

# Algoritmos Não Supervisionados – Apriori



Plataforma completa de aprendizado  
contínuo em programação.

#BoostingPeople

[rocketseat.com.br](http://rocketseat.com.br)

Todos os direitos reservados © Rocketseat S.A.

# Algoritmos Não Supervisionados

Apriori

O objetivo deste módulo é apresentar conceitualmente os principais algoritmos de [regras de associação](#) para que possamos descobrir [relações ocultas entre itens](#) em grandes bases de dados. E faremos um projeto explorando o algoritmo [Apriori](#), onde faremos o [processo completo](#) desde o EDA até a visualização dos resultados.



# Agenda

- O que são regras de associação?
- Um passeio pelos algoritmos de regras de associação
- Casos de Uso e Aplicações
- O que é o algoritmo Apriori
- Como funciona o algoritmo Apriori
- Métricas de algoritmos de regras de associação
- Projeto - Apriori



# O que são regras de associação?

Regras de associação são uma técnica de mineração de dados utilizada para descobrir relações significativas, frequentes e úteis entre conjuntos de itens em grandes bases de dados transacionais. Elas são especialmente úteis para identificar padrões ocultos que podem ajudar na tomada de decisões em várias áreas, como marketing, varejo, saúde e finanças.

Os algoritmos de regras de associação surgiram na **década de 1990**, no contexto da crescente necessidade de análise de grandes volumes de dados gerados por sistemas de transações comerciais. Com o advento dos **sistemas de ponto de venda (POS)** e a **digitalização das transações**, o volume de dados disponíveis aumentou exponencialmente. Esse cenário criou uma demanda por técnicas que pudessem **extrair informações valiosas a partir desses dados massivos**.

# O que são regras de associação?

As regras de associação são expressas na forma de implicações, onde a **presença de um conjunto de itens em uma transação implica a presença de outro conjunto de itens**.

Formalmente, uma regra de associação é representada com:

$(A \rightarrow B)$ , onde  $(A)$  e  $(B)$  são conjuntos de itens, e a regra indica que as transações que contêm  $(A)$  tendem a também conter  $(B)$ .

**Antecedente (A):** O conjunto de itens que aparece antes da implicação. Também chamado de premissa ou corpo da regra.

**Consequente (B):** O conjunto de itens que aparece após a implicação. Também chamado de conclusão ou cabeça da regra.

# O que são regras de associação?

## Fundamentos

**Item:** Um objeto ou entidade de interesse. Por exemplo, em uma base de dados de supermercado, um item pode ser “leite” ou “pão”.

**Itemset:** Um conjunto de itens. Por exemplo, {leite, pão} é um itemset.

**Transações:** Uma transação é uma coleção de itens comprados ou ocorridos juntos. Cada transação é representada por um itemset. Exemplo: Uma transação pode ser {leite, pão, manteiga}.

**Banco de Dados Transacional:** Uma coleção de transações. Cada linha na base de dados representa uma transação.

# O que são regras de associação?

## Principais Objetivos e Benefícios

**Identificação de Relacionamentos Significativos:** Descobrir relações úteis entre itens que podem não ser evidentes à primeira vista. Isso ajuda as empresas a entender melhor os hábitos de compra dos clientes e a desenvolver estratégias mais eficazes.

**Aumento das Vendas e Receita:** Utilizar as associações descobertas para criar promoções cruzadas e pacotes de produtos que incentivem os clientes a comprar mais itens juntos, aumentando assim as vendas e a receita.

**Otimização do Layout das Lojas:** Analisar padrões de compra para otimizar a disposição dos produtos nas lojas físicas, facilitando a navegação dos clientes e aumentando a probabilidade de compras impulsivas.

# O que são regras de associação?

## Principais Objetivos e Benefícios

**Personalização de Ofertas e Campanhas:** Segmentar os clientes com base nos padrões de compra e personalizar ofertas e campanhas de marketing, aumentando a taxa de conversão e a satisfação do cliente.

**Detecção de Fraudes:** Identificar padrões anômalos que possam indicar atividades fraudulentas, ajudando as instituições financeiras a prevenir e mitigar fraudes.

**Melhoria na Gestão de Estoques:** Prever a demanda por produtos com base nos padrões de compra, permitindo uma melhor gestão de estoques e redução de custos com armazenamento e desperdício.

# O que são regras de associação?

## Casos de Uso

**Market Basket Analysis:** Utilizada em supermercados e lojas de varejo para descobrir quais produtos são frequentemente comprados juntos. Isso pode levar à criação de promoções cruzadas, como “compre 1 e leve outro com desconto”.

