



UNIVERSIDADE FEDERAL DO PARÁ
NÚCLEO DE DESENVOLVIMENTO AMAZÔNICO EM ENGENHARIA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

Laciene Melo Garcia

**Avaliação de Desempenho de Algoritmos de Mineração de Dados e
Simulação de Monte Carlo na Descoberta de Tendências no Hambre
Delivery**

Tucuruí – Pará

2025

Laciene Melo Garcia

**Avaliação de Desempenho de Algoritmos de Mineração de Dados e
Simulação de Monte Carlo na Descoberta de Tendências no Hambre
Delivery**

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Núcleo de Desenvolvimento Amazônico em Engenharia, da Universidade Federal do Pará, com requisito para a obtenção do título de Mestre em Computação Aplicada.

Orientador: Prof. Dr. Fabrício de Souza Farias

Tucuruí – Pará

2025

**Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD
Sistema de Bibliotecas da Universidade Federal do Pará
Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a) autor(a)**

G216a Garcia, Laciene Melo.
Avaliação de Desempenho de Algoritmos de Mineração de
Dados e Simulação de Monte Carlo na Descoberta de Tendências
no Hambre Delivery / Laciene Melo Garcia. — 2025.
87 f. : il. color.

Orientador(a): Prof. Dr. Fabrício de Souza Farias
Dissertação (Mestrado) - Universidade Federal do Pará, Núcleo
de Desenvolvimento Amazônico em Engenharia, Mestrado
Profissional em Computação Aplicada, Tucuruí, 2025.

1. Mineração de Dados. 2. Monte Carlo. 3. Food Service.
4. Descoberta de Conhecimento. 5. Regras de Associação. I.
Título.

CDD 004

Laciene Melo Garcia

**Avaliação de Desempenho de Algoritmos de Mineração de Dados e
Simulação de Monte Carlo na Descoberta de Tendências no Hambre
Delivery**

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Núcleo de Desenvolvimento Amazônico em Engenharia, da Universidade Federal do Pará, com requisito para a obtenção do título de Mestre em Computação Aplicada.

Aprovada em 14 de Agosto de 2025.

BANCA EXAMINADORA:

Prof. Dr. Fabricio de Souza Farias

Orientador (PPCA/NDAE/UFPA)

Prof. Dr. Daniel da Conceição Pinheiro

Examinador Interno (PPCA/NDAE/UFPA)

Prof. Dr. Elton Rafael Alves

Examinador Interno (PPCA/NDAE/UFPA)

Prof. Dr. Allan Barbosa Costa

Examinador Externo (UFPA)

Prof. Dr. Andson Marreiros Balieiro

Examinador Externo (UFPE)

Dedico este trabalho ao meu filho que está chegando, mesmo antes de nascer, ele representa uma grande fonte de motivação e inspiração. Que este esforço represente o desejo profundo de construir um futuro melhor, e que um dia sirva de exemplo para que ele siga com coragem, ética e dedicação o caminho que escolher trilhar.

AGRADECIMENTOS

Primeiramente, agradeço a Deus, todo-poderoso, por me conceder a vida, a força e a sabedoria necessárias para realizar este trabalho e chegar até aqui.

Ao meu esposo, Igor Melo Garcia, por me incentivar, por estar ao meu lado nos momentos difíceis e por acreditar na minha capacidade, mesmo quando eu mesma duvido. Ele enfrenta comigo cada desafio e compartilha momentos de felicidade, especialmente agora, com a espera do nosso bebê.

Aos meus pais, Lourival e Odeth, por sempre me apoiarem na vida acadêmica, por garantirem tudo de que precisei e por apresentarem minha vida a Deus em oração. Ao meu irmão, Lailson, por ser um exemplo de integridade na vida e de dedicação aos estudos, mostrando que nunca é tarde para alcançar seus sonhos. Aos meus familiares, tias e amigos, pelo incentivo e pelas orações em todos os momentos.

Ao meu orientador, Dr. Fabrício de Souza Farias, pela constante disposição em ajudar, pelo tempo e esforço dedicados, e por me motivar a ir além, reconhecendo e desenvolvendo meu potencial. Suas orientações foram essenciais para meu crescimento intelectual e profissional.

Ao Laboratório de Programação Extrema (LABEX), pelo apoio ao longo de toda a minha trajetória acadêmica. Aos colegas Leonardo Gonçalves, Joiner Sá e Keverton Gonçalves, por sua colaboração durante o mestrado e na elaboração deste trabalho.

Aos professores do PPCA, pela excelência do conhecimento científico transmitido ao longo do curso.

A Deus toda honra e toda glória eternamente!

RESUMO

O setor *food service* agiliza as transações e contribui para melhorar a qualidade dos produtos e serviços oferecidos, ocasionando crescimento contínuo e valorização de compras via *marketplace*. Com a maior adesão da população à digitalização do *food service*, novas informações e conhecimentos podem ser gerados a partir da análise de tendências extraídas de bases de dados oriundas de transações comerciais. Para a coleta de dados, este setor tem adotado *apps* especializados, que se mostram viáveis para usuários que buscam atendimento em plataformas digitais. Além disso, o uso Inteligência Artificial (IA), em conjunto com os *apps*, tem reconfigurado a forma como os negócios operam, representando uma nova tendência para empresas que oferecem produtos e serviços na modalidade online. Diante desse cenário, surge a necessidade de considerar soluções baseadas em software que apliquem IA para sistematizar a análise de tendências em dados coletados. Por essa razão, este trabalho propõe uma investigação por meio de simulações que avaliam o desempenho computacional, combinando o método Monte Carlo e diferentes algoritmos de mineração de dados, a fim de identificar o modelo mais adequado para o suporte a decisões na gestão do setor de *food service* via *apps*. Para validar a eficácia das simulações, dados reais foram coletados nas lojas parceiras da plataforma Hambre Delivery. Durante as simulações, foram analisados os algoritmos FP-Growth, FP-Max, Apriori e Eclat, considerando escalabilidade, tempo de execução e uso de memória como critérios de desempenho. Os resultados revelaram que o algoritmo Eclat é mais indicado para conjuntos pequenos e de baixa complexidade, enquanto FP-Growth e FP-Max são escaláveis para grandes volumes de dados, sendo mais eficientes quanto ao tempo de execução e ao uso de memória. Além disso, as 27 regras de associação geradas revelaram tendências relevantes, mostrando que a aplicação do Monte Carlo gera padrões mais precisos e confiáveis.

Palavras-chave: Mineração de Dados. Monte Carlo. Food Service. Descoberta de Conhecimento. Regras de Associação. Tendências de Venda.

ABSTRACT

The food service sector streamlines transactions and contributes to the improvement of product and service quality, resulting in continuous growth and increased value of purchases made through marketplaces. With the growing adoption of food service digitalization, new information and insights can be derived from trend analyses based on databases generated from commercial transactions. To collect such data, the sector has implemented specialized applications that have proven effective for users seeking services through digital platforms. Furthermore, the integration of Artificial Intelligence (AI) with these applications has reshaped business operations, representing an emerging trend among companies offering products and services online. In this context, there is an increasing need to consider software-based solutions that leverage AI to systematize the analysis of trends within collected data. This study, therefore, proposes an investigation through simulations designed to evaluate computational performance by combining the Monte Carlo method with various data mining algorithms, aiming to identify the most suitable model to support decision-making in managing the food service sector via apps. To validate the effectiveness of the simulations, real-world data was collected from partner stores on the Hambre Delivery platform. The simulations assessed the FP-Growth, FP-Max, Apriori, and Eclat algorithms, taking into account scalability, execution time, and memory usage as performance metrics. The results indicate that the Eclat algorithm is more appropriate for small, low-complexity datasets, whereas FP-Growth and FP-Max scale well to large data volumes and demonstrate superior efficiency in both execution time and memory consumption. Additionally, the 27 association rules generated revealed significant trends, demonstrating that the application of the Monte Carlo method produces more accurate and reliable patterns.

Keywords: Data Mining, Monte Carlo, Food Service, Knowledge Discovery, Association Rules, Sales Trends.

LISTA DE FIGURAS

Figura 1 - Fluxo das Etapas no Processo de KDD	35
Figura 2 – Adição do Registro de transações na FP-Tree.	39
Figura 3 - Processo de Geração dos Conjuntos de Itens Candidatos.	40
Figura 4 - Telas de Apresentação do App Hambre Delivery.	44
Figura 5 - Telas de Apresentação do App Hambre Loja.	45
Figura 6 - Funcionamento do Firebase Realtime Database na Plataforma Hambre Delivery.	46
Figura 7 - Formato de Armazenamento dos Dados no Realtime Database.....	47
Figura 8 - Dados Brutos em Formato JSON Extraídos do Firebase.....	48
Figura 9 - Arquitetura Geral da Proposta de Software para a Plataforma Hambre Delivery..	51
Figura 10 – Base de Dados em Formato Transacional.....	52
Figura 11 - Trecho de Código Responsável pela Padronização da Escrita do Refrigerante Coca-Cola.....	53
Figura 12 - Resultado da Conversão para o Formato Horizontal.....	54
Figura 13 - Resultado da Conversão para o Formato Vertical.	54
Figura 14 - Resultados das Simulações Exploratórias para a Extração de Itens Frequentes. .	64
Figura 15 - Resultados das Simulações Exploratórias para a Geração de Regras de Associação.	66
Figura 16 - Resultados da Extração de Itens Frequentes Relativos ao Tempo de Execução e Uso de Memória nas Simulações de Sensibilidade.	68
Figura 17 - Resultados das Simulações de Sensibilidade para a Geração de Regras de Associação.....	71

LISTA DE TABELAS

Tabela 1 – Principais Algoritmos e Técnicas de Mineração de dados Adotados na Literatura	28
Tabela 2 – Parâmetros de Configuração das Simulações de Monte Carlo	58
Tabela 3 – Parâmetros Adotados pelos Algoritmos nas Simulações de Monte Carlo.....	60
Tabela 4 - Tempo Acumulado pelos Algoritmos nas Simulações Exploratórias de Extração de Itens Frequentes	64
Tabela 5 - Quantidade de Itens Frequentes Obtidos pelos Algoritmos FP-Growth, FP-Max, Apriori e Eclat nas Simulações Exploratórias	65
Tabela 6 - Quantidade de Regras de Associação Obtidas pelos Algoritmos FP-Growth e Apriori nas Simulações Exploratórias	67
Tabela 7 - Tempo Acumulado pelos Algoritmos nas Simulações de Sensibilidade para a Extração de Itens Frequentes	69
Tabela 8 - Memória Média Utilizada pelos Algoritmos nas Simulações de Sensibilidade para a Extração de Itens Frequentes	69
Tabela 9 - Quantidade de Itens Frequentes Obtidos pelos Algoritmos FP-Growth, FP-Max, Apriori e Eclat nas Simulações de Sensibilidade	70
Tabela 10 - Tempo Acumulado pelos Algoritmos nas Simulações de Sensibilidade para a Geração de Regras de Associação	71
Tabela 11 - Memória Média Utilizada pelos Algoritmos nas Simulações de Sensibilidade para a Geração de Regras de Associação	72
Tabela 12 - Quantidade de Regras de Associação Obtidas pelos Algoritmos FP-Growth e Apriori	72
Tabela 13 - Regras de Associação Referentes às Categorias Doce e Salgado	73
Tabela 14 - Regras de Associação Referentes à Categoria Bebida Regional.....	74
Tabela 15 - Regras de Associação Referentes à Categoria Pizza.....	75
Tabela 16 - Regras de Associação Referentes à Categoria Refeição	77

LISTA DE SIGLAS

ECLAT	Equivalence Class Transformation
FP-Growth	Frequent Pattern Growth
FP-MAX	Frequent Pattern Maximal Frequent Itemsets
FP-tree	Frequent Pattern Tree
IA	Inteligência Artificial
IoT	Internet of Things
KDD	Knowledge Discovery in Databases
NSGA-II	Non-dominated Sorting Genetic Algorithm II
PPR	Popular Product Recommendation
RFM	Recência, Frequência e Valor Monetário
SECTET	Secretaria de Ciência, Tecnologia e Educação Superior, Profissional e Tecnológica
UFPA	Universidade Federal do Pará
WEKA	Waikato Environment for Knowledge Analysis

SUMÁRIO

1	Introdução	17
1.1	Contexto da Pesquisa.....	17
1.2	Motivação.....	18
1.3	Justificativas	19
1.4	Objetivos	19
1.5	Organização do Trabalho	20
2	Trabalhos Relacionados.....	22
2.1	Introdução.....	22
2.2	Mineração de Regras de Associação no Apoio Estratégico em Vendas	22
2.3	Avaliação da Viabilidade de Algoritmos de Regras Associação	24
2.4	Eficiência Computacional na Mineração de Regras de Associação.....	25
2.5	Aplicação de Algoritmos Baseados em Regras de Associação em Sistemas Web	26
2.6	Conclusão	27
3	Referencial Teórico	29
3.1	Introdução.....	29
3.2	Método de Monte Carlo e Bootstrap	30
3.3	Benchmarking de Algoritmos e Base de Dados	32
3.3.1	Escalabilidade.....	32
3.3.2	Tempo de Execução.....	33
3.3.3	Memória.....	34
3.4	<i>Knowledge Discovery in Databases</i> (KDD)	34
3.5	Mineração de Dados Baseada em Regras de Associação.....	36
3.5.1	Métricas para Avaliação de Regras de Associação	37
3.5.2	Estrutura de Dados Frequent Pattern Tree (FP-Tree).....	38
3.5.3	Algoritmos Baseados em Vetores de Transações.....	39
3.5.4	Algoritmo Apriori.....	39

3.5.5	Algoritmo FP-Growth.....	41
3.5.6	Algoritmo FP-Max	41
3.5.7	Algoritmo Eclat	42
3.6	Conclusão	42
4	Plataforma Hambre Delivery	43
4.1	Introdução.....	43
4.2	A Plataforma Hambre Delivery no Contexto do Baixo Tocantins.....	43
4.2.1	App Hambre Delivery	44
4.2.2	App Hambre Loja	45
4.2.3	Firestore.....	46
4.3	Desafios e Oportunidades.....	48
4.4	Conclusão	49
5	Proposta de Software para Descoberta de Tendências na Plataforma Hambre Delivery.....	50
5.1	Introdução.....	50
5.2	Fases do KDD	51
5.2.1	Coleta de Dados.....	51
5.2.2	Pré-processamento de Dados.....	52
5.2.3	Transformação de Dados	53
5.3	Método de Monte Carlo	55
5.3.1	Minação de Dados	55
5.3.2	Interpretação dos Dados	55
5.4	Conclusão	56
6	Caso de Estudo.....	57
6.1	Introdução.....	57
6.2	Método de Monte Carlo para Análise de Desempenho Computacional	57
6.2.1	Simulações Exploratórias	58
6.2.2	Simulações de Sensibilidade	59

6.3	Parâmetros de Configuração dos Algoritmos de Regras de Associação.....	60
6.4	Experimentos de Desempenho Computacional.....	61
6.4.1	Experimento de Extração de Itens Frequentes	61
6.4.2	Experimento de Geração de Regras de Associação.....	61
6.5	Análise das Tendências de Venda da Plataforma Hambre Delivery.....	62
6.6	Conclusão	62
7	Resultados e Discussão	63
7.1	Introdução.....	63
7.2	Resultados das Simulações Exploratórias	63
7.2.1	Experimento de Extração de Itens Frequentes	64
7.2.2	Experimento de Geração de Regras de Associação.....	65
7.3	Resultados das Simulações de Sensibilidade	67
7.3.1	Experimento de Extração de Itens Frequentes	67
7.3.2	Experimento de Geração de Regras de Associação.....	70
7.4	Resultados das Tendências de Venda da Plataforma Hambre Delivery	73
7.4.1	Salgado e Sobremesa	73
7.4.2	Bebida Regional	74
7.4.3	Pizzaria	75
7.4.4	Refeição	76
7.5	Conclusão	77
8	Conclusão	79
8.1	Considerações Finais.....	79
8.2	Trabalhos Futuros.....	80
8.3	Produções Acadêmicas.....	80
	Referências	82

1 INTRODUÇÃO

1.1 Contexto da Pesquisa

A análise de cesta de mercado representa uma alternativa para o aprimoramento da qualidade dos serviços prestados por micro e grandes empresas que buscam compreender os interesses de compra de seus clientes. Historicamente, as informações obtidas a partir dessas análises possibilitam alinhar os produtos oferecidos aos perfis de consumo dos usuários, favorecendo o crescimento e o sucesso dos negócios (Cui et al., 2025).

Diversas ferramentas de digitalização de dados têm sido adotadas para gerar informações úteis aos empreendedores, desde a implantação de sistemas de gestão comercial de ponto de venda (PDV) em supermercados e comércios até *apps* especializados na comunicação entre clientes, produtos e negócios, mais comuns nos setores de vendas *online*, como o de *food service* (Gupta et al., 2025).

No contexto do setor *food service*, a digitalização é uma realidade para estabelecimentos comerciais de pequeno, médio e grande porte. Essa tendência se deve ao aumento do interesse de usuários por compras via *marketplace*, que agilizam processos, facilitam o controle de custos e melhoram a experiência dos clientes (Bujalance-López et al., 2025). De acordo com Grijalba et al. (2024), estabelecimentos que utilizam ferramentas digitais para estruturar seu processo de gestão apresentam maior domínio sobre o perfil dos clientes e mais preparo para enfrentar crises econômicas, o que permite uma reação mais rápida do que empresas menos digitalizadas.

Outro fator do mundo digital que tem contribuído para as inovações tecnológicas no setor *food service* está relacionado à popularização do uso de Inteligência Artificial (IA). A IA tem transformado a forma como os negócios operam, oferecendo às empresas maior controle de estoque, vendas e canais digitais (Rabhi, Beheshti e Gill, 2025).

Embora existam trabalhos com aplicação prática de software e IA na área de *food service*, ainda há desafios técnicos, pois as soluções desenvolvidas e apresentadas na literatura apresentam lacunas que comprometem o atendimento ao volume de clientes e vendas, em relação ao tempo de resposta, qualidade das informações, velocidade de processamento, entre outros (Liu et al., 2024). Desse modo, é necessário investigar e comparar algoritmos baseados

em IA que sejam adequados a esse propósito, compatíveis com os requisitos de desempenho de *hardware* e *software*, e capazes de fornecer ao usuário informações precisas com o menor custo computacional possível.

Diante disso, esta dissertação apresenta uma das soluções desenvolvidas para o projeto Hambre Delivery, que visa à inclusão digital de empreendedores da região do Baixo Tocantins que enfrentaram redução parcial ou total das vendas durante a pandemia de COVID-19. A plataforma Hambre Delivery foi idealizada na forma de dois apps, o Hambre Delivery e o Hambre Loja, implantados nas cidades de Cametá, Baião, Mocajuba, Oeiras do Pará e Limoeiro do Ajuru, com o apoio de captadores locais responsáveis por credenciar os estabelecimentos do setor de *food service*.

A solução proposta consiste na aplicação de uma técnica híbrida de IA fracamente acoplada (Wurst e Morik, 2007), que combina o método de Monte Carlo e algoritmos de Mineração de Dados baseados em Regras de Associação. Visando determinar o desempenho computacional e garantir mais consistência nos resultados, os algoritmos FP-Growth, FP-Max, Apriori e Eclat foram implementados e comparados quanto ao tempo de execução, escalabilidade e consumo de memória. Além disso, as análises foram conduzidas utilizando a base de dados da plataforma Hambre Delivery, o que possibilitou resultados mais condizentes com a realidade do setor *food service*.

1.2 Motivação

A principal motivação para o desenvolvimento desta solução é sistematizar uma abordagem capaz de auxiliar lojas do setor de *food service*, como as que operam por meio do *app* Hambre Loja, a identificar tendências de vendas e tomar decisões estratégicas mais precisas, ágeis e eficientes no uso de tempo e memória, além de ser escalável para cenários com maior volume de dados.

Mesmo com a aplicação de algoritmos de IA para resolver problemas em cenários isolados, é válido destacar que, no contexto de *food service*, podem ocorrer centenas de milhares de transações por segundo, o que torna as tarefas mais complexas.

1.3 Justificativas

Apesar de existirem diversas soluções de mineração de dados, com foco na melhoria de desempenho e em estratégias de vendas, nenhuma delas propõe uma abordagem híbrida que utilize o método de Monte Carlo como ferramenta para avaliar o desempenho computacional de algoritmos baseados em regras de associação, considerando critérios como eficiência no uso de tempo e memória, além da escalabilidade em cenários com grande volume de dados.

Essa lacuna evidencia a necessidade de uma proposta que relacione os conceitos consolidados nas pesquisas sobre análise de cesta de mercado às abordagens de desempenho computacional (Zhao e Yu, 2024; Wang, 2024), beneficiando lojas e equipes de *marketing* na elaboração de estratégias de vendas alinhadas aos interesses dos clientes e que destaque a importância de aliar uma solução estatística mais abrangente à análise de desempenho para orientar a escolha do algoritmo mais adequado a ser implementado em um sistema.

Outra justificativa desta dissertação é sua relevância para a gestão de vendas, principalmente para as lojas do Baixo Tocantins, onde as tendências do *food service* podem ser gerenciadas via *app* Hambre Loja a partir do algoritmo de regras de associação mais preciso apresentado neste trabalho.

Por fim, destaca-se a importância teórica da solução para a comunidade científica, especialmente nas áreas de computação aplicada, pois os resultados obtidos nesta dissertação demonstram que a combinação do Monte Carlo com algoritmos de mineração de dados baseados em regras de associação representa uma abordagem inovadora para aumentar a precisão da avaliação de desempenho computacional, além de contribuir para pesquisas similares.

1.4 Objetivos

O objetivo geral desta dissertação é propor um modelo de avaliação do desempenho computacional de algoritmos de mineração de dados baseados em regras de associação, utilizando o método de Monte Carlo de forma híbrida e fracamente acoplada, visando identificar a solução mais adequada para suporte à decisão na gestão de vendas da plataforma *food service* Hambre Delivery.

A fim de alcançar o objetivo geral, foram delineados os seguintes objetivos específicos:

- Apresentar uma proposta de *software* que utilize as fases de KDD e o método de Monte Carlo para extrair conhecimento útil a partir das transações de venda da plataforma Hambre Delivery.
- Realizar uma investigação fundamentada nos conceitos de *Benchmarking*, visando avaliar o desempenho computacional dos algoritmos de regras de associação Apriori, FP-Growth, FP-Max e Eclat.
- Definir cenários e configurações para os testes de Monte Carlo, considerando simulações exploratórias e de sensibilidade;
- Determinar se o uso do método de Monte Carlo, aliado à mineração de dados baseada em regras de associação, proporciona maior estabilidade no desempenho computacional e contribui para a identificação do algoritmo mais eficiente;
- Analisar as principais tendências de venda da plataforma Hambre Delivery.

1.5 Organização do Trabalho

Esta dissertação é composta por sete Capítulos e as Referências Bibliográficas. Além deste Capítulo de introdução, o restante do trabalho está estruturado da seguinte forma.

