UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS DOIS VIZINHOS
CURSO DE ESPECIALIZAÇÃO EM CIÊNCIA DE DADOS

GERMANO AUGUSTO METZNER DE ANDRADE

UTILIZAÇÃO DE REGRAS DE ASSOCIAÇÃO PARA IDENTIFICAÇÃO DE COMBOS PRESENTEÁVEIS

TRABALHO DE CONCLUSÃO DE CURSO DE ESPECIALIZAÇÃO

DOIS VIZINHOS
2022

GERMANO AUGUSTO METZNER DE ANDRADE

UTILIZAÇÃO DE REGRAS DE ASSOCIAÇÃO PARA IDENTIFICAÇÃO DE COMBOS PRESENTEÁVEIS

Trabalho de Conclusão de Curso de Especialização apresentado ao Curso de Especialização em Ciência de Dados da Universidade Tecnológica Federal do Paraná, como requisito para a obtenção do título de Especialista em Ciência de Dados.

Orientador: Prof. Dr. Jefferson Tales Oliva

DOIS VIZINHOS
2022

GERMANO AUGUSTO METZNER DE ANDRADE

UTILIZAÇÃO DE REGRAS DE ASSOCIAÇÃO PARA IDENTIFICAÇÃO DE COMBOS PRESENTEÁVEIS

Trabalho de Conclusão de Curso de Especialização apresentado ao Curso de Especialização em Ciência de Dados da Universidade Tecnológica Federal do Paraná, como requisito para a obtenção do título de Especialista em Ciência de Dados.

Data de aprovação: 09/junho/2022

Jefferson Tales Oliva
Doutorado
Universidade Tecnológica Federal do Paraná - Câmpus Pato Branco

Rafael Gomes Mantovani
Doutorado
Universidade Tecnológica Federal do Paraná - Câmpus Apucarana

Marco Antonio de Castro Barbosa
Doutorado
Universidade Tecnológica Federal do Paraná - Câmpus Pato Branco

DOIS VIZINHOS
2022

Dedico esse trabalho ao meu pai Álvaro Augusto Maciel de Andrade (in memorian). Só cheguei até aqui por causa de você. Tenho certeza que você deve estar orgulhoso. Te amo!

AGRADECIMENTOS

A Deus, por ter permitido que eu tivesse saúde, foco e disposição para não desanimar durante a realização deste trabalho e todo o período letivo, com certeza não foi fácil.

Aos familiares que me incentivaram nos momentos difíceis e compreenderam a minha ausência enquanto eu me dedicava à realização deste trabalho.

Ao Prof. Dr. Jefferson Tales Oliva, por ter sido meu orientador, conselheiro e amigo durante esta jornada, meu profundo agradecimento.

Ao Prof. Dr. Rafael Gomes Mantovani e ao Prof. Dr. Marco Antonio de Castro Barbosa pelo o apoio, correções e sugestões fornecidos para a realização deste trabalho.

A Universidade Tecnológica Federal do Paraná, pela oportunidade de fazer o curso.

A todos que direta ou indiretamente fizeram parte de minha formação, o meu muito obrigado.

RESUMO

Com a evolução da tecnologia, o volume de dados vem crescendo significativamente durante os anos, e juntamente disso, a necessidade de se extrair informações de grandes conjuntos de dados foi se tornando cada vez maior. O tema de Descoberta de Conhecimento em Banco de Dados tem como objetivo explorar e extrair informações relevantes até então desconhecidos por especialistas, traduzindo tais percepções em mensagens de fácil interpretação. Este processo é composto por diversas técnicas que demandam grandes esforços computacionais que não seriam possíveis sem o advento da programação. O método de Regras de Associação é um desses procedimentos, onde desempenha a função de localizar relações entre diversas informações, sendo muito utilizado para aprimorar o desempenho das vendas em supermercados. O local de estudo deste trabalho é um estabelecimento comercial de varejo que trabalha com milhares de itens, onde sua dificuldade está em reduzir produtos de baixo fluxo armazenados em estoque. O objetivo deste trabalho é utilizar a técnica de Regras de Associação para gerar combos estatisticamente atrativos e exaurir os itens acumulados em estoque. Para realização de tal estudo, foram feitos levantamentos bibliográficos, sendo escolhido o algoritmo Fp-Growth para desempenhar o processamento das informações, visto sua efetividade em executar grandes conjuntos de dados. Dos treze produtos selecionados, foram encontradas associações promissoras para dois dele. Ao todo, sete regras de associação, para períodos de promoção e não promoção, foram encontradas.

Palavras-chave: Fp-Growth; Regras de Associação; Mineração de Dados.

ABSTRACT

With the evolution of the technology, the volume of data has been growing significantly over the years, and along with that, the necessity to extract information from large datasets has become increasing. The theme of Knowledge Discovery in Databases has the objective to explore and extract relevant information until then unknown by specialists, converting such perceptions into messages that are easy to interpret. This process is composed of several techniques that demand great computational efforts that would not be possible without the advent of programming. The Association Rules method is one of these procedures, where it performs the role of finding relationships between different pieces of information, being widely used to improve the performance of sales in supermarkets. The study local of this work is a retail commercial establishment that works with thousands of items, where its difficulty lies in reducing low-flow products stored in stock. The objective of this work is to use the Association Rules technique to generate statistically attractive combos and exhaust items accumulated in stock. To execute this study, bibliographic surveys were realized, where the Fp-Growth algorithm was chosen to perform the processing of the information, given its effectiveness in executing large data sets. Of the thirteen products selected, promising associations were found for two of them. Altogether, seven association rules, for promotion and non-promotion periods, were found.

Keywords: Fp-Growth; Association Rules; Data Mining.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Problemas relacionados a estoque são frequentes desde pequenas as grandes empresas e, corriqueiramente, são utilizados como tema de estudos. Segundo Silva et al. (2018), estoques são reservas financeiras congeladas em forma de mercadorias, o qual podem danificar-se, deteriorar-se, além de gerar custos com locação, acomodação, climatização e iluminação. Além disso, mercadorias que permanecem armazenadas por logos períodos diminuem o capital de giro, uma vez que tais produtos ocupam espaço de produtos de maior giro.

A super estocagem de mercadorias é uma falha na cadeia de suprimentos. Segundo Slack et al. (2009), um item não necessitaria de estoque caso o fornecimento ocorresse conforme o planejado. O excesso de mercadorias, é proveniente de uma super aquisição (com intuito de reduzir custos), sob uma previsão otimista da área de vendas ou uma estratégia de redução de custos, onde a ruptura de um determinado material pode gerar um impacto negativo maior que o custo de armazenagem.

Nesse contexto, ações comerciais são realizadas com a finalidade de reduzir a super estocagem de mercadorias (quando já confirmada), entre elas: a fomentação de campanhas publicitárias regionalizadas, ações de fluxo (compre X reais e concorra a brindes), aplicação de desconto ou aproximar itens de menor e maior giro, que estão relacionados de alguma maneira, em locais próximos. Este último é intuitivamente mais complexo, uma vez que é possível gerar finitas relações, entre dois ou mais itens, exigindo um alto custo computacional para gerar e validar todos os possíveis agrupamentos. Além disso, nem todos os produtos comprados recorrentemente em um mesmo estabelecimento possuem correlação real não alavancando diretamente as vendas (LAVôR, 2003).

Para otimizar a forma como essas relações são encontradas, os autores Agrawal, Imielinski e Swami (1993) introduziram uma classe de métodos estatísticos na área de mineração de dados chamada de regras de associação, capazes de identificar padrões de venda entre diferentes produtos em problemas de cesta de compras. As relações encontradas nesta metodologia são associações $\{X \Rightarrow Y\}$ entre dois ou mais produtos estatisticamente mais interessantes. Segundo Veloso (2004), tais associações são valiosas, pois quando bem empregados, podem incrementar o número de itens vendidos em uma mesma transação (em supermercados, uma das possibilidades é aproximar os produtos relacionados nas prateleiras ou colocá-los em setores próximos).

Durante os anos, tais técnicas se mostraram bastante eficientes, sendo possível encontrar relações interessantes em estabelecimentos que detêm grandes quantidades de produtos. Um dos casos mais emblemáticos, da Wal-Mart, popularizou o uso desta aplicação (FELDENS; CITOLIN; FRIGERI, 1999), tornado-a indispensável em grandes redes varejistas para otimização das vendas.

O estabelecimento comercial estudado neste trabalho, faz parte de um grupo de lojas

de uma franqueadora que atua no ramo de produtos de beleza no estado de Goiás. O portfólio é composto por dezenas de milhares de itens, porém somente uma fração desses itens são promocionados a cada mês, uma vez que as promoções são alinhadas com a estratégia de comunicação proposta pela franqueadora. Em muitos meses, algumas campanhas acabam não tendo o fluxo esperado, gerando super estoque de determinadas mercadorias que, na maioria das vezes, não conseguem se extinguir sem alguma ação específica.

Não é possível reposicionar os itens nas prateleiras para favorecer um determinado produto, pois tal ação é de controle da franqueadora e envolvem questões de posicionamento de marca e de experiência do usuário. Como forma de contornar tal limitação, utilizou-se a abordagem de regras de associação para agregar itens potencialmente relacionados em combos presenteáveis com o intuito de alavancar suas vendas.

1.1 Objetivos

O foco deste trabalho é a identificação de associações de mercadorias de maior e menor fluxo através de métodos estatístico com o intuito de gerar sugestões de combos presenteáveis atrativos para diminuição de produtos super estocados.

1.1.1 Objetivo Geral

Gerar sugestões de combos presenteáveis, de dois ou mais itens, através de métodos estatísticos para aumentar o giro de produtos acumulados em estoque;

1.1.2 Objetivos Específicos

- Abordar as principais técnicas de Regras de Associação;
- Identificar materiais críticos no estabelecimento (super estocagem);
- Identificar regras promissoras entre mercadorias através do algoritmo Fp-Growth;

1.2 Limitações

Não foram encontradas outras obras referentes ao tema de identificação de combos presenteáveis utilizando essa técnica. Deste modo muitas das abordagens escolhidas são hipóteses. Outro fator é que as análises dessa loja não necessariamente se confirmarão em um outro estabelecimento ou se esses comportamentos se manterão constantes com o variar do tempo.

1.3 Estrutura do Trabalho

Nó próximo capítulo é apresentado uma contextualização sobre o que é mineração de dados, regras de associação, os principais algoritmos utilizados, algumas variações e exemplos de aplicação. No capítulo seguinte é abordado a contextualização do problema, os critérios

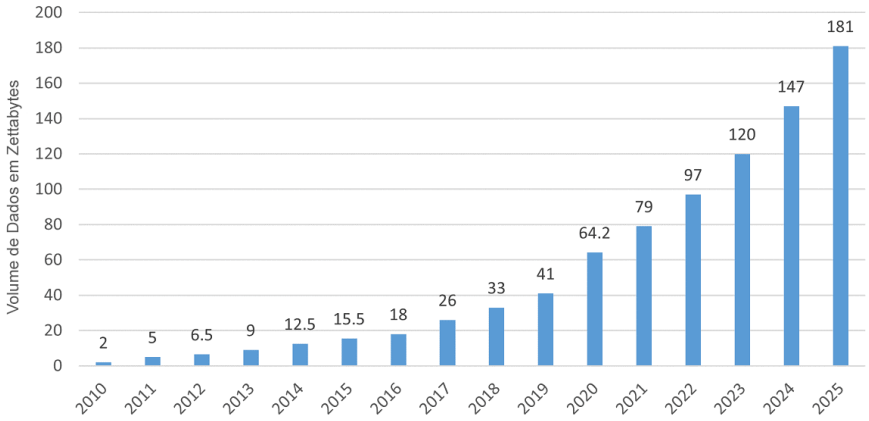
utilizados na modelagem do problema e os critérios de avaliação. Em seguida são apresentados os resultados encontrados, a conclusão e as referências utilizadas.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Mineração de Dados

Com a evolução da tecnologia, o volume de dados gerados vem crescendo significativamente durante os anos. A rede mundial de internet, softwares empresariais, programas relacionados a pesquisas e aplicativos de celulares são algumas das ferramentas que constantemente estão gerando informações. Conforme Reinsel, Rydning e Gantz (2021), a quantidade de dados criados e replicados entre os anos de 2021 até 2025 será maior que o dobro da quantidade de dados criados desde o advento do armazenamento digital (ocorrido em 1956 com o surgimento do primeiro computador com sistema de armazenamento em disco). Tal percepção, juntamente com o uso cada vez mais constante dos meios digitais, fez com que o tema de Mineração de Dados se tornasse cada vez mais indispensável na era da informação (GARCíA; LUENGO; HERRERA, 2015).

Figura 1 – Volume de Dados Criados e Replicados pelo Mundo



Fonte: Adaptado de Reinsel, Rydning e Gantz (2021)

Atualmente, o tema de Mineração de Dados (*data mining*) é aplicado para a descoberta de padrões. Fayyad, Piatetsky-Shapiro e Smyth (1996b) define o tema como um processo lógico, para descoberta de informações relevantes, potencialmente úteis e de fácil compreensão.

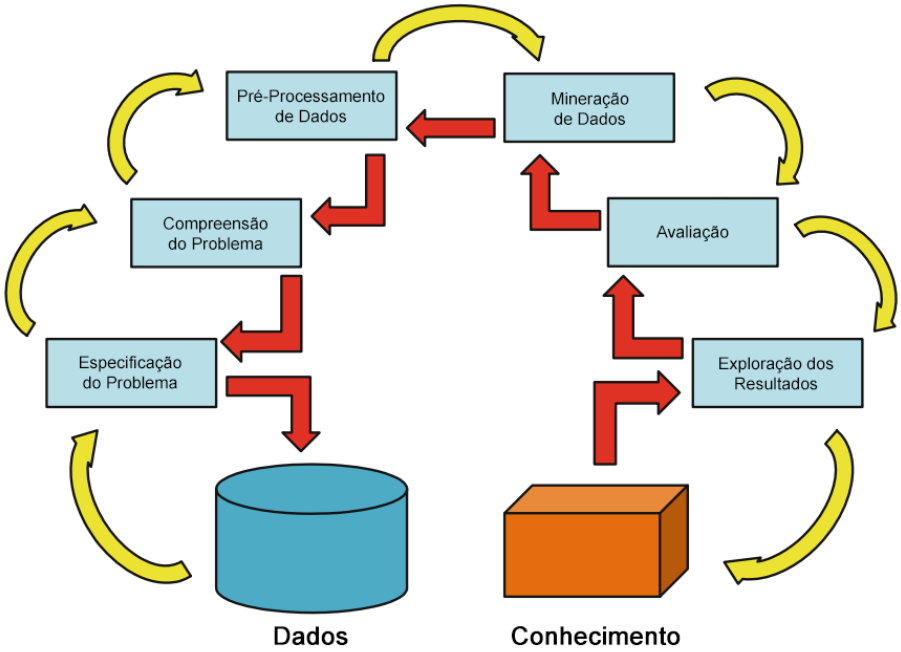
Apesar de não ser a única, mas talvez a mais importante, a etapa de Mineração de Dados faz parte de um grande processo denominado *Knowledge Discovery in Databases* (Descoberta de Conhecimento em Banco de Dados em português), que tem como objetivo explorar bases de dados para a descoberta de padrões relevantes (HAN; PEI; KAMBER, 2011). García, Luengo e Herrera (2015) define seis principais etapas para o processo de Descoberta de Conhecimento em Banco de Dados:

1. **Especificação do Problema:** designa o domínio do problema e os objetivos;

2. **Compreensão do Problema:** inclui a compreensão tanto dos dados selecionados quanto do conhecimento especializado associado para abranger todas as possíveis hipóteses;
3. **Pré-Processamento de Dados:** inclui operações para limpeza de dados (como tratamento da remoção de ruído e dados inconsistentes), integração de dados (onde várias fontes de dados podem ser combinadas em uma), transformação e redução de dados. Em grande parte dos problemas, essa etapa demanda a maior quantidade de tempo do processo de KDD (VICENTE; POLETTO, 2020);
4. **Mineração de Dados:** processo essencial onde os métodos são usados para extrair padrões relevantes. Esta etapa inclui a escolha da técnica de Mineração de Dados mais adequada (como classificação, regressão, agrupamento ou associação), a escolha do algoritmo e o ajuste dos hiperparâmetros visando alcançar o melhor desempenho do modelo;
5. **Avaliação:** estima e interpreta os padrões minerados com base nas hipóteses levantadas;
6. **Exploração dos Resultados:** A última etapa pode envolver o uso direto do conhecimento dos analistas, incorporar o resultado em outro sistema para processos posteriores ou simplesmente relatar o conhecimento descoberto por meio de ferramentas de visualização.

