Regras de Associação
Mineração de Dados

Ronaldo C. Prati

Universidade Federal do ABC (UFABC) - ronaldo.prati@ufabc.edu.br

Mineração de Regreas de Associação

- Dado um conjunto de transações, encontrar regra que predigam a ocorrência de um item baseado na ocorrência de outros item na transação

Transações de supermercado

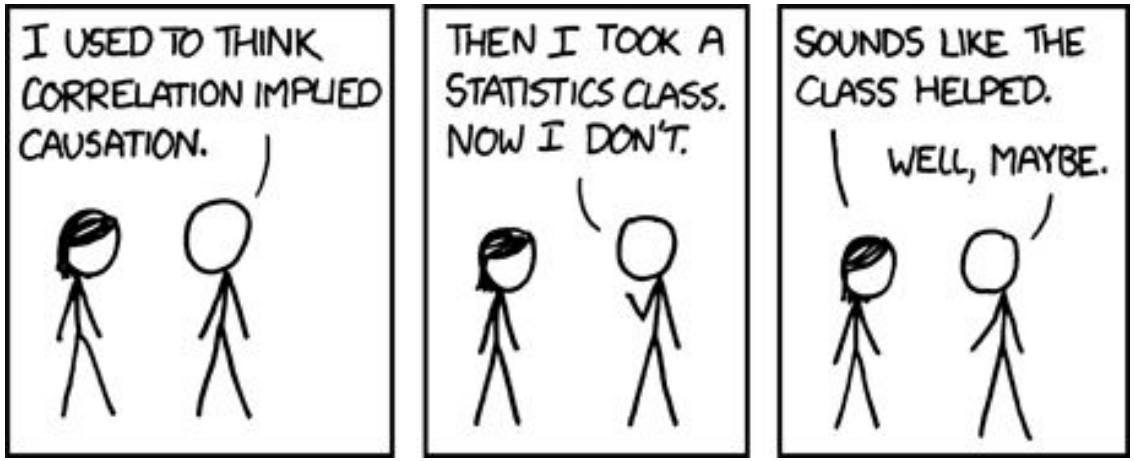
TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Exemplo de regra de associação

{Diaper} → {Beer},
{Milk, Bread} → {Eggs,Coke},
{Beer, Bread} → {Milk},

Implicação implica co-ocorrência, não causalidade!

Correlação não implica causalidade



https://xkcd.com/552/

Definição: itemset frequente

- **Itemset**
  - A coleção de um ou mais itens
    - Exemplo: {Milk, Bread, Diaper}
  - k-itemset
    - Um itemset com k itens
- **Suporte**
  - Fração de transações que contém um itemset
  - E.g. s({Milk, Bread, Diaper}) = 2/5
- **Itemset Frequente**
  - Um itemset cujo suporte é maior ou igual ao suporte mínimo *minsup*

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

Definição: Regra de Associação

- **Regra de Associação**
  - Uma expressão de implicação expression na forma X → Y, em que X e Y são itemsets
  - Exemplo:
    {Milk, Diaper} → {Beer}

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

- **Métrica de avaliação de regra**
  - Suporte (s)
    - Fração de transações que contém ambos X e Y
  - Confidência (c)
    - Mede o quão frequente os itens em Y aparecem em transações que contém X

Definição: Regra de Associação

- **Regra de Associação**
  - Uma expressão de implicação expression na forma X → Y, em que X e Y são itemsets
  - Exemplo:
    {Milk, Diaper} → {Beer}

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

- **Métrica de avaliação de regra**
  - Suporte (s)
    - Fração de transações que contém ambos X e Y
  - Confidência (c)
    - Mede o quão frequente os itens em Y aparecem em transações que contém X

Exemplo:

$$\{\text{Milk}, \text{Diaper}\} \Rightarrow \{\text{Beer}\}$$

$$s = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{|\text{T}|} = \frac{2}{5} = 0.4$$

$$c = \frac{\sigma(\text{Milk}, \text{Diaper}, \text{Beer})}{\sigma(\text{Milk}, \text{Diaper})} = \frac{2}{3} = 0.67$$