Capítulo 2 – Trabalhos Relacionados: apresenta pesquisas acadêmicas e soluções aplicadas no mercado com abordagens semelhantes à proposta de mineração de dados desta dissertação;

Capítulo 3 – Referencial Teórico: Contextualiza todo o aporte teórico e os conceitos adotados na implementação desta pesquisa;

Capítulo 4 – Plataforma Hambre Delivery: Aborda o papel da plataforma Hambre Delivery, destacando sua relevância, funcionamento e impacto econômico e social no setor de *food service*;

Capítulo 5 – Proposta de Software para a Descoberta de Tendências na Plataforma Hambre Delivery: Contempla as etapas metodológicas propostas para a descoberta de conhecimentos em bases de dados transacionais, englobando uma visão geral sobre algoritmos baseados em regras de associação, bem como técnicas empregadas na análise de desempenho computacional, como KDD, *Bootstrap* e Monte Carlo;

Capítulo 6 – Estudo de Caso: Detalha os parâmetros adotados na configuração dos algoritmos de regras de associação, bem como os experimentos de Monte Carlo para avaliação do desempenho computacional em simulações exploratórias e de sensibilidade;

Capítulo 7 – Resultados e Discussão: Apresenta os resultados do uso de Monte Carlo para determinar o algoritmo de melhor desempenho computacional, além de discutir as tendências de vendas da plataforma Hambre Delivery;

Capítulo 8 – Conclusão: Apresenta as considerações finais, os trabalhos futuros e as produções acadêmicas publicadas durante esta pesquisa.

2 TRABALHOS RELACIONADOS

Este capítulo apresenta pesquisas com abordagens semelhantes à solução em mineração de dados proposta nesta dissertação, estando organizado da seguinte forma. A Seção 2.1 faz uma breve introdução sobre este capítulo e contextualiza seus objetivos. A Seção 2.2 aborda pesquisas que utilizam a mineração de dados como apoio estratégico em vendas. A Seção 2.3 apresenta pesquisas que investigam a viabilidade dos algoritmos de associação. A Seção 2.4 aborda estudos voltados para a análise da eficiência computacional. A Seção 2.5 aborda soluções que aplicam algoritmos de associação em sistemas *web*. E, por fim, a Seção 2.6 encerra o capítulo com suas considerações e conclusões.

2.1 Introdução

A análise de cesta de mercado, principalmente para o refinamento de serviços, representa uma realidade para micro e grandes empresas, que buscam compreender melhor os interesses de compra dos clientes. As informações obtidas a partir dessa análise possibilitam alinhar os produtos oferecidos aos perfis de consumo, favorecendo o uso de soluções em mineração de dados, que proporcionam maior controle e dinamismo na realização desse processo. A literatura apresenta estudos promissores para embasar o entendimento e aplicação dessas soluções.

Nesse sentido, este Capítulo apresenta uma revisão bibliográfica das principais abordagens de análise de cesta de mercado e, baseado nessas abordagens, pretende discutir as temáticas que se assemelham à solução proposta nesta dissertação. O foco não é criticar ou destacar aspectos negativos, mas estimular o desenvolvimento de soluções guiadas por aspectos essenciais de qualidade e eficiência.

2.2 Mineração de Regras de Associação no Apoio Estratégico em Vendas

No contexto da queda de vendas, os algoritmos de associação podem auxiliar lojas e equipes de *marketing* na identificação dos fatores que afetam as vendas e, principalmente, na elaboração de estratégias de *marketing* mais efetivas. Sob esse aspecto, foram encontrados na

literatura pesquisas voltadas para análise de transações em lojas de varejo, franquias e supermercados, o que indica um claro interesse na compreensão de padrões de consumo.

Os autores Pradana et al. (2022) realizaram um estudo de caso para identificar as possíveis causas que levaram à queda nas vendas da franquia de vestuário *Hoyjakarta*. A solução propôs o uso do algoritmo FP-Growth como abordagem de apoio à descoberta de padrões relevantes a partir da análise de 571 registros de transações disponibilizados pela franquia. Vale ressaltar que os resultados foram promissores, fornecendo um panorama geral sobre as tendências de vendas de roupas, tanto em casos de vendas isoladas quanto naquelas que impulsionaram a aquisição de outros produtos, mesmo para vestuários que não despertaram tanto interesse dos clientes.

Josephine e Rajan (2023) investigaram o baixo volume de vendas de uma loja de departamento a fim de identificar as preferências dos clientes e melhorar o direcionamento de promoções. Os autores enfatizaram que a escolha do instrumento de análise levou em consideração três algoritmos de mineração de dados, sendo eles Apriori, FP-Growth e Eclat. Essa escolha favoreceu o algoritmo Apriori por ser amplamente utilizado em pesquisas, enquanto os quesitos observados para análise tomaram como base 16.753 transações de venda. Após o procedimento, foram obtidas 20 regras de associação envolvendo os produtos mais procurados em toda a loja, o que favoreceu a equipe de *marketing* na elaboração de campanhas mais direcionadas.

A pesquisa de Bhagampriyal et al. (2023) propôs a elaboração de um modelo híbrido de recomendações de produtos para supermercados, baseado em dois aspectos essenciais, as avaliações recebidas pelos produtos e as transações de vendas realizadas. Em termos práticos, ambas as informações vieram de uma única base de dados proveniente de um supermercado, que disponibilizou no *Kaggle* 38.765 transações, cada uma contendo o produto vendido associado a uma escala numérica de avaliação variando entre 1 e 5. O resultado e o diagnóstico foram obtidos por meio da análise de três algoritmos, sendo eles o PPR (*Popular Product Recommendation*), que ranqueia os produtos mais bem avaliados, e os modelos de mineração de dados Apriori e FP-Growth, usados para identificar os padrões frequentes de compra. Após os testes, os autores concluíram que, devido à eficiência de execução, a utilização conjunta dos algoritmos PPR e FP-Growth é a alternativa mais vantajosa como modelo híbrido. Vale ressaltar que o PPR se limita a recomendações baseadas em popularidade e, por isso, não é aplicável à análise de padrões complexos realizada pelos algoritmos de regras de associação.

2.3 Avaliação da Viabilidade de Algoritmos de Regras Associação

No que se refere à avaliação da viabilidade dos algoritmos de regras associação, especialmente em cenários que demandam soluções mais imediatas, a literatura consultada apresenta pesquisas conduzidas em ambientes comerciais de pequeno porte, que propõem resolver problemas específicos vivenciados no cotidiano de restaurantes e lojas de depósito/retirada (*drop-off*).

Neste contexto, Hanum e Witanti (2023) realizaram um estudo de caso em um restaurante de comida familiar, o *Bebek Pondok Galih*, que visava automatizar o arranjo de pratos e propor cardápios personalizados com base nos padrões de consumo e preferências dos clientes. Por se tratar de uma pesquisa *in loco*, o restaurante forneceu dados registrados em cinco meses, totalizando 8.225 transações. Após normalização, os dados foram minerados pelo algoritmo Apriori, que proporcionou poucas, mas promissoras recomendações, dispensando a necessidade de arranjos manuais.

A pesquisa de Aji e Devi (2023) surgiu como uma iniciativa para impulsionar os lucros dos restaurantes de *fast food* na cidade de *Gresik*, Indonésia, principalmente para atenuar a incompatibilidade entre os produtos oferecidos e o paladar dos clientes. Para obter uma solução efetiva, os autores propuseram utilizar o algoritmo Apriori para analisar as vendas, a fim de mapear os produtos que atendiam ao perfil dos clientes. Por esse motivo, a coleta de dados abrangeu a maioria dos restaurantes durante um ano, totalizando 48.750 vendas realizadas. Após a análise, foram extraídas 10 regras de associação que indicaram grande preferência por produtos da categoria arroz, bebida, combo, frango, *goceng*¹ e lanche. Esses resultados representam uma referência confiável para o setor de *marketing* criar promoções eficazes.

Os autores Fadilah, Faqih e Permana (2025) propuseram um modelo de recomendação para uma loja *online* que atua como ponto *drop-off*. Seguindo essa tendência, buscaram entender as intenções de compra dos clientes e fornecer informações para melhorar o gerenciamento de estoques. Foram coletados 4.999 registros de transações, incluindo brinquedos, cosméticos, vestuário, livros e material escolar. A extração dos padrões de consumo

¹ *Goceng* são produtos de baixo custo, vendidos em torno de 5.000 rúpias indonésias que inclui itens como *sundae*, casquinha de sorvete, pudim, Hambúrguer crocante, espaguete, sopa cremosa e *perkedel* (bolinho de batata indonésio).

foi realizada pelo algoritmo Apriori, que revelou tendências relevantes nos segmentos de brinquedos, cosméticos e vestuário, suficientes para aprimorar a experiência do cliente e otimizar a gestão desses estoques.

2.4 Eficiência Computacional na Mineração de Regras de Associação

Avaliar a eficiência computacional é uma tarefa complexa, mas que permite considerar diversos aspectos: qual algoritmo utilizar, quais critérios considerar e sob quais perspectivas se deve analisar a base de dados. Embora poucos estudos foquem diretamente no desempenho dos algoritmos, a literatura consultada ainda proporciona pesquisas que abordam a qualidade e a eficiência, promovendo tanto a comparação de algoritmos clássicos quanto a análise de novos modelos.

Wang (2023) propôs o modelo NSGA-II, capaz de lidar com problemas relacionados à alta complexidade computacional, objetivando aliar a análise de cesta de mercado com a criação de um módulo de otimização. Para tornar o procedimento mais realista, foram utilizadas 7.000 transações comerciais fornecidas por uma cafeteria, divididas em duas etapas: na primeira, o algoritmo Apriori foi aplicado para identificar os itens frequentes necessários à inicialização do NSGA-II. Na segunda etapa, os modelos Apriori e NSGA-II foram comparados quanto à qualidade das regras de associação, considerando cenários com 50, 70, 90 e 110 iterações. Os resultados revelaram que o NSGA-II conseguiu melhorar a qualidade das regras de associação geradas, mas sua incapacidade de gerar itens frequentes faz dele uma solução não aplicável em análises de desempenho computacional mais aprofundadas.

Os autores Rosadi e Hasibuan (2024) apresentaram uma análise comparativa entre algoritmos de associação, tendo como foco a escolha da abordagem mais apropriada, conforme as especificidades do conjunto de dados, especialmente os mais robustos. O procedimento de testes foi implementado utilizando 38.765 transações de um supermercado disponibilizadas no *Kaggle*, sendo essencial para validar o desempenho entre os algoritmos Apriori e FP-Growth. Mesmo avaliados sob as mesmas condições, o algoritmo FP-Growth conseguiu lidar melhor com grandes volumes de dados.

Neste sentido, Lawal e Matthew (2024) propuseram um estudo comparativo com os mesmos algoritmos para determinar a ferramenta mais eficiente para a descoberta de padrões em cestas de mercado. Por se tratar de uma pesquisa preliminar, os autores optaram pelo uso de um conjunto de dados reduzido, composto por 4.627 transações de um supermercado,

disponível no repositório *Kaggle*. Outro aspecto observado foi a utilização da plataforma WEKA para realizar a análise comparativa em relação ao tempo de execução, que revelou o algoritmo FP-Growth como mais eficiente.

2.5 Aplicação de Algoritmos Baseados em Regras de Associação em Sistemas Web

O uso de algoritmos de associação, principalmente em sistemas de recomendação ou plataformas *web*, tem se incorporado cada vez mais à realidade de micro e grandes empresas, apresentando muitas oportunidades e desafios para pesquisadores e setores de *marketing*. Nesse contexto, a literatura consultada apresenta pesquisas desenvolvidas com foco nesse tipo de integração, que priorizam ferramentas dinâmicas e automatizadas.

O estudo de Sun et al. (2024) propôs um novo modelo de mineração, nomeado ScaDistFIM, para análise de cestas em grandes volumes de dados, visando solucionar problemas de escalabilidade e flexibilidade característicos da era dos dados massivos. Os autores elaboraram um processo de validação mais amplo, que envolveu sua aplicação em um sistema de análise de cestas de mercado, a fim de comprovar sua flexibilidade em um cenário real. Além disso, foram realizados experimentos com um conjunto de dados sintéticos, nos quais se comparou o desempenho ao do algoritmo Spark FP-Growth. Os resultados revelaram que o ScaDistFIM obteve uma economia de tempo de pelo menos 90%, evidenciando escalabilidade superior frente às limitações do Spark FP-Growth. Entretanto, ambos os modelos operam nativamente em ambientes Spark, o que não se aplica em cenários cujos dados são armazenados em formato não relacional.

Yildiz, Şen e Işik (2023) apresentaram um sistema *web* de recomendação de produtos personalizados, voltado a empresas do varejo de moda. A abordagem combina técnicas de mineração de dados baseadas em agrupamento e regras de associação, que ampliam a análise dos perfis dos clientes. O processo de recomendação envolve mensurar os valores RFM (Recência, Frequência e Valor Monetário) de cada cliente, visando agrupá-los por segmento com o algoritmo K-means. Com base nos *clusters* criados, foram geradas regras de associação específicas por meio do algoritmo Apriori. Vale ressaltar que o sistema foi implementado com dados reais, os quais resultaram em recomendações úteis e alinhadas ao perfil de cada grupo, mas a segmentação de clientes não se aplica em estudos voltados para avaliação do desempenho de algoritmos baseados em regras de associação.

A pesquisa de Yan et al. (2024) propôs um sistema de recomendação de produtos inteligentes baseados em IoT (*Internet of Things*), utilizando o algoritmo Apriori e lógica fuzzy para lidar com o acúmulo de dados em sistemas de comércio inteligente. Essa técnica usa regras de associação para analisar o comportamento de compra dos clientes, com base nos produtos adicionados ao carrinho. As regras são geradas por um sistema fuzzy, que fornece as associações para o algoritmo Apriori, responsável por selecionar os produtos a partir dessas informações. Assim, o método amplia as opções de produtos e recomendações aos usuários.

2.6 Conclusão

Este Capítulo apresentou os trabalhos correlatos aos algoritmos de associação propostos para a descoberta de conhecimento na base de dados da plataforma Hambre Delivery. O intuito foi estabelecer uma relação entre os conceitos consolidados nas pesquisas sobre análise de cesta de mercado e as abordagens de desempenho computacional, de caráter mais amplo, abordadas nesta dissertação.

Os trabalhos relatados na Seção 2.2 demonstram soluções que beneficiam lojas e equipes de *marketing* na elaboração de estratégias de vendas mais alinhadas aos interesses dos clientes. Entretanto, os estudos deixam claro que o processo de escolha dos algoritmos de associação foi limitado, com foco restrito à extração de padrões de venda. Testes mais abrangentes, considerando conjuntos de dados maiores e variações nos parâmetros, foram descartados, além de as comparações entre os algoritmos serem superficiais.

Sob outra perspectiva, a Seção 2.3 evidenciou soluções viáveis para ambientes comerciais de pequeno porte, que contribuíram para a resolução de problemas específicos do cotidiano de restaurantes e lojas. Todas as propostas implementadas utilizaram somente um algoritmo, em virtude da necessidade de obter uma solução imediata, o que faz com que o escopo dos modelos não se aplique a contextos variados e complexos.

As pesquisas apresentadas na Seção 2.4 demonstraram a importância de considerar o desempenho computacional antes de escolher qual algoritmo de associação implementar em um sistema. Algumas levam em consideração aspectos como escalabilidade ou tempo de execução, ou até mesmo ambos, mas não observam o uso de memória, essencial para o controle de recursos.

Quanto às soluções conduzidas na Seção 2.5, pode-se perceber que integrar os modelos de associação em sistemas *web* foi satisfatório, principalmente por viabilizar ferramentas dinâmicas e automatizadas para uso empresarial. Além disso, foram apresentadas abordagens inovadoras, como novos algoritmos e a aplicação da lógica *fuzzy*.

Apesar dos trabalhos citados apresentarem diversas soluções para mineração de dados e estratégias de vendas, nenhum deles propõe o uso do MMC para avaliar o desempenho computacional de algoritmos de mineração de associação como ferramentas de suporte à decisão na gestão de vendas. Essa lacuna evidencia a necessidade de uma proposta prática e fundamentada, voltada à avaliação da eficácia dos algoritmos de associação nesses aspectos, a fim de permitir uma integração mais eficiente em plataformas de *e-commerce*.

A Tabela 1 apresenta a frequência de uso dos principais algoritmos e técnicas de mineração de dados adotados nos trabalhos mencionados. Além disso, foram atribuídos conceitos referentes à ocorrência de cada algoritmo nos estudos, em que P indica presença, F indica ausência e NA indica não aplicável.

Tabela 1 – Principais Algoritmos e Técnicas de Mineração de dados Adotados na Literatura

Referência	Algoritmos de Mineração de Dados									Técnicas de Suporte			
	Apriori	Eclat	FP-Growth	FP-Max	K-means	NSGA-II	PPR	ScadistFM	Spark FP-Growth	Monte Carlo	KDD	Bootstrap	Benchmarking
Pradana et al. (2022)	F	F	P	F	F	F	F	F	F	F	F	F	F
Josephine e Rajan (2023)	P	P	P	F	F	F	F	F	F	F	F	F	F
Bhagampriyal et al. (2023)	P	F	P	F	F	F	P	F	F	F	F	F	P
Hanum e Witanti (2023)	P	F	F	F	F	F	F	F	F	F	F	F	F
Aji e Devi (2023)	P	F	F	F	F	F	F	F	F	F	F	F	F
Fadilah, Faqih e Permana (2025)	P	F	F	F	F	F	F	F	F	F	F	F	F
Wang (2023)	P	F	F	F	F	P	F	F	F	F	F	F	P
Rosadi e Hasibuan (2024)	P	F	P	F	F	F	F	F	F	F	F	F	P
Lawal e Matthew (2024)	P	F	P	F	F	F	F	F	F	F	F	F	F
Sun et al. (2024)	F	F	F	F	F	F	F	P	P	F	F	F	P
Yildiz, Şen e Işık (2023)	P	F	F	F	P	F	F	F	F	F	F	F	F
Yan et al. (2024)	P	F	F	F	F	F	F	F	F	F	F	F	F
Dissertação da Autora (2025)	P	P	P	P	NA	NA	NA	NA	NA	P	P	P	P

Fonte: Elaborado pela autora.

3 REFERENCIAL TEÓRICO

Este Capítulo contextualiza os principais aportes teóricos encontrados na literatura e adotados na implementação da solução híbrida fracamente acoplada do método Monte Carlo e mineração de dados baseada em regras de associação proposta nesta dissertação. A Seção 3.1 apresenta uma breve introdução sobre cada assunto, associado ao autor que o utiliza na literatura. A Seção 3.2 apresenta o método de Monte Carlo como ferramenta para a realização de simulações. A Seção 3.3 aborda conceitos de *Benchmarking* sobre a comparação de algoritmos e bases de dados, incluindo critérios como escalabilidade, tempo de execução e memória. A Seção 3.4 aborda os conceitos envolvidos em KDD. A Seção 3.5 descreve as abordagens de mineração de dados focadas em associação, tratando de métricas de desempenho, estruturas de dados e os principais algoritmos. E, por fim, a Seção 3.6 apresenta as conclusões deste Capítulo.

3.1 Introdução

Há muitas abordagens de mineração de dados discutidas na literatura, principalmente no nicho de vendas. Assim, a compreensão de padrões em marketplaces tornou-se prioridade, especialmente quando a proposição de novos serviços está associada às preferências dos clientes (Kansal et al., 2023). Cabe ao empreendimento buscar soluções de software capazes de extrair conhecimento útil em grandes conjuntos de dados (Ritonga, Juledi e Mutia, 2024).

Neste sentido, na literatura foram encontrados métodos estatísticos cuja combinação fracamente acoplada pode resultar em novos *insights* a partir de dados coletados. Dentre eles, serão discutidos conceitos sobre o método de simulação de Monte Carlo (Hastings, 1970), o uso da técnica *Bootstrap* (Chernick, 2011) para geração de dados sintéticos, análises comparativas com base na técnica de *Benchmarking* (Hall e Holmes, 2003) e avaliação de critérios como escalabilidade (Kumar e Gupta, 1991), tempo de execução (Iverson, Ozguner e Potter, 2002) e uso de memória (Coffman e Weaver, 2014). Além disso, no contexto da descoberta de conhecimento em bases de dados, serão considerados conceitos baseados em Arenas, Paucar e Arias (2024) e Fayyad, Piatetsky-Shapiro e Smyth (1996), a partir da abordagem KDD.

Em termos de mineração de dados baseada em regras de associação (Fayyad e Stolorz, 1997), foram encontradas pesquisas que abordam o uso de métricas de associação (Lin, Tseng e Su, 2002), a estrutura de dados FP-Tree para otimização de tarefas (Grahne e Zhu, 2005) e a definição de parâmetros de suporte e confiança (Zaki, 2000). Destacam-se também a aplicação de algoritmos para extrair itens frequentes e gerar regras de associação, como o Apriori (Agrawal e Srikant, 1994) e o FP-Growth (Han, Kamber e Pei, 2012), outros que focam apenas na extração de itens frequentes máximos, como o FP-Max (Hu et al., 2008), ou ainda, utilizam exclusivamente a interseção de listas de transações para extrair itens frequentes, como o algoritmo Eclat (Zaki et al., 1997).

Conforme observado na literatura, diversos estudos têm se dedicado à aplicação de técnicas de IA em problemas práticos. No entanto, muitas dessas soluções adotam abordagens diretas, sem adaptações ao contexto analisado, o que pode implicar custos computacionais e levar a interpretações simplificadas ou imprecisas da realidade representada pelos dados. Vale ressaltar que esta dissertação propõe integrar os métodos de Monte Carlo, *Bootstrap*, *Benchmarking* e mineração de dados para avaliar o desempenho computacional de algoritmos de IA baseados em regras de associação em cenários de *food service*.

3.2 Método de Monte Carlo e Bootstrap

No contexto de simulações, o método de Monte Carlo foi proposto como uma abordagem estatística para a resolução de problemas que envolvem incerteza e variabilidade. Sua consistência é útil em ambientes computacionais complexos, que exigem a exploração de diversas características para a tomada de decisão. Assim, a realização de múltiplas varreduras combinada a métodos estatísticos robustos pode gerar estimativas confiáveis (Hastings, 1970).

No que se refere à aplicação do método, todo o seu funcionamento baseia-se na definição de cenários que simulam a condição real do ambiente de investigação. Essa abordagem deve estar inserida em uma perspectiva que permita a aplicação de cenários variados, compostos por n iterações ($N_ITERATIONS$). Tais aspectos favorecem uma avaliação mais ampla em termos de comparação, resultando em estimativas mais precisas e consistentes. Por essa razão, o Monte Carlo pode ser integrado a técnicas de reamostragem para a geração de dados sintéticos, como o *Bootstrap*.

O *Bootstrap* consiste em uma abordagem estatística voltada à geração de dados sintéticos, seja para criar um novo conjunto ou ampliar dados originais, permitindo adaptações

ao cenário de pesquisa e possibilitando a aplicação de amostragem aleatória com variação controlada, mesmo em contextos que exigem uma base de dados mais volumosa (Chernick, 2011). A técnica viabiliza a inferência em situações de alta dimensionalidade e a estimativa de intervalos de confiança que dispensam suposições de parâmetros, sendo essencial para a avaliação da variabilidade em testes mais profundos (Chen, Huang e Yang, 2025).

