Adatbányászat: Társítási szabályok Alapfogalmak és algoritmusok

6. fejezet

Tan, Steinbach, Kumar

Bevezetés az adatbányászatba

előadás-fóliák

fordította

Ispány Márton

Hogyan kezdődött...

- Rakesh Agrawal, Tomasz Imielinski, Arun N. Swami: Mining Association Rules between Sets of Items in Large Databases. SIGMOD Conference 1993: 207-216
- Rakesh Agrawal, Ramakrishnan Srikant: Fast Algorithms for Mining Association Rules in Large Databases. VLDB 1994: 487-499

- Ezt a két cikket tekintik az adatbányászat születésének.
- Már jó ideje foglalkoznak Asszociációs szabályok és Gyakori tételcsoportok bányászatával
  - Néhányan (iparban és egyetemeken) még mindig.

Vásárlói kosár adatok

- Termékek egy nagy halmaza, pl. egy szupermarket kínálata.
- Kosarak egy nagy halmaza amelyek mindegyike termékek egy kis halmaza, pl. azok a termékek, melyeket egy vásárló egy vásárlás alkalmával vesz (a kosarába tesz).

- Valójában egy általános leképezés (hozzárendelés) kétféle dolog között, ahol az egyik (kosarak) a másik (termékek) egy részhalmaza.
  - Azonban mi a termékek és nem a kosarak közötti kapcsolatokra vagyunk kíváncsiak.
- A módszer a gyakori eseményekre és nem a ritkákra fókuszál (“hosszú farok”).

Gyakori tételcsoport fogalma

TID	*Termékek*
1	Kenyér, Tej
2	Kenyér, Pelenka, Sör, Tojás
3	Tej, Pelenka, Sör, Kóla
4	Kenyér, Tej, Pelenka, Sör
5	Kenyér, Tej, Pelenka, Kóla

- **Tételcsoport**
  - Egy vagy több tétel összessége.
    - Példa: {Tej, Kenyér, Pelenka}
  - *k*-tételcsoport
    - k számú tételt tartalmazó tételcsoport
- **Támogatottsági érték (σ)**
  - Egy tételcsoport előfordulási gyakorisága.
  - Pl. σ({Tej, Kenyér, Pelenka}) = 2
- **Támogatottság**
  - Egy tételcsoportot tartalmazó tranzakciók aránya.
  - Példa: s({Tej, Kenyér, Pelenka}) = 2/5
- **Gyakori tételcsoport**
  - Egy olyan tételcsoport, amely támogatottsága nagyobb vagy egyenlő egy *minsup* küszöb értéknél.

Alkalmazások

- Tételek = termékek; kosarak = termékek összessége melyet egy vásárló a kosarába tesz a boltban.
  - Példa: adott, hogy sok vásárló vesz együtt sört és pelenkát:

    akciózzuk a pelenkákat és emeljük a sör árát.
  - Csak akkor hasznos ha sokan vesznek együtt pelenkát és sört.
- Kosarak = Web lapok; tételek = szavak.
  - Példa: Szokatlan szavak melyek együtt fordulnak elő nagy számú dokumentumban, pl. “Brad” és “Angelina,” egy érdekes kapcsolatot jelölhet.
- Kosarak = mondatok; tételek = dokumentumok melyek ezeket a mondatokat tartalmazzák.
  - Példa: Azok a tételek melyet túl gyakran fordulnak elő együtt plágiumot jelenthetnek.
  - Vegyük észre, hogy a tételeknek nem kell benne lenniük a kosarakban.

Gyakori tételcsoportok bányászata

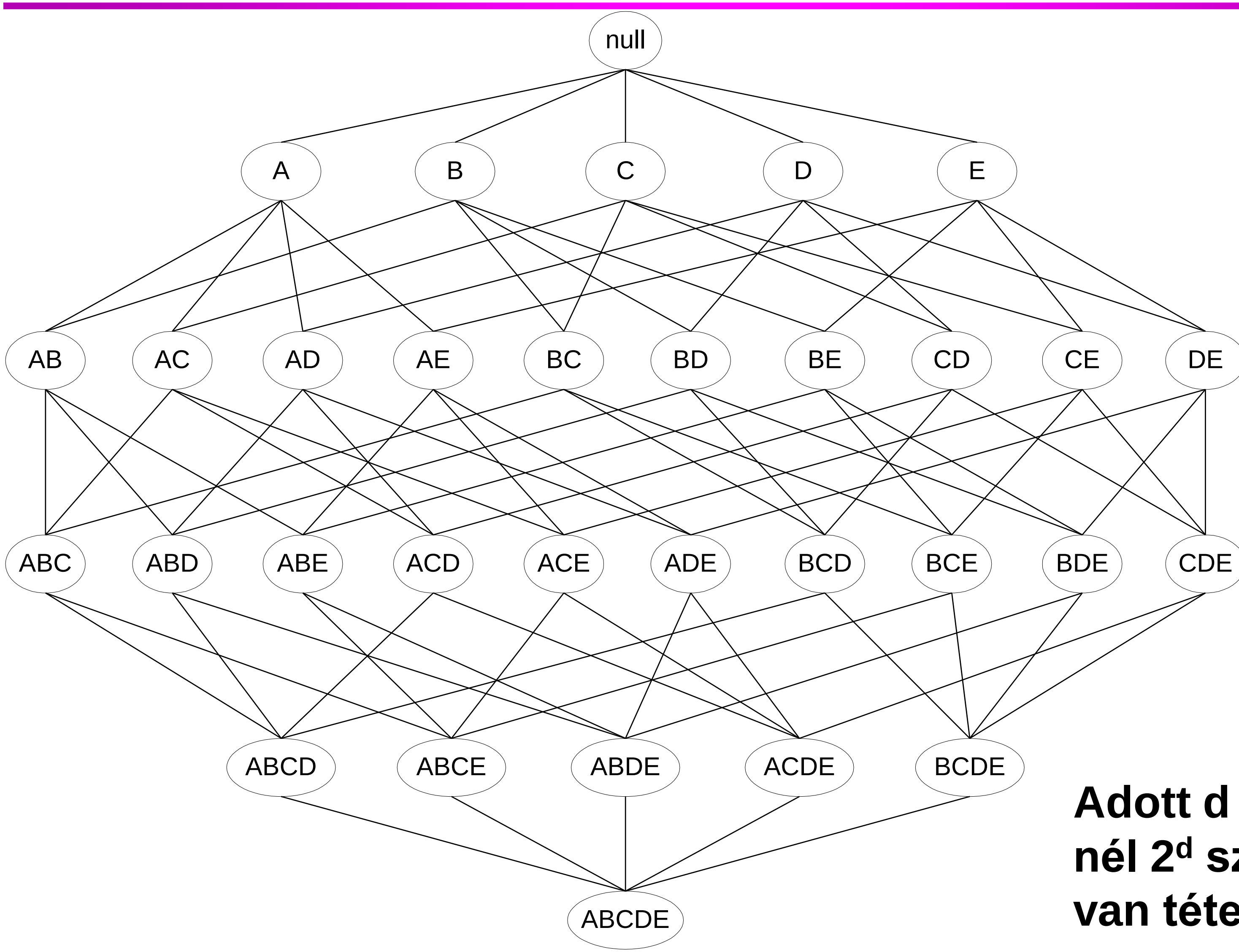
- Input: A tranzakciók T halmaza tételek egy I halmaza felett
- Output: Tételeknek az összes olyan halmaza I-ben melyre
  - support ≥ *minsup* threshold

- Feladat paraméterei:
  - N = |T|: tranzakciók száma
  - d = |I|: (különböző) tételek száma
  - w: a tranzakciók maximális szélessége
  - A lehetséges tételcsoportok száma?

$$M = 2^d$$

- Feladat mérete:
  - WalMart 100 000 tételt árusít és kosarak millióit tartja nyilván.
  - A Web szavak milliárdjait és sok milliárd lapot tartalmaz.

Gyakori tételcsoportok előállítása

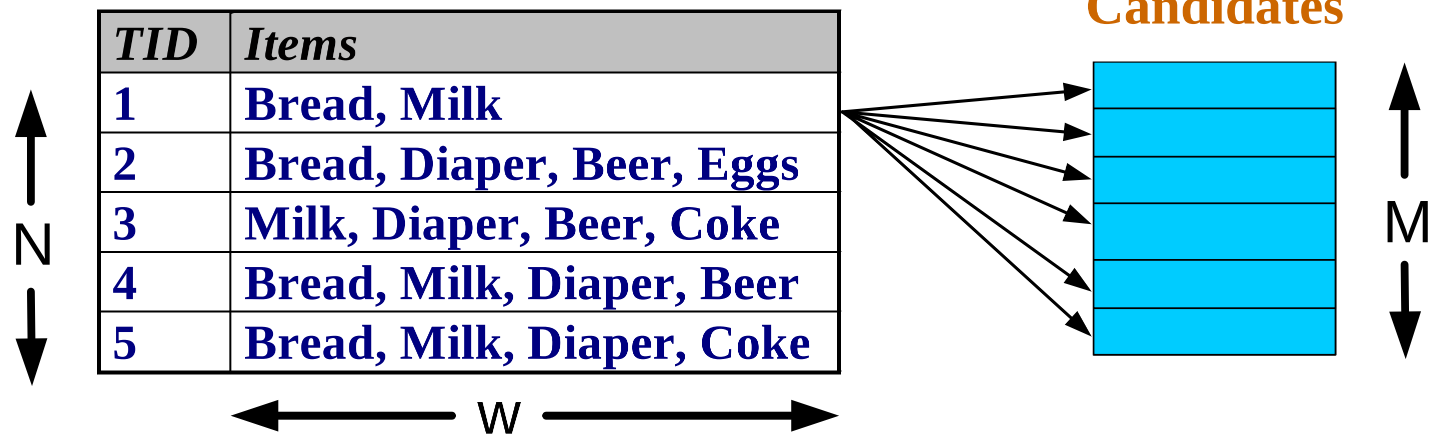


**Adott d számú tétel-
nél 2^d számú jelölt
van tételcsoportra**

Gyakori tételcsoportok előállítása

- Nyers erő megközelítés:
  - Minden csúcs a gráfban egy jelölt gyakori tételcsoportra.
  - Számítsuk ki minden jelölt támogatottságát az adatbázis átfésülésével.

Transactions **List of Candidates**

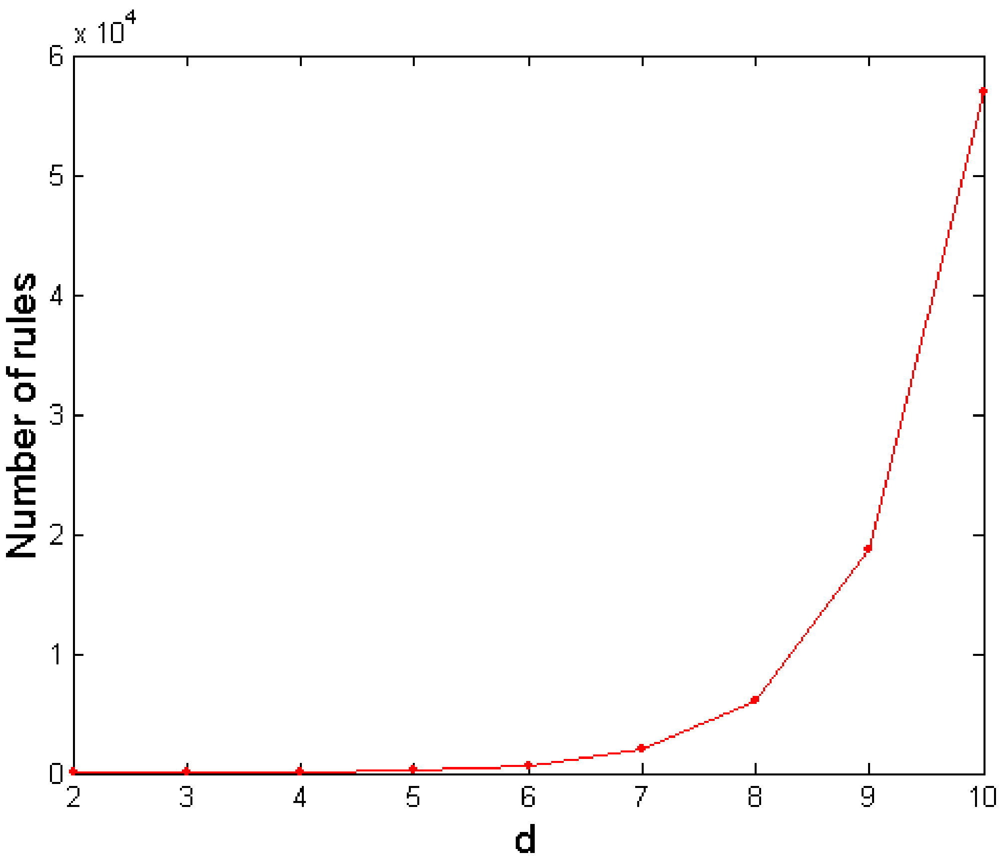


TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

  - Vessünk össze minden tranzakciót minden jelölttel!
  - Komplexitás ~ O(NMw) => Költséges mivel $M = 2^d$!!!

Kiszámítási komplexitás

- Adott *d* számú tétel esetén:
  - Az összes tételcsoport száma = 2^d
  - Az összes lehetséges társítási szabály száma:



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

$$= 3^d - 2^{d+1} + 1$$

Ha d=6 akkor R = 602 szabály

A számítási modell

- Általában az adatokat egy flat fájlban tároljuk és nem egy adatbázisban.
  - Ezt a fájl a lemezen tároljuk.
  - A tárolás kosaranként történik.
  - A kosarakat párokká, hármasokká stb. bontjuk ki ahogy beolvassuk őket.
    - A kibontás során k egymásba ágyazott ciklust használunk hogy az összes k elemű halmazt előállítsuk.

