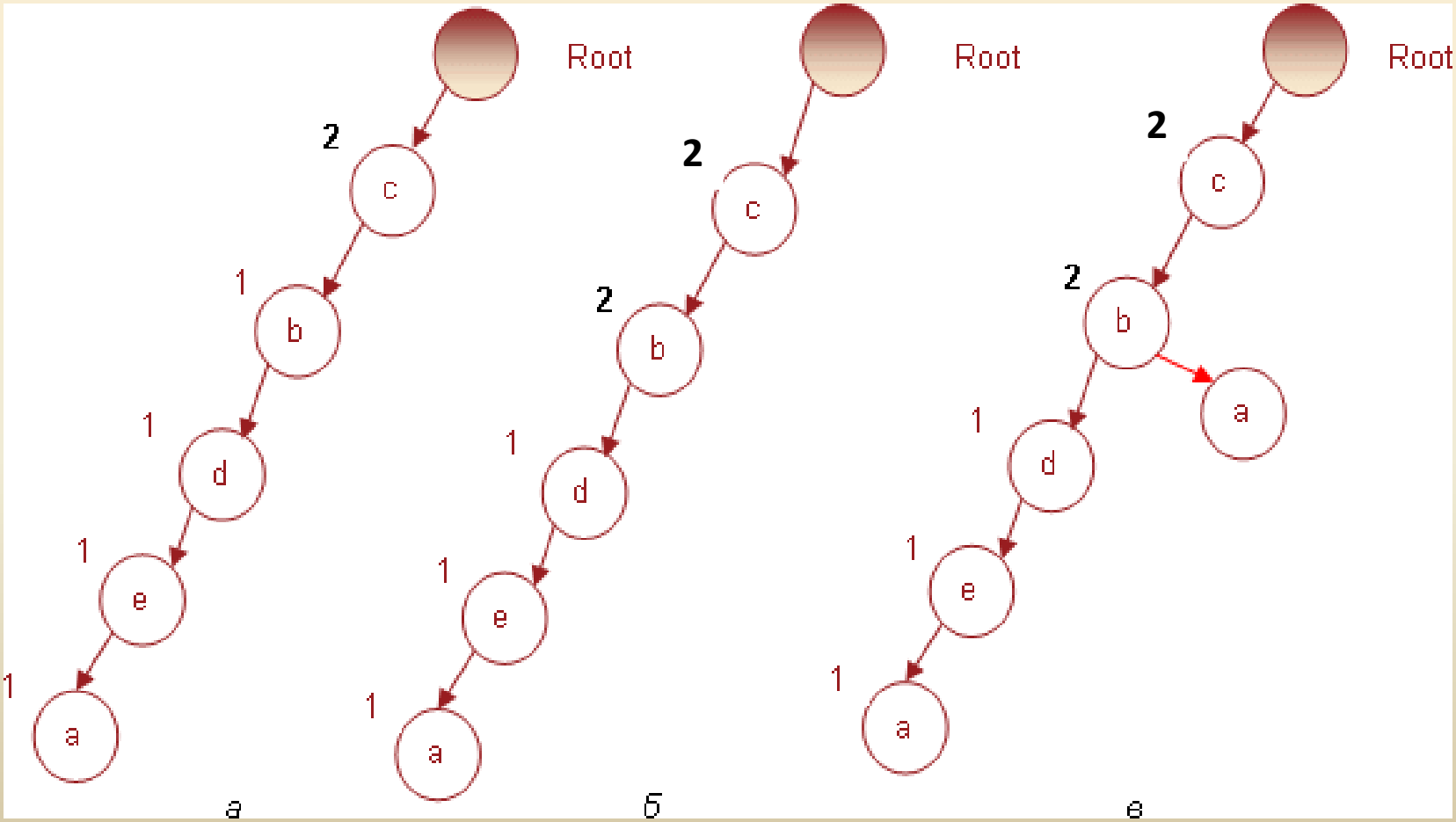
#1: c b d e a

#2: c b a


Root
1
c
Root
1
c
1
b
Root
1
c
1
b
1
d
Root
1
c
1
b
1
d
1
e
Root
1
c
1
b
1
d
1
e
1
a
а
б
в
г
д