Tarefa de Mineração de Regras de Associação

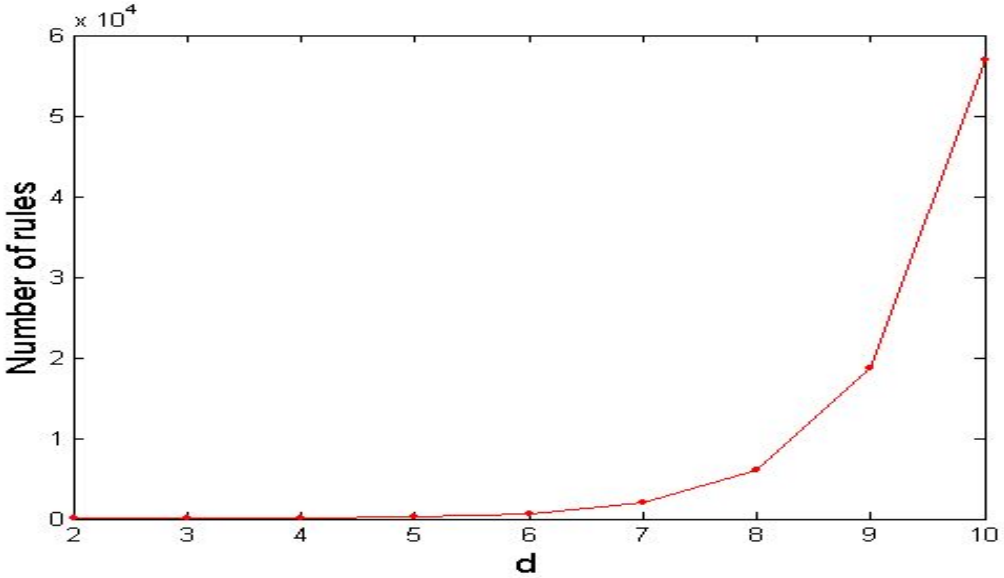
- Dado um conjunto de transações T, o objetivo da mineração de regras de associação é encontrar todas as regras que
  - suporte ≥ *minsup*
  - confidência ≥ *minconf*

- Abordagem força-bruta:
  - Liste todas as possíveis regras de associação
  - Compute o suporte e confidência de cada regra
  - Pode as regras que não atendam os critérios de *minsup* e *minconf*

  ⇒ Computacionalmente proibitivo!

Complexidade Computacional

- Dados d itens únicos:
  - Número total de itemsets = 2^d
  - Número total de possíveis 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

Mineração de regras de associação

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

Exemplo de Regras:

{Milk,Diaper} → {Beer} (s=0.4, c=0.67)
{Milk,Beer} → {Diaper} (s=0.4, c=1.0)
{Diaper,Beer} → {Milk} (s=0.4, c=0.67)
{Beer} → {Milk,Diaper} (s=0.4, c=0.67)
{Diaper} → {Milk,Beer} (s=0.4, c=0.5)
{Milk} → {Diaper,Beer} (s=0.4, c=0.5)

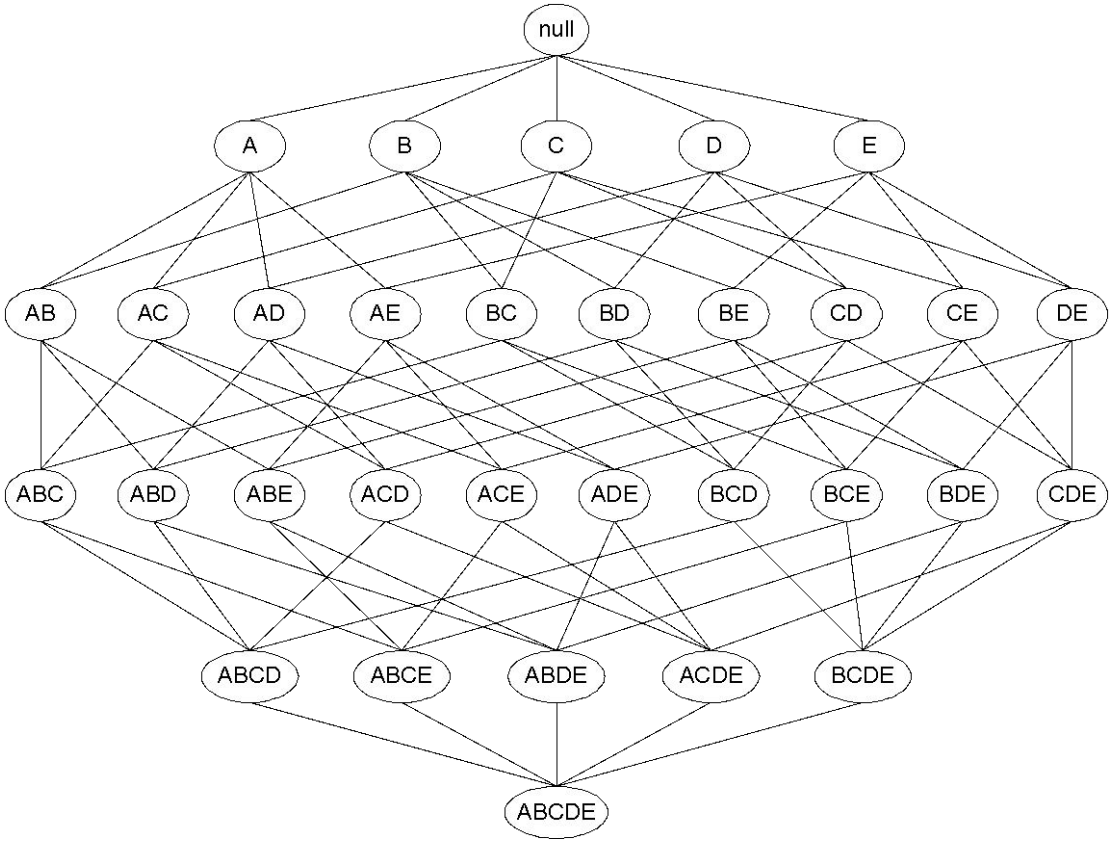
Observações:

