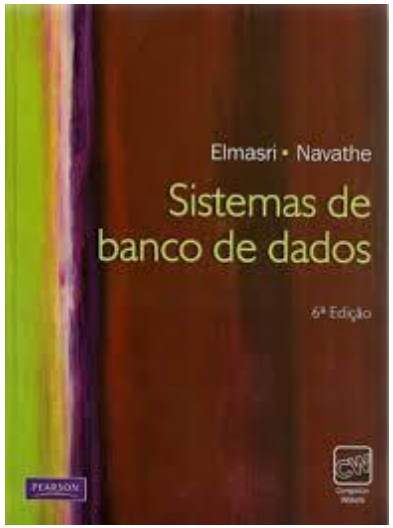
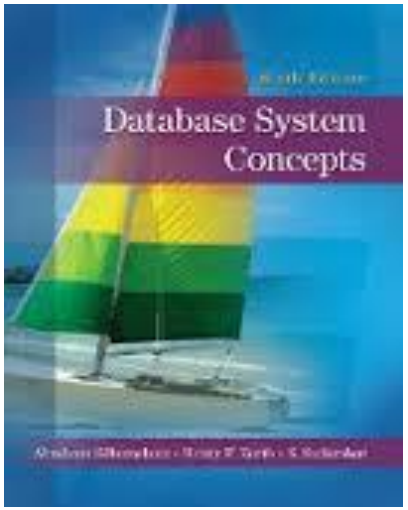
Unidade 23 – Conceitos de Mineração de Dados – Parte 2

Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP

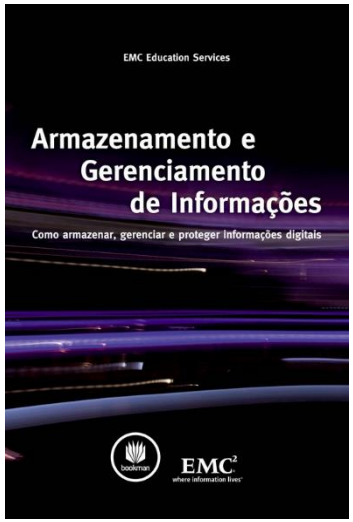
Bibliografia



Sistemas de Banco de Dados
Elmasri / Navathe 6ª edição



Sistema de Banco de Dados
Korth, Silberschatz – Sixth Editon



http://education.EMC.com/ismbook

Tipos de Conhecimentos descobertos pela Mineração de Dados

- A mineração de dados enfoca o **Conhecimento Indutivo**, que descobre **novas regras** e **padrões** com base nos **dados fornecidos**;

- É comum descrever-se o conhecimento descoberto durante a Mineração de Dados por:

  - ✓ **Regras de Associação;**
  - ✓ **Hierarquias de Classificação;**
  - ✓ **Padrões Sequenciais;**
  - ✓ **Padrões dentro de séries temporais;**
  - ✓ **Agrupamento.**

Regras de Associação

- É uma das técnicas mais conhecidas de mineração de dados, devido ao problema da Análise da Cesta de Compras.
- Consiste em identificar o relacionamento dos itens mais frequentes em um determinado conjunto de dados, e permite obter resultados do tipo: **SE** compra **leite** e **pão TAMBÉM** compra **manteiga**.
- Exemplo:
  - ✓ **Quando uma compradora compra bolsa, ela provavelmente compra sapatos;**
  - ✓ **Uma imagem de raio X contendo características a e b provavelmente também exibirá a característica c;**

Regra 1: SE *idade* = *jovem* AND *estudante* = *não* ENTÃO *compra computadores* = *não*
Regra 2: SE *idade* = *jovem* AND *estudante* = *sim* ENTÃO *compra computadores* = *sim*
Regra 3: SE *idade* = *média* ENTÃO *compra computadores* = *sim*
Regra 4: SE *idade* = *adulto* AND *avaliação de crédito* = *excelente* ENTÃO *compra computadores* = *sim*
Regra 5: SE *idade* = *adulto* AND *avaliação de crédito* = *ruim* ENTÃO *compra computadores* = *não*