**Sistemas de Recomendação:** Plataformas de e-commerce utilizam regras de associação para recomendar produtos ou filmes com base nas compras ou visualizações anteriores dos clientes.

**Detecção de Fraudes:** Instituições financeiras utilizam regras de associação para identificar padrões de comportamento que indicam fraudes, como transações incomuns ou suspeitas em contas bancárias.

# O que são regras de associação?

## Casos de Uso

**Análise de Dados Médicos:** Na saúde, regras de associação podem ser usadas para descobrir padrões em dados de pacientes que ajudam a identificar fatores de risco e a melhorar diagnósticos e tratamentos.

**Telecomunicações:** Empresas de telecomunicações utilizam regras de associação para analisar o comportamento dos clientes e identificar padrões que podem indicar churn (cancelamento de serviços), permitindo a criação de estratégias de retenção mais eficazes.

# Um passeio pelos algoritmos de regras de associação

## Apriori

Identifica itemsets frequentes gerando candidatos e utilizando a propriedade anti-monotônica, que afirma que se um itemset é frequente, todos os seus subconjuntos também são frequentes.

## FP-Growth

Utiliza uma estrutura de árvore (FP-tree) para armazenar itemsets frequentes de forma compacta, evitando a geração de candidatos como no Apriori.

## Eclat

Usa uma abordagem de interseção de listas de transações para identificar itemsets frequentes. É baseado na decomposição da busca de itemsets em classes de equivalência.

## AIS

Foi um dos primeiros algoritmos de regras de associação. Ele gera itemsets frequentes de maneira iterativa durante a leitura do banco de dados.

# O que é o algoritmo Apriori

O algoritmo **Apriori**, um dos primeiros e mais conhecidos algoritmos de regras de associação, foi introduzido por Rakesh Agrawal e Ramakrishnan Srikant em 1994. Eles apresentaram o algoritmo em um artigo intitulado “Fast Algorithms for Mining Association Rules”, publicado na conferência internacional Very Large Data Bases (VLDB).

A motivação para desenvolver o Apriori e outros algoritmos de regras de associação foi a **necessidade de encontrar padrões frequentes em bases de dados transacionais**. O exemplo clássico usado para ilustrar esses algoritmos é a **análise de cestas de compras (market basket analysis)**, onde o objetivo é descobrir associações entre produtos comprados juntos pelos clientes.

# Como funciona o algoritmo Apriori

**Reunir Dados Iniciais:** Imagine que temos uma lista de compras de vários clientes. Cada lista de compras é chamada de “transação”. O primeiro passo é olhar para todos os itens únicos comprados.

**Encontrar Itens Populares:** Primeiro, contamos quantas vezes cada item individual aparece nas listas de compras. Por exemplo, se leite aparece em 3 listas de 5, ele é comprado em 60% das vezes.

**Filtrar Itens Frequentes:** Definimos um limite mínimo de popularidade, chamado “suporte mínimo”. Só consideramos os itens que atingem ou superam esse limite. Por exemplo, se decidirmos que um item precisa estar em pelo menos 50% das listas, mantemos apenas os itens que atendem a esse critério.

# Como funciona o algoritmo Apriori

**Criar Combinações de Itens:** Agora, olhamos para todas as combinações de dois itens a partir dos itens populares. Por exemplo, combinamos “leite e pão”, “leite e manteiga”, e assim por diante.

**Contar Combinações Populares:** Contamos quantas vezes cada combinação de dois itens aparece nas listas de compras. Por exemplo, se “leite e pão” aparecem juntos em 40% das listas, essa combinação é mantida.

**Repetir com Combinações Maiores:** Repetimos o processo, agora olhando para combinações de três itens a partir das combinações populares de dois itens. Continuamos aumentando o número de itens na combinação e contando quantas vezes aparecem juntos, até que não haja mais combinações populares.

# Como funciona o algoritmo Apriori

**Criar Regras de Associação:** Com as combinações populares, criamos regras. Por exemplo, se “leite e pão” são populares juntos, podemos criar a regra: “Se alguém compra leite, também compra pão”. Calculamos a “confiança” dessas regras, que é a frequência com que a combinação ocorre em relação a um dos itens. Por exemplo, se toda vez que alguém compra leite, eles também compram pão, a confiança é alta.