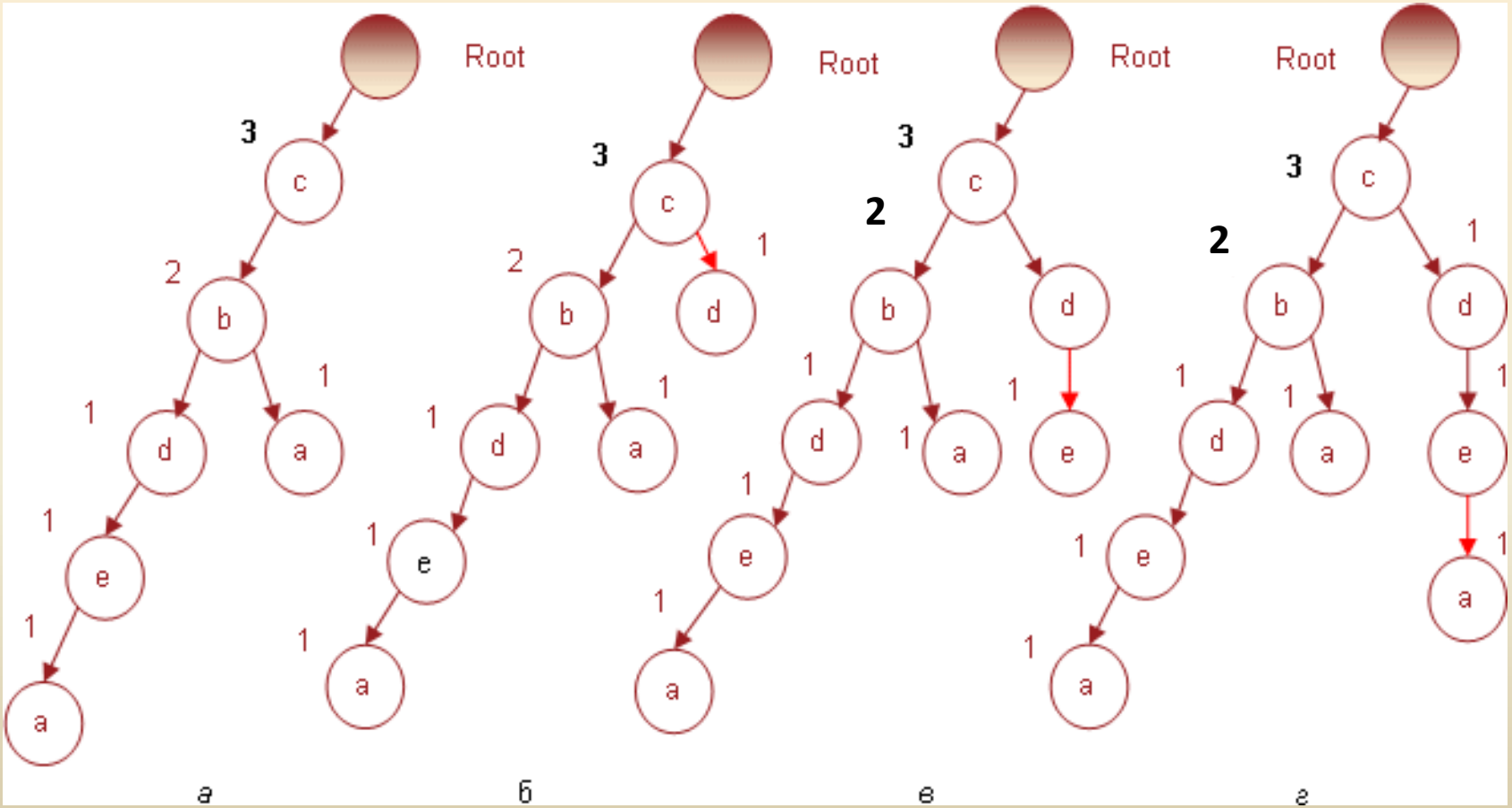
#2: c b a

#3: c d e a

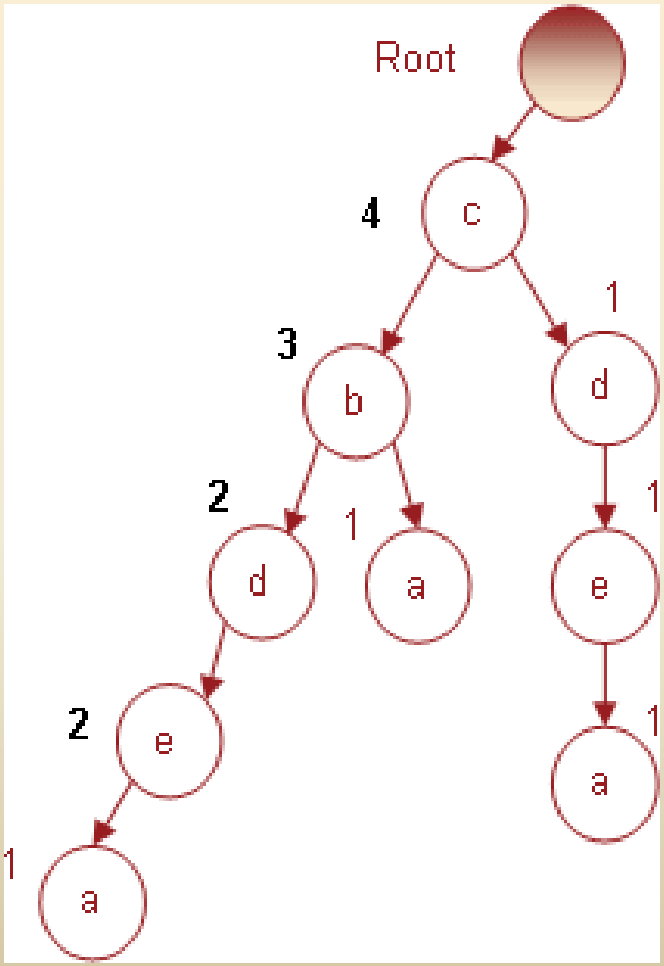


#3: c d e a

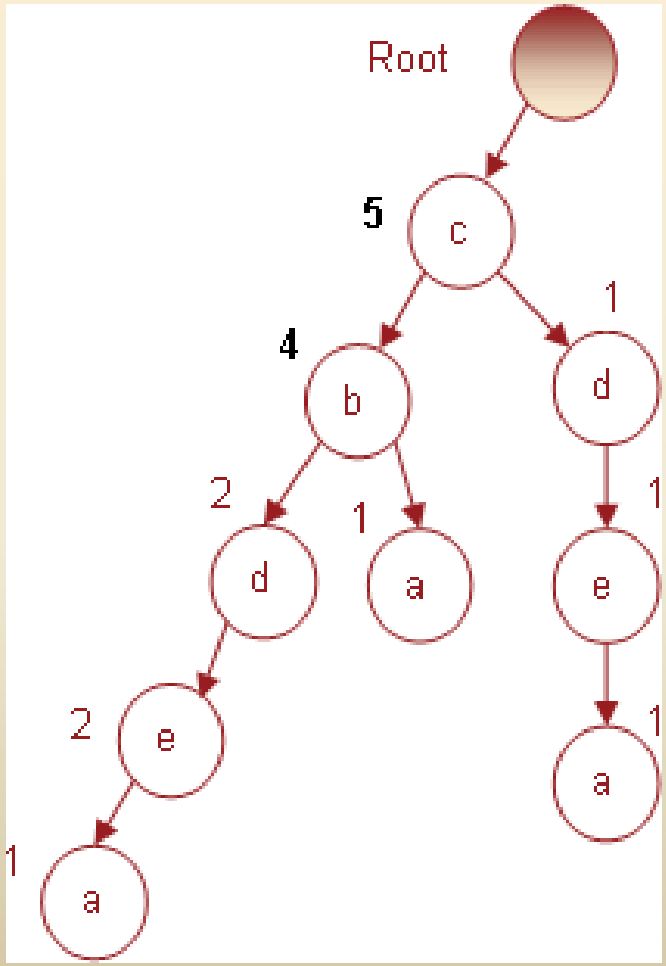
#4: c b d e


Root
3
c
2
b
1
d
a
1
e
1
a
а
Root
3
c
2
b
1
d
a
1
e
1
a
б
Root
3
c
2
b
d
1
d
a
1
e
1
a
в
Root
3
c
2
b
d
1
d
a
1
e
1
a
г