Példa fájl: kiskereskedelem

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
30 31 32
33 34 35
36 37 38 39 40 41 42 43 44 45 46
38 39 47 48
38 39 48 49 50 51 52 53 54 55 56 57 58
32 41 59 60 61 62
3 39 48
63 64 65 66 67 68
32 69
48 70 71 72
39 73 74 75 76 77 78 79
36 38 39 41 48 79 80 81
82 83 84
41 85 86 87 88
39 48 89 90 91 92 93 94 95 96 97 98 99 100 101
36 38 39 48 89
39 41 102 103 104 105 106 107 108
38 39 41 109 110
39 111 112 113 114 115 116 117 118
119 120 121 122 123 124 125 126 127 128 129 130 131 132 133
48 134 135 136
39 48 137 138 139 140 141 142 143 144 145 146 147 148 149
39 150 151 152
38 39 56 153 154 155
```

Példa: tételek pozitív egészek, és mindegyik kosár a fájl egy sorának felel meg az egészeket egy space-szel elválasztva

Gyakori tételcsoportok előállítása

- Csökkentsük a jelöltek számát (*M*)
  - Teljes keresés: $M=2^d$
  - Használjunk vágási módszereket *M* csökkentésére.

- Csökkentsük a tranzakciók számát (*N*)
  - Csökkentsük N-et a tételcsoportok számának növekedésével.
  - Használjunk DHP (direct hashing and pruning – közvetlen hasító és vágó) illetve vertikálisan bányászó algoritmusokat.

- Csökkentsük az összehasonlítások számát (NM)
  - Használjunk hatékony adatszerkezeteket a jelöltek és a tranzakciók tárolására.
  - Nem szükséges minden jelöltet és tranzakciót összehasonlítani.

A jelöltek számának csökkentése

- Apriori elv:
  - Ha egy tételcsoport gyakori, akkor minden részhalmaza is gyakori.

- Az apriori elv a támogatottság következő tulajdonságán alapszik:

$$\forall X, Y : (X \subseteq Y) \Rightarrow s(X) \geq s(Y)$$

  - Egy tételcsoport támogatottsága sohasem haladhatja meg részhalmazainak támogatottságát.
  - Ez a támogatottság ún. anti-monoton tulajdonsága.

Az Apriori algoritmus

szintenkénti megközelítés

C_k = k méretű tételcsoport jelöltek
L_k = k méretű gyakori tételcsoportok

1. $k = 1$, C_1 = összes tétel
2. While C_k nem üres

Gyakori tételcsoport generálás

3. Adatbázis átfésülésével találjuk meg C_k-ban a gyakori tételcsoportokat és tegyük L_k-ba

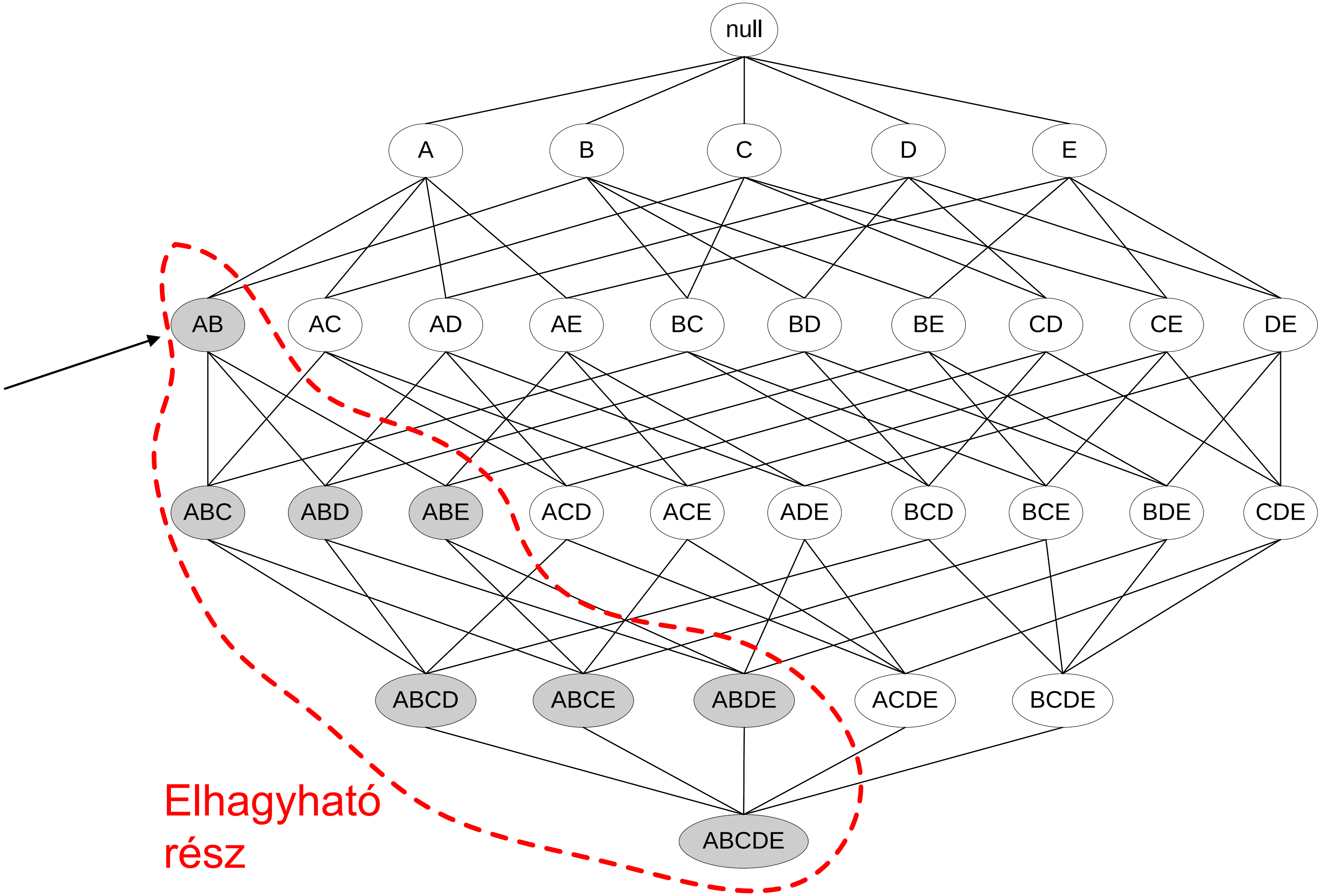
Jelölt generálás

4. Használjuk L_k-t a tételcsoport jelöltek $k+1$ méretű C_{k+1} halmazának előállítására
5. $k = k+1$

R. Agrawal, R. Srikant: "Fast Algorithms for Mining Association Rules", *Proc. of the 20th Int'l Conference on Very Large Databases*, 1994.

Az Apriori elv szemléltetése

Nem gyakori



Elhagyható rész

Az Apriori elv szemléltetése

Tétel	Darab
Kenyér	**4**
Kóla	**2**
Tej	**4**
Sör	**3**
Pelenka	**4**
Tojás	**1**

Tételek (1-tétel csoportok)

Tétel csoport	Darab
{Kenyér,Tej}	**3**
{Kenyér,Sör}	**2**
{Kenyér, Pelenka}	**3**
{Tej,Sör}	**2**
{Tej,Pelenka}	**3**
{Sör,Pelenka}	**3**

Párok (2-tétel csoportok)

(Nincs szükség olyan jelöltek előállítására, melyek a kólát és a tojást tartalmazzák.)

Minimális támogatottság = 3

Hármasok (3-tétel csop.)

Tételcsoport	Darab
{Kenyér,Tej,Pelenka}	**3**

Ha minden részhalmazt figyelembe veszünk:

$${}^6C_1 + {}^6C_2 + {}^6C_3 = 41$$

Támogatottság alapú eltávolításnál:

$$6 + 6 + 1 = 13$$

Jelölt generálás

- Alapelv (Apriori):
  - Egy (k+1)-tételcsoport csak akkor lehet gyakori jelölt ha az összes k méretű részhalmaza (biztosan) gyakori.
- Alapötlet:
  - Konstruáljunk egy k+1 méretű jelöltet k méretű gyakori tételcsoportok összekombinálásával.
    - Ha $k = 1$ akkor vegyük a gyakori tételcsoportok összes párját
    - Ha $k > 1$ akkor egyesítsük tételcsoportok olyan párjait melyek csak egy tételben különböznek
    - Minden egyes generált tételcsoport jelöltnél győződjünk meg, hogy az összes k elemű részhalmaza gyakori-e.

A C_{k+1}-beli jelöltek generálása

- **Feltevés: a tételcsoportokban a tételek rendezettek**
  - **Pl., ha az egészek nagyság szerint növekvő sorba rendezettek, ha a rekordok lexiko-grafikusan rendezettek**
    - **A rendezettség biztosítja, hogy ha y > x esetén y előfordul x előtt, akkor x nincs benne a tételcsoportban**
- **Az L_k-beli tételek szintén rendezettek**

Hozzunk létre egy (k+1)-tételcsoportot két olyan k-tételcsoport egyesítésével amelyeknek az első k-1 tétele ugyanaz.

Tétel1	Tétel2	Tétel3
1	2	3
1	2	5
1	4	5

Kihagytunk valamit?
Mi a helyzet az alábbi jelölttel?

1	2	4	5

C_{k+1}–beli jelöltek generálása SQL-ben

- ***self-join*** L_k

  insert into C_{k+1}

  select $p.item_1, p.item_2, \ldots, p.item_k, q.item_k$

  from L_k p, L_k q

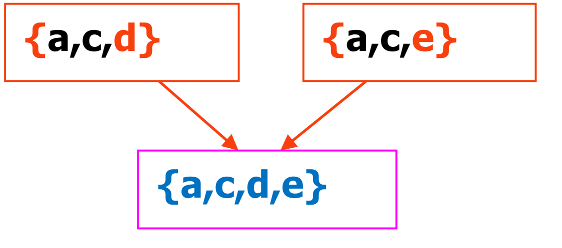
  where $p.item_1=q.item_1, \ldots, p.item_{k-1}=q.item_{k-1}, p.item_k < q.item_k$

Példa

- L_3={abc, abd, acd, ace, bcd}
- ***Self-joining***: $L_3 * L_3$
  - **abcd: abc és abd tételcsoportokból**
  - **acde: acd és ace tételcsoportokból**

Tétel1	Tétel2	Tétel3
a	b	c
a	b	d
a	c	d
a	c	e
b	c	d

Tétel1	Tétel2	Tétel3