- Todas as regras acima são partições binárias do mesmo itemset: {Milk, Diaper, Beer}
- Regras originárias do mesmo itemset tem igual suporte, mas podem ter confidência diferente
- Dessa maneira, podemos dissociar o cálculo do suporte e confidência

Mineração de regras de associação

- Abordagem de dois passos:
  1. Geração dos itemsets frequentes
     - Gera todos itemsets que suporte ≥ minsup
  2. Geração de Regra
     - Gera regras com alta confidência de cada itemset frequente, em que cada regra é uma partição binária de um itemset frequente
- A geração dos itemsets frequentes ainda é uma tarefa computacionalmente cara

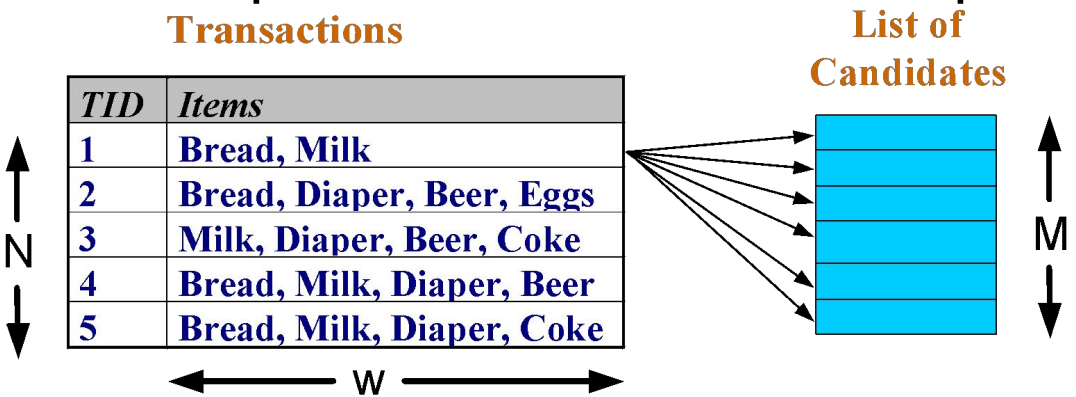
Geração de Itemsets Frequentes



Dados d itens, existe 2^d itemsets candidatos possíveis

Geração de Itemsets Frequentes

- Abordagem força-bruta:
  - Cada itemset na treliça é um itemset candidato frequente
  - Conta o suporte de cada candidato percorrendo a base



  - "Casa" cada transação com cada candidato
  - Complexidade ~ O(NMw) => Caro, uma vez que M = 2^d !!!

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 a medida que o tamanho dos itemsets cresce
- Reduzir o número de comparações (NM)
  - Usar estruturas de dados eficientes para armazenar as transações candidatas
  - Evitar a necessidade de casar cada candidato com cada transação

Reduzir o número de candidatos

- Princípio Apriori:
  - Se um itemset é frequente, então todos os subconjuntos também são frequentes

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

- O princípio Apriori principle é válido devido ao seguinte princípio:
  - Suporte de um itemset nunca excede o suporte de seus subconjuntos
  - Este princípio é conhecido como propriedade anti-monotônica do suporte