Regras de Associação

- É uma das técnicas mais conhecidas de mineração de dados, devido ao problema da Análise da Cesta de Compras.
- Consiste em identificar o relacionamento dos itens mais frequentes em um determinado conjunto de dados, e permite obter resultados do tipo: **SE** compra **leite** e **pão TAMBÉM** compra **manteiga**.
- Exemplo:
  - ✓ **Quando uma compradora compra bolsa, ela provavelmente compra sapatos;**
  - ✓ **Uma imagem de raio X contendo características a e b provavelmente também exibirá a característica c;**

Regra 1: SE *idade* = *jovem* AND *estudante* = *não* ENTÃO *compra computadores* = *não*
Regra 2: SE *idade* = *jovem* AND *estudante* = *sim* ENTÃO *compra computadores* = *sim*
Regra 3: SE *idade* = *média* ENTÃO *compra computadores* = *sim*
Regra 4: SE *idade* = *adulto* AND *avaliação de crédito* = *excelente* ENTÃO *compra computadores* = *sim*
Regra 5: SE *idade* = *adulto* AND *avaliação de crédito* = *ruim* ENTÃO *compra computadores* = *não*

Regras de Associação
Problema da Análise da Cesta de Compras.

- Para ilustrar a técnica de Mineração de Dados – **Regras de Associação**, considere um banco de dados com uma coleção de transações relativas aos dados de cesta de mercado;
- A cesta de mercado corresponde aos **conjuntos** de **itens** que um consumidor compra em um supermercado durante uma visita;
- Considere quatro **transações** em uma amostra aleatória, conforme figura abaixo:

Id_transação	Hora	Itens_comprados
101	6:35	leite, pão, biscoito, suco
792	7:38	leite, suco
1130	8:05	leite, ovos
1735	8:40	pão, biscoito, café

Regras de Associação
Problema da Análise da Cesta de Compras.

- Uma regra de associação tem a forma **X** => **Y**, onde **X** = { x_1, x_2, ... , x_n } e **Y** = { y_1, y_2, ..., y_n} são conjuntos de itens;
- Essa associação indica que, se um cliente compra **X**, ele também **provavelmente** comprará **Y**;
- Em geral, qualquer regra de associação tem a forma **LHS** => **RHS**, onde **LHS** é o conjunto de itens do lado esquerdo (Left Hand Side) e **RHS** é o conjunto de itens do lado direito (Right Hand Side);
- Exemplo: **X** = { leite } **Y** = { suco }

Id_transação	Hora	Itens_comprados
101	6:35	leite, pão, biscoito, suco
792	7:38	leite, suco
1130	8:05	leite, ovos
1735	8:40	pão, biscoito, café

Suporte para uma Regra de Associação

- O **suporte** para uma regra de associação **LHS => RHS** se refere à frequência de vezes com que um itemset específico ocorre no banco de dados;
- Ou seja, o **suporte** é o percentual de transações que contêm o itemset considerado;
- Se o **suporte** for baixo, isso implica que não existe uma evidência forte de que os itens ocorrem juntos, pois ocorrem em apenas uma fração das transações;
- Suporte para uma regra também é conhecido por **prevalência** da regra;
- Exemplo: a) Suporte de **leite => suco** é de **50**%

  b) Suporte de **pão => suco** é de **25**%

Id_transação	Hora	Itens_comprados
101	6:35	leite, pão, biscoito, suco
792	7:38	leite, suco
1130	8:05	leite, ovos
1735	8:40	pão, biscoito, café

Confiança para uma Regra de Associação

- A **confiança** para uma regra de associação **LHS => RHS** é definida por:

  Suporte (LHS U RHS) / Suporte (LHS)

- Pode-se pensar na confiança como sendo a probabilidade de que os itens no RHS sejam comprados, **dado** que os itens no LSH foram comprados;

- Outro termo para confiança de regra de associação é **força da regra**;

- Exemplos:

  a) confiança de **leite => suco = 2/3 (67%),** significando que, das **três** transações em que ocorre leite, **duas** contêm suco;

  b) confiança de **pão => suco = 1/2 (50%),** significando que, das **duas** transações em que ocorre pão, **uma** contém suco.

Id_transação	Hora	Itens_comprados
101	6:35	leite, pão, biscoito, suco
792	7:38	leite, suco
1130	8:05	leite, ovos
1735	8:40	pão, biscoito, café

Qual a relação entre Suporte e Confiança?