a	b	c
a	b	d
a	c	d
a	c	e
b	c	d



$$p.item_1 = q.item_1, p.item_2 = q.item_2, p.item_3 < q.item_3$$

A C_{k+1}-beli jelöltek generálása

- Készen vagyunk? Valóban jó minden jelölt?

Item 1	Item 2	Item 3
1	2	3
1	2	5
1	4	5

1	2	3	5

Jó ez a jelölt?

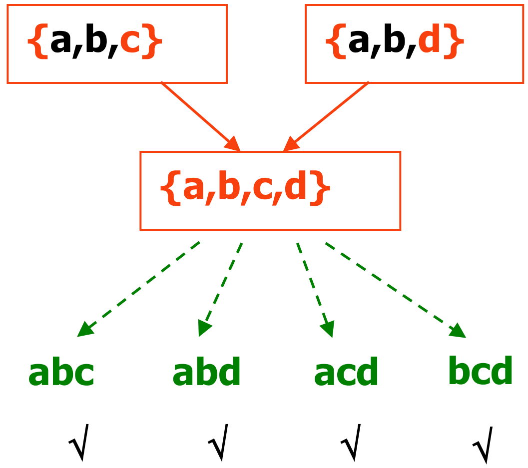
Nem. Az (1,3,5) és (2,3,5) részhalmazoknak is gyakoriaknak kell lenniük.

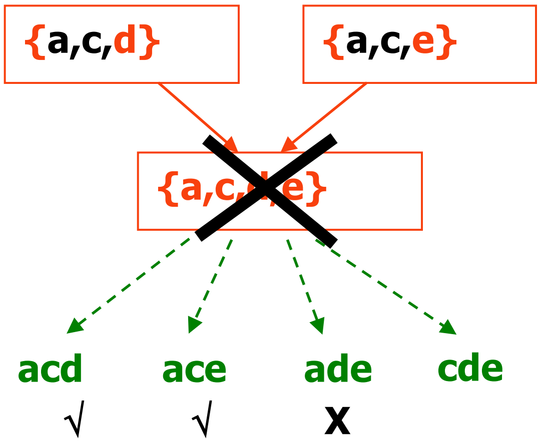
Apriori elv!

- Tisztítási lépés:
  – Minden egyes (k+1)-tételcsoport jelöltnél állítsuk elő az összes k-rész-tételcsoportját
  – Töröljük azt a jelöltet, amely részhalmazként egy olyan k-tételcsoportot tartalmaz amely nem gyakori

Példa

- L_3={abc, abd, acd, ace, bcd}
- *Self-joining*: $L_3 * L_3$
  - *abcd*: *abc* **és** *abd* ***tételcsoportokból***
  - *acde*: *acd* **és** *ace* ***tételcsoportokból***
- ***Tisztítás:***
  - *abcd* **–t megtartjuk mivel az összes részhalmaza tételcsoport L_3 -ban**
  - *acde* **töröljük mivel** *ade* **nem gyakori L_3 -ban**
- C_4={abcd}





A C_{k+1}-beli jelöltek generálása

- Adott az összes gyakori k-tételcsoport L_k halmaza
- 1. lépés: self-join L_k
  - Hozzuk létre a C_{k+1} halmazt azon gyakori k-tételcsoport párok egyesítésével, amelyeknek az első k-1 tétele közös
- 2. lépés: tisztítás
  - Töröljük C_{k+1} –ból azokat a tételcsoportokat, amelyek olyan rész k-tételcsoportot tartalmaznak, amelyek nem gyakoriak

Az összehasonlítások csökkentése

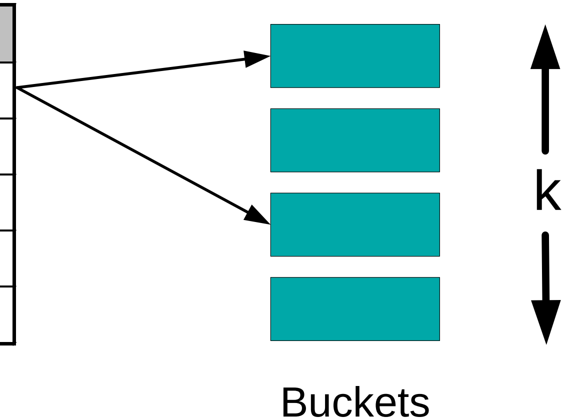
- Jelöltek leszámlálása:
  - A tranzakciós adatbázis átfésülésével határozzuk meg minden tételcsoport jelölt támogatottságát.
  - Az összehasonlítások számának csökkentése érdekében a jelölteket tároljuk hash szerkezetben.
    - Ahelyett, hogy minden tranzakciót minden jelölttel összehasonlítunk, használjunk hasított kupacokat a jelöltekre.

Transactions

N

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Hash Structure



A hasító fa előállítása

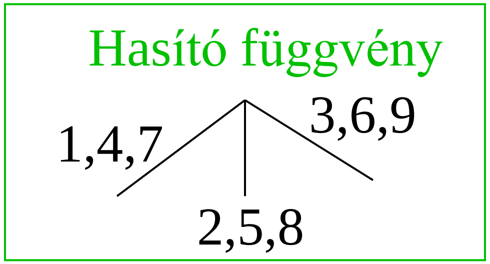
Legyen adott 15 tételcsoport jelöltünk, melyek hossza 3:

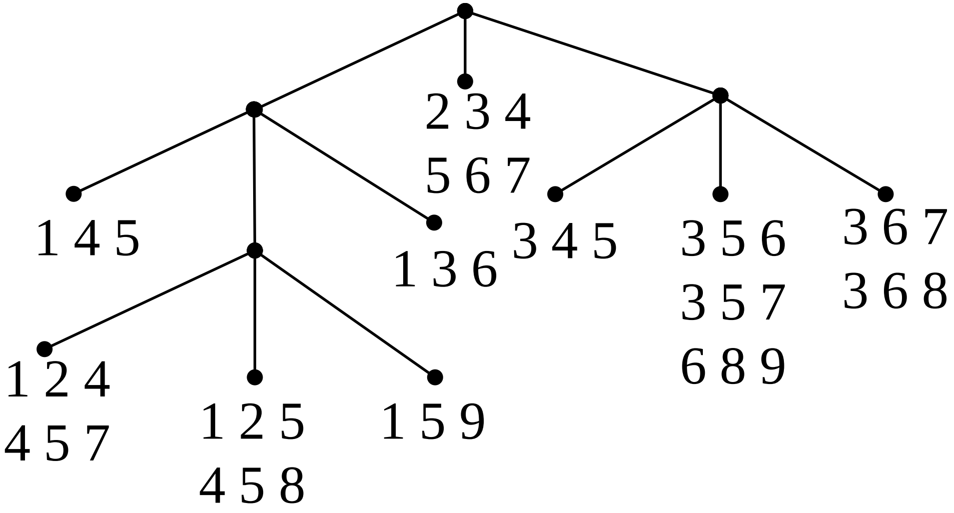
{1 4 5}, {1 2 4}, {4 5 7}, {1 2 5}, {4 5 8}, {1 5 9}, {1 3 6}, {2 3 4}, {5 6 7}, {3 4 5}, {3 5 6}, {3 5 7}, {6 8 9}, {3 6 7}, {3 6 8}

Szükségünk van:

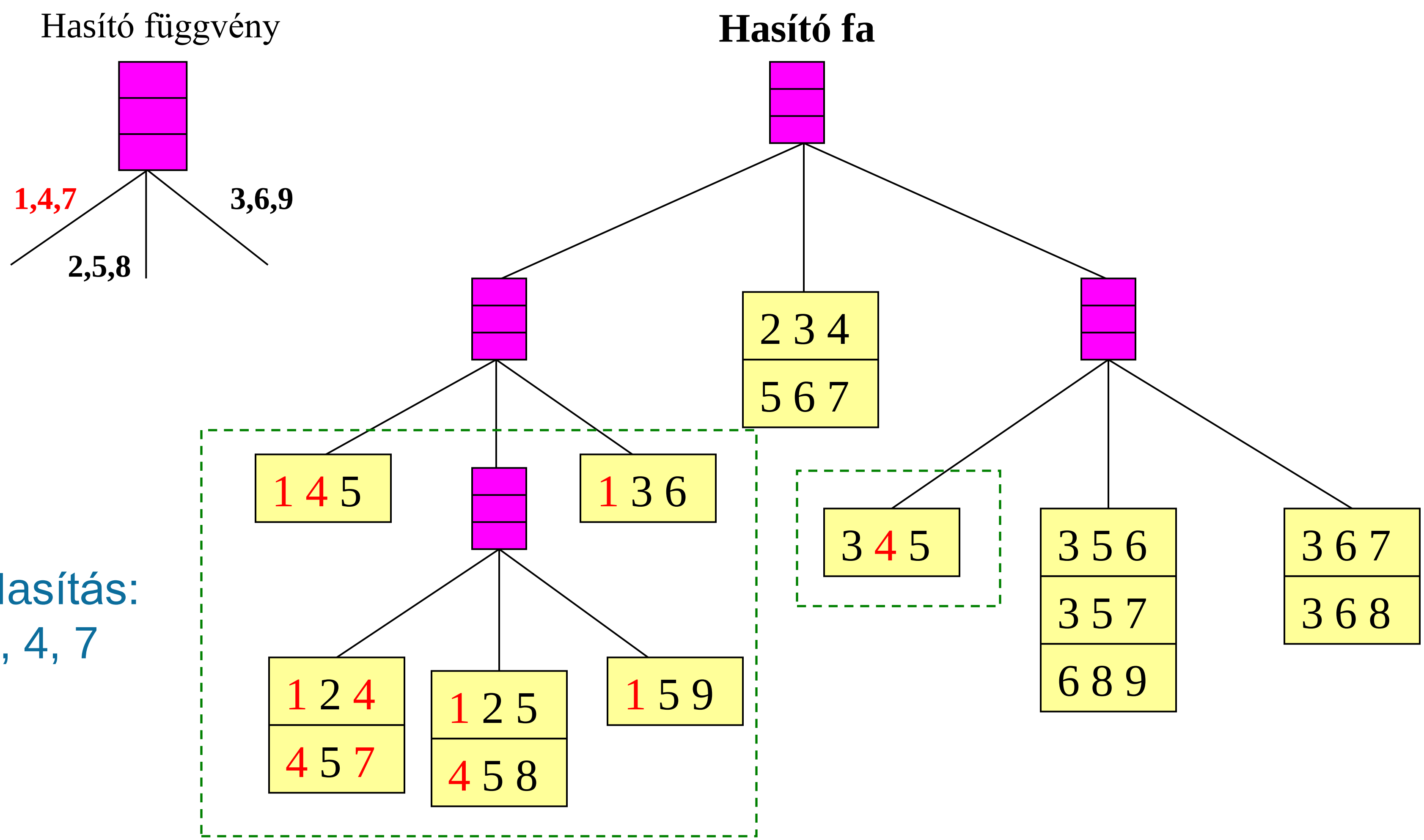
- **Hasító függvényre**
- **Maximális levélnagyságra: egy levélben tárolt maximális jelölt-számra**

  (ha a jelöltek száma ezt túl lépi, akkor vágjuk ketté a csomópontot)



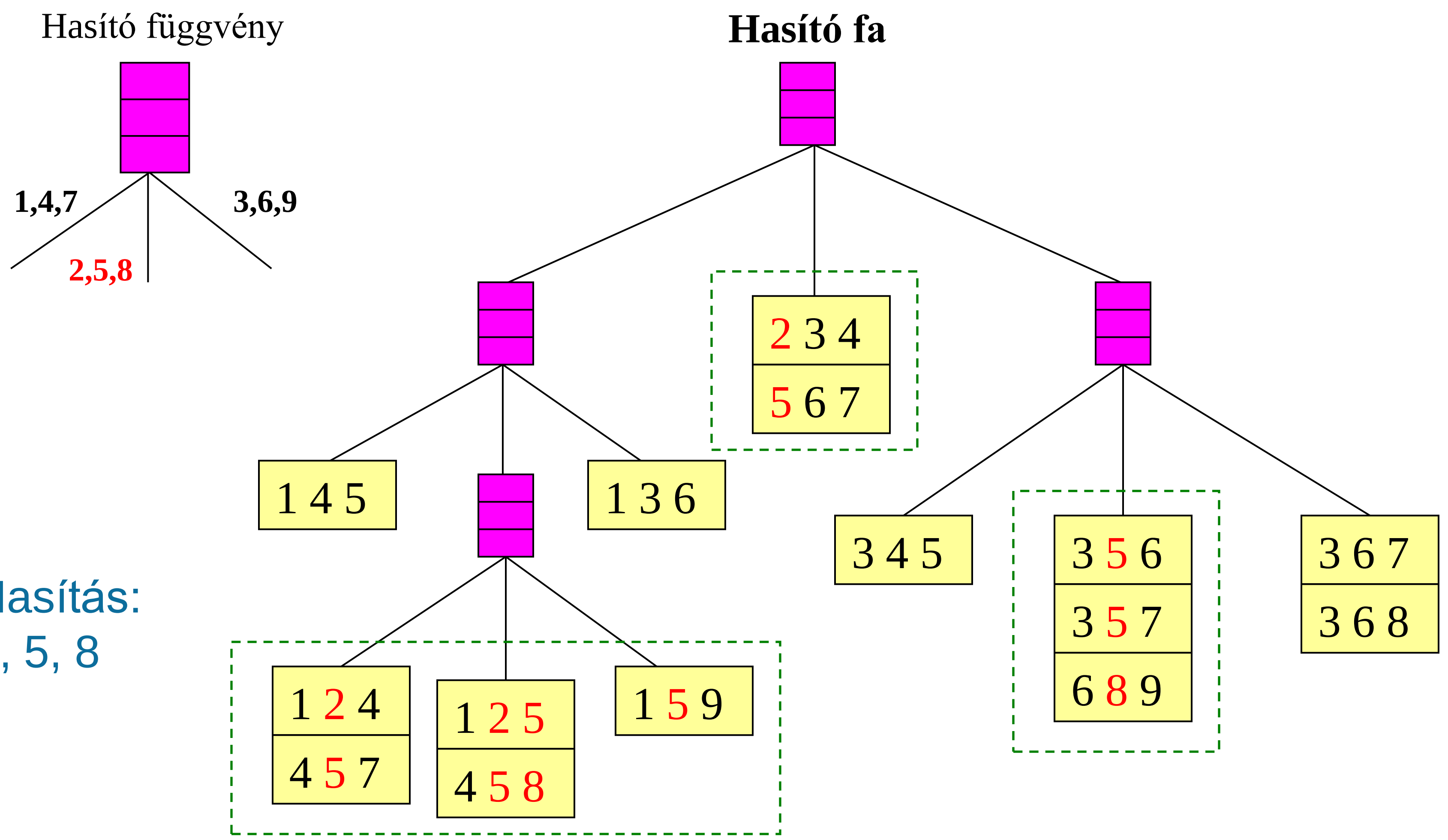


Hasító fa



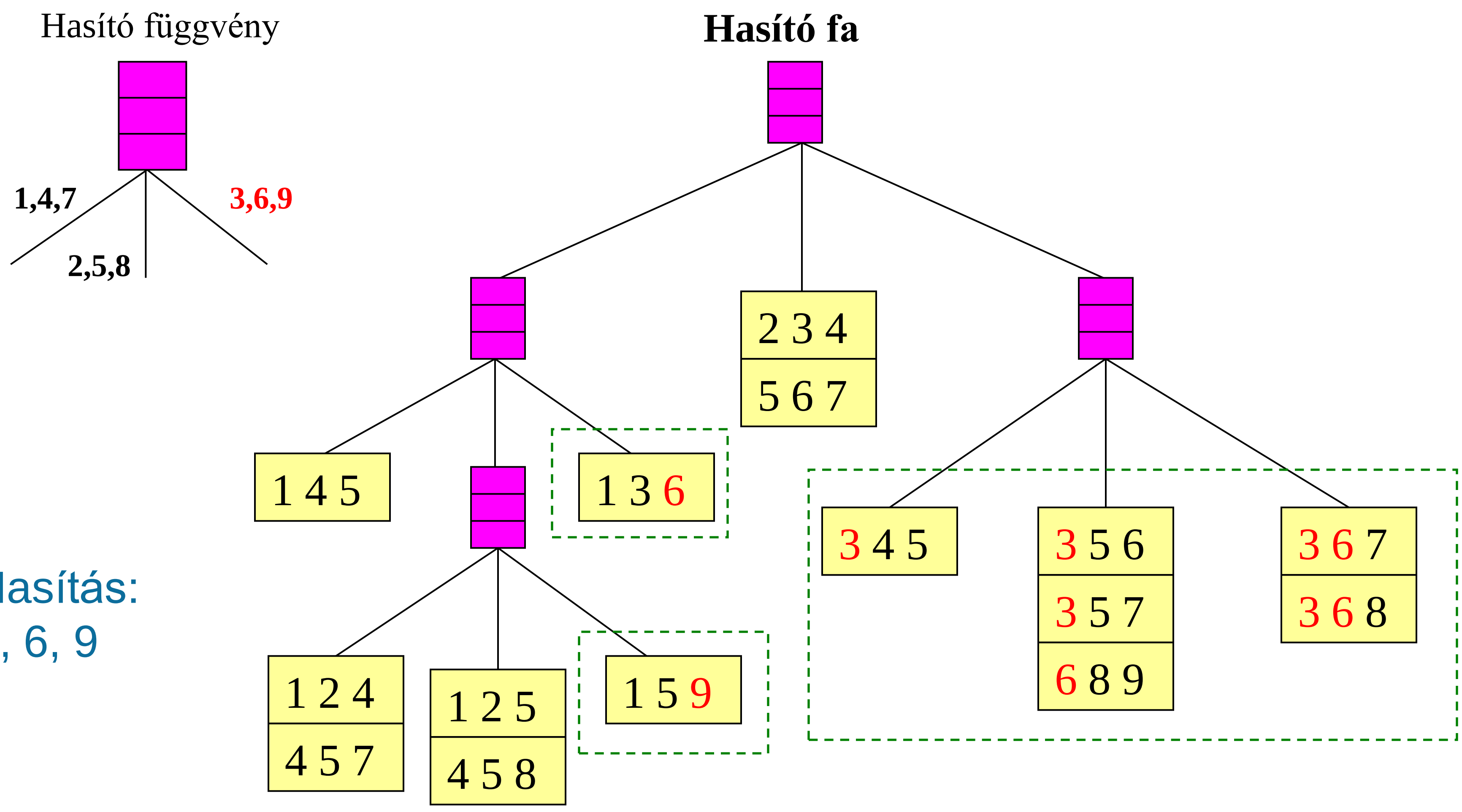
Hasítás:
1, 4, 7

Hasító fa



Hasítás:
2, 5, 8

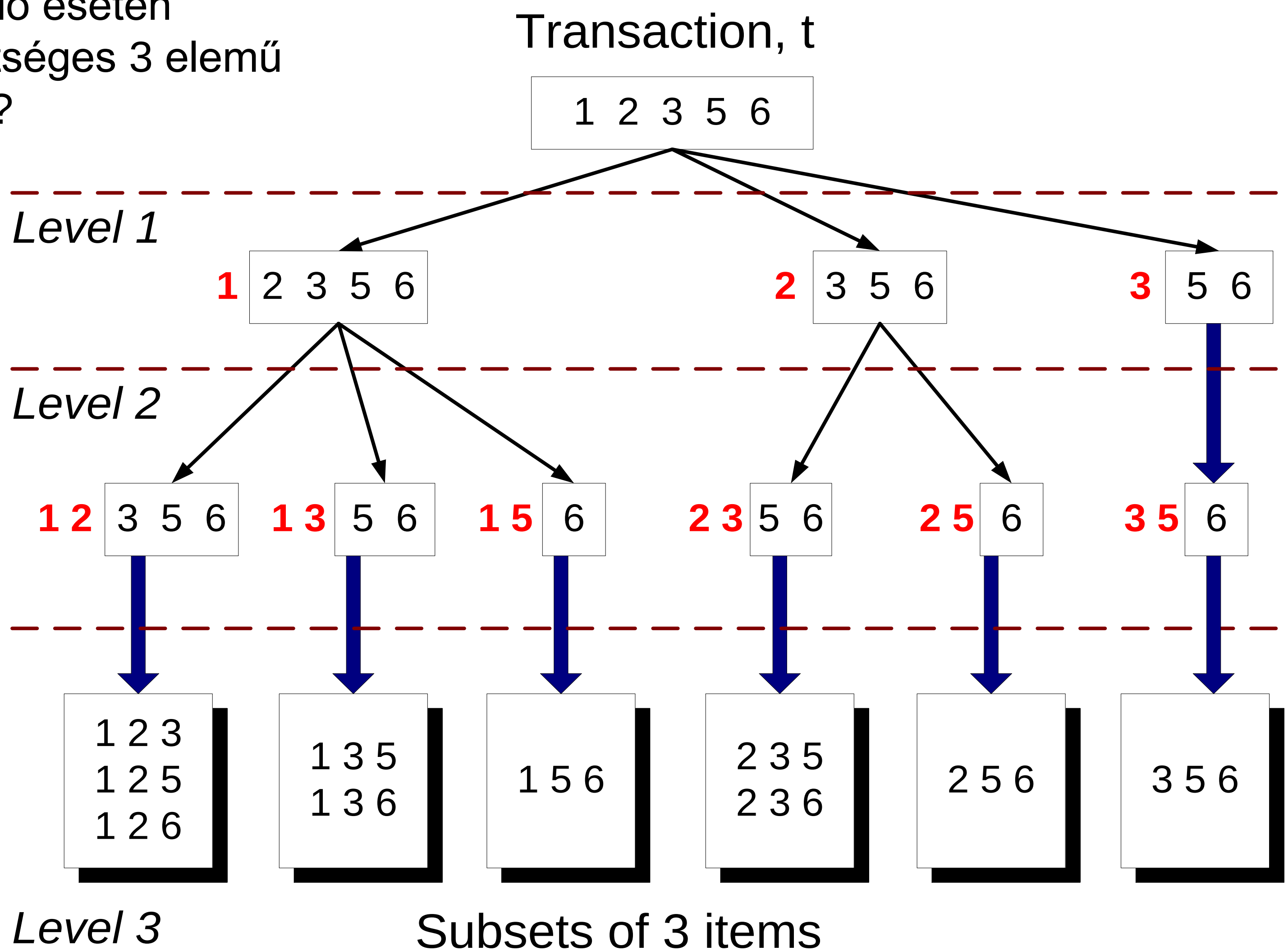
Hasító fa



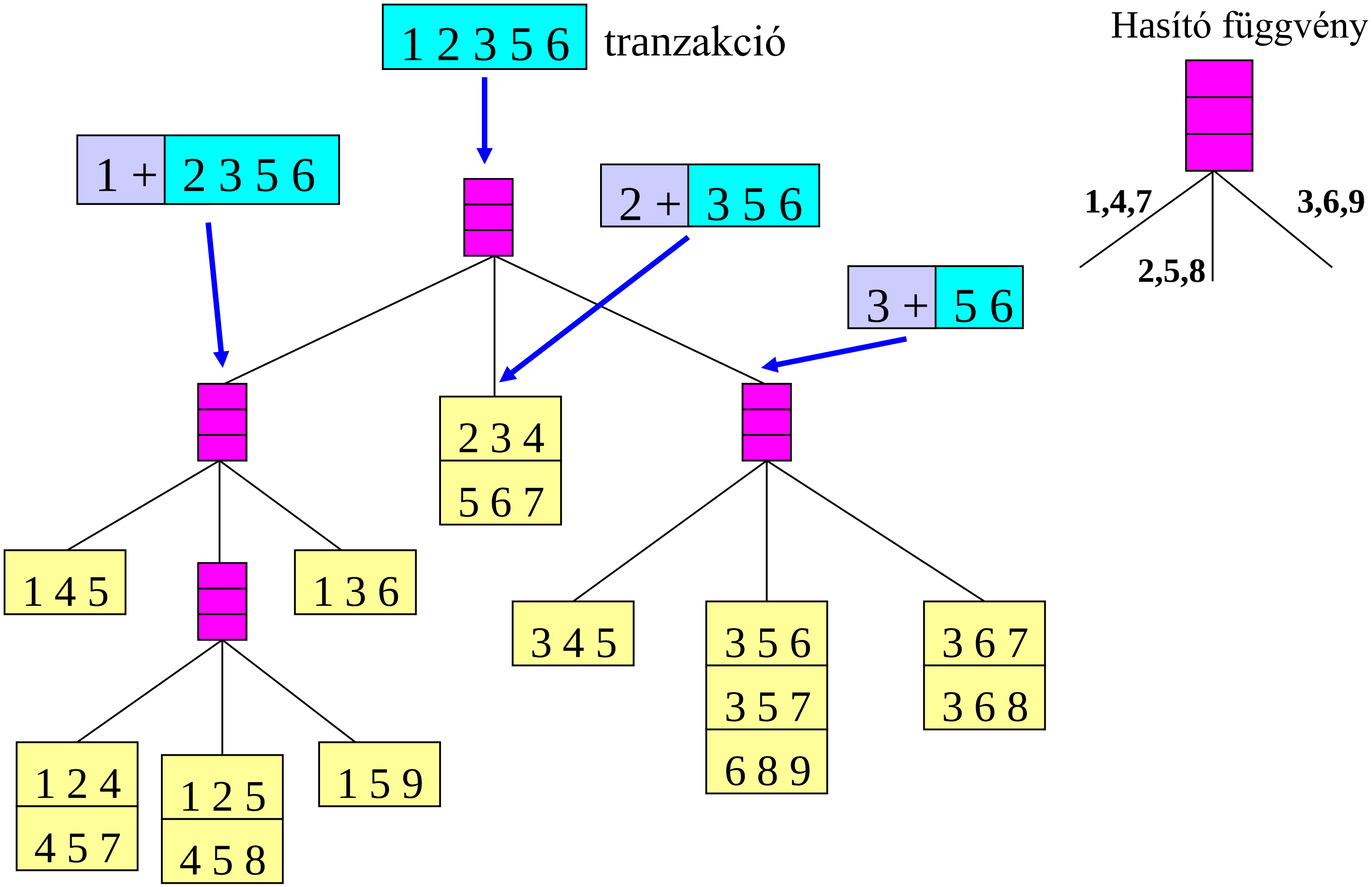
Hasítás:
3, 6, 9

Részhalmaz műveletek

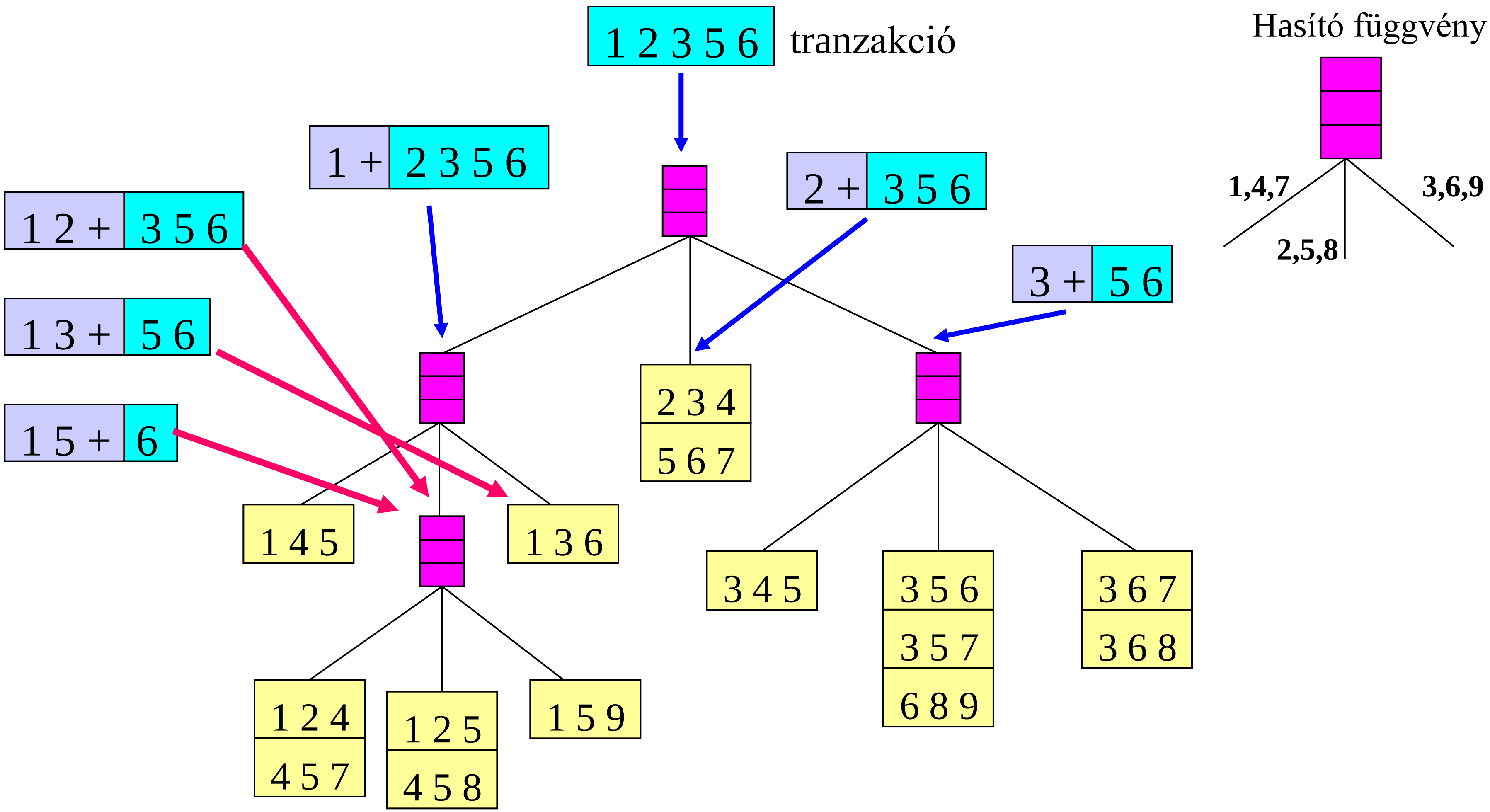
Egy t tranzakció esetén melyek a lehetséges 3 elemű részhalmazok?



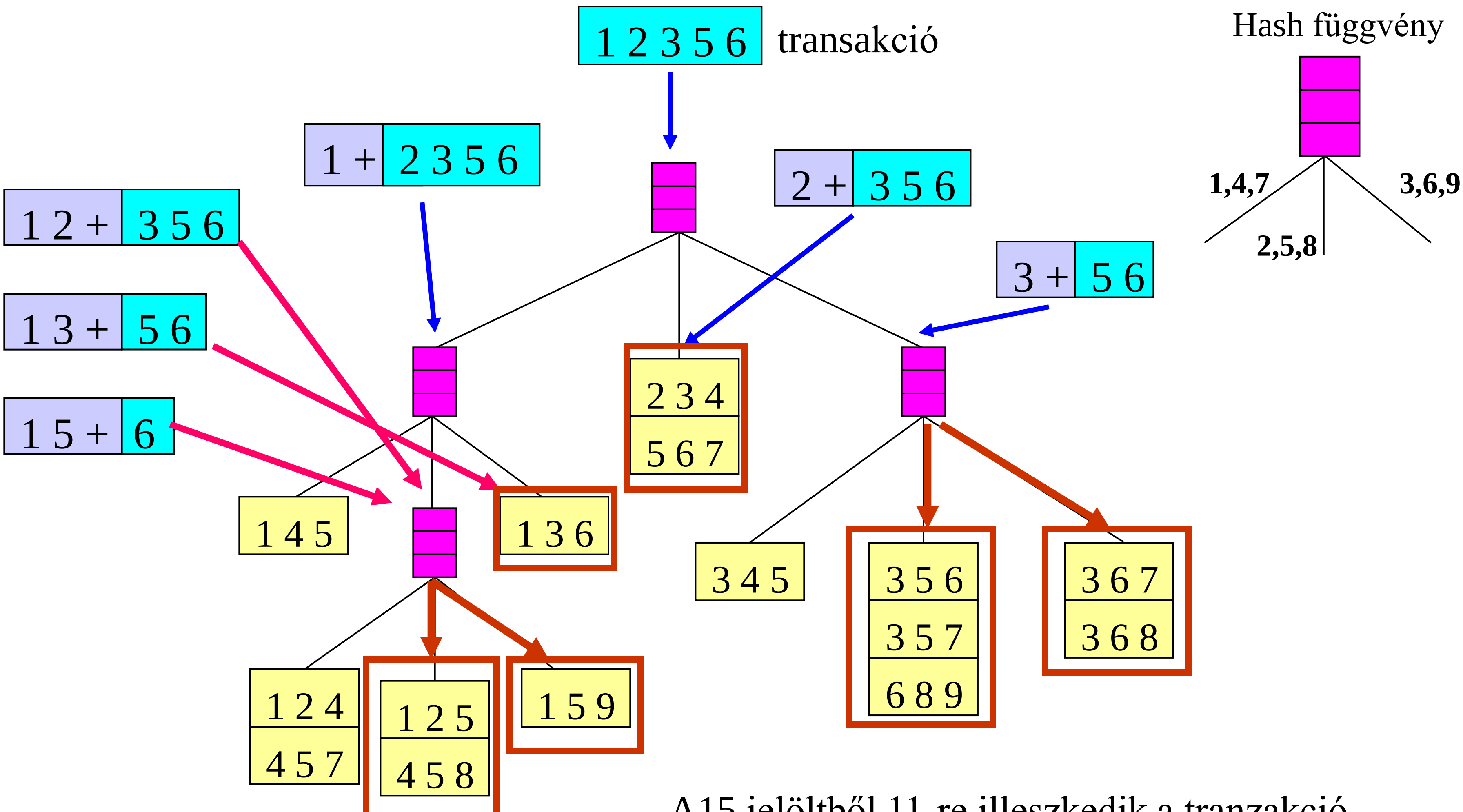
Részhalmaz műveletek a hasító fában



Részhalmaz műveletek a hasító fában



Részhalmaz műveletek a hasító fában



A15 jelöltből 11-re illeszkedik a tranzakció

A komplexitást befolyásoló tényezők

- A minimális támogatottság megválasztása
  - Csökkentése több gyakori tételcsoportot eredményez.
  - Növelheti a jelöltek számát és a gyakori tételcsoportok hosszát.
- Az adatállomány dimenziója (tételek száma)
  - Több hely szükséges a tételek támogatottságának tárolására.
  - Ha a gyakori tételek száma is nő, akkor a számításigény és az I/O költség is nő.
- Az adatbázis mérete
  - Mivel az apriori többször végigfésüli az adatbázist a futási idő nő a tranzakció számmal.
- Átlagos tranzakció szélesség
  - A tranzakció szélesség együtt nő az adathalmaz (tételek) növekedésével.
  - Növelheti a gyakori tételcsoportok maximális hosszát és a hasító fa szélességét (a tranzakcióbeli részhalmazok száma együtt nő a szélességével).

Gyakori tételcsoportok kompakt reprezentációja

- Egyes tételcsoportok redundánsak mivel azonos a támogatottságuk egyes bővítéseikével.

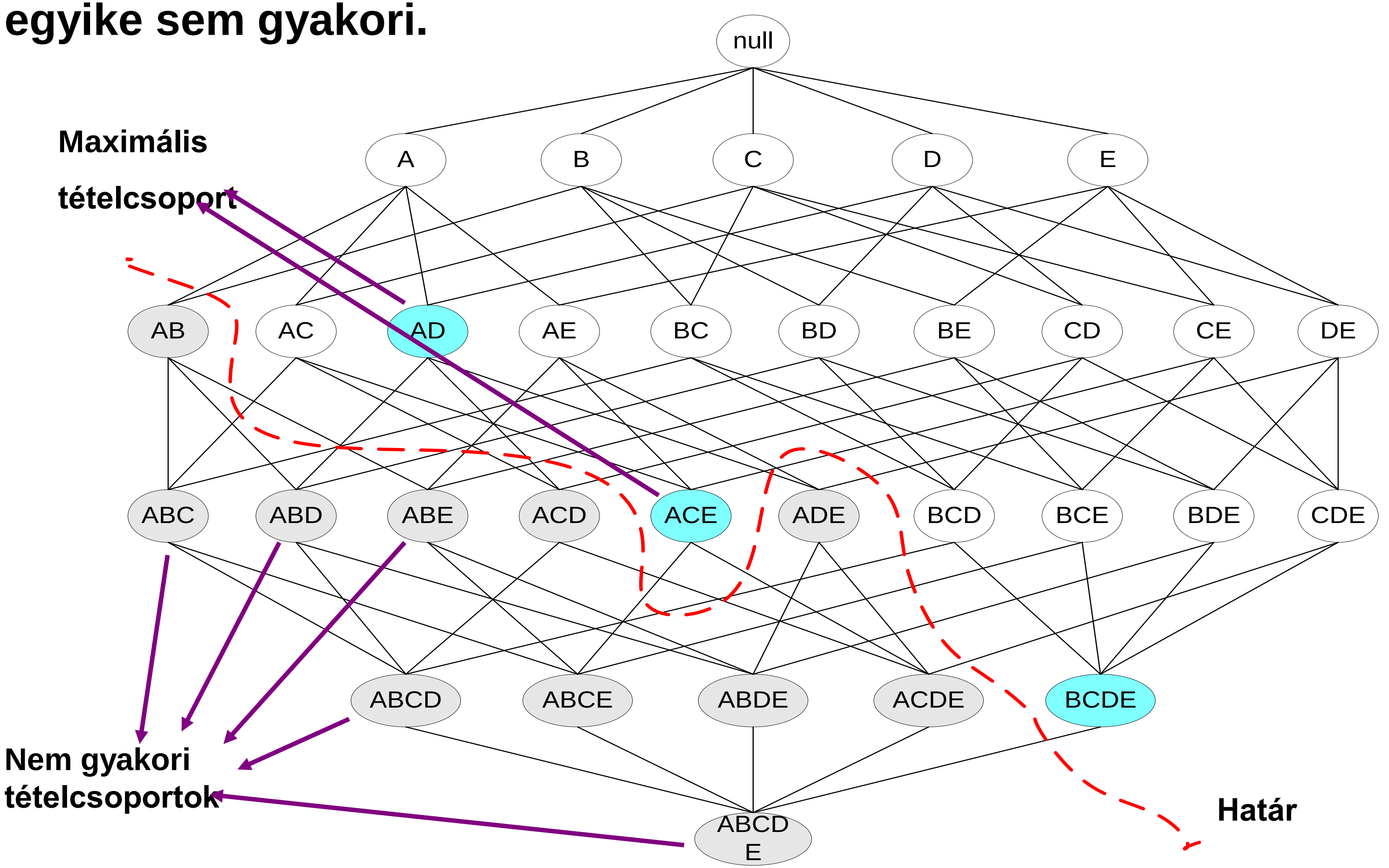
TID	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**

- Gyakori tételcsoportok száma $= 3 \times \sum_{k=1}^{10} \binom{10}{k}$

- Kompakt reprezetációra van szükség!

Maximális gyakori tételcsoport

Egy gyakori tételcsoport maximális, ha közvetlen bővítéseinek egyike sem gyakori.



Zárt tételcsoport

- Egy tételcsoport zárt, ha közvetlen bővítéseinek egyikével sem egyezik meg a támogatottsága.

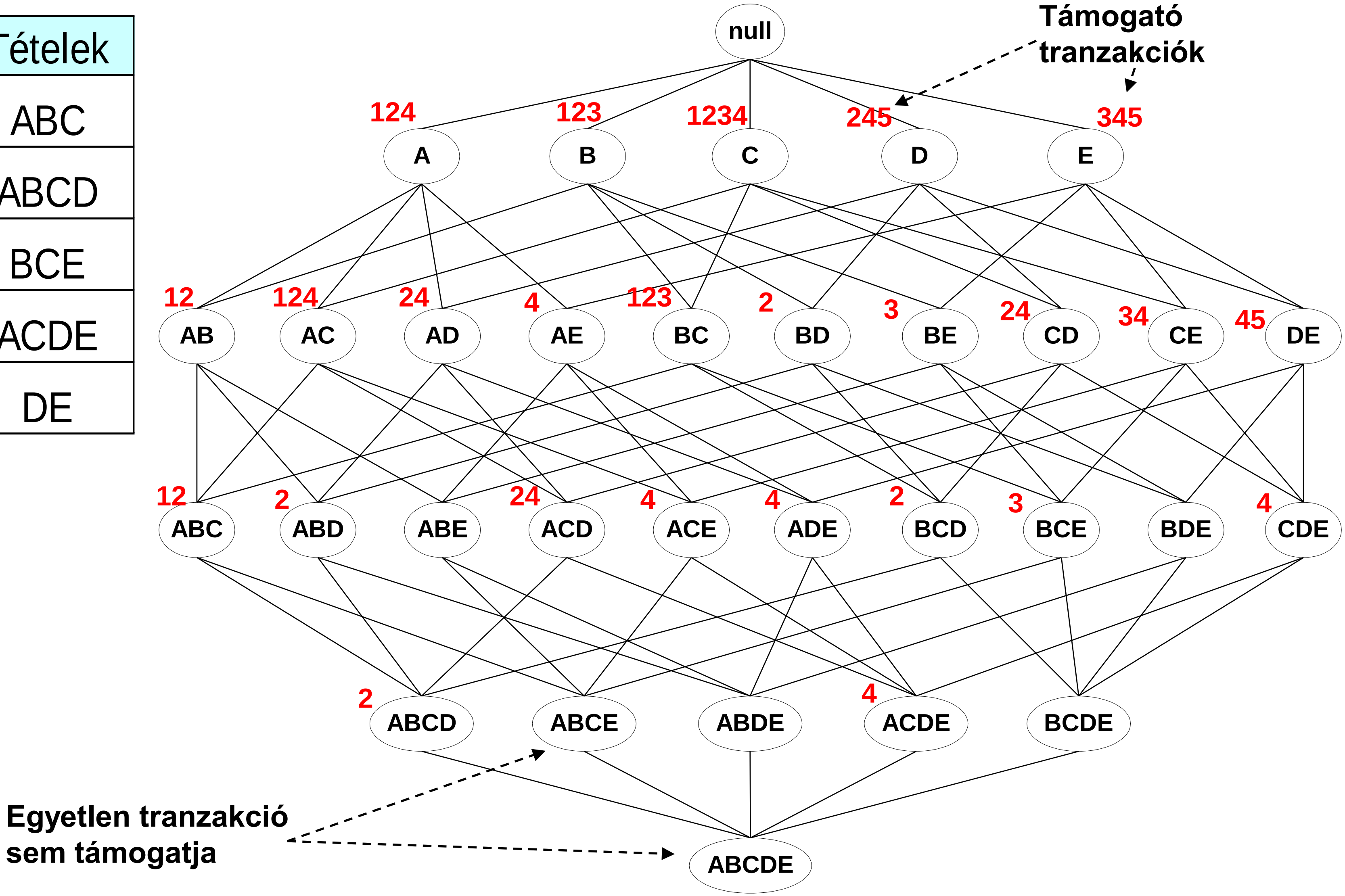
TID	Tételek
1	{A,B}
2	{B,C,D}
3	{A,B,C,D}
4	{A,B,D}
5	{A,B,C,D}

Tételcsoport	Támogatottság
{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

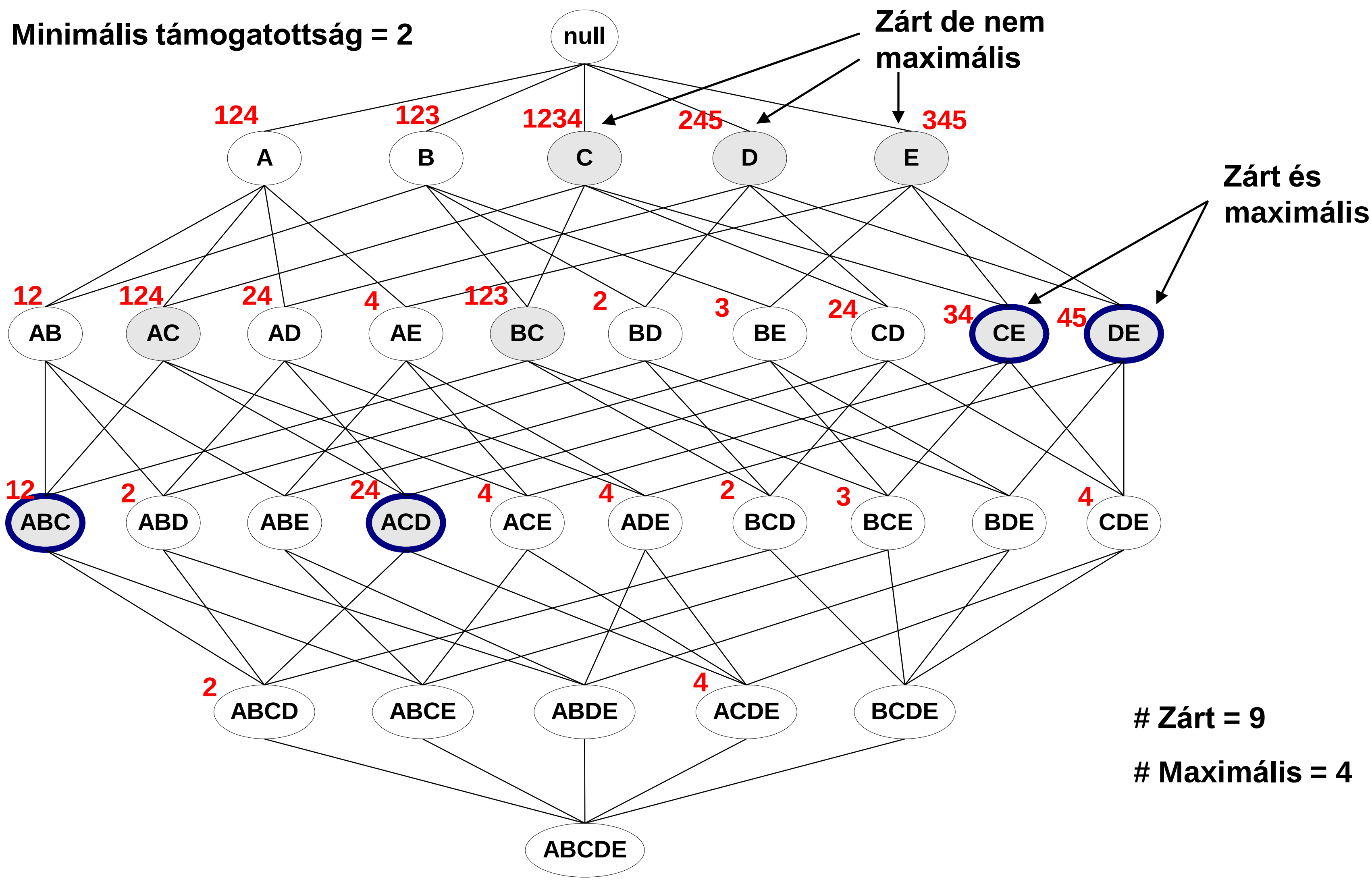
Tételcsoport	Támogatottság
{A,B,C}	2
{A,B,D}	3
{A,C,D}	2
{B,C,D}	3
{A,B,C,D}	2

Maximális vagy zárt tételcsoportok