A Figura 2 ilustra o fluxo do processo de Descoberta de Conhecimento em Banco de Dados, onde cada bloco (em azul) representa uma etapa do processo e as setas em amarelo indicam o sentido. Cada etapa depende de sua antecessora, sendo necessário retomar para a etapa anterior (setas vermelhas) caso não seja encontrado o resultado esperado.

Figura 2 – Etapas de Descoberta de Conhecimento em Banco de Dados



Fonte: Adaptado de García, Luengo e Herrera (2015)

2.2 Técnicas de Mineração de Dados

Técnicas de *data mining* são procedimentos científicos capazes de extrair informações relevantes previamente desconhecidas dentro de um conjunto de dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996a). Cada técnica depende do contexto em que está inserido e do tipo de informação que se deseja alcançar. Algumas tem como objetivo retratar uma informação já existente dentro de um conjunto de dados (modelo descritivo), ou prever uma informação conforme o padrão previamente existente (modelo preditivo).

Inúmeras técnicas são conhecidas no contexto de mineração de dados, entretanto, podemos encaixá-las em em cinco grandes metodologias conforme mencionada por Lavôr (2003): detecção de sequências, regressão, classificação, clusterização e regras de associação.

2.2.1 Detecção de Sequências

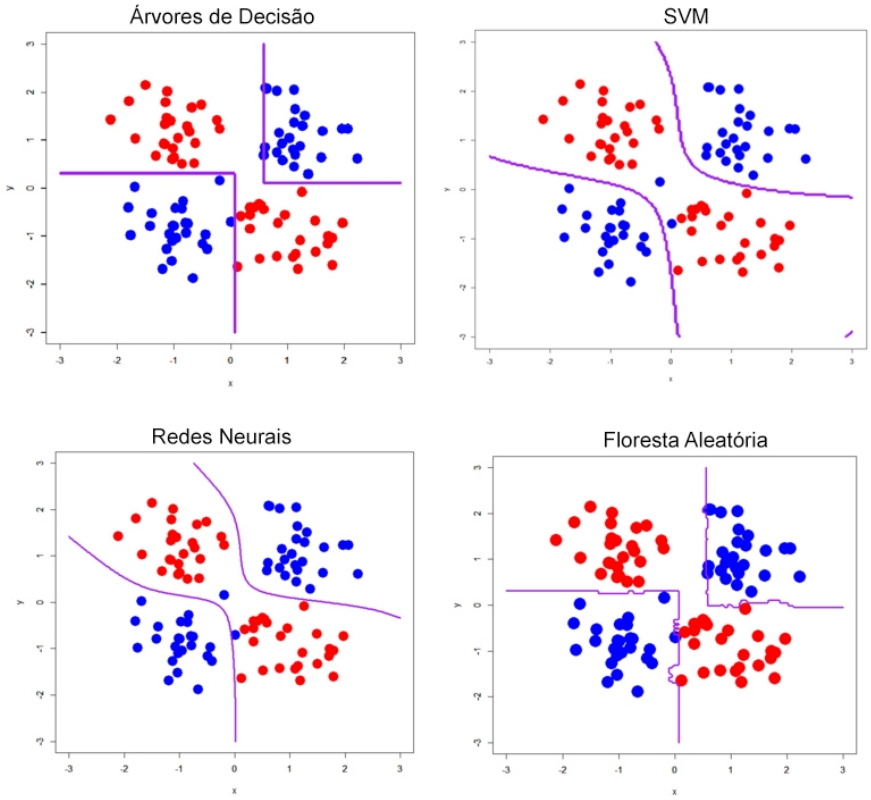
A abordagem de detecção de sequências consiste na localização de padrões sequenciais existentes entre itens dentro do banco de dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996b), onde a técnica indica a possibilidade de certos eventos ocorrerem sequencialmente ao longo de um período de tempo. Por exemplo, pode ser constatado que, após a venda de um celular, 30% dos consumidores abrirão uma ocorrência reclamando sobre o desempenho da bateria dentro do período de um ano. Outro uso interessante dessa aplicação é na área da medicina, onde o histórico do paciente pode indicar o surgimento de doenças futuras. Algumas das técnicas encontradas na literatura para a detecção de sequências são: Estatística, Teoria de Conjuntos e Máquina de Vetores de Suporte (*Support Vector Machine*) conforme Xing, Pei e Keogh (2010) e *Long short-term memory* conforme utilizado por Jurgovsky et al. (2018).

2.2.2 Classificação

Classificação é uma técnica que busca prever, ou associar, elementos pertencentes a um conjunto de dados dentro de uma ou mais classes pré-definidas. Os algoritmos buscam encontrar padrões implícitos nos atributos dos dados informados pelo especialista, onde, após o ajuste do modelo, os dados posteriormente inseridos são alocados nas classes que possuem os atributos mais similares aos padrões encontrados.

A Figura 3, por exemplo, mostra o comportamento de quatro diferentes algoritmos executados na mesma base de dados: Árvores de Decisão, SVM, Redes Neurais e Floresta Aleatória. Na imagem os pontos vermelhos e azuis representam observações de duas diferentes classes e em cada plano cartesiano há linhas limítrofes (em rosa) que diferenciam cada uma das classes de acordo com o resultado de cada algoritmo.

Figura 3 – Algoritmos de Classificação



Fonte: Adaptado de OZAKI (2015)

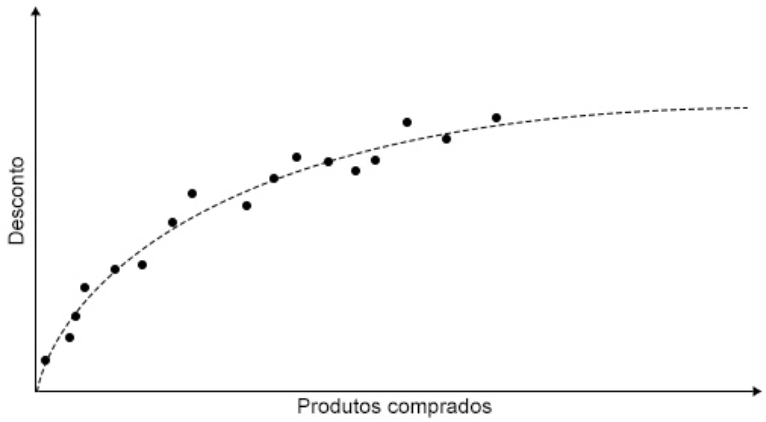
Atualmente há diversos algoritmos conhecidos na literatura que realizam esta tarefa. Verma, Pal e Kumar (2019) citam ao menos dez diferentes técnicas aplicadas ao tema de doenças de pele, entre as quais podemos citar os algoritmos KNN, Naive Bayes, AdaBoost e Redes Neurais Convolucionais (CNN).

2.2.3 Regressão

Técnicas de regressão tem como objetivo identificar padrões preditivos dentro de um conjunto de dados por meio da correlação entre variáveis. Para encontrar padrões, a técnica busca projetar uma função que retorne o valor numérico esperado dado um conjunto de parâmetros selecionados. A função é então utilizada para prever o valor a ser assumido por um novo item inserido no conjunto de dados (LAVôR, 2003).

Dentre as possíveis aplicações podemos citar a previsão de desconto de um determinado insumo conforme a quantidade adquirida, exemplo ilustrado na Figura 4. No gráfico, os pontos pretos representam, de forma genérica, a quantidade de insumos adquiridos versus o desconto dado entre diferentes lojas, e a curva tracejada representa a função não linear que busca estimar o desconto dado a partir de uma determinada quantidade de insumos.

Figura 4 – Exemplo de Regressão



Fonte: autoria própria

Regressões lineares e polinomiais são técnicas bem conhecidas para a tarefa de regressão (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996a). No entanto muitos dos algoritmos utilizados na classificação também possuem as sua adaptações para a função de regressão, como por exemplo KNN, Árvores de Decisão, AdaBoost e Redes Neurais (PEDREGOSA et al., 2011).

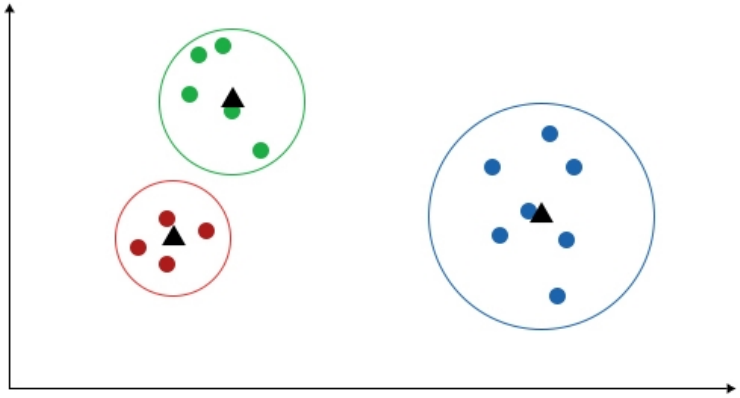
2.2.4 Clusterização

O processo de clusterização é semelhante a técnica de classificação, onde cada um dos elementos do banco de dados é classificado dentro de um grupo (*cluster*) no final do processo. Mas, diferente do modelo anterior, o processo não necessita de uma pré-classificação do usuário, sendo o algoritmo responsável pela identificação e agrupamento dos *clusters*, também chamado na literatura como aprendizado não supervisionado (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996b).

Nesta técnica, os *clusters* devem apresentar alta similaridade (apresentar um padrão próximo ou uma característica similar), mas ao mesmo tempo serem distintos de outros *clusters*. Segundo Lavôr (2003), toda clusterização é feita com objetivo de maximizar a homogeneidade dentro de cada agrupamento e maximizar a heterogeneidade entre *clusters*. Entre as principais técnicas podemos citar K-means, que tem como objetivo escolher centroides que minimizem a distância entre todos os pontos considerando a N quantidade de *clusters* requeridos Pedregosa et al. (2011).

A Figura 5 mostra um exemplo de clusterização utilizando a técnica K-means, onde cada circunferência representa um diferente *cluster* encontrado pelo algoritmo e os triângulos os centroides de cada agrupamento.

Figura 5 – Exemplo de Clusterização



Fonte: autoria própria

2.2.5 Regras de Associação

Regras de associações são métodos que indicam a possibilidade de dois ou mais eventos ocorrerem simultaneamente. Normalmente, a técnica é aplicada quando se deseja encontrar relações entre milhares de itens, por exemplo em um supermercado. De cordo com Feldens, Citolin e Frigeri (1999), um dos casos mais emblemáticos foi descoberto pela rede americana Wal-Mart, que identificou que 60% das mães que compravam bonecas Barbie também levavam uma barra de chocolate. Dada essa informação, a rede de supermercados reposicionou as gôndolas para que as seções de doces e brinquedos ficassem próximas, como resultado, as vendas de doces deram um salto. Como este é o tema de escopo do trabalho, o tópico de regras de associação é aprofundado com mais detalhes na seção seguinte.

2.3 Método de Regras de Associação

A metodologia de regras de associação, possibilita encontrar relações entre dois ou mais itens, de uma base de dados, banco de dados relacional ou demais repositórios (KUMBHARE; CHOBE, 2014). As combinações encontradas através das regras possuem o formato {X $\Rightarrow$ Y}, na qual a associação de exemplo {Pão $\Rightarrow$ Leite} nos indica que há uma chance de que o cliente escolherá leite (consequente da regra) considerando que ele adquiriu anteriormente pão (antecedente da regra). Tal princípio pode ser aplicado para desvendar padrões em processos de trabalho, problemas de cestas de compra e *loss-leader analysis* (quando um produto sem lucratividade é exposto para atrair clientes).

Agrawal, Imielinski e Swami (1993) definem o método de regras de associação no seguinte formato:

Seja $I = \{I_1, I_2, I_3, ..., I_m\}$ um conjunto de itens, denominado *itemset*, e seja T uma base de dados que contenha várias transações de *itemsets* I, temos que $I \subseteq T$. X e Y são conjuntos de itens específicos, tal que $X \subseteq T$ e $Y \subseteq T$. A regra de associação é uma implicação

de forma $\{X \Rightarrow Y\}$, onde $X \subset I$, $Y \subset I$ e $X \cap Y = \varnothing$.

Uma regra $\{X \Rightarrow Y\}$ vêm associada a um valor de suporte e confiança, onde a regra $\{X \Rightarrow Y\}$ tem o suporte s, se s% das transações em T contém $X \cup Y$ e é válida no conjunto de transações T, com grau de confiança c, se c% das transações em T que contenham X também contém Y.