Ilustração do princípio Apriori

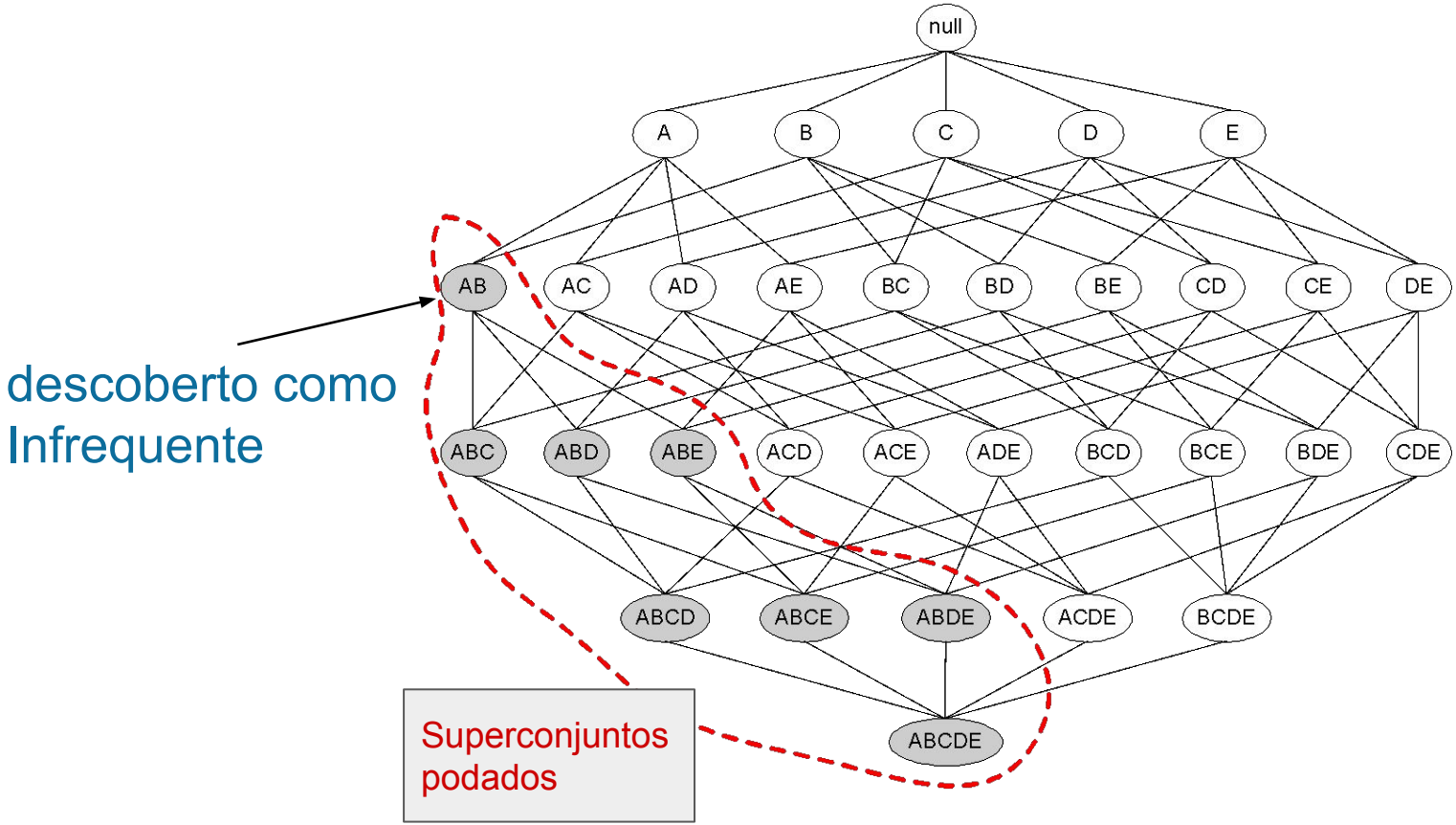


Ilustração do princípio Apriori

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

Suporte Mínimo = 3

Itens
(1-itemsets)

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Ilustração do princípio Apriori

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

Suporte Mínimo = 3

Itens
(1-itemsets)

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Ilustração do princípio Apriori

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Itens
(1-itemsets)

Itemset
{Bread,Milk}
{Bread, Beer }
{Bread,Diaper}
{Beer, Milk}
{Diaper, Milk}
{Beer,Diaper}

Pares (2-itemsets)

(Não é necessário gerar candidatos envolvendo Coke ou Eggs)

Suporte Mínimo = 3

Ilustração do princípio Apriori

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Items
(1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Beer, Bread}	2
{Bread,Diaper}	3
{Beer,Milk}	2
{Diaper,Milk}	3
{Beer,Diaper}	3

Pares (2-itemsets)

(Não é necessário gerar candidatos envolvendo Coke ou Eggs)

Suporte Mínimo = 3

Ilustração do princípio Apriori

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Items
(1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Pares (2-itemsets)

(Não é necessário gerar candidatos envolvendo Coke ou Eggs)

Suporte Mínimo = 3

Itemset
{ Beer, Diaper, Milk}
{ Beer,Bread,Diaper}
{Bread, Diaper, Milk}
{ Beer, Bread, Milk}

Triplas
(3-itemsets)

Ilustração do princípio Apriori

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Items
(1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Pares (2-itemsets)

(Não é necessário gerar candidatos envolvendo Coke ou Eggs)

Suporte Mínimo = 3

Itemset	Count
{ Beer, Diaper, Milk}	2
{ Beer,Bread, Diaper}	2
{Bread, Diaper, Milk}	2
{Beer, Bread, Milk}	1

Triplas
(3-itemsets)

Ilustração do princípio Apriori

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Items
(1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Pares (2-itemsets)

(Não é necessário gerar candidatos envolvendo Coke ou Eggs)