# Como funciona o algoritmo Apriori

Exemplo:

Transação	Itemset
1	{Leite, Pão, Manteiga}
2	{Leite, Pão}
3	{Pão, Manteiga}

# Como funciona o algoritmo Apriori

Exemplo:

Transação	Itemset
1	{Leite, Pão, Manteiga}
2	{Leite, Pão}
3	{Pão, Manteiga}

Contar  
Frequência dos  
Itens

Leite = 2  
Pão = 3  
Manteiga = 2

Calcular o  
Suporte dos Itens

Leite = 2/3 (67%)  
Pão = 3/3 (100%)  
Manteiga = 2/3  
(67%)

Definir Suporte  
Mínimo

Suporte Mínimo =  
50%

Filtrar Itens com  
Base no Suporte  
Mínimo

Leite > 50%  
(mantido)  
Pão > 50%  
Manteiga > 50%  
(mantido)

Criar  
Combinações de  
Dois Itens

{Leite, Pão}  
{Leite, Manteiga}  
{Pão, Manteiga}

# Como funciona o algoritmo Apriori

## Exemplo:

Transação	Itemset
1	{Leite, Pão, Manteiga}
2	{Leite, Pão}
3	{Pão, Manteiga}

Contar  
Frequência dos  
Itemsets

{Leite, Pão} = 2  
{Leite, Manteiga} = 1  
{Pão, Manteiga} = 2

Calcular o  
Suporte dos  
Itemsets

{Leite, Pão} = 2/3 = 67%  
{Leite, Manteiga} = 1/3 = 33%  
{Pão, Manteiga} = 2/3 = 67%

Filtrar Itemsets  
com Base no  
Suporte Mínimo

{Leite, Pão} > 50% (mantido)  
{Leite, Manteiga} < 50% (removido)  
{Pão, Manteiga} > 50% (mantido)

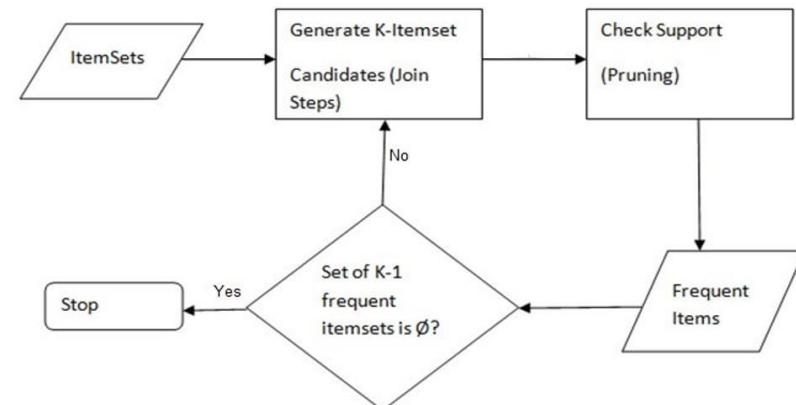
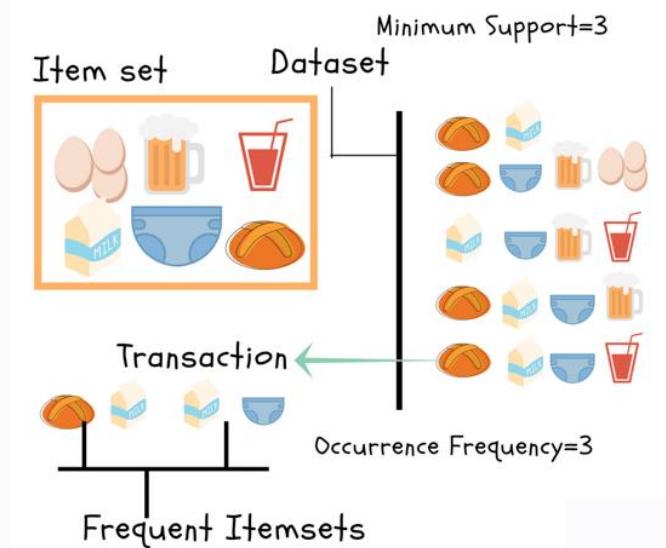
Existem  
Combinações  
Populares ?

Não

Gerar Regras de  
Associação

Leite -> Pão  
Confiança: 2/2 = 100%  
  
Pão -> Manteiga  
Confiança: 2/3 = 67%

# Como funciona o algoritmo Apriori



# Métricas de Algoritmos de Regras de Associação

**Suporte:** Mede a frequência que um itemset aparece na base de dados

$$Suporte(X) = \frac{\text{Número de Transações que tenham } X}{\text{Número Total de Transações}}$$

**Importância:** Identifica itemsets frequentes que são relevantes para a geração de regras de associação. Um suporte baixo pode indicar que o padrão é raro e, portanto, menos interessante.