A aplicabilidade combinada dos métodos de Monte Carlo e *Bootstrap* leva em consideração as especificidades da base de dados. Um conjunto de transações do setor de *food service* pode apresentar características únicas que influenciam diretamente sua composição, como as categorias de venda. Por isso, é fundamental definir previamente os parâmetros que preservem essas características na geração dos dados sintéticos.

O processo de geração da base sintética começa com a definição dos parâmetros *N_SAMPLE* e *CARDINALITY*. O primeiro representa a quantidade de transações a serem geradas, enquanto o segundo indica os limites mínimo (*MIN_ITENS*) e máximo (*MAX_ITENS*) de itens por transação. A amostragem aleatória com variação controlada utiliza uma lista com todos os itens únicos de cada categoria de venda, juntamente com as transações originais (*BASE_ORIGINAL*).

Durante a geração por categoria, uma transação original é selecionada aleatoriamente e expandida com novos itens até atingir um número dentro do intervalo definido por *CARDINALITY*. Esse processo se repete até alcançar a quantidade de transações especificada em *N_SAMPLE*. Ao final, as transações sintéticas são agregadas e embaralhadas para compor o conjunto sintetizado.

Após a obtenção do conjunto de transações sintético, as simulações de Monte Carlo são realizadas conforme o número de repetições *N_ITERATIONS* definido para cada cenário. Em cada iteração, uma nova transação é selecionada aleatoriamente, com o uso do parâmetro *RANDOM_STATE* para manter o controle da aleatoriedade durante a seleção das amostras. As amostras são então submetidas a algoritmos de mineração de dados, cujo desempenho é avaliado por meio de medidas estatísticas como média, desvio padrão e margem de erro.

Cabe à média fornecer uma estimativa equilibrada do resultado, enquanto o desvio padrão avalia a dispersão dos valores em relação a ela. Em termos mais amplos, a margem de erro representa a faixa ao redor da estimativa onde se espera que o valor real esteja. No contexto da mineração de dados, a aplicação dos métodos de Monte Carlo e *Bootstrap* permite

variabilidade tanto na repetição das simulações quanto no volume de dados, fatores fundamentais para a obtenção de resultados mais precisos e confiáveis (Luhayb, 2023).

3.3 Benchmarking de Algoritmos e Base de Dados

A aplicação do *Benchmarking*, a partir da combinação dos métodos Monte Carlo e *Bootstrap*, constitui uma abordagem eficaz para solucionar um problema específico. No âmbito da mineração de dados, é amplamente reconhecido que existem múltiplos tipos de algoritmos, sendo os classificadores apenas uma parte dos métodos disponíveis. Por isso, o desempenho de cada um pode variar conforme a natureza do problema analisado, ou seja, não há uma solução capaz de resolver todos os problemas da forma mais eficiente (Shen et al., 2024). É necessário investigar, por meio de simulações, a eficiência computacional desses algoritmos, considerando aspectos como uso de processamento, memória e outros parâmetros relevantes (Eckman, Hendersone e Shashaani, 2023).

Em uma pesquisa de cunho amplo sobre mineração de dados, diversos algoritmos podem ser incorporados. Muitas dessas abordagens têm características semelhantes, mas também singularidades, e apenas o conhecimento prévio dos métodos não é suficiente para escolher o mais condizente com o foco da investigação (Papakyriakou e Barbounakis, 2022).

A adoção de métodos estatísticos como Monte Carlo e *Bootstrap* em simulações pode ampliar a representatividade dos resultados e reforçar a confiabilidade na análise dos algoritmos de IA. Por esse motivo, é essencial avaliar as possibilidades sob aspectos consistentes, considerando uma visão geral das abordagens. Nesse panorama, o *Benchmarking* atua como uma solução teórica comparativa que propõe critérios de desempenho para avaliar algoritmos e conjuntos de dados (Hall e Holmes, 2003).

Para essa pesquisa, os critérios considerados incluem escalabilidade, tempo de execução e uso de memória. A forma como são aplicados permite identificar vantagens e limitações dos algoritmos, fornecendo *insights* que garantem uma escolha baseada em dados empíricos.

3.3.1 Escalabilidade

A avaliação do desempenho de algoritmos e conjuntos de dados quanto ao aumento da demanda de processamento é amplamente considerada em estudos que preveem o crescimento contínuo dos dados. Assim, a escalabilidade envolve a execução de simulações que incluem o

aumento no número de transações, volume de dados ou quantidade de usuários simultâneos (Kumar e Gupta, 1991).

Esse comportamento pode ser representado por modelos de desempenho (tempo de resposta, uso de memória, volume de dados), derivados da Lei de *Little* ou de filas (como M/M/1), que permitem identificar gargalos e pontos de saturação (Jain, 1991). A equação (1) representa o modelo de escalabilidade sob carga.

$$R(n) = \frac{n}{C - n} \quad (1)$$

Dada a equação, a escalabilidade sob carga é expressa pelo número de requisições simultâneas n , em relação à capacidade máxima do sistema antes da saturação C . Com isso, é possível mensurar o crescimento de $R(n)$ conforme n se aproxima de C , possibilitando avaliar o impacto no desempenho de cada abordagem em velocidade de escalonamento, mediante a integração com um método que ofereça suporte a esse tipo de análise.

Além disso, as simulações devem ser inicializadas com diferentes configurações que englobem as métricas necessárias ao funcionamento dos algoritmos. Estabelecer esse controle proporciona o cenário ideal para analisar e mensurar como uma solução lida com o aumento da carga de trabalho, seja no uso de memória ou tempo de execução.

3.3.2 Tempo de Execução

O tempo de execução é uma medida de avaliação igualmente eficiente às demais, sendo sua coleta essencial para análise comparativa em termos de velocidade, pois abrange tanto o processamento dos dados quanto a geração dos resultados (Iverson, Ozguner e Potter, 2002).

O processo de medir o tempo de execução de um algoritmo é realizado por meio do registro do tempo ao início e final da execução. Essa tarefa pode ser desempenhada por meio de funções de temporização presentes na maioria das linguagens de programação. Assim, é possível calcular o tempo absoluto, sendo este o resultado da diferença entre o tempo final T_f e o inicial T_i , conforme expresso na equação (2).

$$T_{abs} = T_f - T_i \quad (2)$$

Considerando que o foco de análise mude, sendo necessário avaliar algoritmos em cenários de simulação variados com diferentes números de iterações, a média absoluta de cada

algoritmo passa a ser a mais adequada, pois ajuda a suavizar as variações que podem ocorrer devido a fatores externos, a equação (3) expressa o cálculo da média absoluta.

$$MA = \frac{1}{n} \sum_{i=1}^n T_i \quad (3)$$

Desse modo, a medida é um somatório do tempo absoluto T_i em cada iteração, em razão do total de n iterações da simulação.

A partir desse contexto, a aplicação de ambas as métricas é relevante para entender a eficiência prática, possibilitando a escolha mais condizente para aplicações específicas que exigem um processamento eficiente no quesito velocidade.

3.3.3 Memória

A memória é um recurso importante para qualquer sistema com alta demanda por processamento de dados. A falta de padronização no desenvolvimento de uma abordagem pode esgotar os recursos, causando falhas ou prejudicando o desempenho (Coffman e Weaver, 2014). O ideal é adotar estratégias que assegurem uma execução rápida e eficiente.

O processo de avaliação começa com o monitoramento do uso de memória durante a execução do algoritmo, principalmente para identificar as áreas de alto consumo. A partir disso, é possível propor técnicas de otimização para minimizá-lo. Dentre elas, estruturas de dados temporárias ou permanentes para armazenar informações de forma otimizada e comprimida, como a FP-Tree, por exemplo. Além disso, a análise pode ser estendida para a comparação entre diferentes algoritmos, considerando o consumo de memória RAM, cache, virtual, etc.

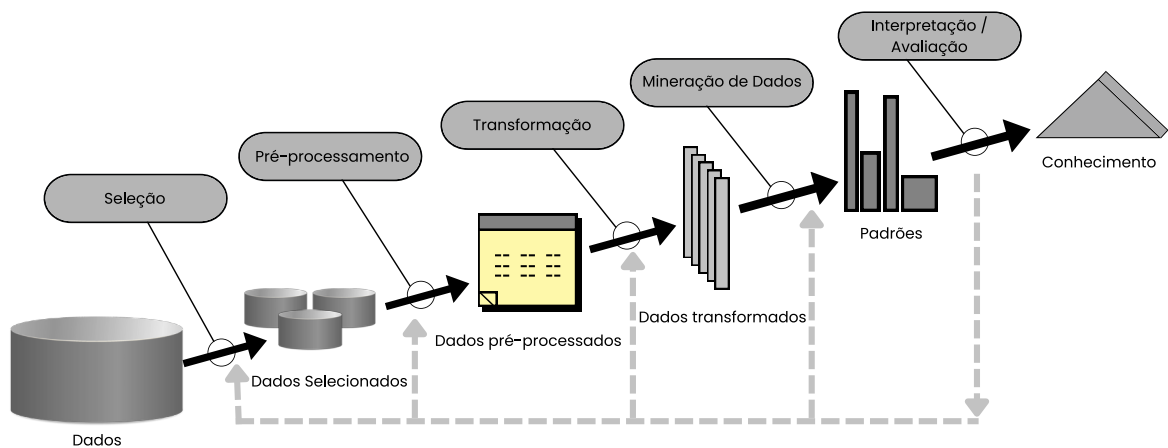
Deste modo, o conhecimento obtido sobre as abordagens analisadas oferece os critérios necessários para embasar a escolha do algoritmo que mais se alinha a esse recurso, possibilitando aliar eficiência à capacidade de processar grandes volumes de dados.

3.4 *Knowledge Discovery in Databases (KDD)*

O uso de técnicas de mineração de dados, que abrangem a descoberta de conhecimentos em grandes volumes de dados, revela a importância de um controle sistemático no tratamento dos dados até a identificação de padrões válidos (Arenas, Paucar e Arias, 2024).

Nessa perspectiva, a técnica de KDD proposta por Fayyad, Piatetsky-Shapiro e Smyth (1996) se destaca por sua capacidade de lidar com grandes volumes de dados. Em um processo minucioso, os dados são tratados para estimular a descoberta de padrões ocultos, especialmente em dados brutos, por meio de etapas que agregam robustez aos métodos de análise e possibilitam a extração de informações relevantes. A Figura 1 ilustra o fluxo de análise do KDD.

Figura 1 - Fluxo das Etapas no Processo de KDD



Fonte: Fayyad, Piatetsky-Shapiro, Smyth (1996).

Inicialmente, os dados se encontram em sua forma bruta, com a presença de informações ruidosas ou fora do escopo da pesquisa. Por isso, a primeira etapa do método propõe a Seleção dos dados de interesse, alinhando a descoberta de conhecimento ao escopo de análise.

Usualmente, as técnicas de seleção de amostragem e de atributos são implementadas para identificar os dados que atendem a um contexto de investigação. A seleção de amostragem permite extrair amostras que melhor representam o escopo da análise (Han et al., 2012), enquanto a seleção de atributos estabelece critérios para aceitar ou rejeitar dados com base em características específicas (Zargari e Voorhis, 2012), como um status que pode indicar situações pendente, em andamento, cancelada ou concluída.

Mesmo após a *Seleção*, pode haver inconsistências no formato dos dados, que interferem na validade da análise. Por isso, na etapa de Pré-processamento, os dados passam por uma limpeza minuciosa, que envolve a identificação e correção de inconsistências, tratamento de valores ausentes, remoção de duplicatas e padronização de formatos. A identificação de inconsistências utiliza restrições e regras lógicas para detectar e corrigir erros (Rammelaere e Geerts, 2020). O tratamento de valores ausentes, remoção de duplicatas e

padronização incluem a imputação de dados faltantes por média, mediana ou algoritmos específicos, exclusão de registros repetidos para evitar distorções na análise e uniformização de unidades, datas, nomes e outros formatos (Lara et al., 2014).

Uma vez normalizados, os dados prosseguem para a etapa de Transformação, onde são convertidos para um formato adequado à aplicação dos modelos de mineração. As técnicas mais utilizadas estão relacionadas à padronização, redução de dimensionalidade, conversão de tipos e transformação baseada em lógica. A padronização compreende o ajuste da escala dos dados para garantir comparabilidade entre atributos (Maazalahi e Hosseini, 2024). A redução de dimensionalidade consiste em transformar o espaço de atributos para reduzir a complexidade (Hegland, 2001). A conversão de tipos é a transformação de dados categóricos em numéricos ou vice-versa, enquanto a transformação baseada em lógica engloba o uso de regras descritivas para preparar dados complexos (Lara et al., 2014).

Posteriormente, os dados transformados passam para a etapa de Mineração de Dados, na qual são identificados padrões no conjunto normalizado. Esse processo pode variar conforme o foco da abordagem, seja classificação, regras de associação, agrupamento, entre outros. Além disso, um ou mais algoritmos podem ser aplicados em análises exploratórias ou mais complexas. No contexto do setor de *food service*, algoritmos baseados em regras de associação podem ser úteis para extrair padrões de venda, especialmente quando há ausência de *feedback* do cliente sobre um produto. A Seção 2.5 apresentará os algoritmos utilizados neste estudo.

Por fim, os padrões descobertos seguem para a etapa de Interpretação e Avaliação, responsável por compreender e verificar os resultados extraídos durante o processo de mineração de dados. A Interpretação envolve traduzir esses padrões ou modelos em informações compreensíveis para especialistas, a fim de analisar sua relevância e aplicabilidade no escopo de interesse. Enquanto a Avaliação utiliza métricas apropriadas, como precisão, acurácia e sensibilidade, além da comparação com *Benchmarks*, para garantir que os padrões descobertos sejam estatisticamente significativos, úteis e robustos.

3.5 Mineração de Dados Baseada em Regras de Associação

Como exposto em KDD, a mineração de dados constitui uma etapa central no processo de análise (Fayyad e Stolorz, 1997). Há diversas abordagens nas quais as pesquisas se baseiam para avaliar aspectos relevantes. Dentre elas, os algoritmos de regra de associação possibilitam

a análise de características específicas, utilizando métricas para identificação de padrões e estruturas de dados otimizadas.

3.5.1 Métricas para Avaliação de Regras de Associação

A validação dos padrões extraídos por algoritmos de regras de associação requer o uso de métricas para determinar sua frequência e força. Por isso, em uma análise contínua são aplicadas métricas de avaliação de suporte, confiança e *lift* (Lin, Tseng e Su, 2002). Quando bem configuradas, conseguem avaliar itens frequentes e regras de associação, contribuindo de forma efetiva para a análise e interpretação de resultados.

Seguindo essa abordagem, uma regra de associação é candidata a item frequente quando satisfaz a um suporte mínimo (sup_{min}). Relativo a isso, a escolha de um suporte é responsável por definir a porcentagem de itens que serão incluídos no conjunto de itens frequentes. Logo, a condição $A \Rightarrow B$ está relacionada à probabilidade de $P(A \cap B)$, que representa a frequência de ocorrência dos itens em um conjunto, a equação (4) expressa o cálculo de suporte:

$$suporte(A \Rightarrow B) = P(A \cap B) = \frac{|A \cap B|}{N} \quad (4)$$

O *suporte* é calculado com base na quantidade de transações que incluem A e B , sendo o módulo da interseção $|A \cap B|$ desses itens, dividido pelo total de transações N que formam o conjunto de dados.

Por outro lado, a métrica de confiança mede a precisão de uma regra de associação, considerando a probabilidade de B ocorrer em uma transação dado que A ocorreu. Matematicamente, isso é expresso pela probabilidade condicional $P(B|A)$ conforme ilustra a equação (5).

$$confiança(A \Rightarrow B) = P(B|A) = \frac{suporte(A \cup B)}{suporte(A)} \quad (5)$$

Dada a equação, a *confiança* é calculada por meio da frequência da união $A \cup B$, sendo exatamente o número de vezes que esses itens são encontrados juntos, dividido pelo número de vezes que A aparece nas transações. A maneira como essa métrica atua permite determinar a precisão de uma regra de associação, porém sua forma de abordagem é limitada, deixando uma lacuna quanto à real força de uma regra de associação.

Neste sentido, a métrica de *lift* surge como uma medida simples de correlação (Han et al., 2012), cujo foco é avaliar a força de uma regra de associação, considerando o quanto os eventos de uma regra são dependentes e correlacionados. A equação (6) ilustra a métrica de *lift*.

$$lift(A, B) = \frac{P(A \cup B)}{P(A) \times P(B)} \quad (6)$$

Logo, o valor de *lift* para uma associação é calculado como a razão entre a probabilidade do conjunto de itens A e B e o produto das probabilidades individuais desses itens. O resultado e o diagnóstico são determinados pela condição matemática expressa na equação (7).

$$lift(A, B) = \begin{cases} lift > 1 \\ lift = 1 \\ lift < 1 \end{cases} \quad (7)$$

Um *lift* maior que 1 indica uma associação positiva, ou seja, a presença de um item eleva a probabilidade de presença do outro. Se o *lift* for igual a 1, sugere que os itens são independentes e não afetam a ocorrência um do outro. Caso o *lift* seja menor que 1, indica uma associação negativa, em que a presença de um item reduz a probabilidade de ocorrência do outro.

3.5.2 Estrutura de Dados Frequent Pattern Tree (FP-Tree)

A FP-Tree desempenha um papel crucial na mineração de dados, especialmente no processamento de grandes volumes de informações. Essa estrutura de dados foi projetada para armazenar conjuntos esparsos de forma comprimida, otimizando a extração de itens frequentes e regras de associação (Grahne e Zhu, 2005). Ao eliminar a necessidade de gerar conjuntos candidatos, a abordagem da FP-Tree torna essas tarefas mais eficientes.

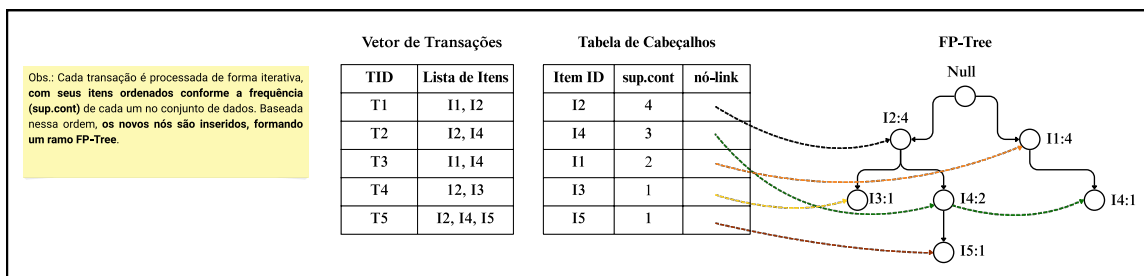
Nesse contexto, a aplicabilidade funcional dessa estrutura de dados está condicionada ao fato de os dados estarem no formato horizontal de um vetor de transações, onde cada transação T_n possui uma lista de itens, isto é, $D = \{T_1(i_1, i_2), T_2(i_1, i_3), \dots, T_n\}$.

O processo de construção da FP-Tree começa com a criação de um nó raiz vazio, que serve como ponto de partida para a inserção das transações na árvore. Cada transação é processada de forma iterativa, com seus itens ordenados conforme a frequência (sup_{cont}) de cada um no conjunto de dados. Baseada nessa ordem, os novos nós são inseridos, formando um ramo FP-Tree.

Quando uma nova transação é inserida e corresponde a um ramo existente na árvore, o sup_{cont} dos nós ao longo desse ramo é incrementado. Esse processo possibilita que a árvore mantenha um registro preciso da frequência de itens.

Após todas as transações estarem registradas, uma tabela de cabeçalho é criada para facilitar a navegação dos algoritmos de regras de associação. Seu diferencial está em fornecer um acesso direto aos nós de um item específico (Han et al., 2004). Os nós na FP-Tree são ligados por ponteiros, que formam uma lista conectada que pode ser facilmente percorrida. Figura 2 ilustra o registro de transações na FP-Tree.

Figura 2 – Adição do Registro de transações na FP-Tree



Fonte: Adaptado de Han et al. (2012).

3.5.3 Algoritmos Baseados em Vetores de Transações

A mineração de dados baseada em regras de associação apresenta modelos computacionais robustos, que se assemelham bastante na finalidade de análise, tanto no formato do conjunto de dados utilizado em procedimentos quanto na configuração inicial dos parâmetros mínimos de suporte (sup_{min}) e confiança ($conf_{min}$) (Zaki, 2000).

Sob essa perspectiva, os algoritmos Apriori e FP-Growth são amplamente utilizados para extrair itens frequentes e gerar regras de associação, especificamente em um vetor de transações, onde o processo de análise é orientado pela predefinição dos parâmetros de sup_{min} e $conf_{min}$ (Han et al., 2012). Por sua vez, o algoritmo FP-Max considera o mesmo formato de dados e um sup_{min} para extrair itens frequentes máximos (Hu et al., 2008).

Deste modo, a inicialização adequada dos dados transacionais e a definição das métricas de mineração são essenciais para a execução do processo.

3.5.4 Algoritmo Apriori

O algoritmo Apriori realiza um processo iterativo de junção e poda de candidatos, baseado no princípio de que, se um conjunto de itens é frequente, todos os seus subconjuntos

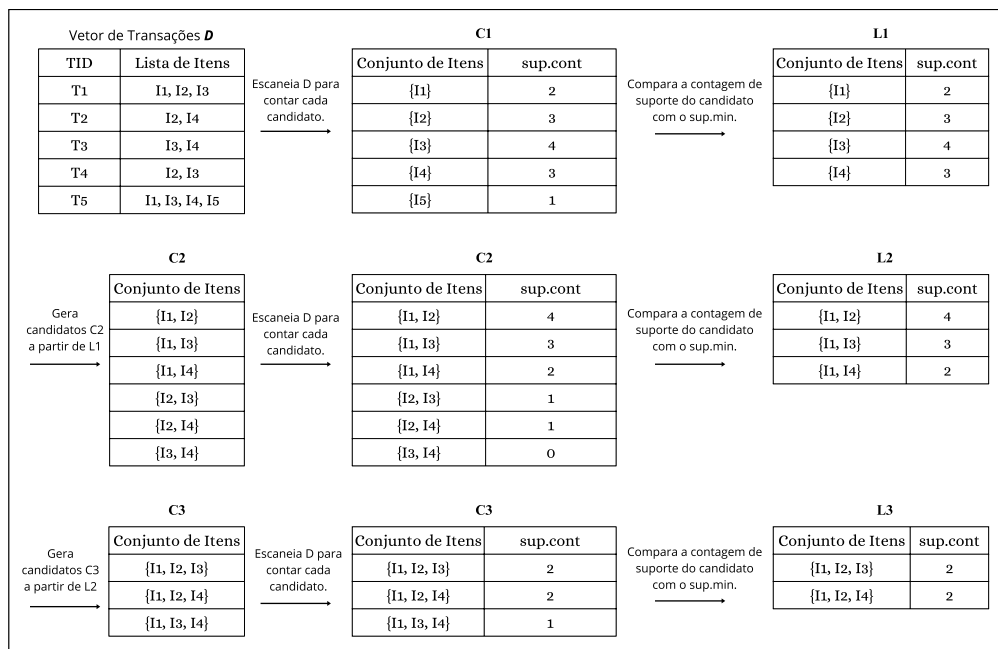
também devem ser frequentes. Assim, o conhecimento prévio do vetor de transações é essencial para determinar a relevância das associações (Agrawal e Srikant, 1994).

