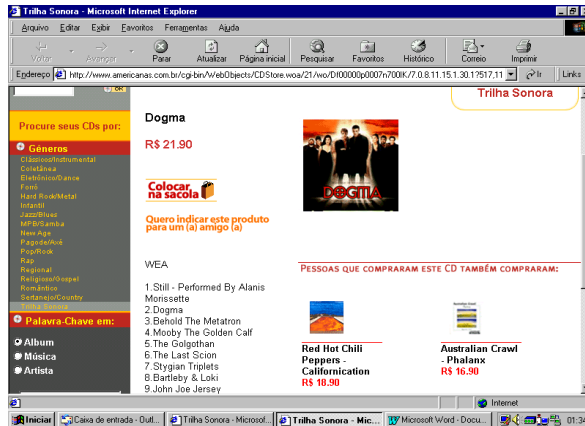


APRENDIZADO DE MÁQUINA - AM

Regras de Associação

Exemplo: vendas casadas

Sei que quem compra A também compra B.

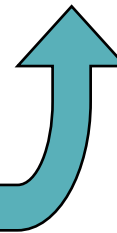


**Compra de
produto**

PRODUTO A

PRODUTO A

PRODUTO B



**Oferta de
produto relacionado**

Amazon.com

a: quarta-feira, 10 de maio de 2006 08:04

a: alvares@inf.ufrgs.br

unto: "Multi-Agent-Based Simulation VI : International Workshop, MABS 2005, Utrecht, The Netherlands, July 25, 2005, Revised and Invited Papers (Lecture Notes ... / Lecture Notes in Artificial Intelligence)"

amazonwire **PODCAST**

Interviews and Exclusives--Books, Music, Movies,
and Those Who Create Them [▶ Learn more](#)

amazon.com
and you're done.™

Books

Dear Amazon.com Customer,

We've noticed that customers who have purchased [Agents Breaking Away : 7th European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW '96, Eindhoven, The Netherlands, January 22 ... / Lecture Notes in Artificial Intelligence](#) by Walter van de Velde also purchased books by Jaime S. Sichman. For this reason, you might like to know that Jaime S. Sichman's *Multi-Agent-Based Simulation VI : International Workshop, MABS 2005, Utrecht, The Netherlands, July 25, 2005, Revised and Invited Papers (Lecture Notes ... / Lecture Notes in Artificial Intelligence)* will be released in paperback soon. You can pre-order your copy by following the link below.



[Multi-Agent-Based Simulation VI :
International Workshop, MABS
2005, Utrecht, The Netherlands.](#)

or See more in Books

Mineração de regras de associação

- Dado um conjunto de **transações**, encontre regras para a predição da **ocorrência de itens** baseado na **ocorrência de outros itens** na transação

transações

<i>TID</i>	<i>Items</i>
1	pão, leite
2	pão, fralda, cerveja, ovos
3	leite, fraldas, cerveja, coca
4	pão, leite, fraldas, cerveja
5	pão, leite, fraldas, coca

Exemplos de regras de associação

$\{\text{fraldas}\} \rightarrow \{\text{cerveja}\},$
 $\{\text{leite, pão}\} \rightarrow \{\text{ovos, coca}\},$
 $\{\text{cerveja, pão}\} \rightarrow \{\text{leite}\},$

Implicação significa co-ocorrência, e não causa!!!

Definições: conjuntos de itens frequentes (frequent itemsets)

❑ **Itemset (conjunto de itens)**

- ❑ Um conjunto de um ou mais itens
 - Exemplo: {leite, pão, fralda}
- ❑ k-itemset
 - Um *itemset* com *k* itens

❑ **Suporte (σ)**

- ❑ **Frequência** de ocorrência de um conjunto de itens (*itemset*)
- ❑ Ex: $\sigma(\{\text{leite, pão}\}) = 3$

❑ **Suporte (s)**

- ❑ **Fração das transações** que contêm um *itemset*
- ❑ Ex: $s(\{\text{leite, pão, fralda}\}) = 2/5$

❑ **Conjunto de itens frequentes**

- ❑ Um *itemset* cujo suporte é maior ou igual a um dado limite *minsup*

<i>TID</i>	<i>Items</i>
1	pão, leite
2	pão, fralda, cerveja, ovos
3	leite, fralda, cerveja, coca
4	pão, leite, fralda, cerveja
5	pão, leite, fralda, coca

Definição: regra de associação

- Regras de associação
 - Uma expressão da forma $X \rightarrow Y$, onde X e Y são conjuntos **disjuntos** de itens
 - Exemplo:
 $\{\text{leite, fralda}\} \rightarrow \{\text{cerveja}\}$
(significado: quem compra leite e fralda também compra cerveja na mesma transação)

<i>TID</i>	<i>Items</i>
1	pão, leite
2	pão, fralda, cerveja, ovos
3	leite, fralda, cerveja, coca
4	pão, leite, fralda, cerveja
5	pão, leite, fralda, coca

Exemplo:

$\{\text{leite, fralda}\} \Rightarrow \{\text{cerveja}\}$

- Métricas de avaliação das regras
 - Suporte (s)
 - ◆ Fração das transações que contêm X e Y
 - Confiança (c)
 - ◆ Mede a frequência com que Y aparece nas transações que contêm X

$$s = \frac{\sigma(\text{leite, fralda, cerveja})}{|T|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{leite, fralda, cerveja})}{\sigma(\text{leite, fralda})} = \frac{2}{3} = 0.67$$

Regras de associação

- Regras de associação ou regras associativas têm a forma

$$X \Rightarrow Y$$

- onde X e Y são conjuntos de itens que ocorrem juntos em uma transação e $X \cap Y = \phi$
- significando que se encontrarmos o conjunto de itens X em uma transação, então temos uma boa chance de encontrar também o conjunto de itens Y na mesma transação.

Mineração de regras de associação

- Dado um conjunto de transações T , o objetivo da mineração de regras de associação é encontrar todas as regras com
 - ▣ suporte $\geq \text{minsup}$
 - ▣ confiança $\geq \text{minconf}$
- Abordagem da força bruta:
 - ▣ liste todas as possíveis regras de associação
 - ▣ calcule o suporte e a confiança para cada regra
 - ▣ corte as regras que não satisfazem minsup ou minconf
 - **Computacionalmente proibitivo!**

Problema: número de regras geradas

Considerando 4 itens: A, B, C e D, sem considerar suporte e confiança podemos ter:

<i>conjunto</i>	<i>Regras possíveis</i>	<i>Número de regras</i>
$\{AB\}$	$A \rightarrow B; B \rightarrow A$	2
$\{AC\}$	$A \rightarrow C; C \rightarrow A$	2
$\{AD\}$	$A \rightarrow D; D \rightarrow A$	2
$\{BC\}$	$B \rightarrow C; C \rightarrow B$	2
$\{BD\}$	$B \rightarrow D; D \rightarrow B$	2
$\{CD\}$	$C \rightarrow D; D \rightarrow C$	2
$\{ABC\}$	$A \rightarrow BC; B \rightarrow AC; C \rightarrow AB; BC \rightarrow A; AC \rightarrow B; AB \rightarrow C$	6
$\{ABD\}$	$A \rightarrow BD; B \rightarrow AD; D \rightarrow AB; BD \rightarrow A; AD \rightarrow B; AB \rightarrow D$	6
$\{ACD\}$	$A \rightarrow DC; D \rightarrow AC; C \rightarrow AD; DC \rightarrow A; AC \rightarrow D; AD \rightarrow C$	6
$\{BCD\}$	$D \rightarrow BC; B \rightarrow DC; C \rightarrow DB; BC \rightarrow D; DC \rightarrow B; DB \rightarrow C$	6
$\{ABCD\}$	$A \rightarrow BCD; B \rightarrow ACD; C \rightarrow ABD; D \rightarrow ABC; AB \rightarrow CD; AC \rightarrow BD; AD \rightarrow BC; BC \rightarrow AD; BD \rightarrow AC; CD \rightarrow AB; BCD \rightarrow A; ACD \rightarrow B; ABD \rightarrow C; ABC \rightarrow D;$	14
		50

Minerando regras de associação

TID	Items
1	pão, leite
2	pão, fralda, cerveja, ovos
3	leite, fralda, cerveja, coca
4	pão, leite, fralda, cerveja
5	pão, leite, fralda, coca

Exemplos de regras:

$\{\text{leite, fralda}\} \rightarrow \{\text{cerveja}\}$ ($s=0.4$, $c=0.67$)

$\{\text{leite, cerveja}\} \rightarrow \{\text{fralda}\}$ ($s=0.4$, $c=1.0$)

$\{\text{fralda, cerveja}\} \rightarrow \{\text{leite}\}$ ($s=0.4$, $c=0.67$)

$\{\text{cerveja}\} \rightarrow \{\text{leite, fralda}\}$ ($s=0.4$, $c=0.67$)

$\{\text{fralda}\} \rightarrow \{\text{leite, cerveja}\}$ ($s=0.4$, $c=0.5$)

$\{\text{leite}\} \rightarrow \{\text{fralda, cerveja}\}$ ($s=0.4$, $c=0.5$)

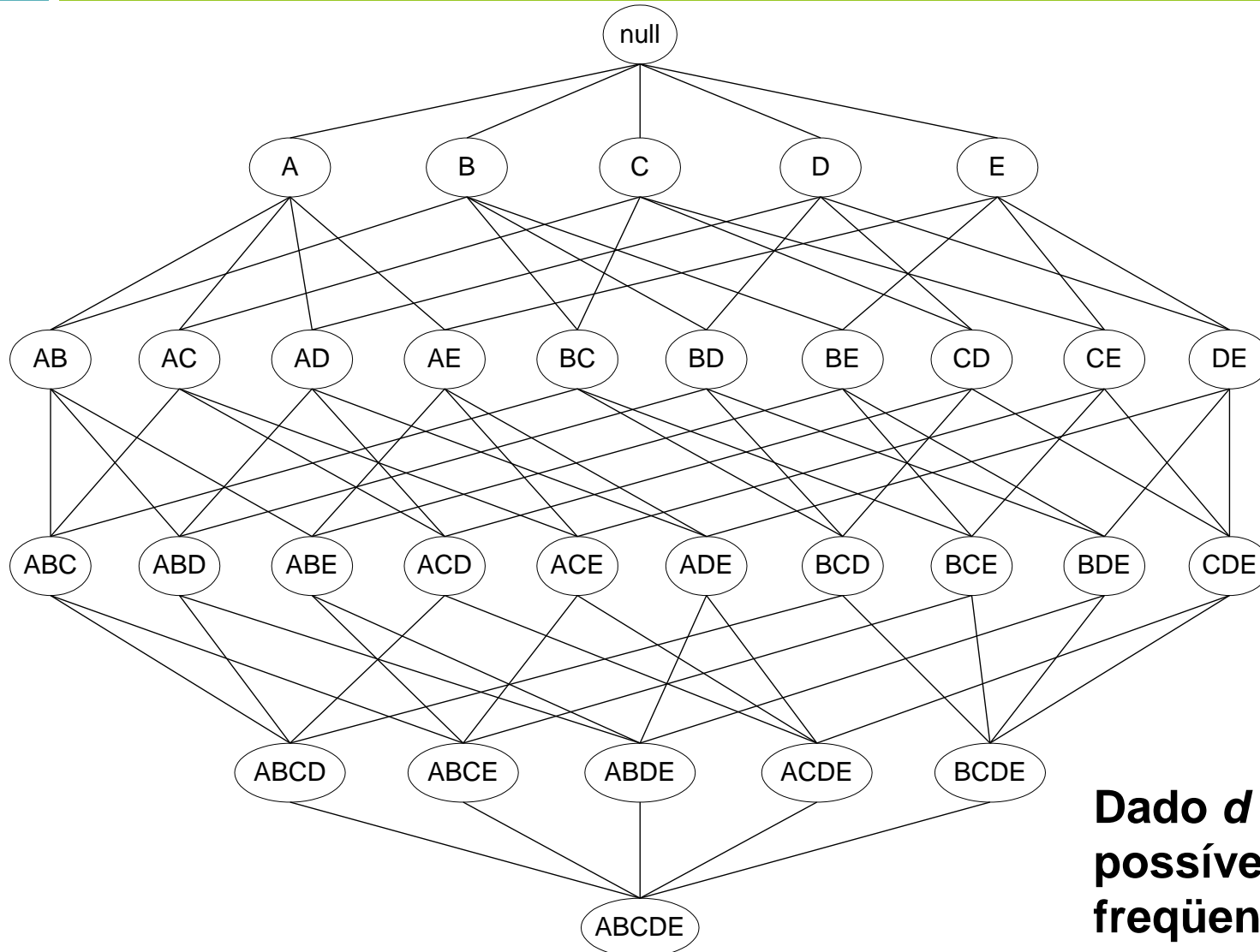
□ Observações:

- Todas as regras acima são partições binárias do mesmo itemset:
 $\{\text{leite, fralda, cerveja}\}$
- Regras originadas do mesmo itemset têm o mesmo suporte mas podem ter confianças diferentes
- Então, podemos separar o suporte da confiança

Mineração de regras de associação

- Abordagem em dois passos:
 - Geração dos itens frequentes
 - gerar todos os itemsets com suporte \geq minsup
 - Geração das regras
 - gerar regras de alta confiança para cada itemset, onde cada regra é uma partição binária de um itemset frequente
- A geração dos conjuntos de itens frequentes ainda é computacionalmente custosa