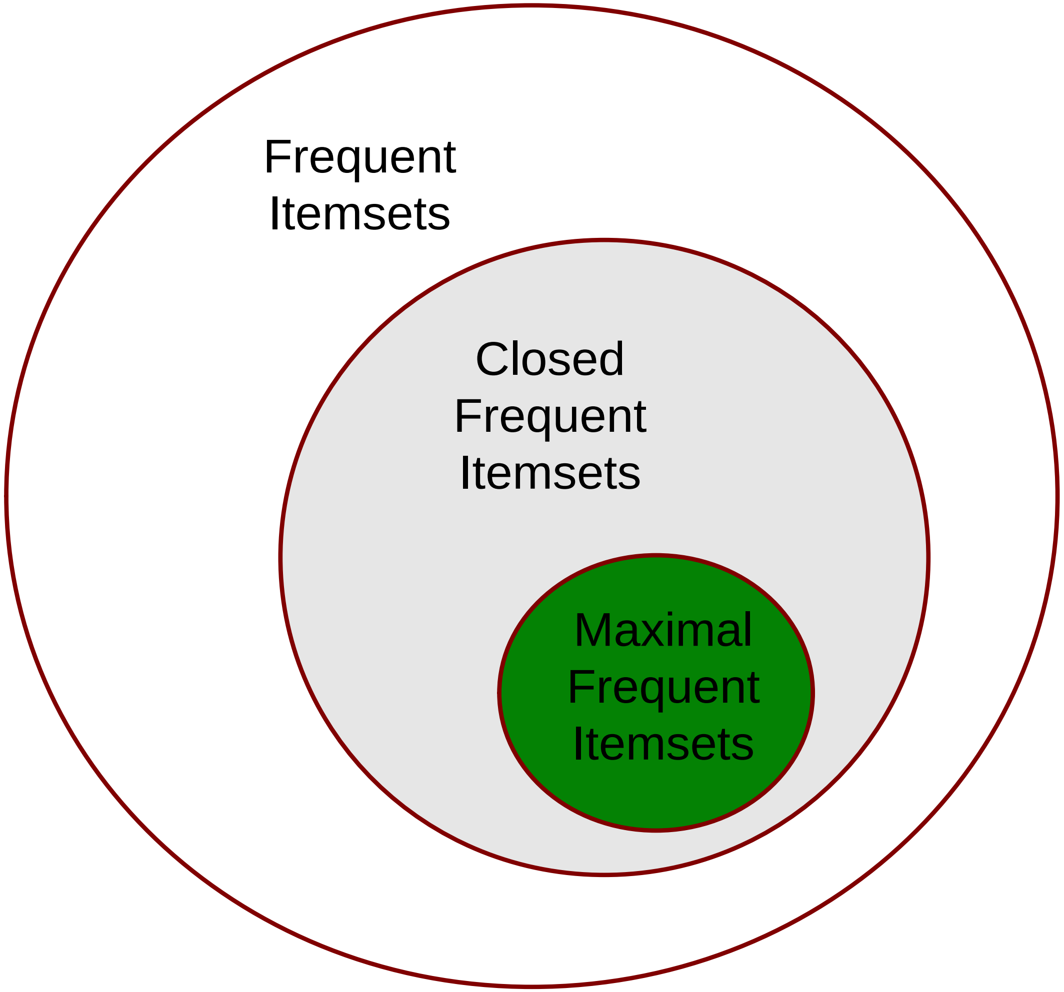
TID	Tételek
1	ABC
2	ABCD
3	BCE
4	ACDE
5	DE



Maximális vagy zárt gyakori tételcsoportok

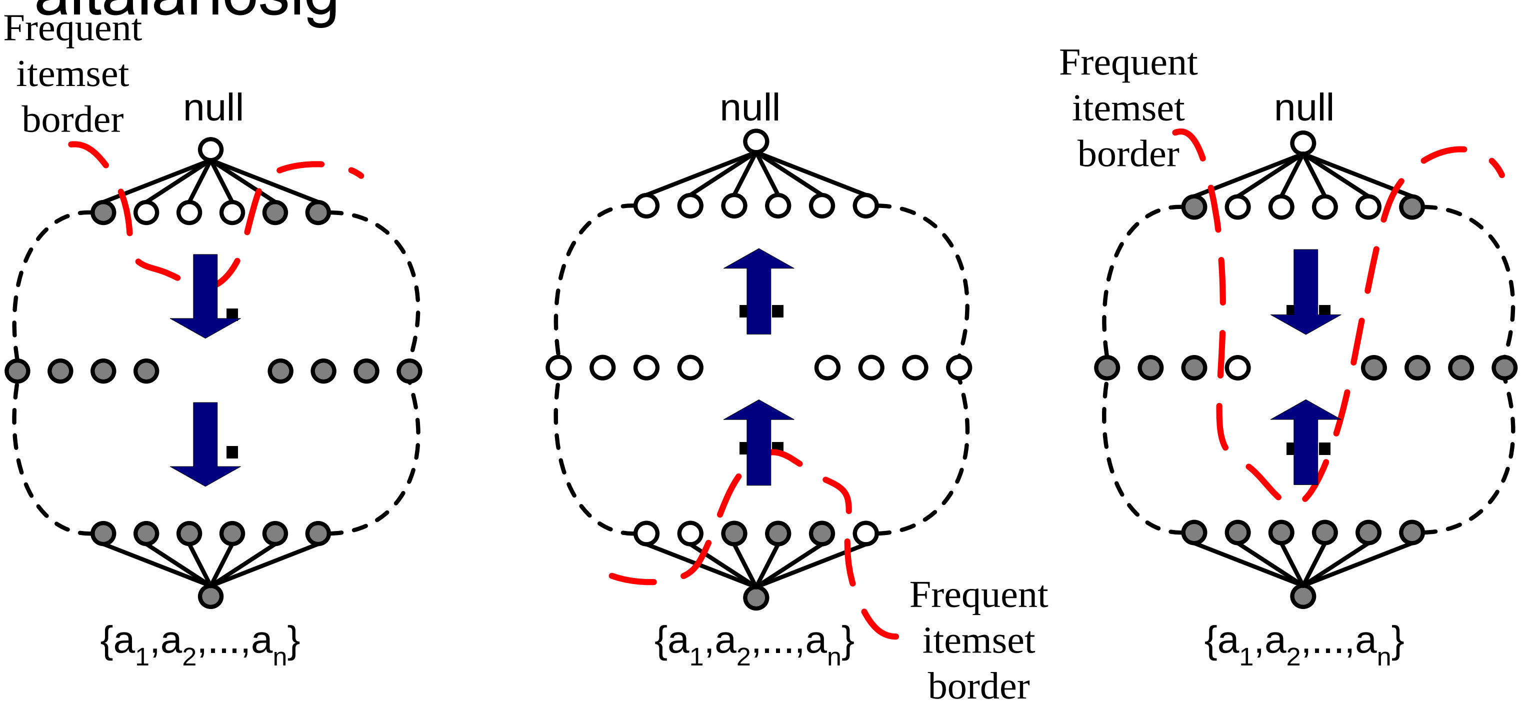


Maximális vagy zárt tételcsoportok



További módszerek gyakori tételcsoportok előállítására

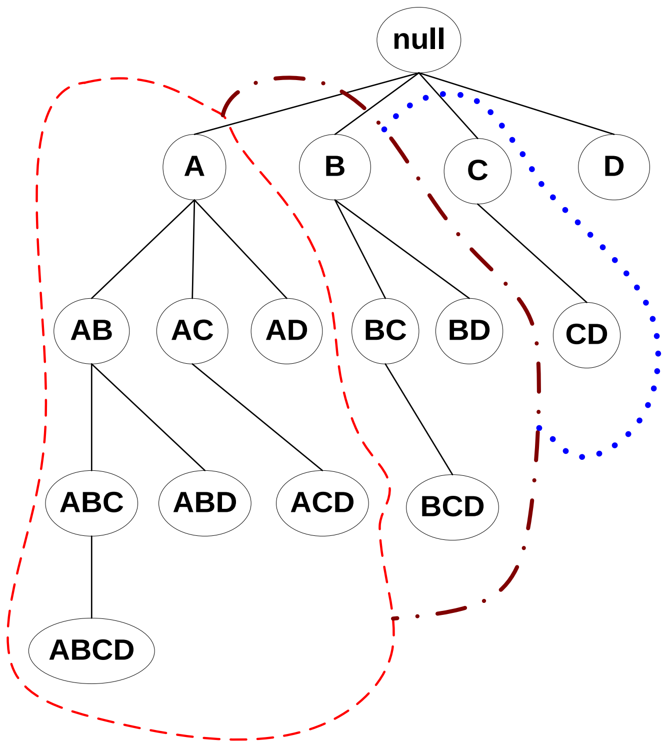
- Átkelés a tételcsoport gráfon
  - Általánostól a speciálisig vagy speciálistól az általánosig



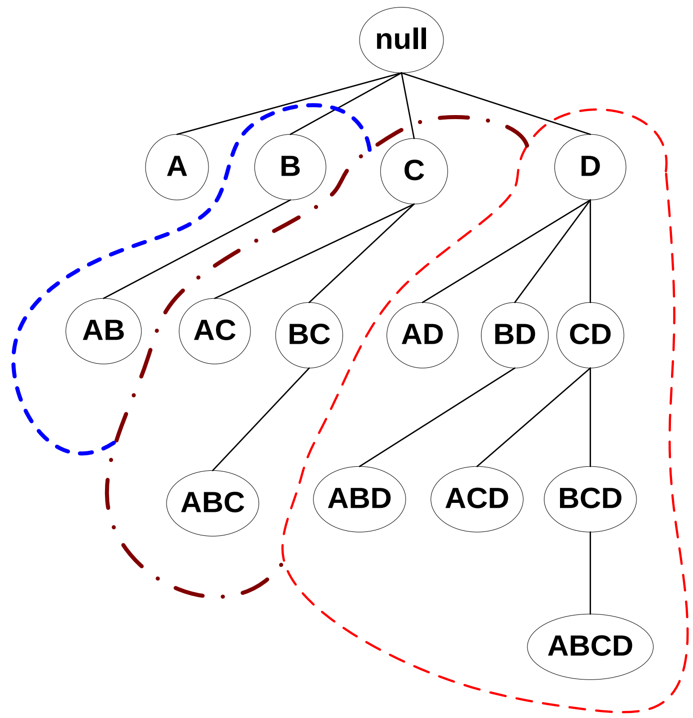
(a) General-to-specific (b) Specific-to-general (c) Bidirectional

További módszerek gyakori tételcsoportok előállítására

- Átkelés a tételcsoport gráfon
  - Ekvivalencia osztályok



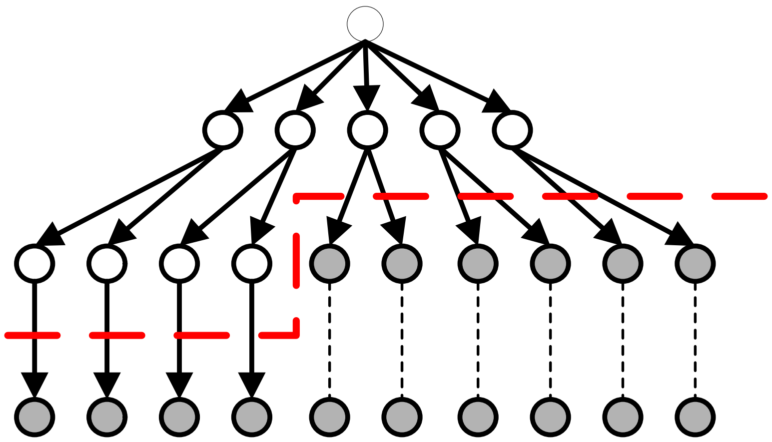
(a) Prefix tree



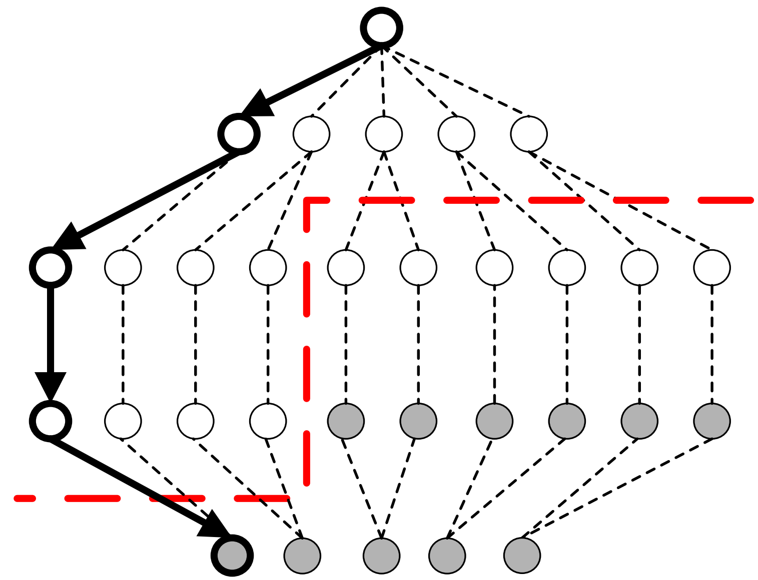
(b) Suffix tree

További módszerek gyakori tételcsoportok előállítására

- Átkelés a tételcsoport gráfon
  - Szélességi vagy mélységi keresés



(a) Breadth first



(b) Depth first

További módszerek gyakori tételcsoportok előállítására

- Az adatbázis reprezentációja
  - Horizontális vagy vertikális elrendezés

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				

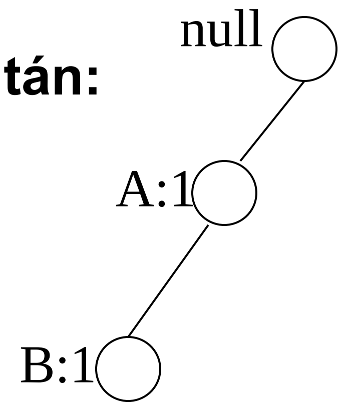
FP-növelő (FP-growth) algortimus

- FP: frequent pattern – gyakori mintázat
- FP-fát használva az adatbázis egy tömörített reprezentációját alkalmazzuk.
- Amint létrehoztuk az FP-fát használjuk azt gyakori tételcsoportok bányászatára az oszd meg és uralkodj elv segítségével.

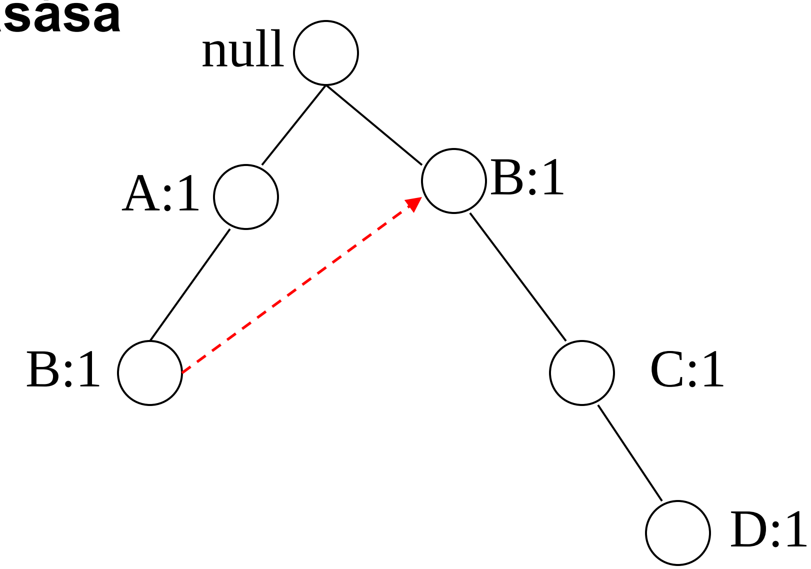
FP-fa konstrukciója

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

TID=1 beolvasása után:



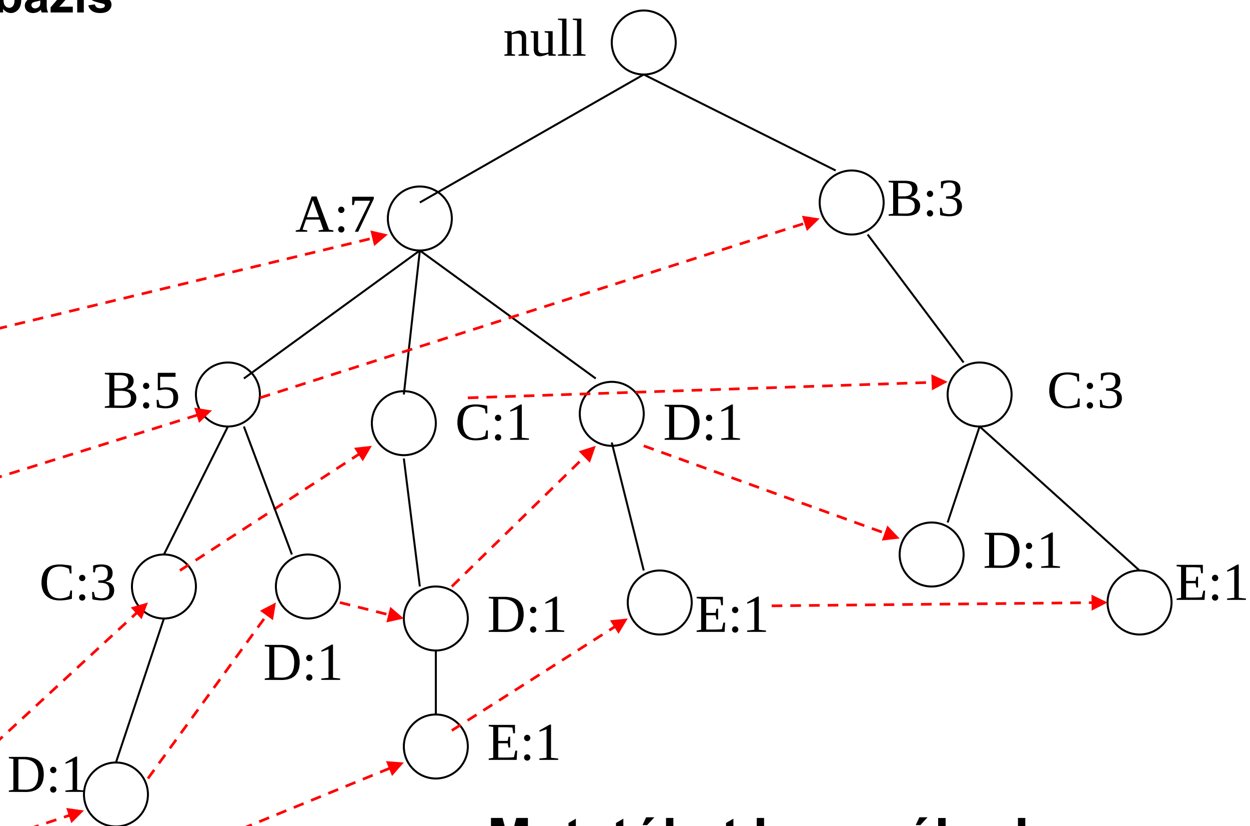
TID=2 beolvasása után:



FP-fa konstrukciója

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

Tranzakciós adatbázis

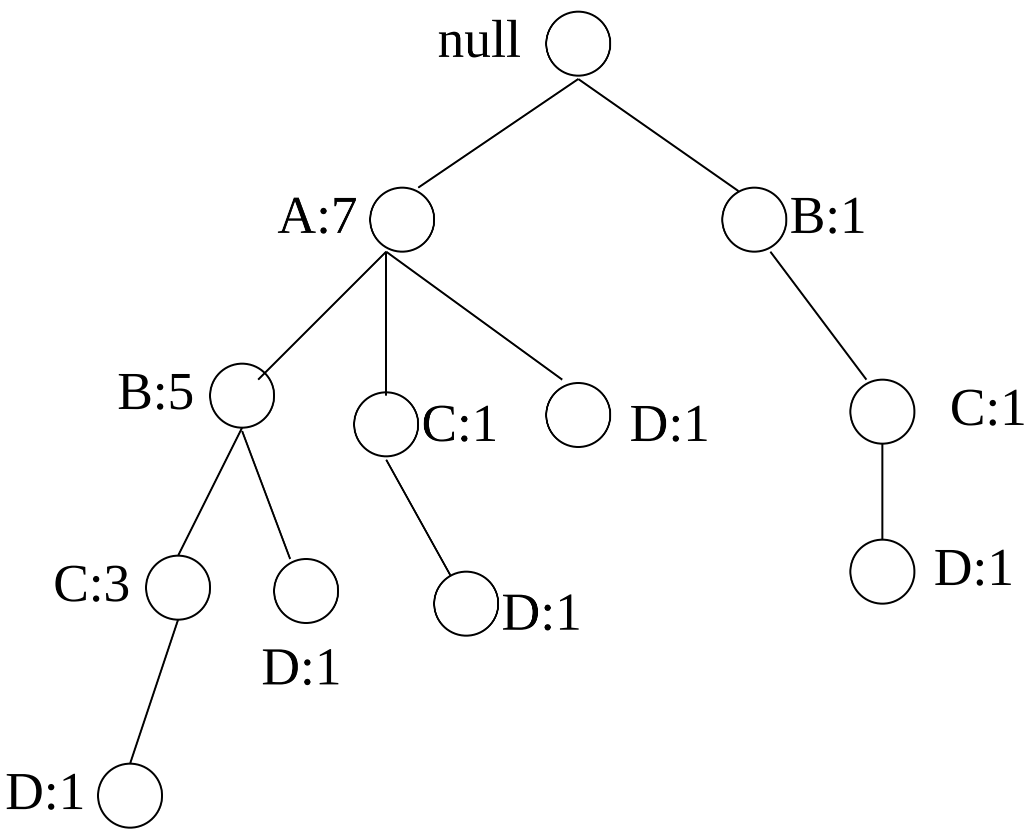


Fejléc tábla

Tétel	Mutató
A	
B	
C	
D	
E	

Mutatókat használunk a gyakori tételcsoportok előállítására

FP-növelés



Feltételes mintázat bázis D-re:

P = {(A:1,B:1,C:1),
(A:1,B:1),
(A:1,C:1),
(A:1),
(B:1,C:1)}

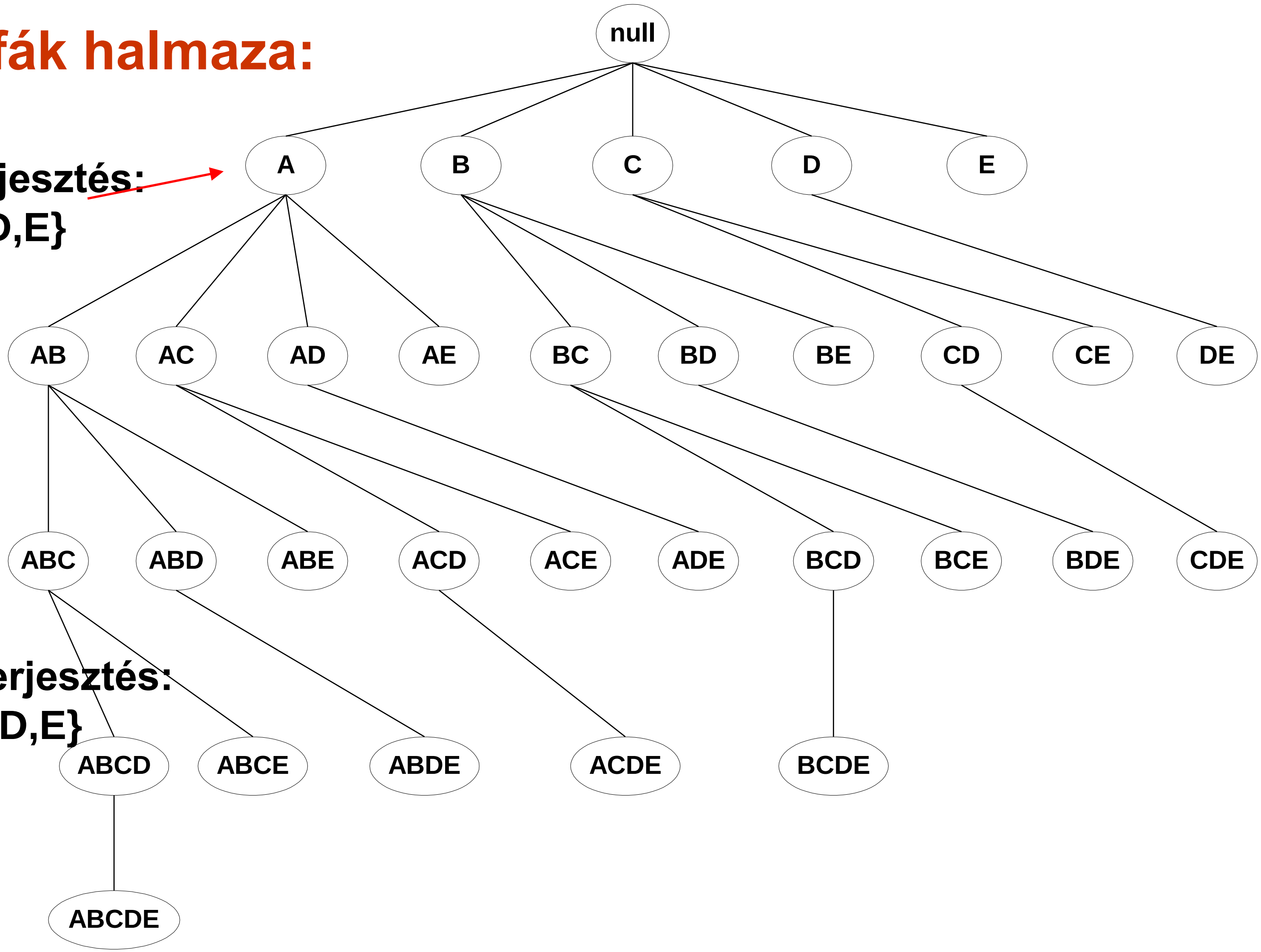
Alkalmazzuk rekurzívan az FP-növelő algoritmust P-n

Talált gyakori tételcsoportok (támogatottság > 1):
AD, BD, CD, ACD, BCD

A fa levetítése

Leszámlált fák halmaza:

Lehetséges kiterjesztés:
E(A) = {B,C,D,E}



Lehetséges kiterjesztés:
E(ABC) = {D,E}

A fa levetítése

- A tételeket rendezzük lexikografikus sorrendbe.
- Minden P csúcs a következő információkat tárolja:
  - A P csúcshoz tartozó tételcsoport.
  - P lehetséges lexikografikus kiterjesztéseinek listája: E(P)
  - Egy mutató, amely az ős csúcshoz tartozó levetített adatbázishoz tartozik.
  - Egy bitvektor, amely azokat a tételcsoportot tartalmazó tranzakciókról tartalmaz információkat, amelyek a levetített adatbázisnak is elemei.

A levetített adatbázis

Eredeti adatbázis:

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

Az A tétel számára levetített adatbázis:

TID	Tételek
1	{B}
2	{}
3	{C,D,E}
4	{D,E}
5	{B,C}
6	{B,C,D}
7	{}
8	{B,C}
9	{B,D}
10	{}

Minden T tranzakcióra az A csomópont levetített tranzakciója $T \cap E(A)$

ECLAT algoritmus

- Minden tételre tároljuk le a hozzá tartozó tranzakciók listáját (tids)

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				

TID-lista

ECLAT

- Egy tetszőleges k-tételcsoport támogatottságát határoz-zuk meg két (k-1) részhalmaza tid-listájának metszetével.

A
1
4
5
6
7
8
9

$\wedge$

B
1
2
5
7
8
10

$\rightarrow$

AB
1
5
7
8

- 3-féle megközelítés:
  – Fentről lefelé, lentről felfelé és hibrid
- Előny: nagyon gyors támogatottság számolás.
- Hátrány: az átmeneti tid-listák túl nagyok lehetnek a memória számára.

Társítási szabályok bányászata

- Tranzakciók egy adott halmazában keressünk olyan szabályokat, amelyek egyes tételek előfordulását előrejelzik más tételek előfordulása alapján.

Vásárlói kosár tranzakciók

TID	*Termékek*
1	Kenyér, Tej
2	Kenyér, Pelenka, Sör, Tojás
3	Tej, Pelenka, Sör, Kóla
4	Kenyér, Tej, Pelenka, Sör
5	Kenyér, Tej, Pelenka, Kóla

Példák társítási szabályra

{Pelenka} $\rightarrow$ {Sör},
{Tej, Kenyér} $\rightarrow$ {Tojás, Kóla},
{Sör, Kenyér} $\rightarrow$ {Tej},

A következtetés együttes előfordu-lásra utal és nem oksági viszonyra!

Társítási szabály fogalma

- **Társítási szabály**
  - Egy X → Y alakú következtetés, ahol X és Y tételcsoportok.
  - Példa:
    {Tej, Pelenka} → {Sör}

TID	*Termékek*
1	**Kenyér, Tej**
2	**Kenyér, Pelenka, Sör, Tojás**
3	**Tej, Pelenka, Sör, Kóla**
4	**Kenyér, Tej, Pelenka, Sör**
5	**Kenyér, Tej, Pelenka, Kóla**

- **Szabály kiértékelési metrikák**
  - Támogatottság (s)
    - Azon tranzakciók aránya, amelyek az X és Y tételcsoportot egyaránt tartalmazzák.
  - Megbízhatóság (c)
    - Azt méri, hogy az Y-beli tételek milyen gyakran jelennek meg olyan tranzakciókban, melyek tartalmazzák X-et.