Inicialmente, o algoritmo identifica todos os itens únicos e forma o conjunto de candidato C_1 . Em seguida, determina o suporte de cada item, a partir da contagem de ocorrências sup_{cont} em todas as transações. Os itens que atendem ao sup_{min} formam o conjunto L_1 de itens frequentes, representado por $L_1 = \{i_1, i_2, i_3, \dots, i_n\}$.

Após a primeira poda, o algoritmo utiliza o conjunto L_1 para gerar os pares combinados de itens que irão compor o conjunto de candidatos C_2 . Mediante o procedimento de varredura no vetor D , são calculados os suportes desses pares para formar o conjunto de itens frequentes L_2 , no qual são incluídos apenas os pares que satisfazem ao sup_{min} , a representação visual dessas combinações seria, por exemplo, $L_2 = \{(i_1, i_2), (i_1, i_3), (i_2, i_3), \dots, (i_a, i_b)\}$.

O processo continua de forma iterativa, gerando combinações de três itens para formar o conjunto de candidatos C_3 a partir de L_2 , e assim por diante, até que não haja mais combinações possíveis para formar novos conjuntos candidatos. A Figura 3 ilustra o processo de geração de candidatos a itens frequentes, onde a contagem mínima de suporte é 2.

Figura 3 - Processo de Geração dos Conjuntos de Itens Candidatos



Fonte: Adaptado de Han et al. (2012).

Uma vez extraído o conjunto final de itens frequentes, o algoritmo gera regras candidatas utilizando uma busca combinatória que explora todas as formas possíveis de dividir

cada conjunto de itens em antecedente e conseqüente para gerar regras. Em seguida, o algoritmo calcula a confiança de cada regra candidata, e aquelas que atendem a $conf_{min}$ são consideradas fortes. Deste modo, representam as associações mais significativas de toda a análise.

3.5.5 Algoritmo FP-Growth

O algoritmo FP-Growth consiste em uma técnica direcionada à mineração de grandes conjuntos de dados, sua abordagem inicial inclui representar o conjunto de transações em uma estrutura FP-Tree compacta, que facilita os procedimentos de análise e elimina a necessidade de gerar conjuntos candidatos.

Mediante a construção da FP-Tree, o algoritmo começa a extração de itens frequentes, seguindo a ordem dos itens dispostos na tabela de cabeçalhos. Para cada item, é projetada uma subárvore composta por ele e seu ramo derivado, sob essa perspectiva ocorre uma mineração recursiva para encontrar padrões frequentes. Cada nó item tem seu sup_{cont} avaliado, e aqueles que não atendem ao sup_{min} são podados, até que restem apenas os conjuntos de itens que irão compor a lista de itens frequentes.

Após identificar a lista itens frequentes, o algoritmo inicia o processo de gerar regras de associação, dividindo cada conjunto de itens em antecedente e conseqüente para formar uma possível regra. Essas regras são avaliadas quanto à força, e apenas as que atendem ou excedem a $conf_{min}$ são mantidas, assegurando sua validade e relevância.

3.5.6 Algoritmo FP-Max

O algoritmo FP-Max consiste em uma abordagem voltada apenas para a extração de itens frequentes máximos e surge como uma variação do FP-Growth. Desse modo, utiliza a estrutura FP-Tree para identificar e armazenar as informações sobre os conjuntos de itens frequentes que atendem ao sup_{min} .

Apoiado nisso, o algoritmo inicia a varredura na FP-Tree em busca dos itens frequentes máximos. Cada conjunto de itens é considerado um máximo frequente a partir de duas condições. A primeira é não ser um subconjunto $\{i_1, i_2\}$, cujos elementos estejam inteiramente contidos em outro conjunto $\{i_1, i_2, i_3\}$. A segunda é não ter superconjuntos frequentes $\{i_1, i_2, i_3\}$ que contenham todos os seus elementos.

Uma vez que o conjunto de itens atenda às duas condições, nenhum outro conjunto que o contenha pode ser um máximo frequente, sendo aplicada a poda para eliminar todos os seus

subconjuntos e superconjuntos. O processo continua de forma iterativa até que todos os conjuntos de itens frequentes máximos sejam extraídos.

3.5.7 Algoritmo Eclat

O algoritmo Eclat consiste em uma técnica de extração de itens frequentes, que utiliza a interseção de listas de transações como base para a análise (Zaki et al., 1997). Nesse contexto, o formato vertical de dados permite uma busca em profundidade mais eficiente, especialmente quando o conjunto de dados apresenta baixa cardinalidade de itens por transação.

Em termos conceituais, o formato de dados vertical pode ser representado como um conjunto de pares (i, T_n) , sendo i um item e T_n o conjunto de transações que incluem i . Então, se houver dois itens A e B , por exemplo, suas representações seriam $A = \{T_1, T_2\}$ e $B = \{T_1, T_2, T_3\}$. A partir da definição do conjunto de dados no formato adequado e da escolha de um sup_{min} , o processo de extração de itens frequentes pode ser realizado.

Inicialmente, o algoritmo procede com a interseção das listas de transações (TIDset), isto significa contar as transações que contêm ambos os itens juntos, por exemplo os itens A e B aparecem em T_1 e T_2 , logo a frequência de ocorrência do sup_{cont} é igual a 2. Assim, os TIDset que atendem ou excedem o sup_{min} são tidos como frequentes.

Desta forma, o processo de interseção continua até que não seja mais possível encontrar conjuntos que satisfaçam o suporte definido, o que resulta na obtenção da lista final composta pelos itens frequentes.

3.6 Conclusão

Este capítulo apresentou os principais conceitos encontrados na literatura que embasam os métodos de mineração de dados utilizados no desenvolvimento prático desta dissertação, servindo para a compreensão das escolhas metodológicas adotadas pela autora.

Além disso, foram abordados métodos voltados à aplicação de testes, como o método de Monte Carlo para a realização de simulações, combinado com a técnica de *Bootstrap* para geração de dados sintéticos, e a técnica de *Benchmarking* para comparação de algoritmos e bases de dados, considerando critérios como escalabilidade, tempo de execução e uso de memória. Também foram explorados conceitos sobre KDD, mineração de dados, métricas de desempenho, estrutura de dados FP-Tree e algoritmos de regras de associação.

4 PLATAFORMA HAMBRE DELIVERY

Este Capítulo aborda o papel da plataforma Hambre Delivery, destacando sua relevância, funcionamento e impacto econômico e social no setor de *food service*. A Seção 4.1 faz uma introdução sobre este capítulo e contextualiza seus objetivos. A Seção 4.2 apresenta a plataforma Hambre Delivery no contexto da Região do Baixo Tocantins. A Seção 4.3 discute os desafios e oportunidades observadas na implantação da plataforma. Por fim, a Seção 4.4 encerra o capítulo com suas considerações e conclusões.

4.1 Introdução

No contexto de *delivery*, os aplicativos de *smartphones* auxiliam clientes e empreendedores nas vendas locais, proporcionando praticidade, autonomia e controle na realização das tarefas diárias. Em conformidade com Silchenko (2024), uma ferramenta para uso em *delivery* deve se atentar para três fatores essenciais: proporcionar usabilidade intuitiva (*user-on*), garantir confiabilidade e eficiência (*service-on*) e considerar a realidade da região onde está inserido (*reality-on*).

A partir disso, este Capítulo apresenta a proposta da plataforma Hambre Delivery, destacando sua importância para o desenvolvimento da região do Baixo Tocantins, assim como seu funcionamento, impacto e os desafios enfrentados para sua implementação.

4.2 A Plataforma Hambre Delivery no Contexto do Baixo Tocantins

O mercado de *delivery*, principalmente o segmento de *food service*, tem passado por mudanças significativas relacionadas às necessidades de consumo dos clientes. O motivo para esse fato está na compreensão de que a pandemia de COVID-19 impulsionou o crescimento do comércio *online* em *smartphones*, possibilitando vantagens práticas em relação aos métodos de compra convencionais, ou seja, além de ser eficiente, proporciona conforto, mobilidade e autonomia ao usuário no processo de compra (Rezaei et al., 2024).

Considerando o cenário de *lockdown* vivenciado no Brasil no período de pandemia, muitos empreendedores com estabelecimentos de micro e médio porte tiveram dificuldades de adesão, uso e adaptação a essas ferramentas. Além disso, a falta de conhecimento sobre a

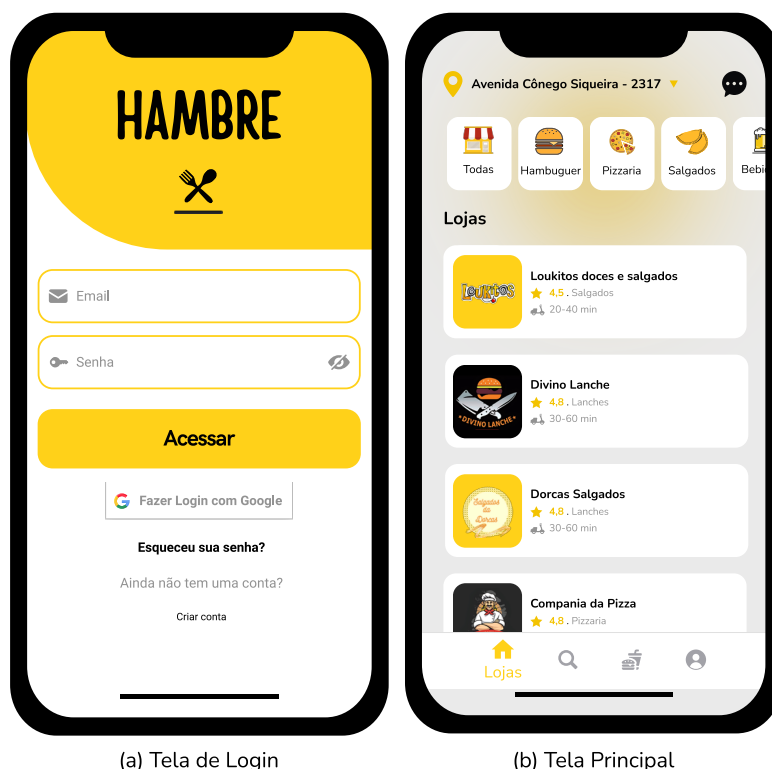
implantação do serviço de *delivery* contribuiu para o comprometimento da renda em negócios do setor alimentício. De acordo com Gonçalves (2024), há cidades brasileiras em que parte da renda local provém de negócios familiares, tais como na região do Baixo Tocantins, no Pará.

Neste sentido, a pesquisa do autor propôs o desenvolvimento dos aplicativos Hambre *Delivery* e Hambre Loja, iniciativas genuinamente paraenses voltadas ao crescimento econômico, à inclusão digital e ao fortalecimento de empreendedores do Baixo Tocantins. O projeto foi conduzido pela UFPA e custeado pela SECTET, oferecendo uma plataforma *food service* acessível para a realização de pedidos e entrega de produtos locais.

4.2.1 App Hambre Delivery

O *app* Hambre Delivery foi desenvolvido como uma solução exclusiva para o cliente final, oferecendo acesso direto às lojas da região do Baixo Tocantins, conforme a cidade informada pelo usuário. Por meio do aplicativo, é possível visualizar os produtos do cardápio, realizar pedidos, acompanhar em tempo real todas as etapas da entrega e avaliar a qualidade do serviço prestado. A Figura 4 ilustra duas telas de apresentação do *software*. A Figura 4 (a) representa a tela de login. A Figura 4 (b) ilustra a tela principal, onde estão disponíveis os filtros de categoria e a lista de lojas.

Figura 4 - Telas de Apresentação do App Hambre Delivery



(a) Tela de Login

(b) Tela Principal

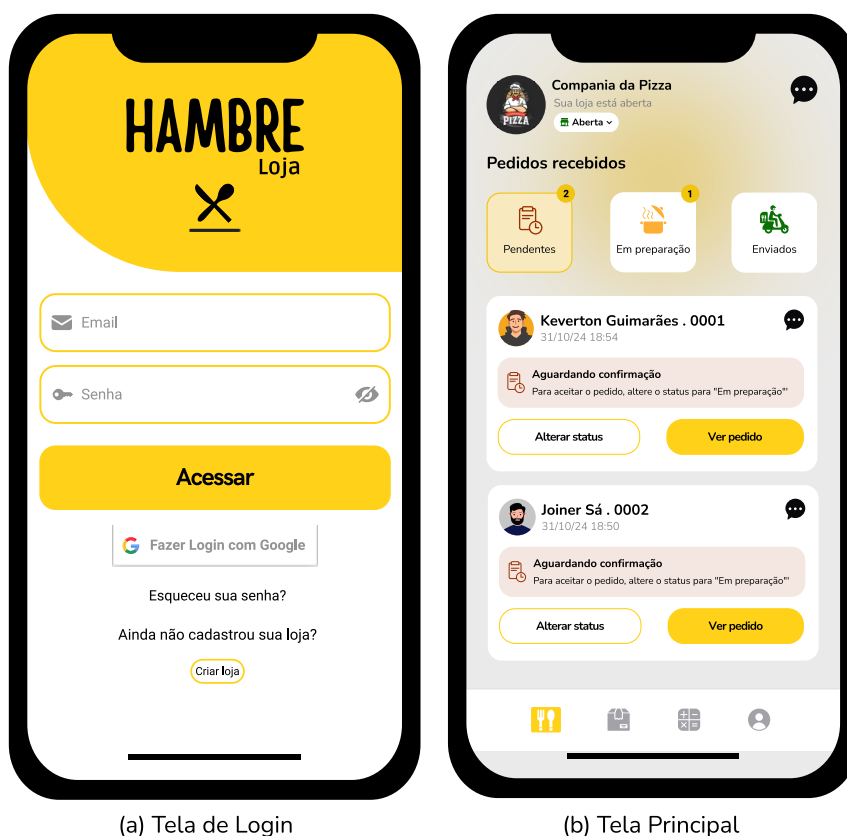
Fonte: Adaptado de Gonçalves (2024).

O Hambre Delivery para clientes está em operação desde 2022, possibilitando a digitalização e o acesso rápido a produtos e serviços conforme as demandas do público-alvo. O *software* apresenta oito categorias de venda, sendo elas: refeição, bebida regional, bebida alcoólica, hambúrguer, pizza, salgado, sobremesa e vitamina, que permitem filtrar as lojas correspondentes. Essa versatilidade permitiu a implantação de uma ferramenta especializada na cultura regional, representando um diferencial em relação aos demais softwares disponíveis no mercado *food service*.

4.2.2 App Hambre Loja

O *app* Hambre Loja foi desenvolvido para uso exclusivo dos comerciantes e aprimorado por meio de visitas *in loco*, nas quais foram oferecidas capacitações para facilitar a adaptação à ferramenta. Esse processo gerou *feedbacks* valiosos, que evidenciaram a necessidade de recursos mais robustos para o gerenciamento de pedidos, organização do cardápio e otimização das entregas, sendo esses aspectos essenciais para melhorar o desempenho comercial em uma plataforma *food service*. A Figura 5 ilustra as telas iniciais do *software*. A Figura 5 (a) representa a tela de login. A Figura 5 (b) ilustra a tela principal de acompanhamento de pedidos.

Figura 5 - Telas de Apresentação do App Hambre Loja



(a) Tela de Login

(b) Tela Principal

Fonte: Adaptado de Gonçalves (2024).

Dentre as funcionalidades disponíveis para as lojas, destacam-se a montagem de cardápio, acompanhamento de pedidos, controle de caixa, configuração de perfis de entrega, endereçamentos, horários comerciais e pagamentos. Com esse modelo de negócio, as lojas dispõem de uma ferramenta robusta para gerência de seus estabelecimentos.

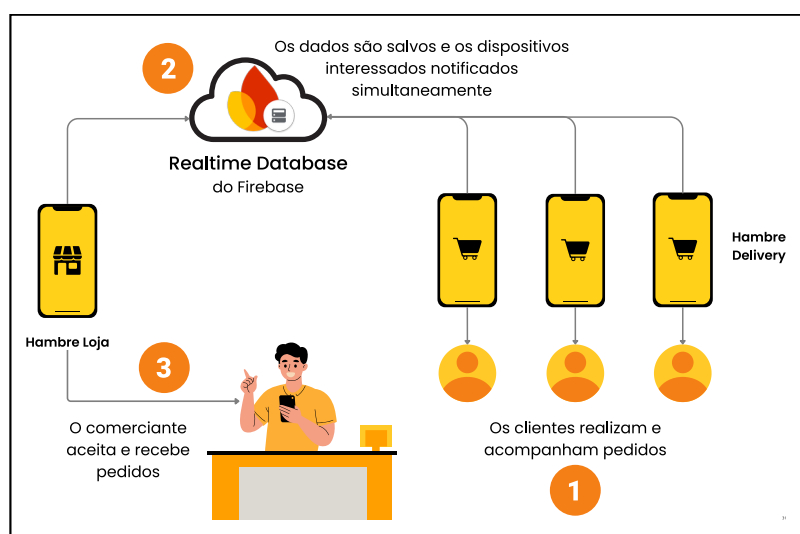
À medida que os comerciantes passaram a utilizar o *software*, surgiram novas demandas com base na experiência de uso do *app* e no comportamento dos clientes, entre as quais se destacou a falta de *feedback* sobre os produtos após a finalização dos pedidos. Essa perspectiva evidencia o principal objetivo desta dissertação de oferecer às lojas uma solução de suporte à decisão na gestão de vendas, baseada nas preferências de compra dos clientes.

4.2.3 Firebase

O projeto Hambre Delivery considerou que os softwares implementados deveriam ser sincronizados em tempo real, uma vez que os aplicativos Hambre Delivery e Hambre Loja apresentavam funcionalidades de comunicação específicas relacionadas ao acompanhamento de pedidos, que demandavam a troca simultânea de informações entre clientes e lojas.

Diante disso, foi adotado como *backend* o Firebase, uma plataforma do Google que oferece soluções em nuvem para o desenvolvimento de aplicativos móveis, incluindo banco de dados em tempo real, autenticação de usuários e hospedagem integrada. O diferencial desse recurso é o *Realtime Database*, que armazena dados na nuvem e notifica todos os dispositivos interessados simultaneamente. A Figura 6 detalha o funcionamento do Firebase *Realtime Database* para os aplicativos Hambre Delivery e Hambre Loja.

Figura 6 - Funcionamento do Firebase Realtime Database na Plataforma Hambre Delivery



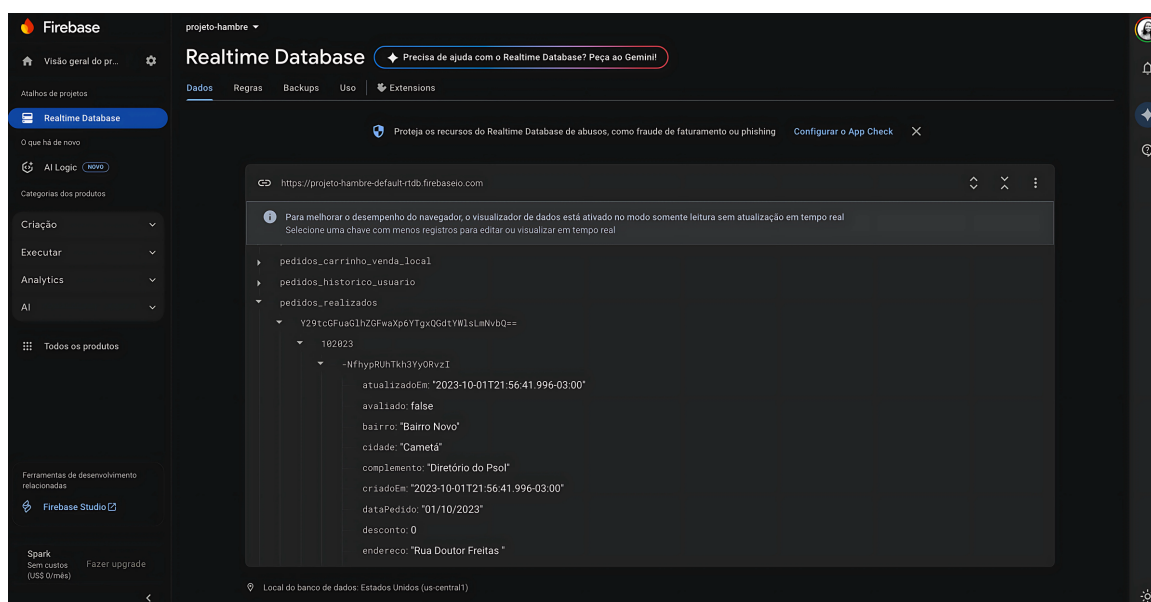
Fonte: Elaborado pela autora.

O processo inicia com a realização de pedidos pelos clientes, seguido pelo envio da solicitação à loja. Após o aceite, o Firebase *Realtime Database* atualiza o status em tempo real até a finalização da entrega. Mesmo que o usuário fique *offline*, o SDK utiliza um cache local que registra alterações e sincroniza os dados com a nuvem assim que o dispositivo fica *online*. A segurança dos dados é mantida por regras de acesso no banco de dados dos servidores do Firebase, garantindo que os usuários acessem apenas seus próprios dados ou informações às quais estejam relacionados, em qualquer dispositivo, seja *web* ou *mobile*.

Ambas as aplicações Hambre Delivery e Hambre Loja foram desenvolvidas na linguagem de programação *Java*, utilizando como ambiente de desenvolvimento nativo a plataforma *Android Studio*. Os recursos do *Realtime Database* seguem o formato de banco de dados não-relacional, por isso os dados são armazenados em uma estrutura hierárquica JSON (*JavaScript Object Notation*), onde cada nó pode conter objetos, arrays ou valores simples.

A troca de informações em tempo real ocorre por meio do protocolo *WebSocket*, que mantém uma conexão persistente entre o cliente e o servidor. Além disso, quando a comunicação *WebSocket* não está disponível, é utilizado de forma alternativa o protocolo HTTP (*Hypertext Transfer Protocol*) como *fallback*. A arquitetura do servidor é baseada em REST (*Representational State Transfer*), recebendo informações em formato JSON. A Figura 7 representa o formato de armazenamento de dados do projeto Hambre Delivery no *Realtime Database* do Firebase, enquanto a Figura 8 apresenta os dados brutos em formato JSON, extraídos diretamente do Firebase.

Figura 7 - Formato de Armazenamento dos Dados no Realtime Database



Fonte: Dados da pesquisa.