O suporte (*support*) é a medida indica a quantidade de vezes (frequência) que a combinação $\{X \Rightarrow Y\}$ aparece em todas as transações T, de modo que:

$$Suporte(X \Rightarrow Y) = \frac{Frequência\ de\ X\ e\ Y}{T}, suporte : [0,1] \quad (1)$$

Já a confiança (*confidence*), estima a possível probabilidade de um cliente escolher Y considerando que ele comprou previamente X. A confiança de $\{X \Rightarrow Y\}$ é dada pela fórmula:

$$Confiança(X \Rightarrow Y) = \frac{Suporte(X \Rightarrow Y)}{Suporte(X)}, confiança : [0,1] \quad (2)$$

Para melhorar a compreensão, é utilizado um exemplo citado por Vasconcelos e Carvalho (2018): imagine que você é dono de um supermercado e procura entender o comportamento de compra de seus clientes. A Tabela 1 mostra o histórico de transações de seu supermercado em um determinado período de tempo. Na tabela, o valor 0 significa que o produto não foi comprado na determinada transação e 1 significa que foi comprado:

Tabela 1 – Exemplo de Regras de Associação

	Leite	Café	Cerveja	Pão	Manteiga	Arroz	Feijão
Transação 01	0	1	0	1	1	0	0
Transação 02	1	0	1	1	1	0	0
Transação 03	0	1	0	1	1	0	0
Transação 04	1	1	0	1	1	0	0
Transação 05	0	0	1	0	0	0	0
Transação 06	0	0	0	0	1	0	0
Transação 07	0	0	0	1	0	0	0
Transação 08	0	0	0	0	0	0	1
Transação 09	0	0	0	0	0	1	1
Transação 10	0	0	0	0	0	1	0

Fonte: Adaptado de Vasconcelos e Carvalho (2018)

De todos os itens adquiridos, você deseja compreender a relação de dependência existente entre dois produtos: café e pão. O primeiro passo, é calcular a frequência de compra desses itens (o que denominamos anteriormente de suporte). Para isso, verifica-se que a combinação $\{Café \Rightarrow Pão\}$ está presente apenas nas transações 1, 3 e 4, em seguida é calculado o número total de transações, em nosso problema este número é 10. Logo, a frequência (suporte) em que estes itens são escolhidos simultaneamente é de $3/10 = 0.3$. É

importante ressaltar que o suporte de {Café ⇒ Pão} e {Pão ⇒ Café} sempre será igual. Desse modo, o valor do suporte independe da ordem dos termos.

Dessa maneira, a confiança de {Café ⇒ Pão} pode ser calculada dividindo-se a frequência em que café e pão aparecem simultaneamente em uma transação (valor 3, já calculado anteriormente), dividido pela quantidade de vezes (suporte) em que café foi comprado, café apareceu como opção nas transações 1, 3 e 4, ou seja, três vezes. Utilizando a fórmula temos que a confiança de {Café ⇒ Pão} é 3/3 ou 1. Como resultado, temos que 100% das vezes que os clientes compraram café também levaram pão e que essa combinação apareceu em 30% das compras realizadas.

Diferente do suporte, a confiança de {Café ⇒ Pão} pode não ter a mesma probabilidade da confiança de {Pão ⇒ Café}. No nosso exemplo, a confiança de {Pão ⇒ Café} é de 3/5, ou 60%. Outro fator que deve ser levado em conta é que em uma situação real o problema se torna muito mais custoso, uma vez que existirão milhares de transações e itens que podem ser associados. Este custo é uma das principais limitações do problema é melhor abordado nos tópicos seguintes. Entretanto, várias técnicas podem ser utilizadas para minimizar o número de operações, por exemplo, restringir o universo a ser analisado ou agrupar os itens em itens mais genéricos (denominar o achocolatado A e o achocolatado B como achocolatado), conforme Borgelt e Kruse (2002).

2.4 Método de Geração de Regras de Associação

Segundo Assunção (2011), uma abordagem possível para minerar regras de associação é através de força-bruta, na qual, o algoritmo busca todas as combinações de *N* itens. Pang-Ning, Steinbach e Kumar (2006), em seu trabalho, formulou a equação matemática que define a quantidade possível de combinações *R* dentro de um conjunto de dados, tal que seja possível encontrar *d - 1* itens no antecedente e consequente:

$$R = \sum_{k=1}^{d-1} \left[\left(\frac{d}{k} \right) \times \sum_{j=1}^{d-k} \left(\frac{d-k}{j} \right) \right] \tag{3}$$

$$R = 3^d - 2^{d+1} + 1 \tag{4}$$

No entanto, procurar todas as *N* combinações através de força bruta todas se torna impraticável quando há grandes conjuntos de dados. Desse modo, muitos dos algoritmos, inclusive aqueles abordados neste trabalho, dividem a estratégia de produzir as regras de associação em duas subtarefas para otimizar o processamento:

- Gerar os *Itemsets* Candidatos: etapa onde são gerados os candidatos que obedecem o suporte mínimo informado;
- Criar as Associações: gera as regras de associação conforme a liste de *itemsets* frequentes do passo anterior.

A etapa de geração dos *itemsets* candidatos (também chamado de associações candidatas ou combinações frequentes) é a mais custosa e, por esse motivo, a mais estudada dentro da literatura (HAN; PEI; YIN, 2000). Para trazer maior profundidade sobre o tema, são apresentados os principais algoritmos utilizados: Apriori e o Fp-Growth.

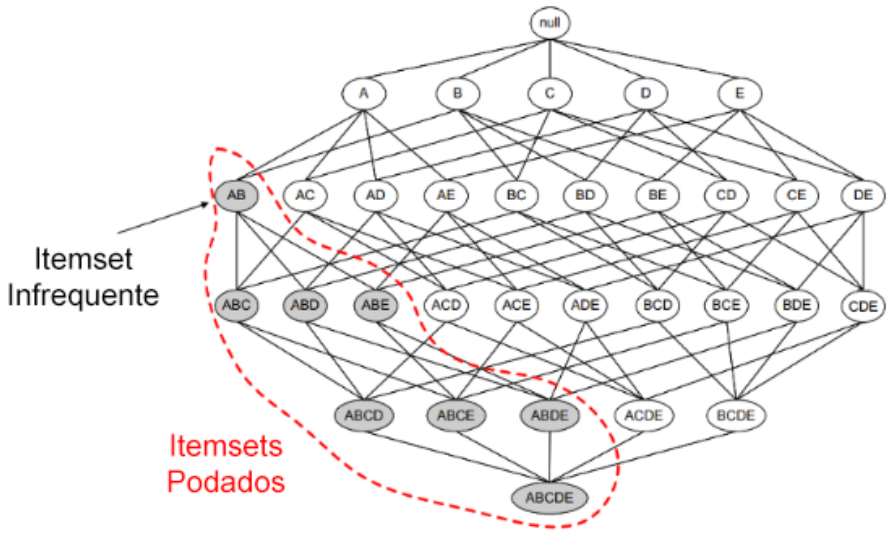
2.4.1 Apriori

De acordo com Agrawal, Imielinski e Swami (1993), o algoritmo Apriori foi um precursor para a disseminação e crescimento dos estudos de regras de associação. Esse algoritmo é comumente utilizado como porta de entrada para exemplificação e ensinamento sobre o tema.

No Apriori, um conjunto de dados com *k* itens pode gerar cerca de $2^k + 1$ *itemsets* candidatos. Para localizar as *N* possíveis combinações do problema, é necessário, primeiramente, calcular todos os possíveis *itemsets* e filtrar aqueles que estejam abaixo do suporte mínimo esperado. Tal abordagem é altamente custosa, onde, o algoritmo Apriori explora o uso do suporte como limitador na etapa de geração de *itemsets* candidatos.

Agrawal, Imielinski e Swami (1993) identificou que, se um *itemset* for infrequente (abaixo do suporte mínimo), todos os conjuntos que contenham aquele *itemset* também não serão frequentes. Tal propriedade é importante uma vez que reduz o custo computacional para se gerar as associações candidatas no espaço de busca. Na Figura 6 está exemplificado o processo de poda, onde o conjunto AB é infrequente e todos os subconjuntos derivados dele também serão:

Figura 6 – Exemplificação da Poda do Algoritmo Apriori



Fonte: Adaptado de ICHI.PRO (2020)

O algoritmo Apriori pode ser executado conforme o pseudocódigo da Figura 7 (LAVôR, 2003):

Figura 7 – Pseudocódigo do Algoritmo Apriori

1) $F_1 = \{1\text{-itemset}\}$
2) Para cada $k = 2; L_{K-1} \neq 0; (passo = 1)$ faça
3) $C_k = Novos\_Candidatos\ (F_{k-1})$
4) Para todas as transações $t \in D$ faça
5) C_t = subconjunto (C_k, t); // Candidatos contidos em t
6) Para todos os candidatos $c \in C_t$
7) $c.suporte = c.suporte + 1$
8) $F_K = \{c \in C_k \mid c.suporte \geq suporte\_mínimo\}$
9) $F = F \cup F_k$

Fonte: Lavôr (2003)

Considere *Fk* a lista de *itemsets* frequentes com *k* itens e *F* a lista de todos os *itemsets* frequentes. Primeiramente, encontra-se o conjunto *F1* que é a lista de conjuntos de itens frequentes com apenas um elemento.

Na sequência, em cada passo *k* do algoritmo o elemento *Ck* é gerado utilizando a lista de *Fk-1*, o qual é a lista *itemsets* frequentes com *k-1* elementos. Nesta etapa, a base de dados é percorrida para calcular o suporte de cada candidato. Os candidatos com suporte igual ou superior ao suporte mínimo são incluídos na lista de *itemsets* frequentes.

A função Novo_Candidatos do algoritmo Apriori gera a lista de candidatos utilizando apenas os *itemsets* frequentes do passo anterior, dessa forma, o Apriori evita calcular combinações candidatas que contenham algum subconjunto que não seja frequente.

Para exemplificar o funcionamento do Apriori, é utilizado a base de dados genérica localizada na Tabela 2:

Tabela 2 – Base de Dados

Ik	Itens
I1	A, C, D, E
I2	A, B, C, D, F
I3	A, D, E
I4	D
I5	A, B
I6	A, B, D, E

Fonte: autoria própria

Na primeira etapa, o algoritmo Apriori considera todos os itens como candidatos. Para calcular o suporte, é realizada uma passagem pelo banco de dados:

Tabela 3 – Primeira Etapa Apriori

Item	Suporte
A	0.83
B	0.5
C	0.33
D	0.83
E	0.5
F	0.16

Fonte: autoria própria

Na segunda etapa, são filtrados os itens que possuem o suporte menor que o suporte mínimo escolhido, nesse exemplo foi definido o suporte mínimo como 0.5 de forma arbitrária:

Tabela 4 – Segunda Etapa Apriori

Item	Suporte
A	0.83
B	0.5
D	0.83
E	0.5

Fonte: autoria própria

Nesta terceira etapa, é criada uma nova tabela com a combinação dos itens da Tabela 4. Os valores de suporte das combinações candidatas são calculadas através de uma nova passagem no banco de dados:

Tabela 5 – Terceira Etapa Apriori

Item	Suporte
A, B	0.5
A, D	0.67
A, E	0.5
B, D	0.33
B, E	0.16
D, E	0.5

Fonte: autoria própria

Assim como na segunda etapa, são filtradas as combinações que possuem suporte inferior ao suporte mínimo:

Tabela 6 – Quarta Etapa Apriori

Item	Suporte
A, B	0.5
A, D	0.67
A, E	0.5
D, E	0.5

Fonte: autoria própria

No quinto passo, as combinações da etapa anterior são agrupadas e uma nova passagem pela base é realizada, gerando a nova lista de possíveis candidatos. Porém, diferentemente das etapas anteriores, somente os itens {A, D, E} geram um agrupamento válido.

Tabela 7 – Quinta Etapa Apriori

Item	Suporte
A, D, E	0.5

Fonte: autoria própria

A combinação {A, B, D} não é válida, uma vez que é composta pelo agrupamento {B, D}, que é considerado infrequente (menor que o suporte mínimo). Esta mesma regra impede que as combinações {A, B, E} e {B, D, E} sejam consideradas válidas, pois são compostas respectivamente pelos agrupamentos {B, E} e {B, D}, que são infrequentes.

Uma vez que o suporte mínimo da regra {A, D, E} é frequente, e que não é mais possível criar novos agrupamentos, o algoritmo gera a lista de *itemsets* frequentes com todos os agrupamentos. A Tabela 8 mostra o resultado de todas as regras encontradas que respeitem o suporte mínimo do problema.

Tabela 8 – Conjunto de Itemsets Frequentes

Item	Suporte
A	0.83
D	0.83
A, D	0.67
B	0.5
E	0.5
A, B	0.5
A, E	0.5
D, E	0.5
A, D, E	0.5

Fonte: autoria própria

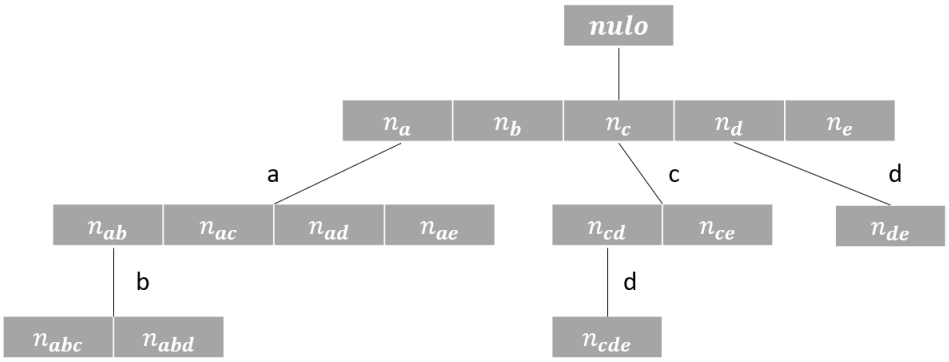
O algoritmo Apriori, realiza o cálculo de suporte, filtragem e agrupamento dos candidatos tantas vezes quanto forem necessárias até que não seja possível mais nenhuma combinação.

No entanto, tal abordagem não possui boa performance em bases transacionais extensas (KUMBHARE; CHOBE, 2014). Dessa forma, outros algoritmos surgiram para contornar tal limitação.

2.4.2 Fp-Growth

O Fp-Growth foi proposto no ano de 2004 como uma alternativa ao popular Apriori (HAN et al., 2004). O algoritmo gera a lista de *itemsets* frequentes sem a necessidade de percorrer múltiplas vezes a base de dados. Segundo Raschka (2020), tal característica é possível, pois o algoritmo dimensiona a lista de possíveis candidatos em uma estrutura compacta chamada Fp-Tree (árvore de padrão frequente). Desse modo, a busca se torna mais eficiente, tornando-o ideal para performar em grandes conjuntos de dados (KUMBHARE; CHOBE, 2014). A Figura 8 retrata a construção de uma Fp-Tree genérica.

Figura 8 – Exemplo de Árvore Criada no Algoritmo Fp-Growth



Fonte: Adaptado de Borgelt e Kruse (2002)

O Fp-Growth possui três etapas que são necessárias para a identificação dos *itemsets* frequêntes:

1. Preparação dos Dados: os elementos são ordenados conforme o valor de suporte, os elementos abaixo do suporte mínimo são removidos;
2. Construção da Fp-Tree: os elementos da etapa anterior são consolidados, de forma lógica, em uma estrutura de árvore, o qual permite uma maior compactação da estrutura. Dessa maneira os elementos são acessados de forma eficiente otimizando o tempo das passagens do algoritmo;
3. Identificação das Combinações: após a construção da Fp-Tree, a árvore é percorrida buscando-se as combinações válidas;

Para exemplificação de cada uma das etapas, utilizaremos novamente a base de dados transacional da Tabela 2. O primeiro passo da etapa de preparação dos dados é ranquear os elementos conforme o valor de seu suporte (ilustrado na Tabela 9).