Relação entre Suporte e Confiança

- **Suporte** e **confiança** necessariamente não andam lado a lado;
- O objetivo da mineração de regras de associação, então, é gerar todas as regras possíveis que excedam alguns patamares mínimos de **Suporte** e **Confiança** especificados pelo usuário;
- Existem alguns algoritmos para a geração de regras de associação, destacando-se o Algoritmo de **Apriori** (**IBM**, Agrawal Ramakrishnan Srikant, 1993).

Suporte e Confiança Mínimos

- **Suporte** Mínimo: É a frequência mínimo que um item deve ter para que seja considerado frequente; (**Minimum Support**)

- **Confiança** Mínima: É a confiança mínima que um item precisa ter para que seja considerado confiável. (**Minimum Confidence**)

Regras Fortes (Strong Rules)

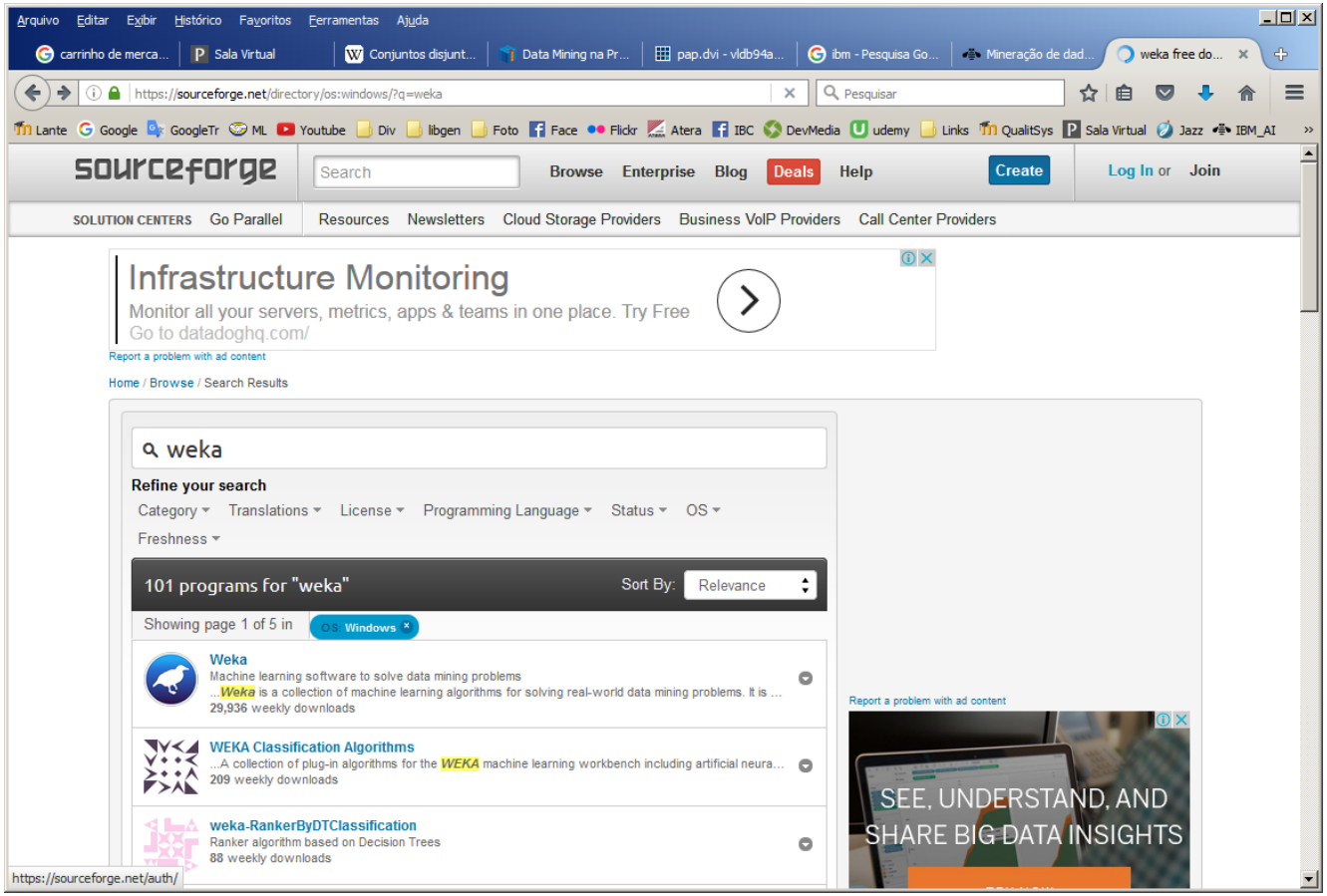
- São aquelas que atingem o **mínimo** de **suporte** e o **mínimo** de **confiança**;