Figura 8 - Dados Brutos em Formato JSON Extraídos do Firebase

```
(1) ~NM9dDRB0d9dHbNvbA7": {
(2)   "idEmpresa": "Y2fyZG9zb2RvcmlsZW50TFAZ21haWwUy29%",
(3)   "idPedido": "~NM9dDRB0d9dHbNvbA7",
(4)   "idUsuario": "Y2FyYjG9zbWFlcmJlajW8wOEbnbWfPbC5jb20=",
(5)   "itens": [
(6)     {
(7)       "descricaoProduto": "",
(8)       "idProduto": "-NHK3BUHKIMrZlpx-TL-",
(9)       "nomeProduto": "Frango Assado na Brasa",
(10)      "preco": 15,
(11)      "quantidade": 1,
(12)      "urlImagem": "https://firebasestorage.googleapis.com/v0/b/projeto-hambre.appspot.com/o/Imagens%2FProdutos%2FY2fyZG9zb2RvcmlsZW50TFAZ21haWwUy29%2F1669402529031.jpeg?alt=media&token=51bbfcc6-e527-46a7-b106-fa9e300d5258"
(13)    },
(14)    ]
(15)   },
(16)   "nome": "Carlos Mauricio",
(17)   "nomeLoja": "Assados Bom Gosto",
(18)   "numEndereco": "0",
(19)   "status": "Entregue",
(20)   "statusVisualizacao": "N",
(21)   "taxaEmpresa": 0,
(22)   "telefone": "+55 (91) 98144-0543",
(23)   "tipoPagamento": "PIX",
(24)   "total": 15,
(25)   "totalParcial": 15,
(26)   "tokenUsuario": 0,
(27)   "urlLogoEmpresa": "https://firebasestorage.googleapis.com/v0/b/projeto-hambre.appspot.com/o/Imagens%2FPerfilLoja%2FY2fyZG9zb2RvcmlsZW50TFAZ21haWwUy29%2Fperfil.jpeg?alt=media&token=e2a41f77-3dff-46ac-8c1f5-9986e48c000e",
(28)   "visualizacao": "S",
(29)   "zeroTaxaEndereco": 0,
(30)   "zeroTaxaLocal": 0
(31) }
```

Obs.: Os dados mais relevantes para análise de vendas são **idPedido** e **nomeProduto**, este último pertencente ao array de **itens**.

Fonte: Dados da pesquisa.

O nó de pedidos realizados, ilustrado na Figura 7, segue uma estrutura hierárquica composta pelo ID da loja, número do pedido e uma chave de identificação aleatória, que permite ao *Realtime Database* saber qual estrutura deve ser atualizada simultaneamente. Dentro dessa chave aleatória, é possível observar algumas informações específicas sobre o pedido, como atualização, avaliação, bairro, cidade, complemento, criação, desconto e endereço.

A Figura 8 complementa, em detalhamento, os dados envolvidos na realização de um pedido, dentre os quais os mais relevantes para a composição de um conjunto de dados transacional para análise de tendências de venda são os parâmetros *idPedido* e *nomeProduto*, este último pertencente ao array de *itens*, uma vez que um único pedido pode conter diversos produtos.

4.3 Desafios e Oportunidades

Conforme exposto anteriormente, o projeto definiu como missão o atendimento da região do Baixo Tocantins, tendo como meta inicial a implantação da plataforma Hambre Delivery em cinco cidades, sendo elas Cametá, Baião, Mocajuba, Oeiras do Pará e Limoeiro do Ajuru.

O grande diferencial nesse processo foi a utilização de captadores em cada cidade, responsáveis por visitar os empreendimentos locais e apresentar os benefícios de aderir à plataforma, seu modo de uso e consultoria especializada, deixando claro a disponibilidade de um serviço completamente isento de cobrança aos participantes.

Uma observação relevante a ser levantada é o fato de predominar nessas localidades a prática de vendas convencional, na qual o cliente se desloca até a loja ou realiza seus pedidos através das redes sociais, por exemplo, via *WhatsApp*. Assim, o desafio enfrentado pelos captadores foi convencer os clientes e empreendedores da viabilidade do canal de vendas oferecido pelo aplicativo, demonstrando a praticidade dessa abordagem em relação ao método convencional de vendas. Deste modo, a plataforma buscou ir além, não apenas automatizar métodos manuais, mas oferecer uma solução eficiente para clientes e lojistas.

4.4 Conclusão

Este capítulo apresentou a proposta da plataforma *Hambre Delivery*, destacando sua importância para o setor de *food service*, suas funcionalidades, o fluxo de funcionamento do *Firestore Realtime Database*, além de contextualizar seu impacto na região do Baixo Tocantins.

Diante disso, é possível afirmar que a iniciativa contribuiu para o desenvolvimento econômico e a inclusão digital de empreendedores que, durante a pandemia de COVID-19, vivenciaram transformações significativas na forma de consumo de seus clientes. Além disso, foram discutidos os principais desafios enfrentados na implantação da plataforma, assim como as oportunidades geradas por sua adoção no comércio regional.

5 PROPOSTA DE SOFTWARE PARA DESCOBERTA DE TENDÊNCIAS NA PLATAFORMA HAMBRE DELIVERY

Este Capítulo apresenta a proposta de software voltada à descoberta de tendências na plataforma Hambre Delivery. A pesquisa caracteriza-se como aplicada, de natureza exploratória, com foco na mineração de dados para a extração de conhecimento. A proposta também contempla uma visão geral sobre algoritmos baseados em regras de associação, bem como técnicas empregadas na análise de desempenho computacional, como KDD, *Bootstrap* e Monte Carlo.

O restante deste Capítulo está organizado conforme descrito a seguir. A Seção 5.1 faz uma breve introdução e contextualiza os objetivos propostos. A Seção 5.2 descreve as fases de KDD aplicadas na proposta de software. A Seção 5.3 apresenta o método híbrido, que combina o método de Monte Carlo com as fases de Mineração de Dados e Interpretação/Avaliação. Por fim, na Seção 5.4, são apresentadas considerações e conclusões sobre este Capítulo.

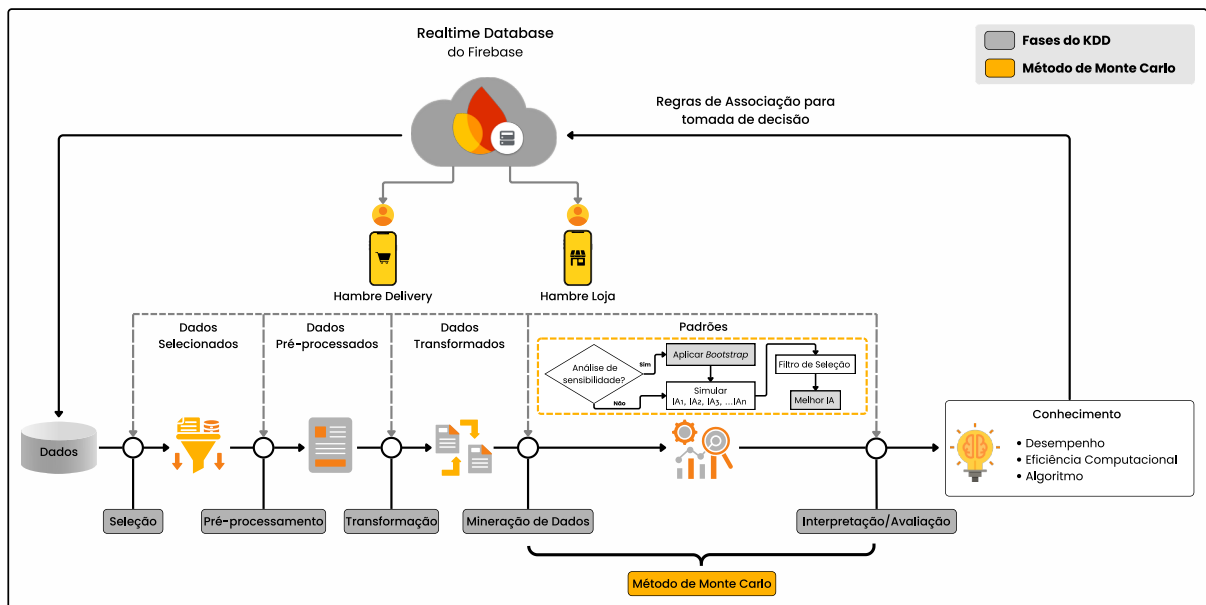
5.1 Introdução

O projeto de software para a descoberta de tendências de venda visa o desenvolvimento de um sistema inteligente de apoio à decisão voltado ao setor de *food service*. A solução propõe a integração à plataforma Hambre Delivery, composta por dois aplicativos, o Hambre Delivery voltado a clientes e o Hambre Loja direcionado a lojistas. Em termos práticos, o projeto incorpora um módulo analítico capaz de extrair conhecimento útil a partir dos dados operacionais de transações gerados na plataforma.

O fluxo funcional do módulo é composto por duas grandes fases de aplicação: KDD e Monte Carlo. A primeira contempla os processos estruturados de seleção, pré-processamento e transformação de dados, responsáveis por convertê-los em um formato adequado à análise. Na segunda fase, o Monte Carlo é aplicado nos processos de mineração de dados e interpretação/avaliação, a fim de analisar o desempenho computacional e a eficiência dos modelos de IA, considerando a obtenção do algoritmo mais adequado para extrair regras de associação conforme o contexto da plataforma.

Ao final do processo, o módulo analítico apresenta o conhecimento aplicável das regras de associação geradas, que servem de base para decisões estratégicas no aplicativo Hambre Loja, como ajustes de cardápios e campanhas promocionais. Para o contexto deste trabalho, o KDD proposto na literatura foi adaptado para representar a adoção híbrida do método de Monte Carlo e algoritmos de mineração de dados baseados em regras de associação. Diante do exposto, a Figura 9 ilustra a arquitetura geral da proposta de software para a plataforma Hambre Delivery.

Figura 9 - Arquitetura Geral da Proposta de Software para a Plataforma Hambre Delivery



Fonte: Adaptado de Fayyad, Piatetsky-Shapiro, Smyth (1996).

5.2 Fases do KDD

Esta seção apresenta a aplicação dos processos de seleção, pré-processamento e transformação, responsáveis pelo refinamento dos dados para o sistema inteligente de apoio à decisão na plataforma Hambre Delivery.

5.2.1 Coleta de Dados

A coleta de dados foi realizada em um período condizente com o lançamento da plataforma Hambre Delivery até se estabelecer como *marketplace de food service*. Por sua vez, esse período de atividade envolveu vendas em contextos variados como promoções, feriados, finais de semana, eventos especiais e dias úteis.

Este processo ocorreu entre 02/2022 e 03/2024, totalizando 24 meses de amostras transacionais que representam a relação entre pedidos, clientes e lojas, além de fornecer informações relevantes para a análise de vendas. Por meio das técnicas de seleção de amostragem e de atributos, foram coletados 2.191 registros de pedidos concluídos na plataforma, em formato não relacional JSON, conforme apresentado no Capítulo 3 desta dissertação.

Desses registros, foram extraídos dois atributos principais necessários para compor o conjunto de transações: o identificador único *idPedido* e o array de *itens*, sendo que a quantidade de itens adquiridos depende da vontade do usuário no momento da compra. Além disso, o array foi convertido em uma *string* contendo os nomes dos produtos, resultando em um formato adequado para análise de transações. A Figura 10 ilustra a base de dados transacional.

Figura 10 – Base de Dados em Formato Transacional

idPedido	Itens
-NHyc4uaD6nlvkGNCCa	Caldo carne com ovo, Caldo de frango, Cola cola Lata
-NI2qPIpEiPuNdjMQEQa	Batata frita com bacon e cheddar
-NI825MVLgyxvdhbpq3c	Pizza Media (M) Mussarela
-NIm9ho3zs49pBFICrH6	Marmitex frango empanado, Marmitex bife de carne
-NJpAWhuWD8yipzvIgt_	Açaí de 1 litro, Farinha de Mandioca
...	...
-NJpOI7Belp81UIAR8-H	Filé (F), Paraense (F), Nordestina (F), Refrigerante Coca-cola

Fonte: Elaborado pela autora.

5.2.2 Pré-processamento de Dados

Após a coleta de dados, notou-se inicialmente uma heterogeneidade na descrição textual de alguns registros. Entretanto, em uma análise exploratória, foi possível notar que a divergência de formatos textuais era mais ampla e que registros ruidosos poderiam direcionar a pesquisa para resultados incertos.

Para fins de esclarecimento, as divergências incluíam emojis, caracteres especiais, espaços extras, números, medidas volumétricas, itens duplicados no mesmo registro e diferentes formas de escrever o mesmo item, a exemplo do item "*Refrigerante Coca-Cola*", que foi escrito como "*Coca-Cola 2 litros*", "*Coca Original*", "*Coca cola 2L*", "*Coca cola Lata*",

"143 Coca de 1L", "144 Coca-Cola 2L", "Coca Cola", "Cola cola" ou "Refrigerante Coca-Cola 310 ml".

Após essa avaliação, foram levantados alguns argumentos para expressar o que levou a base de dados a conter tantos dados ruidosos. Em virtude da expansão do Hambre Delivery como *food service*, várias lojas foram captadas. Entretanto, a diferença entre estabelecimentos evidenciou a necessidade de criar cardápios com produtos e preços variados, sendo esse o principal motivo da falta de padronização textual, além da variedade de calendários sazonais.

Nesse contexto, foram aplicadas técnicas de identificação e correção de inconsistências, remoção de duplicatas e padronização de formatos. A Figura 11 demonstra o trecho de código em Python responsável pela padronização do produto "Refrigerante Coca-Cola".

Figura 11 - Trecho de Código Responsável pela Padronização da Escrita do Refrigerante Coca-Cola

```
(1) # Lista de variações identificadas para "Refrigerante Coca-Cola"
(2) coca_cola_variants = [
(3)     'Coca-Cola 2 litros', 'Coca Original', 'Coca cola 2L',
(4)     'Coca cola 600 mL', 'Coca cola Lata', '143 Coca de 1L',
(5)     '144 Coca-Cola 2L', 'Coca Cola', 'Cola cola', 'Refrigerante Coca-Cola 310 ml'
(6) ]
(7)
(8) # Função para substituir ocorrências de variações por um nome padronizado
(9) def standardize_name(text, variations, standard_name):
(10)     for variant in variations:
(11)         text = text.replace(variant, standard_name)
(12)     return text
(13)
(14) # Aplicação da padronização na coluna "products"
(15) dataset['products'] = dataset['products'].apply(standardize_name, args=(coca_cola_variants, 'Refrigerante Coca-Cola'))
```

Fonte: Elaborado pela autora.

5.2.3 Transformação de Dados

Nesta etapa, foi considerado o formato de dados mais adequado para aplicar os modelos de mineração. A análise exploratória inicial revelou desafios relacionados à dimensionalidade e à conversão de formato. Embora a quantidade de registros na base não seja extensa, observou-se muitos registros com apenas um item, o que pode reduzir a geração de itens frequentes e regras de associação. A identificação desse problema também foi abordada na pesquisa de Kamalov, Moussa e Reyes (2023), cuja solução foi a remoção de itens únicos.

Com base nisso, foi aplicada a técnica de redução de dimensionalidade para filtrar registros com mais de um item comprado. Esse procedimento resultou em 622 registros únicos e consistentes. Após a filtragem, os registros foram submetidos às técnicas de conversão de

tipos e transformação baseada em lógica, com o intuito de gerar duas estruturas de dados: horizontal e vertical.

Para obter a estrutura horizontal, os registros foram transformados em um vetor de transações, onde cada transação consiste em uma lista de itens. Isso se caracteriza como o cenário ideal para a análise de dados, utilizando os algoritmos Apriori, FP-Growth e FP-Max. A Figura 12 ilustra a saída da conversão para o formato de vetor de transações horizontal.

Figura 12 - Resultado da Conversão para o Formato Horizontal

```
(1) # formato horizontal de dados
(2) horizontal_format = [
(3)   {'Refrigerante Coca-Cola', 'Filé Parmegiana'},
(4)   {'Caldo de Frango', 'Caldo Carne com Ovo', 'Refrigerante Coca-Cola'},
(5)   {'Refrigerante Coca-Cola', 'Pizza Filé com Bacon'},
(6)   {'BIFE', 'Frango Empanado'},
(7)   {'Pizza Marguerita', 'Pizza Mista do Pará', 'Pizza Portuguesa'},
(8)   {'Pizza Metade Filé com bacon', 'Pizza Metade Calabresa'},
(9)   {'Pizza Metade Portuguesa', 'Refrigerante Guaraná', 'Pizza Metade Mussarela'},
(10)  {'Pizza Metade 4 queijos', 'Refrigerante Pepsi', 'Pizza Metade Calabresa'}
(11)  ...,
(12)  {'Hambúrguer X-Tudo', 'Batata Frita Simples', 'Refrigerante Guaraná'}
(13) ]
```

Fonte: Elaborado pela autora.

Por outro lado, no formato vertical, os dados foram organizados como um dicionário de itens ou *hash table*, onde cada item está associado a uma lista com todas as transações em que aparece na base de dados normalizada, sendo este o formato aceito para a mineração de dados utilizando o algoritmo Eclat. A Figura 13 ilustra a saída da conversão para o formato vertical.

Figura 13 - Resultado da Conversão para o Formato Vertical

```
(1) # formato vertical de dados
(2) vertical_format = {
(3) Pizza Filé com Bacon: [4, 77, 81]
(4) Pizza Metade Camarão com jambú: [5, 27, 84, 92]
(5) Pizza Metade Filé: [5, 6, 7, 20, 21, 23, 25, 30, 31, 56, 57, 58, 64, 75, 82]
(6) Pizza Metade Portuguesa: [5, 16, 19, 37, 54, 70, 134, 150, 152, 155, 156, 159, 165, 166, 167, 168, 187, 188, 189]
(7) Pizza Metade Nordestina: [6, 8, 18, 22, 31, 34, 35, 44, 61, 67, 68, 86]
(8) Pizza Metade Marguerita: [6, 28, 56, 57, 58, 92]
(9) Pizza Metade Filé com bacon: [6, 14, 15, 17, 22, 33]
(10) Pizza Metade Mista do Pará: [7]
(11) Pizza Metade Gaúcha: [7, 18]
(12) Pizza Metade Brasileira: [7]
(13) ...
(14) }
```

Fonte: Elaborado pela autora.

5.3 Método de Monte Carlo

Esta seção apresenta o método híbrido fracamente acoplado que combina o método de Monte Carlo e as fases de Mineração de Dados e Interpretação/Avaliação para avaliar o desempenho de algoritmos baseados em regras de associação, por meio de um conjunto de boas práticas que possibilitam a replicação em diferentes aplicações do setor de *food service*.

5.3.1 Mineração de Dados

No que se refere à etapa de mineração de dados, foi adotada uma análise mais direcionada à avaliação de desempenho dos algoritmos Apriori, FP-Growth, FP-Max e Eclat, apresentados no Capítulo 2, tendo como instrumento de observação as simulações de Monte Carlo. Essa abordagem consiste em um módulo analítico, cujo funcionamento é baseado na aplicação de simulações exploratórias e de sensibilidade. A primeira permite aliar o desempenho computacional a um baixo volume de dados, enquanto a segunda avalia a eficiência computacional a partir de dados sintéticos mais robustos, gerados com base em dados reais.

A Figura 9 detalha, na etapa de mineração, o fluxo de funcionamento do módulo analítico, que inicia com uma estrutura de decisão responsável por verificar o cenário de aplicação. Caso a condição seja satisfeita, aplica-se a técnica de *Bootstrap* para ampliar sinteticamente o volume de dados e contribuir para a compreensão de um problema mais amplo. Qualquer que seja o resultado da verificação, o fluxo prossegue para a simulação de n tipos de IA. Durante as simulações, os algoritmos são avaliados por filtros de desempenho relacionados ao tempo de execução, uso de memória e escalabilidade, permitindo a identificação do modelo mais eficiente em cada iteração de Monte Carlo.

5.3.2 Interpretação dos Dados

Com base nas simulações de Monte Carlo, os padrões obtidos na aplicação de cada algoritmo de IA são armazenados no decorrer de cada iteração, o que permite ao módulo analítico avaliar o desempenho geral conforme o comportamento apresentado diante da variação de parâmetros e a coerência dos padrões encontrados, considerando a efetividade dos modelos avaliados.

Ao final das simulações, o módulo analítico indica o melhor resultado por meio de gráficos comparativos e tabelas compostas pelos padrões identificados, que informam o

desempenho e a eficiência computacional de todos os algoritmos testados. Esse conhecimento colabora com a escolha do algoritmo de IA mais eficaz para a obtenção de regras de associação para suporte à decisão no *app* Hambre Loja.

5.4 Conclusão

Este capítulo apresentou a proposta de um software para o desenvolvimento de um sistema inteligente, que visa identificar tendências de vendas e oferecer suporte à tomada de decisão estratégica no setor de *food service*. A solução propõe técnicas para análise do desempenho computacional, adaptando os processos metodológicos de KDD para representar a adoção híbrida, fracamente acoplada, do método de Monte Carlo e de algoritmos de mineração de dados baseados em regras de associação, possibilitando um módulo analítico capaz de extrair conhecimento útil a partir das transações de vendas oriundas da plataforma Hambre Delivery.

6 CASO DE ESTUDO

Este Capítulo descreve os parâmetros para a aplicação do método de Monte Carlo e dos algoritmos propostos no Capítulo 3, bem como os experimentos comparativos de desempenho adotados nas simulações exploratórias e de sensibilidade.

O restante deste Capítulo está organizado conforme descrito a seguir. Na Seção 6.1, é apresentada uma introdução contextualizando os objetivos do Caso de Estudo. A Seção 6.2 apresenta as configurações e a aplicação do método de Monte Carlo para desempenho computacional nas fases das simulações exploratórias e de sensibilidade. A Seção 6.3 descreve os parâmetros dos algoritmos de mineração de dados baseados em regras de associação. A Seção 6.4 detalha os experimentos de desempenho computacional. A Seção 6.5 apresenta a aplicação da análise das tendências de venda. Por fim, na Seção 6.6, são apresentadas considerações e conclusões sobre este caso de estudo.

6.1 Introdução

Analisar e mensurar a qualidade e eficácia de algoritmos de mineração é uma tarefa sensível, que exige conhecimento adequado, atento e imparcial sobre os aspectos a serem avaliados, como cenários de simulação e configurações escolhidas. Nesse contexto, cada parâmetro foi definido com foco na obtenção de resultados mais próximos do ideal.

Tais experimentos foram realizados em um MacBook Air operando com o sistema macOS Sequoia e integrado a um processador M1. Quanto à memória, são 16 GB unificados entre uma CPU de 8 núcleos e uma GPU de 7 núcleos. Além disso, as análises foram desenvolvidas na linguagem de programação Python.

Motivado por isso, este Capítulo tem como objetivo apresentar e justificar a escolha de parâmetros de configuração, bem como fornecer detalhes técnicos sobre os cenários de simulação e realização dos experimentos de desempenho computacional.

6.2 Método de Monte Carlo para Análise de Desempenho Computacional

A análise de desempenho foi realizada por meio de três cenários de simulação de Monte Carlo, que permitem verificar a confiabilidade e a precisão de critérios como escalabilidade,

memória e tempo de execução. A Tabela 2 apresenta os parâmetros técnicos adotados para a aplicação do método de Monte Carlo.

Tabela 2 – Parâmetros de Configuração das Simulações de Monte Carlo

Parâmetro	Descrição	Valor
N_ITERATIONS	Número de interações por cenário	100, 1.000, 10.000
BASE_ORIGINAL	Tamanho de base de dados normalizada	622
N_SAMPLE	Amostragem gerada por cenário	73.000
RANDOM_STATE	Controle da aleatoriedade (<i>Seed</i>)	42
CARDINALITY	Geração mínima e máxima de itens por transação	3 a 7

Fonte: Elaborado pela autora.