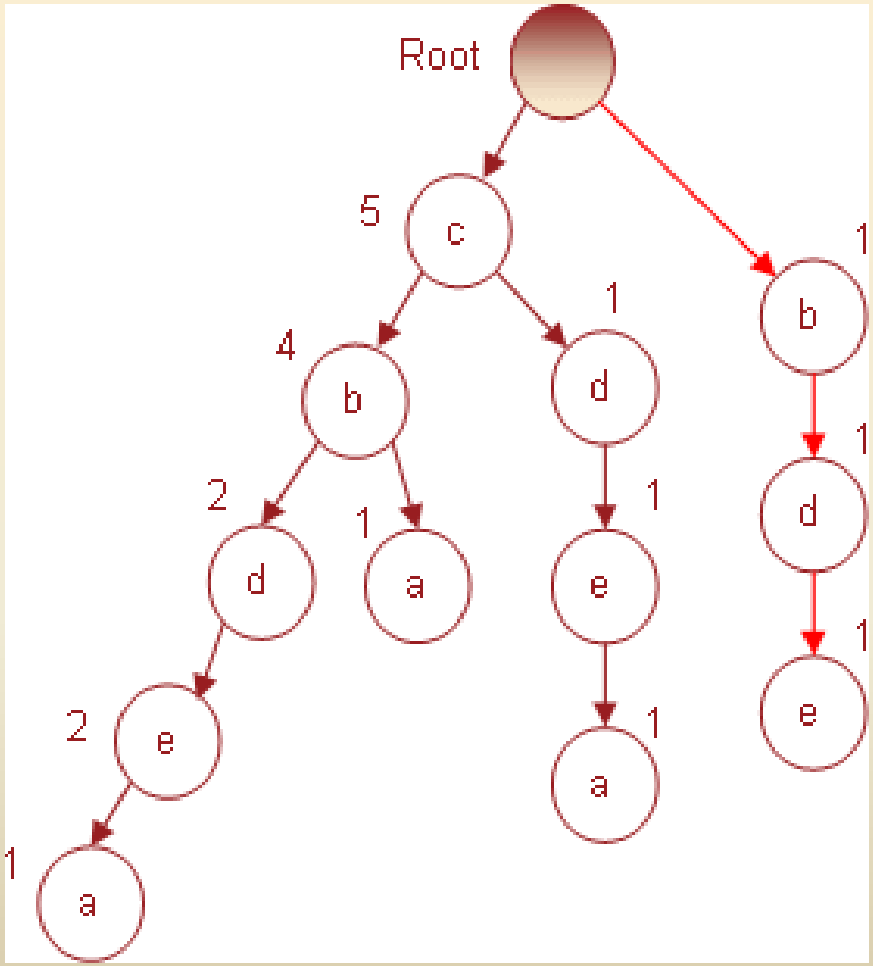
#5: c b



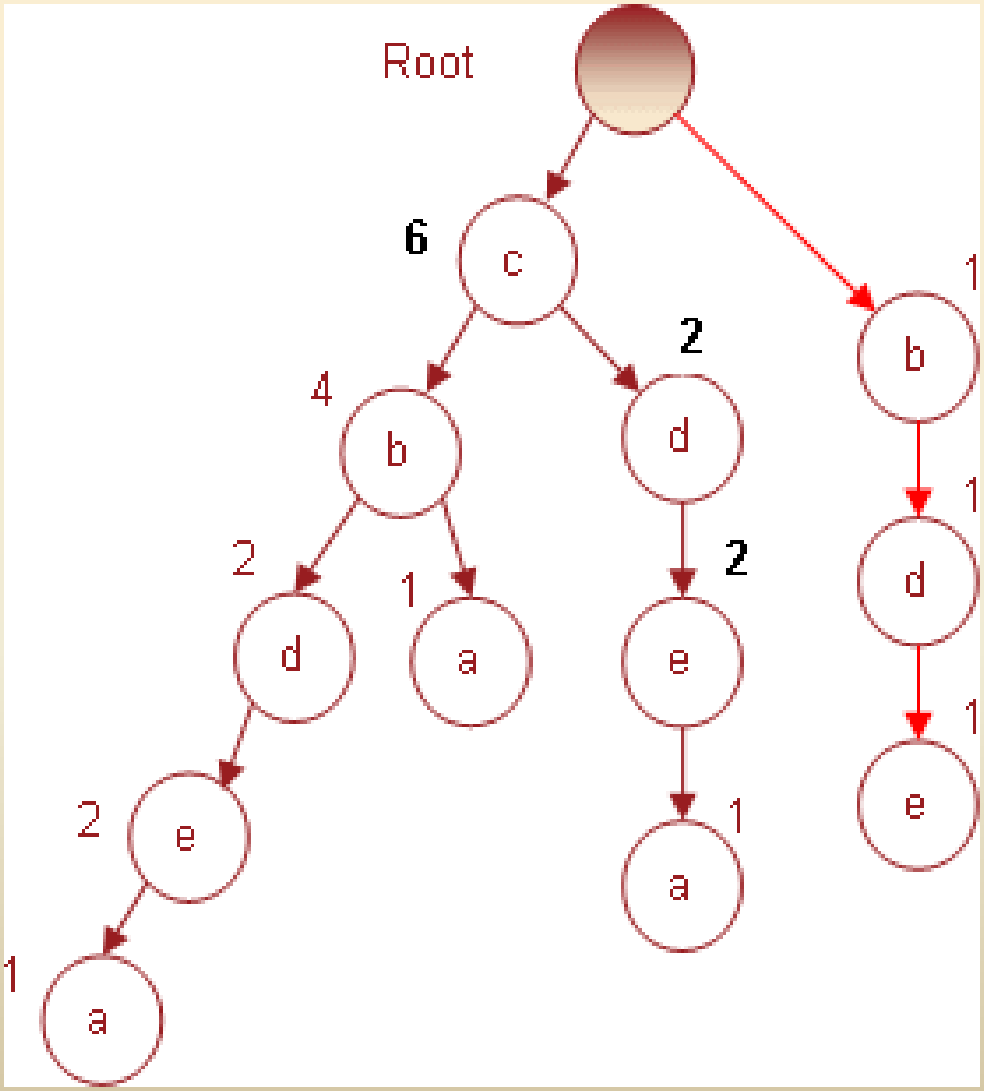
#6: b d e



#7: c d e



Итоговое FPG-дерево



FPG: Извлечение частых наборов из дерева

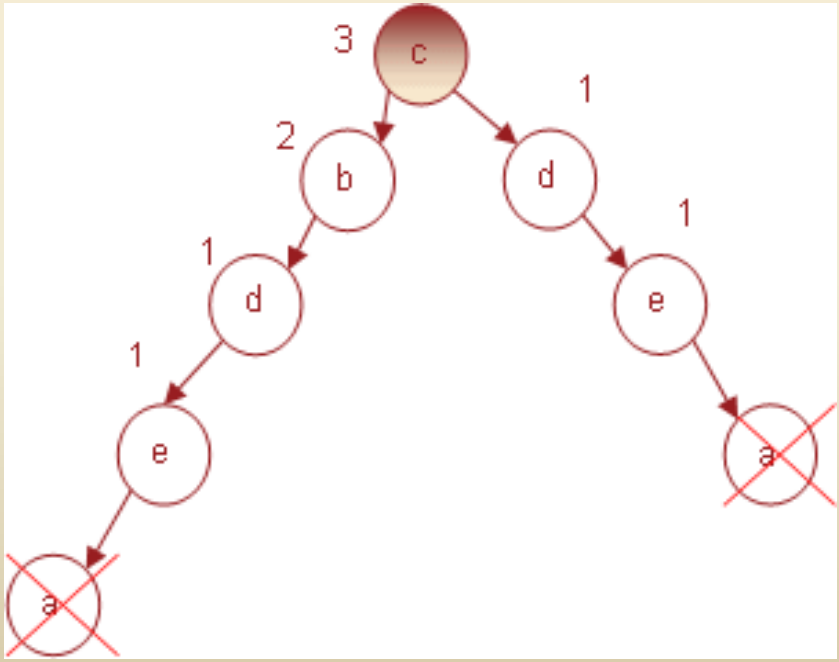
- Для каждого характерного признака
- Находим пути в дереве до узлов, связанных с выбранным признаком