Mineração de regras de associação

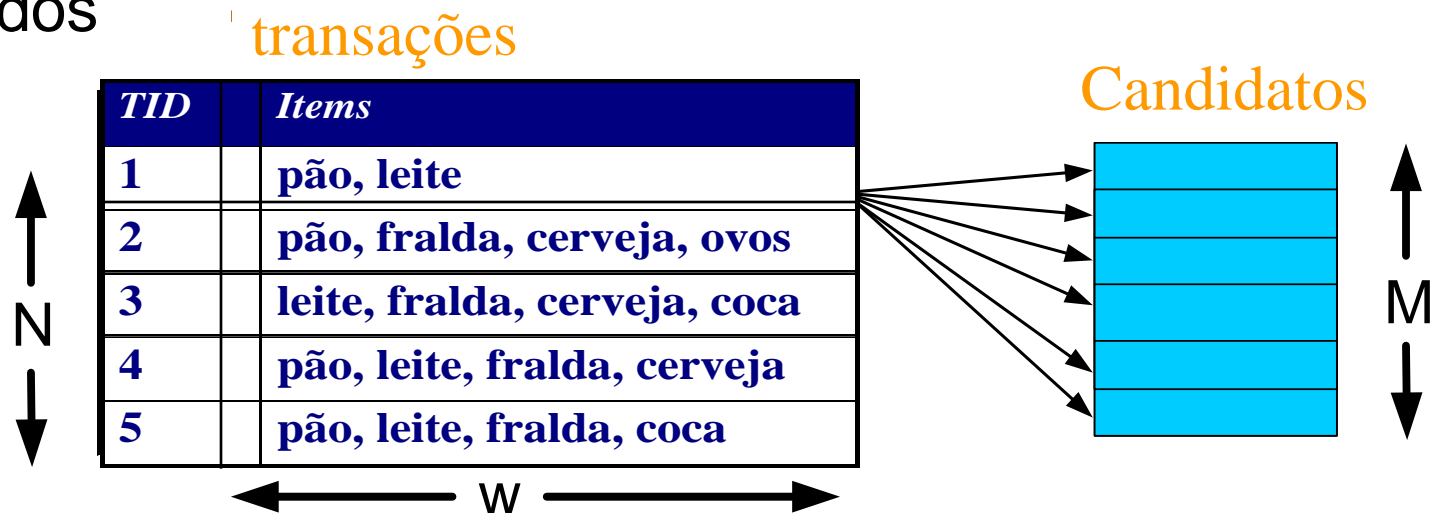


Dado d items, há 2^d possíveis itemsets frequentes

Geração de *itemsets* freqüentes

❑ Abordagem de força bruta:

- ❑ Cada *itemset* no reticulado (lattice) é um conjunto freqüente **candidato**
- ❑ Calcule o suporte de cada candidato lendo o conjunto de dados



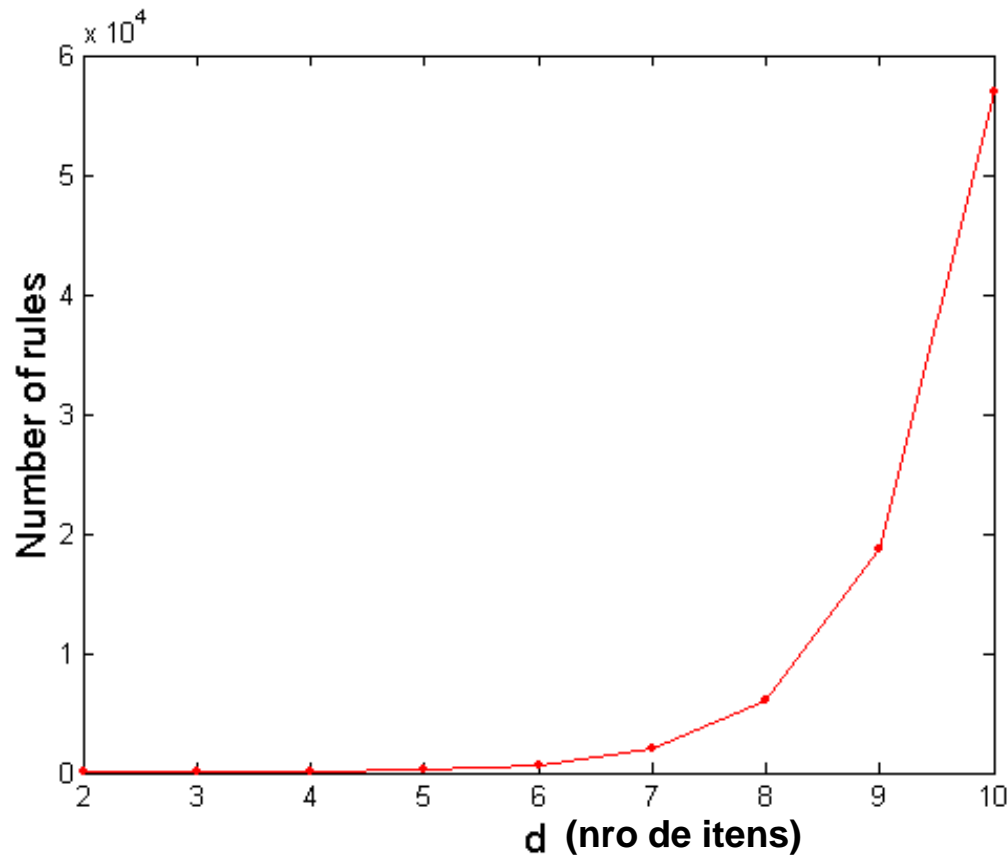
- ❑ Complexidade $\sim O(NMw) \Rightarrow$ **Custoso pois $M = 2^d$!!!**

Complexidade

□ Dado d items:

□ número total de itemsets = 2^d

□ número total possível de regras de associação:



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

se $d=6$, $R = 602$ regras

se $d=10$, $R= 57.002$ regras

Estratégias para a geração de itemsets frequentes

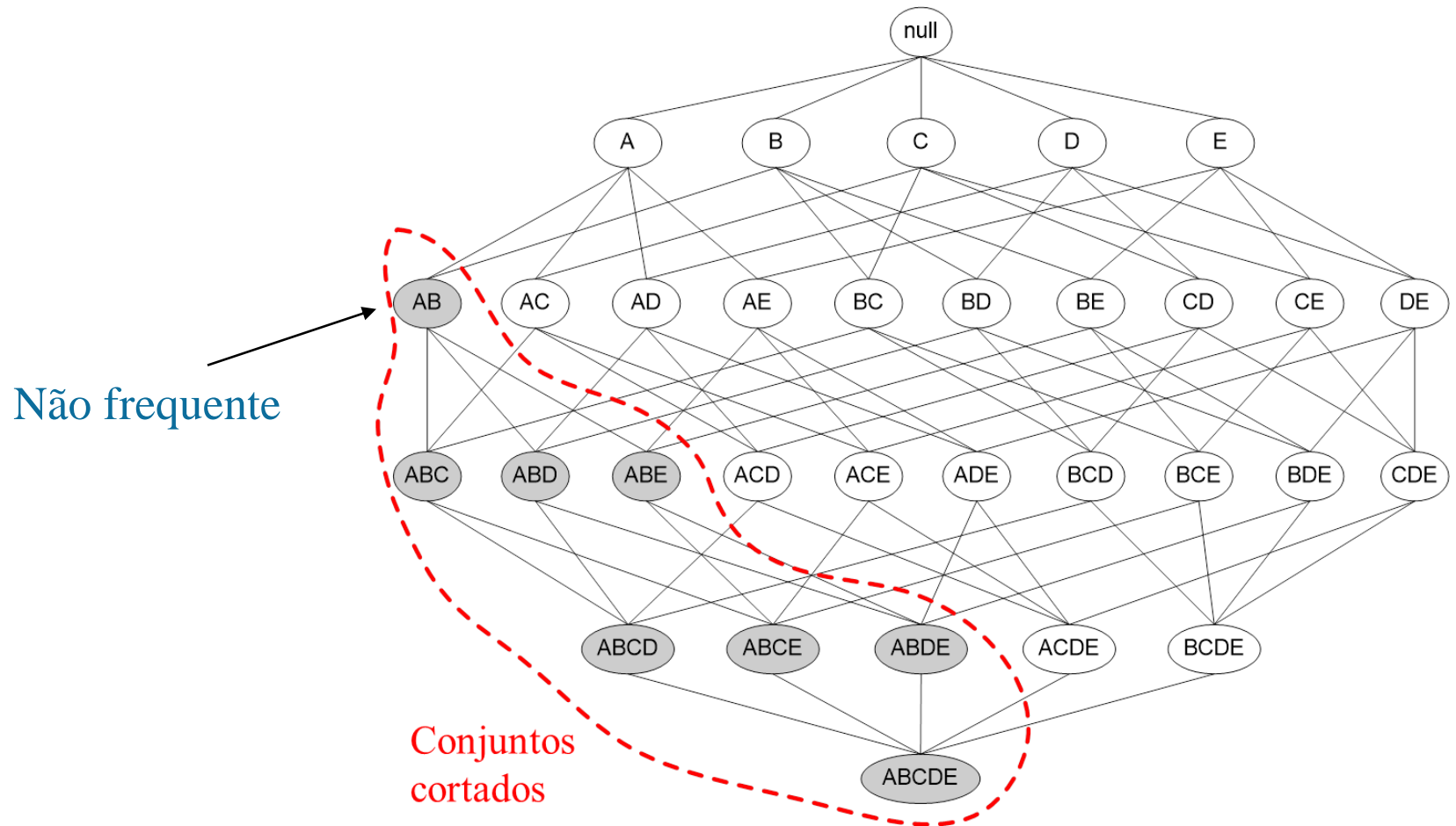
- Reduzir o **número de candidatos** (M)
 - Busca completa: $M=2^d$
 - Usar técnicas de poda para reduzir M
- Reduzir o **número de transações** (N)
 - Reduzir o tamanho de N quando o número de itemsets aumenta
 - Usado pelo DHP e algoritmos baseados em mineração vertical
- Reduzir o **número de comparações** (NM)
 - Usar estruturas de dados eficientes para armazenar os candidatos ou as transações
 - Sem necessidade de comparar cada candidato com cada transação

Reduzindo o número de candidatos

- **Princípio do algoritmo Apriori :**
 - Se um itemset é freqüente então todos os seus subconjuntos também são freqüentes
- Este princípio é devido a seguinte propriedade do suporte:
 - O suporte de um itemset nunca é maior que o suporte de seus subconjuntos
 - Isto é conhecido como a propriedade anti-monotônica do suporte

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

Ilustrando o princípio do Apriori



Ilustrando o princípio do Apriori

Item	Count
pão	4
coca	2
leite	4
cerveja	3
fralda	4
ovos	1

Items (1-itemsets)

Itemset	Count
{pão,leite}	3
{pão,cerveja}	2
{pão,fralda}	3
{leite,cerveja}	2
{leite,fralda}	3
{cerveja,fralda}	3

Pares (2-itemsets)

(Não há necessidade de Gerar candidatos com **coca** ou **ovos**)

Suporte mínimo= 3

Triplas (3-itemsets)

Itemset	Count
{pão,leite,fralda}	3

Se todos os conjuntos são considerados,

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

Com o corte baseado no suporte,

$$6 + 6 + 1 = 13$$

O algoritmo Apriori

- Método:
 - seja $k=1$
 - Obtenha conjuntos frequentes de tamanho 1
 - Repita enquanto novos itemsets frequentes forem obtidos
 - Obtenha itemsets candidatos de tamanho $(k+1)$ a partir de itemsets de tamanho k (*não inclua itemsets candidatos contendo subconjuntos de tamanho k infrequentes*)
 - Conte o suporte de cada candidato varrendo o BD
 - Elimine candidatos não frequentes, deixando só os frequentes

Algoritmo Apriori

- (1) Dado um limiar de suporte minsup, no primeiro passo encontre os itens que aparecem ao menos numa fração das transações igual a minsup. Este conjunto é chamado **L1**, dos itens frequentes.
- (2) Os pares dos itens em **L1** se tornam pares candidatos **C2** para o segundo passo. Os pares em **C2** cuja contagem alcançar minsup são os pares frequentes **L2**.
- (3) As trincas candidatas C3 são aqueles conjuntos {A, B, C} tais que todos os {A, B}, {A, C} e {B, C} estão em **L2**. No terceiro passo, conte a ocorrência das trincas em **C3**; aquelas cuja contagem alcançar minsup são as trincas frequentes, **L3**.
- (4) Proceda da mesma forma para tuplas de ordem mais elevada, até os conjuntos se tornarem vazios. **Li** são os conjuntos frequentes de tamanho i; **C_{i+1}** é o conjunto de tamanho i+1 tal que cada subconjunto de tamanho i está em **Li**.

Exemplo de descoberta de regras de associação

- Dada a tabela abaixo onde cada registro corresponde a uma transação de um cliente, com itens assumindo valores binários (sim/não), descobrir todas as regras associativas com suporte $\geq 0,3$ e grau de certeza (confiança) $\geq 0,8$.

TID	leite	café	cerveja	pão	manteiga	arroz	feijão
1	não	sim	não	sim	sim	não	não
2	sim	não	sim	sim	sim	não	não
3	não	sim	não	sim	sim	não	não
4	sim	sim	não	sim	sim	não	não
5	não	não	sim	não	não	não	não
6	não	não	não	não	sim	não	não
7	não	não	não	sim	não	não	não
8	não	não	não	não	não	não	sim
9	não	não	não	não	não	sim	sim
10	não	não	não	não	não	sim	não

Exemplo de descoberta de regras de associação

- Dada uma regra de associação “Se compra X então compra Y”, os fatores sup e conf são:

$$sup = \frac{\text{Número de registros com X e Y}}{\text{Número total de registros}} \quad \quad \quad conf = \frac{\text{Número de registros com X e Y}}{\text{Número de registros com X}}$$

- (1) Calcular o suporte de conjuntos com um item.

Determinar os itens frequentes com $sup \geq 0,3$.

- (2) Calcular o suporte de conjuntos com dois itens.

Determinar conjuntos de itens frequentes com $sup \geq 0,3$.

Obs: se um item não é frequente em (1), pode ser ignorado aqui.

Descobrir as regras com alto fator de certeza.

- (3) Calcular o suporte de conjuntos com três itens.

Determinar conjuntos de itens frequentes com $sup \geq 0,3$.

Obs: pelo mesmo motivo anterior, só é necessário se considerar conjuntos de itens que são frequentes pelo passo anterior.