Algoritmo de Apriori

- Implementado em diversas ferramentas de **Data Mining** (mineração de dados), como o **Weka**;
- O algoritmo recebe como argumento um conjunto de transações **T**, o valor percentual **S** como o **Suporte** e um valor percentual **C** para a **confiança**.
- O algoritmo gera um conjunto de regras no formato **A => B [Suporte, confiança],** onde o conjunto **A** é chamado de **antecedente da regra** e o conjunto **B** é chamado de **consequente**.
- Cada regra gerada deve ser seu **Suporte** e sua **confiança** **maior ou igual** ao **Suporte** e **Confiança** **mínimo** passado para o algoritmo, respectivamente;
- Necessita de várias interações com o Banco de Dados, mas é relativamente fácil de ser implementado.

WEKA

- Disponível em **https://sourceforge.net/directory/os:windows/?q=weka**



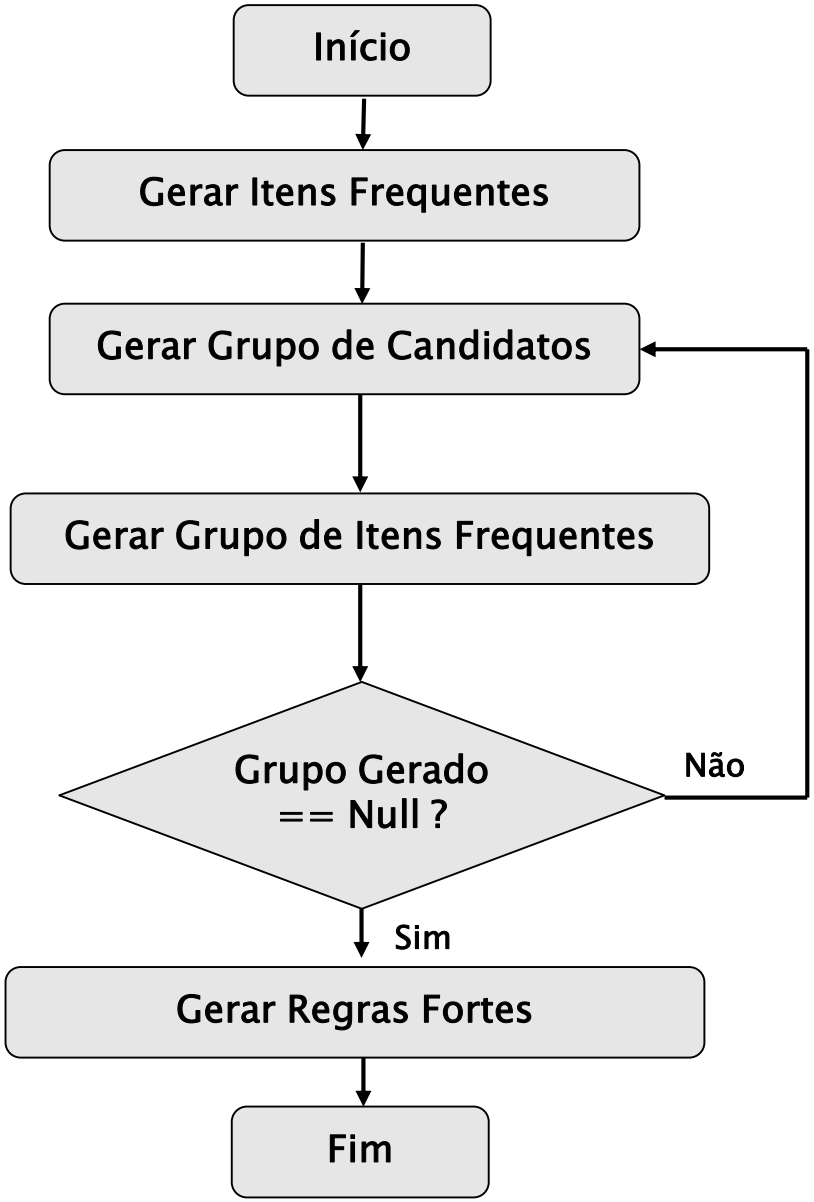
WEKA

- **WEKA** é um produto da Universidade de Waikato (Nova Zelândia) – 1997;
- GNU General Public License (GPL);
- Escrito na linguagem **Java**™;
- Contém uma **GUI** para interagir com arquivos de dados e produzir resultados visuais.

Algoritmo de Apriori

- O algoritmo APRIORI é dividido em duas partes;
- O algoritmo recebe como argumento um conjunto de transações **T**, o valor percentual **S** como o **Suporte** e um valor percentual **C** para a **Confiança**;
- Na primeira parte são selecionados todos os subconjuntos de **T** que podem ser utilizados em alguma regra, ou seja, que contenham o **Suporte** acima do Suporte mínimo **S**;