- Строим условное дерево, ветки которого завершаются выбранным признаком
- Веса конечных узлов переносятся на верхние узлы; при слиянии путей веса складываются
- Подсчитываем число вхождений дополняющих признаков в условное дерево (встречаемость в отобранных путях)
- Строим наборы, содержащие дополняющие признаки с числом вхождений не меньшими, чем установленный порог

Признак: a

Пути: (**c b d e a**), **(c b a**), (**c d e a**)

Частоты дополняющих признаков: (c, 3), (b,2), (d, 2), (e,2)

Допустимые характерные наборы: (a c, 3)

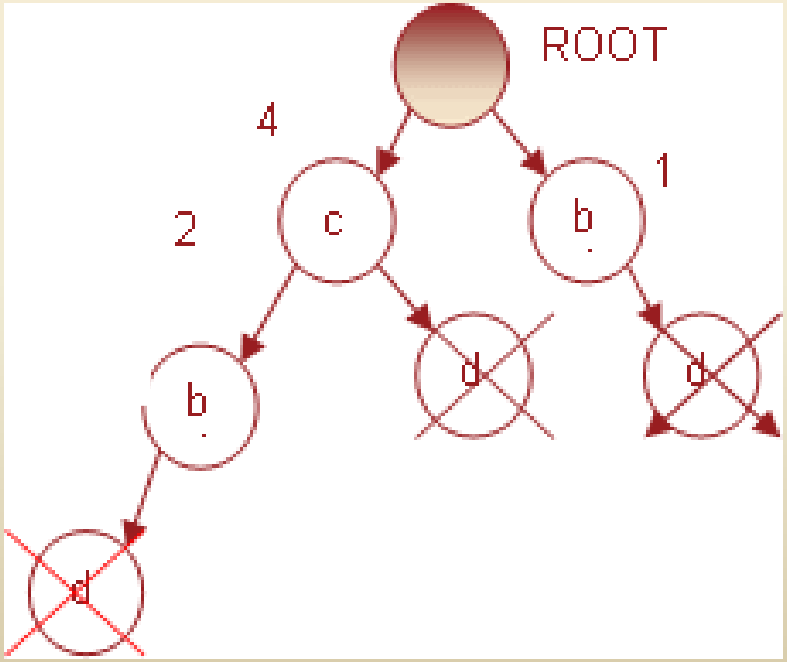


Признак: d

Пути: (**c b d**), **(c d**), (**b d**)