Tabela 9 – Primeiro Passo da Etapa de Preparação dos Dados

Item	Suporte
A	0.83
D	0.83
B	0.5
E	0.5
C	0.33
F	0.16

Fonte: autoria própria

Os itens abaixo do suporte mínimo (definimos anteriormente um suporte mínimo de 0.5) são descartados:

Tabela 10 – Segundo Passo da Etapa de Preparação dos Dados

Item	Suporte
A	0.83
D	0.83
B	0.5
E	0.5

Fonte: autoria própria

Em seguida, reordenamos os itens da base de dados conforme a ordem da etapa anterior e os itens infrequentes são removidos durante a etapa de reordenamento:

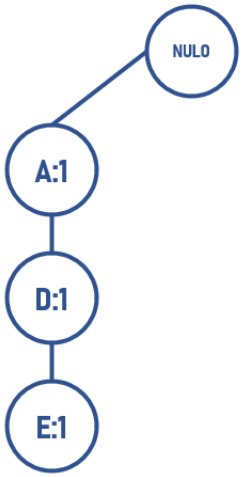
Tabela 11 – Terceiro Passo da Etapa de Preparação dos Dados

Ik	Itens	Itens Arranjados
I1	A, C, D, E	A, D, E
I2	A, B, C, D, F	A, D, B
I3	A, D, E	A, D, E
I4	D	D
I5	A, B	A, B
I6	A, B, D, E	A, D, B, E

Fonte: autoria própria

Finalizada a etapa de 'Preparação dos Dados', começamos a construir a Fp-Tree conforme a ordem dos *itemsets* e dos itens rearranjados anteriormente. Primeiramente é criado um nó nulo que será a raiz da nossa árvore, e em seguida, os itens do primeiro *itemset* {A, D, E} serão adicionados em uma rede de nós conforme o seu aparecimento (conforme ilustrado na Figura 9). Dentro de cada nó, é inserido o valor correspondente ao número de vezes que o item apareceu dentro daquele *itemset*.

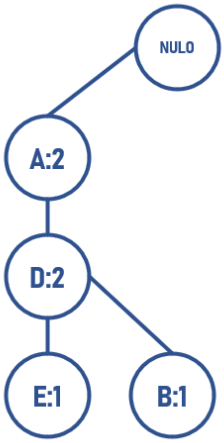
Figura 9 – Primeiro Passo da Construção da Fp-Tree



Fonte: autoria própria

O segundo *itemset* possui os itens {A, D, B}, perceba que os itens {A, D} já foram adicionados na nossa Fp-Tree. Desse modo, como os dois primeiros itens possuem a mesma ordenação, o caminho já construído pela árvore será considerando, adicionando apenas o item {B} como uma ramificação do nó {D} e acrescentando um ao valor dos nós já criados:

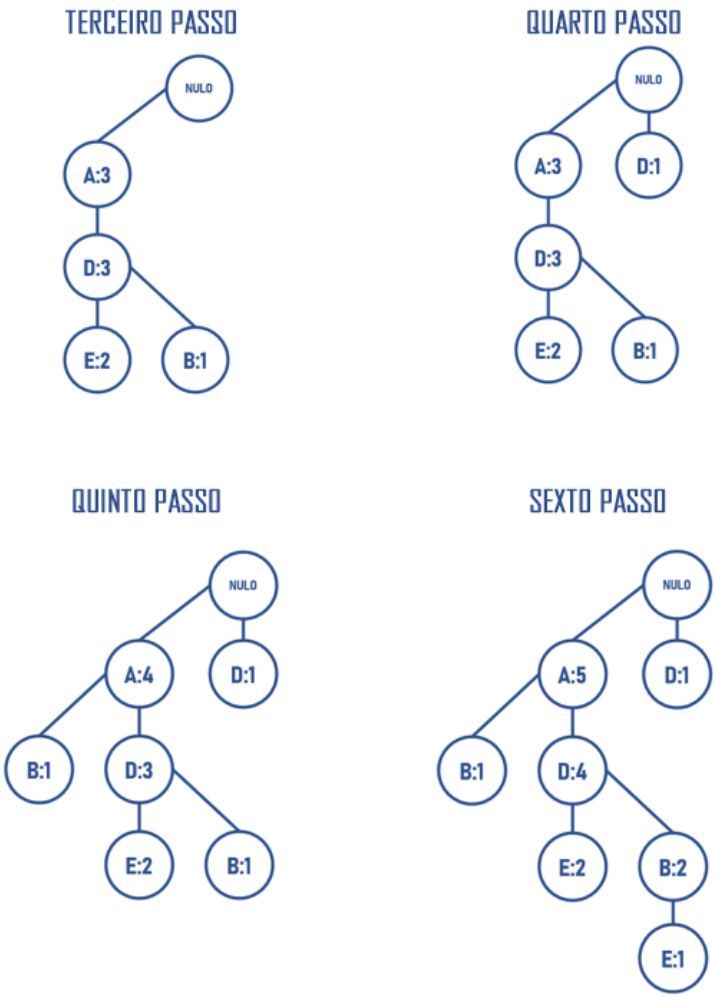
Figura 10 – Segundo Passo da Construção da Fp-Tree



Fonte: autoria própria

Os *itemsets* 3, 4, 5 e 6 seguem a mesma metodologia: caso a ordem dos itens respeite um caminho já inserido na Fp-Tree, este deverá ser priorizado adicionando o valor 1 aos nós percorridos. Caso encontre alguma divergência, é feita a ramificação com o valor inicial 1 nos novos nós. As etapas das inserções são mostradas nab Figura 11.

Figura 11 – Passos 3, 4, 5 e 6 da Construção da Fp-Tree



Fonte: autoria própria

Após a construção da Fp-Tree, inicia-se a abordagem de 'Identificação das Combinações' frequentes. Para isso, utiliza-se da Fp-Tree gerada nos passos anteriores para buscar os caminhos terminados com as letras {E, B, D, A}. Note que esta ordenação é a inversa da lista de itens frequentes gerados na segunda etapa do algoritmo. Tal metodologia é utilizada uma vez que os itens menos frequentes tendem a aparecer com mais frequência nos nós folha da Fp-Tree.

Dentre os caminhos que devemos avaliar, começaremos com os caminhos terminados com a letra {E}. Nesse caso, com base na Figura 11, note que existe dois caminhos {A, D, E} e {A, D, B, E}, estes caminhos terão os respectivos pesos (ou frequências) de valor 2 e 1, uma vez que o número associado ao nó {E} nesses caminhos possui tais valores.

Tabela 12 – Primeiro Passo de Identificação das Combinações

Termina com	Caminhos
E	ADE:2, ADBE:1

Fonte: autoria própria

Em seguida, deve-se encontrar os pesos pertencentes a cada uma das letras dos caminhos encontrados anteriormente. Para isso, é somado os pesos dos caminhos em que cada letra aparece, por exemplo: o item {A} aparece em ambos os caminhos, então basta somar o valor dos dois caminhos para dar o peso de $\{A\} = 2 + 1 = 3$. Por sua vez, o item {B} aparece apenas no segundo caminho {A, D, B, E} de peso 1, logo este será o peso deste item. Isso deverá ser feito com os demais itens.

Tabela 13 – Segundo Passo de Identificação das Combinações

Termina com	Caminhos	Quantidade de cada item
E	ADE:2, ADBE:1	A:3, D:3, B:1, E:3

Fonte: autoria própria

Com os respectivos pesos, é necessário remover os itens com o peso inferior ao peso mínimo (ou quantidade mínima) do problema. Para encontrar o peso mínimo, basta multiplicar o suporte mínimo pelo número de *itemsets* existentes.

$$\textit{Peso mínimo} = \textit{Suporte Mínimo} * \textit{Qtde Itemsets} \tag{5}$$

Perceba que, na Tabela 13, o item B está marcado com a cor vermelha, pois este item não possui o peso mínimo exigido ($0.5 * 6 = 3$). Tal propriedade é idêntica ao Apriori, uma vez que é provado que um item não frequente gera apenas combinações infrequentes. Em seguida, todas as combinações existentes entre as letras restantes são listadas com suas respectivas frequências (com exceção do próprio item que, quando respeitado a frequência mínima, também é adicionado a essa lista).

Tabela 14 – Terceiro Passo de Identificação das Combinações

Termina com	Caminhos	Quantidade de cada item	Combinações candidatas
E	ADE:2, ADBE:1	A:3, D:3, B:1, E:3	AE:3, DE:3, ADE: 3, E:3

Fonte: autoria própria

Com a possível lista de combinações candidatas, é avaliado novamente se é respeitada a frequência mínima do problema. Se sim, tais combinações são consideradas como *itemsets* frequentes. Para os caminhos terminados com a letra {E}, nenhuma combinação candidata se confirmou infrequente.

Tabela 15 – Quarto Passo de Identificação das Combinações

Termina com	Caminhos	Quantidade de cada item	Combinações candidatas	Itemsets frequentes
E	ADE:2, ADBE:1	A:3, D:3, B:1, E:3	AE:3, DE:3, ADE: 3, E:3	AE:3, DE:3, ADE: 3, E:3

Fonte: autoria própria

O mesmo processo realizado com os caminhos terminados com a {E} deve ser realizado com as letras {B, D, A}. O resultado dessas iterações é apresentada na Tabela 16.

Tabela 16 – Resultado da Identificação das Combinações

Termina com	Caminhos	Quantidade de cada item	Combinações candidatas	Itemsets frequentes
E	ADE:2, ADBE:1	A:3, D:3, B:1, E:3	AE:3, DE:3, ADE: 3, E:3	AE:3, DE:3, ADE: 3, E:3
B	AB:1, ADB:2	A:3, D:2, B:3	AB:3, B:3	AB:3, B:3
D	AD:4, D:1	A:4, D:5	AD:4, D:5	AD: 4, D:5
A	A:5	A:5	A:5	A:5

Fonte: autoria própria

Uma vez encontrada a lista de *itemsets* frequentes, podemos aplicar os pesos das associações encontradas na Equação (5) para termos o mesmo resultado encontrado pelo Apriori na Tabela 8. Neste exemplo, quanto menor for o suporte mínimo especificado, maiores são as chances de combinações candidatas serem consideradas frequentes e, consequentemente, maiores os custos computacionais envolvidos. O algoritmo Fp-Growth, por percorrer uma menor quantidade de vezes o conjunto de dados, reduz essa complexidade computacional (LAVôR, 2003). Com base nessa característica, o Fp-Growth é utilizado como algoritmo gerador de *itemsets* candidatos neste trabalho.

2.5 Geração e Interpretação das Regras de Associação

Conforme mencionado anteriormente, a maioria dos algoritmos utiliza a lista de *itemsets* frequentes, gerada no tópico anterior, para gerar as regras de associações, cujos resultados possuem associação direta ao suporte mínimo que é indicado pelo usuário. Porém, essa metodologia gera algumas dificuldades conforme mencionado por Webb e Zhang (2005):

- Se o suporte mínimo for muito alto, podem ser capturados itens com frequências elevadas que não possuem associação direta, por exemplo, arroz e detergente. Também é preciso levar em conta que grandes bases de dados (como supermercados) possuem produtos com frequências baixas por possuírem marcas distintas de um mesmo item;
- Caso o suporte mínimo seja muito baixo, o processo de geração de *itemsets* frequentes pode ser impraticável dependendo da base de dados e período analisado.

Normalmente uma quantidade excessiva de regras são geradas quando o suporte é baixo. Porém, segundo Assunção (2011), apenas algumas regras são provavelmente interessantes para o especialista e isso se deve ao fato de que muitas relações são óbvias, ou seja, tal padrão já é conhecido, como a combinação {Pão ⇒ Manteiga}. Entretanto, a correlação {Fralda ⇒ Cerveja} pode ser considerada importante caso seja confirmada sua efetividade.

Uma regra é dita interessante, uma vez que nos trás uma informação útil. Na literatura, existem medidas com o objetivo de avaliar o interesse ou a efetividade de uma determinada regra, como por exemplo o *lift* (BRIN et al., 1997). Elas podem ser utilizadas como tomada de decisão sobre quais regras devem ser mantidas e, ao mesmo tempo, reduzir o tempo de análise do especialista filtrando as associações indesejadas (LAVôR, 2003).

A medida de sustentação (*lift*) nos indica se as regras selecionadas ocorrem de maneira aleatória, onde Brin et al. (1997) definine o *lift* de {X ⇒ Y} como:

$$lift(X \Rightarrow Y) = \frac{Suporte(X \Rightarrow Y)}{Suporte(X)\ x\ Suporte(Y)} = \frac{Confiança(X \Rightarrow Y)}{Suporte(Y)}, lift : [0, \infty] \qquad (6)$$

A interpretação do *lift* é dada por:

- $Lift = 1$, indica que a relação não possui correlação direta e que ocorre por acaso;
- $Lift < 1$, indica que a relação é negativa e que a combinação ocorre com frequência menor do que selecionada ao acaso;
- $Lift > 1$, indica que a relação é positiva e que a combinação ocorre com maior frequência do que selecionada ao acaso;

Para ilustrar a utilização do *lift*, prosseguiremos com a construção das regras de associação da Tabela 2. Após a identificação dos *itemsets* frequentes do Fp-Growth, é preciso distinguir quais combinações possuem relação de alavancagem. A primeira etapa é aplicar a medida de confiança mencionada na Equação (2), o qual mede a possibilidade, considerando o histórico dos dados, de um cliente comprar o item Y uma vez que o item X foi escolhido anteriormente. Assim como o suporte, um valor mínimo de confiança é exigido para reduzir o universo de possíveis regras. Para esse problema, utilizaremos arbitrariamente uma confiança mínima de 70%, onde, na Tabela 17, já foram removidas as regras que estão abaixo do valor mínimo estipulado.

Tabela 17 – Regras de Associação do Problema

Regra	Antecedente	Consequente	Suporte	Confiança
{A ⇒ D}	A	D	0.67	0.8
{D ⇒ A}	D	A	0.67	0.8
{B ⇒ A}	B	A	0.5	1
{E ⇒ A}	E	A	0.5	1
{E ⇒ D}	E	D	0.5	1
{D, E ⇒ A}	D, E	A	0.5	1
{A, E ⇒ D}	A, E	D	0.5	1
{E ⇒ A, D}	E	A, D	0.5	1
{A, D ⇒ E}	A, D	E	0.5	0.75

Fonte: autoria própria

Em uma primeira visão, todas as regras encontradas na Tabela 17 parecem interessantes. No entanto, quando aplicamos a medida de *lift*, conforme ilustrado na Tabela 18, percebemos que as regras {A ⇒ D} e {D ⇒ A} não são atrativas, pois o *lift* indica uma associação que ocorre em uma menor frequência que a esperada ao acaso. Vale notar que, assim como o suporte, o *lift* não possui sentido. Pois tanto o numerador como o denominador da Equação (6) serão os mesmos em {X ⇒ Y} como em {Y ⇒ X}. Desse modo as combinações de {A, D ⇒ E} e {E ⇒ A, D} possuem os mesmos valores.