- A segunda parte do algoritmo faz a geração das regras a partir dos subconjuntos gerados na primeira parte, sendo que estas regras devem ter uma confiança maior que a **Confiança** mínima **C**.

Algoritmo de Apriori



Algoritmo de Apriori
Exemplo - Motivação

- Descobrir o **comportamento** dos consumidores em um mercado;
- **Organizar** as prateleiras de modo a deixar **os produtos relacionados mais próximos** e assim, **maximizar** as vendas.

Exemplo

Banco de Dados	
Itens de Compra	**ID**
Leite, Pão, Bolacha, Suco	1
Leite, Suco	2
Leite, Ovos	3
Pão, Bolacha, Café	4

Suporte Mínimo	Confiança Mínima
50%	75%

Algoritmo de Apriori – Primeira Etapa

- Gerar a tabela de Candidatos com suas frequências (Suporte)

Banco de Dados	
Itens de Compra	**ID**
Leite, Pão, Bolacha, Suco	1
Leite, Suco	2
Leite, Ovos	3
Pão, Bolacha, Café	4

Suporte Mínimo
50%

Confiança Mínima
75%

Algoritmo de Apriori – Primeira Etapa

- Gerar a tabela de Candidatos com suas frequências (Suporte)

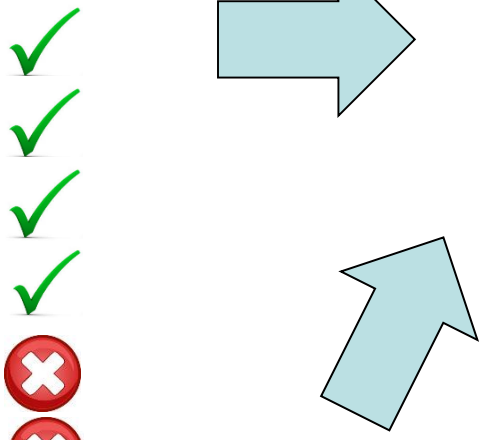
Banco de Dados	
Itens de Compra	**ID**
Leite, Pão, Bolacha, Suco	1
Leite, Suco	2
Leite, Ovos	3
Pão, Bolacha, Café	4

Candidatos	
Frequência	Item
¾ = 75%	Leite
2/4 = 50%	Pão
2/4 = 50%	Bolacha
2/4 = 50%	Suco
¼ = 25%	Ovos
¼ = 25%	Café

Algoritmo de Apriori – Segunda Etapa

- Gerar a tabela com itens frequentes (Análise do Suporte)

Candidatos	
Frequência	Item
¾ = 75%	Leite
2/4 = 50%	Pão
2/4 = 50%	Bolacha
2/4 = 50%	Suco
¼ = 25%	Ovos
¼ = 25%	Café



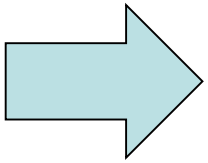
Frequentes	
Frequência	Item
¾ = 75%	Leite
2/4 = 50%	Pão
2/4 = 50%	Bolacha
2/4 = 50%	Suco