Частоты дополняющих признаков: (c, 4), (b,2),

Допустимые характерные наборы: (d c, 4)

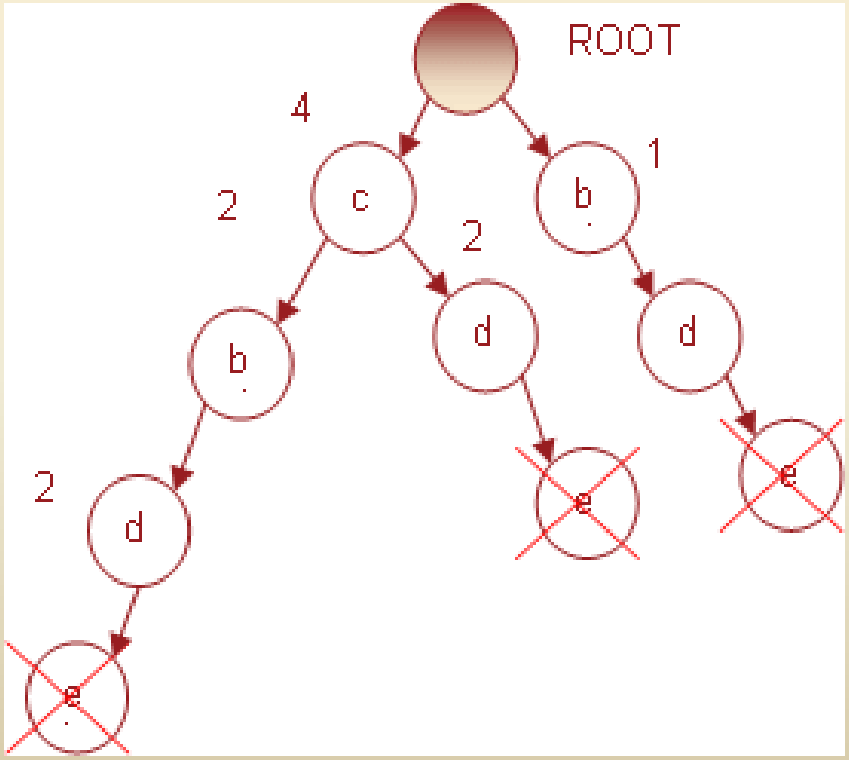


Признак: e

Пути: (**c b d e**, 2) (**c d e**, 2) (**b d e**, 1)

Частоты дополняющих признаков: (**d**, 5), (**c**, 4), (**b**, 3)

Характерные наборы: (**d**, **e**, 5), (**d**, **c**, **e**, 4), (**d**, **b**, **e**, 3)