Példa:

$$\{\text{Tej}, \text{Pelenka}\} \Rightarrow \text{Sör}$$

$$s = \frac{\sigma(\text{Tej}, \text{Pelenka}, \text{Sör})}{|T|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Tej}, \text{Pelenka}, \text{Sör})}{\sigma(\text{Tej}, \text{Pelenka})} = \frac{2}{3} = 0.67$$

Társítási szabályok bányászatának feladata

- Tranzakciók egy adott T halmaza esetén a társítási szabály bányászat célja az összes olyan szabály megtalálása, amelyre
  - támogatottság ≥ *minsup* küszöb,
  - megbízhatóság ≥ *minconf* küszöb.
- Nyers erő megközelítés:
  - Vegyük lajstromba az összes társítási szabályt.
  - Számoljuk ki a támogatottságot és a megbízhatóságot.
  - Távolítsuk el azokat a szabályokat, melyek a *minsup* és *minconf* küszöbnek nem tesznek eleget.

  $\Rightarrow$ Kiszámítási szempontból végrehajthatatlan!

Társítási szabályok bányászata

TID	*Termékek*
1	**Kenyér, Tej**
2	**Kenyér, Pelenka, Sör, Tojás**
3	**Tej, Pelenka, Sör, Kóla**
4	**Kenyér, Tej, Pelenka, Sör**
5	**Kenyér, Tej, Pelenka, Kóla**

Példák szabályokra:

{Tej, Pelenka} $\rightarrow$ {Sör} (s=0.4, c=0.67)
{Tej, Sör} $\rightarrow$ {Pelenka} (s=0.4, c=1.0)
{Pelenka, Sör} $\rightarrow$ {Tej} (s=0.4, c=0.67)
{Sör} $\rightarrow$ {Tej, Pelenka} (s=0.4, c=0.67)
{Pelenka} $\rightarrow$ {Tej, sör} (s=0.4, c=0.5)
{Tej} $\rightarrow$ {Pelenka, Sör} (s=0.4, c=0.5)

Észrevételek:

- Az összes fenti szabály ugyanannak a tételcsoportnak bináris partíciója:
  {Tej, Pelenka, Sör}
- Az ugyanarra a tételcsoportra visszavezethető szabályoknak azonos a támogatottsága a megbízhatósága viszont eltérő lehet.
- Így a támogatottsági és megbízhatósági követelményeket elválaszthatjuk.

Társítási szabályok bányászata

- Kétlépéses megközelítés:
  1. Gyakori tételcsoportok előállítása
     – Állítsuk elő az összes olyan tételcsoportot, melyre támogatottság $\geq$ minsup.
  2. Szabály generálás
     – Állítsuk elő azokat a magas megbízhatóságú szabályokat minden gyakori tételcsoportra, amelyek a tételcsoport bináris partíciói.
- A gyakori tételcsoportok előállítása még mindig kiszámításilag költséges.

Szabály generálás

- Egy adott L gyakori tételcsoportra találjuk meg az összes olyan nemüres $f \subset L$ részhalmazt, melyre $f \rightarrow L - f$ eleget tesz a minimális megbízhatósági követelménynek.
  - Ha {A,B,C,D} gyakori tételcsoport, akkor a szabály jelöltek:

$ABC \rightarrow D$,	$ABD \rightarrow C$,	$ACD \rightarrow B$,	$BCD \rightarrow A$,
$A \rightarrow BCD$,	$B \rightarrow ACD$,	$C \rightarrow ABD$,	$D \rightarrow ABC$
$AB \rightarrow CD$,	$AC \rightarrow BD$,	$AD \rightarrow BC$,	$BC \rightarrow AD$,
$BD \rightarrow AC$,	$CD \rightarrow AB$,		

- Ha $|L| = k$, akkor $2^k - 2$ társítási szabály jelölt van (figyelmen kívül hagyva a $L \rightarrow \emptyset$ és $\emptyset \rightarrow L$ szabályokat)

Szabály generálás

- Hogyan állíthatunk elő hatékonyan szabályokat gyakori tételcsoportokból?
  - A megbízhatóság általában nem rendelkezik az anti-monotonitás tulajdonsággal:

    $c(ABC \rightarrow D)$ lehet kisebb vagy nagyobb mint $c(AB \rightarrow D)$

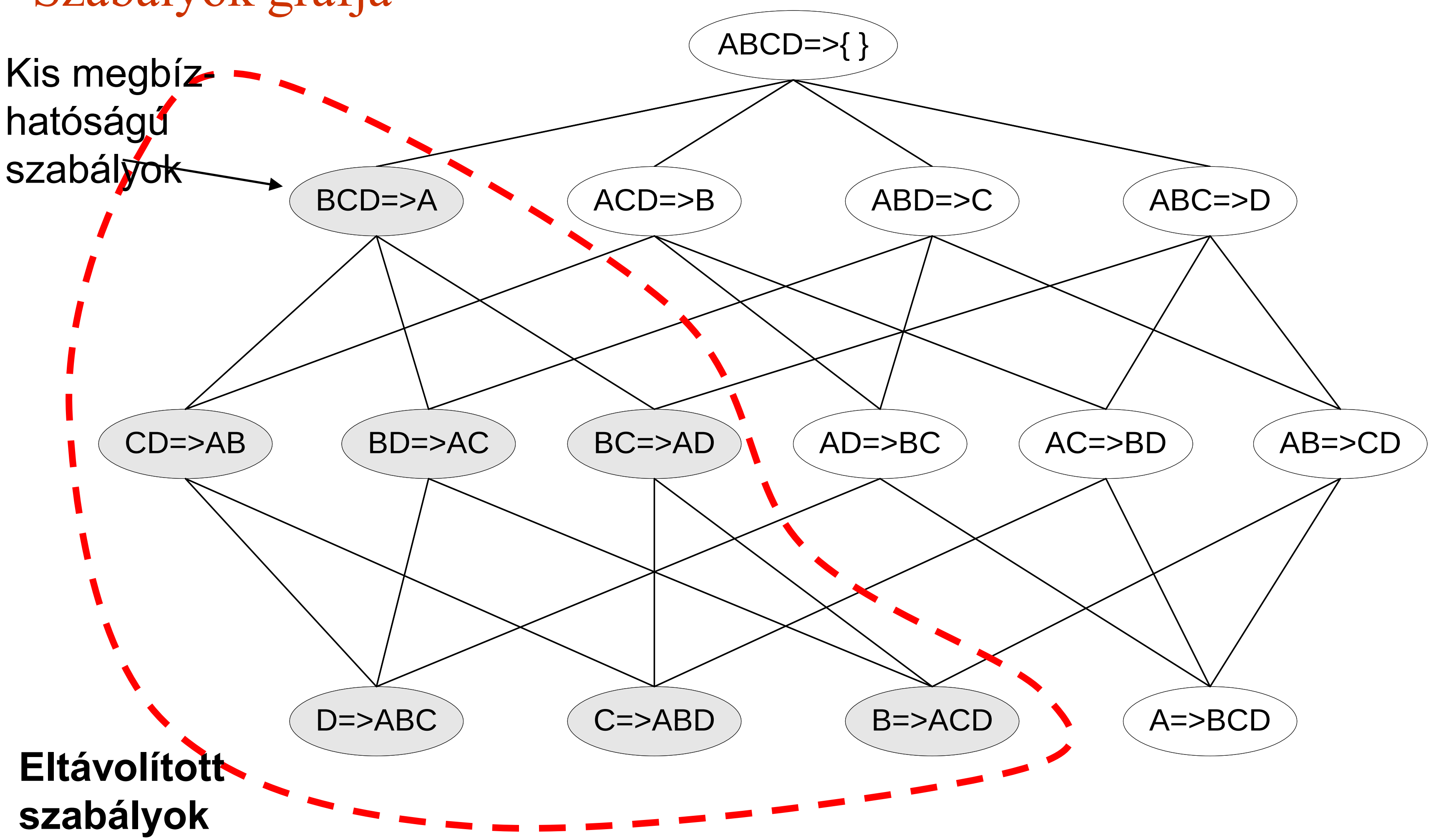
  - Azonban az ugyanabból a tételcsoportból képzett szabályok megbízhatósága már anti-monoton.
  - Például ha L = {A,B,C,D}:

    $$c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$$

    - A megbízhatóság anti-monoton a szabály jobboldalán lévő tételek számát tekintve.

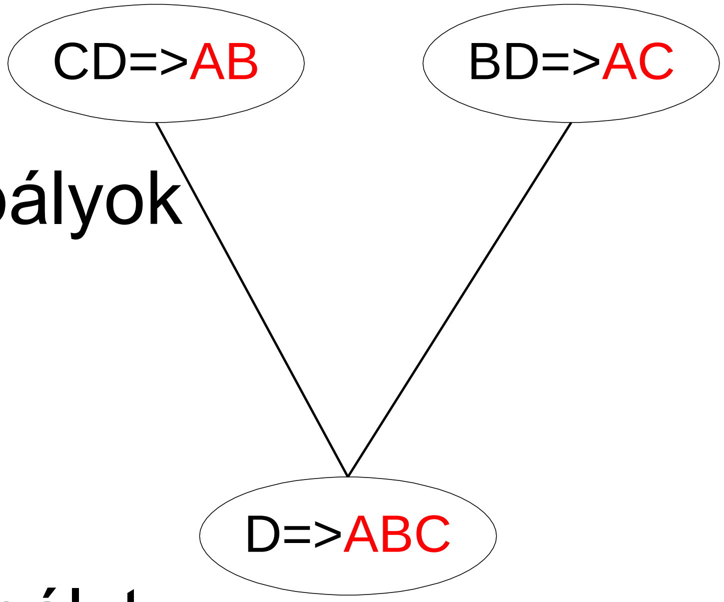
Szabály generálás az apriori algoritmussal

Szabályok gráfja



Szabály generálás az apriori algoritmussal

- Egy szabály-jelöltet két olyan szabály egyesítésével kapunk, amelyeknek ugyanaz a prefixe a szabály következményében.



- A CD=>AB és BD=>AC szabályok egyesítése a D => ABC

  szabályt adja.

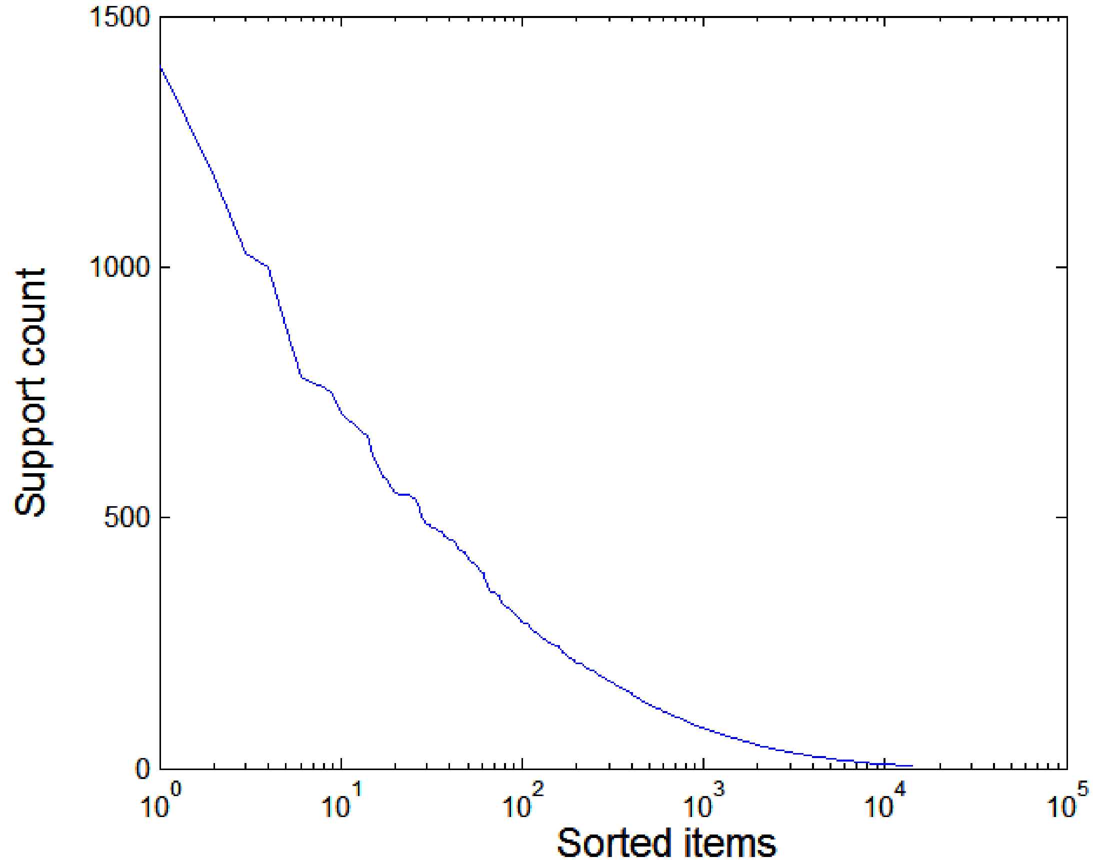
- Távolítsuk el a D=>ABC szabályt,

  ha annak AD=>BC részhalmazának nem elég nagy a megbízhatósága.

A támogatottság eloszlásának hatása

- Sok valós adatállománynál a támogatottság eloszlása ferde.

Egy kiskereskedelmi adatállomány támogatottsági eloszlása



A támogatottság eloszlásának hatása

- Hogyan válasszuk meg a megfelelő *minsup* szintet?
  – Ha a *minsup* túl nagy, akkor elveszthetünk olyan tételcsoportokat, amelyek érdekes ritka tételeket tartalmazhatnak (pl. drága termékek).

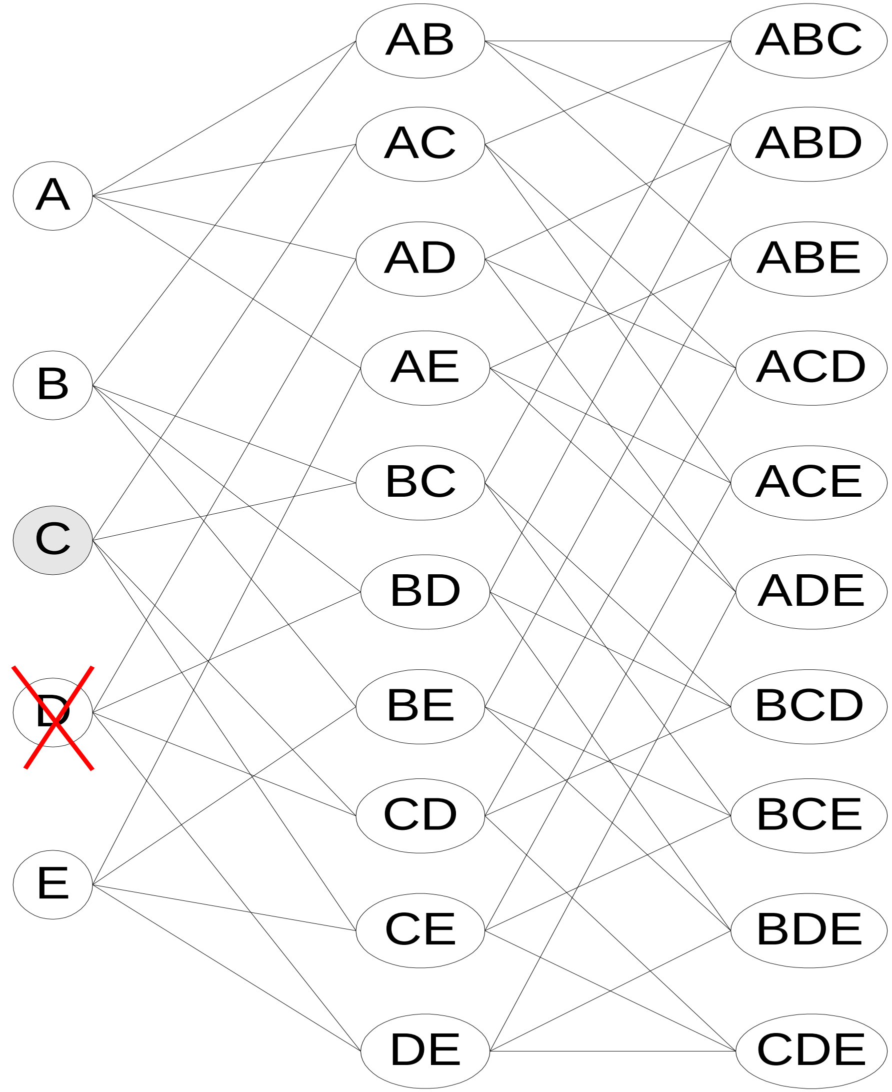
  – Ha a *minsup* túl kicsi, akkor az algoritmus kiszámítási-lag költséges és a tételcsoportok száma is nagyon nagy.
- Egy közös minimális támogatottsági szint használata nem biztos, hogy hatékony.

Többszörös minimális támogatottság

- Hogyan alkalmazzuk a többszörös minimális támogatottságot?
  - MS(i): az i tétel minimális támogatottsága
  - Pl.: MS(Tej)=5%, MS(Kóla) = 3%,