Poucas diretrizes foram encontradas na literatura para embasar a escolha do número de iterações por cenário de Monte Carlo, pois depende de alguns fatores, como tipo de abordagem de mineração e modelos computacionais, além de que, por serem mais específicas, podem se direcionar a finalidades diferentes, como por exemplo, análise de vendas. Dessa forma, os cenários de simulação de Monte Carlo tomaram por base critérios adaptados de Heijungs (2020) e Tanaka et al. (2021), que sugerem a definição do parâmetro *N_ITERATIONS* para a realização de simulações entre 100 e 10.000 iterações, mas essa escolha depende do tamanho do conjunto de dados e o tipo de análise.

Considerando o foco desta investigação, o cenário de 100 iterações foi definido para confirmar o comportamento correto dos modelos e possíveis erros, sem gastar muitos recursos. Enquanto que os cenários com 1.000 e 10.000 iterações foram definidos a fim de obter mais consistência nas análises de desempenho computacional.

Conforme a proposta de software apresentada no Capítulo 4, o processo de mineração de dados foi realizado em duas fases: na primeira, foram definidas simulações exploratórias, que permitissem analisar os parâmetros de configuração e o comportamento dos algoritmos, visando eventuais ajustes. Na segunda, foram definidas simulações de sensibilidade, utilizando mais critérios de comparação para sustentar a investigação.

6.2.1 Simulações Exploratórias

Com base nos princípios de *Benchmarking* para avaliação de desempenho computacional, as simulações foram definidas considerando os critérios tempo de execução e escalabilidade. A expectativa é que a investigação forneça um diagnóstico inicial que permita

revisitar as etapas de KDD e as configurações de Monte Carlo, especialmente em casos de erro que requerem ajustes essenciais ao longo da pesquisa.

Em paralelo aos cenários de Monte Carlo estabelecidos, foram utilizadas as 622 amostras de transações normalizadas da plataforma Hambre Delivery, conforme o parâmetro *BASE_ORIGINAL*. Para manter a consistência entre as simulações, o parâmetro *RANDOM_STATE* foi fixado para controlar a aleatoriedade na seleção das amostras, garantindo a confiabilidade das medidas de tempo médio absoluto, desvio padrão e margem de erro, permitindo mensurar a escalabilidade e o tempo de execução dos algoritmos em cada cenário.

6.2.2 Simulações de Sensibilidade

Quanto às simulações de sensibilidade, as análises seguiram um escopo mais amplo, a fim de garantir que a proposta de software seja robusta, confiável e capaz de lidar com o crescimento de dados a longo prazo, conforme apontado no estudo de Farias et al. (2016). Além dos critérios de escalabilidade e tempo de execução, a avaliação considerou o consumo de memória e o uso de um conjunto de dados mais robusto, alinhado aos princípios do *Benchmarking*.

A obtenção de um conjunto de dados robusto e gerado sinteticamente seguiu um processo mais específico e lógico, iniciado pela avaliação dos principais aplicativos de *delivery* de alimentos atuando no Brasil, tendo como base o número de *downloads* e avaliações positivas (entre quatro e cinco estrelas), pois esses aspectos representam o que o público realmente utiliza. Assim, foram selecionados três aplicativos: o *iFood*, seguido do *Zé Delivery de Bebidas* e do *Mais Delivery*, dentre os quais se destacou o *iFood* com os melhores índices.

Em seguida, para gerar os registros sintéticos necessários para a análise de sensibilidade, levou-se em consideração o número de pedidos/mês registrados no *iFood* durante seu quarto ano de atuação no mercado de *delivery*. De acordo com Ozemela (2024), o *iFood* iniciou suas atividades em 2011, utilizando como suporte para a realização de pedidos um guia de cardápio impresso, e nesse período a média de vendas registradas era de 12 mil pedidos mensais, mas em 2012 o aplicativo alcançou o marco de 73 mil pedidos mensais, evidenciando um crescimento 6 vezes maior do que em seu primeiro ano.

Esse volume de 73 mil pedidos serviu de referência para a definição do parâmetro *N_SAMPLE*, utilizado na técnica de *Bootstrap* para a geração de dados sintéticos. O diferencial desse conjunto sintético foi preservar as características da base original da plataforma Hambre

Delivery, conforme as categorias de venda: refeição, bebida regional, bebida alcoólica, hambúrguer, pizza, salgado, sobremesa e vitamina. Assim, o parâmetro *CARDINALITY* foi estabelecido para ampliar a quantidade de itens por transação, adicionando novos itens de forma aleatória, conforme as características de cada transação.

A partir dessas configurações, as técnicas de Monte Carlo e *Bootstrap* foram aplicadas em paralelo para mensurar o tempo de execução, uso de memória e escalabilidade dos algoritmos de mineração de dados, obtendo variações estatísticas de tempo médio absoluto, desvio padrão e margem de erro em cada cenário de Monte Carlo.

6.3 Parâmetros de Configuração dos Algoritmos de Regras de Associação

Destacando a carência de um padrão único para a execução dos algoritmos, foram utilizadas neste trabalho as configurações de suporte e confiança mais recorrentes em pesquisas da área. Esses parâmetros tomaram por base a extração de padrões de alta utilidade, possibilitando controle e confiabilidade para mensurar o desempenho de cada algoritmo. A Tabela 3 ilustra os parâmetros adotados nas simulações.

Tabela 3 – Parâmetros Adotados pelos Algoritmos nas Simulações de Monte Carlo

Parâmetro	Valor			Referência
Suporte (%)	0.6	0.8	1	Agrawal, Imieliński e Swami (1993)
Confiança (%)	50	60	-	Suma e Sagar (2006) e Siswanto et al. (2024)

Fonte: Elaborado pela autora.

O estudo de Agrawal, Imieliński e Swami (1993) estabelece o uso do parâmetro de suporte entre 0.1% e 5% como critério para identificar itens frequentes e regras de associação, a fim de extrair uma maior quantidade de padrões. Esse intervalo permite adaptações ao tipo de abordagem avaliada. Por isso, os parâmetros de suporte foram definidos em 0.6%, 0.8% e 1% para manter uma sequência nas simulações.

Na literatura, também foram encontradas abordagens relacionadas ao parâmetro de confiança visando controle e consistência. Os autores Suma e Sagar (2006) ressaltam que deve haver equilíbrio na geração de regras de associação fortes e fracas, pois reunir o conhecimento necessário sobre um conjunto de transações requer uma análise apropriada e imparcial dos diversos pontos das regras geradas. À vista disso, os autores recomendam a definição do parâmetro de confiança em 50%.

Por outro lado, Siswanto et al. (2024) defendem o uso de uma abordagem centrada na obtenção de regras de associação mais fortes, ou seja, todo o processo de mineração é focado na extração de padrões que ocorrem com mais frequência. Por isso, os autores definiram a configuração do parâmetro de confiança em 60%.

Após este levantamento, os parâmetros observados na literatura foram elencados para compor as configurações necessárias à implementação dos algoritmos nas simulações exploratórias e de sensibilidade.

6.4 Experimentos de Desempenho Computacional

Esta Seção apresenta os experimentos aplicados em cada cenário de Monte Carlo para as simulações exploratória e de sensibilidade. Mesmo com ajustes específicos de cada fase, foram mantidos os mesmos experimentos de extração de itens frequentes e geração de regras de associação, a fim de garantir a comparabilidade de resultados.

6.4.1 Experimento de Extração de Itens Frequentes

Este experimento propõe avaliar o desempenho na extração de itens frequentes, considerando as variações de suporte 0.6%, 0.8% e 1%. Como instrumentos essenciais desse processo, foram selecionados os algoritmos Apriori, FP-Growth, FP-Max e Eclat, que desempenham o papel de guiar, investigar e promover a identificação de padrões relevantes.

O procedimento de comparação se deu mediante a aplicação de três cenários de Monte Carlo, definidos para executar os algoritmos por 100, 1.000 e 10.000 iterações. Vale ressaltar a realização do monitoramento dos recursos computacionais utilizados por cada modelo, o que possibilitou mensurar a média absoluta e o desvio padrão durante as variações de suporte.

6.4.2 Experimento de Geração de Regras de Associação

Este experimento consiste em uma abordagem comparativa com foco na geração de regras de associação. A escolha dos instrumentos de investigação levou em consideração principalmente a capacidade de gerar essas regras. Dentro desse contexto, vale destacar a impossibilidade de certos algoritmos de cumprir essa tarefa. Assim, foram selecionados para análise os algoritmos Apriori e FP-Growth.

Quanto às simulações de Monte Carlo, optou-se por manter as mesmas condições de número de iterações em cada cenário por algoritmo, garantindo continuidade e coerência em relação ao experimento de extração de itens frequentes. A análise foi aplicada utilizando os parâmetros de suporte em 0.6% e as variações de confiança em 50% e 60%.

Com base no monitoramento realizado em cada cenário de simulação, foi possível mensurar a média absoluta e o desvio padrão, proporcionando uma avaliação ampla do desempenho dos modelos analisados.

6.5 Análise das Tendências de Venda da Plataforma Hambre Delivery

Este experimento propõe analisar as tendências de venda presentes no conjunto de dados com 622 registros de transações da plataforma Hambre Delivery. Além disso, como instrumento de aplicação, será adotado o algoritmo de melhor desempenho no experimento de regras de associação.

Conforme isso, foram adotados os parâmetros de suporte em 1% e confiança em 50%, seguindo algumas especificações utilizadas nos experimentos anteriores, uma vez que o algoritmo deverá utilizar ambos para obter as regras de associação. A escolha desses parâmetros busca garantir maior confiabilidade, a fim de que as tendências encontradas reflitam diretamente as características desse conjunto.

6.6 Conclusão

Este Capítulo apresentou as configurações adotadas para a aplicação do Monte Carlo e dos algoritmos de mineração de dados baseados em regras de associação, além das simulações exploratórias e de sensibilidade, cada uma composta por experimentos com a finalidade de comparar e apoiar a escolha do algoritmo mais eficiente conforme o contexto de análise. Assim, os experimentos foram aplicados e os resultados coletados.

7 RESULTADOS E DISCUSSÃO

Este Capítulo apresenta os resultados obtidos com base na Proposta de Software e no Caso de Estudo apresentados nos Capítulos 5 e 6, que possibilitaram a descoberta de conhecimento a partir dos experimentos realizados nas simulações exploratórias e de sensibilidade.

O restante deste Capítulo está organizado da seguinte forma, na Seção 7.1 faz a introdução deste Capítulo e contextualiza seus objetivos. A Seção 7.2 apresenta os resultados das simulações exploratórias. A Seção 7.3 descreve os resultados das simulações de sensibilidade. Na Seção 7.4 são apresentadas as tendências de venda da plataforma Hambre Delivery. E por fim, a Seção 7.5 apresenta as considerações e conclusões deste Capítulo.

7.1 Introdução

Avaliar e compreender o comportamento, a qualidade e a eficiência de algoritmos em uma análise comparativa permite explorar amplamente as possibilidades de implementação, considerando tanto as vantagens quanto as desvantagens.

A partir disso, este Capítulo visa apresentar, dentro das simulações propostas, os resultados dos experimentos de extração de itens frequentes e geração de regras de associação, considerando as métricas de mineração (suporte e confiança), o tempo de execução, o uso de memória e a escalabilidade (cenários de Monte Carlo). Além disso, serão discutidas as tendências de vendas identificadas pelo algoritmo de melhor desempenho na geração de regras de associação.

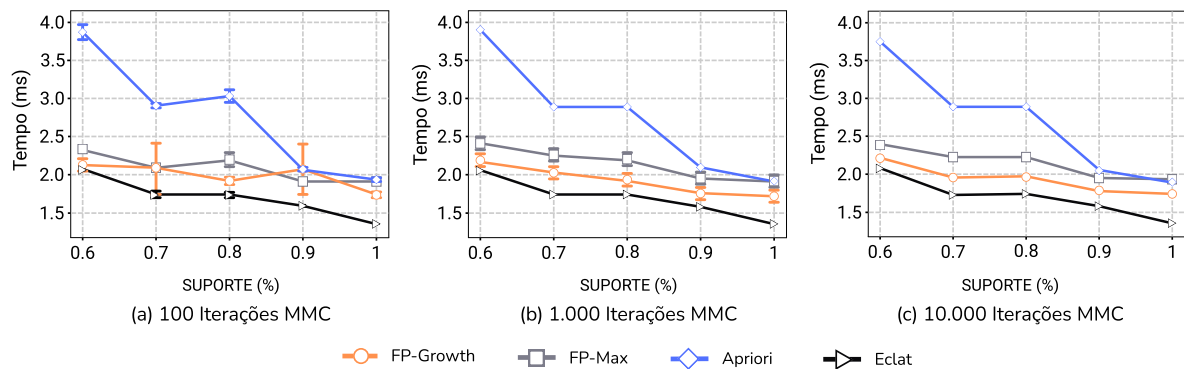
7.2 Resultados das Simulações Exploratórias

Esta seção detalha os resultados das simulações exploratórias referentes ao desempenho dos algoritmos de regras de associação, sob critérios de escalabilidade e tempo de execução. Outro aspecto relevante foi a aplicação dessas simulações no conjunto de dados com 622 registros de transações, visando à escolha da ferramenta mais adequada para identificar os padrões de venda do Hambre Delivery.

7.2.1 Experimento de Extração de Itens Frequentes

Com base nos requisitos estabelecidos, foram obtidos os resultados de desempenho dos algoritmos FP-Growth, FP-Max, Apriori e Eclat. A Figura 14 apresenta o experimento de extração de itens frequentes, englobando os três cenários de Monte Carlo. A Figura 14 (a) representa o cenário inicial, com 100 iterações. A Figura 14 (b) ilustra um cenário mais amplo, definido com 1.000 iterações. Por fim, a Figura 14 (c) detalha o cenário com 10.000 iterações.

Figura 14 - Resultados das Simulações Exploratórias para a Extração de Itens Frequentes



Fonte: Resultados da pesquisa.

Os resultados da extração de itens frequentes nos cenários de 100 e 1.000 iterações revelaram um desvio padrão significativo no tempo de execução dos algoritmos FP-Growth, FP-Max, Apriori e Eclat, que reduzem com o aumento das iterações. Todavia, o cenário de 10.000 iterações evidencia o comportamento mais estável dos modelos.

A Tabela 4 sintetiza o tempo acumulado em milissegundos pelos algoritmos em cada cenário de Monte Carlo, enquanto a Tabela 5 informa a quantidade de itens frequentes extraídos por eles nas variações de suporte 0.6%, 0.8% e 1%.

Tabela 4 - Tempo Acumulado pelos Algoritmos nas Simulações Exploratórias de Extração de Itens Frequentes

Monte Carlo	Suportes (%)	Tempo (ms)			
		FP-Growth	FP-Max	Apriori	Eclat
100	0.6 - 1	987	1.028	1.374	846
1.000	0.6 - 1	9.745	10.544	13.560	8.469
10.000	0.6 - 1	97.350	105.750	133.620	85.210
Total (ms)		108.082	117.322	148.554	94.525
Total (min)		1.80	1.95	2.47	1.57

Fonte: Resultados da pesquisa.

Tabela 5 - Quantidade de Itens Frequentes Obtidos pelos Algoritmos FP-Growth, FP-Max, Apriori e Eclat nas Simulações Exploratórias

Suporte (%)	Algoritmos			
	FP-Growth	FP-Max	Apriori	Eclat
0.6	168	72	168	238
0.8	126	63	126	168
1	83	49	83	91

Fonte: Resultados da pesquisa.

Quanto ao desempenho dos modelos, pode-se perceber que os algoritmos FP-Growth e FP-Max registraram tempos de execução satisfatórios, que permitiram concluir as simulações em 1.80 e 1.95 (min) nos três cenários de Monte Carlo, respectivamente, com uma diferença sutil entre eles. Isso porque ambos utilizam a estrutura FP-Tree, que otimiza a extração de itens frequentes. Nessa perspectiva, a análise comparativa apresentada na pesquisa de Nande, Patne e Kelkar (2025) alcançou resultados semelhantes.

Por outro lado, o algoritmo Apriori precisou de mais tempo para realizar as simulações, levando cerca de 2.47 (min) para concluir os três cenários de Monte Carlo, mesmo em conjunto de dados de baixa complexidade. Os mesmos autores destacam que esse processamento lento se deve às múltiplas varreduras necessárias para gerar conjuntos candidatos a itens frequentes.

À frente dos outros modelos analisados, o algoritmo Eclat apresentou o melhor desempenho, concluindo as simulações em 1.57 (min). Essa capacidade de processamento está relacionada a dois fatores: à mineração vertical e ao pequeno volume de transações com baixa cardinalidade de itens por transação. Além disso, o algoritmo Eclat extraiu a maior quantidade de itens frequentes, obtendo cerca de 238, 168 e 91 itens frequentes nas variações de suporte 0.6%, 0.8% e 1%, conforme ilustra a Tabela 5.

No contexto da escalabilidade, os cenários de Monte Carlo foram eficazes para cumprir sua finalidade. Em termos práticos, a realização das simulações com um número elevado de iterações registrou tempos reduzidos justamente por envolver uma quantidade pequena de dados, caracterizando uma abordagem viável nessas condições.

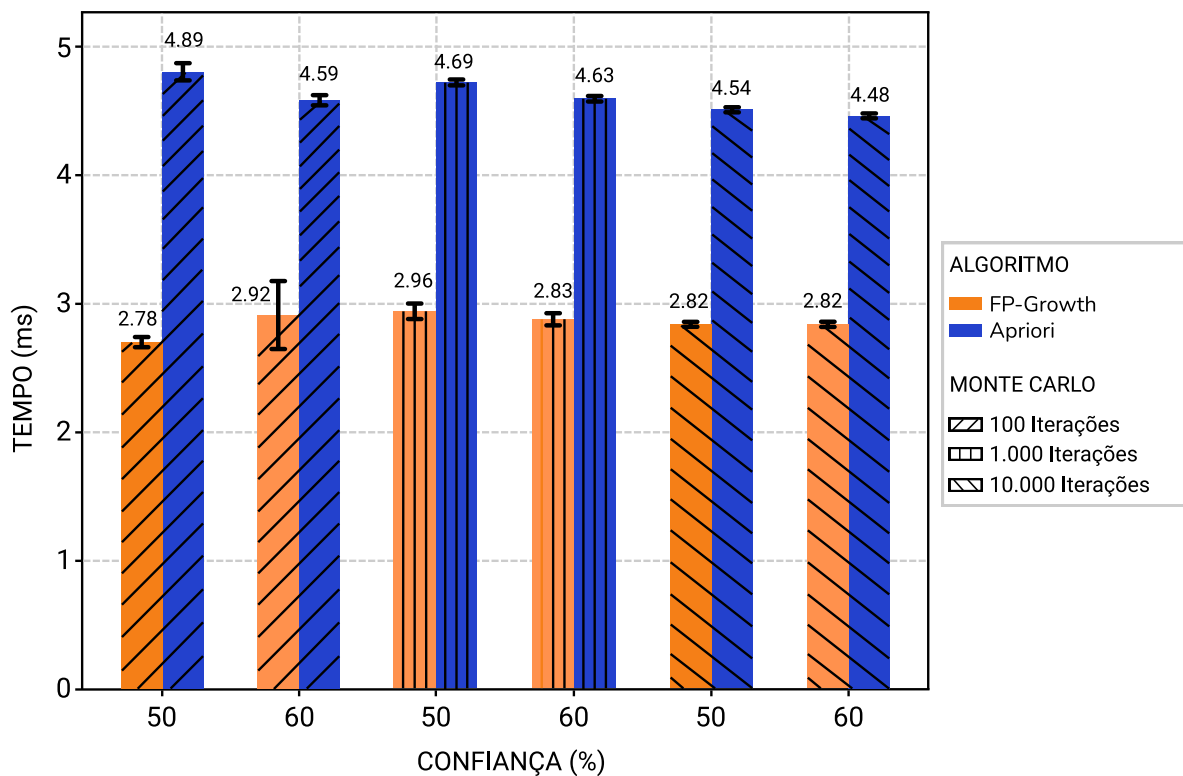
7.2.2 Experimento de Geração de Regras de Associação

Com base nos resultados obtidos anteriormente, foi adotado neste experimento o parâmetro de suporte em 0.6%, visando geração máxima de regras de associação. Os aspectos

observados para análise e discussão foram determinados sob a variação de confiança em 50% e 60%.

A Figura 15 evidencia o desempenho dos algoritmos FP-Growth e Apriori quanto ao tempo de execução nas simulações de regras de associação. Cada simulação de Monte Carlo é representada por linhas dispostas sobre as barras dos algoritmos. As linhas inclinadas à direita ilustram o cenário de 100 iterações. As linhas verticais representam o cenário de 1.000 iterações. Por fim, as linhas inclinadas à esquerda representam o cenário de 10.000 iterações.

Figura 15 - Resultados das Simulações Exploratórias para a Geração de Regras de Associação



Fonte: Resultados da pesquisa.

Os resultados revelam redução no tempo de execução e no desvio padrão conforme o aumento do número de iterações, indicando consistência entre os padrões alcançados. Essa confiabilidade estatística é mais evidente no cenário de 10.000 iterações, possibilitando considerações mais precisas sobre o desempenho dos algoritmos.

Ainda neste cenário, o algoritmo FP-Growth alcançou tempos muito baixos, levando em média cerca de 2.82 (ms) para gerar regras de associação nas variações de confiança de 50% e 60%. Por outro lado, o algoritmo Apriori precisou de mais tempo para concluir a tarefa, registrando 4.54 e 4.48 (ms) nas mesmas variações.

Complementando a análise, a Tabela 6 apresenta a quantidade de regras de associação geradas pelos algoritmos FP-Growth e Apriori nas variações de confiança de 50% e 60%. Ambos atuam sob as mesmas condições e, dependendo do conjunto de dados, as regras obtidas são geralmente equivalentes.

Tabela 6 - Quantidade de Regras de Associação Obtidas pelos Algoritmos FP-Growth e Apriori nas Simulações Exploratórias

Confiança (%)	Algoritmos	
	FP-Growth	Apriori
50	119	119
60	101	101

Fonte: Resultados da pesquisa.

O desempenho do algoritmo FP-Growth nas simulações é uma vantagem decorrente da FP-Tree. Nas mesmas condições, o algoritmo Apriori precisou de mais tempo para extrair a mesma quantidade de regras de associação, um fator recorrente em todos os cenários de Monte Carlo. Esse processamento mais lento exige o uso de mais recursos computacionais, mesmo em um conjunto de dados pequeno.

Vale ressaltar que o procedimento prático dos três cenários de Monte Carlo foi rápido, possibilitando testar a escalabilidade dos modelos em iterações maiores. Em termos de versatilidade, o algoritmo FP-Growth representa a escolha mais adequada para gerar regras de associação, por sua consistência na mineração de dados.