Suporte Mínimo = 3

Considerando todos subconjuntos,
$^{6}C_{1} + {}^{6}C_{2} + {}^{6}C_{3} = 6 + 15 + 20 = 41$
Com poda baseada no suporte,
$^{6}C_{1} + {}^{4}C_{2} + 1 = 6 + 6 + 1 = 13$

Itemset	Count
{ Beer, Diaper, Milk}	2
{ Beer,Bread, Diaper}	2
{Bread, Diaper, Milk}	2
{Beer, Bread, Milk}	1

Triplas
(3-itemsets)

Algoritmo Apriori

- F_k: k-itemsets frequentes
- L_k: k-itemsets candidatos

- Algoritmo
  - Seja k=1
  - Gere F_1 = {1-itemsets frequentes}
  - Repetir até F_k é vazio
    - **Geração de Candidato**: Gere L_{k+1} a partir de F_k
    - **Poda de Candidatos**: Pode os itemsets candidatos em L_{k+1} contendo subconjuntos de tamanho k que são infrequentes
    - **Contagem de suporte**: Calcula o suporte de cada candidato candidate em L_{k+1} percorrendo a DB
    - **Eliminação de Candidato**: Elimine os candidatos em L_{k+1} que não são frequentes, ficando apenas com aqueles que são frequentes => F_{k+1}

Geração de Candidato: método de força-bruta

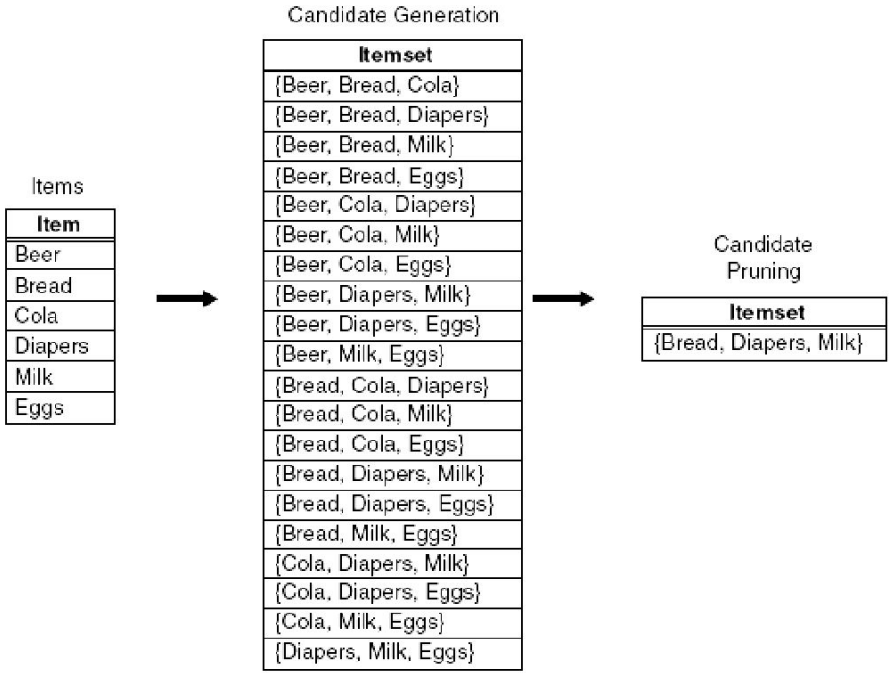


Figure 6.6. A brute-force method for generating candidate 3-itemsets.

Geração de candidato: Una F_{k-1} e F_1 itemsets

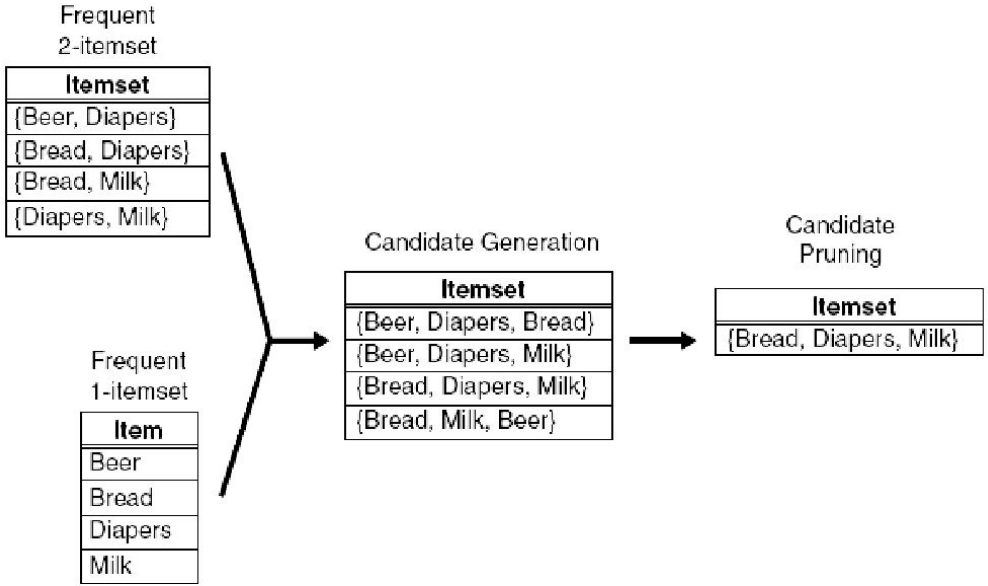


Figure 6.7. Generating and pruning candidate k-itemsets by merging a frequent $(k-1)$-itemset with a frequent item. Note that some of the candidates are unnecessary because their subsets are infrequent.