    MS(Brokkoli)=0.1%, MS(Lazac)=0.5%
  - MS({Tej, Brokkoli}) = min (MS(Tej), MS(Brokkoli))
    = 0.1%
  - Kihívás: a támogatottság többé nem anti-monoton
    - Tegyük fel: Support(Tej, Kóla) = 1.5% és
      Support(Tej, Kóla, Brokkoli) = 0.5%
    - {Tej,Kóla} nem gyakori, azonban {Tej,Kóla,Brokkoli} gyakori

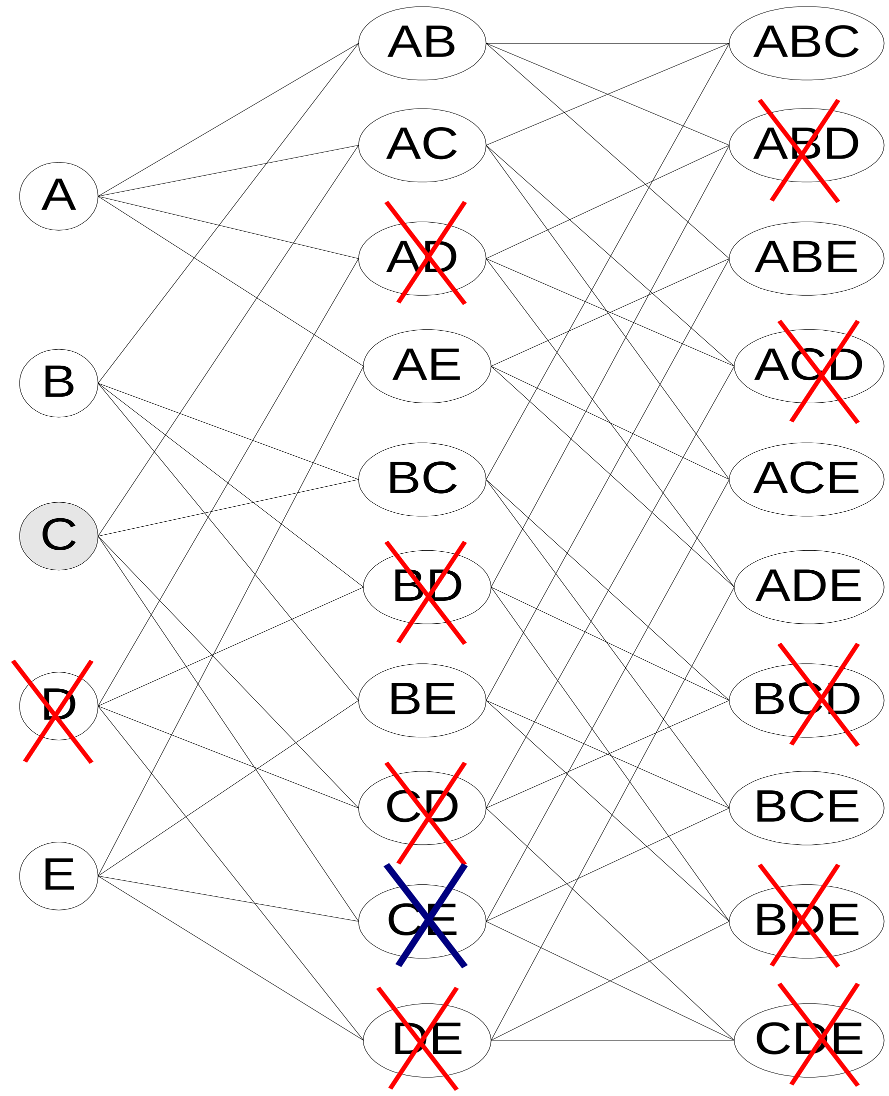
Többszörös minimális támogatottság

Item	MS(I)	Sup(I)
A	0.10%	0.25%
B	0.20%	0.26%
C	0.30%	0.29%
D	0.50%	0.05%
E	3%	4.20%



Többszörös minimális támogatottság

Item	MS(I)	Sup(I)
A	0.10%	0.25%
B	0.20%	0.26%
C	0.30%	0.29%
D	0.50%	0.05%
E	3%	4.20%



Többszörös minimális támogatottság (Liu 1999)

- Rendezzük a tételeket minimális támogatottságuk alapján növekvő sorrendbe.
  - Pl.: MS(Tej)=5%, MS(Kóla) = 3%, MS(Brokkoli)=0.1%, MS(Lazac)=0.5%
  - Rendezés: Brokkoli, Lazac, Kóla, Tej

- Az alábbi módon kell módosítani az Apriori algoritmust:
  - L_1 : gyakori tételek halmaza
  - F_1 : azon tételek halmaza, amelyek támogatottsága $\geq$ MS(1) ahol $MS(1) = \min_i(MS(i))$
  - C_2 : azon 2-tételcsoport jelöltek, amelyeket L_1 helyett F_1-ből generálhatunk

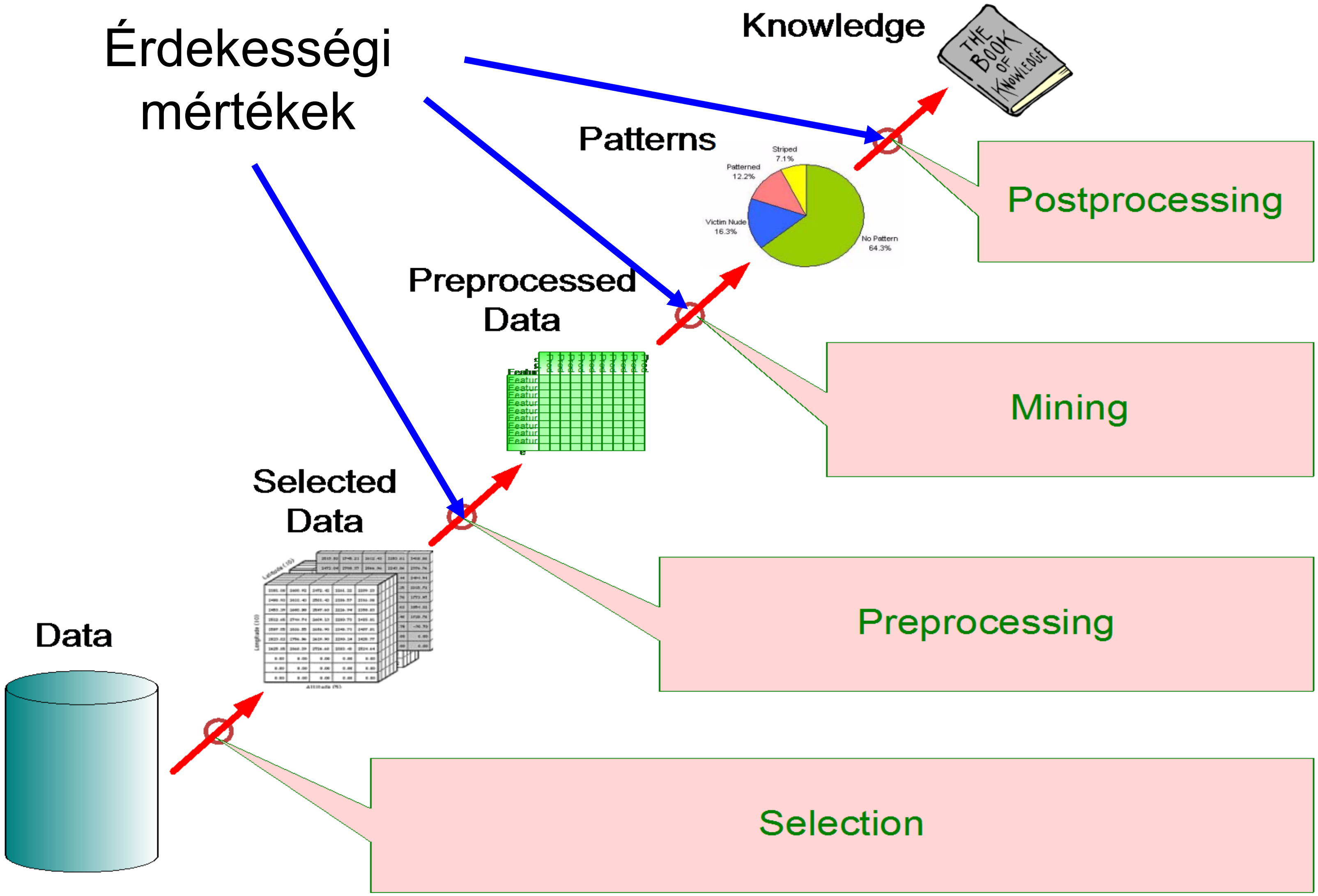
Többszörös minimális támogatottság (Liu 1999)

- Az Apriori algoritmus módosítása:
  - A hagyományos Apriori:
    - A (k+1)-tételcsoportokat két gyakori k-tételcsoport egyesítésével állítjuk elő.
    - Egy jelöltet törlünk ha bármelyik k hosszú résztételcsoportja nem gyakori.
  - A törlés lépést módosítani kell:
    - Csak akkor töröljünk, ha a résztételcsoport tartalmazza az első tételt.
    - Pl.: Jelölt={Brokkoli, Kóla, Tej} (minimális támogatottság szerint rendezve)
    - {Brokkoli, Kóla} és {Brokkoli, Tej} gyakoriak, azonban {Kóla, Tej} már nem gyakori
      - A jelöltet nem töröljük mivel a {Kóla,Tej} nem tartalmazza az első tételt, azaz a Brokkolit.

Mintázat kiértékelés

- A társítási szabály algoritmusok hajlamosak túl sok szabályt szolgáltatni.
  - Sok közülük nem érdekes vagy redundáns.
  - Redundáns ha $\{A,B,C\} \rightarrow \{D\}$ és $\{A,B\} \rightarrow \{D\}$ szabályoknak megegyezik a támogatottsága és a megbízhatósága.
- Érdekességi mértékeket használhatunk az eredményül kapott minták törlésére vagy sorba rendezésére.
- A társítási szabályok bevezetésekor csak a támogatottság és megbízhatóság mértékeket alkalmazták.

Érdekességi mértékek alkalmazása



Érdekességi mértékek meghatározása

- Egy adott $X \rightarrow Y$ szabály esetén az érdekességi mértékek meghatározásához szükséges információk egy kontingencia táblából kaphatóak.

Kontingencia tábla az $X \rightarrow Y$ szabályra

	Y	$\overline{Y}$	
X	f_{11}	f_{10}	f_{1+}
$\overline{X}$	f_{01}	f_{00}	f_{o+}
	f_{+1}	f_{+0}	$\|T\|$

f_{11}: X és Y támogatottsága
f_{10}: X és $\overline{Y}$ támogatottsága
f_{01}: $\overline{X}$ és Y támogatottsága
f_{00}: $\overline{X}$ és $\overline{Y}$ támogatottsága

Számos mérőszám definiálására használható

- támogatottság, megbízhatóság, lift, Gini, J-mérték stb.

A megbízhatóság hátránya

	Kávé	$\overline{\text{Kávé}}$	
Tea	15	5	20
$\overline{\text{Tea}}$	75	5	80
	90	10	100

Társítási szabály: Tea → Kávé

Megbízhatóság= P(Kávé|Tea) = 0.75

azonban P(Coffee) = 0.9

⇒ Bár a megbízhatóság nagy, a szabály megtévesztő

⇒ P(Kávé|$\overline{\text{Tea}}$) = 0.9375

Statisztikai függetlenség

- 1000 hallgató populációja
  - 600 hallgató tud úszni (S)
  - 700 hallgató tud biciklizni (B)
  - 420 hallgató tud úszni és biciklizni (S,B)

  - $P(S \wedge B) = 420/1000 = 0.42$
  - $P(S) \times P(B) = 0.6 \times 0.7 = 0.42$

  - $P(S \wedge B) = P(S) \times P(B)$ => Statisztikai függetlenség
  - $P(S \wedge B) > P(S) \times P(B)$ => Pozitív korreláció
  - $P(S \wedge B) < P(S) \times P(B)$ => Negatív korreláció

Statisztika alapú mérőszámok

- Az alábbi mérőszámok figyelembe veszik a statisztikus függetlenséget

$$Lift = \frac{P(Y \mid X)}{P(Y)}$$

$$Érdekesség = \frac{P(X,Y)}{P(X)P(Y)}$$

$$PS = P(X,Y) - P(X)P(Y)$$

$$\phi = \frac{P(X,Y) - P(X)P(Y)}{\sqrt{P(X)[1-P(X)]P(Y)[1-P(Y)]}}$$

Példa: Lift/Érdekesség

	Kávé	$\overline{\text{Kávé}}$	
Tea	15	5	20
$\overline{\text{Tea}}$	75	5	80
	90	10	100

Társítási szabály: Tea $\rightarrow$ Kávé

Megbízhatóság= P(Kávé|Tea) = 0.75

azonban P(Kávé) = 0.9

$\Rightarrow$ Lift = 0.75/0.9= 0.8333 (< 1, ezért negatívan asszociált)

A lift és érdekesség hátránya

	Y	$\overline{Y}$	
X	10	0	10
$\overline{X}$	0	90	90
	10	90	100

	Y	$\overline{Y}$	
X	90	0	90
$\overline{X}$	0	10	10
	90	10	100

$$Lift = \frac{0.1}{(0.1)(0.1)} = 10$$

$$Lift = \frac{0.9}{(0.9)(0.9)} = 1.11$$

Statisztikus függetlenség:

If P(X,Y)=P(X)P(Y) => Lift = 1

Számos mérték ismert az irodalomban

Egyesek közülük jók bizonyos alkalmazásoknál, másoknál azonban nem