7.3 Resultados das Simulações de Sensibilidade

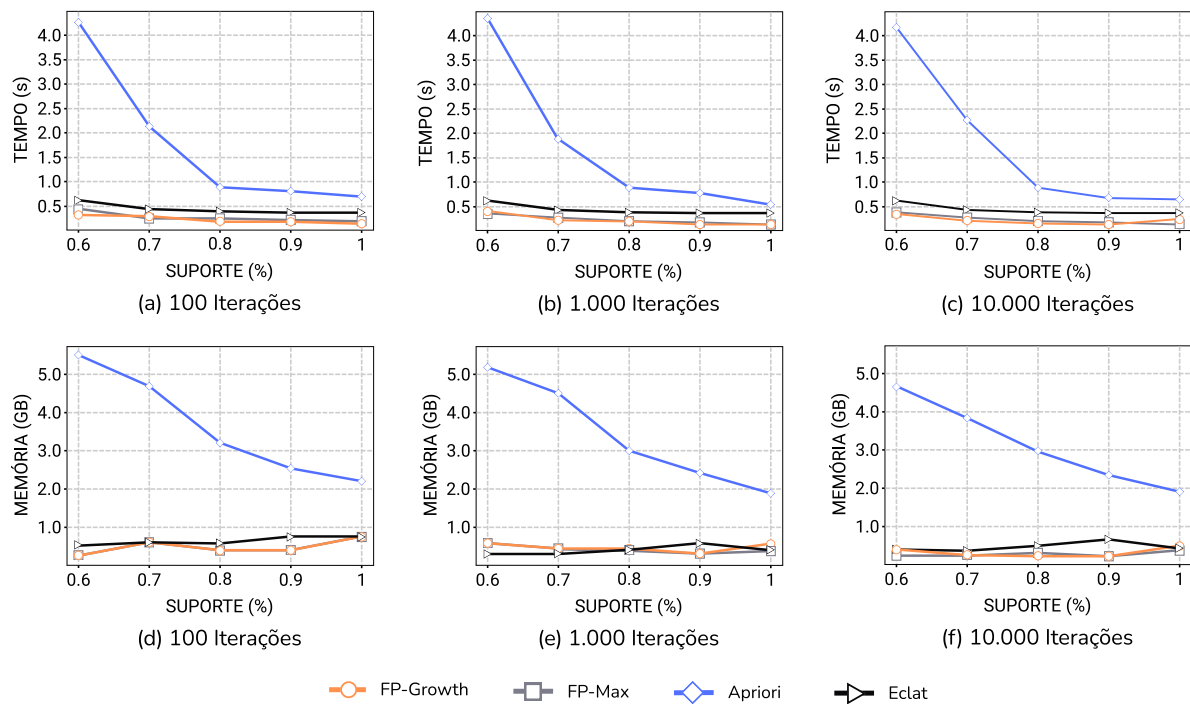
Esta seção apresenta os resultados das simulações de sensibilidade conduzidas em um conjunto sintético de 73 mil registros, gerado a partir dos dados originais da plataforma Hambre Delivery. A análise pretende avaliar a escalabilidade, o tempo de execução e o uso de memória dos algoritmos, a fim de identificar o melhor desempenho, possíveis gargalos e limites operacionais.

7.3.1 Experimento de Extração de Itens Frequentes

Conforme a proposta deste experimento, foi comparado o desempenho dos algoritmos FP-Growth, FP-Max, Apriori e Eclat. A Figura 16 apresenta os resultados da extração de itens

frequentes, relativos ao tempo de execução e ao uso de memória, em três cenários de Monte Carlo definidos com 100, 1.000 e 10.000 iterações. A Figura 16 (a), (b) e (c) ilustram as simulações de tempo de execução. A Figura 16 (d), (e) e (f) ilustram as simulações de uso de memória.

Figura 16 - Resultados da Extração de Itens Frequentes Relativos ao Tempo de Execução e Uso de Memória nas Simulações de Sensibilidade



Fonte: Resultados da pesquisa.

Os resultados de extração de itens frequentes revelam coerência entre os recursos computacionais utilizados, indicando que a oscilação no tempo de execução foi proporcional ao uso de memória. Além disso, o desvio padrão foi praticamente nulo nas variações do Monte Carlo, resultando em padrões semelhantes. Mesmo que as simulações de tempo e memória tenham sido realizadas em momentos distintos para evitar sobrecarga, ambas apresentaram resultados compatíveis, reforçando a validade dos dados.

Complementando a análise, a Tabela 7 sintetiza os resultados das simulações de itens frequentes, detalhando o tempo acumulado pelos algoritmos FP-Growth, FP-Max, Apriori e Eclat em cada cenário de Monte Carlo, enquanto a Tabela 8 informa a memória média utilizada.

Tabela 7 - Tempo Acumulado pelos Algoritmos nas Simulações de Sensibilidade para a Extração de Itens Frequentes

Monte Carlo	Suportes (%)	Tempo (s)			
		FP-Growth	FP-Max	Apriori	Eclat
100	0.6 - 1	143.10	148.03	853.18	222.65
1.000	0.6 - 1	1.478,16	1.489,02	8.366,04	2.123,99
10.000	0.6 - 1	15.206,10	15.157,68	86.905,42	21.459,86
Total (s)		16.827,36	16.794,73	96.124,64	23.806,50
Total (h)		4.68	4.67	26.62	6.62

Fonte: Resultados da pesquisa.

Tabela 8 - Memória Média Utilizada pelos Algoritmos nas Simulações de Sensibilidade para a Extração de Itens Frequentes

Monte Carlo	Suportes (%)	Memória (GB)			
		FP-Growth	FP-Max	Apriori	Eclat
100	0.6 - 1	0.403	0.404	3.5	0.471
1.000	0.6 - 1	0.459	0.431	3.7	0.422
10.000	0.6 - 1	0.187	0.150	3.1	0.285
Média Total (GB)		0.350	0.329	3.43	0.393

Fonte: Resultados da pesquisa.

Quanto ao desempenho dos modelos, pode-se perceber que os algoritmos FP-Growth e FP-Max obtiveram os melhores resultados nos critérios tempo de execução e uso de memória. Os valores registrados para o tempo acumulado em horas indicam que as simulações foram concluídas em 4.68 e 4.67 (h), com uso médio de memória de 0.350 e 0.329 (GB), respectivamente.

Nas mesmas condições, o algoritmo Apriori revelou o pior resultado entre os modelos analisados, levando cerca de 26.62 (h) para realizar as simulações. Esse desempenho mais lento aumentou o uso de memória, requisitando uma média de 3.43 (GB). Outros *scripts* de monitoramento tiveram que ser interrompidos, pois a divisão de tarefas simultâneas limitava o uso de memória, gerando gargalos.

Por outro lado, o algoritmo Eclat demonstrou desempenho melhor que o Apriori e mais próximo dos modelos FP-Growth e FP-Max, sendo necessárias 6.62 (h) para concluir as simulações. Esse aspecto é validado pela memória média registrada, com uso de 0.393 (GB).

Contribuindo para a análise, a Tabela 9 informa a quantidade de itens frequentes extraídos pelos algoritmos nas variações de suporte 0.6%, 0.8% e 1%. Os resultados destacam que os algoritmos FP-Growth, Apriori e Eclat obtiveram a mesma quantidade de itens frequentes, enquanto o FP-Max extraiu uma quantidade reduzida, coerente com sua lógica de identificar itens frequentes máximos.

Tabela 9 - Quantidade de Itens Frequentes Obtidos pelos Algoritmos FP-Growth, FP-Max, Apriori e Eclat nas Simulações de Sensibilidade

Suporte (%)	Algoritmos			
	FP-Growth	FP-Max	Apriori	Eclat
0.6	737	397	737	737
0.8	517	307	517	517
1	410	230	410	410

Fonte: Resultados da pesquisa.

A partir dos resultados discutidos, é possível concluir que os algoritmos FP-Growth e FP-Max são mais escaláveis para lidar com grandes volumes de transações e alta cardinalidade de itens frequentes por transação. Essa adequação proporcionou os melhores resultados em tempo de execução e uso de memória. Contudo, a escolha entre FP-Growth e FP-Max depende do tipo de análise, da complexidade da base de dados e da quantidade de itens frequentes que se pretende extrair.

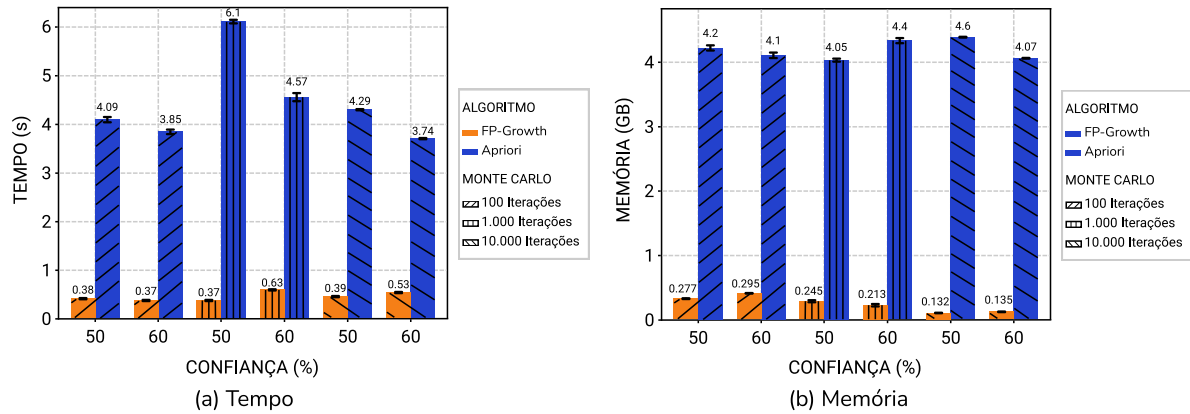
7.3.2 Experimento de Geração de Regras de Associação

Considerando o enfoque deste experimento e os resultados consistentes obtidos nas simulações exploratórias, foram mantidas as mesmas configurações de suporte em 0.6% e as variações de confiança em 50% e 60%.

A Figura 17 ilustra o desempenho dos algoritmos FP-Growth e Apriori na geração de regras de associação, sob os aspectos de tempo de execução e uso de memória. Cada simulação de Monte Carlo é identificada por linhas de variação sobre as barras dos algoritmos. As linhas inclinadas à direita representam o cenário com 100 iterações. As linhas verticais ilustram o

cenário de 1.000 iterações. Por fim, as linhas inclinadas à esquerda representam a simulação de 10.000 iterações.

Figura 17 - Resultados das Simulações de Sensibilidade para a Geração de Regras de Associação



Fonte: Resultados da pesquisa.

Os resultados da geração de regras de associação demonstram consistência entre os recursos computacionais analisados, revelando que as oscilações no tempo de execução são proporcionais ao uso de memória, além de apresentarem redução no desvio padrão conforme o aumento das iterações nos cenários de Monte Carlo.

Em termos computacionais, as simulações de tempo de execução e de memória foram realizadas separadamente para evitar possíveis gargalos. A Tabela 10 sintetiza os resultados das simulações de regras de associação, detalhando o tempo acumulado pelos algoritmos FP-Growth e Apriori em cada cenário de Monte Carlo, enquanto a Tabela 11 detalha a memória média utilizada.

Tabela 10 - Tempo Acumulado pelos Algoritmos nas Simulações de Sensibilidade para a Geração de Regras de Associação

Monte Carlo	Confiança (%)	Tempos (s)	
		FP-Growth	Apriori
100	50 - 60	76,12	795,67
1.000	50 - 60	1.011,09	10.675,16
10.000	50 - 60	9.313,27	80.407,33
Total (s)		10.400,48	91.878,16
Total (h)		2,89	28,41

Fonte: Resultados da pesquisa.

Tabela 11 - Memória Média Utilizada pelos Algoritmos nas Simulações de Sensibilidade para a Geração de Regras de Associação

Monte Carlo	Confiança (%)	Memória (GB)	
		FP-Growth	Apriori
100	50 - 60	0.286	4.08
1.000	50 - 60	0.229	4.12
10.000	50 - 60	0.134	4.23
Média Total (GB)		0.216	4.14

Fonte: Resultados da pesquisa.

Em relação à análise de desempenho dos modelos, pode-se observar que o algoritmo FP-Growth apresentou os melhores resultados, levando cerca de 2.89 (h) para concluir as simulações. Esse desempenho refletiu no baixo consumo de memória apresentado, com uso médio de 0.216 (GB).

Em condições semelhantes, o algoritmo Apriori apresentou os piores resultados, sendo necessário 28.41 (h) para realizar as simulações. Esse desempenho mais lento contribuiu para o uso elevado de memória, que demandou uma média de 4.14 (GB).

A Tabela 12 complementa a análise, detalhando a quantidade de regras de associação geradas pelos algoritmos Apriori e FP-Growth nas variações de confiança de 50% e 60%. Como ambos atuam sob as mesmas condições de configuração, as regras de associação obtidas foram as mesmas.

Tabela 12 - Quantidade de Regras de Associação Obtidas pelos Algoritmos FP-Growth e Apriori

Confiança (%)	Algoritmos	
	FP-Growth	Apriori
50	795	795
60	269	269

Fonte: Resultados da pesquisa.

Diante dos resultados obtidos, é possível concluir que o algoritmo FP-Growth demonstrou ser mais escalável na mineração de grandes volumes de dados, especialmente no tempo de execução e no uso de memória. Mesmo que o algoritmo Apriori tenha gerado a mesma quantidade de regras de associação, sua lógica exige elevado consumo computacional, comprometendo seu desempenho em cenários mais complexos.

7.4 Resultados das Tendências de Venda da Plataforma Hambre Delivery

Esta seção apresenta os resultados da simulação de tendências de venda aplicada mediante o algoritmo FP-Growth nos 622 registros de transações da plataforma Hambre Delivery. Por se tratar de uma análise em escopo reduzido, foram adotados os parâmetros de suporte em 1% e confiança de 50%, que resultaram em 27 regras de associação referentes aos setores de salgado, sobremesa, bebida regional, pizzaria e refeição. Cada categoria será discutida separadamente, a fim de facilitar o entendimento dos padrões de venda e promover uma análise mais específica.

7.4.1 Salgado e Sobremesa

A rotatividade do segmento de sobremesa e salgado está fortemente ligada aos esforços de inovação dos empreendedores familiares, que buscam manter o interesse de seus clientes. Desse modo, a Tabela 13 destaca os padrões relevantes desses setores, com associações rotuladas de A1 até A7. Ambas as categorias foram reunidas na mesma tabela devido à relação entre elas e à quantidade restrita de padrões da categoria sobremesa.

Tabela 13 - Regras de Associação Referentes às Categorias Doce e Salgado

ID	Antecedente	Consequente	Suporte (%)	Confiança (%)	Lift
A1	Pão Careca	Pão Hambúrguer	1.61	100	51.83
A2	Pão Hambúrguer	Pão Careca	1.61	83.33	51.83
A3	Lanche de Empadas, Canudinho com Creme de Frango	Salgados	1.29	80	9.21
A4	Pastel Risóle	Refrigerante Guaraná	1.45	75	6.14
A5	Lanche de Empadas, Salgados	Canudinho com Creme de Frango	1.29	66.67	14.81
A6	Canudinho com Creme de Frango	Salgados	2.89	64.29	7.4
A7	Doce de Brigadeiro	Salgados	1.77	50	5.76

Fonte: Resultados da pesquisa.

Quanto às tendências registradas na Tabela 13, pode-se perceber que as regras A1 e A2 são complementares na montagem de lanches. Isso indica que os itens *Pão Careca* e *Pão Hambúrguer* são frequentemente adquiridos juntos, tendo 100% de confiança na relação direta e 83.33% na inversa, sugerindo uma associação extremamente forte.

Também se destacam as regras A3, A5 e A6, que indicam a preferência dos clientes pela combinação de *Lanches de Empadas*, *Canudinho com creme de frango* e *Salgados*. A regra A3 revela que essa combinação ocorre em 80% dos casos, sendo uma tendência recorrente de lanches mistos para mais pessoas. Outro padrão relevante é a associação A4 entre *Pastel Risóle* e *Refrigerante Guaraná*, que sugere uma busca de 75% pela formação de combos para consumo imediato, devido à praticidade de adquiri-los juntos a um preço acessível.

Além disso, a aquisição simultânea de itens como *Doce de Brigadeiro* e *Salgados*, representada pela regra A7, contribui para a composição de menus equilibrados de lanche e sobremesa. A confiança de 50% sugere que essa combinação tende a ocorrer em datas comemorativas, o que justifica sua frequência ocasional em relação às demais.

7.4.2 Bebida Regional

A região do Baixo Tocantins apresenta um mercado interno expressivo do segmento de bebida regional, sendo um aspecto fortemente incorporado à cultura amazônica. A Tabela 14 detalha 4 regras de associação, referentes a esse setor, rotuladas de B1 até B4.

Tabela 14 - Regras de Associação Referentes à Categoria Bebida Regional

Nº	Antecedente	Consequente	Suporte (%)	Confiança (%)	Lift
B1	Farinha de Mandioca	Açaí de 1 Litro	4.34	84.38	14.58
B2	Açaí de 1 Litro	Farinha de Mandioca	4.34	75.0	14.58
B3	Açaí de Meio Litro	Açaí de 1 Litro	2.09	72.22	12.48
B4	Açaí de Meio Litro	Farinha de Mandioca	1.45	50.0	9.72

Fonte: Resultados da pesquisa.

Considerando as tendências dispostas na Tabela 14, as regras B1 e B2 indicam uma prática cultural consolidada no consumo de *Farinha de Mandioca* e *Açaí de 1 Litro*. Por serem complementares a qualquer refeição, os consumidores que compram *Farinha de Mandioca* frequentemente levam o *Açaí de 1 Litro*, com 84.38% de confiança na relação direta e 75% na inversa. Essa frequência confirma uma associação positiva muito superior ao que seria esperado por acaso.

De forma semelhante, a regra B3 evidencia que a associação entre *Açaí de Meio Litro* e *Açaí de 1 Litro* ocorre em 72.22% das compras. Esse resultado permite levantar hipóteses que explicam a demanda pelo produto, como a aquisição para diferentes membros da família, a preferência por tamanhos variados e o hábito de comprá-lo em diferentes quantidades.

Enquanto a regra B4 demonstra que a combinação entre *Açaí de Meio Litro* e *Farinha de Mandioca* é preferível em 50% das compras, sendo um hábito de consumo relacionado a clientes que moram sozinhos, o que justifica uma demanda reduzida desse grupo.

7.4.3 Pizzaria

O setor de pizzarias demonstrou ser o ramo de maior preferência entre os clientes do *app* *Hambre Delivery*, destacando-se pelas combinações de sabores e opções de bebidas. A Tabela 15 reúne 14 regras de associação, rotuladas de C1 até C14.

Tabela 15 - Regras de Associação Referentes à Categoria Pizza

Nº	Antecedente	Consequente	Suporte (%)	Confiança (%)	Lift
C1	Pizza Metade Bacon	Pizza Metade Portuguesa	1.93	100.0	31.1
C2	Pizza Metade 5 Queijos, Refrigerante Coca-Cola	Pizza Metade Calabresa	2.41	100.0	12.69
C3	Pizza Metade Frango	Pizza Metade Mussarela	1.29	100.0	25.92
C4	Pizza Peito de Peru	Suco de Abacaxi	1.13	100.0	69.11
C5	Pizza Metade 5 Queijos, Pizza Metade Calabresa	Refrigerante Coca-Cola	2.41	88.24	5.6
C6	Pizza Metade Filé com Bacon	Pizza Metade Calabresa	1.13	87.5	11.11
C7	Pizza Metade 5 Queijos	Pizza Metade Calabresa	2.73	85.0	10.79
C8	Suco de Abacaxi	Pizza Peito de Peru	1.13	77.78	69.11
C9	Pizza Metade 5 Queijos	Pizza Metade Calabresa, Refrigerante Coca-Cola	2.41	75.0	22.21
C10	Pizza Metade 5 Queijos	Refrigerante Coca-Cola	2.41	75.0	4.76
C11	Pizza Metade Calabresa, Refrigerante Coca-Cola	Pizza Metade 5 Queijos	2.41	71.43	22.21
C12	Pizza Metade Portuguesa	Pizza Metade Bacon	1.93	60.0	31.1
C13	Pizza Calabresa	Refrigerante Coca-Cola	1.45	60.0	3.81
C14	Pizza Metade Mussarela	Pizza Metade Calabresa	2.25	58.33	7.4

Fonte: Resultados da pesquisa.

Conforme os resultados apresentados na Tabela 15, as regras C1 e C12 revelam uma associação bidirecional entre os sabores bacon e portuguesa. Essa combinação representa uma tendência mais clássica, na qual os clientes que escolhem a *Pizza Metade Bacon* tendem a combiná-la com a *Pizza Metade Portuguesa*, com 100% de confiança na escolha direta e 60% na inversa.

Por sua vez, as regras C2, C5, C9 e C11 evidenciam a preferência dos clientes pela combinação de *Pizza Metade 5 Queijos*, *Refrigerante Coca-Cola* e *Pizza Metade Calabresa*. A associação C2 indica que essa relação ocorre em 100% das transações, sendo uma tendência recorrente entre grupos de amigos e famílias.

Seguindo um padrão mais popular entre os consumidores, os sabores frango e mussarela representam uma escolha mais confiável. A regra C3 revela que a combinação *Pizza Metade Frango* e *Pizza Metade Mussarela* ocorre com uma confiança de 100%, sendo uma opção mais versátil para agradar diferentes paladares.

As regras C4 e C8 destacam a preferência por ingredientes ou produtos mais suaves, relacionando *Pizza Peito de Peru* e *Suco de Abacaxi*, com 100% de confiança na escolha direta e 77.78% na inversa. Por outro lado, a regra C6 associa *Pizza Metade Filé* com *Bacon* e *Pizza Metade Calabresa*, indicando uma preferência de 87.5% por sabores mais intensos e encorpados.

Considerando o equilíbrio entre suave e intenso, as regras C7 e C14 revelam que a escolha da *Pizza Metade 5 Queijos* costuma ser associada à *Pizza Metade Calabresa*, com 85% de confiança. Assim como, a *Pizza Metade Mussarela* geralmente é combinada com a *Pizza Metade Calabresa*, com 58.33% de confiança, reforçando um padrão de contraste entre sabores.

Por fim, as regras C10 e C13 indicam uma tendência de consumo que associa pizzas de sabores populares ao *Refrigerante Coca-Cola*. A regra C10 destaca que a combinação entre *Pizza Metade 5 Queijos* e *Refrigerante Coca-Cola* ocorre com 75% de confiança, enquanto a regra C13 sugere que 60% dos clientes que escolhem *Pizza Calabresa* também incluem *Refrigerante Coca-Cola* em seus pedidos, indicando uma preferência consolidada por essa bebida como acompanhamento.

7.4.4 Refeição

Em relação ao setor de refeições, foi identificada uma preferência por refeições de pronta entrega, especialmente entre consumidores que possuem uma rotina agitada, com pouco tempo disponível para cozinhar. A Tabela 16 descreve 2 regras de associação, rotuladas de D1 a D2.