Geração de Candidatos: Método F_{k-1} x F_{k-1}

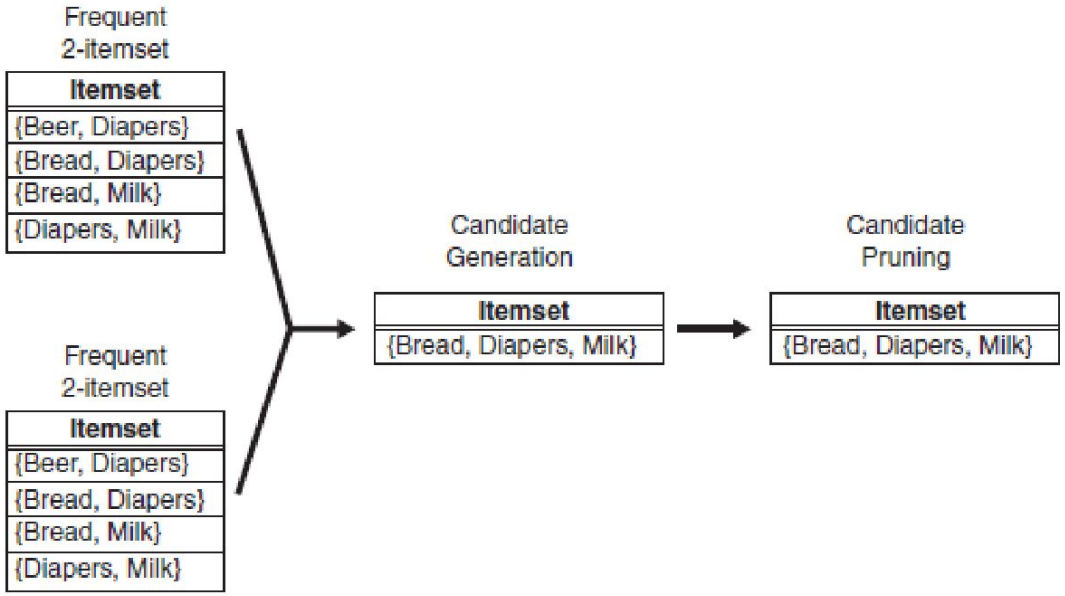


Figure 6.8. Generating and pruning candidate k-itemsets by merging pairs of frequent $(k-1)$-itemsets.

Geração de Candidatos: Método F_{k-1} x F_{k-1}

- Una os conjuntos frequentes (k-1)-itemsets se os primeiros (k-2) items são idênticos

- F_3 = {ABC,ABD,ABE,ACD,BCD,BDE,CDE}
  - Merge(**AB**C, **AB**D) = **AB**CD
  - Merge(**AB**C, **AB**E) = **AB**CE
  - Merge(**AB**D, **AB**E) = **AB**DE

  - Não unir(**A**BD,**A**CD) pois eles compartilham apenas um prefixo de tamanho 1 ao invés de tamanho 2

Poda de Candidatos

- Seja F_3 = {ABC,ABD,ABE,ACD,BCD,BDE,CDE} o conjunto de 3-itemsets frequentes
- L_4 = {ABCD,ABCE,ABDE} é o conjunto de 4-itemsets candidatos gerado (do slide anterior)
- Poda de Candidatos
  - Pode ABCE porque ACE e BCE não são frequentes
  - Pode ABDE porque ADE não é frequente
- Após a poda de candidatos: L_4 = {ABCD}

Ilustração do princípio Apriori

Item	Count
Bread	4
Coke	2
Milk	4
Beer	3
Diaper	4
Eggs	1

Items
(1-itemsets)

Itemset	Count
{Bread,Milk}	3
{Bread,Beer}	2
{Bread,Diaper}	3
{Milk,Beer}	2
{Milk,Diaper}	3
{Beer,Diaper}	3

Pares (2-itemsets)

(Não é necessário gerar candidatos envolvendo Coke ou Eggs)

Suporte Mínimo = 3

Considerando todos subconjuntos,
${}^6C_1 + {}^6C_2 + {}^6C_3 = 6 + 15 + 20 = 41$
Com poda baseada no suporte,
${}^6C_1 + {}^4C_2 + 1 = 6 + 6 + 1 = 13$