Milyen kritériumokat használjunk annak eldöntésére, hogy egy mérték jó vagy rossz?

Mi a helyzet az Apriori stílusú támogatottságon alapuló törléssel? Hogyan hat ez a mértékre?

#	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)	$\frac{P(A,B)P(\overline{A},\overline{B})}{P(A,\overline{B})P(\overline{A},B)}$
4	Yule's Q	$\frac{P(A,B)P(\overline{AB})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{AB})+P(A,\overline{B})P(\overline{A},B)} = \frac{\alpha-1}{\alpha+1}$
5	Yule's Y	$\frac{\sqrt{P(A,B)P(\overline{AB})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{AB})}+\sqrt{P(A,\overline{B})P(\overline{A},B)}} = \frac{\sqrt{\alpha}-1}{\sqrt{\alpha}+1}$
6	Kappa (κ)	$\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$
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)	$\max\Big(P(A,B)\log(\frac{P(B\|A)}{P(B)}) + P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}),$ $P(A,B)\log(\frac{P(A\|B)}{P(A)}) + P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})\Big)$
9	Gini index (G)	$\max\Big(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A})[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]$ $-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B})[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]$ $-P(A)^2-P(\overline{A})^2\Big)$
10	Support (s)	$P(A,B)$
11	Confidence (c)	$\max(P(B\|A), P(A\|B))$
12	Laplace (L)	$\max\Big(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2}\Big)$
13	Conviction (V)	$\max\Big(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})}\Big)$
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)	$\max\Big(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)}\Big)$
18	Added Value (AV)	$\max(P(B\|A)-P(B), P(A\|B)-P(A))$
19	Collective strength (S)	$\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$
20	Jaccard (ζ)	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
21	Klosgen (K)	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$

Egy jó mérték tulajdonságai

- Piatetsky-Shapiro:
  Egy jó M mértéknek az alábbi 3 tulajdonságot kell kielégíteni:
  - M(A,B) = 0 ha A és B statisztikusan független
  - M(A,B) monoton nő P(A,B)-vel amennyiben P(A) és P(B) változatlan marad
  - M(A,B) monoton csökken P(A)-val [vagy P(B)-vel] amennyiben P(A,B) és P(B) [vagy P(A)] változatlan marad

Különböző mértékek összehasonlítása

Példák: 10 kontigencia tábla

Example	f_{11}	f_{10}	f_{01}	f_{00}
E1	8123	83	424	1370
E2	8330	2	622	1046
E3	9481	94	127	298
E4	3954	3080	5	2961
E5	2886	1363	1320	4431
E6	1500	2000	500	6000
E7	4000	2000	1000	3000
E8	4000	2000	2000	2000
E9	1720	7121	5	1154
E10	61	2483	4	7452

A kontingencia táblák rangsorolá-sa különböző mértékek szerint:

#	ϕ	λ	α	Q	Y	κ	M	J	G	s	c	L	V	I	IS	PS	F	AV	S	ζ	K
E1	1	1	3	3	3	1	2	2	1	3	5	5	4	6	2	2	4	6	1	2	5
E2	2	2	1	1	1	2	1	3	2	2	1	1	1	8	3	5	1	8	2	3	6
E3	3	3	4	4	4	3	3	8	7	1	4	4	6	10	1	8	6	10	3	1	10
E4	4	7	2	2	2	5	4	1	3	6	2	2	2	4	4	1	2	3	4	5	1
E5	5	4	8	8	8	4	7	5	4	7	9	9	9	3	6	3	9	4	5	6	3
E6	6	6	7	7	7	7	6	4	6	9	8	8	7	2	8	6	7	2	7	8	2
E7	7	5	9	9	9	6	8	6	5	4	7	7	8	5	5	4	8	5	6	4	4
E8	8	9	10	10	10	8	10	10	8	4	10	10	10	9	7	7	10	9	8	7	9
E9	9	9	5	5	5	9	9	7	9	8	3	3	3	7	9	9	3	7	9	9	8
E10	10	8	6	6	6	10	5	9	10	10	6	6	5	1	10	10	5	1	10	10	7

Változók permutációjának hatása

	B	$\overline{B}$
A	p	q
$\overline{A}$	r	s

$\Rightarrow$

	A	$\overline{A}$
B	p	r
$\overline{B}$	q	s

M(A,B) = M(B,A)?

Szimmetrikus mértékek:

- támogatottság (s), lift, együttes erő (S), koszinusz (IS), Jaccard stb.

Aszimmetrikus mértékek:

- megbízhatóság, meggyőződés, Laplace, J-mérték stb.

Sor/oszlop átskálázás hatása

Fokozat-nem példa (Mosteller, 1968):

	Férfi	Nő	
Magas	2	3	5
Alacsony	1	4	5
	3	7	10

	Férfi	Nő	
Magas	4	30	34
Alacsony	2	40	42
	6	70	76

2x 10x

Mosteller:

A mögöttes kapcsolat erőssége nem függhet a férfiak és nők relatív számától a mintában.

Az inverzió művelet hatása

	A	B	C	D	E	F
1. tranzakció	1	0	0	1	0	0
	0	0	1	1	1	0
	0	0	1	1	1	0
	0	0	1	1	1	0
	0	1	1	0	1	1
	0	0	1	1	1	0
	0	0	1	1	1	0
	0	0	1	1	1	0
	0	0	1	1	1	0
N. tranzakció	1	0	0	1	0	0
	(a)		(b)		(c)	

Példa: ϕ-együttható

- A ϕ-együttható a folytonos változókra ismert korrelációs együttható analógja.

	Y	$\overline{Y}$	
X	60	10	70
$\overline{X}$	10	20	30
	70	30	100

	Y	$\overline{Y}$	
X	20	10	30
$\overline{X}$	10	60	70
	30	70	100

$$\phi = \frac{0.6 - 0.7 \times 0.7}{\sqrt{0.7 \times 0.3 \times 0.7 \times 0.3}} = 0.5238$$

$$\phi = \frac{0.2 - 0.3 \times 0.3}{\sqrt{0.7 \times 0.3 \times 0.7 \times 0.3}} = 0.5238$$

A ϕ együttható mindkét táblára ugyanaz

0 hozzáadásának hatása

	B	$\overline{B}$
A	p	q
$\overline{A}$	r	s

	B	$\overline{B}$
A	p	q
$\overline{A}$	r	s + k

Invariáns mértékek:

- támogatottság, koszinusz, Jaccard stb.

Nem-invariáns mértékek:

- korreláció, Gini, kölcsönös információ, esélyhányados