# Métricas de Algoritmos de Regras de Associação

**Confiança:** Mede a probabilidade de que o item B esteja presente em uma transação que contém o item A.

$$\text{Confiança}(A \Rightarrow B) = \frac{\text{Número de Transações contendo } A \text{ e } B}{\text{Número Total de Transações contendo } A}$$

**Importância:** Avalia a força da regra de associação. Alta confiança indica uma forte relação entre os itens A e B.

# Métricas de Algoritmos de Regras de Associação

**Lift:** Mede a importância da regra de associação, comparando a confiança da regra com a frequência esperada dos itens se fossem independentes.

$$Lift(A \Rightarrow B) = \frac{Confiança\ (A \Rightarrow B)}{Suporte\ (B)}$$

**Maior que 1:** Indica uma associação positiva, (os itens ocorrem juntos mais frequentemente do que o esperado)

**Igual a 1:** Indica que os itens são independentes.

**Menor que 1:** indica uma associação negativa (os itens ocorrem juntos menos frequentemente do que o esperado)

# Métricas de Algoritmos de Regras de Associação

**Convicção:** Avalia a relação entre a confiança da regra e a expectativa de não ocorrência do consequente, considerando o suporte.

$$Convicção(A \Rightarrow B) = \frac{1 - Suporte(B)}{1 - Confiança(A \Rightarrow B)}$$

**Maior que 1:** Indica uma dependência positiva, ou seja, B é mais provável de ocorrer com A do que por acaso.

**Igual a 1:** Indica que A e B são independentes.

**Menor que 1:** Indica uma dependência negativa, ou seja, B é menos provável de ocorrer com A do que por acaso.

# Métricas de Algoritmos de Regras de Associação

**Leverage:** Mede a diferença entre a frequência observada de (A) e (B) ocorrendo juntos e a frequência esperada se (A) e (B) fossem independentes

$$\text{Leverage}(A \Rightarrow B) = \text{Suporte}(A \cap B) - (\text{Suporte}(A) \times \text{Suporte}(B))$$

**Positiva:** A probabilidade de compra de A aumenta a probabilidade de compra de B, mais do que seria esperado por acaso

**Zero:** Eles ocorrem juntos com a mesma frequência que ocorreria por acaso.

**Negativa:** A probabilidade de compra de A diminui a probabilidade de compra de B, mais do que seria esperado por acaso

# Métricas de Algoritmos de Regras de Associação

Use [Lift](#) para uma comparação relativa que mede a força da associação em relação à independência dos itens. Ideal para avaliar a relevância prática das regras.

Use [Convicção](#) para medir a dependência entre itens, especialmente quando lida com itens raros e deseja uma medida da certeza da ocorrência do consequente. Ideal para regras com alta confiança.

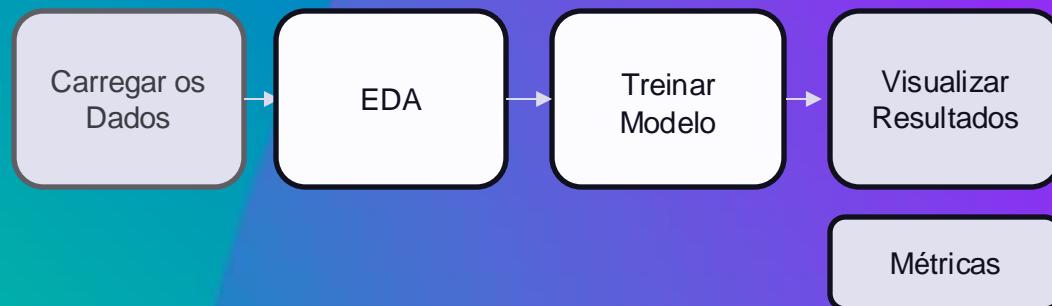
Use [Leverage](#) para uma medida direta da diferença entre a ocorrência observada e a esperada. Ideal para dados平衡ados.

# Projeto - Apriori

Um **marketplace** deseja construir um **sistema mais eficiente de recomendação de produtos** e, para isso, quer usar um recorte do histórico de transações realizadas pelos clientes. Este histórico de transações contém informações como o departamento do produto comprado e a quantidade de itens comprados.

Desta forma, para que seja possível desenvolver este sistema, iremos realizar uma análise de market basket através do **algoritmo Apriori** que irá nos gerar algumas regras de associação que podem ser utilizadas para este modelo de recomendação.

# Estrutura do Projeto



# Code Time ...



Rocketseat © 2023

Todos os direitos reservados

[rocketseat.com.br](https://rocketseat.com.br)