Tabela 16 - Regras de Associação Referentes à Categoria Refeição

Nº	Antecedente	Consequente	Suporte (%)	Confiança (%)	Lift
D1	Calabresa	Misto de Carne	1.77	100.0	47.85
D2	Misto de Carne	Calabresa	1.77	84.62	47.85

Fonte: Resultados da pesquisa.

Considerando os resultados da Tabela 16, as regras D1 e D2 evidenciam uma associação bidirecional entre as refeições de calabresa e carne. Essa combinação indica a preferência de um perfil específico de consumidor, em que 100% dos clientes que compram *Calabresa* tendem a levar *Misto de Carne*, enquanto 84.62% realizam a combinação inversa. Essa relação quase obrigatória entre os pratos sugere uma opção de refeição combinada que reforce esse hábito.

7.5 Conclusão

Este Capítulo apresentou os resultados das simulações propostas para identificar o algoritmo mais adequado à tarefa de extrair regras de associação, além de discutir as tendências de venda descobertas na base de dados da plataforma Hambre Delivery. As análises foram conduzidas considerando a capacidade dos algoritmos de manter seu desempenho mesmo quando as condições experimentais são alteradas.

As simulações realizadas na Seção 6.2 demonstraram que o algoritmo Eclat possui a capacidade de processamento mais adequada para extrair itens frequentes em conjuntos de transações pequenos e com baixa cardinalidade, o que reduziu o tempo médio registrado nas simulações de Monte Carlo. Enquanto o algoritmo FP-Growth evidenciou ser mais escalável na geração de regras de associação, devido à sua lógica baseada na FP-Tree, que otimiza o processo de mineração, reduzindo o tempo de execução.

Por outro lado, as simulações de sensibilidade, detalhadas na Seção 6.3, revelaram que os algoritmos FP-Growth e FP-Max foram os mais eficientes para extrair itens frequentes, obtendo tempos de execução muito baixos e uso reduzido de memória. Em termos de versatilidade, a lógica da FP-Tree utilizada por ambos se adequa melhor às bases de dados maiores, especialmente sob extrema carga de uso. Definir qual deles utilizar depende da complexidade do conjunto de transações e da quantidade de itens que se pretende extrair, uma vez que o FP-Max retorna apenas os itens frequentes máximos.

Na mesma Seção, as simulações de geração de regras de associação confirmaram o desempenho do algoritmo FP-Growth, evidenciado na Seção anterior. Mesmo com o aumento no volume de transações, o tempo de execução requisitado ao sistema foi condizente com o uso de memória, revelando o FP-Growth como o mais eficiente em termos de escalabilidade, nas simulações de Monte Carlo.

Quanto à análise de tendências de venda na Seção 6.4, foram obtidas 27 regras de associação sobre os padrões mais consumidos pelos clientes no *app* Hambre Delivery. Nesse aspecto, os setores com maior volume de vendas são salgado, sobremesa, pizzaria, refeição e bebida regional. Dentre eles, a pizzaria se destacou como o ramo de maior preferência.

A partir dos resultados, os algoritmos FP-Growth e FP-Max são alternativas viáveis para integrar ao sistema de recomendações da plataforma Hambre Delivery, apresentando a escalabilidade necessária a longo prazo e contribuindo com as melhores opções de venda para clientes e lojas.

8 CONCLUSÃO

8.1 Considerações Finais

Esta dissertação alcançou o objetivo principal de avaliar o desempenho computacional de algoritmos de mineração de regras de associação, combinando o Método de Monte Carlo para obter resultados mais precisos e identificar o algoritmo mais eficiente para suportar a decisão no *app* Hambre Loja.

O alcance dessa finalidade requereu a realização de metas específicas, como a seleção do conjunto de transações da plataforma Hambre Delivery, a implementação dos algoritmos Apriori, FP-Growth, FP-Max e Eclat, a definição de configurações e cenários de Monte Carlo com base em simulações exploratórias e de sensibilidade, além da criação de scripts para coletar e avaliar o desempenho quanto à escalabilidade, tempo de execução e uso de memória.

A aplicação de métodos combinados, tendo em vista o desempenho computacional, revelou duas realidades distintas, conforme os experimentos. Na primeira, foi observado que a aplicação de simulações exploratórias em um conjunto de transações pequeno indicou que o algoritmo Eclat cumpriu melhor a tarefa de extrair itens frequentes, em virtude da baixa cardinalidade por transação, enquanto o algoritmo FP-Growth se mostrou mais eficiente para gerar regras de associação, principalmente pelo uso da FP-Tree. Além de ambos serem escaláveis quanto ao tempo de execução, a proposta dos algoritmos pode ser generalizada para ambientes em que as vendas estão em fase inicial e não requerem modelos extremamente complexos e robustos.

Na segunda, as simulações de sensibilidade aplicadas em um conjunto de dados sintético com mais transações e maior cardinalidade de itens por transação revelaram que ambos os algoritmos FP-Growth e FP-Max foram eficientes na extração de itens frequentes, mas o algoritmo FP-Growth também foi eficaz na geração de regras de associação, uma vez que ambos utilizam a FP-Tree para otimizar essa tarefa de mineração de dados. Por esse motivo, foram escaláveis em relação ao tempo de execução e ao uso de memória, comprovando que essa abordagem pode ser ampliada para ambientes de vendas consolidados no mercado de *food service*, que apresentam conjuntos de dados mais complexos e necessitam lidar com tendências de vendas mais precisas.

Por mais que esta dissertação utilize métodos de análise conhecidos, adaptando critérios e diferentes métricas, nenhum trabalho apresenta uma abordagem que combine o método de Monte Carlo e algoritmos de mineração de dados baseados em regras de associação, especificamente para avaliar o desempenho computacional.

Diante deste trabalho, é possível concluir que a abordagem proposta é viável para apoiar a escolha do modelo mais estável como suporte à decisão de tendências de venda. Reitera-se que a aplicação de simulações de Monte Carlo a partir de 1.000 iterações resultará em resultados mais precisos, demonstrando que essa sistematização de testes pode ser generalizada para outros contextos de mineração de dados, seja em conjuntos de dados mais simples ou mais robustos.

8.2 Trabalhos Futuros

Para trabalhos futuros, são propostos os seguintes itens:

- Integrar *scripts* ao código-fonte do projeto no servidor para pré-processamento e transformação da base de dados transacional da plataforma Hambre Delivery;
- Expandir o código-fonte do projeto para integrar o algoritmo de regras de associação mais eficiente à versão final do app Hambre Loja;
- Disponibilizar uma nova versão do *app* Hambre Loja no Google Play para lojistas.

8.3 Produções Acadêmicas

Durante a realização desta dissertação, foi apresentado um trabalho em comunicação oral e aprovado um artigo resultante da pesquisa. Além disso, foram desenvolvidas outras produções acadêmicas, como um artigo atualmente em processo de avaliação e o registro de softwares, cujos temas não estão diretamente relacionados a esta dissertação.

Destaca-se que parte do desenvolvimento deste trabalho foi apresentada em comunicação oral no I Seminário de Extensão do CUTINS/UFPA – 2024, com o título “Análise de Desempenho de Algoritmos de Mineração de Dados Aplicados ao Banco de Dados do E-commerce Hambre Delivery”. O artigo intitulado “Avaliação de Desempenho de Algoritmos de Mineração de Dados e Simulação de Monte Carlo na Descoberta de Tendências no App Hambre Delivery”, aprovado pela revista RECIMA21.

Além disso, foram realizados dois registros de software no Instituto Nacional da Propriedade Industrial (INPI), com a participação da autora desta dissertação. Eles podem ser citados como:

- TOSTES, M. E. L.; BEZERRA, U. H.; MUNIZ, J. R. S.; RODRIGUES, C. E. M.; FARIAS, F. S.; GONÇALVES, L. N.; LIMA, W. G.; MELO, L. A.; SÁ, J. S.; SILVA, E. N.; MACHADO, C. A. S.; CAVALCANTE, G. P. S.; BARROS, F. J. B.; ARAÚJO, J. P. L.; ALCÂNTARA NETO, M. C.; BARBOSA, Y. H. S.; CRUZ, H. A. O.; CARVALHO, I. S.; LOBATO, E. P. S.; ANDRADE, V. B.; DUARTE, A. R. C. L. M.; LIMA, Á. S.; MORAES, W. L. S.; ALBUQUERQUE, B. S.; PAIXÃO JUNIOR, U. C.; MORAES, J. L. A. A. M.; MOCBEL, M. A. R.; CARVALHO, J. A. R.; CARDOSO, C. M. M.; MACHADO, A.; CARVALHO, C. C. M. M.; CERQUEIRA, E. C.; ROSÁRIO, D. L.; ARAÚJO, F. R.; MEDEIROS, I. L.; PRAZERES, L. C.; COSTA, S. C. P.; FONTENELE, G. M.; FREITAS, P. H. O.; TABORA, J. M.; BATALHA, I. S.; LOPES, A. V. R. **Sistema Inteligente de Gestão Eficiente de Mobilidade Elétrica Multimodal**. Titular: Universidade Federal do Pará; Norte Energia S.A. Procurador: José Carlos Vaz e Dias. Número do registro: BR512024000965-5, data de registro: 03/04/2023, Instituição de registro: INPI - Instituto Nacional da Propriedade Industrial.
- FARIAS, F. S.; GONÇALVES, L. N.; MOCBEL, M. A. R.; SÁ, J. S.; GARCIA, L. MELO; DIAS, R. F.; OLIVEIRA, R. B. S.; RODRIGUES, P. R. C.; CASTRO, V. C.; CARMO, A. A. **Exatas**. Titular: Universidade Federal do Pará. Procurador: José Carlos Vaz e Dias. Número do registro: BR512025002643-9, data de registro: 26/06/2025, Instituição de registro: INPI - Instituto Nacional da Propriedade Industrial.

REFERÊNCIAS

- AGRAWAL, R.; IMIELIŃSKI, T.; SWAMI, A. Mining association rules between sets of items in large databases. **ACM SIGMOD Record**, v. 22, n. 2, p. 207–216, 1993.
- AGRAWAL, R.; SRIKANT, R. **Fast Algorithms for Mining Association Rules**. Proceedings of the 20th International Conference on Very Large Data Bases (VLDB). **Anais...**Morgan Kaufmann, 12 set. 1994.
- AJI, G. W.; DEVI, P. A. R. Data Mining Implementation For Product Transaction Patterns Using Apriori Method. **Sinkron**, v. 8, n. 1, p. 421–432, 2023.
- ARENAS, L. A.; PAUCAR, I. R.; ARIAS, C. Y. Data mining for predictive analysis in gynecology: a focus on cervical health. **International Journal of Electrical and Computer Engineering (IJECE)**, v. 14, n. 3, p. 2822, 1 jun. 2024.
- BHAGAMPRIYAL, M. et al. **Recommendation Systems for Supermarket**. 3rd International Conference on Innovative Practices in Technology and Management (ICIPTM). **Anais...**Uttar Pradesh, India: IEEE, 2023.
- BUJALANCE-LÓPEZ, L. et al. Restaurant revenue management: a systematic literature review and future challenges. **British Food Journal**, v. 127, n. 6, p. 2169–2196, 1 abr. 2025.
- CHEN, Z.; HUANG, H.; YANG, C. Comparative Analysis of Bootstrap Techniques for Confidence Interval Estimation in Spatial Covariance Parameters With Large Spatial Data. **Environmetrics**, v. 36, n. 3, 2025.
- CHERNICK, M. R. et al. Bootstrap Methods. In: LOVRIC, M. (Ed.). **International Encyclopedia of Statistical Science**. Berlin: Springer, 2011. p. 169–174.
- COFFMAN, J.; WEAVER, A. C. An Empirical Performance Evaluation of Relational Keyword Search Techniques. **IEEE Transactions on Knowledge and Data Engineering**, v. 26, n. 1, p. 30–42, 1 Jan. 2014.
- CUI, Y. et al. Network Analysis of Two-Stage Customer Decisions with Preference-Guided Market Segmentation. **Journal of Computing and Information Science in Engineering**, v. 25, n. 6, p. 15, 4 abr. 2025.

ECKMAN, D. J.; HENDERSON, S. G.; SHASHAANI, S. Diagnostic Tools for Evaluating and Comparing Simulation-Optimization Algorithms. **INFORMS journal on computing**, v. 35, n. 2, p. 350–367, 2023.

FADILAH, E.; FAQIH, A.; PERMANA, S. E. Using the Apriori Algorithm to Identify Purchase Patterns for Enhancing Sales in Personal Shopper Services. **Journal of Artificial Intelligence and Engineering Applications (JAIEA)**, v. 4, n. 2, p. 745–751, 2025.

FARIAS, F. et al. Cost- and energy-efficient backhaul options for heterogeneous mobile network deployments. **Photonic Network Communications**, v. 32, n. 3, p. 422–437, 22 Nov. 2016.

FAYYAD, U.; PIATETSKY-SHAPIO, G.; SMYTH, P. From Data Mining to Knowledge Discovery in Databases. **AI Magazine**, v. 17, n. 3, p. 37–37, 15 mar. 1996.

FAYYAD, U.; STOLORZ, P. Data mining and KDD: Promise and challenges. **Future Generation Computer Systems**, v. 13, n. 2-3, p. 99–115, Nov. 1997.

GONÇALVES, K. R. G. **Desenvolvimento e Avaliação do Aplicativo Hambre Delivery: Apresentação do Produto e Análise de Desempenho Usando Inteligência Artificial**. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Faculdade de Sistemas de Informação—Cametá: UFPA, 2024.

GRAHNE, G.; ZHU, J. Fast algorithms for frequent itemset mining using FP-trees. **IEEE Transactions on Knowledge and Data Engineering**, v. 17, n. 10, p. 1347–1362, out. 2005.

GRIJALBA, M. A. et al. Does the use of digital tools improve a firm's performance? **Review of Managerial Science**, v. 19, p. 2193–2210, 2 mar. 2024.

GUPTA, B. B. et al. Big data innovations in enterprise information systems: strategies formation for new generation entrepreneurs. **Enterprise Information Systems**, v. 19, n. 1-2, 6 Jan. 2025.

HALL, M. A.; HOLMES, G. Benchmarking Attribute Selection Techniques for Discrete Class Data Mining. **IEEE Transactions on Knowledge and Data Engineering**, v. 15, n. 6, p. 1437–1447, Nov. 2003.

HAN, J. et al. Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach. **Data Mining and Knowledge Discovery**, v. 8, n. 1, p. 53–87, Jan. 2004.

HAN, J.; KAMBER, M.; PEI, J. **Data Mining: Concepts and Techniques**. 3. ed. Burlington: Elsevier, 2012.

HANUM, A. Y.; WITANTI, A. Food Menu Arrangement Recommendations Using Transaction History Data Using the Apriori Method (Case Study: Bebek Pondok Galih). **Jurnal Disprotek**, v. 14, n. 2, p. 149–155, 2023.

HASTINGS, W. K. Monte Carlo sampling methods using Markov chains and their applications. **Biometrika**, v. 57, n. 1, p. 97–109, 1 abr. 1970.

HEGLAND, M. Data mining techniques. **Acta Numerica**, v. 10, p. 313–355, 2001.

HEIJUNGS, R. On the number of Monte Carlo runs in comparative probabilistic LCA. **The International Journal of Life Cycle Assessment**, v. 25, n. 2, p. 394–402, 2020.

HU, T. et al. Discovery of maximum length frequent itemsets. **Information Sciences**, v. 178, n. 1, p. 69–87, Jan. 2008.

IVERSON, M. A.; F. OZGUNER; POTTER, L. C. **Statistical prediction of task execution times through analytic benchmarking for scheduling in a heterogeneous environment**. Proceedings. Eighth Heterogeneous Computing Workshop (HCW'99). **Anais...**San Juan, USE: IEEE, 2002.

JAIN, R. **The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling**. 1. ed. New York: Wiley, 1991. p. 685

JOSEPHINE, H.; RAJAN, D. **Enhancing Customer Experience and Sales Performance in a Retail Store Using Association Rule Mining and Market Basket Analysis**. 14th International Conference on Computing Communication and Networking Technologies (ICCCNT). **Anais...**Delhi, India: IEEE, 2023.

KAMALOV, F.; MOUSSA, S.; REYES, J. A. Regularized Information Loss for Improved Model Selection. **Lecture Notes on Data Engineering and Communications Technologies**, v. 171, p. 677–686, 2023.

KANSAL, M. et al. **Implementing Market Basket Analysis Using Eclat and Apriori Algorithm on Grocery Product Marketing Strategy**. 3rd International Conference on Advancement in Electronics & Communication Engineering (AECE). **Anais...GHAZIABAD: IEEE, 2023.**

KUMAR, V.; GUPTA, A. **Analysis of scalability of parallel algorithms and architectures**. Proceedings of the 5th international conference on Supercomputing - ICS '91. **Anais...New York: ACM Press, 1991.**

LARA, J. A. et al. Data preparation for KDD through automatic reasoning based on description logic. **Information Systems**, v. 44, p. 54–72, 2014.

LAWAL, M. M.; MATTHEW, O. T. FP-Growth Algorithm: Mining Association Rules without Candidate Sets Generation. **KASU Journal of Computer Science**, v. 1, n. 2, p. 392–411, 30 jun. 2024.

LIN, W.-Y.; TSENG, M.-C.; SU, J.-H. A Confidence-Lift Support Specification for Interesting Associations Mining. **Advances in Knowledge Discovery and Data Mining**, v. 2336, p. 148–158, 2002.

LIU, W. et al. Mobile platform expansion: How does it affect the incumbent food delivery app and other sales channels? **Journal of Retailing**, v. 100, n. 3, p. 422–438, 1 jun. 2024.

LUHAYB, A. The bootstrap method for Monte Carlo integration inference. **Journal of King Saud University - Science**, v. 35, n. 6, 2023.

MAAZALAH, M.; HOSSEINI, S. K-means and meta-heuristic Algorithms for Intrusion Detection Systems. **Cluster Computing**, v. 27, n. 8, p. 10377–10419, 2024.

NANDE, P.; PATNE, M.; KELKAR, S. **Evaluating Performance of Apriori, Apriori TID, and FP-Growth Algorithms: A Comparison on Diverse Datasets**. 1º Conferência Internacional sobre Aplicações AIML para Engenharia e Tecnologia (ICAET). **Anais...India: IEEE, 2025.**

OZEMELA, L. **iFood alcança marco histórico de 100 milhões de pedidos em um só mês**. iFood, 2024. Disponível em: <https://institucional.ifood.com.br/noticias/ifood-alcanca-marco-historico-de-100-milhoes-de-pedidos-em-um-so-mes/?utm_source=chatgpt.com>. Acesso em: 11 abr. 2025

PAPAKYRIAKOU, D.; BARBOUNAKIS, I. S. Data Mining Methods: A Review. **International Journal of Computer Applications**, v. 183, n. 48, p. 5–19, 18 Jan. 2022.

PRADANA, M. R. et al. **Market Basket Analysis Using FP-Growth Algorithm on Retail Sales Data**. 9th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI). **Anais...**Jakarta, Indonesia: IEEE, 2022.

RABHI, F.; BEHESHTI, A.; GILL, A. Editorial: Business transformation through AI-enabled technologies. **Frontiers in Artificial Intelligence**, v. 8, 11 mar. 2025.

RAMMELAERE, J.; GEERTS, F. Cleaning Data with Forbidden Itemsets. **IEEE Transactions on Knowledge and Data Engineering**, v. 32, n. 8, p. 1489–1501, 2020.

REZAEI, M. et al. Navigating change: examining the influence of COVID-19 on knowledge-sharing dynamics in family firms within the restaurant and fast-food industry. **Journal of knowledge management**, v. 28, n. 7, 2024.

RITONGA, M. Z.; JULEDI, A. P.; MUTIA, R. Implementation of Data Mining to Determine Sales Patterns Using the Apriori Method. **sinkron**, v. 8, n. 2, p. 1100–1106, 18 abr. 2024.

ROSADI, M.; HASIBUAN, M. S. Comparison of Apriori and FP-Growth Algorithms in Analyzing Association Rules. **PIKSEL Penelitian Ilmu Komputer Sistem Embedded and Logic**, v. 12, n. 2, p. 399–408, 2024.

SHEN, L. et al. On Efficient Training of Large-Scale Deep Learning Models. **ACM Computing Surveys**, v. 57, n. 3, p. 36, 2024.

SILCHENKO, V. Methodological Principles of Managing the Effectiveness of Food Delivery Applications. **Наукові інновації та передові технології**, v. 36, n. 8, 2024.

SISWANTO, B. et al. SDFP-growth Algorithm as a Novelty of Association Rule Mining Optimization. **IEEE access**, v. 12, p. 21491–21502, 1 Jan. 2024.

SUMA, D.; SAGAR, L. Data Mining Techniques. **Technometrics**, v. 48, n. 1, p. 159–160, Fev. 2006.

SUN, X. et al. A Scalable and Flexible Basket Analysis System for Big Transaction Data in Spark. **Information Processing and Management**, v. 61, n. 2, p. 103577–103577, 2024.

TANAKA, T. et al. Optimality Between Time of Estimation and Reliability of Model Results in the Monte Carlo Method: A Case for a CGE Model. **Computational Economics**, v. 59, n. 1, p. 151–176, 3 Jan. 2021.

WANG, Y. **Enhanced Algorithm and Comparative Study of Sales Forecasting Model Based on Data Mining Technology**. International Conference on Optimization Computing and Wireless Communication (ICOCWC). **Anais...**Debre Tabor, Ethiopia: IEEE, 29 Jan. 2024.

WANG, Z. **A Multi-Level Association Rule Mining Algorithm Based on NSGA-II for Market Basket Analysis**. 2023 4th Information Communication Technologies Conference (ICTC). **Anais...**Nanjing, China: IEEE, 17 Maio 2023.

WURST, M.; MORIK, K. Distributed feature extraction in a p2p setting — a case study. **Future Generation Computer Systems**, v. 23, n. 1, p. 69–75, 1 Jan. 2007.

YAN, S.-R. et al. Implementation of a Product-Recommender System in an IoT-Based Smart Shopping Using Fuzzy Logic and Apriori Algorithm. **IEEE Transactions on Engineering Management**, v. 71, p. 4940–4954, 2024.

YILDIZ, E.; ŞEN, C. G.; IŞIK, E. E. A Hyper-Personalized Product Recommendation System Focused on Customer Segmentation: An Application in the Fashion Retail Industry. **Journal of Theoretical and Applied Electronic Commerce Research**, v. 18, n. 1, p. 571–596, 11 mar. 2023.

ZAKI, M. J. et al. New algorithms for fast discovery of association rules. **Knowledge Discovery and Data Mining**, p. 283–286, 14 ago. 1997.

ZAKI, M. J. Scalable algorithms for association mining. **IEEE Transactions on Knowledge and Data Engineering**, v. 12, n. 3, p. 372–390, 2000.

ZARGARI, S.; VOORHIS, D. **Feature Selection in the Corrected KDD-dataset**. Third International Conference on Emerging Intelligent Data and Web Technologies. **Anais...**Bucharest, Romania: IEEE, 2012.

ZHAO, D.; YU, N. **Application of Data Mining Techniques in Enterprise Decision Support Systems**. Proceedings of the 3rd International Conference on Big Data Economy and Digital Management (BDEDM). **Anais...**Ningbo: EAI, 18 jun. 2024.