Tabela 18 – Resultado Final das Regras de Associação

Regra	Antecedente	Consequente	Suporte	Confiança	Lift
{A ⇒ D}	A	D	0.67	0.8	0.96
{D ⇒ A}	D	A	0.67	0.8	0.96
{B ⇒ A}	B	A	0.5	1	1.2
{E ⇒ A}	E	A	0.5	1	1.2
{E ⇒ D}	E	D	0.5	1	1.2
{D, E ⇒ A}	D, E	A	0.5	1	1.2
{A, E ⇒ D}	A, E	D	0.5	1	1.2
{E ⇒ A, D}	E	A, D	0.5	1	1.5
{A, D ⇒ E}	A, D	E	0.5	0.75	1.5

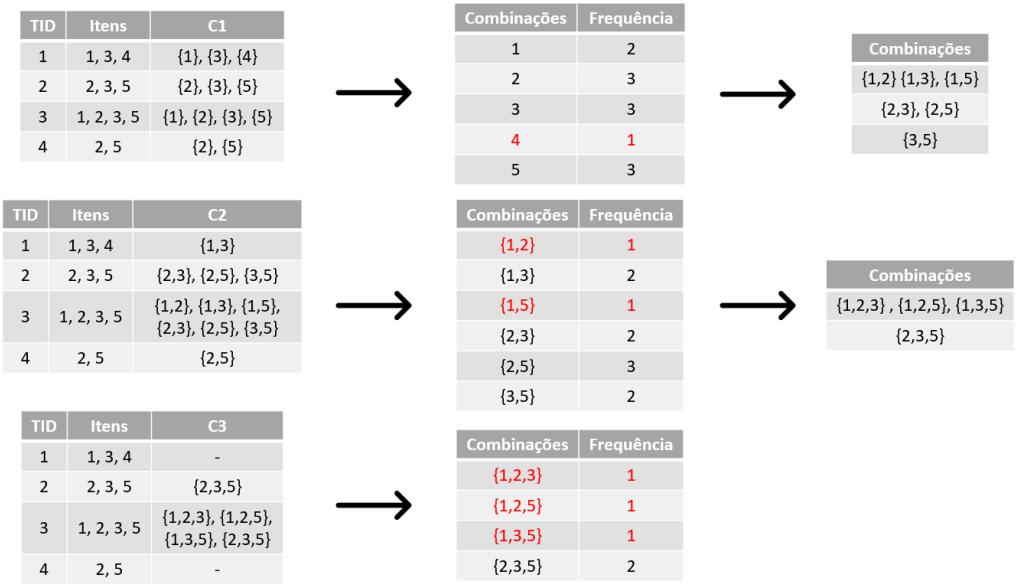
Fonte: autoria própria

2.6 Trabalhos Relacionados

Conforme visto anteriormente, os algoritmos trabalham percorrendo múltiplas vezes a base de dados para encontrar as combinações frequentes. Essa estratégia possui um alto custo computacional, fazendo com que o algoritmo necessite de uma grande quantidade de tempo para executar bases de dados extensas e/ou com múltiplos atributos. Dessa forma, variações das técnicas anteriores foram propostas para melhorar suas performances.

O AprioriTid (AGRAWAL; SRIKANT et al., 1994) utiliza a mesma abordagem do Apriori na primeira passagem pelo banco de dados. A partir de então, é gerado dentro de cada transação um conjunto denominado *Ck*, que guarda as combinações frequentes encontradas nesta passagem. As passagens posteriores ocorrem dentro da *Ck* do passo anterior gerando uma nova lista de combinações frequentes. Em cada passagem dentro da *Ck*, a quantidade de itens dentro de cada combinação é acrescida em um, onde o algoritmo termina quando não é mais possível gerar combinações que não atingem o suporte mínimo. A Figura 12, ilustra o funcionamento do AprioriTid para um conjunto de dados qualquer, onde a frequência mínima possui o valor 2 (ou suporte de 50%).

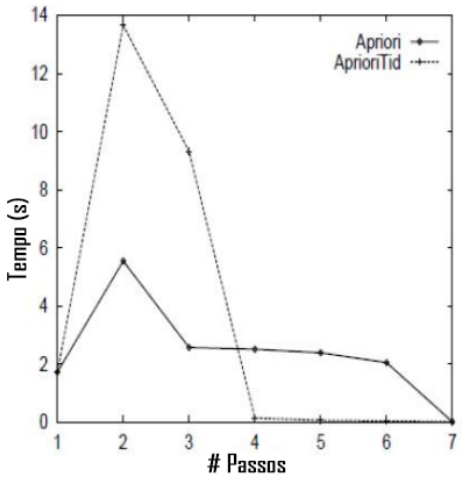
Figura 12 – Funcionamento do Algoritmo AprioriTid



Fonte: autoria própria

Percebe-se que, quanto mais passagens são realizadas, maior a quantidade de itens dentro das combinações candidatas. Isso gera nos passos posteriores uma quantidade menor de combinações candidatas e consequentemente um menor tempo de execução. Porém, tal adaptação se mostra inferior ao Apriori nos primeiros passos da execução. A Figura 13 ilustra o tempo necessário de execução durante cada iteração do Apriori e AprioriTid. Nota-se que no início o Apriori performa de maneira mais eficiente. Porém, no decorrer das iterações, o AprioriTid performa melhor.

Figura 13 – Tempo de Execução por Passos do Apriori e AprioriTid (suporte mínimo = 0.75)



Fonte: Adaptado de Khurana e Sharma (2013)

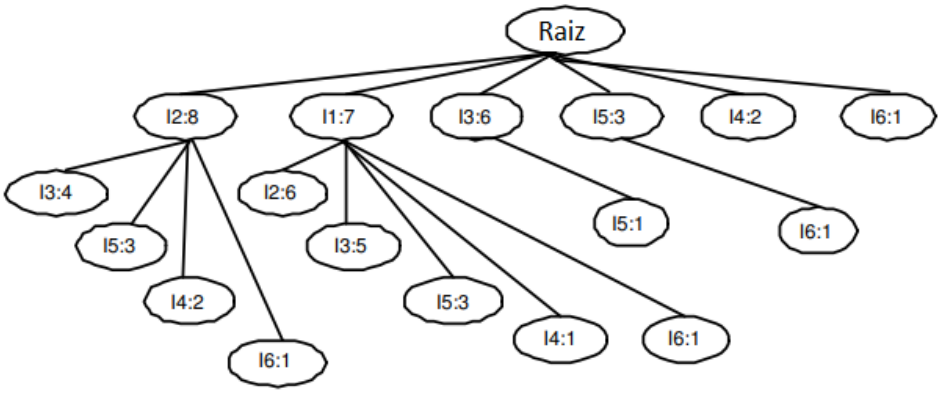
Como forma de contornar esse problema Agrawal, Srikant et al. (1994) sugeriram o algoritmo Apriori Hybrid, o qual utiliza a abordagem de identificação de combinações candidatas do Apriori nos passos iniciais e em seguida utiliza a abordagem do AprioriTid nos passos posteriores.

Os próximos algoritmos: Rapid Association Rule Mining (RARM) e Fp-Max, diferente do AprioriTid e Apriori Hybrid, são métodos especializados. Eles são adaptações criadas, a partir de métodos existentes, para minerar de maneira mais eficiente observações específica ou para maximizar um determinado comportamento.

O algoritmo RARM, por exemplo, possui uma disposição similar ao Fp-Growth, onde os dados são estruturados no formato de árvore denominada SOTrieIT. Porém, diferente da tecnica original, a estrutura SOTrieIT possui a altura máxima de ordem 3 sendo a raiz o nó principal. Essa estrutura performa de maneira mais eficiente, uma vez que sua aplicação é focada na localização de *itemsets* com apenas um e dois itens (DAS; NG; WOON, 2001).

Assim como a técnica original, o algoritmo percorre a base de dados localizando *itemsets* ainda não inseridos na SOTrieIT e as adicionando em novos nós/caminhos com o valor de suporte um. Caso a ordem encontrada já existe na SOTrieIT, é acrescido um valor de suporte ao nó/caminho presente. A Figura 14 ilustra o resultado da execução da estrutura SOTrieIT genérico.

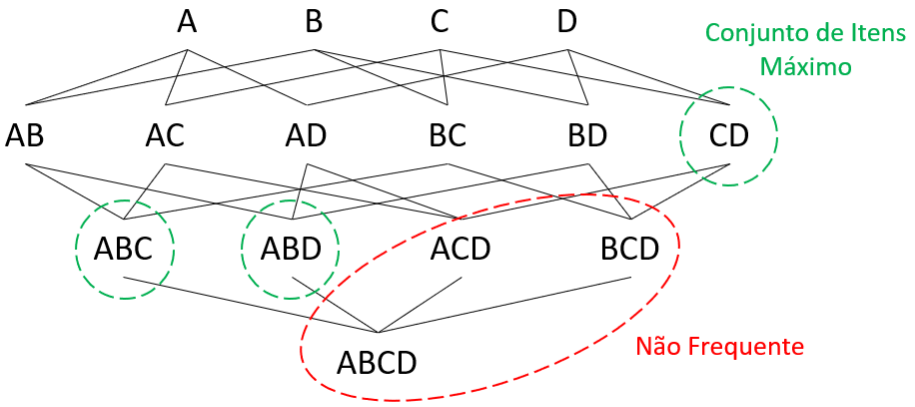
Figura 14 – Resultado da SOTrieIT



Fonte: Adaptado de Zhao e Bhowmick (2003)

O algoritmo Fp-Max também é uma variante do Fp-Growth, porém este algoritmo se concentra na obtenção de conjuntos de itens máximos dentro de um conjunto de dados. Um conjunto de itens X é dito máximo se X é frequente e não existe um super padrão frequente contendo X. Ou seja, um padrão frequente X não pode ser um subpadrão de um padrão frequente maior para se qualificar para a definição de conjunto de itens máximo (GRAHNE; ZHU, 2003). Para melhor compreensão é ilustrado na Figura 15 um exemplo genérico. No modelo, os agrupamentos ACD, BCD e ABCD são considerados infrequentes, desse modo as combinações CD, ABC e ABD são conjuntos de itens máximo uma vez são os maiores agrupamentos possíveis e nenhuma das combinações é um subpadrão de outro agrupamento.

Figura 15 – Exemplo de Conjuntos de Itens Máximo



Fonte: autoria própria

2.6.1 Aplicações Práticas

SCHONHORST (2010) realizou um estudo sobre os dados transacionais de um supermercado num período de quatro meses. Os dados desse supermercado foram extraídos

via linguagem SQL, pré-processados e minerados pelo algoritmo Apriori onde, ao total, foram processadas 256 mil transações. Além dos parâmetros de suporte e confiança, o autor utilizou o parâmetro *lift*, o mesmo utilizado em nosso trabalho, para verificar a viabilidade das regras. Como resultado, foram identificadas várias associações relevantes. Porém o autor afirma que é necessário direcionar o foco do trabalho para um problema específico, pois as inúmeras relações tendem a dispersar o foco do especialista não visualizando algumas associações relevantes.

Já o autor Silva (2004) utilizou o métodos de regras de associação para comparar características físicas e socio econômicas aos procedimentos clínicos e de internações na cidade de Londrina-PR. Diferente da abordagem usual, foram processadas várias combinações entre dois, três e quatro atributos em cada etapa. Essa abordagem foi utilizada para reduzir o número explosivo de possíveis combinações dentro de cada execução. Através do método o autor conseguiu identificar erros processuais (erro de nomenclatura de procedimentos), confirmar hipóteses existentes além de localizar regras relevantes, por exemplo o tipo de hospital e o caráter das internações.

Uma outra possível utilização das regras de mineração é para identificar relações de comportamento de usuários dentro de páginas web. O autor Brusso (2000) utilizou informações extraídas de logs, pertencentes a páginas web de duas universidades, para extrair associações de acesso entre páginas. Uma das páginas analisadas, por exemplo, continha materiais referente a uma das matérias ministradas. Através do algoritmo Apriori, o autor conseguiu verificar se uma bibliografia complementar era utilizada ou não após o acesso da aula pelos alunos. Uma das dificuldades apontadas pelo autor é a quantidade de informações relevantes, onde o autor também sugere uma análise focal, e não genérica, do que se busca identificar.

3 ESTUDO DE CASO

Neste capítulo é apresentado a contextualização do problema, o modo de extração dos dados, a forma de processamento das informações, bem como os parâmetros escolhidos durante a execução do algoritmo.

3.1 Apresentação do Contexto

A empresa estudada é uma franqueadora, do ramo de produtos de beleza, que concede sua marca para vários franqueados em todo o território brasileiro. Tais produtos são vendidos em estabelecimento físico (loja) ou adquiridos através de seus revendedores.

A franqueadora adota, para ambos os segmentos, a estratégia de diversificação e massificação de lançamentos e promoções em todos os meses. Esta ação tem o objetivo despertar o interesse do consumidor nas novas promoções e incentivar a constante visitação de seus ambientes.

Em alguns meses, além dos produtos já comercializados pela franqueadora, são ofertados lançamentos como produtos de edição limitada, normalmente atrelados a algum desconto ou promoção. Os produtos podem ser exclusivos da franqueadora ou de edição limitada (tendo ou não parceria). Essas mercadorias têm o objetivo de atrair novos consumidores e testar novas tendências de mercado. Em alguns casos, caso haja boa recepção do público, tais produtos passam a fazer parte do portfólio oficial da franqueadora.

Nem todos os produtos, principalmente os de edição limitada, atingem a perspectiva estimada. Isso pode ocorrer devido a má previsão de demanda ou impacto direto da concorrência. Caso não sejam vendidos, as chances desses produtos permaneceram estocados por grandes períodos é grande, uma vez que as promoções e comunicações atreladas a essas mercadorias é extinta. Além disso, o franqueado não pode criar promoções próprias ou reposicionar os produtos nas prateleiras para favorecer um determinado produto, pois tais ações são de controle da franqueadora e envolvem questões de posicionamento de marca e de experiência do usuário.

Apenas alguma ações podem ser utilizadas pelo franqueado para dar vazão a produtos com pouco giro, entre elas: ações de fluxo (compre X reais de um determinado produto e concorra a brindes), bonificação de venda para as vendedoras de loja ou unificação de produtos em combos presenteáveis. A estratégia de combos, onde dois ou mais produtos são agrupados em uma embalagem, é bastante utilizada, uma vez que um combo pode alavancar itens que não são vendidos frequentemente e não precisam estar atrelados a uma vantagem financeira (GOMES; LIMA, 2021).

Neste estudo de caso, utilizou-se a abordagem de regras de associações para identificar oportunidades de combos presenteáveis para produtos super estocados em uma determinada loja. No entanto, algumas condições precisam ser impostas (algumas exigidas pela franqueadora

para proteção da marca e da sua identidade visual e outras para limitar o escopo do problema):

- Produtos de lançamento e edição limitada deverão estar agrupados exclusivamente com outros produtos da mesma marca;
- Produtos de categorias diferentes, porém da mesma marca, serão considerados válidos para formação do combo;
- O período analisado foi de doze meses (o intervalo de um ano reduz o viés das promoções que ocorrem a cada mês, o qual podem alterar o resultado de maneira geral). O período selecionado foi de 19/02/2021 até 18/02/2022;
- Apenas as vendas de uma das lojas do franqueado foram analisadas (como a franqueadora atua em todo o território nacional pode haver diferença nos padrões de consumo entre diferentes cidades e regiões);

3.2 Itens Críticos

Mesmo em grandes quantidades, nem todos os itens podem ser considerados críticos, assim como nem todos os produtos abaixo do estoque crítico podem ser considerados bons. Dessa maneira, foram determinadas duas medidas para avaliar e definir os itens que serão estudados (também chamados de itens críticos): estoque e cobertura.