Suporte Mínimo
50%

Algoritmo de Apriori – Terceira Etapa

- Gerar a tabela de candidatos a partir da combinação dos itens frequentes e calcular suas respectivas frequências (Suporte);

Frequentes	
Frequência	Item
¾ = 75%	Leite
2/4 = 50%	Pão
2/4 = 50%	Bolacha
2/4 = 50%	Suco



Candidatos	
Frequência	Item
1/4 = 25%	Leite,Pão
1/4 = 25%	Leite,Bolacha
2/4 = 50%	Leite,Suco
2/4 = 50%	Pão, Bolacha
¼ = 25%	Pão, Suco
¼ = 25%	Bolacha, Suco

Banco de Dados	
Itens de Compra	ID
Leite, Pão, Bolacha, Suco	1
Leite, Suco	2
Leite, Ovos	3
Pão, Bolacha, Café	4

Algoritmo de Apriori – Quarta Etapa

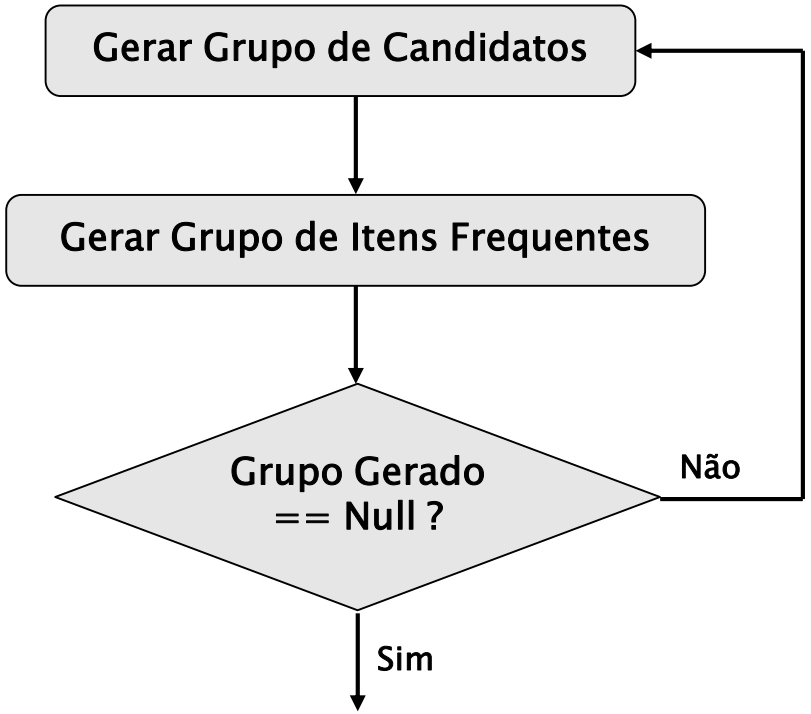
- Gerar tabela de grupos de itens frequentes;

Candidatos	
Frequência	Item
1/4 = 25%	Leite,Pão
1/4 = 25%	Leite,Bolacha
2/4 = 50%	Leite,Suco
2/4 = 50%	Pão, Bolacha
¼ = 25%	Pão, Suco
¼ = 25%	Bolacha, Suco

Frequentes	
Frequência	Item
2/4 = 50%	Leite,Suco
2/4 = 50%	Pão, Bolacha

Suporte Mínimo
50%

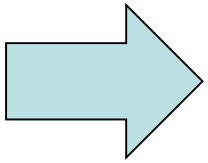
Algoritmo de Apriori



Algoritmo de Apriori – Terceira Etapa

- Gerar tabela de candidatos a partir da combinação dos itens frequentes e calcular suas respectivas frequências (suporte);

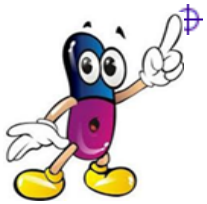
Frequentes	
Frequência	Item
2/4 = 50%	Leite,Suco
2/4 = 50%	Pão, Bolacha



Candidatos	
Frequência	Item
1/4 = 25%	Leite, Suco e Pão
1/4 = 25%	Leite, Suco e Bolacha
1/4 = 25%	Pão, Bolacha e Leite
1/4 = 25 %	Pão, Bolacha e Suco