Exemplo de descoberta de regras de associação

C_1

Conjunto de itens	suporte
{leite}	2
{café}	3
{cerveja}	2
{pão}	5
{manteiga}	5
{arroz}	2
{feijão}	2

L_1

Conjunto de itens	suporte
{café}	3
{pão}	5
{manteiga}	5

Exemplo de descoberta de regras de associação

C_2, L_2

Conjunto de itens	suporte
{café, pão}	3
{café, manteiga}	3
{pão, manteiga}	4

C_3, L_3

Conjunto de itens	suporte
{café, pão, manteiga}	3

Exemplo de descoberta de regras de associação

- Regras candidatas com dois itens com o seu valor de certeza:

- Conjunto de itens: {café, pão}

- Se **café** Então **pão** $\text{conf} = 1,0$
- Se **pão** Então **café** $\text{conf} = 0,6$

- Conjunto de itens: {café, manteiga}

- Se **café** Então **manteiga** $\text{conf} = 1,0$
- Se **manteiga** Então **café** $\text{conf} = 0,6$

- Conjunto de itens: {pão, manteiga}

- Se **pão** Então **manteiga** $\text{conf} = 0,8$
- Se **manteiga** Então **pão** $\text{conf} = 0,8$

Conjunto de itens	suporte
{café, pão}	3
{café, manteiga}	3
{pão, manteiga}	4

Exemplo de descoberta de regras de associação

- Regras candidatas com três itens com o seu valor de certeza:

- Conjunto de itens: {café, manteiga, pão}

- Se **café, manteiga** Então **pão** conf = 1,0
- Se **café, pão** Então **manteiga** conf = 1,0
- Se **manteiga, pão** Então **café** conf = 0,75
- Se **café** Então **manteiga, pão** conf = 1,0
- Se **manteiga** Então **café, pão** conf = 0,6
- Se **pão** Então **café, manteiga** conf = 0,6

Conjunto de itens	suporte
{café, pão, manteiga}	3

- Padrões descobertos, **minsup = 0,3** e **minconf = 0,8**:

- Se **café** Então **pão** conf = 1,0
- Se **café** Então **manteiga** conf = 1,0
- Se **pão** Então **manteiga** conf = 0,8
- Se **manteiga** Então **pão** conf = 0,8
- Se **café, manteiga** Então **pão** conf = 1,0
- Se **café, pão** Então **manteiga** conf = 1,0
- Se **café** Então **manteiga, pão** conf = 1,0

Exercício

- Considere a tabela de transações abaixo:

Tid	Itens comprados
1	A, C, D
2	B, C, E
3	A, B, C, E
4	B, E
5	A, B, C, E

- Quais são os conjuntos frequentes, considerando 50% como suporte mínimo?
- Qual a confiança da regra $B \rightarrow CE$?

LIFT

- Suporte e confiança são usados como filtros, para diminuir o número de regras geradas, gerando apenas regras de melhor qualidade
 - mas, se considerarmos a regra
 - **Se A então B** com confiança de 90%
 - podemos garantir que seja uma regra interessante?

LIFT

- a regra (1) Se A então B com confiança de 90%
 - NÃO é interessante se B aparece em cerca de 90% das transações, pois a regra não acrescentou nada em termos de conhecimento.

- já a regra (2): Se C então D com confiança de 70%
 - é muito mais importante se D aparece, digamos, em 10% das transações.

- lift = confiança da regra / suporte do consequente
 - lift da regra (1) = $0,9 / 0,9 = 1$
 - lift da regra (2) = $0,7 / 0,1 = 7$

Challenges of frequent itemset mining

□ Challenges

- Multiple scans of transaction database
- Huge number of candidates
- Tedious workload of support counting of candidates

□ Improving Apriori: general ideas

- Reduce passes of transaction database scans
- Shrink number of candidates
- Facilitate support counting of candidates

Regras Redundantes

Tid	Itemset
1	A, C, D, T, W
2	C, D, W
3	A, D, T, W
4	A, C, D, W
5	A, C, D, T, W
6	C, D, T

$$A \rightarrow W \quad s=4/6 \quad c=4/4$$

$$A \rightarrow D, W \quad s=4/6 \quad c=4/4$$

TidSet	Frequent sets L
123456	{D}
12456	{C}, {C,D}
12345	{W}, {D,W}
1245	{C,W}, {C,D,W}
1345	{A}, {A,D}, {A,W}, {A,D,W}
1356	{T}, {D,T}
145	{A,C}, {A,C,W}, {A,C,D}, {A,C,D,W}
135	{A,T}, {T,W}, {A,D,T}, {A,T,W}, {D,T,W}, {A,D,T,W}
156	{C,T}, {C,D,T}

Compact representation of frequent itemsets

- Some itemsets are redundant because they have identical support as their supersets

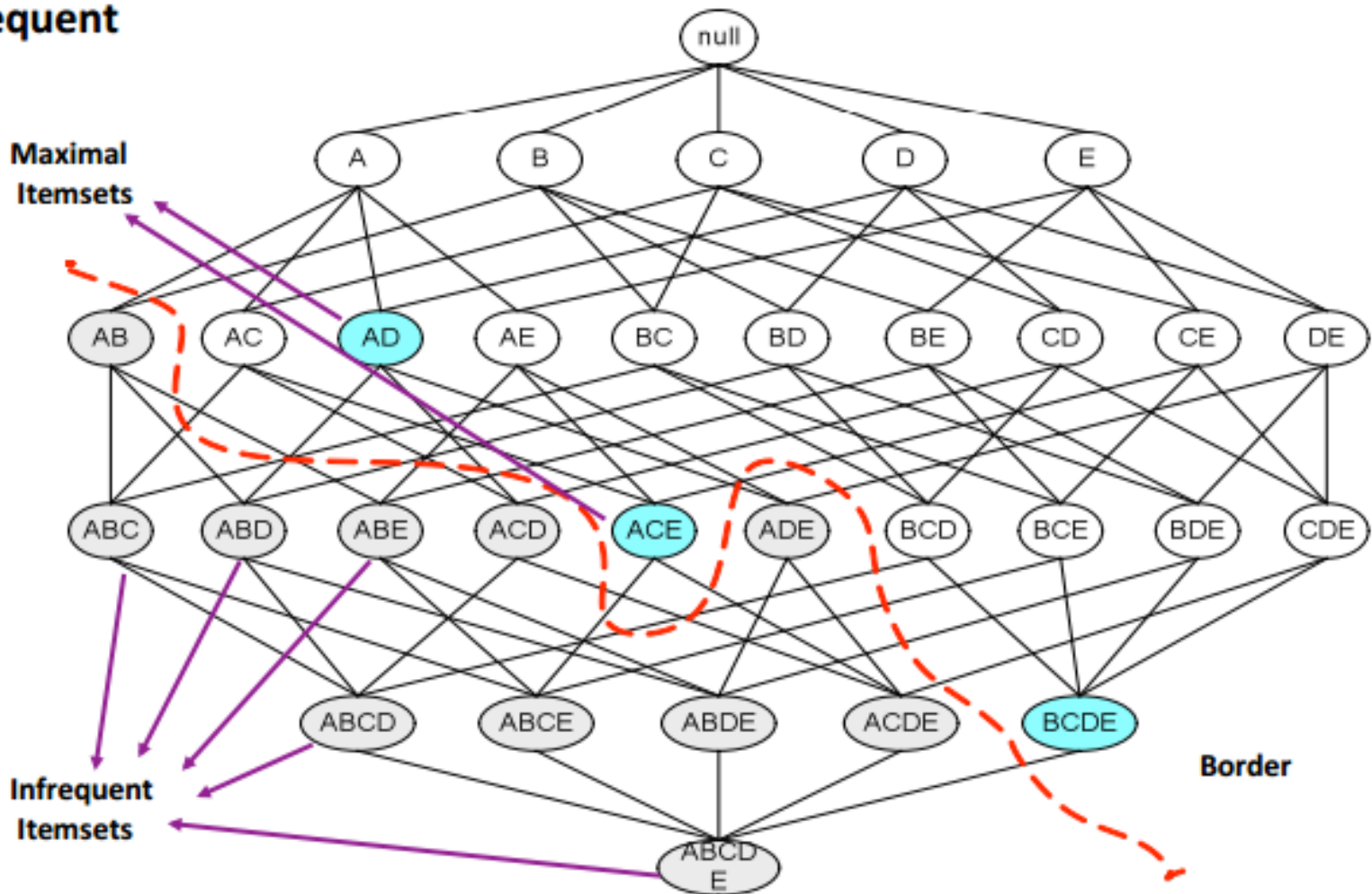
TID	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	1	1	1	1	1	1	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	1	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	1	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	1	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	1	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	1	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	0	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	0	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	0	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	0	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	0	1	1	1	1	1	1	1	1	1