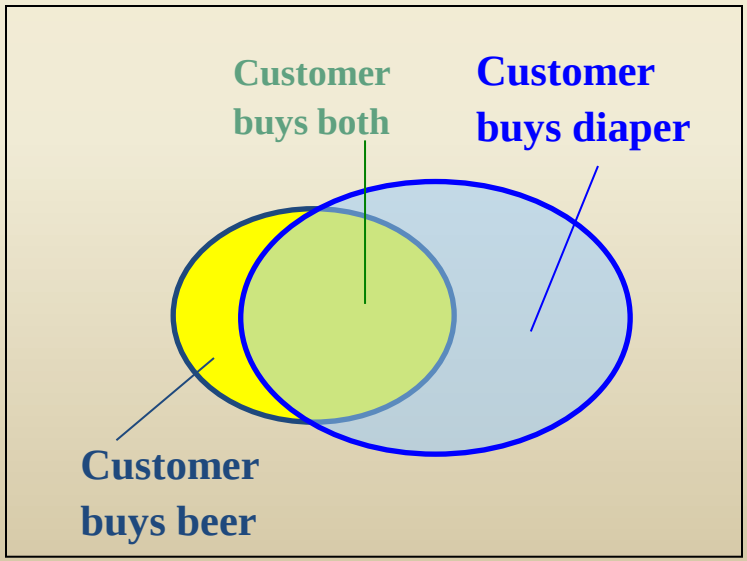
Ассоциативные правила

- Правила вида

  IF **Условие** THEN **Вывод**

  IF Beer THEN Pampers

- Правила формируются на базе найденных характерных комбинаций, используя дополнительные оценки:
  - Оценка *Confidence(A->B) = Supp(A,B) / Supp(A)*



Conf(«diaper»-> «beer») = 45%

Conf(«beer» -> «diaper») = 85%

Мера «интересности» правила: Lift

- *play basketball* $\Rightarrow$ *eat cereal* [40%, 66.7%] - плохое правило
  - Общий % студентов потребляющих хлопья 75% > 66.7%.
- *play basketball* $\Rightarrow$ *not eat cereal* [20%, 33.3%] более продуктивное правило, хотя и с меньшими Support и Confidence
- Мера зависимости/независимости признаков: lift

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

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

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

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

Другие меры «интересности» набора

symbol	measure	range	formula
ϕ	ϕ-coefficient	-1...1	$\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
Q	Yule's Q	-1 ...1	$\frac{P(A,B)P(\overline{A},\overline{B})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{A},\overline{B})+P(A,\overline{B})P(\overline{A},B)}$
Y	Yule's Y	-1 ...1	$\frac{\sqrt{P(A,B)P(\overline{A},\overline{B})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{A},\overline{B})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}}$
k	Cohen's	-1 ...1	$\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$
PS	Piatetsky-Shapiro's	-0.25 ...0.25	$P(A,B)-P(A)P(B)$
F	Certainty factor	-1 ...1	$\max(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)})$
AV	added value	-0.5 ...1	$\max(P(B\|A)-P(B), P(A\|B)-P(A))$
K	Klosgen's Q	-0.33 ...0.38	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$
g	Goodman-kruskal's	0 ...1	$\frac{\Sigma_j \max_k P(A_j,B_k)+\Sigma_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-max_k P(B_k)}$
M	Mutual Information	0 ...1	$\frac{\Sigma_i\Sigma_j P(A_i,B_j)\log\frac{P(A_i,B_j)}{P(A_i)P(B_j)}}{\min(-\Sigma_i P(A_i)\log P(A_i)\log P(A_i), -\Sigma_i P(B_i)\log P(B_i)\log P(B_i))}$
J	J-Measure	0 ...1	$\max(P(A,B)\log(\frac{P(B\|A)}{P(B)})+P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}))$ $P(A,B)\log(\frac{P(A\|B)}{P(A)})+P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})$
G	Gini index	0 ...1	$\max(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A}[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B}[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]-P(A)^2-P(\overline{A})^2)$
s	support	0 ...1	$P(A,B)$
c	confidence	0 ...1	$max(P(B\|A), P(A\|B))$
L	Laplace	0 ...1	$\max(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2})$
IS	Cosine	0 ...1	$\frac{P(A,B)}{\sqrt{P(A)P(B)}}$
γ	coherence(Jaccard)	0 ...1	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
α	all_confidence	0 ...1	$\frac{P(A,B)}{\max(P(A),P(B))}$
o	odds ratio	0 ...∞	$\frac{P(A,B)P(\overline{A},\overline{B})}{P(\overline{A},B)P(A,\overline{B})}$
V	Conviction	0.5 ...∞	$\max(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})})$
λ	lift	0 ...∞	$\frac{P(A,B)}{P(A)P(B)}$
S	Collective strength	0 ...∞	$\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$
χ^2	χ^2	0 ...∞	$\Sigma_i \frac{(P(A_i)-E_i)^2}{E_i}$

Tan, Kumar, Sritastava @KDD'02)