Itemset	Count
{Bread, Diaper, Milk}	2

Triplas
(3-itemsets)

O uso do método $F_{k-1}xF_{k-1}$ para a geração de candidatos resulta em somente um 3-itemset. Ele é eliminado após a contagem de suporte

Geração da Regra

- Dado um itemset frequente L, encontrar todos subconjuntos não vazios $f \subset L$ tal que $f \rightarrow L - f$ satisfaz o critério de confidência mínima
  - Se {A,B,C,D} é um itemset frequente, as regras candidatas são:

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

- Se $|L| = k$, então existe $2^k - 2$ regras de associação candidatas (ignorando $L \rightarrow \emptyset$ e $\emptyset \rightarrow L$)

Geração da Regra

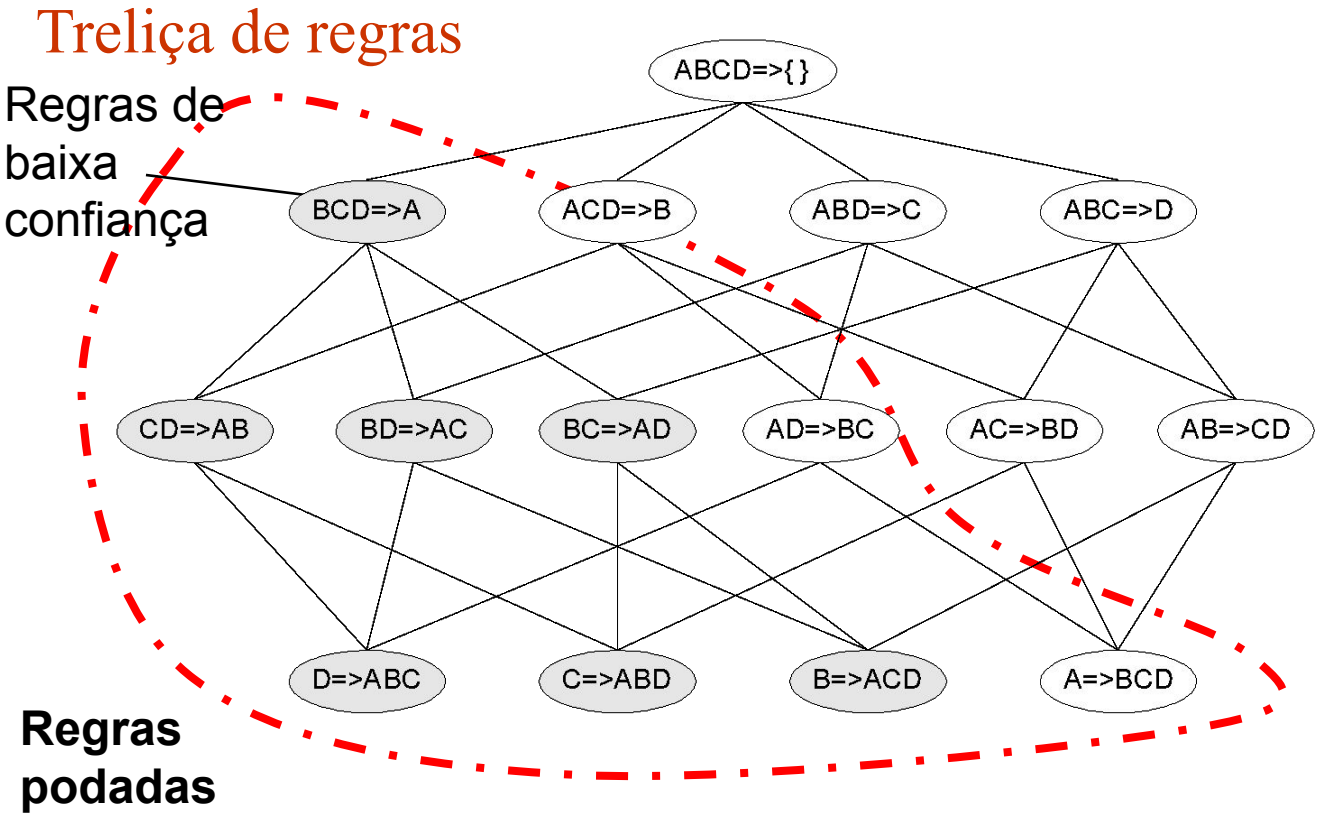
- Em geral, a confidência não tem uma propriedade de anti-monotonia

  $c(ABC \rightarrow D)$ pode ser maior que $c(AB \rightarrow D)$

- Mas a confidência das regras geradas a partir de um mesmo itemset tem a propriedade de anti-monotonia
  - E.g., Suponha que {A,B,C,D} é um 4-itemset frequente:

    $c(ABC \rightarrow D) \geq c(AB \rightarrow CD) \geq c(A \rightarrow BCD)$
  - Confidência é anti-monotônica com respeito ao número de itens do lado direito da regra

Geração da Regra



Avaliação de Regras

- Algoritmos de regras de associação podem produzir um grande número de regras
- Medidas de interesse podem ser usadas para podar/ordenar as regras
  - Na formulação original, suporte e confidência são as únicas métricas utilizadas

Calculando medidas de interesse

- Dado X → Y ou {X,Y}, as informações necessárias para calcular a medida de interesse pode ser obtida pela matriz de contingência

Matriz de contingência

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

f_{11}: suporte de X e Y
f_{10}: suporte de X e $\overline{Y}$
f_{01}: suporte de $\overline{X}$ e Y
f_{00}: suporte de $\overline{X}$ e $\overline{Y}$

Usada para definir várias métricas

- suporte, confidência, Gini, entropia, etc.

Deficiências da Confidencia

Consumidores	Tea	Coffee	…
C1	0	1	…
C2	1	0	…
C3	1	1	…
C4	1	0	…
…			

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

Regra de Associação: Tea → Coffee

Confidência ≅ P(Coffee|Tea) = 15/20 = 0.75

Confidência > 50%, significa que pessoas bebem chá são mais prováveis de beberem café do que não beber café

Então a regra parece razoável

Drawback of Confidence

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

Regra de Associação: Tea → Coffee

Confidência = P(Coffee|Tea) = 15/20 = 0.75

mas P(Coffee) = 0.9, o que significa que saber o fato que a pessoa bebe chá reduz a probabilidade que a pessoa bebe café

⇒ Observe que P(Coffee|$\overline{\text{Tea}}$) = 75/80 = 0.9375

Medidas para regras de associação

- Então, que tipo de regras queremos?
  - Confidência (X → Y) deve ser suficientemente alta
    - Para garantir que quem compra X tem mais chance de comprar Y que não comprar Y
  - Confidência(X → Y) > suporte(Y)
    - Caso contrário, a regra será enganosa porque ter o item X na verdade reduz as chances de ter o item Y na mesma transação
    - Existe outra medida que captura essa restrição?
      - Resposta: Sim. Existem várias.

Independência estatística

- O critério

  confidência($X \rightarrow Y$) = suporte(Y)

  é equivalente a:

  - $P(Y|X) = P(Y)$
  - $P(X,Y) = P(X) \times P(Y)$

  Se $P(X,Y) > P(X) \times P(Y)$: X & Y são positivamente correlacionadas

  Se $P(X,Y) < P(X) \times P(Y)$: X & Y são negativamente correlacionadas

Medidas que levam em consideração independência estatística

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

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

lift é usada para regras, enquanto interesse é usada para itemsets

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

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

Exemplo: Lift/Interest

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

Regra de Associação: Tea → Coffee

Confidence= P(Coffee|Tea) = 0.75

ms P(Coffee) = 0.9

⇒ Lift = 0.75/0.9= 0.8333 (< 1, então é negativamente associada)

Então, é o suficiente usar lift para poda?

Existem diversas medidas propostas na literatura

#	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)	$\max\left(P(A,B)\log(\frac{P(B\|A)}{P(B)}) + P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}),\right.$ $\left. P(A,B)\log(\frac{P(A\|B)}{P(A)}) + P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(A)})\right)$
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)	$\max\left(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2}\right)$
13	Conviction (V)	$\max\left(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})}\right)$
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)	$\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$
20	Jaccard (ζ)	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
21	Klosgen (K)	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$

Simpson's Paradox

Buy HDTV	Buy Exercise Machine		
	Yes	No	
Yes	99	81	180
No	54	66	120
	153	147	300

$$c(\{\text{HDTV} = \text{Yes}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 99/180 = 55\%$$

$$c(\{\text{HDTV} = \text{No}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 54/120 = 45\%$$

=> Consumidores que compram HDTV são tem mais probabilidade de comprar máquinas de exercício

Simpson's Paradox

Customer Group	Buy HDTV	Buy Exercise Machine		Total
		Yes	No	
College Students	Yes	1	9	10
	No	4	30	34
Working Adult	Yes	98	72	170
	No	50	36	86

College students:

$$c(\{\text{HDTV} = \text{Yes}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 1/10 = 10\%$$

$$c(\{\text{HDTV} = \text{No}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 4/34 = 11.8\%$$

Working adults:

$$c(\{\text{HDTV} = \text{Yes}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 98/170 = 57.7\%$$

$$c(\{\text{HDTV} = \text{No}\} \rightarrow \{\text{Exercise Machine} = \text{Yes}\}) = 50/86 = 58.1\%$$

Simpson's Paradox

- Relações observadas nos dados podem ser influenciadas pela presença de fatores confusores (variáveis ocultas)
  - Variáveis ocultas podem fazer com que a direção do relacionamento seja revertido!