3.2.1 Estoque

O estoque é um local onde ficam armazenados os produtos com sua devida acomodação e segurança, podendo estar localizado em uma sala, galpão ou centro-logístico. Ter poucas unidades de um determinado produto pode causar o que é conhecido como 'ruptura', onde o cliente busca determinado item, sendo que ele está em falta. Isto é crítico, pois a falta de um produto pode impedir a venda de mais itens. Em contrapartida, ter muitos produtos pode causar a sobrecarga de estoque, fazendo com que ocupem espaço de outros itens de maior vazão. Em nosso problema, o número máximo de itens em estoque, definido pela franqueadora, para a maioria dos itens é de cem unidades.

3.2.2 Cobertura

Cobertura é uma estimativa média de quantos dias o produto ficará em estoque, podendo ser calculada utilizando a Equação (7):

$$Cobertura = \frac{Quantidade\ de\ itens\ em\ estoque}{Quantidade\ média\ de\ itens\ vendidos\ por\ dia} \tag{7}$$

A 'quantidade média de itens vendidos por dia' normalmente é calculada dividindo a soma total das unidades vendidas em um ano pelo número de dias em que a loja ficou aberta por este mesmo período. Quando é lançamento, a soma dos itens vendidos é dividida pela quantidade de dias em que ele ficou disponível para venda. A cobertura ideal definida pela franqueadora é de 45-90 dias. Produtos com cobertura inferior a 45 dias devem ser solicitados

para reabastecimento e produtos com cobertura superior a 90 dias podem ser considerados como produtos super estocados.

Exemplo: se um item teve 600 unidades vendidas em um ano e a loja ficou aberta por 300 dias, logo ele tem uma média de venda de 2 itens/dia. Caso o estoque dessa loja seja de 100 unidades, logo ele terá estoque de 50 dias.

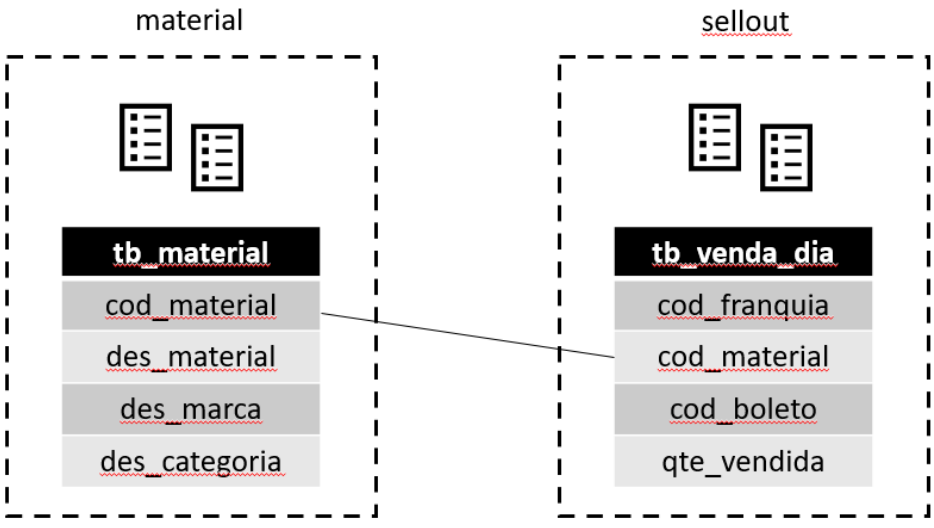
Neste trabalho foram considerados como itens críticos produtos com mais de 100 unidades em estoque e cobertura superior à 90 dias conforme sugestão da franqueadora.

3.3 Extração dos Dados

O sistema de pagamento disponível em todas as lojas do franqueado está conectado com o servidor próprio da franqueadora. Desta maneira, todos os dados de *sell out* a nível de pedido e produto estão disponíveis para consulta. Os dados históricos de venda são armazenados em servidores na nuvem do Google Cloud Platform (GCP) e podem ser consultados via *script* SQL na plataforma BigQuery (um dos aplicativos pertencentes ao GCP).

O esquema dos dados, referente as informações de pedidos, está representado na Figura 16, onde na hierarquia 'material' está localizada a tabela 'tb_material' que possui as informações relacionadas aos itens, e na hierarquia 'sellout' está localizada a tabela 'tb_venda_dia' que possui os dados relacionados as vendas.

Figura 16 – Estrutura dos Dados no GCP



Fonte: autoria própria

Dentre as várias informações disponíveis, foram extraídas apenas as características (*features*) que possuíam relevância para o problema, as informações são:

- Código do pedido (ou número da transação);
- Código e descrição do produto;
- Marca do produto;

- Categoria do produto;
- Quantidade de produtos vendidos.

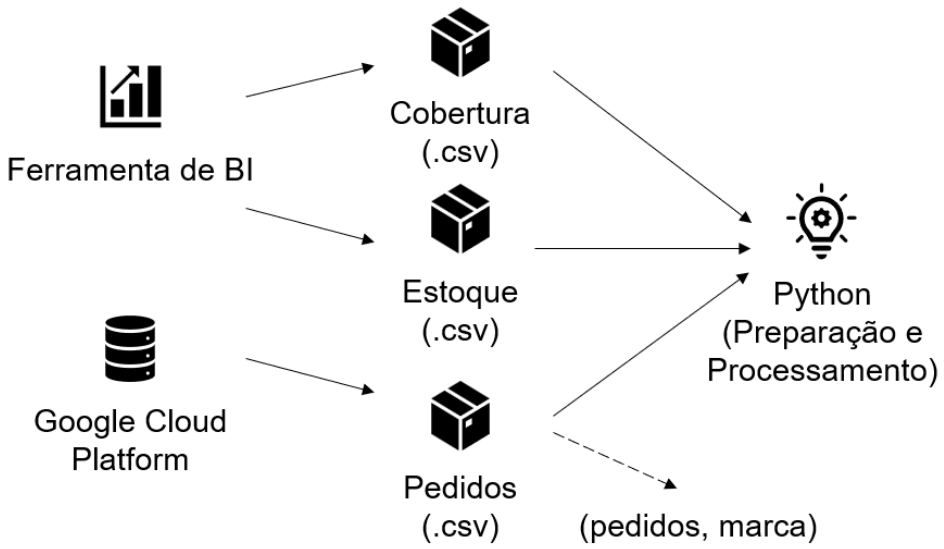
Importante ressaltar que a informação "quantidade de produtos vendidos" não é relevante para computar as regras de associação, uma vez que tal informação não se traduz em nenhuma informação relevante. Porém, os dados foram extraídos para validar se os dados de cobertura, extraídos do sistema, estavam corretos. Dados de categoria, descrição do material e código de loja também não são úteis para o algoritmo de regras de associação. No entanto, são úteis para a análise posterior do especialista.

Os dados de cobertura e estoque foram extraídos da plataforma de BI da empresa. Ambos os arquivos possuem o formato .csv contendo exclusivamente o código de cada item e a informação de estoque e cobertura do ponto de venda analisado.

3.4 Processamento dos Dados

O processamento das informações (cobertura, estoque e pedidos) foi realizado em linguagem Python via ambiente Google Colab. A linguagem foi escolhida uma vez que possui grande suporte da comunidade e pelo fato de possuir uma vasta gama de bibliotecas com várias ferramentas já implementadas para manipulação e tratamento de dados para ciência de dados. O modelo com as etapas da extração e processamento dos dados está presente na Figura 17.

Figura 17 – Modelo do Problema



Fonte: autoria própria

Dentre as várias opções de bibliotecas disponíveis, o Mlxtend (RASCHKA, 2018) foi escolhido, pois além de possuir todos os parâmetros e medidas já mencionadas consegue projetar visualmente bem as relações encontradas, facilitando a leitura e compreensão das informações.

Assim como exemplificado no trabalho, a biblioteca Mlxtend realiza a etapa de geração dos *itemsets* frequentes e das regras de associação em etapas distintas. Foi utilizado a função *fp-growth* para calcular as associações que respeitassem o suporte mínimo informado (definido no capítulo de resultados) e a função *association-rules* para calcular os parâmetros de confiança e a medida de interesse *lift*.

3.5 Avaliação das Regras

Além de gerar e filtrar as regras irrelevantes, a forma como essas regras serão utilizadas é de suma importância para a resolução do problema. Nosso contexto, mesmo se assemelhando a um problema de cesta de compras, possui particularidades, conforme apresentadas a seguir.

Em um mercado, na maioria das vezes, o estoque de um produto não é considerado, sendo o objetivo principal alavancar o número de itens vendidos por consumidor. Em nosso contexto, sabemos que os itens estudados são produtos com grandes quantidades em estoque e que não são vendidos frequentemente dado o alto valor de cobertura. Desse modo, é fácil deduzir que raramente estes itens são considerados a primeira opção de um consumidor em loja e que não são eles que alavancarão a venda, principalmente de um combo. Outro fator importante é que, durante os meses, há descontos de outros produtos que impulsionam a venda dos itens críticos estudados. Dessa maneira, é preciso estudar uma estratégia para cada situação encontrada em loja e verificar como utilizar cada regra dentro desses contextos.

A primeira situação, provavelmente a mais comum, é considerar que tanto o produto em questão quanto os demais itens de mesma marca não estão em promoção. Nessa condição, regras com altos valores de suporte (frequentes) são mais atrativas do que aquelas menos frequentes e com altos valores de confiança.

A segunda situação, sendo a mais desejável pela franquia, é que os itens considerados como críticos estejam em promoção. Em tais condições, normalmente a própria indústria informa quais os combos que devem ser montados, não sendo necessário nossa ação.

A terceira situação acontece quando um item que está em promoção possui relação de alavancagem com um dos itens críticos. Nessa condição particular, as regras de associação são relevantes, uma vez que o produto precedente está mais barato, existe uma maior chance do item crítico ser selecionado. Para esses casos, daremos preferência para as agregações com valores de confiança elevados, possibilitando assim combos mais eficientes. É importante destacar que uma regra possui dois valores de confiança, um para quando ele é precedente (ele alavanca a venda de outro material) e outro quando é consequente (quando outro item alavanca a venda do material analisado). Como queremos procurar os itens promocionados que alavancam as vendas dos produtos críticos, foram selecionadas as regras nas quais os produtos desejados estejam como consequente na regra (no lado direito da seta).

4 RESULTADOS

Primeiramente, foram extraídos os dados referentes as vendas realizadas para a loja específica do estudo, onde o período selecionado foi 19/02/2021 a 18/02/2022. A Figura 18 ilustra o script SQL utilizado na ferramenta de SQL presente no GCP. O resultado da extração resultou um arquivo CSV contendo 109.456 linhas ilustrado na Figura 19.

Figura 18 – Script Utilizado para Extração dos Dados de Venda

```
WITH MATERIAL AS
   (
   SELECT DISTINCT
   cod_material, des_material, des_marca_material, des_categoria_material
   FROM `material.tb_material`
   )

SELECT
cod_franquia, cod_boleto, CUPOM.cod_material, des_material, des_marca_material, des_categoria_material,
SUM(qte_vendida) AS qtde
FROM `sellout.tb_venda_dia` as CUPOM
INNER JOIN MATERIAL
ON MATERIAL.cod_material = CUPOM.cod_material
WHERE dt_venda BETWEEN '2021-02-19' AND '2022-02-18'
AND cod_franquia = 'CODIGO_XYZ'
GROUP BY
cod_franquia, cod_boleto, CUPOM.cod_material, des_material, des_marca_material, des_categoria_material
ORDER BY cod_boleto, CUPOM.cod_material
```

Fonte: autoria própria

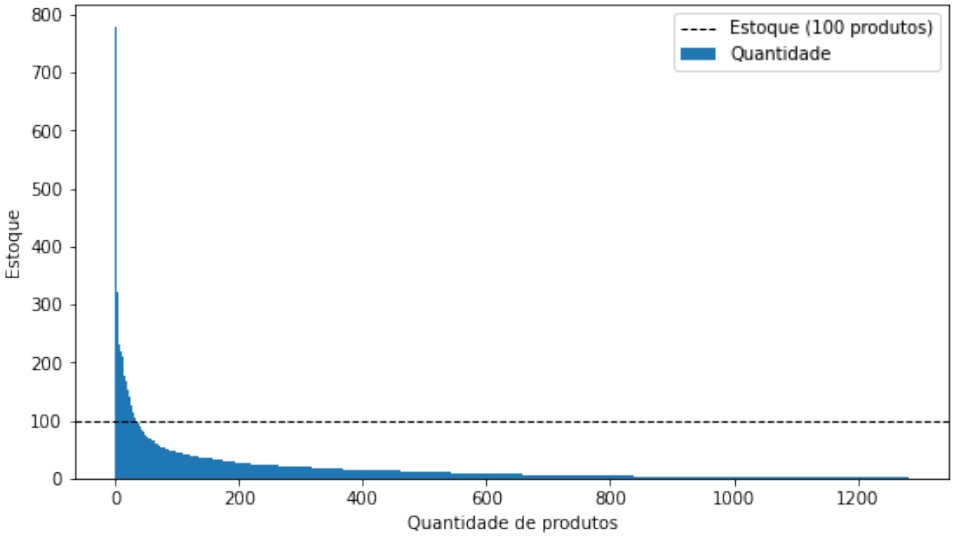
Figura 19 – Arquivo de Pedidos Extraído do GCP

	A	B	C	D	E	F	G
1	cod_franquia	cod_boleto	cod_material	des_material	des_marca	des_categoria	qte_vendida
2	5959	LOJ00000000004020210804000000595944	P0001	NOME GENÉRICO P0001	MARCA 01	PERFUMARIA	1
3	5959	LOJ00000000004020210804000000595944	P0723	NOME GENÉRICO P0723	MARCA 29	PERFUMARIA	2
4	5959	LOJ00000000009220210806000000595944	P0002	NOME GENÉRICO P0002	MARCA 02	PERFUMARIA	1
5	5959	LOJ00000000009220210806000000595944	P0003	NOME GENÉRICO P0003	MARCA 02	PERFUMARIA	1
6	5959	LOJ00000000009220210806000000595944	P0236	NOME GENÉRICO P0236	MARCA 12	PERFUMARIA	3

Fonte: autoria própria

As informações referente ao estoque atual da franquia foram extraídas de um dashboard já existente na companhia. O documento resultante foi um arquivo CSV contendo uma lista de 1.382 produtos distintos e a quantidade armazenada em estoque de cada material. Os produtos foram então ordenados conforme a sua quantidade sendo o resultado ilustrado na Figura 20. A linha tracejada na cor preta indica a divisa entre o número ideal de estoque e a super-estocagem definida pela franqueadora, e as barras azuis indicam as unidades disponíveis de cada produto. Itens a esquerda e acima da linha tracejada são os produtos que têm possibilidade de serem decretados críticos. Ao todo, apenas 42 produtos possuem uma quantidade em estoque superior a 100 unidades.