Különböző mértékek különböző tulajdonságokkal

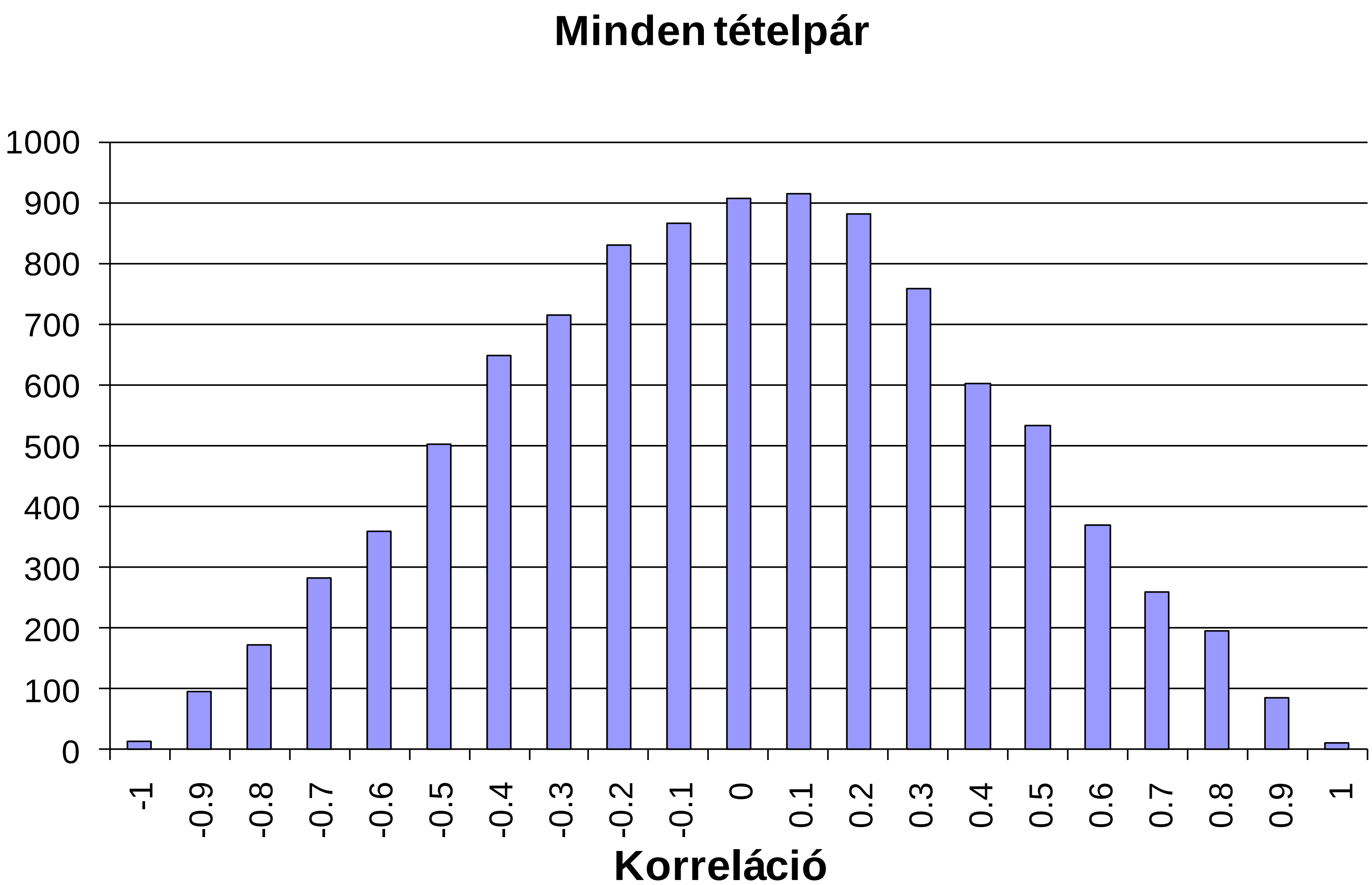
Symbol	Measure	Range	P1	P2	P3	O1	O2	O3	O3'	O4
Φ	Correlation	-1 … 0 … 1	Yes	Yes	Yes	Yes	No	Yes	Yes	No
λ	Lambda	0 … 1	Yes	No	No	Yes	No	No*	Yes	No
α	Odds ratio	0 … 1 … ∞	Yes*	Yes	Yes	Yes	Yes	Yes*	Yes	No
Q	Yule's Q	-1 … 0 … 1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
Y	Yule's Y	-1 … 0 … 1	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
κ	Cohen's	-1 … 0 … 1	Yes	Yes	Yes	Yes	No	No	Yes	No
M	Mutual Information	0 … 1	Yes	Yes	Yes	Yes	No	No*	Yes	No
J	J-Measure	0 … 1	Yes	No	No	No	No	No	No	No
G	Gini Index	0 … 1	Yes	No	No	No	No	No*	Yes	No
s	Support	0 … 1	No	Yes	No	Yes	No	No	No	No
c	Confidence	0 … 1	No	Yes	No	Yes	No	No	No	Yes
L	Laplace	0 … 1	No	Yes	No	Yes	No	No	No	No
V	Conviction	0.5 … 1 … ∞	No	Yes	No	Yes**	No	No	Yes	No
I	Interest	0 … 1 … ∞	Yes*	Yes	Yes	Yes	No	No	No	No
IS	IS (cosine)	0 .. 1	No	Yes	Yes	Yes	No	No	No	Yes
PS	Piatetsky-Shapiro's	-0.25 … 0 … 0.25	Yes	Yes	Yes	Yes	No	Yes	Yes	No
F	Certainty factor	-1 … 0 … 1	Yes	Yes	Yes	No	No	No	Yes	No
AV	Added value	0.5 … 1 … 1	Yes	Yes	Yes	No	No	No	No	No
S	Collective strength	0 … 1 … ∞	No	Yes	Yes	Yes	No	Yes*	Yes	No
ζ	Jaccard	0 .. 1	No	Yes	Yes	Yes	No	No	No	Yes
K	Klosgen's	$\left(\sqrt{\frac{2}{\sqrt{3}}-1}\right)\left(2-\sqrt{3}-\frac{1}{\sqrt{3}}\right) \ldots 0 \ldots \frac{2}{3\sqrt{3}}$	Yes	Yes	Yes	No	No	No	No	No

Támogatottság alapú törlés

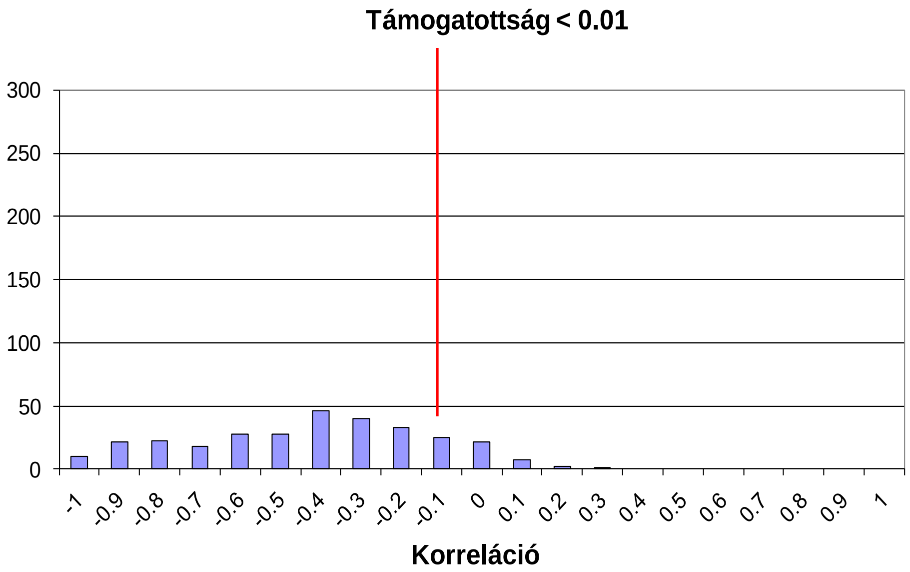
- A legtöbb társítási szabály bányászó algoritmus használ támogatottsági mértéket szabályok és tételcsoportok eltávolítására.

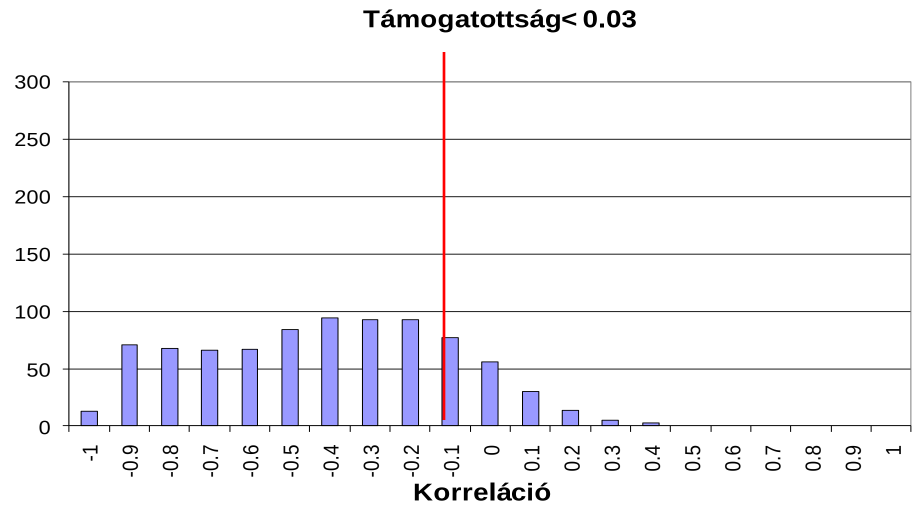
- A támogatottság alapú törlés hatását a tételcsoportok korrelációján vizsgáljuk.
  - Generáljunk 10000 véletlen kontingencia táblát.
  - Számoljuk ki minden táblára a támogatottságot és a páronkénti korrelációt.
  - Alkalmazzunk támogatottság alapú törlést és vizsgáljuk meg az eltávolított táblákat.

A támogatottság alapú törlés hatása

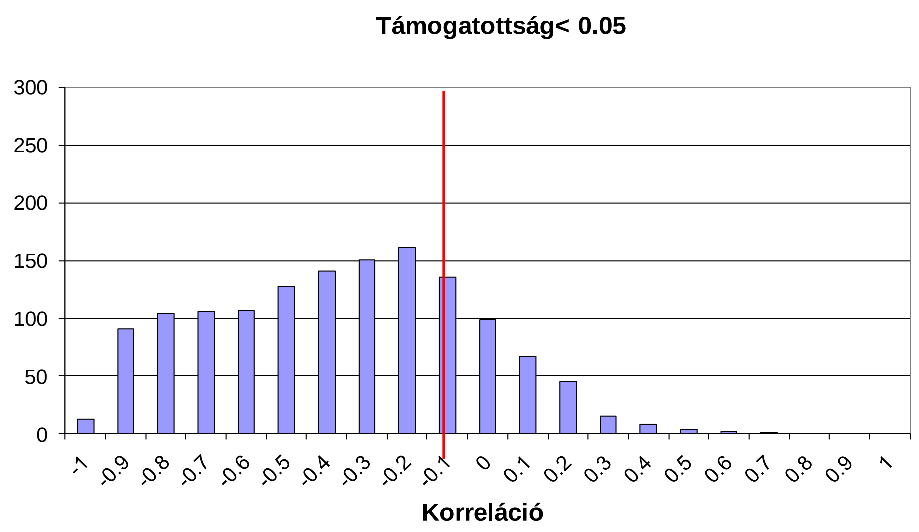


A támogatottság alapú törlés hatása





A támogatottság alapú törlés főként a negatívan korrelált tételcsoportokat távolítja el.



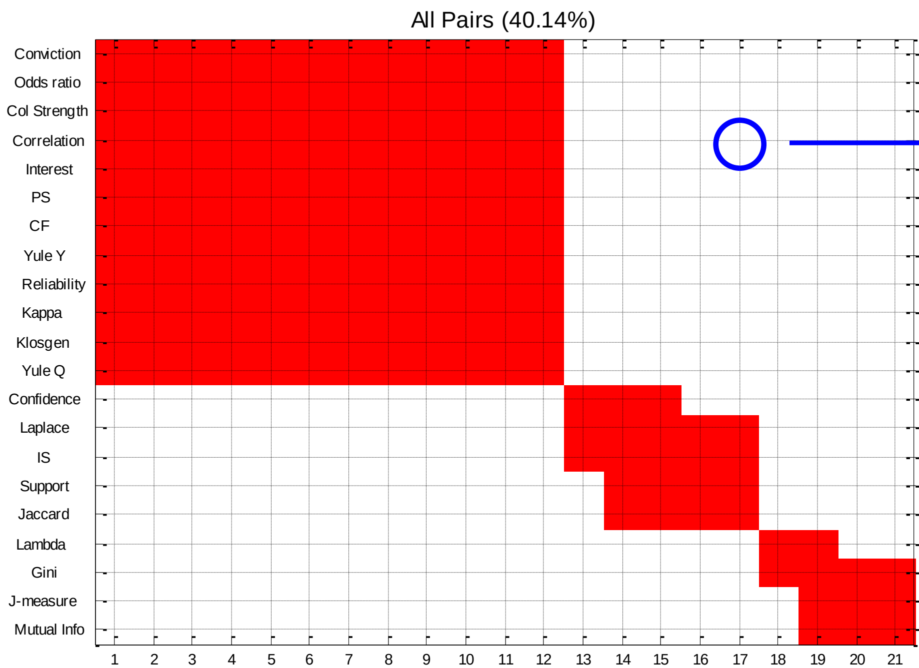
A támogatottság alapú törlés hatása

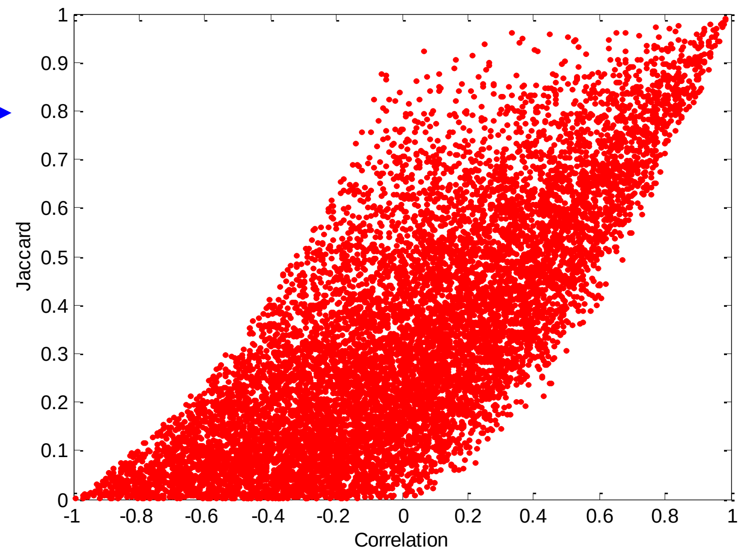
- Vizsgáljuk meg milyen hatása van a támogatott-ság alapú törlésnek más mértékekre.

- Lépések:
  - Generáljunk 10000 véletlen kontingencia táblát.
  - Rangsoroljuk a táblákat a különböző mértékek szerint.
  - Számoljuk ki a páronkénti korrelációt a mértékek között.

A támogatottság alapú törlés hatása

- Támogatottság alapú törlés nélkül (az összes pár).





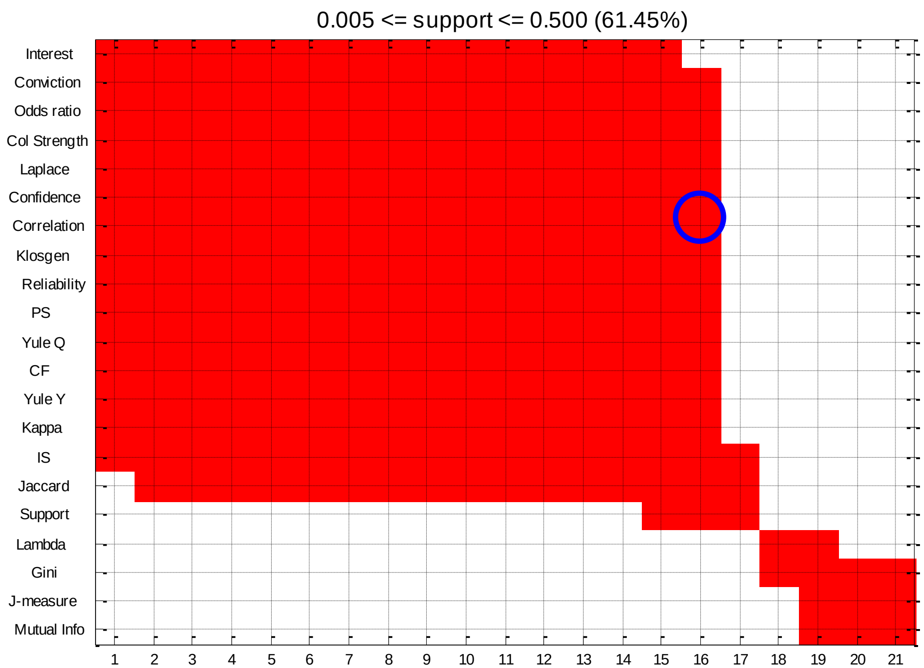
A korreláció és a Jaccard mérték közötti pontdiagram

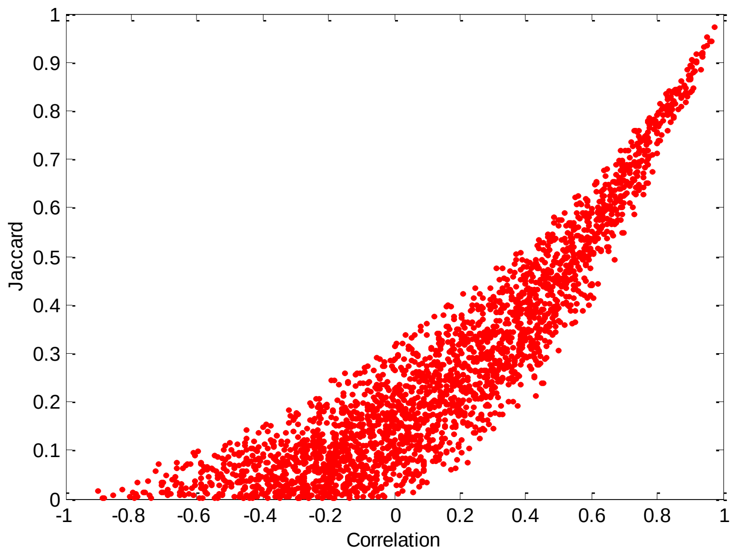
- A vörös cellák jelölik azon mértékpárok közötti korrelációkat, melyek > 0.85

- A párok 40.14%-nak a korrelációja > 0.85

A támogatottság alapú törlés hatása

- $0.5\% \leq$ támogatottság $\leq 50\%$



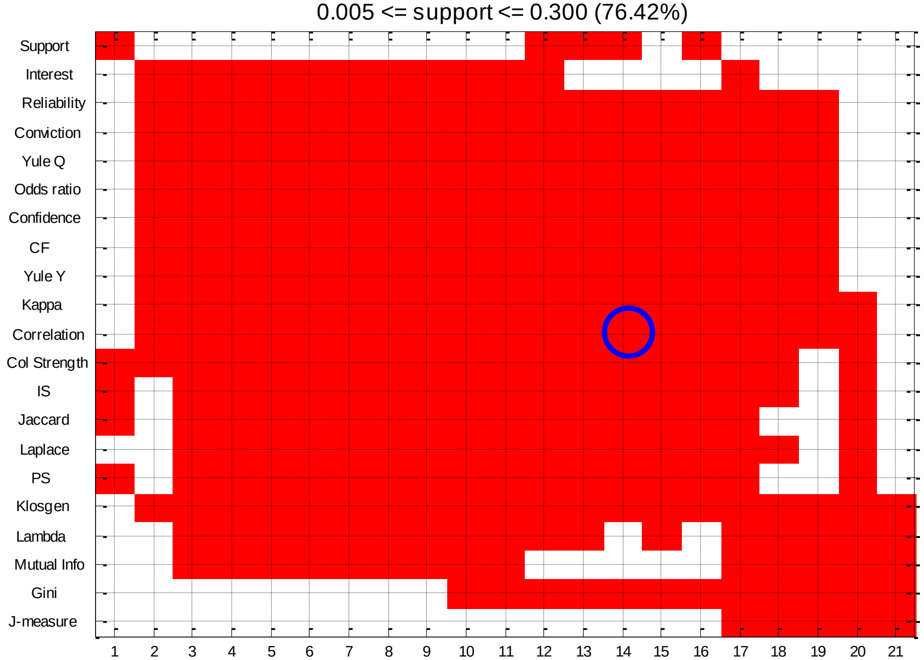


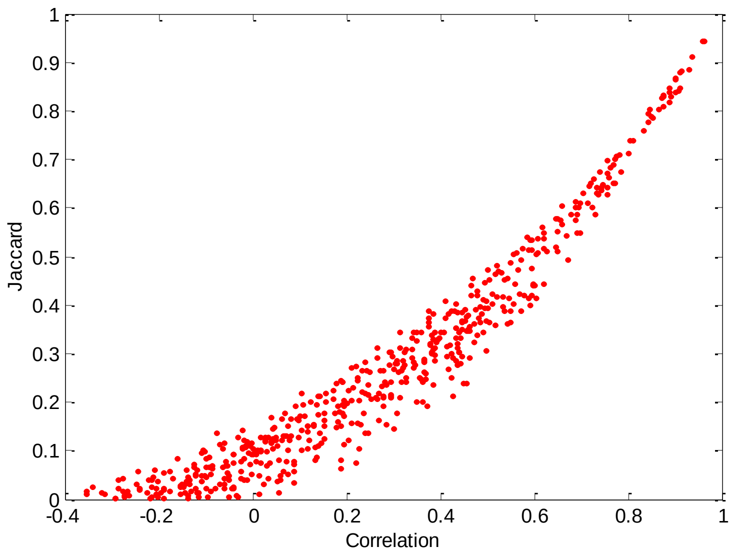
A korreláció és a Jaccard mérték közötti pontdiagram

- A párok 61.45%-ának a korrelációja > 0.85

A támogatottság alapú törlés hatása

- $0.5\% \leq$ támogatottság $\leq 30\%$





A korreláció és a Jaccard mérték közötti pontdiagram

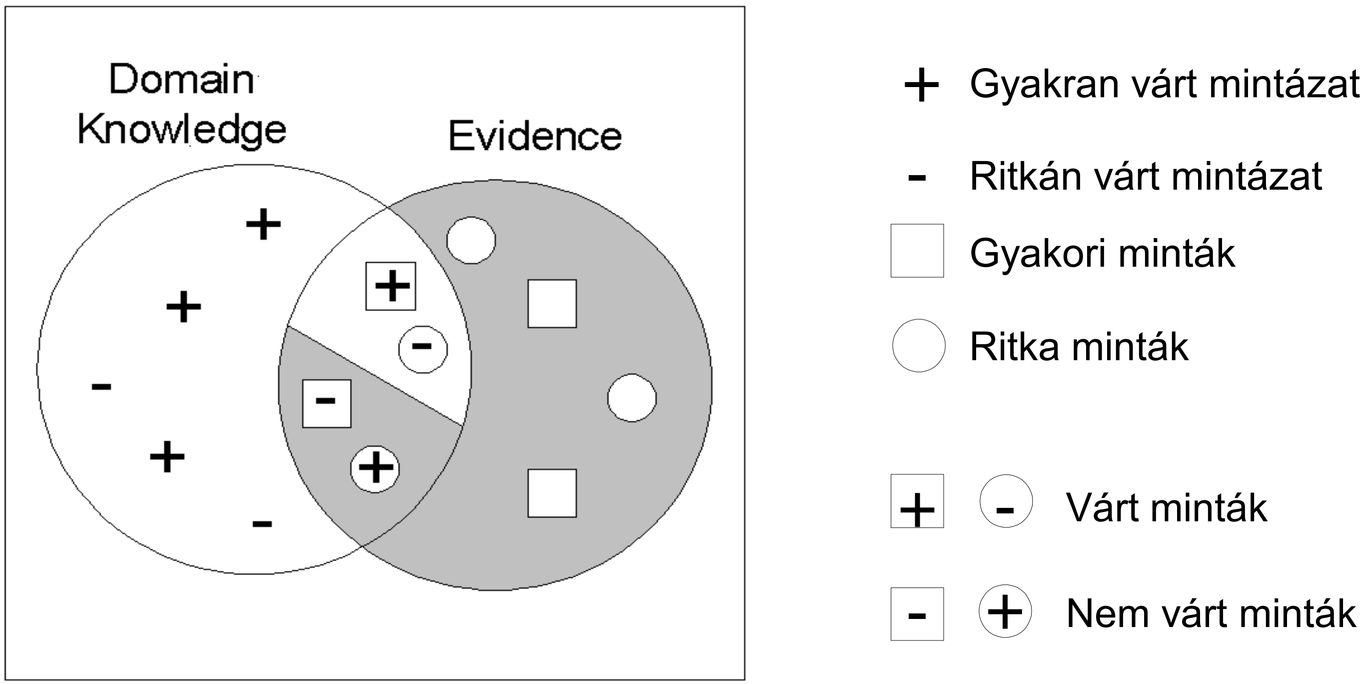
- A párok 76.42%-ának a korrelációja > 0.85

Szubjektív érdekességi mértékek

- Objektív mérték:
  - A mintázatok rangsorolása az adatokból számolt statisztikákon alapszik.
  - Pl. a 21 asszociációs mérték (támogatottság, megbíz-hatóság, Laplace, Gini, kölcsönös információ, Jaccard stb.).
- Szubjektív mérték:
  - A mintázatok rangsorolása a felhasználó értelmezésén alapszik.
    - Egy mintázat szubjektíven érdekes ha ellentmond a felhasználó várakozásának (Silberschatz & Tuzhilin).
    - Egy mintázat szubjektíven érdekes ha cselekvésre ösztönöz (Silberschatz & Tuzhilin).

Érdekesség váratlanság nyomán

- A felhasználók várakozásait kell modellezni (szakterületi tudás).



- A felhasználók várakozásait kell kombinálni az adatokból jövő bizonyossággal (pl. kinyert mintázat).

Érdekesség váratlanság nyomán

- Web adatok (Cooley et al. 2001)
  - Szakterületi tudás a honlap szerkezetében.
  - Adott F = $\{X_1, X_2, \ldots, X_k\}$ tételcsoport (X_i : Web lapok)
    - L: az oldalakhoz kapcsolódó linkek száma
    - lfactor = L / (k $\times$ k-1)
    - cfactor = 1 (ha a gráf összefüggő), 0 (nem összefüggő gráf)
  - Stukturális bizonyosság = cfactor $\times$ lfactor
  - Használati bizonyosság $= \dfrac{P(X_1 \cap X_2 \cap \ldots \cap X_k)}{P(X_1 \cup X_2 \cup \ldots \cup X_k)}$
  - Használjuk a Dempster-Shafer elméletet a szakterületi tudás és az adatokból származó bizonyosság kombinálására.