- Number of frequent itemsets
- Need a compact representation

$$= 3 \times \sum_{k=1}^{10} \binom{10}{k}$$

Maximal frequent itemset

An itemset is maximal frequent if none of its immediate supersets is frequent



Closed Itemset

- An itemset is closed if none of its immediate supersets has the same support as the itemset

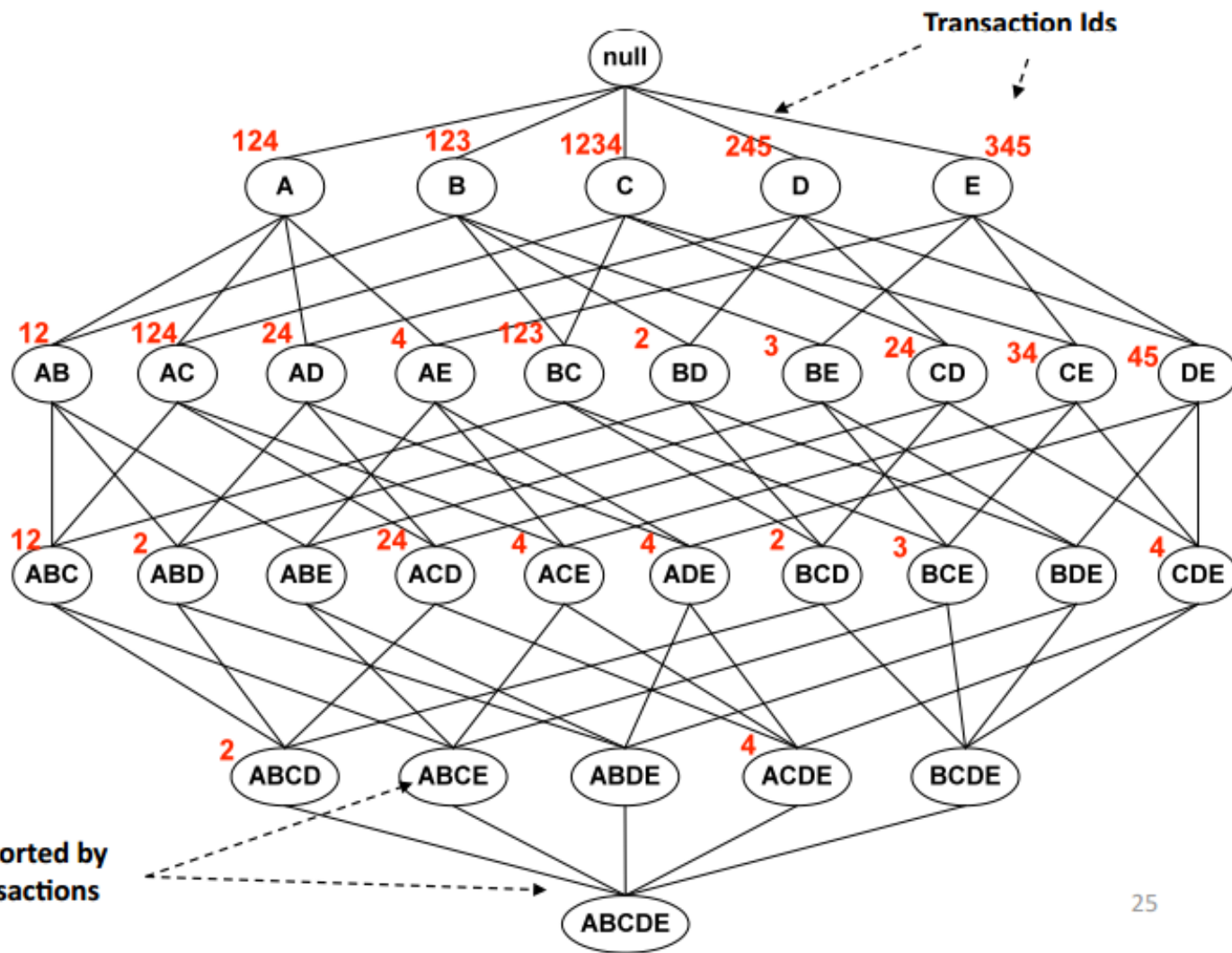
TID	Items
1	{A,B}
2	{B,C,D}
3	{A,B,C,D}
4	{A,B,D}
5	{A,B,C,D}