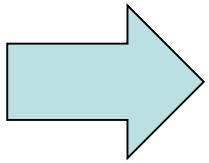
Banco de Dados	
Itens de Compra	ID
Leite, Pão, Bolacha, Suco	1
Leite, Suco	2
Leite, Ovos	3
Pão, Bolacha, Café	4

Algoritmo de Apriori – Quarta Etapa

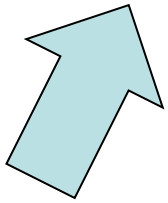


Gerar tabela de grupo de itens frequentes;

Candidatos	
Frequência	Item
1/4 = 25%	Leite, Suco e Pão
1/4 = 25%	Leite, Suco e Bolacha
1/4 = 25%	Pão, Bolacha e Leite
1/4 = 25 %	Pão, Bolacha e Suco

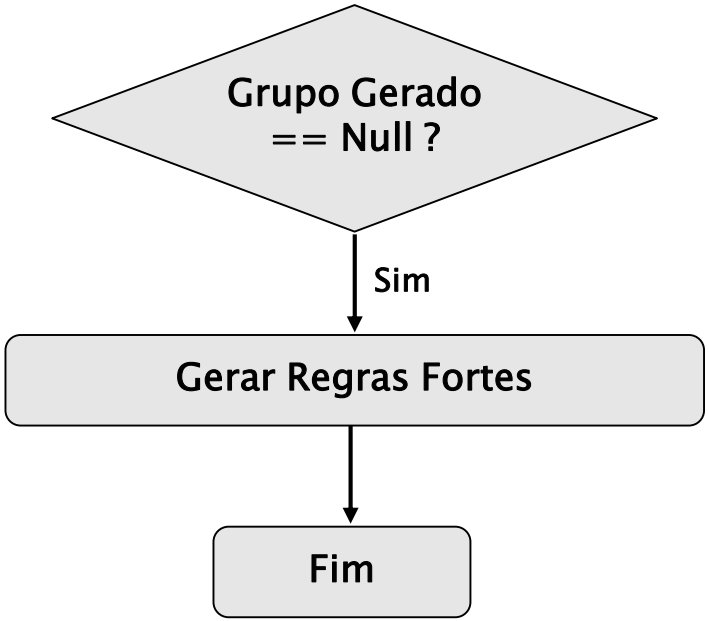


Frequentes	
Frequência	Item



Suporte Mínimo
50%

Algoritmo de Apriori



Algoritmo de Apriori – Quinta Etapa

- A partir do **último grupo** de itens frequentes, calcular suas respectivas confianças;

Confiança
A => B É o número de tuplas que contem A e B dividido pelo número de tuplas que contém A

Banco de Dados	
Itens de Compra	**ID**
Leite, Pão, Bolacha, Suco	1
Leite, Suco	2
Leite, Ovos	3
Pão, Bolacha, Café	4

Frequentes	
Frequência	Item
2/4 = 50%	**Leite,Suco**
2/4 = 50%	**Pão, Bolacha**

Combinações	Suporte	Confiança
Leite => Suco	50%	2/3 = 67%
Suco => Leite	50%	2/2 = 100%
Pão => Bolacha	50%	2/2 = 100%
Bolacha => Pão	50%	2/2 = 100%

Algoritmo de Apriori – Sexta Etapa

- Verificar Regras Fortes;

Regras Fortes

São as regras que atingirem o suporte e a confiança **mínimas;**

Combinações	Suporte	Confiança	
Leite => Suco	50%	2/3 = 67%	✗
Suco => Leite	50%	2/2 = 100%	✓
Pão => Bolacha	50%	2/2 = 100%	✓
Bolacha => Pão	50%	2/2 = 100%	✓

Suporte Mínimo
50%

Confiança Mínima
75%

Algoritmo de Apriori – Sexta Etapa

Combinações	Suporte	Confiança
Suco => Leite	50%	2/2 = 100%
Pão => Bolacha	50%	2/2 = 100%
Bolacha => Pão	50%	2/2 = 100%

- Então, a partir da tabela resultante:

- ✓ Quem compra **SUCO**, compra **LEITE**;
- ✓ Quem compra **PÃO**, compra **BOLACHA**;
- ✓ Quem compra **BOLACHA**, compra **PÃO**.