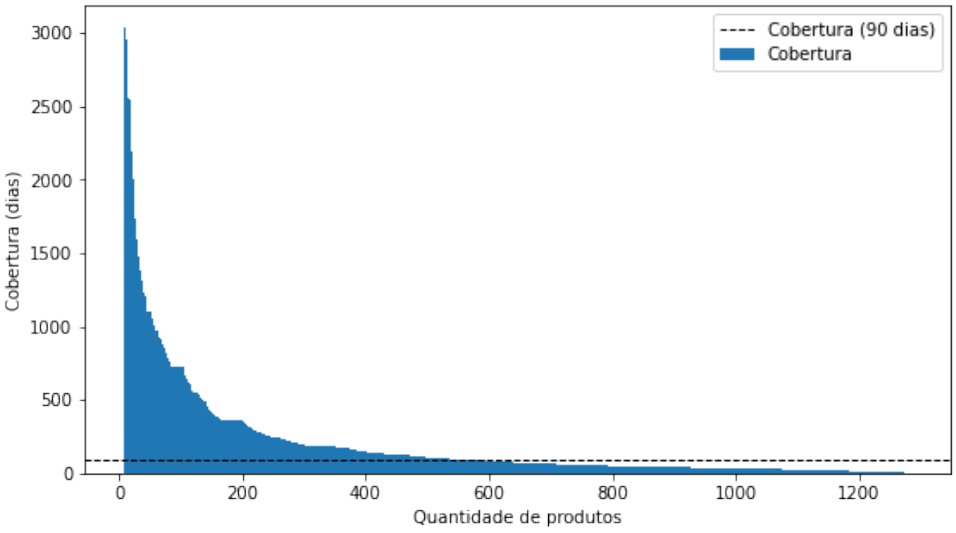
Figura 20 – Unidades em Estoque por Produto



Fonte: autoria própria

Assim como realizado com o estoque do franqueado, a abordagem de ranqueamento também foi aplicada para cobertura dos 1.382 itens. A Figura 21 faz menção ao parâmetro de cobertura onde a linha tracejada preta representa a cobertura ideal máxima recomendada pela franqueadora e as barras azuis, o valor numérico calculado para a cobertura de cada produto. Percebe-se que 620 produtos (quase 50% dos itens) possuem um estoque calculado superior a 90 dias, considerando a média de venda por dia de cada produto em um ano.

Figura 21 – Cobertura dos Produtos em Estoque



Fonte: autoria própria

Na Tabela 19 estão listados os produtos que foram localizados acima da zona crítica tanto de estoque como de cobertura e que são o escopo desse trabalho (denominados itens

críticos). É importante ressaltar que os itens tiveram suas descrições alteradas para assegurar a privacidade das informações da franqueadora.

Tabela 19 – Lista de Itens Críticos

Produto	Marca	Estoque	Cobertura
P0001	Marca 01	176	1.329
P0002	Marca 02	123	976
P0003	Marca 02	106	790
P0004	Marca 03	104	654
P0005	Marca 01	178	605
P0006	Marca 04	150	372
P0007	Marca 05	142	250
P0008	Marca 06	320	218
P0009	Marca 07	127	184
P0010	Marca 08	224	142
P0011	Marca 01	113	125
P0012	Marca 09	153	115
P0013	Marca 10	103	110

Fonte: autoria própria

Com o objetivo de avaliar a oportunidade de sugestão de combos, foram calculadas todas as possíveis combinações candidatas por marca com diferentes valores de suporte (não possível é calcular associações entre todos os produtos, pois é uma exigência da franqueadora que combos contenham itens da mesma marca). Na Tabela 20, é mostrado o número de combinações candidatas entre diferentes valores de suporte por marca.

Tabela 20 – Combinações Candidatas por Suporte

Marca	Transações	Suporte = 5%	Suporte = 1%	Suporte = 0.5%
Marca 01	1.562	0	11	19
Marca 02	3.313	0	0	0
Marca 03	4.946	0	4	12
Marca 04	10.356	0	0	8
Marca 05	5.026	0	2	7
Marca 06	287	0	0	1
Marca 07	3.634	0	5	10
Marca 08	2.852	0	5	7
Marca 09	11.263	0	6	12
Marca 10	1.727	0	8	21

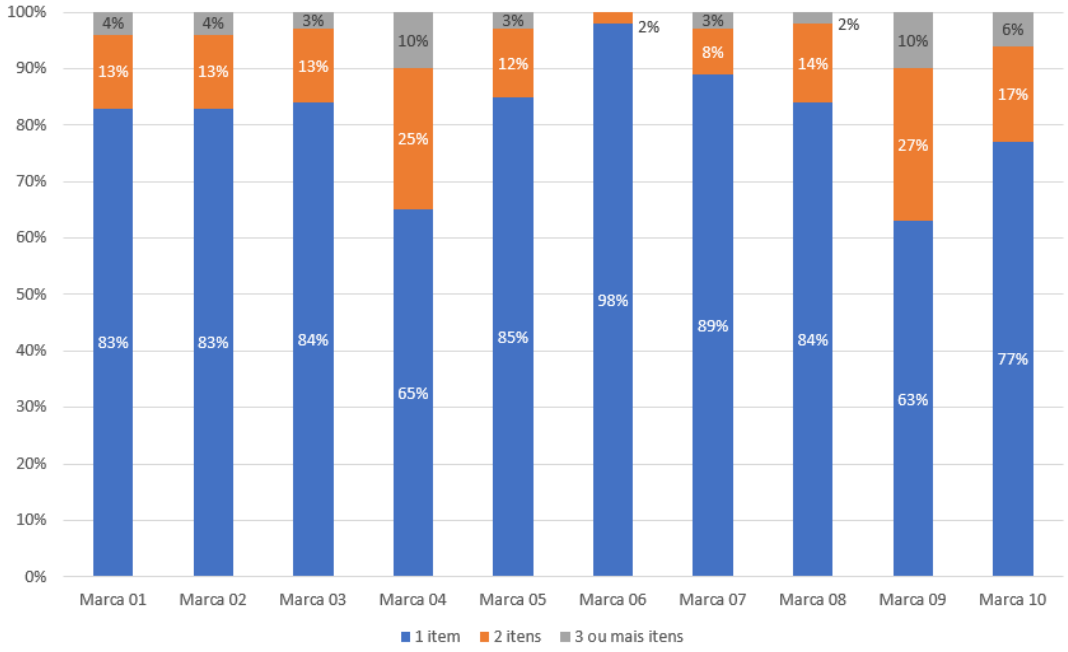
Fonte: autoria própria

Analisando a Tabela 20 é possível verificar que não existe nenhuma regra de associação encontrada com frequência maior que 5%, mas que é possível encontrar combinações candidatas

a partir de 1% em algumas situações. Isso é preocupante uma vez que regras mesmo interessantes, mas que com baixas frequências, podem não ser relevantes para trazer o volume necessário de vendas e reduzir a cobertura dos itens considerados críticos.

A Figura 22 complementa a visão, nos mostrando o percentual de transações com 1, 2 ou mais itens distintos por marca. Observando a tabela juntamente com o gráfico podemos deduzir que o número baixo de combinações candidatas pode ser reflexo de que o consumidor, na sua maioria, prefere não comprar mais de um item da mesma marca. Outra razão para esse comportamento pode ser o preço dos produtos, porém é preciso de um estudo mais detalhado para avaliar o impacto dos preços na venda dos produtos.

Figura 22 – Percentual de Produtos por Transação da Mesma Marca



Fonte: autoria própria

É importante notar que as Marcas 04 e 09 possuem um perfil distinto das demais, ambas possuem um número mais expressivo tanto em percentual de pedidos com mais de dois produtos como também em número de transações (>10.000). Este fator pode estar associado com a categoria em que essas marcas estão inseridas ou com o valor desses produtos.

Com base nessas percepções, são realizadas duas abordagens: a primeira onde é utilizado um suporte mínimo de 0,5% para os itens pertencentes as Marcas 04 e 09 e outra, com um suporte de 1%, para os itens das marcas restantes. Tal abordagem se deve ao fato de que associações com frequências inferiores a 1%, para itens com menos de 10.000 transações, gerariam regras com aparecimentos extremamente baixos (considerando que em média cada marca possuiu aproximadamente 4.497 transações, teríamos combinações que apareceriam no máximo 45 vezes ao ano com o suporte mínimo de 1%). No entanto para marcas com mais de 10.000 transações, e que possuem um histórico de maior número de associações, é possível

reduzir este suporte.

4.1 Suporte Mínimo de 0,5%

Após o processamento dos dados, foram encontradas apenas dois *itemsets* frequentes para os itens críticos P0006 e P0012. No entanto, nenhum *itemset* possui mais de um produto, o que significa que não há agrupamentos de dois ou mais produtos com frequência superior ao suporte mínimo informado. Apenas os itens críticos individualizados possuem frequência superior a 0,5% (fator que impossibilita a geração de regras de associação). As combinações candidatas encontradas são apresentadas na Tabela 21.

Tabela 21 – *Itemsets* Frequentes com Suporte de 0,5%

Itemsets Frequentes	Suporte
P0006	0.0121
P0012	0.0107

Fonte: autoria própria

4.2 Suporte Mínimo de 1%

Para cada um dos itens críticos (exceto dos produtos P0006 e P0012 que já foram processados anteriormente) são calculados os *itemsets* frequentes que respeitem o suporte mínimo de 1%. A lista que contem todas as associações encontradas está presente na Tabela 22. Nesta lista, percebe-se que apenas os produtos P0011 e P0013 possuem *itemsets* frequentes com mais de um produto, ou seja, apenas os dois itens podem produzir regras de associação.

As regras, que respeitam a confiança mínima de 1%, foram calculadas no ambiente Google Colab através da biblioteca *mlxtend* em Python. A Figura 23 ilustra o resultado obtido pelo usuário após o processamento neste ambiente, tendo o resultado completo, de todas as regras, apresentado na Tabela 24

Figura 23 – Output dos Dados da Biblioteca Mlxtend

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift
0	(MATERIAL 0011)	(MATERIAL 0689)	0.165919	0.311980	0.040999	0.247104	0.792053
1	(MATERIAL 0689)	(MATERIAL 0011)	0.311980	0.165919	0.040999	0.131417	0.792053
6	(MATERIAL 0685)	(MATERIAL 0684)	0.079436	0.065983	0.033312	0.419355	6.355465
7	(MATERIAL 0684)	(MATERIAL 0685)	0.065983	0.079436	0.033312	0.504854	6.355465
9	(MATERIAL 0684)	(MATERIAL 0011)	0.065983	0.165919	0.017937	0.271845	1.638415
8	(MATERIAL 0011)	(MATERIAL 0684)	0.165919	0.065983	0.017937	0.108108	1.638415
3	(MATERIAL 0685)	(MATERIAL 0011)	0.079436	0.165919	0.014734	0.185484	1.117916
2	(MATERIAL 0011)	(MATERIAL 0685)	0.165919	0.079436	0.014734	0.088803	1.117916
18	(MATERIAL 0665)	(MATERIAL 0689)	0.037156	0.311980	0.012172	0.327586	1.050025
10	(MATERIAL 0689)	(MATERIAL 0684)	0.311980	0.065983	0.012172	0.039014	0.591276
11	(MATERIAL 0684)	(MATERIAL 0689)	0.065983	0.311980	0.012172	0.184466	0.591276

Fonte: autoria própria

Tabela 22 – *Itemsets* Frequentes com Suporte de 1%

Itemsets Frequentes	Suporte
P0001	0.0256
P0002	0.0159
P0003	0.0138
P0004	0.0117
P0005	0.0377
P0007	0.0401
P0008	0.3135
P0009	0.0742
P0010	0.0410
P0011	0.1658
P0011, P0689	0.0409
P0011, P0684	0.0179
P0011, P0685	0.0147
P0011, P0684, P0685	0.0108
P0011, P0665	0.0108
P0013, P0597	0.1910
P0013, P0597	0.0225
P0013, P0641	0.0162
P0013, P1466	0.0121
P0013, P1429	0.0110
P0013, P0598	0.0104

Fonte: autoria própria

Ao analisar a Tabela 24, observa-se que para o produto P0011 foram encontradas 14 regras de associação. A regra {P0011, P0685 ⇒ P0684}, por exemplo, em primeiro momento pode ser a mais desejável, visto que possui uma confiança de 0.7391. No entanto, percebe-se que o produto P0011 está inserido como precedente. O item ser precedente significa que o produto P0011 tem a probabilidade de alavancar a venda de P0684, o que no nosso caso não é desejado. Tal informação seria valiosa caso o produto P0011 tivesse alta demanda, o que possivelmente alavancaria a venda do P0684. Porém, como queremos alavancar a venda do P0011, a perspectiva de antecedente não é desejada, mas sim a de consequente (onde outro material tem a probabilidade de alavancar a venda do produto que queremos).

Quando olhamos na perspectiva de consequente, a regra {P0684, P0685 ⇒ P0011} possui uma confiança de 0.3269 e um suporte de 0.0109. Tal detalhe nos mostra que em 32,7% das vezes que o comprador adquirir os produtos P0684 e P0685, há a possibilidade dele comprar o produto P0011. Essa informação é relevante, pois caso os produtos P0684 e P0685 apareçam em promoção, estes alavancarão indiretamente a compra do P0011. Outras regras como {P0684 ⇒ P0011, P0685} e {P0685 ⇒ P0011, P0684} também são recomendadas nos casos de promoção do precedente.

Para as associações de apenas dois produtos, as regras {P0684 ⇒ P0011} e {P0685 ⇒

P0011} são as indicadas para exposição recorrente (onde os produtos não estão promocionados). Isso se deve pelo de ser a regra mais frequente e ocorrer em 1,8% das transações. É importante destacar que a regra {P0689 ⇒ P0011}, apesar de ser mais frequente, incidindo em 4,1% das transações, possui o valor de *lift* menor que um, o que indica que a associação acontece ao acaso e não possui relação real. A regra {P0665 ⇒ P0011} possui confiança de 0.2881 e é aconselhada quando o precedente está promocionado.

O produto P0013 possui 10 regras encontradas, onde cinco regras são como antecedente (não nos interessa) e cinco como consequente, onde, diferente do produto P0011, não foram encontradas regras com mais de dois produtos associados.

Podemos observar na Tabela 24 que das cinco regras (onde o produto P0013 esta como consequente), apenas a regra {P1466 ⇒ P0013} nos é interessante, uma vez que as demais regras não possuem associação real (*lift* $<= 1$). Desse modo, a associação {P1466 ⇒ P0013}, que possui suporte de 0.0122 e confiança de 0.2308, é recomendada para o uso de exposição frequente e para os casos em que o produto P1466 é promocionado.

Na Tabela 23 está ilustrado o resultado final com as regras que obedecem o *lift* e o suporte mínimo e o tipo de recomendação sugerida para cada associação.