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

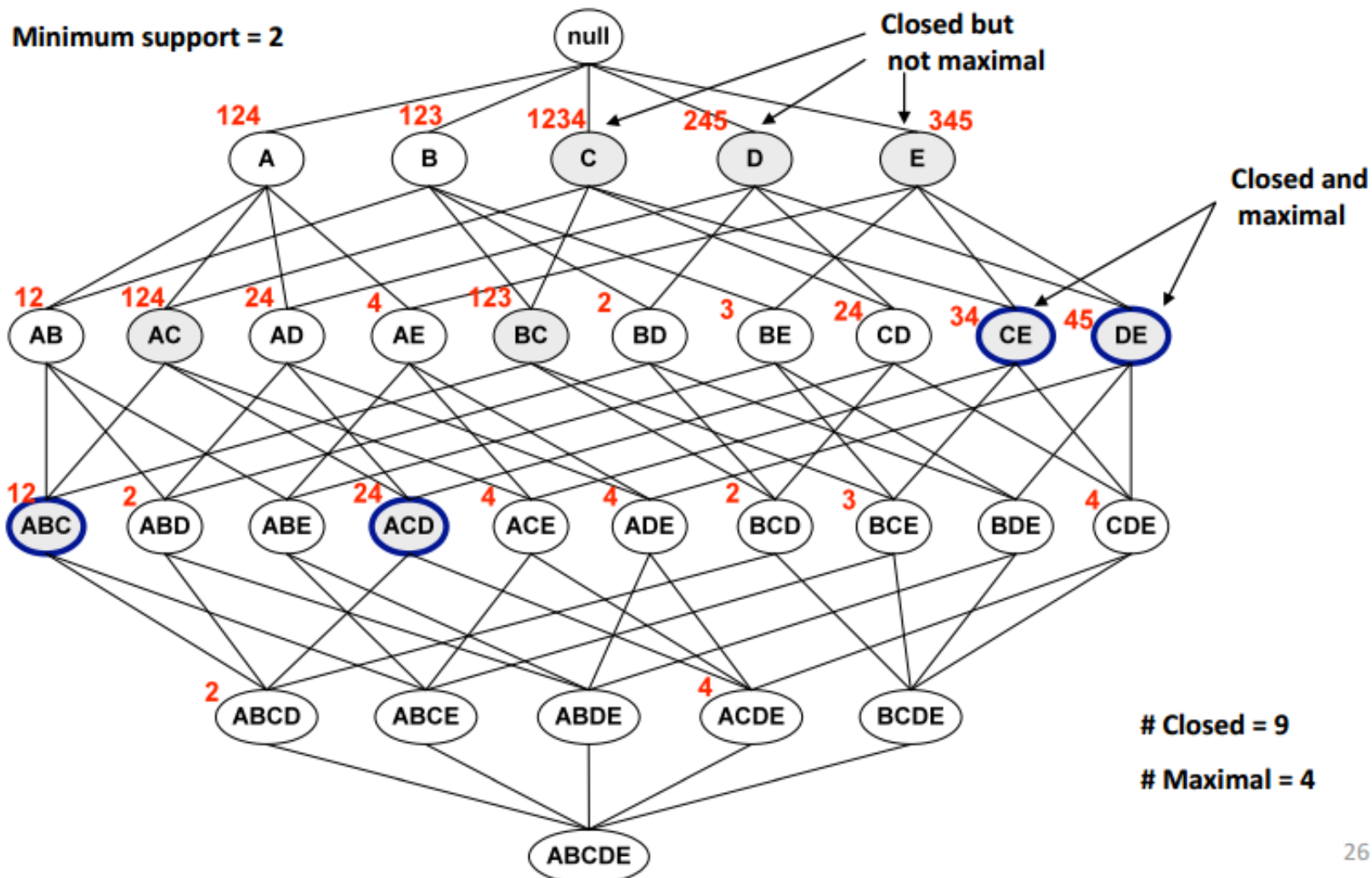
Itemset	Support
{A,B,C}	2
{A,B,D}	3
{A,C,D}	2
{B,C,D}	3
{A,B,C,D}	2

Maximal x Closed itemsets

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

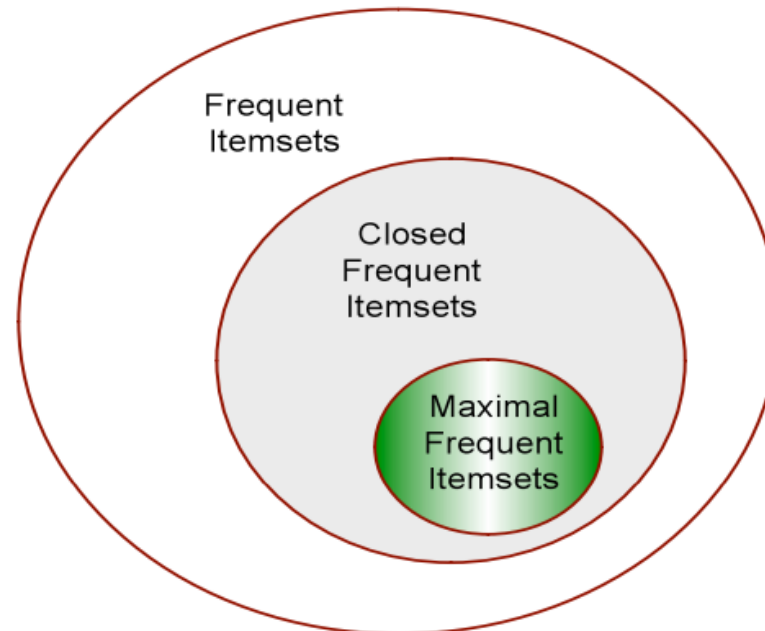


Maximal x Closed itemsets



Relationship between Frequent Itemset Representations

- It is important to point out the relationship between frequent itemsets, closed frequent itemsets and maximal frequent itemsets.
 - Closed frequent itemsets are more widely used than maximal frequent itemset because when efficiency is more important than space



Exercício

- Considere a tabela de transações abaixo:

Tid	Itens comprados
1	A, C, D
2	B, C, E
3	A, B, C, E
4	B, E
5	A, B, C, E

- Quais são os conjuntos frequentes fechados, considerando 50% como suporte mínimo?

Alternative methods for frequent itemset generation

- Representation of Database
 - horizontal vs vertical data layout

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				

ECLAT

- For each item, store a list of transaction ids (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-list

FP-growth Algorithm

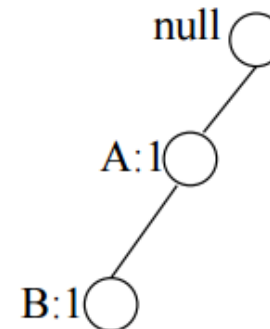


- Use a compressed representation of the database using an **FP-tree**
- Once an FP-tree has been constructed, it uses a recursive divide-and-conquer approach to mine the frequent itemsets

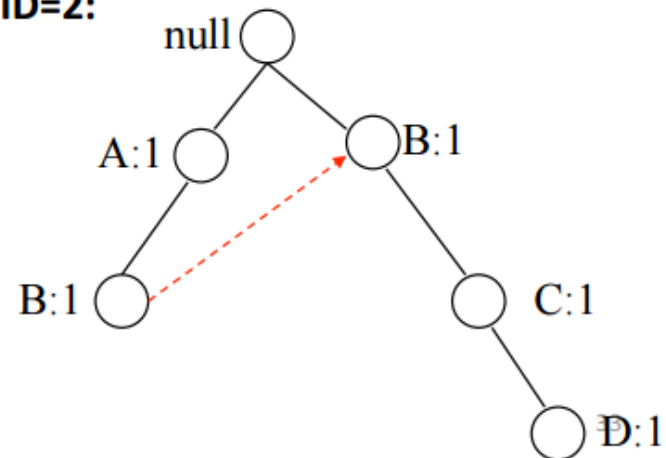
FP-tree construction

TID	Items
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}

After reading TID=1:



After reading TID=2:



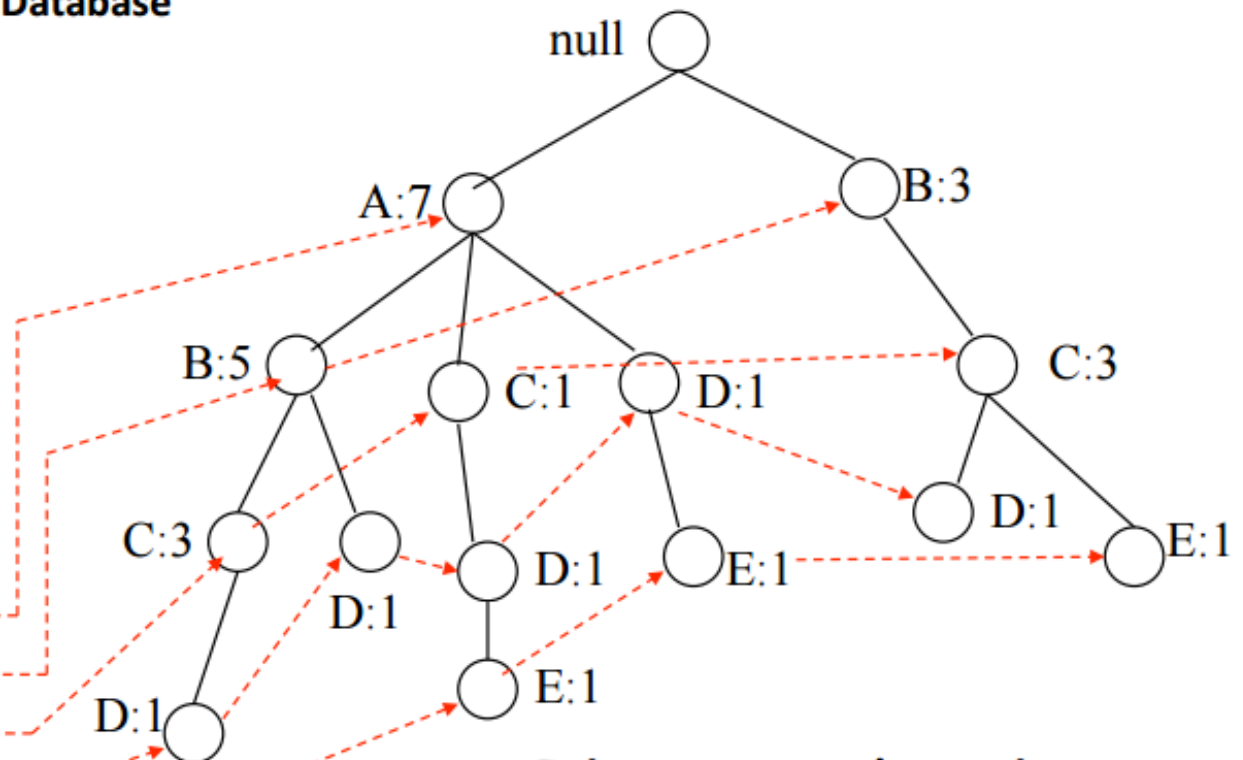
FP-tree construction

TID	Items
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}

Transaction Database

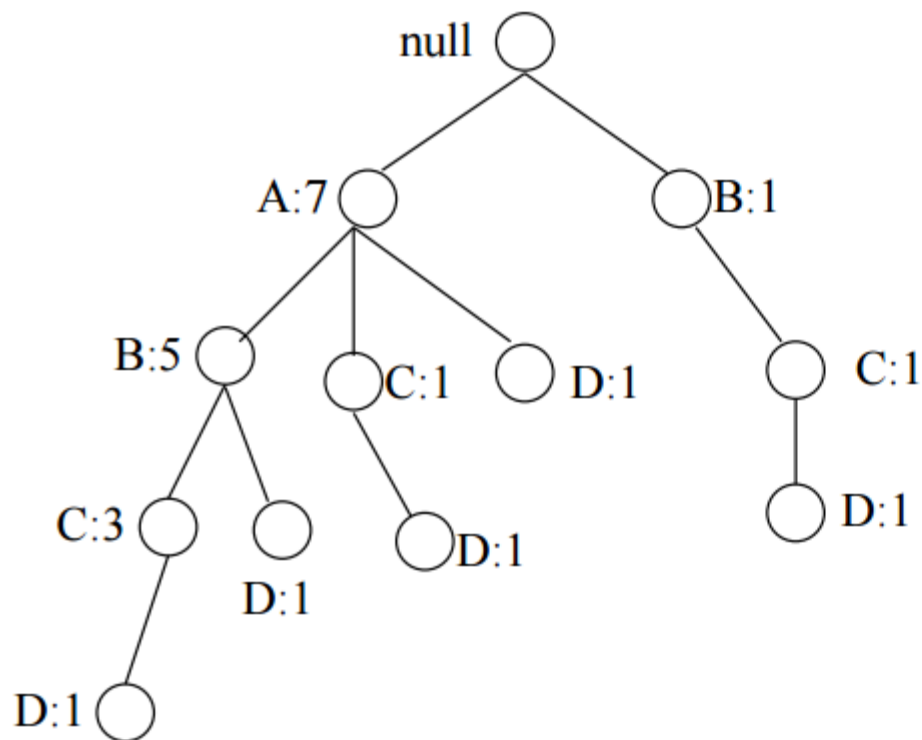
Header table

Item	Pointer
A	
B	
C	
D	
E	



Pointers are used to assist frequent itemset generation

FP-growth



Conditional Pattern base for D:

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

Recursively apply FP-growth on P

Frequent Itemsets found (with sup > 1):

AD, BD, CD, ACD, BCD

Bibliografia

- TAN, P-N,; STEINBACH, M; KUMAR, V. **Introduction to Data Mining**, Boston, Addison Wesley, 2006
- AGRAWAL, R.; IMIELINSKI, T.; SWAMI, A. **Mining association rules between sets of items in large databases**. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, SIGMOD, 1993, Washington, D.C. Proceedings... New York: ACM Press, 1993. p. 207-216.
- AGRAWAL, R.; SRIKANT, R. **Fast Algorithms for Mining Association Rules in Large Databases**. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATABASES, VLDB, 20., 1994, San Francisco. Proceedings... California: Morgan Kaufmann, 1994. p.487 – 499.
- ZAKI. M. **Generating Non-redundant Association Rules**. In: ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, KDD, 6., 2000, Boston. Proceedings... [S.I.]: ACM, 2000. p.34-43.
- ZAKI., M.; HSIAO, C. **CHARM: An Efficient Algorithm for Closed Itemset Mining**. In: INTERNATIONAL CONFERENCE ON DATA MINING, SIAM, 2., 2002, Arlington. Proceedings... [S.I.]:SIAM, 2002.
- HAN, J., PEI, J., and YIN, Y. **Mining frequent patterns without candidate generation**. In: ACM SIGMOD INTERNATIONAL CONFERENCE ON MANAGEMENT OF DATA, SIGMOD, 2000, Dallas. P.1-12.