Tabela 23 – Resultado Final

Produto	Regra	Suporte	Confiança	Lift	Recomendação
P0013	{P1466 ⇒ P0013}	0,012	0,230	1,207	Recorrente
P0011	{P0684 ⇒ P0011}	0,017	0,271	1,639	Recorrente
P0011	{P0685 ⇒ P0011}	0,014	0,185	1,118	Recorrente
P0011	{P0665 ⇒ P0011}	0,010	0,288	1,737	Promoção
P0011	{P0685 ⇒ P0011, P0684}	0,010	0,137	7,648	Promoção
P0011	{P0684 ⇒ P0011, P0685}	0,010	0,165	11,208	Promoção
P0011	{P0684, P0685 ⇒ P0011}	0,010	0,326	1,971	Promoção

Fonte: autoria própria

Tabela 24 – Regras de Associação dos Itens Críticos

Antecedente	Consequente	Sup. Antecedente	Sup. Consequente	Suporte	Confiança	Lift
P0011	P0689	0,1658	0,3118	0,0410	0,2471	0,7925
P0011	P0684	0,1658	0,0659	0,0179	0,1081	1,6394
P0011	P0685	0,1658	0,0794	0,0147	0,0888	1,1186
P0011	P0665	0,1658	0,0378	0,0109	0,0656	1,7377
P0011, P0685	P0684	0,0147	0,0659	0,0109	0,7391	11,2089
P0011	P0684, P0685	0,1658	0,0333	0,0109	0,0656	1,9716
P0011, P0684	P0685	0,0179	0,0794	0,0109	0,6071	7,6480
P0689	P0011	0,3118	0,1658	0,0410	0,1314	0,7925
P0684	P0011	0,0659	0,1658	0,0179	0,2718	1,6394
P0685	P0011	0,0794	0,1658	0,0147	0,1855	1,1186
P0665	P0011	0,0378	0,1658	0,0109	0,2881	1,7377
P0685	P0011, P0684	0,0794	0,0179	0,0109	0,1371	7,6480
P0684	P0011, P0685	0,0659	0,0147	0,0109	0,1650	11,2089
P0684, P0685	P0011	0,0333	0,1658	0,0109	0,3269	1,9716
P0013	P0597	0,1911	0,1621	0,0226	0,1182	0,7289
P0013	P0641	0,1911	0,1100	0,0162	0,0848	0,7712
P0013	P1466	0,1911	0,0527	0,0122	0,0636	1,2076
P0013	P1429	0,1911	0,0625	0,0110	0,0576	0,9206
P0013	P0598	0,1911	0,0620	0,0104	0,0545	0,8803
P0597	P0013	0,1621	0,1911	0,0226	0,1393	0,7298
P0641	P0013	0,1100	0,1911	0,0162	0,1474	0,7712
P1466	P0013	0,0527	0,1911	0,0122	0,2308	1,2076
P1429	P0013	0,0625	0,1911	0,0110	0,1759	0,1759
P0598	P0013	0,0620	0,1911	0,0104	0,1682	0,1682

Fonte: autoria própria

5 CONCLUSÃO

Neste trabalho utilizou-se uma abordagem conhecida na literatura para identificar associações entre mercadorias de menor e maior fluxo para sugerir combos presenteáveis atrativos com o objetivo e reduzir produtos super estocados. A abordagem constituiu na análise de dados históricos de venda de uma determinada loja para identificar associações relevantes destes itens. Os produtos críticos foram identificados através de critérios impostos pela franqueadora e a construção das regras de associação foi efetuada com ajuda de uma implementação do Fp-Growth via linguagem Python.

Nos resultados, percebeu-se que as marcas (associadas aos produtos críticos) não apresentaram associações com frequência superior a 5%. Tal resultado está relacionado ao fato de que em média 80% das transações possuem apenas um material por transação (Figura 22). Desse modo, todas as possíveis combinações estão contempladas em um universo de apenas 20%, sendo o suporte de 1% o mais adequado visto a relação de regras encontradas e frequência esperada.

Dos treze itens críticos, apenas os produtos P0011 e P0013 apresentaram resultados que justificassem a estratégia de combos presenteáveis (Tabela 24). Para o produto P0011 a regra $\{P0684 \Rightarrow P0011\}$ é a recomendada para ser o combo frequente em loja (devido ao seu valor de suporte), enquanto as regras $\{P0011, P0685 \Rightarrow P0684\}$, $\{P0665 \Rightarrow P0011\}$ e $\{P0684 \Rightarrow P0011\}$ são recomendadas em caso de promoção dos antecedentes. Para o produto P0013, apenas a associação $\{P1466 \Rightarrow P0013\}$ se mostrou interessante para ambos os contextos.

A medida de interesse *lift*, se mostrou importante para o direcionamento do resultado, uma vez que se concluiu que muitas das regras aparentemente úteis são na verdade aleatórias e não possuem utilidade real. Em nosso contexto, a medida indicou que uma das regras do produto P0011 e quatro regras do produto P0013 são estatisticamente irrelevantes mesmo com ótimos valores de suporte e confiança.

O trabalho se mostrou valoroso dentro de sua proposta. Dos treze produtos selecionados como críticos, foi possível identificar catorze combinações relevantes para dois deles. Além disso, foi possível definir associações relevantes em períodos de promoção e não-promoção para ambos. De modo geral, tal metodologia se mostrou bastante eficiente, uma vez que performou bem para uma base de dados relativamente grande (acima de 100 mil linhas), sendo também bastante replicável e de fácil compreensão (não exige grandes conhecimentos estatísticos).

5.1 Limitações

O intervalo de um ano foi escolhido pelo fato de que, todas as marcas e mercadorias são promocionadas em algum momento dentro desse período, fazendo com que não houvesse um viés se fosse considerado um período menor (por exemplo, se fosse escolhido um período

de seis meses, algumas marcas poderiam ter sido promocionadas e outras não, alterando assim os resultados). É sabido que algumas mercadorias alavancam melhor que outras quando promocionadas. Desse modo, pode haver diferenciação no ranking das associações em períodos em que o produto está ou não promocionado.

Neste trabalho, também não foram aprofundados temas referentes ao preço pela integridade dos dados da franqueadora. Porém, assim como as promoções, o preço do produto é um grande fator de alavancagem das mercadorias. Materiais que não alcançaram o suporte mínimo provavelmente sofrem as consequências de preços mal dimensionados.

5.2 Trabalhos Futuros

Em pesquisas futuras, é aconselhável investigar se os resultados obtidos são focais (apenas servem para a resolução do problema desta loja) ou se podem ser expandidos para as demais regiões: há regionalização? Fatores como preço, período e tipo de desconto também podem contribuir nas análises futuras com o intuito de obter combos mais efetivos entre os diferentes ciclos.

Referências

AGRAWAL, R.; IMIELINSKI, T.; SWAMI, A. Mining association rules between sets of items in large databases. In: **Proceedings of the 1993 ACM SIGMOD international conference on Management of data**. Washington: DC, 1993. p. 207–216. Citado 3 vezes nas páginas 11, 19 e 22.

AGRAWAL, R.; SRIKANT, R. et al. Fast algorithms for mining association rules. In: CITESEER. **Proc. 20th int. conf. very large data bases, VLDB**. Santiago, Chile, 1994. v. 1215, p. 487–499. Citado 2 vezes nas páginas 34 e 35.

ASSUNçãO, A. S. d. **Descoberta direta e eficiente de regras de associação ótimas**. Tese (Doutorado) — Universidade de São Paulo, 2011. Citado 2 vezes nas páginas 21 e 32.

BORGELT, C.; KRUSE, R. Induction of association rules: Apriori implementation. In: SPRINGER. **Compstat**. Berlin, Germany, 2002. p. 395–400. Citado 2 vezes nas páginas 21 e 26.

BRIN, S. et al. Dynamic itemset counting and implication rules for market basket data. In: **Proceedings of the 1997 ACM SIGMOD international conference on Management of data**. Tucson: Arizona, 1997. p. 255–264. Citado na página 32.

BRUSSO, M. J. Access miner: uma proposta para a extração de regras de associação aplicada à mineração do uso da web. 2000. Citado na página 37.

DAS, A.; NG, W.-K.; WOON, Y.-K. Rapid association rule mining. In: **Proceedings of the tenth international conference on Information and knowledge management**. Atlanta, GA, USA: ACM, 2001. p. 474–481. Citado na página 35.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. **AI magazine**, v. 17, n. 3, p. 37–37, 1996. Citado 2 vezes nas páginas 16 e 18.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. The kdd process for extracting useful knowledge from volumes of data. **Communications of the ACM**, ACM New York, NY, USA, v. 39, n. 11, p. 27–34, 1996. Citado 3 vezes nas páginas 14, 16 e 18.

FELDENS, M.; CITOLIN, I.; FRIGERI, S. Metodologias para implementação da inteligência do negócio: desenvolvimento de sistema de informação para database marketing. **Caxias do Sul: Revista do CCET**, 1999. Citado 2 vezes nas páginas 11 e 19.

GARCíA, S.; LUENGO, J.; HERRERA, F. **Data preprocessing in data mining**. Heidelberg, New York: Springer, 2015. v. 72. Citado 2 vezes nas páginas 14 e 15.

GOMES, P. H. V.; LIMA, G. A. de. A atuação da controladoria na formaçõ do preço de venda com base no mercado: um estudo de caso em uma distribuidora de fortaleza/ce. **Gestão Contemporânea**, v. 11, n. 2, p. 54–72, 2021. Citado na página 38.

GRAHNE, G.; ZHU, J. Efficiently using prefix-trees in mining frequent itemsets. In: **Proceeding of the ICDM'03 international workshop on frequent itemset mining implementations**. Melbourne, FL: FIMI, 2003. v. 90, p. 65. Citado na página 36.

HAN, J.; PEI, J.; KAMBER, M. **Data mining: concepts and techniques**. 3. ed. 225 Wyman Street, Waltham, MA 02451, USA: Elsevier, 2011. ISBN 9780123814791. Citado na página 14.

HAN, J.; PEI, J.; YIN, Y. Mining frequent patterns without candidate generation. **ACM sigmod record**, ACM New York, NY, USA, v. 29, n. 2, p. 1–12, 2000. Citado na página 22.

HAN, J. et al. Mining frequent patterns without candidate generation: A frequent-pattern tree approach. **Data mining and knowledge discovery**, Springer, v. 8, n. 1, p. 53–87, 2004. Citado na página 26.

ICHI.PRO. **A priori: explicação detalhada sobre mineração de regras de associação e implementação de Python**. 2020. Disponível em: <https://bityli.com/YITJFO>. Acesso em: 12 de dezembro de 2021. Citado na página 22.

JURGOVSKY, J. et al. Sequence classification for credit-card fraud detection. **Expert Systems with Applications**, Elsevier, v. 100, p. 234–245, 2018. Citado na página 16.

KHURANA, K.; SHARMA, S. A comparative analysis of association rule mining algorithms. **International Journal of Scientific and Research Publications**, Citeseer, v. 3, n. 5, p. 0, 2013. Citado na página 35.

KUMBHARE, T. A.; CHOBE, S. V. An overview of association rule mining algorithms. **International Journal of Computer Science and Information Technologies**, Citeseer, v. 5, n. 1, p. 927–930, 2014. Citado 2 vezes nas páginas 19 e 26.

LAVôR, R. M. de P. **Implementação de serviços relacionados à mineração de regras de associação**. Tese (Doutorado) — Master's thesis, Universidade Federal do Rio de Janeiro, Instituto de Matemática, Núcleo de Computação Eletrônica, 2003. Citado 8 vezes nas páginas 11, 16, 17, 18, 22, 23, 31 e 32.

OZAKI, T. J. **What kind of decision boundaries does Deep Learning (Deep Belief Net) draw? Practice with R and h2o package**. 2015. Disponível em: <https://tjo-en.hatenablog.com/entry/2015/02/15/194003>. Acesso em: 10 de abril de 2022. Citado na página 17.

PANG-NING, T.; STEINBACH, M.; KUMAR, V. Introduction to data mining: Pearson addison wesley boston. 2006. Citado na página 21.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011. Citado na página 18.

RASCHKA, S. Mlxtend: Providing machine learning and data science utilities and extensions to python's scientific computing stack. **The Journal of Open Source Software**, The Open Journal, v. 3, n. 24, abr. 2018. Disponível em: <http://joss.theoj.org/papers/10.21105/joss.00638>. Citado na página 41.

RASCHKA, S. **Frequent Itemsets via the FP-Growth Algorithm**. 2020. Disponível em: <http://rasbt.github.io/mlxtend/user_guide/frequent_patterns/fpgrowth/>. Acesso em: 09 de dezembro de 2021. Citado na página 26.

REINSEL, D.; RYDNING, J.; GANTZ, J. F. Worldwide global datasphere forecast, 2021–2025: The world keeps creating more data—now, what do we do with it all. **IDC Corporate USA**, 2021. Citado na página 14.

SCHONHORST, G. B. Mineração de regras de associação aplicada à modelagem dos dados transacionais de um supermercado. 2010. Citado na página 36.

SILVA, G. C. Mineração de regras de associação aplicada a dados da secretaria municipal de saúde de londrina pr. 2004. Citado na página 37.

SILVA, V. G. M. et al. Controle de estoque: um estudo sobre a eficiência da gestão de estoque numa distribuidora atacadista em divinópolis, mg. **Research, Society and Development**, Grupo de Pesquisa Metodologias em Ensino e Aprendizagem em Ciências, v. 7, n. 5, p. e575152, 2018. Citado na página 11.

SLACK, N. et al. **Administração da produção**. São Paulo-SP: Atlas, 2009. v. 2. Citado na página 11.

VASCONCELOS, L. M. R. de; CARVALHO, C. L. de. Aplicação de regras de associação para mineração de dados na web. **Revista Telfract**, v. 1, n. 1, 2018. Citado na página 20.

VELOSO, M. J. P. S. A. Regras de associação aplicadas a um método de apoio ao planeamento de recursos humanos. Faculdade de Economia da Universidade do Porto, 2004. Citado na página 11.

VERMA, A. K.; PAL, S.; KUMAR, S. Classification of skin disease using ensemble data mining techniques. **Asian Pacific journal of cancer prevention: APJCP**, Shahid Beheshti University of Medical Sciences, v. 20, n. 6, p. 1887, 2019. Citado na página 17.

VICENTE, M. J.; POLETTO, A. S. R. d. S. O papel da mineração de dados no contexto de big data e ciência de dados. Fundação Educacional do Município de Assis, 2020. Citado na página 15.

WEBB, G. I.; ZHANG, S. K-optimal rule discovery. **Data Mining and Knowledge Discovery**, Springer, v. 10, n. 1, p. 39–79, 2005. Citado na página 31.

XING, Z.; PEI, J.; KEOGH, E. A brief survey on sequence classification. **ACM Sigkdd Explorations Newsletter**, ACM New York, NY, USA, v. 12, n. 1, p. 40–48, 2010. Citado na página 16.

ZHAO, Q.; BHOWMICK, S. S. Association rule mining: A survey. **Nanyang Technological University, Singapore**, v. 135, 2003. Citado na página 36.